User and Reference Manual
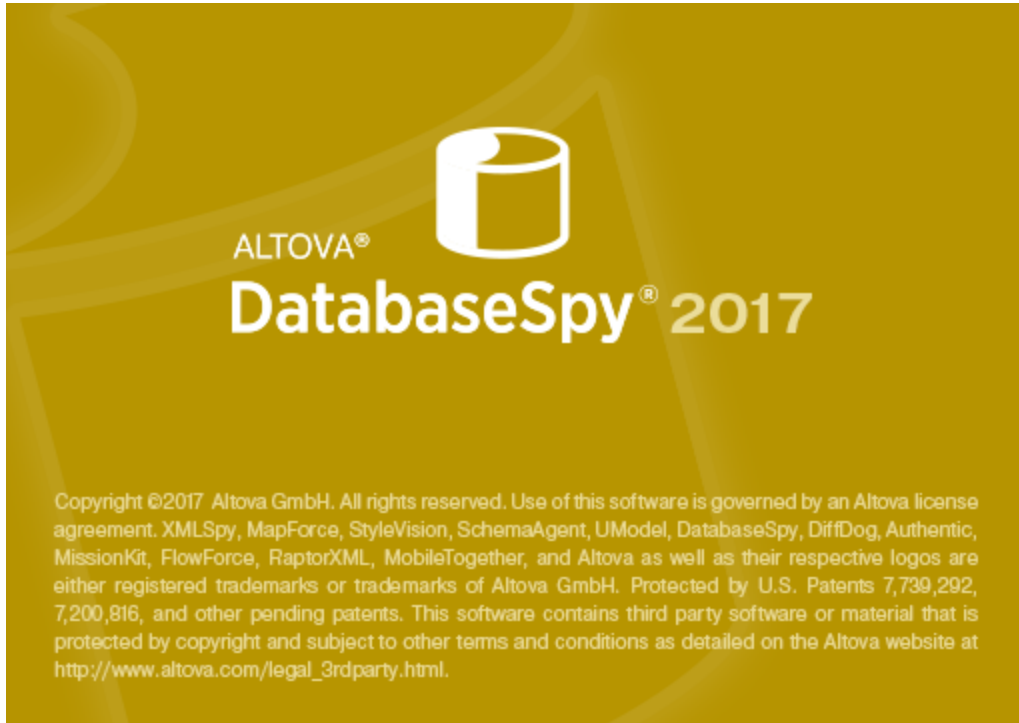
Altova DatabaseSpy 2017 User & Reference Manual

Published: 2017

© 2017 Altova GmbH

Table of Contents

6 Browsing Data Sources   252

7 Designing Databases   268

8 Retrieving and Editing Data   278

16   User Reference   440

Chapter 1

Altova DatabaseSpy 2017

1 Altova DatabaseSpy 2017

Altova DatabaseSpy 2017 Professional Edition is a Windows database client application with a consistent and easy-to-use interface that simplifies querying, visualizing, managing, comparing, designing, and constructing relational databases. It allows database administrators, database developers, and other database-oriented IT professionals seamless access to all the databases in their organization and is uniquely beneficial when you need to interact with multiple databases created at different times by different developers.
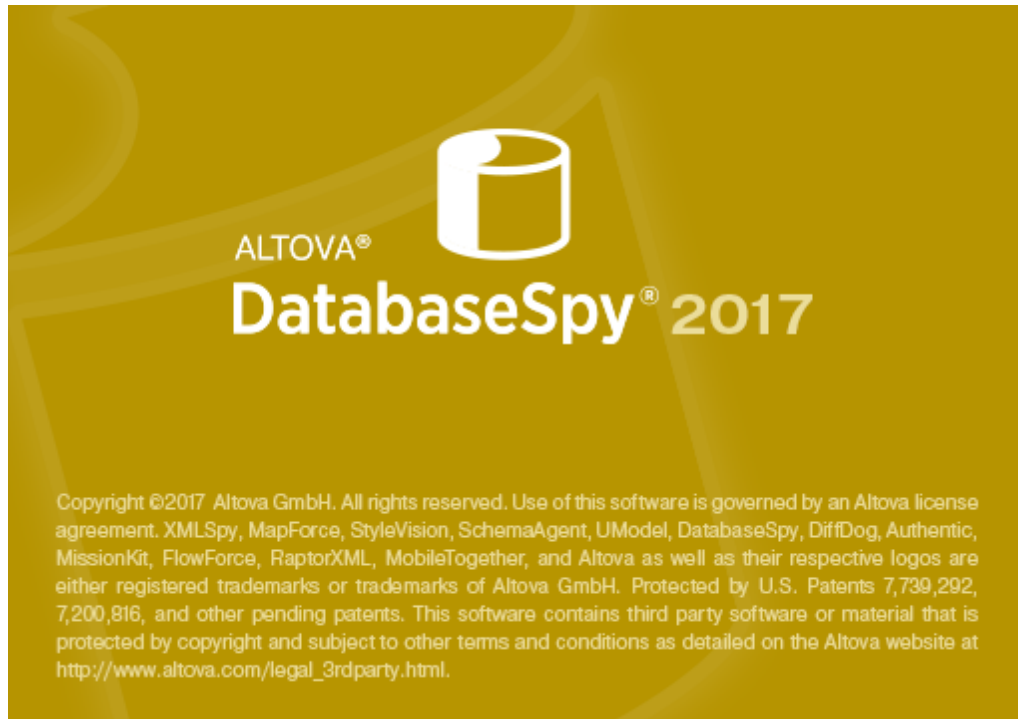
Last updated: 31 March 2017

Chapter 2

Introduction

2   Introduction

This User Manual contains tutorials and explanations of the various DatabaseSpy features to help you get started. It also contains a comprehensive reference section that describes DatabaseSpy features in detail. The User Manual consists of the following sections:

- An Introduction, which lists system requirements and the supported databases.
- A description of the of the DatabaseSpy interface
- A Tutorial section that shows you how to quickly get started with DatabaseSpy and makes you familiar with DatabaseSpy's features.
- A description of how to use projects to organize important elements of your database projects such as Global Resources, database connections, SQL files, and database design files.
- A detailed description of the Online Browser as well as how to browse data sources and locate database objects.
- A description of the Design Editor which can be used to graphically maintain the structure and properties of your databases.
- A description of how to retrieve and edit data using the SQL Editor.
- Descriptions of how you can work with SQL scripts in the SQL Editor.
- A description of how to compare data in database tables and merge the content of two compared tables as well as how to compare the structure of database schemas
- A description of how to maintain XML data in DatabaseSpy.
- A description of how to import data from XML and CSV files into your databases.
- A description of how to export data from your databases to files of the formats XML, XML Structure, CSV, HTML, and Excel.
- A detailed descriptions of the various options you can set in DatabaseSpy.
- A User Reference that contains a description of database objects and how they are created in DatabaseSpy as well as a description of all menu commands available in DatabaseSpy.

2.1   **File Paths**

File paths given in this documentation will not be the same for all operating systems. Note the following correspondences:

- *(My) Documents folder:* Example files are located in the **Altova** sub-folder of the **(My) Documents** folder.

Windows 10 Windows 8 Windows 7 Windows Vista Windows Server 2008 Windows Server 2012 Windows Server 2016	`C:\Users\<username>\Documents`
Windows XP Windows Server 2003	`C:\Documents and Settings\<username>\My Documents`

- *Application folder:* The Application folder is the folder where your Altova application is located (by default, `C:/Program Files/Altova`). When DatabaseSpy 32-bit is installed on a 64-bit operating system, the path is `C:/Program Files (x86)/Altova`.

2.2   System Requirements

DatabaseSpy runs on the following operating systems:

- Windows XP/Vista, Windows 7/8/10
- Windows Server 2003/2008/2012/2016

Both the 32-bit and 64-bit platforms are supported.

2.3   Supported Databases in DatabaseSpy

The following databases are supported. The available root object for each database is also listed. While Altova endeavors to support other databases, successful connection and data processing have only been tested with the databases listed below. If your Altova application is a 64-bit version, ensure that you have access to the 64-bit database drivers needed for the specific database you are connecting to.

Database	Root Object	Notes
Firebird 2.5.4	database	
IBM DB2 8.x, 9.1, 9.5, 9.7, 10.1, 10.5	schema	
IBM DB2 for i 6.1, 7.1	schema	Logical files are supported and shown as views.
IBM Informix 11.70	database	
Microsoft Access 2003, 2007, 2010, 2013	database	
Microsoft Azure SQL Database	database	SQL Server 2016 codebase
Microsoft SQL Server 2005, 2008, 2012, 2014, 2016	database	
MySQL 5.0, 5.1, 5.5, 5.6	database	
Oracle 9i, 10g, 11g, 12c	schema	
PostgreSQL 8.0, 8.1, 8.2, 8.3, 9.0.10, 9.1.6, 9.2.1, 9.4, 9.5	database	PostgreSQL connections are supported both as native connections and driver-based connections through interfaces (drivers) such as ODBC or JDBC. Native connections do not require any drivers.
Progress OpenEdge 11.6	database	With Progress OpenEdge databases, there may be cases where a table cannot be deleted using the SQL script generated by DatabaseSpy. The workaround is to drop the table by entering a DROP TABLE statement in the SQL Editor window, instead of running the generated Database Structure Change Script. The limitation originates in the fact that, in Progress OpenEdge, you cannot drop the first index created on the table, except by dropping the table. DatabaseSpy, on the other hand, attempts to remove all table constraints (which may be linked to the first index created on the table) before removing the actual table.

SQLite 3.x	database	SQLite connections are supported as native, direct connections to the SQLite database file. No separate drivers are required.
Sybase ASE15	database	

In addition, DatabaseSpy provides basic support for virtually any database when connected via ODBC. This includes databases such as Mimer SQL, as well as connections to Microsoft Excel work sheets and text files. Note, however, that DatabaseSpy's functionality is limited when connected to a database that is not natively supported.

Chapter 3

The DatabaseSpy Interface

3   The DatabaseSpy Interface

When you open DatabaseSpy for the first time, the DatabaseSpy interface will show two vertical areas: on the left side, the Project window and the Properties window are displayed and tabs for the Online Browser and the Overview window are visible. These windows as well as the Data Inspector window, which can be displayed on demand, are referred to as the **side windows**.
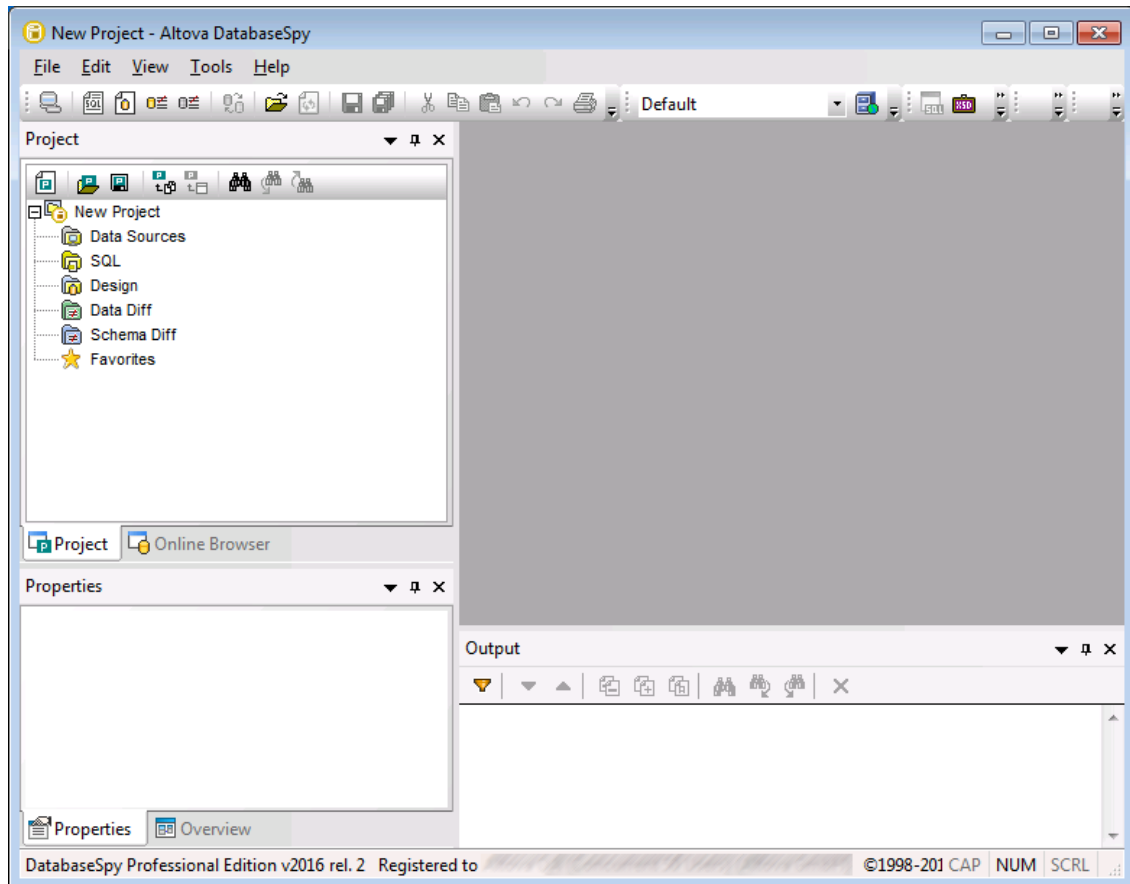
In the lower part of the right side, the Output window is displayed, whereas the upper part is still unused (*screenshot above*). This space will later be occupied by the **editor and comparison windows** (SQL Editor, Design Editor, or the Data or Schema Comparison windows, respectively). Except for the SQL Editor, an active connection to a data source is required to display these windows (*screenshot below*).
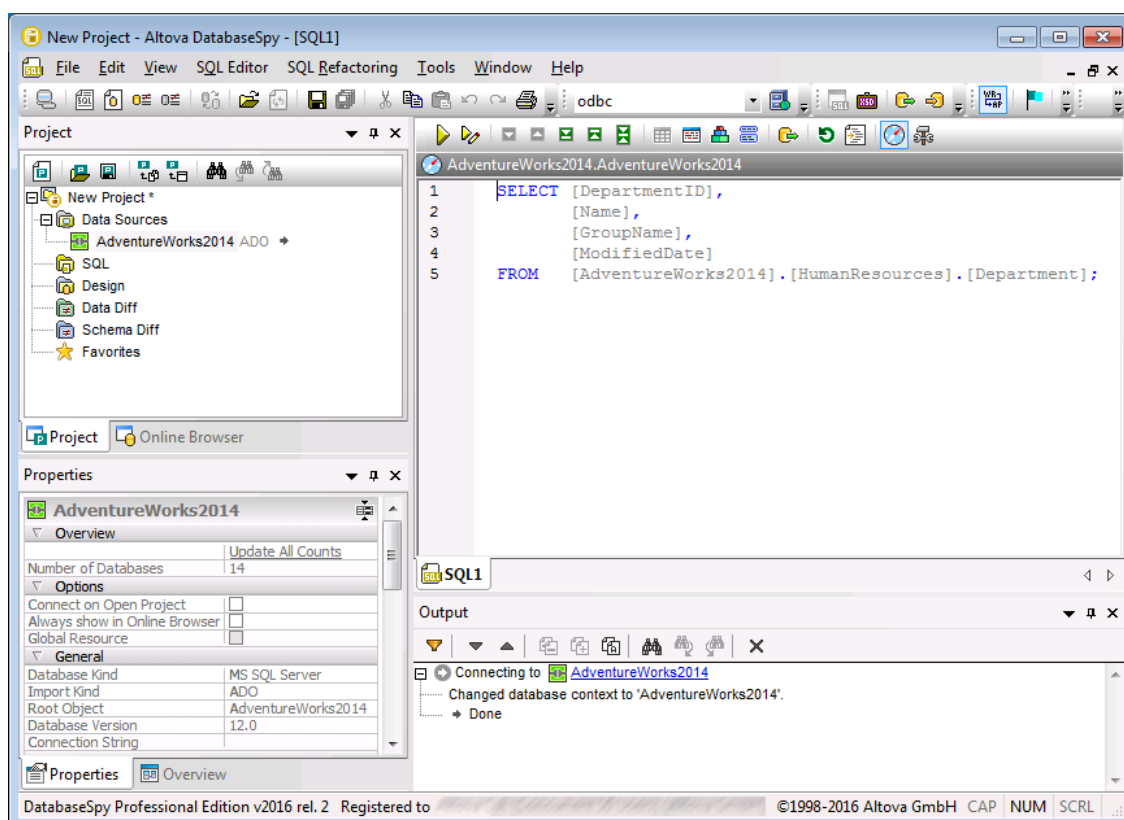
The different windows can be disabled or enabled via the **View** menu and, in addition, you can change the appearance of an individual window by selecting the little arrow in the title bar of a window and choosing the desired option from the drop-down list.

Hiding Side windows, Output and Change Script windows

The **View** menu also contains two options, **Toggle All Side Windows** and **Toggle Output and Change Script Windows**, that can be used to quickly hide these windows if you need all the available space for displaying the content in the SQL Editor, Design Editor, Data Comparison, or Schema Comparison window.

3.1   Side Windows

3.1.1   Project Window
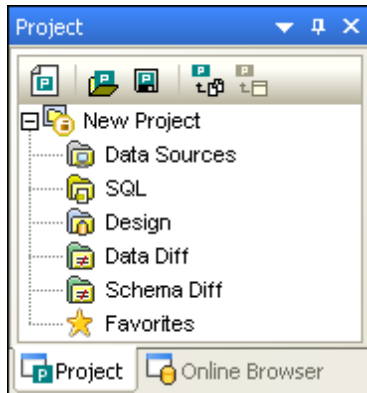
The Project window displays all data source connections as well as any SQL files, design documents, or saved data and schema comparisons that you add to your project. In addition, project favorites are shown which allow you to access items that you need frequently even faster.

Project toolbar

Create a new project: Clicking this button closes the active project in the Project window and displays a new, empty project. If a project contains unsaved changes when you click this button, a dialog box appears and you can choose to save changes in the editor or comparison windows and project modifications. All editor and comparison windows are set offline.

Open a project: Shows the **Open Project** dialog box where you can choose a *.qprj file to open in DatabaseSpy. If there is an unsaved project already open in the Project window, you are prompted to save this project.

Save project: Saves the project under its project name. If you save a new project for the first time, the **Save As** dialog box is displayed where you can specify a project name and choose a location. The project name then also appears in the Project window.

Add files to the project: Shows the **Open** dialog box where you can browse for DatabaseSpy documents (SQL files, design files, data comparison files, or schema comparison files) that should be added to the project. Files are added to the corresponding folder in the Project window.

Add the active file: Adds the active file to the project. If the file has not been saved yet, you are prompted to enter a name and location for the file. The file is then added to the corresponding folder in the Project window.

Context menu options

Depending on where you right-click into the Project window, different options are displayed in the context menu. The corresponding Toolbar icons are shown to the left, and keyboard shortcuts to the right of the options if they are available.

New Project	**Ctrl+Shift +N**	Creates a new project.	Project Name
Open Project	**Ctrl+Shift +O**	Opens a project.	Project Name

Save Project	**Ctrl+S**	Saves the project under its current name.	Project Name
Save Project As...		Opens the **Save As** dialog box where you can define a new name and/or path for the project.	Project Name
Add Files to Project...		Opens the **Open** dialog box where you can select files to be added to the project.	Project Name SQL Design Data Diff Schema Diff
Add Active File to Project		Includes the active file with the project and adds it to its corresponding folder in the Project window.	Project Name SQL Design Data Diff Schema Diff
Connect to all Data Sources		Connects to all data sources that are included in the project.	Data Sources
Disconnect from all Data Sources		Disconnects from all active data sources in the project.	Data Sources
Remove all Data Sources		Removes all data sources from the project.	Data Sources
Expand | Siblings **Expand | Children**		Expands all folders on the same level as, or all children of, the selected item.	Data Sources SQL Design Data Diff Schema Diff
Collapse | Siblings **Collapse | Children**		Collapses all folders on the same level as, or all children of, the selected item.	Data Sources SQL Design Data Diff Schema Diff
Add a new Data Source...		Opens the **Add a Data Source** dialog box where you can define a new data source connection.	Data Sources
Convert all to Global Resources		Converts all data sources in the project to global resources. Note that data source names must not contain blanks.	Data Sources
Connect		Connects to the selected data source.	Data source
Disconnect		Disconnects from the selected data source.	Data source
Remove	**Del**	Removes the selected item from the project.	Data source Sub-folders SQL file Design file Data comparison file Schema comparison file
Rename	**F2**	Renames the selected item. Note that you can only rename	Data source Sub-folders

		disconnected data sources.	
	Convert to Global Resource	Converts the selected data source into a global resource.	Data source
	Copy Global Resource into Project	Creates a copy of the global resource and adds it to the project as a normal data source.	Data source
	Edit Global Resource	Opens the **Global Resource** dialog box where you can edit the global data source connection.	Data source
	Create Folder...	Creates a sub-folder for the selected item.	SQL Design Data Diff Schema Diff
	Execute All SQL Files	Opens all SQL files that are contained in the SQL folder in new SQL Editor windows and executes them automatically. Note that a data source must be assigned to the files.	SQL
	Open	Opens the selected file in its corresponding editor or comparison window, respectively.	SQL file Design file Data comparison file Schema comparison file
	Locate file...	Shows the selected file in Windows Explorer.	SQL file Design file Data comparison file Schema comparison file
	Execute SQL	Opens and executes the SQL file in an SQL Editor window. Note that a data source must be assigned to the file.	SQL file
	Remove all favorites	Removes all items from the Favorites folder.	Favorites

3.1.2   Online Browser

Altova web site: database browser

The Online Browser displays all the database objects of the database you are connected to. It is organized in folders for Tables, Views, XML Schemas, Procedures, and Functions and each of these folders provides sub-folders where the relevant column, key, data type, or constraint information is displayed (*see left screenshot*).
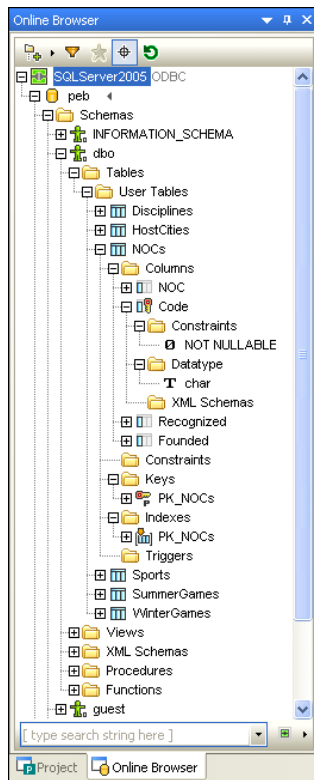
The Online Browser serves as a starting point for most of the database actions in DatabaseSpy: retrieve or edit data, create SQL statements, show database items in the Design Editor, compare data or schemas, export database data, or show the row count for tables. All these actions can be started from the Online Browser with a few mouse clicks.

The Online Browser is only populated when connections exist to data sources, or if the `Always show in Online Browser` check box has been activated for a data source in its properties.
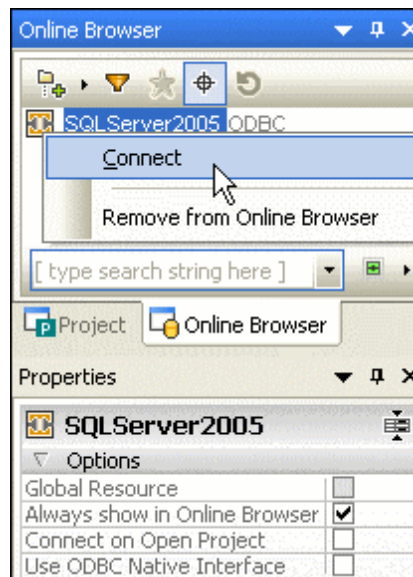
This will always display the data source connection in the Online Browser, even if it is disconnected, and you can connect to a data source directly from within the Online Browser without having to change to the Project window first (*see right screenshot*).

Online Browser toolbar

Layout: Click this button to choose from among several layouts to display the database objects. In the default Folders layout, the database objects are presented in a hierarchical manner.

Filter folder content (Ctrl+Alt+F): Choose this button if you want to reduce the number of displayed items by means of a filter.

Show Favorites: You can add database objects that you use frequently to your favorites, these objects appear then in bold in the Online Browser and are available in the Favorites folder of the Project window. The **Show Favorites** icon allows you to switch between viewing all objects and showing only favorites.

Object Locator (Ctrl+L): For finding specific objects in the database you can use the Object Locator. This button toggles the Object Locator on and off.

Refresh Data Source (F5): Clicking this button refreshes the data source to reflect the latest changes in the database structure and updates the display of the Online Browser.

3.1.3   Properties Window

The Properties window always displays the properties of selected object in the active window. For a better overview, each element has assigned a particular color in the properties title bar. You can change these colors in the Design Editor options. If no object is selected, the properties of the editor or comparison window itself will be displayed.

Hiding empty properties

You can customize the appearance of the Properties window by clicking one of the following buttons at the right edge of the header line:

Hide empty properties: Hides all lines where no entry appears.

Show empty properties: Restores all lines and displays also empty properties.

Displaying the count of child objects

For performance reasons, information on child objects of an element is loaded only on demand, that is, if that particular element is expanded in the Online Browser or displayed in the Design Editor. However, you can nevertheless obtain this information by clicking the appropriate "Update Count" field in the object properties.

Properties	▼ 📌 ✕
�'t dbo	📑
▽ Overview	
	Update All Counts
Number of Tables	6
Number of Views	Update Count
Number of XML Schemas	Update Count
Number of Procedures	Update Count
Number of Functions	Update Count

Properties	▼ 📌 ✕
🎟 Disciplines (dbo)	📑
▽ Overview	
Number of Columns	5
Number of Check Constraints	0
Number of Primary Keys	1
Number of Foreign Keys	1
Number of Unique Keys	0
Number of Indexes	1
Number of Triggers	0

If you expand an element to show its child objects, the number of objects will also be shown when you click the parent element again.

3.1.4   Overview Window
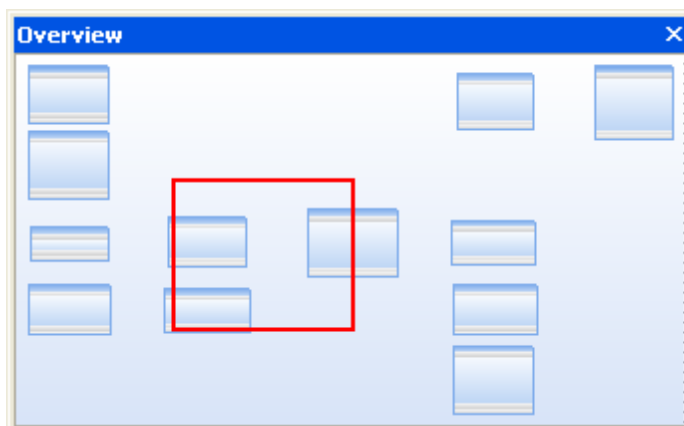
The **Overview window** serves as a navigator for large design files that occupy more space than is available in the Design Editor window. The Overview window shows the currently displayed area in the design and you can move the focus by clicking into the rectangle and moving it to the desired location in the design file.

By default, the Overview window is a tabbed window in the bottom left part of the application. You can, however, undock the window and drag it to any position within the application.

3.1.5   Data Inspector Window
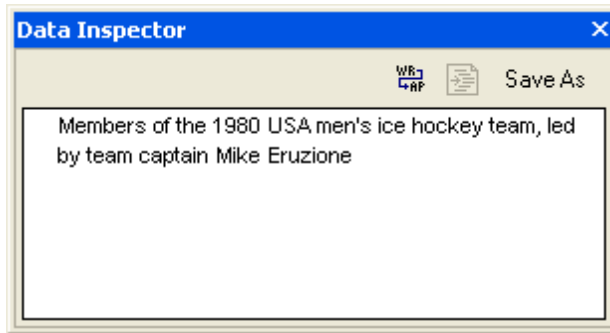
The Data Inspector Window is a convenient way to display the content of large data cells when you have only limited space to view the data directly in the result grid. The Data Inspector can be called via the **View** menu or by clicking the **Show Data Inspector window** while one ore more data cells are selected in the Result window or in the Database Data Comparison window.

You can resize the Data Inspector window to view all the data it contains, or dock it to the application. In addition, the following buttons are available in the Data Inspector:

Word wrap: Automatically wraps the text in the window so as to make all the text visible. This is a toggle command, that is, word wrap is turned off again if you click the button again.

Pretty Print: Displays the text in a hierarchical way so that the tags are indented properly.
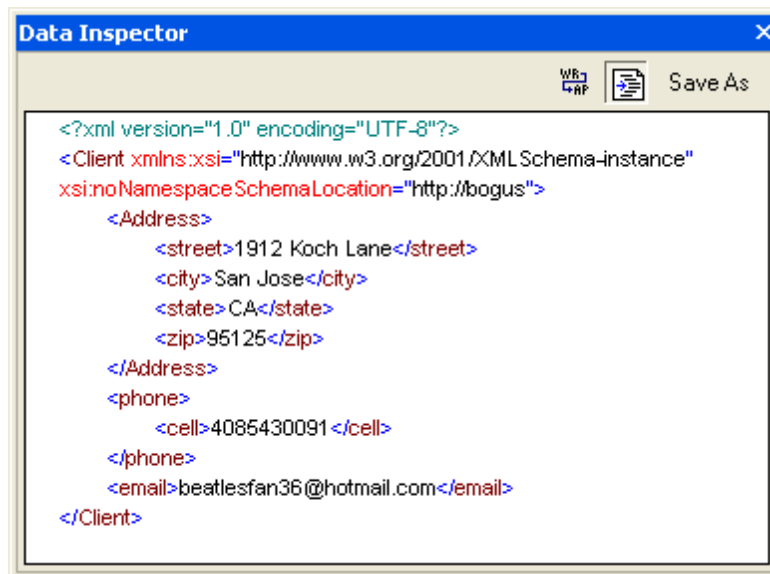
Save As: Opens the standard Windows **Save As** dialog box where you can specify a path and file name for the content of the data cell. The default file format is XML for XML columns and text for any other columns. You can also choose a different file format if you select "All Files" in the `Save as type` drop down list.

3.2   Output Window
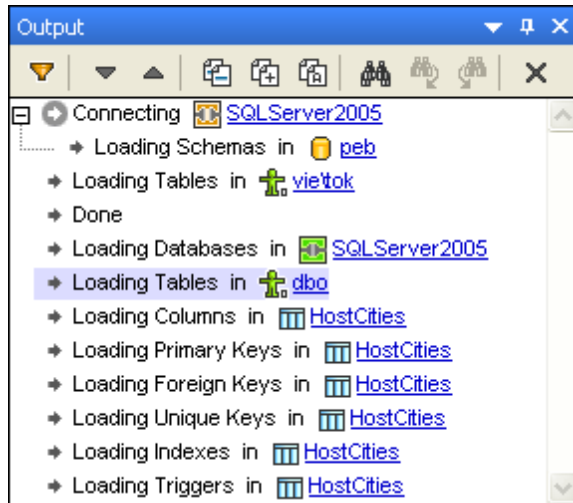
The **Output window** in DatabaseSpy contains loading and error messages. Underlined parts in the Output window are hyperlinks which enable you to jump to a certain database object in the Online Browser.

Toolbar options

The message window provides a toolbar and a context menu that allow for the navigation inside the messages and include filters for hiding certain parts of the message. The corresponding toolbar icons are shown to the left of the options if they are available.

	Summary, Success, Warning, Error, Progress	When checked, this parts of the message are displayed in the message window.
▼	**Next**	Jumps to and highlights the next message.
▲	**Previous**	Jumps to and highlights the previous message.
	Copy message	Copies the selected message to the clipboard.
	Copy message with children	Copies the selected message including its children to the clipboard.
	Copy all	Copies all messages in the message window to the clipboard.
	Find	Opens the **Find** dialog box.
	Find previous	Jumps to the previous occurrence of the string specified in the **Find** dialog box.
	Find next	Jumps to the next occurrence of the string specified in the **Find** dialog box.
✕	**Clear**	Removes all messages from the message window.

In addition, the toolbar provides the following icon:

▼	**Filter**	Opens a popup menu from where you can select the individual message parts for display. Furthermore, you can check all or none of these options with a single mouse click by selecting either **Check All** or **Uncheck All** from the popup menu.

3.3   Database Structure Change Script

The **Database Structure Change Script** records any changes you make to the database structure in an SQL script. This script is either generated instantly or on demand, depending on the settings in the Change Script options.
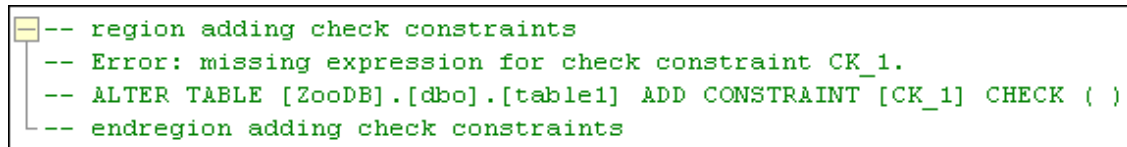
```
Database Structure Change Script                                          ×
  ▷ | 🗐 | 📰 | 💾 | ✕
   1      ⊟-- region change columns
   2       ALTER TABLE [ZooDB].[dbo].[tblAnimalBirths] ALTER COLUMN [Mother]
            int  NOT NULL ;
   3       ALTER TABLE [ZooDB].[dbo].[tblFeedSuppliers] ALTER COLUMN
            [SupplierState]  varchar (20) COLLATE SQL_Latin1_General_CP1_CI_AS
            NULL ;
   4      └-- endregion change columns
   5      ⊟-- region creating tables
   6       CREATE TABLE [ZooDB].[dbo].[visitors] ( [date] datetime NOT NULL ,
            [location] varchar (255) NULL , [total] int NULL ) ;
   7      └-- endregion creating tables
   8      ⊟-- region adding columns
   9       ALTER TABLE [ZooDB].[dbo].[tblFeedSuppliers] ADD [SupplierEmail]
            varchar (255) NULL ;
  10      └-- endregion adding columns
```

In order to provide a better overview if changes are made in large scale, the change script is structured into regions, which can be collapsed or expanded when checking the script. Please note that collapsing a region will not exclude this part of the script from execution; collapsed regions are expanded automatically when the change script is being executed.

If you have several data sources connected and you change the structure in more than one of them, a separate tab is created for each data source in the Database Structure Change Script window.

When defining primary keys or check constraints, the Database Structure Change Script window also displays comments on the validity of the definition.

```
⊟-- region adding check constraints
 -- Error: missing expression for check constraint CK_1.
 -- ALTER TABLE [ZooDB].[dbo].[table1] ADD CONSTRAINT [CK_1] CHECK ( )
└-- endregion adding check constraints
```

Toolbar options

The following options are provided in the Database Structure Change Script toolbar:

▷	**Execute Change Script**	Executes the change script and thus implements the recorded changes in the database. In the Change Script options, you can define the time delay during execution of the script.
■	**Stop**	Terminates the execution of the change script, the remaining changes are not implemented in the database and the Database Structure Change Script window is cleared. This button is only visible while a change script is being executed. **Please note**: When you click the **Stop** button, all changes

that have not been executed so far will be lost.

Generate Change Script   Generates or updates, respectively, a change script that reflects all the changes you have made in the database design since the change script has last been updated. This button is only active if you have defined in the Change Script options that the change script be generated on demand.

Edit Change Script   Opens the change script in a new SQL Editor window where you can edit it.
Please note: The script will be removed from the Database Structure Change Script window and the changes will be lost unless you execute the script in the SQL Editor.

Save Change Script   Opens the standard Windows **Save As** dialog box where you can save the script as an SQL file.

Discard Changes   Discards all changes recorded in the change script. The changes you made in the design will not be implemented in the database and the Database Structure Change Script window will be cleared.

3.4 Editors and Comparison Windows

3.4.1 SQL Editor

Altova web site: SQL Editor

The SQL Editor is used to display, edit, and execute SQL statements. It provides a Message tab that displays success or error messages and warnings, and a Result tab for queries where you also can edit data if the SQL Editor runs in Editing mode.
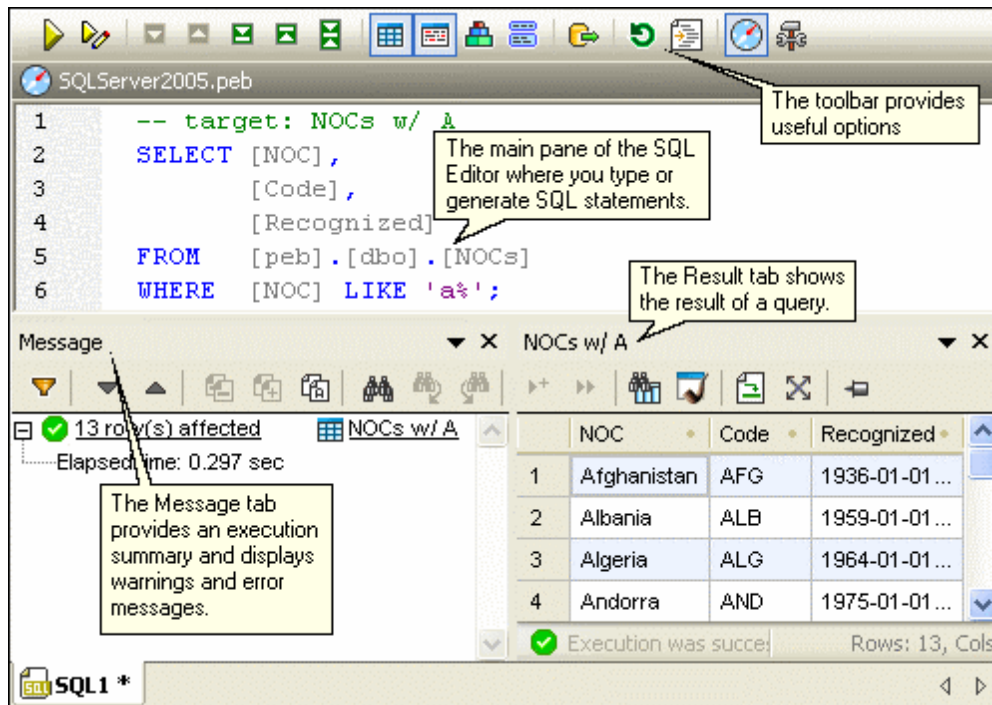
The toolbar options in an SQL Editor window are also included in the following menus:

SQL Editor

SQL Refactoring

Tools
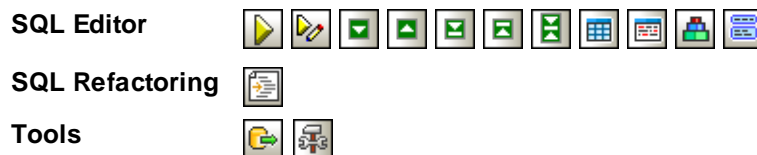
In addition, the toolbar provides the following icons:

Reparse the whole document — Upon clicking this button the statements in the currently active SQL Editor window will be reparsed once again. This way, parsing errors that occurred during the initial parsing of the statement can be corrected automatically.

Show/Hide execution target bar — Toggles the display of the Execution Target Bar on and off. Note that the Execution Target Bar will also be toggled off for all SQL Editor and Design Editor windows that you open subsequently.

Message Tab

The **Message tab** provides a statistical overview on the SQL statement that has been executed last and reports errors that might have occurred during execution.

Message
▼ | ▼ ▲ | 🗐 🗐 🗐 | 🗗 🗗 🗗 | ✕
⊟ ✅ 26 row(s) affected   ⊞ Result1
└─ ⊟ 🗗 Data Editing
└────Updating is not allowed for column 'DisciplineID' because it is an auto increment column.
└────Insert is not allowed for column 'DisciplineID' because it is an auto increment column.
└────Elapsed time: 4.516 sec
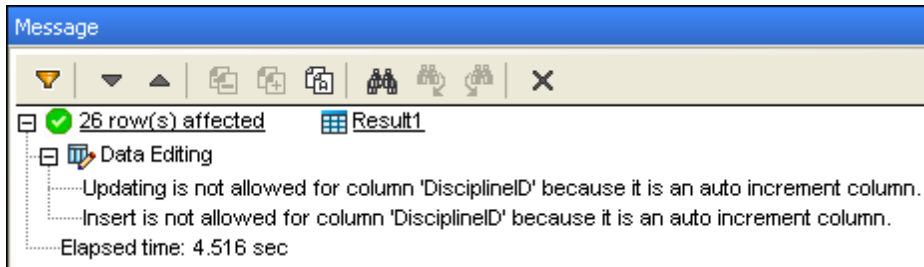
Underlined parts in the Message tab are hyperlinks which enable you to jump to a certain part of the SQL script in the SQL Editor, or to the corresponding Result tab that holds the result of the respective SQL statement.

Message options

The message window provides a toolbar and a context menu that allow for the navigation inside the messages and include filters for hiding certain parts of the message. The corresponding toolbar icons are shown to the left of the options if they are available.

	Summary, Success, Warning, Error, Progress	When checked, this parts of the message are displayed in the message window.
▼	**Next**	Jumps to and highlights the next message.
▲	**Previous**	Jumps to and highlights the previous message.
🗐	**Copy message**	Copies the selected message to the clipboard.
🗐	**Copy message with children**	Copies the selected message including its children to the clipboard.
🗐	**Copy all**	Copies all messages in the message window to the clipboard.
🗗	**Find**	Opens the **Find** dialog box.
🗗	**Find previous**	Jumps to the previous occurrence of the string specified in the **Find** dialog box.
🗗	**Find next**	Jumps to the next occurrence of the string specified in the **Find** dialog box.
✕	**Clear**	Removes all messages from the message window.

In addition, the toolbar provides the following icon:

▼	**Filter**	Opens a popup menu from where you can select the individual message parts for display. Furthermore, you can check all or none of these options with a single mouse click by selecting either **Check All** or **Uncheck All** from the popup menu.

Result Tab

The **Result tab** of the SQL Editor shows the record sets that were retrieved as a result of the database query. If an SQL Editor window contains more than one query, the results are displayed in individual tabs (*see screenshot below*) or stacked in a single window.
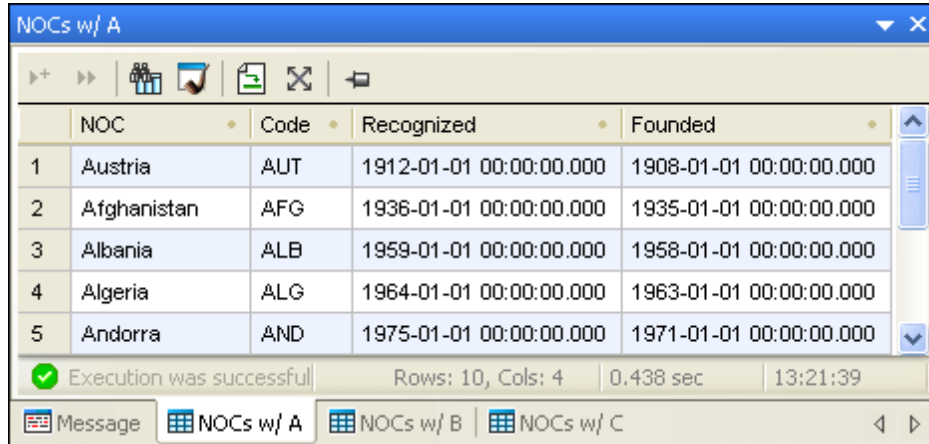
The status bar at the bottom of the window displays information on the progress of the query: whether the retrieval executed successfully, was aborted or has been stopped by the user. In addition, the number of rows and columns retrieved as well as the amount of time necessary for retrieval, and the time when the query was executed.

Maximum number of result tabs

In DatabaseSpy, a maximum number of 30 result tabs is allowed. If you execute a query that would produce more than 30 result tabs, only the first 30 result tabs are generated and a corresponding message is displayed in the message tab.
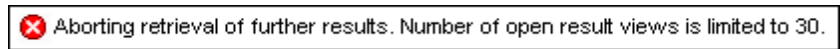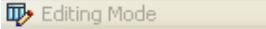
Result options

When you right-click anywhere in the Result tab of the SQL Editor, a context menu opens which provides options for working with the results of a query. The corresponding Toolbar icons are shown to the left, and keyboard shortcuts to the right of the options if they are available. Please note that the available options in the content menu change dynamically depending on the command you have used for retrieving the data in the Result window. The Result tab can therefore be in three different states: Execution was successful, Stopped retrieval, or Editing mode.

	Execution was successful	The query has been started by executing a normal SELECT or by right-clicking tables, columns or views in the Online Browser and selecting **Retrieve data	All rows** from the context menu.	
	Go to statement		Jumps to the SQL Editor window and highlights the group of SQL statements that produced the respective result.
	Auto Size Columns		Adjusts the column width of all columns to provide an optimized view of text contained in

			the columns. Please note that the maximum column width for auto-sizing in the Result tab is 250 pixels. In data cells containing a large amount of text, only the first part of the text is displayed followed by an ellipsis (...). You can, however, always drag the column border to show more text.
	Selection		Allows you to select the row or column in which the cursor is currently positioned, or the entire table. Choose the desired option from the sub-menu.
	Sorting		Opens a sub-menu to sort the table either ascending or descending and uses the column where the cursor is currently positioned as a sort key. To restore the default sort order, choose the corresponding option from the sub-menu.
	Copy selected cells	**Ctrl+C**	Copies the selected cells to the clipboard. You can select several individual cells by holding down the **Ctrl** button and clicking the desired cells or select a consecutive number of neighboring cells by clicking a cell and pressing the **Shift** key while clicking a second cell. The content of the cells can be pasted as tabbed text into a text editor or spreadsheet calculator, or can replace existing cells in the result grid. This menu option is only available if data editing in the Result window is enabled.
	Copy selected cells with header		Copies the selected cells as well as the corresponding headers (i.e., the column names) to the clipboard.
	Show in Data Inspector Window		Opens the Data Inspector window where you can conveniently view the content of large data cells.
	Show result toolbar		Shows or hides the toolbar in the Result tab of the SQL Editor window.
	Auto Hide		currently not available
	Hide		Hides the Results tab of the SQL Editor window. In order to show the Results tab again, click the **Toggle Result Window(s)** button in the toolbar of the SQL Editor window or choose the menu option **SQL Editor | Result view**.
	Stopped Retrieval		The query has been started by right-clicking tables, columns or views in the Online Browser and selecting **Retrieve data |**

			First n rows from the context menu. When the SQL Editor is in the Stopped Retrieval mode (i.e., not all rows of data have been retrieved yet or if the **Stop Retrieval** ■ button has been clicked), the following additional options are available in the Result tab:
▶⁺	**Retrieve next n rows**		Retrieves the next n rows from the query in the active SQL Editor window.
▶▶	**Retrieve outstanding rows**		Retrieves all the remaining rows from the query in the active SQL Editor window.
	Editing Mode		The query has been started by executing a SELECT statement with the **Execute for Data Editing** command, or by right-clicking tables, columns or views in the Online Browser and selecting **Edit data** from the context menu. When the SQL Editor is in the Editing mode, the following additional options are available in the Result tab:
	Paste	**Ctrl+V**	Pastes the content of the clipboard into the cell —and, if applicable, its neighboring cells— where the cursor is currently positioned. Note that if you use the **Edit cell** command before pasting, the entire content of the clipboard will be pasted into the selected cell as tabbed text.
	Paste as new rows	**Ctrl+Shift+V**	Adds new rows to the result grid and pastes the content of the clipboard into the new rows. Note that only content that has been copied using the **Copy selected cells** command can be pasted as new rows.
	Edit cell		Enables the data cell for editing and selects the content of the cell. This menu option is only available if no limitations with regard to editing records apply.
∅	**Set Null**		Resets the cell to the NULL value.
Def	**Set Default**		Resets the cell to the default value. This menu option is only available if a default value has been defined for the cell.
	Undo Changes for this Cell		Rejects the changes that have been made to a data cell but not yet committed to the database.
+	**Append a new row**	**Alt+Ins**	Adds a new line to the result grid, where you can add records to the database table. The data must be committed to the database by clicking the Commit button. This menu option is only available if no limitations with regard to adding records apply.

☒	**Delete row**		[Deletes the row](#) in which the cursor is currently positioned. This menu option is only available if no limitations with regard to deleting records apply.

In addition, the toolbar provides the following icons:

🔍	**Find**	Opens a **Find** dialog box allowing you to search a particular string in the Result window.
⤢	**Maximize result view**	Enlarges the size of the Result window to a maximum.
↺	**Undo all changes**	Rejects all changes that have been made in the result grid since the data has been selected from the database or since updated data has last been committed to the database, respectively. This toolbar option is only available if the Result window is in the Editing mode.
📌	**Pin down**	Pins the result tab so that it remains visible even if other queries are executed in the same window.

Special options for XML columns

When the Result window is in the Editing mode and XML columns are present in the current grid, the following menu opens if you click the 🖳 button:

```
Load XML Document from File...
Save XML Document to File...

Assign XML Schema...
```

Load XML Document from File...	Opens the standard Windows **Open** dialog box from where you can choose an XML file to load into the selected cell in the result grid.
Save XML Document to File...	Opens the standard Windows **Save As** dialog box where you can choose a location to save the content of the selected cell as XML file.
Assign XML Schema...	Opens the **Choose XML Schema** dialog box where you can [assign an XML schema](#) for validation of the XML file stored in the database field.

XML support is available for the following databases:

- DB2 9.1, 9.5, 9.7, 10.1, 10.5
- SQL Server 2005, 2008, 2012, 2014, 2016
- Oracle 9i, 10g, 11g, 12c
- PostgreSQL 8.3, 9.0.10, 9.1.6, 9.2.1, 9.4, 9.5

Note: PostgreSQL does not have XML schema management (add schema, drop schema).

3.4.2   Design Editor

Altova web site: 🌐 database design

The **Design Editor** is a graphical user interface where you can view your database objects and their relations as graphical objects. A table including its columns, indexes, keys, and constraints is displayed in a so-called table design (*see screenshot below*). You can add new tables or edit existing ones or just drag a table into a Design Editor window to see what its structure looks like.

When you add new objects (e.g., tables or columns) to the database or change the structure of existing ones, these changes are not immediately effective in the database. Any changes in the database design are recorded in the Database Structure Change Script and can then either be executed or rejected.
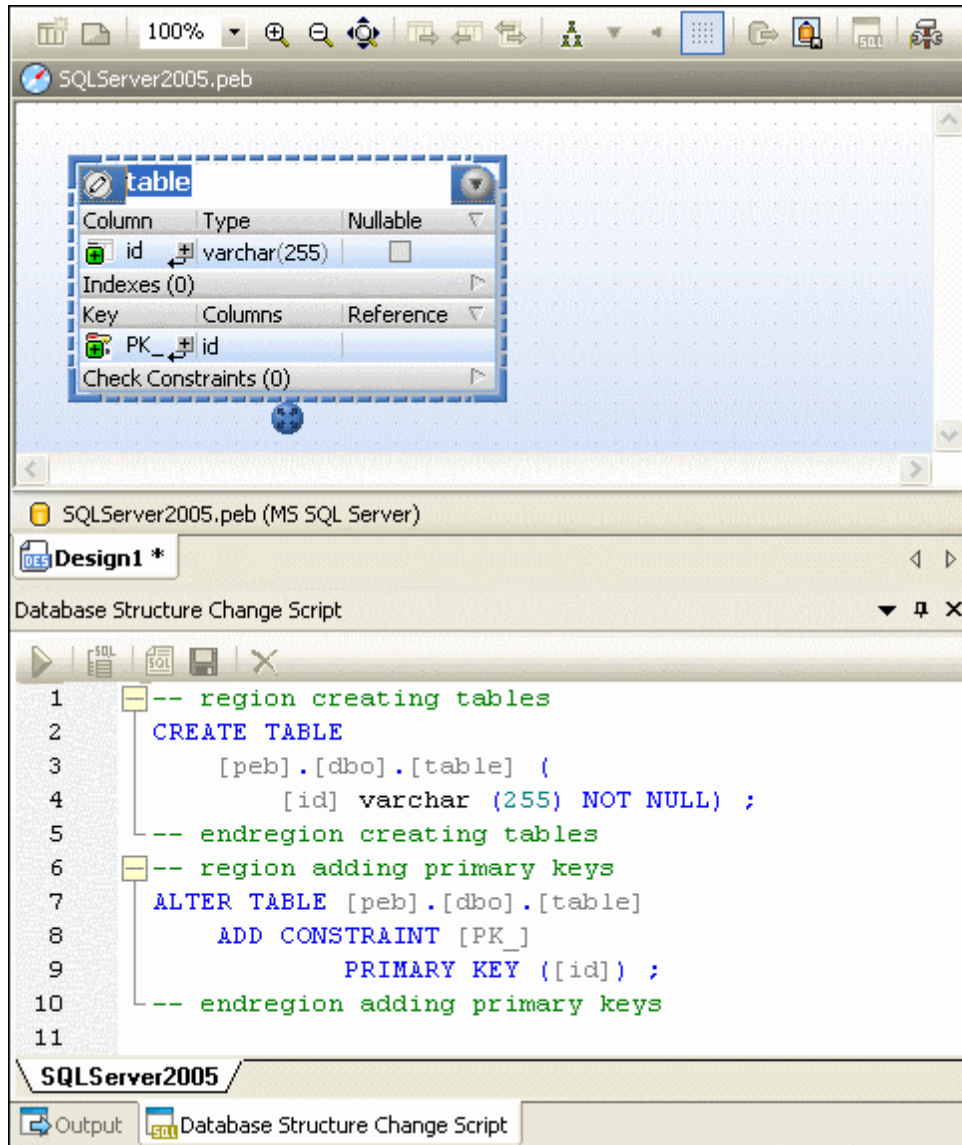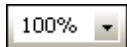
Design options

When you right-click in an unoccupied area of the Design Editor, several general options are available in the context menu which are listed below. The corresponding Toolbar icons are shown to the left, and keyboard shortcuts to the right of the options if they are available.

Create new Table	**Ctrl+T**	Creates a new table in the currently active Design Editor window and automatically assigns it to the active data source, that is, the data source which is selected in the Online Browser.
Add Sticky Note	**Alt+N**	Adds a Sticky Note to the Design Editor window.
Select | Select User Tables	**Alt+T**	Selects all user tables that are contained in the active Design Editor window.
Select | Select System Tables	**Alt+S**	Selects all system tables that are contained in the active Design Editor window.

	Auto Layout whole Diagram		Shows the design in a hierarchical layout where all relations between the tables are considered and the tables are laid out properly.
	Save Diagram as Image		Saves the currently active design as Portable Network Graphic (*.png) or Enhanced Windows Metafile (*.EMF).

In addition, the toolbar provides the following icons:

100% ▾			Changes the **zoom factor** of the Design Editor window; enter a value or select a zoom factor from the drop-down list.
	Zoom in	**Ctrl+Num +**	Increases the zoom factor of the Design Editor window by 10 percent. You can also zoom in by scrolling (with the scroll-wheel of the mouse) while keeping the **Ctrl** key pressed.
	Zoom Out	**Ctrl+Num -**	Decreases the zoom factor of the Design Editor window by 10 percent. You can also zoom out by scrolling (with the scroll-wheel of the mouse) while keeping the **Ctrl** key pressed.
	Zoom to Fit		Sizes the working area to include all tables in the current Design Editor window.
	Insert Referenced Tables		Inserts all related tables that are referenced by a constraint of the selected table.
	Insert Referencing Tables		Inserts all related tables that include a constraint referencing the selected table.
	Insert All Related Tables		Inserts both referenced and referencing tables.
	Expand selected Tables		Expands all tables that are selected in the current Design Editor window.
	Collapse selected Tables		Collapses all tables that are selected in the current Design Editor window.
	Snap to Grid	**Ctrl+G**	Shows/hides the grid and toggles the snap-to-grid function on and off.
	Export		Opens the **Export database data** dialog box where the Table radio button is already preselected.
	Update Change Script		Starts the generation of a change script or updates it after changes occurred to the database structure. Please note that this button is grayed out if in the General options you have activated the Instantly radio button in the Database Structure Change Script group box.
	Show Options		Opens the Design Editor options dialog box where you can customize the settings for the Design Editor.

When the Design Editor displays one or more tables, a variety of options is provided if you right-click into a table design. Which options are included in the context menu depends on where you

click within the table design.

3.4.3   Data Comparison Window

The Data Comparison window allows users to select and display tables from two data sources (that may or may not be different) for the purposes of comparing the data in these tables. Mappings are defined either automatically by DatabaseSpy or manually by the user in this view. Mapping of tables can be fine-tuned to include a subset of the table's columns. Note that you need an active connection to a data source to be able to display a Data Comparison window.
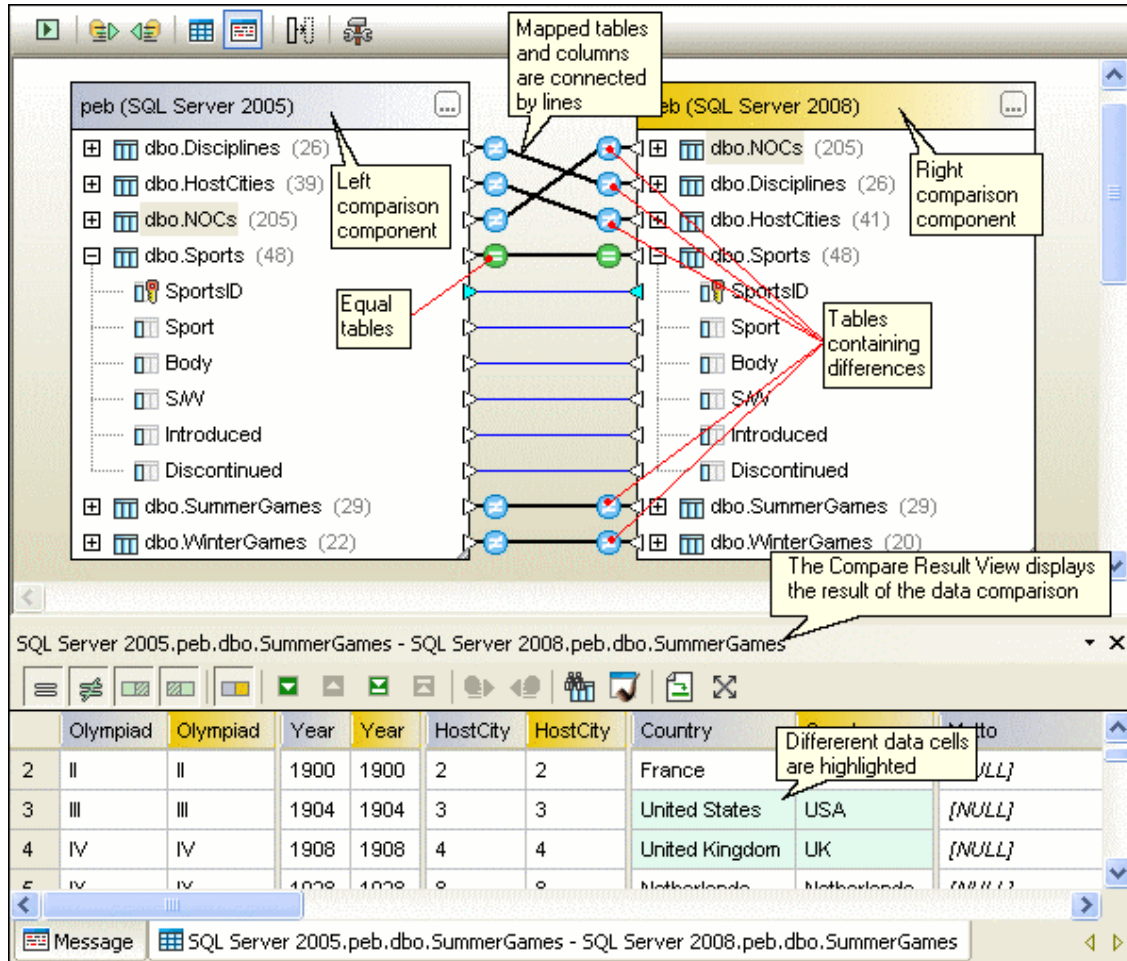
Data comparison options

When you right-click in an unoccupied area of the Data Comparison window, several options are available in the context menu which are listed below. The corresponding Toolbar icons are shown to the left of the options if they are available.

▶	**Compare tables**	Starts a data comparison of the tables that are currently mapped in the Data Comparison window.
⇨	**Merge Left to Right**	Merges the mapped tables so that the table on the right side is updated with the data of the table on the left side of the comparison. If you start this command from the toolbar and one or more tables are selected, only the selected tables will be merged.

	Merge Right to Left	Merges the mapped tables so that the table on the left side is updated with the data of the table on the right side of the comparison. If you start this command from the toolbar and one or more tables are selected, only the selected tables will be merged.
	Show results	Shows the result of the data comparison in a new tab of the Comparison Result View (if the data has not been compared yet) or displays the corresponding result tab (if the data has already been compared before).
	Open in new Schema Comparison	Opens all tables that are included in the data comparison in a new Schema Comparison window.
	SQL and Data | Show merge script: Left to Right	Generates a merge script for all different tables in the data comparison and updates the table in the right component with the data from the table in the left component.
	SQL and Data | Show merge script: Right to Left	Generates a merge script for all different tables in the data comparison and updates the table in the left component with the data from the table in the right component.
	SQL and Data | Show restore script: Left	Generates a script for the left side of the comparison that restores the data to the values that were stored in the database before the merge. Note that the restore script must be generated before the merge script is executed!
	SQL and Data | Show restore script: Right	Generates a script for the right side of the comparison that restores the data to the values that were stored in the database before the merge. Note that the restore script must be generated before the merge script is executed!
	Sort tables | Ascending	Sorts the tables in the comparison component ascending.
	Sort tables | Descending	Sorts the tables in the comparison component descending.
	Sort tables | Ascending mapped first	Sorts the tables in the comparison component ascending and displays mapped tables on top.
	Sort tables | Descending mapped first	Sorts the tables in the comparison component descending and displays mapped tables on top.
	Map items	Maps corresponding tables in the Data Comparison window.
	Unmap items	Removes all mappings from the Data Comparison window.
	Expand tables	Expands all tables in the Data Comparison window and shows the table columns.
	Collapse tables	Collapses all tables in the Data Comparison window.
	Autolayout	Optimizes the component size and tries to display as much of the two components as possible in the comparison window.

In addition, the toolbar provides the following icons:

	Stop Comparison	Stops the currently running comparison process. This button is only visible while a comparison is running.
	Toggle Message Window	Toggles the Message tab of the Data Comparison window on and off.
	Show Options	Opens the Data Compare page of the **Options** dialog box.

Compare Result View

The result of a database data comparison is displayed in the Compare Result View which also provides context menu options and toolbar buttons for merging data to the left or right side of the comparison.
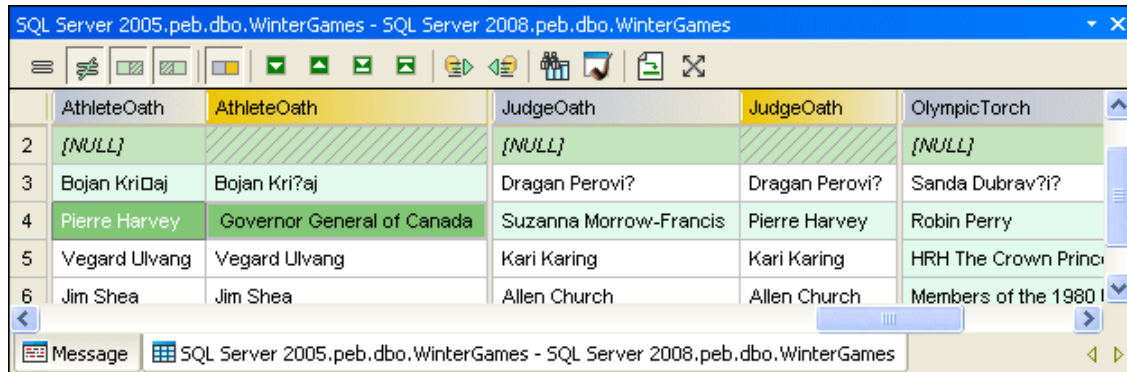
If the tables contain columns with different data types on the left and right sides which prevent data from being merged in one of the tables, the toolbar contains two additional icons which can be used to show or hide such columns. Note that in the screenshot below the **Show/Hide columns which don't have any differences** option has been toggled off.
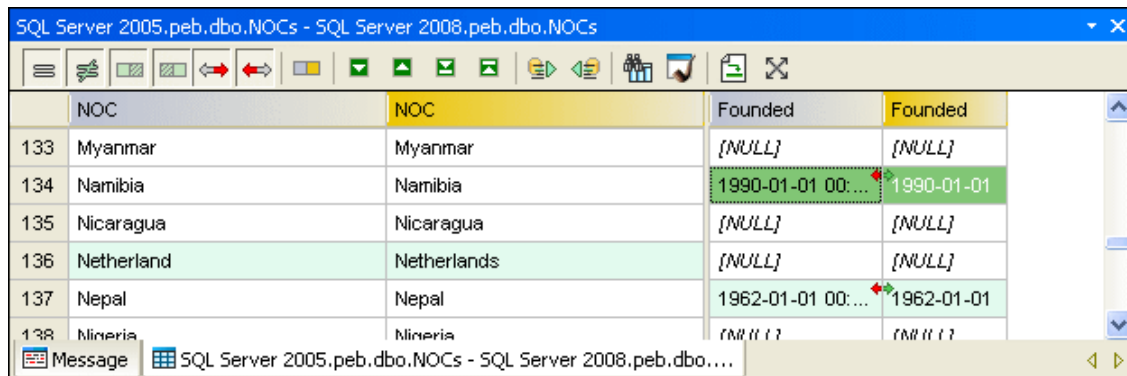
Comparison result options

The Compare Result view provides context menu options and toolbar buttons for filtering the content displayed in the Compare Result View, for navigating the differences, and for merging the differences between the two tables. The corresponding Toolbar icons are shown to the left, and keyboard shortcuts to the right of the options if they are available.

	Auto Size Columns	Adjusts the column width for all columns in the result grid so as to display the content properly.
▤	**Show/Hide all equal rows**	Shows or hides rows that are equal in the left and right table.
⧳	**Show/Hide all different rows**	Shows or hides rows that are different in the left and right table.
▧	**Show/Hide rows that are only on the left**	Toggles on and off the display of rows that are present in the left table only.

	Show/Hide rows that are only on the right	Toggles on and off the display of rows that are present in the right table only.
	Show/Hide rows that cannot be merged from left to right	Shows or hides rows that are different in the left and right component and can be merged only from the right to the left side and not from the left to the right side. This button is only visible if columns with conflicting data types are compared.
	Show/Hide rows that cannot be merged from right to left	Shows or hides rows that are different in the left and right component and can be merged only from the left to the right side and not from the right to the left side. This button is only visible if columns with conflicting data types are compared.
	Next difference	Selects the next difference as the current difference.
	Previous difference	Selects the previous difference as the current difference.
	Last difference	Selects the last difference in the document as the current difference.
	First difference	Selects the first difference in the document as the current difference.
	Merge data from left to right	Updates the selected data cell in the right table using the data contained in the data cell of the left table. Note that the change is committed to the database immediately without a change script being generated.
	Merge data from right to left	Updates the selected data cell in the left table using the data contained in the data cell of the right table. Note that the change is committed to the database immediately without a change script being generated.
	Copy selected cells Ctrl+C	Copies the selected cells to the clipboard. You can select several individual cells by holding down the **Ctrl** button and clicking the desired cells or select a consecutive number of neighboring cells by clicking a cell and pressing the **Shift** key while clicking a second cell. The content of the cells can be pasted as tabbed text into a text editor or spreadsheet calculator.
	Copy selected cells with header	Copies the selected cells as well as the corresponding headers (i.e., the column names) to the clipboard.
	Hide	Hides the Compare Result view of the Data Comparison window. In order to show the Compare Result view again, select the table in one of the components and click the **Show result** button in the toolbar of the Data Comparison window or choose the menu option **Data Comparison | Show results**.

Additionally, the toolbar provides the following icons:

	Show/Hide columns which don't have any differences	Toggles on and off the display of columns that are equal in the left and right table.
	Find	Opens the **Find** dialog box where you can search for data in the Compare Result View.
	Show the Data Inspector window	Displays the content of the selected cell in the Data Inspector window.

🔲	**Go to table mapping**	Shows the mapping of the compared tables in the Data Comparison window.
⤬	**Maximize the result window**	Enlarges the size of the Compare Result View to a maximum. To display the Data Comparison window again, click the **Go to table mapping** button.

3.4.4 Schema Comparison Window

The Schema Comparison window allows users to choose two schemas from two data sources (that may or may not be different) and display their items for the purpose of comparing the structure in these schemas. Mappings are defined either automatically by DatabaseSpy or manually by the user in this view.
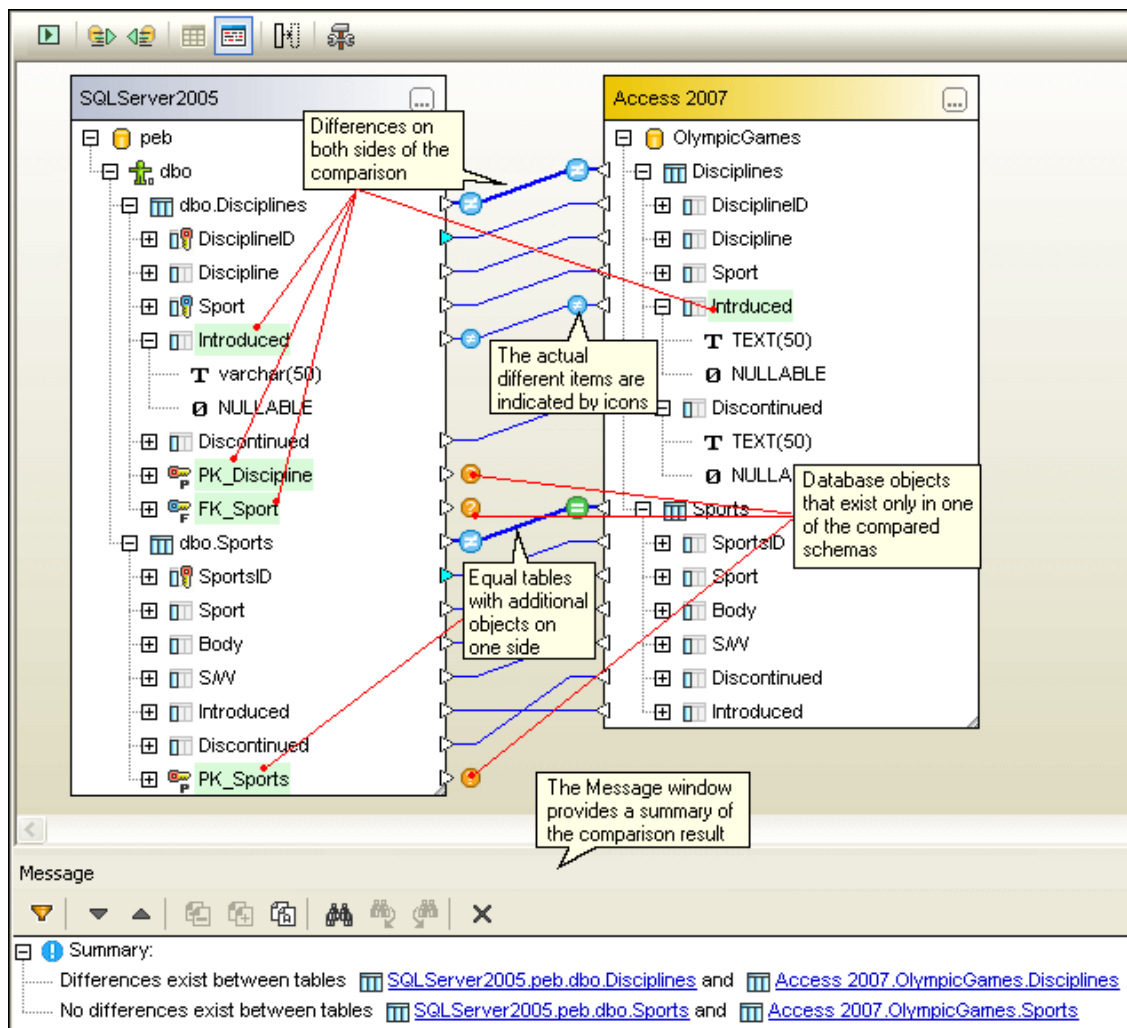
Schema comparison options

When you right-click in an unoccupied area of the Schema Comparison window, several options are available in the context menu which are listed below. The corresponding Toolbar icons are shown to the left of the options if they are available.

▶	**Compare items**	Starts a schema comparison of the items that are currently

mapped in the Schema Comparison window.

	Show merge script: Left to Right	Generates a merge script for all different items in the schema comparison and updates the structure of the items in the right component with the structure from the items in the left component. If you start this command from the toolbar and one or more items are selected, the merge script will be generated only for the selected items.
	Show merge script: Right to Left	Generates a merge script for all different items in the schema comparison and updates the structure of the items in the left component with the structure from the items in the right component. If you start this command from the toolbar and one or more items are selected, the merge script will be generated only for the selected items.
	Open in new Data Comparison	Opens all tables that are included in the schema comparison in a new Data Comparison window
	**Sort items	Ascending**	Sorts the items in the comparison component ascending.
	**Sort items	Descending**	Sorts the items in the comparison component descending.
	**Sort items	Ascending mapped first**	Sorts the items in the comparison component ascending and displays mapped items on top.
	**Sort items	Descending mapped first**	Sorts the items in the comparison component descending and displays mapped items on top.
	Map items	Maps corresponding items in the Schema Comparison window.
	Unmap items	Removes all mappings from the Schema Comparison window.
	Expand items	Expands all items in the Schema Comparison window and shows the child objects of the items.
	Collapse tables	Collapses all tables in the Schema Comparison window.
	Autolayout	Optimizes the component size and tries to display as much of the two components as possible in the comparison window.

In addition, the toolbar provides the following icons:

	Stop Comparison	Stops the currently running comparison process. This button is only visible while a comparison is running.
	Toggle Message Window	Toggles the Message tab of the Schema Comparison window on and off.
	Show Options	Opens the Data Compare page of the **Options** dialog box.

3.5   Execution Target Bar

The Execution Target Bar is an optional toolbar which displays the data source and root object currently assigned to the SQL Editor window or—if the data source connection is active—the Design Editor.
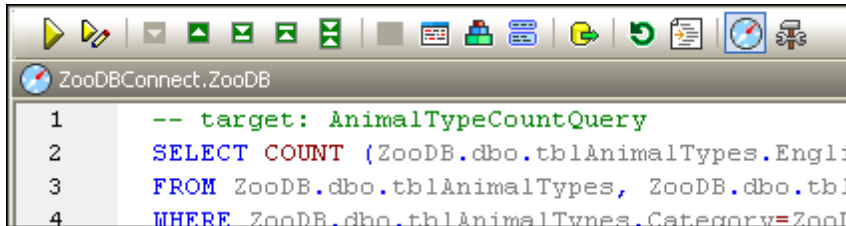
The Execution Target Bar is switched on for all SQL Editor windows by default. To hide the bar for individual SQL Editor windows, click the **Show/Hide execution target bar** button in the SQL Editor toolbar. Note that disabling the Execution Target Bar for an SQL Editor window will deactivate it in all SQL Editor and Design Editor windows that you open subsequently.

When you move your mouse cursor over the Execution Target Bar, the different elements appear as hyperlinks, enabling you to jump the Properties window where you can change the respective element.

Data source offline

If the data source to which an SQL Editor window is assigned is offline, the Execution Target Bar provides a **Connect** button for conveniently establishing a connection.
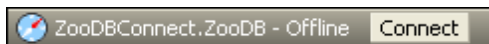
No data source defined

For an SQL Editor window that has no data source defined, you can use the hyperlink functionality to jump to the Properties window if you want to specify a data source. Click the "Offline" text in the Execution Target Bar to access the `Data Source` drop-down list where you can select a data source for the SQL Editor window.
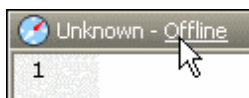
Differing root objects

If the root object specified for the SQL Editor window differs from the root object of the data source the window is currently connected to, the execution target bar shows a tooltip providing a more detailed description of the problem when you place the mouse cursor over its icon. By clicking the **Synchronize** button you can set the SQL script's root object to the currently active root object of its data source.
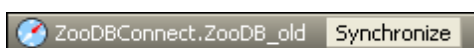
3.6   Menu Bar, Toolbars, and Status Bar

The **menu bar** contains the various application menus. The following conventions apply:

- If commands in a menu are **not** applicable in an editor or at a particular location in the document, they are unavailable.
- Some menu commands pop up a sub-menu with a list of additional options. Menu commands with sub-menus are indicated with a right-pointing arrowhead to the right of the command name.
- Some menu commands pop up a dialog that prompts you for further information required to carry out the selected command. Such commands are indicated with an ellipsis (...) after the name of the command.
- To access a menu command, click the menu name and then the command. If a sub-menu is indicated for a menu item, the submenu opens when you mouseover the menu item. Click the required sub-menu item.
- A menu can be opened from the keyboard by pressing the appropriate key combination. The key combination for each menu is **Alt+***KEY*, where *KEY* is the underlined letter in the menu name. For example, the key combination for the **File** menu is **Alt+F**.
- A menu command (that is, a command in a menu) can be selected by sequentially selecting (i) the menu with its key combination (see previous point), and then (ii) the key combination for the specific command (**Alt+***KEY*, where *KEY* is the underlined letter in the command name). For example, to open a file (**File | Open**), press **Alt+F** and then **Alt +O**.
- Some menu commands can be selected **directly** by pressing a special **shortcut** key or key combination (**Ctrl+***KEY*). Commands which have shortcuts associated with them are indicated with the shortcut key or key combination listed to the right of the command. For example, you can use the shortcut key combination **Ctrl+N** to open a new SQL Editor; the shortcut key **F2** to rename a database object.

Toolbars

The toolbars contain buttons that are shortcuts for commands found in the menus. The name of the command appears when you place your mouse pointer over the button. To execute the command, click the button.

Toolbar buttons are arranged in groups. In the Tools | Customize | Toolbars dialog, or when right-clicking into the toolbar and selecting **Customize**, you can specify which toolbar groups are to be displayed. In the GUI, you can also drag toolbar groups by their handles (or title bars) to alternative locations on the screen. Double-clicking the handle causes the toolbar to undock and to float; double-clicking its title bar causes the toolbar to dock at its previous location.

Status Bar

The Status Bar is located at the bottom of the application window and displays (i) status information about the connection to data sources, the loading of database content, and the loading of files, and (ii) information about menu commands and command shortcuts in the toolbars when the mouse cursor is placed over these. If you are using the 64-bit version of DatabaseSpy, this is indicated in the status bar with the suffix (x64) after the application name. There is no suffix for the 32-bit version.

3.7   Arranging the Information Windows

All information windows can be docked/undocked by double-clicking the title bar. When docked, the auto-hide feature can be activated by clicking the drawing-pin icon in the title bar. When auto-hidden, the window is minimized as a tab at an edge of the application window. An auto-hidden window can be re-docked by rolling it out from the edge (by mousing over its tab) and clicking the drawing-pin icon in the title bar. When you click the little arrow that is located to the right of a data source connection or favorite item and shows the respective object in the Online Browser, DatabaseSpy will automatically hide the Project window and roll out the Online Browser to display the object.

Context Menu
The context menu can be accessed by right-clicking a window tab or title bar.
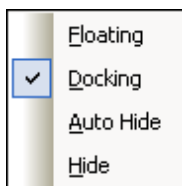
Click the required option to cause that window to float, be docked, or be auto-hidden or hidden.

Drag-and-drop
You can drag a window by its tab or title bar and place it at a desired location.
Additionally, you can dock the window in another window or in the interface using placement controls that appear when you drag a window:

- When you drag a window over another window, a placement control appears (*see screenshot below*). This control is divided into five placement sectors. Releasing the mouse key on any of these sectors docks the dragged window into the respective sector of the target window. The four arrow sectors dock the dragged window into the respective sides of the target window. The center button docks the dragged window as a tab of the target window. You can also dock a window as a tab in another window by dragging it to the tab bar and dropping it there.
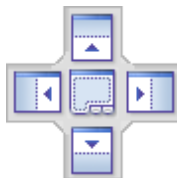
- When you drag a window, a placement control consisting of four arrows appears. Each arrow corresponds to one side of the Editor or Comparison window, respectively. Releasing a dragged window over one of these arrows docks the dragged window into one side of the Editor or Comparison window, respectively.
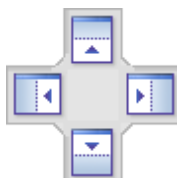
You can also double-click the title bar of a window to toggle it between its docked and floating positions.

Chapter 4

Tutorials

4   Tutorials

This section contains the following tutorials:

- DatabaseSpy QuickStart Tutorial: A tutorial that runs you through the main features of DatabaseSpy, that are: connecting to a database, browsing the database and viewing its structure, querying and editing data in the database, and importing and exporting data.
- DatabaseSpy Tutorial: A tutorial that provides you with in-depth information on all of DatabaseSpy's features. You will learn how to create a database from scratch and how to use DatabaseSpy to maintain the data in it by using the SQL Editor and the Design Editor.

4.1   DatabaseSpy QuickStart Tutorial

This tutorial takes you through several tasks which provide an overview of how to use DatabaseSpy, using an example Microsoft Access 2003 database.

You will learn how to:

- Connect to an existing Microsoft Access 2003 database
- Browse the data in the database
- View database structure and relationships graphically
- Query the database and create a view from that query
- Update data and add as well as delete table rows
- Generate a script to import data into the database
- Export data from the database to XML files

Installation and configuration

This tutorial assumes that you have successfully installed DatabaseSpy on your computer and received a free evaluation key-code, or are a registered user of the product. The evaluation version of DatabaseSpy is fully functional but limited to a 30-day period. You can request a regular license from our secure web server or through any one of our resellers.

Tutorial example files

The tutorial files are available in the \Altova\DatabaseSpy2017\DatabaseSpyExamples\Tutorial sub-folder of the (My) Documents folder. A project containing a connection to a Microsoft Access 2003 database is also available to you in the tutorial folder. The project also contains SQL scripts that will be used in the course of this tutorial.

4.1.1   Connecting to a Database

DatabaseSpy uses **projects** to organize databases and SQL files. Before you can connect to a database in DatabaseSpy, you must therefore create or open a database project. In order to provide you with a quick overview of the most important features of DatabaseSpy, a tutorial project has been added to the installation files and is available in the \Altova\DatabaseSpy2017 \DatabaseSpyExamples\Tutorial sub-folder of the (My) Documents folder. Instructions on how to create a new project from scratch can be found in the DatabaseSpy Tutorial.

Objective

In this section of the tutorial, you will learn how to connect to a database. Specifically, you will learn how to do the following:

- Open an existing database project
- Connect to a data source

Commands used in this section

Open project: This command is located in the Project window in the menu bar. Click this icon to display the standard Windows **Open** dialog box where you can choose an existing database project.

Connect: This command is located in the context menu that opens when you right-click a data source connection name. You can also double-click the data source connection name to connect to the database.

Opening an Existing Database Project

To connect to a database in DatabaseSpy, you must first start DatabaseSpy and then add a data source to a DatabaseSpy project. For the QuickStart Tutorial, a project is already available in the \Altova\DatabaseSpy2017\DatabaseSpyExamples\Tutorial sub-folder of the (My) Documents folder.

Starting DatabaseSpy

To start DatabaseSpy, double-click the DatabaseSpy icon on your desktop or use the **Start | All Programs** menu to access the DatabaseSpy program. DatabaseSpy is started with a new, empty project and displays the **Add a Data Source** dialog box by default. Close the dialog box to begin with the tutorial.
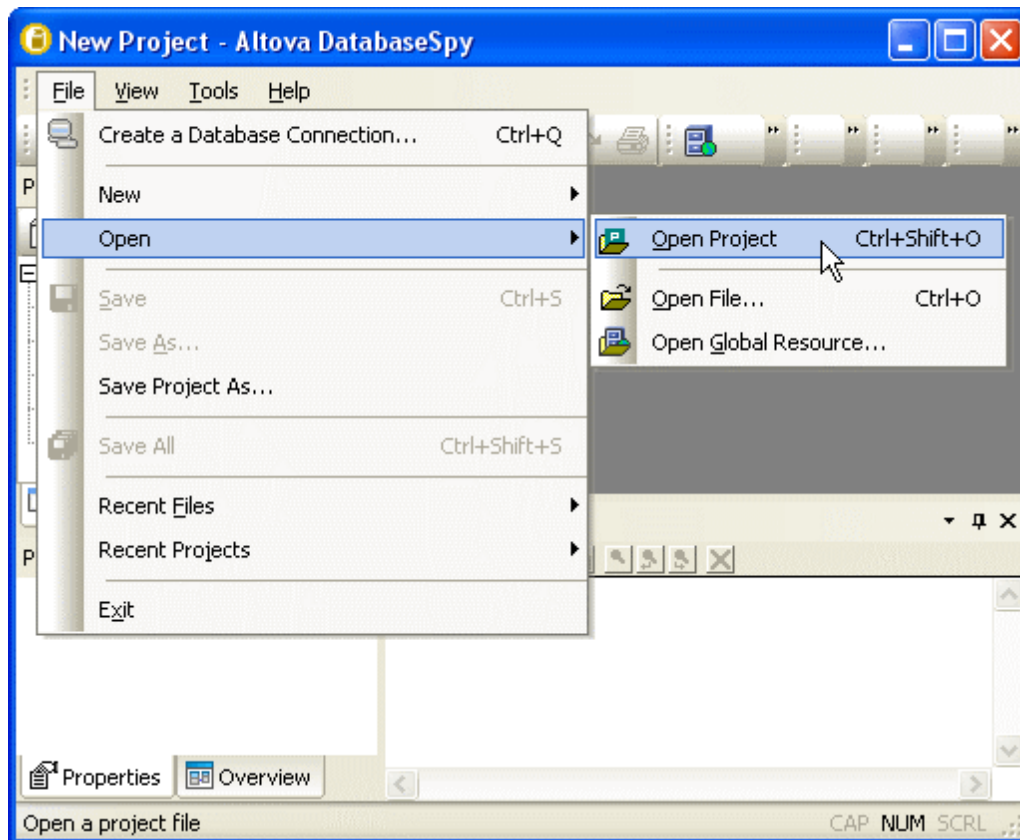
To open an existing database project:

1. Select the menu option **File | Open | Open project**.
   The **Open** dialog box is displayed.

2. Select the `access_db` project that is available in the tutorial folder<%DBSPY-TUT%> sub-folder of the (My) Documents folder.

3. Click **Open**.
   The access_db project appears in the Project window and the data source connection `TutorialDB` is displayed in the Data Sources folder.
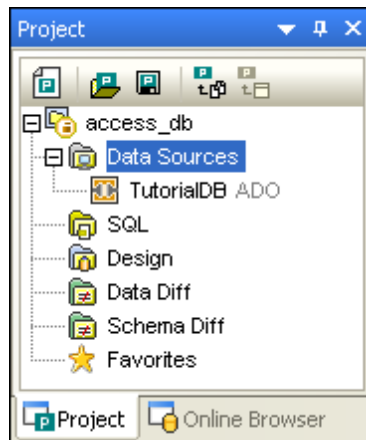
Connecting to a Data Source

Once you have created a project and added a data source to it, you must connect to this data source to gain access to your database.
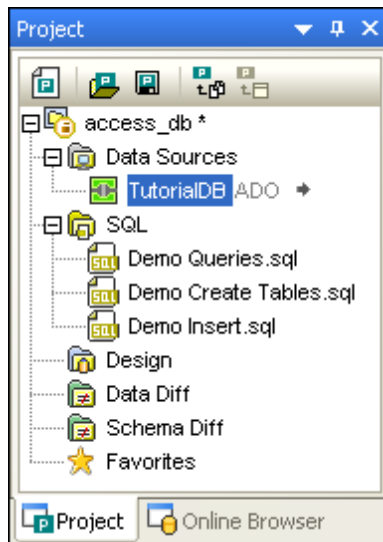
To connect to a data source:

- In the Project window, double-click the data source connection name `TutorialDB` in the Data Sources folder, or right-click it and choose **Connect** from the context menu.

  The color of the connection icon changes to green and an arrow appears to the right of it.

4.1.2   Browsing the Database

The **Online Browser** of DatabaseSpy displays all database objects of a connected data source. You can view the tables or retrieve data with a single mouse click, or navigate between the individual database objects using the navigation bar.

Objective

In this section of the tutorial, you will learn how to browse the structure and view the data of the example database in the Online Browser. Specifically, you will learn how to do the following:

- Display the database tables in the Online Browser

- Retrieve data from a specific table

Please note: By default, the Online Browser is available only for connected data sources. To be able to view disconnected database objects in the Online Browser, you must either first connect to the database in the Project window or activate the `Always show in Online Browser` check box in the data source properties.
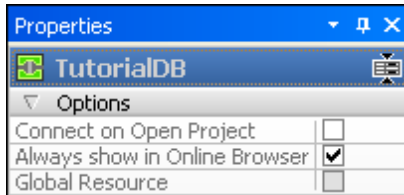
Commands used in this section

    Show in Online Browser: This command is located to the right of an active data source connection in the Project window. Click the arrow to activate the Online Browser tab.

    Retrieve data: This command is included in the context menu that opens when you right-click a database table or column. Choose a command from the sub-menu to query the selected table or column.

Displaying Tables in the Online Browser

The Online Browser displays the database structure hierarchically and allows for displaying the tables and columns included in a database as well as viewing and changing their properties.
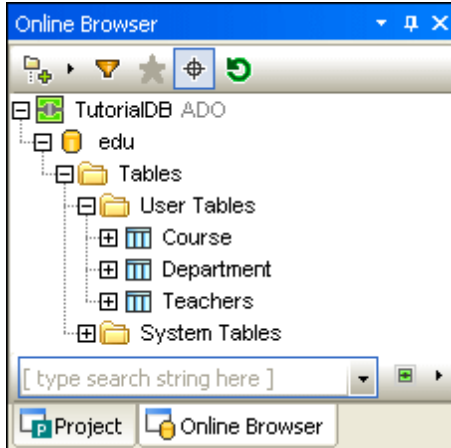
To be able to see the database structure in the Online Browser, you must connect to the data source connection.

To display the database tables in the Online Browser:

1. If the data source is not connected, connect to the database.

2. In the Project window, click the arrow to the right of the `TutorialDB` connection you just made active.

Alternatively, you can also click the Online Browser tab.
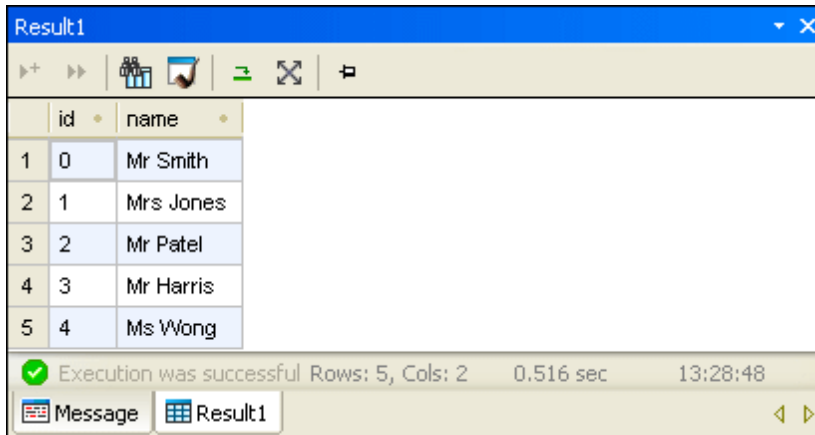
The Online Browser window is now active. The `Course`, `Department`, and `Teachers` tables that comprise the edu database are now visible in the User Tables folder.

Retrieving Data from Tables

The Online Browser makes it easy to retrieve and view data from any table. The context menu that opens when you right-click any database object in the Online Browser provides an option that opens an SQL Editor window, generates a query and automatically executes it.

```
1        SELECT [id],
2               [name]
3        FROM   [Teachers];
```

The query is shown in the top pane of the SQL Editor and the retrieved data is shown in the Result tab below.

To retrieve data from the Teachers table:

- In the Online Browser, right-click the Teachers table and select **Retrieve data | All rows** from the context menu. Alternatively, select the table and press the keyboard shortcut **Ctrl+Alt+R**.
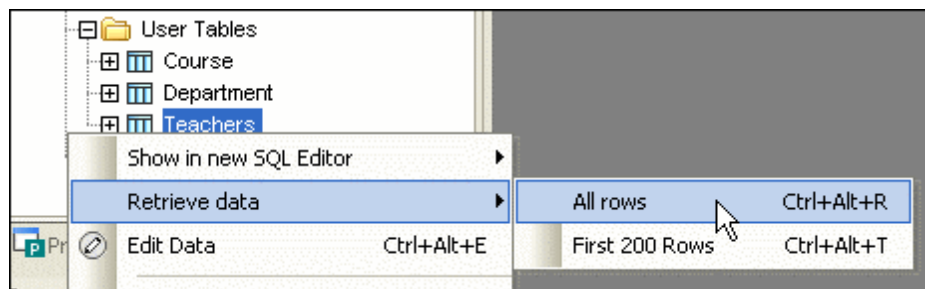
The SQL statement and the results of the query appear in a new SQL Editor window. You might have to click the maximize button of the SQL Editor window to see more of the data.

4.1.3   Viewing the Database Structure

DatabaseSpy's **Design Editor** allows for a graphical display of individual database objects or even the entire database structure. You can drag and drop database elements into a Design Editor window or select the respective option from the context menu. The content of Design Editor

windows can be saved to design files or images.

Objective

In this section of the tutorial, you will learn how to display a graphical overview of the database structure in the Design Editor. Specifically, you will learn how to do the following:

- Open a Design Editor window from the toolbar
- Drag and drop database elements into a Design Editor window
- Optimize the layout of a Design Editor window

Commands used in this section

Design Editor: This command is located in the **File | New** menu and is also available as a toolbar icon in the Standard toolbar. Selecting this command opens a new Design Editor window assigned to the active data source connection.

Show in new Design Editor: This command is located in the context menu of the selected database object. Choosing this command displays the selected database object in a new Design Editor window.

Hierarchical layout: This command is available as toolbar icon in the toolbar of the Design Editor window. Alternatively, you can also use the menu option **Design Editor | Auto Layout whole Diagram** to show you database objects in the Design Editor in a properly laid-out way.

Showing Tables in the Design Editor

You have two possibilities to show tables in the Design Editor: Either you first open a new Design Editor window in the **File** menu or from the Standard toolbar and then drag and drop the selected database elements, or you right-click the database elements in the Online Browser and choose **Show in new Design Editor** from the context menu.

To open a new Design Editor window:

- Click the **Design Editor** icon in the Standard toolbar, or select the menu option **File | New | Design Editor**, or press **Ctrl+D**.
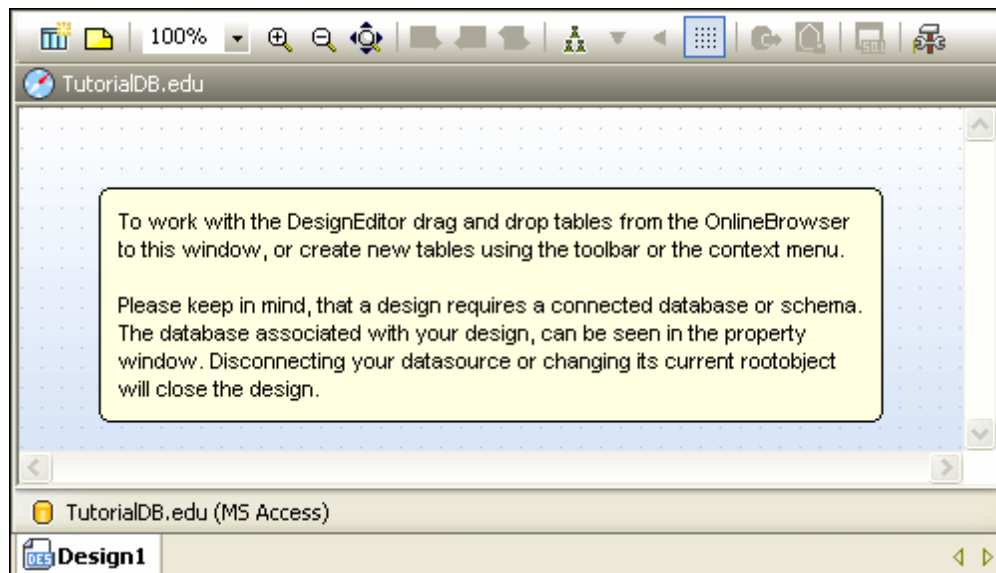
An empty Design Editor window is displayed and assigned to the active data source connection `TutorialDB`. Note that the window contains a brief instruction on how to get started with the Design Editor. This text box will disappear as soon as you drag database objects into, or create a new table in, the Design Editor.

To show tables in the Design Editor:

- In the Online Browser, select the `Course`, `Department`, and `Teachers` tables and drag and drop the tables into the Design Editor window. You can do this by dragging them individually, or using **Ctrl+click** to select them as a group.
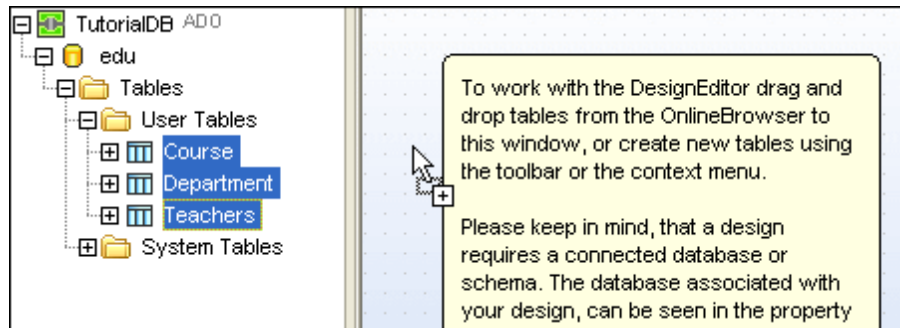
Alternatively, you can also right-click the tables and choose **Show in new Design Editor** from the context menu.

To auto-layout the Design Editor window:

- In the toolbar of the Design Editor, click the **Hierarchical Layout** icon to optimize the formatting.

4.1.4   Querying the Database

DatabaseSpy allows you to query a database by choosing a menu option from the context menu in the Online Browser or to execute SQL scripts directly from the Project window.

Objective

In this section of the tutorial, you will learn how to query a database and how to create a view. Specifically, you will learn how to do the following:

- Execute a SELECT statement provided as an SQL script
- Create a view from that statement
- Retrieve data from a view

Commands used in this section

Open: This command is located in the context menu of SQL files stored in the SQL folder and opens the selected file in the SQL Editor. It is also the default option when double-clicking these files.

Execute SQL: In the Project window, this command is located in the context menu of SQL files stored in the SQL folder. If an SQL Editor window containing a script is already open, this command can also be called from the **SQL Editor** menu or by clicking the **Execute** button in the toolbar of the SQL Editor window. Choosing this command immediately executes the SQL and shows the results in a Result tab.

Create View As: This command is located in the context menu that opens when you

right-click anywhere in an SQL Editor window. Alternatively, you can also select the
menu option **SQL Refactoring | Create View As**. Choosing this option creates a view of
the query in the active SQL Editor window and stores that view in the Views folder.

Refresh: This command is located in the context menu that opens when you right-click a
database in the Online Browser and it is also available as an icon in the toolbar of an SQL
Editor window. Clicking this menu option refreshes the display of the database in the
Online Browser.

Current statement: This command is located in the toolbar of an SQL Editor window.
Click this icon to select the statement in which the cursor is currently located.

Executing an SQL Script

In DatabaseSpy, SQL scripts can be stored together with the database connection in the SQL
folder of the project. This way, scripts that you use frequently are at hand any time and you can
execute them with a double-click.
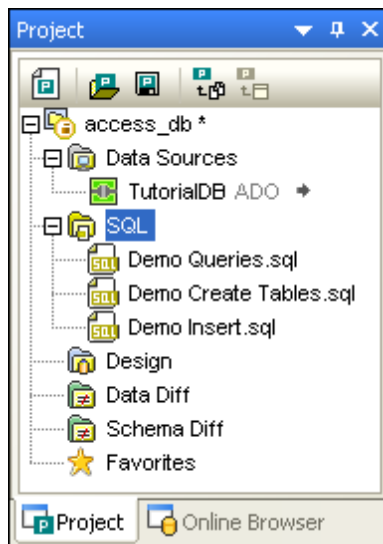
To execute an SQL script:

1. In the Project window, double-click the SQL folder if it is not already expanded.
    The SQL scripts in the project appear below the SQL folder.

2. Double-click the `Demo Queries.sql` entry. An SQL Editor window containing the
    script opens.

3. In the Properties window, make sure that "Semicolons" is selected in the `Group
    statements for execution with` drop-down list.

4. Click the **Execute** ▶ button or press **F5**.

The results of the query appear in a results tab named "Art Teachers".

Creating a View

A view is a stored query. Views are stored in a separate folder in the Online Browser and can be used like tables. You can use DatabaseSpy to create a view from a SELECT statement.

To create a view from the SELECT statement:

1. In the SQL Editor window, right-click into the statement and select **Create View As** from the context menu. Alternatively, select the menu option **SQL Refactoring | Create View As**.
    The statement is changed into a CREATE VIEW statement, and the `View1` placeholder is automatically highlighted.

```
1      -- All Teachers in the Art Department
2      -- target: Art Teachers
3
4      CREATE VIEW [View1] AS SELECT Teachers.name as
       [Teacher Name], Course.name as [Course Name] FROM
       Teachers, Course WHERE Course.teacherid = Teachers.
       id AND Course.deptid = 2;
```

2. Change the view name from "View1" to "ArtTeacherView".

```
1        -- All Teachers in the Art Department
2        -- target: Art Teachers
3
4        CREATE VIEW [ArtTeacherView] AS SELECT Teachers.name
          as [Teacher Name], Course.name as [Course Name]
         FROM Teachers, Course WHERE Course.teacherid =
         Teachers.id AND Course.deptid = 2;
```

3. In the Properties window, make sure that "Semicolons" is selected in the `Group statements for execution with` drop-down list.

4. Click the **Execute** [▷] button or press **F5**.
    A view is created in the edu database. Execute the statement to get a table that shows which teachers are teaching which courses.

5. Right-click the edu database in the Online Browser and select **Refresh** from the context menu or click the **Refresh** button in the toolbar of the Online Browser.
    The Online Browser shows a new folder called Views and the newly created view appears in this folder (you may have to double-click the Views folder to see the view).

```
☐ 🔷 TutorialDB ADO
  ☐ 🟠 edu
     ☐ 📁 Tables
        ☐ 📁 User Tables
           ⊞ 🎟 Course
           ⊞ 🎟 Department
           ⊞ 🎟 Teachers
        ⊞ 📁 System Tables
     ☐ 📁 Views
        ⊞ 👓 ArtTeacherView
```
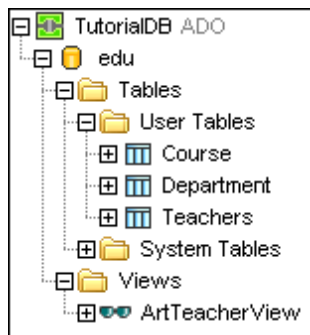
Retrieving Data from a View

Creating views considerably facilitates the querying of a database. Let's assume that you frequently need to select the Art teachers and their courses from our tutorial edu database. A view allows you to query the view as you would query any table, i.e., by creating a view named "ArtTeacherView", instead of typing the statement

```
SELECT Teachers.name as [Teacher Name], Course.name as [Course Name]
FROM Teachers, Course WHERE Course.teacherid = Teachers.id AND
Course.deptid = 2
```

you can simply type

```
SELECT * FROM [ArtTeacherView]
```

and will receive the same results in the Result tab. What is more, you can also use the context menu options in the Online Browser to retrieve data or generate SQL statements.

To enter a SELECT statement for a view:

1. Add a semicolon to the CREATE VIEW statement in the SQL Editor window and hit **Enter**.

2. Enter the statement `SELECT * FROM [ArtTeacherView]`

3. To execute only the SELECT statement in the SQL Editor, select the statement using

the mouse or the **Current statement** ![icon] icon in the toolbar.

```
1      -- All Teachers in the Art Department
2      -- target: Art Teachers
3
4      CREATE VIEW [ArtTeacherView] AS SELECT
       Teachers.name as [Teacher Name], Course.name
       as [Course Name] FROM Teachers, Course WHERE
       Course.teacherid = Teachers.id AND Course.
       deptid = 2;
5      SELECT * FROM [ArtTeacherView]
```

4. In the Properties window, make sure that "Semicolons" is selected in the `Group statements for execution with` drop-down list.

5. Click the **Execute** ![button] button or press **F5**.
    The results of the executed SQL statement are displayed in the Result1 tab.

To retrieve data from a view:

*   In the Online Browser, right-click the `ArtTeacherView` and select **Retrieve data | All rows** from the context menu. Alternatively, you can also select the view and press **Ctrl +Alt+R**.

    A SELECT statement is generated in a new SQL Editor window and executed automatically. The results of the query are displayed in the Result1 tab.

4.1.5   Updating Database Data

In DatabaseSpy, you can edit table content directly in the Result window of the SQL Editor. For this purpose, you have to switch the Result window into the Editing mode first. Then you can change the content of the individual cells and add or delete rows in the result grid.

Objective

In this section of the tutorial, you will learn how to switch the Result window into the Editing mode and how to edit database data in the Result grid. Specifically, you will learn how to do the following:

*   Select a table for editing in the Result window
*   Execute a SELECT statement to make data editable in the Result window
*   Change the content of individual database fields
*   Add an additional row of database data to the result grid
*   Delete a row of database data in the result grid

Commands used in this section

![Edit Data icon]   **Edit Data**: This command is located in the context menu that opens when right-clicking a table or column in the Online Browser. Choose this command to generate an SQL SELECT statement that is automatically executed in the SQL Editor and switches the Result window into the Editing mode. This command works for all supported database kinds for both ADO and ODBC connections.

![Execute for Data Editing icon]   **Execute for Data Editing**: This command is located in the **SQL Editor** menu and is available as an icon in the SQL Editor toolbar. Click this command to execute a SELECT

statement in the SQL Editor and switch the Result window into the Editing mode. Please note that this command works only for SELECT statements, and may be available only for a limited number of database kinds when connected via ADO connections.

Commit : This command is available as a button in the Result window toolbar. Click this button to commit the changes you have made in the result grid to the database. The **Commit** button is only visible if the Result window is in the Editing mode.

Append a new row: This command is available as a button in the Result window toolbar. Click this button to add a new row to the result grid. Alternatively, you can also right-click anywhere in the result grid and choose **Append a new row** from the context menu. The **Append a new row** command is only available if the Result window is in the Editing mode.

Delete row: This command is available as a button in the Result window toolbar. Click this button to delete the row in which the cursor is currently located. Alternatively, you can also right-click a cell in the row to be deleted and choose **Delete row** from the context menu. The **Delete row** command is only available if the Result window is in the Editing mode.

Selecting a Table for Editing

In order to be able to modify the table content in your tutorial database, you have to switch the Result window of the SQL Editor into the Editing mode. Whether or not data can be edited in the result grid is indicated by the status bar of the Result window.

The easiest way to modify table data in the Result window is to right-click the respective table in the Online Browser and choose the **Edit Data** command from the context menu (*see screenshot*).
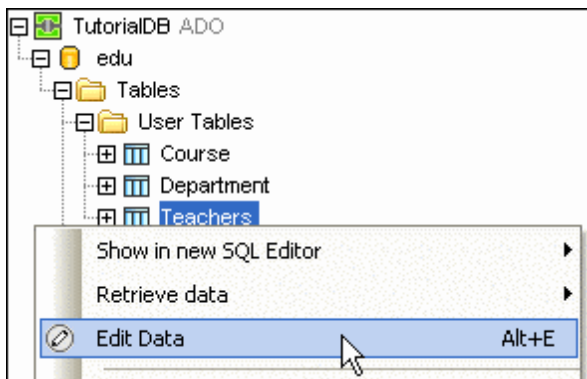
DatabaseSpy creates a SELECT statement in a new SQL Editor, executes it immediately, and switches the Result window into the Editing mode. The Editing mode is indicated in the Status bar and the Result window toolbar shows four additional icons.

You can now double-click or right-click a cell in the result grid and update the table content. Some columns may not be editable because they are identity columns or are computed using content from other columns, for instance.

To select the Teachers table for editing:

- In the Online Browser, right-click the Teachers table and select **Edit Data** from the context menu. Alternatively, you can also select the table and press **Ctrl+Alt+E**.

Executing for Data Editing

DatabaseSpy provides an additional way to switch the Result window into the Editing mode: You can execute a SELECT statement that is already present in an SQL Editor window for editing. Please note, however, that this option may be restricted to ODBC connections for some databases.

When supported for the currently used data source connection, the **Execute for Data Editing** command is available in the **SQL Editor menu** and as a button in the SQL Editor toolbar. You can start the editing from the Online Browser like any normal retrieval.
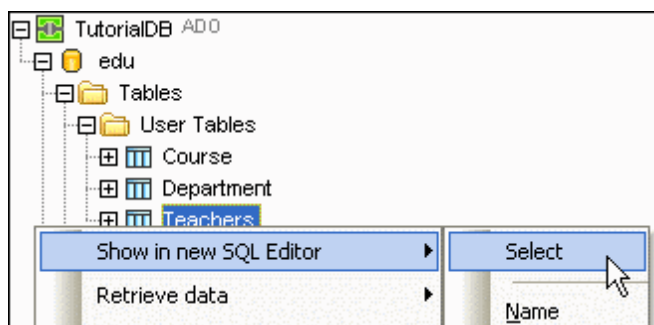
DatabaseSpy creates a SELECT statement in a new SQL Editor window.

```
1        SELECT [id],
2               [name]
3        FROM   [Teachers];
```

You can now choose from among the following options:

- Click the **Execute** button for a normal retrieval.

- Click the **Execute for Data Editing** 🖉 button to switch the Result window into the Editing mode and allow for updating the table data.
  Only one SELECT statement may be present in the SQL Editor window at a time. Please note that editing of data may be restricted to particular columns or—depending on the data source connection you are using—may not be possible at all. Please see chapter Editing database data for details.

To execute a SELECT statement for data editing:

1. In the Online Browser, right-click the Teachers table and select **Show in new SQL Editor | Select** from the context menu.

2. In the SQL Editor, click the **Execute for Data Editing** 🖉 button in the SQL Editor toolbar. Alternatively, you can also choose the menu option **SQL Editor | Execute for Data Editing**.

Updating Table Content

When the Result window is in the Editing mode, you can edit the data that is displayed in the result grid like in any common spreadsheet application. Double-click a cell in the result grid and update the content as required. Set a database field to NULL or assign the default value with a click of the mouse. You can also undo the changes that you have made to a cell.
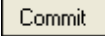
Let's assume that you need to change the course structure in your edu database that you use in this tutorial. First, select the Course table for editing to see the current structure and switch the Result window into the Editing mode so that you are able to make your changes directly in the result grid.

	id	name	descr	deptid	teacherid	upperlimit
1	0	Chemistry 101	First year Chemistry	0	2	70
2	1	Algebra 101	First year Algebra	1	0	65
3	2	Biology 102	First year Biology	0	1	70
4	3	Calculus 102	First year Calculus	1	0	65
5	4	Biology 202	Second year Biology	0	1	40
6	5	Music 101	Entry level instrumental music	2	3	30
7	6	Music 102	Entry level vocal	2	4	15
8	7	Art 101	Entry level visual arts course	2	4	40

Now make the following changes to the Course table:
- Change the number in the name column so that its three digits reflect the year, the department, and a consecutive number
- Set the teacherid column for "Entry level instrumental music" to NULL (let's assume the teacher retired and no replacement has been nominated yet)
- Set the upper limit for "First year Calculus" to the default of "70".

	id	name	descr	deptid	teacherid	upperlimit
1	0	Chemistry 101	First year Chemistry	0	2	70
2	1	Algebra 111	First year Algebra	1	0	65
3	2	Biology 102	First year Biology	0	1	70
4	3	Calculus 112	First year Calculus	1	0	70
5	4	Biology 201	Second year Biology	0	1	40
6	5	Music 121	Entry level instrumental music	2	*NULL*	30
7	6	Music 122	Entry level vocal	2	4	15
8	7	Art 123	Entry level visual arts course	2	4	40

The changes have so far only be made in the result grid, they have not been committed to the database yet. DatabaseSpy provides the **Undo changes for this cell** command in the context menu to undo the changes for an individual cell, and furthermore allows you to also undo all changes that have been made. In the toolbar, the **Undo all** button is activated for this purpose. To save the changes and commit the new data to the database, you have to click the Commit button.

To update the Course table in the Result window:

1. In the Online Browser, right-click the Course table and select **Edit Data** from the context menu.

2. In the Result window, double-click the respective fields in the name column and change the subject numbers so that they reflect the following pattern: first digit: year, second digit: department id, third digit: consecutive number.

3. Right-click the teacherid field in the sixth row (Entry level instrumental music) and choose **Set Null** from the context menu.

4. Right-click the `upperlimit` column in the fourth row (First year Calculus) and choose **Set Default** from the context menu.

5. Click the Commit button to save the changes and commit the data to the database.

Adding Table Rows

You can add new records to a table by appending a new row to the grid in the Result window and entering the required data. DatabaseSpy provides an option in the SQL Editor options to automatically enter the default values of the individual columns into the new row.

To add a new subject to the Course table:

1. Make sure that the Result window is still in the Editing mode and the content of the Course table is displayed.

2. Optionally, select the menu option **Tools | Options** and activate the `Initialize cells with default values` check box in the Data Editing group box of the SQL Editor: Result View section.
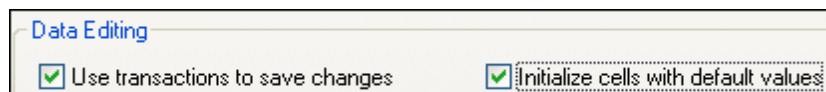
3. In the Result window, click the **Append a new row** icon in the toolbar.

7	6	Music 102	Entry level vocal	2	4	15
8	7	Art 101	Entry level visual arts course	2	4	40
9	0	*NULL*	*NULL*	0	0	70

Note that the default value of "70" has been automatically inserted into the upperlimit column. All other columns do not have a default value defined, therefore "0" or "NULL", respectively, is inserted. If the inserting of default values option is deactivated in the SQL Editor options, a blank line is inserted.

4. Using the Tab key to jump to the next field, enter the following values:

| 8 | Music 221 | Intermediate level vocal | 2 | 4 | 15 |

5. Click the Commit button to save the changes and commit the data to the database.

Deleting Table Rows

Unneeded table rows can be easily deleted when the Result window is in the Editing mode. The row is first marked for deletion in the result grid and you need to confirm the deletion by clicking the Commit button.

Let's assume that the "Entry level visual arts course" will not be available for students anymore and has to be removed from the list of courses in the tutorial database. To do so, place the cursor in a field of the row and click the **Delete row** button in the toolbar. The row is marked for deletion in the result grid.

6	5	Music 101	Entry level instrumental music	2	3	30
7	6	Music 102	Entry level vocal	2	4	15
8	7	Art 101	Entry level visual arts course	2	4	40

Note that the row is still present in the database and that you can still undo the deletion by clicking into the deleted row and choosing **Undo Changes for this Cell** from the context menu.

To actually delete the row from the table, you have to click the Commit button. The row is removed from the result grid and no longer available in the database.

To delete a course from the Course table:

1. Make sure that the Result window is still in the [Editing mode](#) and the content of the Course table is displayed.

2. In the Result window, place the cursor in any cell of the eighth row (Entry level visual arts course).

3. Click the **Delete row** button in the toolbar.
   The row is marked for deletion in the result grid.

4. Click the Commit button to delete the data from the database.

4.1.6 Importing Data

Data contained in CSV files can be imported to a database. DatabaseSpy provides an import dialog box that allows you to preview the data before actually importing them and to choose whether the import is immediately executed or an SQL script should be generated which can be executed later.

Objective

In this section of the tutorial, you will learn how to import data to a database. Specifically you will learn how to do the following:

- Preview the import of a CSV file
- Generate the script necessary to carry out the import

You will not need to actually import the data, as the supplied database already contains it.

Commands used in this section

Import data to the database: This command is located in the **Tools** menu and is available as an icon in the Tools toolbar. Clicking this icon opens the **Import data to the database** dialog box.

Generating an Import Script

All the options needed for importing data to a database can be defined in the **Import data to the database** dialog box. In this dialog box, you can also choose between immediate import and generation of an SQL script. In this tutorial, we will generate an SQL script since the data in the supplied import file is already contained in the database.
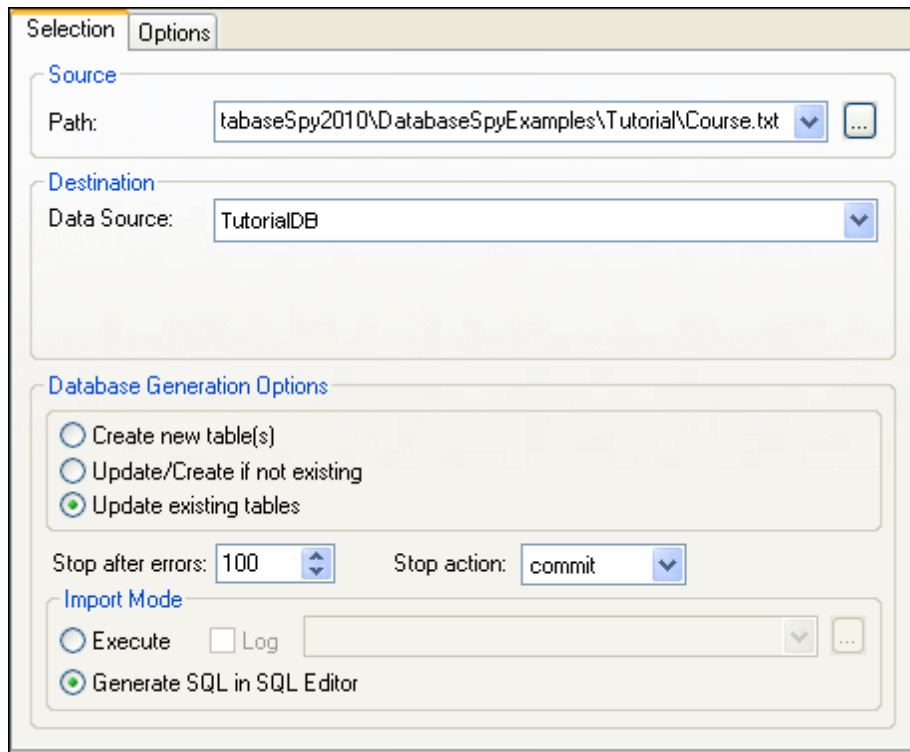
To generate an import script:

1. If you are not connected, connect to the TutorialDB data source connection.

2. Select the menu option **Tools | Import data to the database...**, or click the **Import** button in the Tools toolbar, or press **Ctrl+I**.
    The **Import data to the database** dialog box is displayed.

3.   Click the **CSV** icon on the left side.

4.   In the Source group box, enter the path to the source file Course.txt, which is located in the tutorial folder, or click the ⊡ button to browse to the file.

5.   In the Destination group box, choose the `Update existing tables` radio button.

6.   In the Import Mode group box, choose the `Generate SQL in SQL Editor` radio button.

7.   Click the **Apply Options** button in the Preview group box.
     The Course data and column names are shown in the preview window.
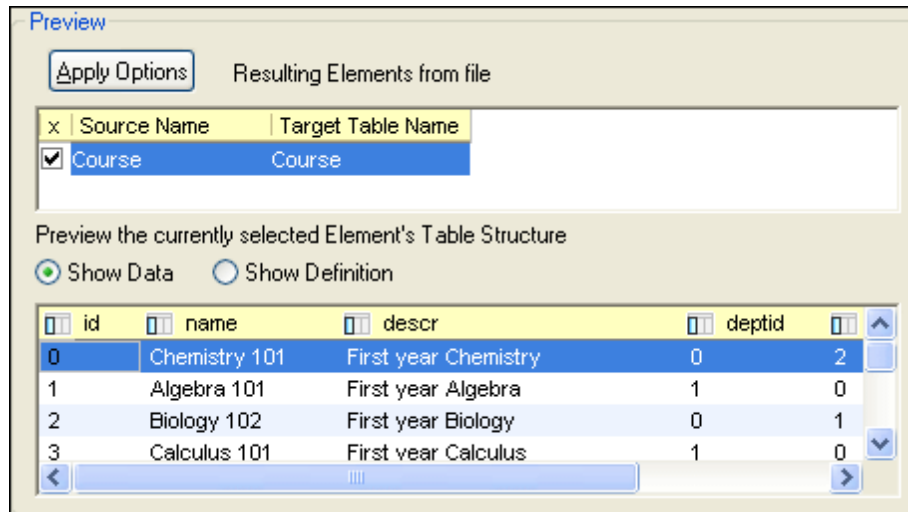
8.   Click the **Generate SQL** button.
     A script to insert data is generated and a dialog box appears informing you about the success of the SQL generation.

9.   Click **OK** to switch to the newly created SQL Editor window.
     The SQL script contains an INSERT statement for each row that is to be inserted into the `Course` table.

4.1.7   Exporting Data

DatabaseSpy allows you to export your database data to several types of files. In this tutorial, we will look at how to export data to XML files.

Objective

In this section of the tutorial, you will learn how to export database data. Specifically, you will learn how to do the following:

- Select tables for export
- Preview data before exporting
- Export columns as attributes
- Change column names for the XML export file
- Export the complete edu database to XML files

Commands used in this section

Export database data: This command is located in the **Tools** menu and is available as an icon in the Tools toolbar. Clicking this icon opens the **Export database data** dialog box.

Exporting to XML

In this tutorial, you will export the entire edu tutorial database to XML files. In the **Export database data** dialog box, you can define the path for the XML files and preview the individual tables prior to actually exporting them. The Preview window also allows for a selection of the columns to be exported as well as for renaming the column header for the XML files.
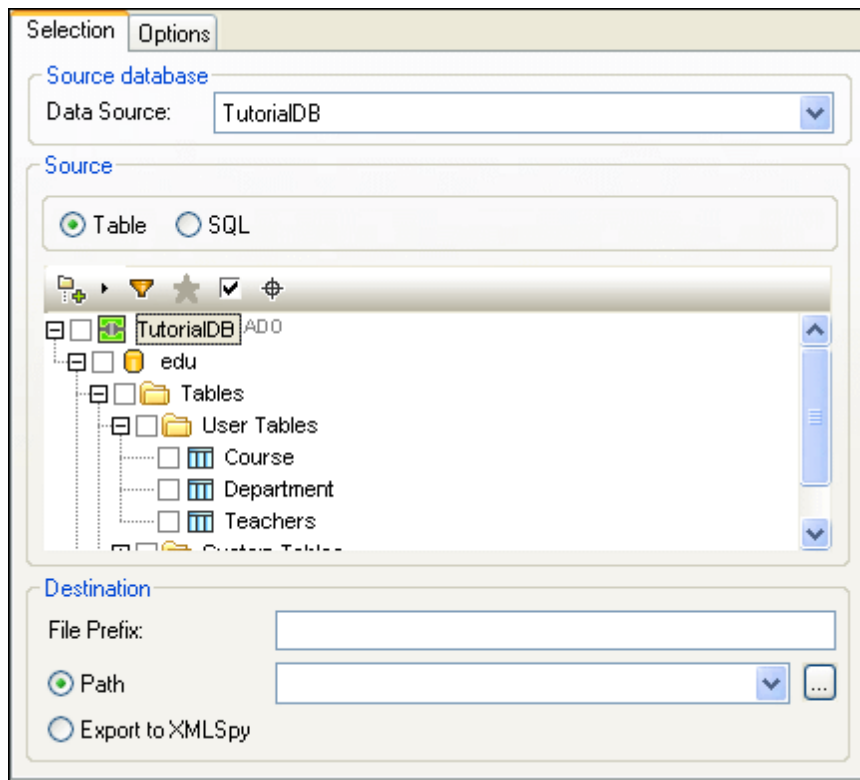
To export the edu database:

1. If you are not connected, [connect] to the `TutorialDB` data source connection.

2. Select the menu option **Tools | Export database data...**, or click the **Export** button in the Tools toolbar, or press **Ctrl+E**.

3. In the **Export data base data** dialog box, click the **XML** icon on the left side of the

dialog box.

4. In the Source database group box, select the Data Source `TutorialDB` from the drop-down list, if it has not been automatically selected.

5. In the Source group box, choose the `Table` radio button.

6. Select the `Course`, `Department` and `Teachers` tables, by clicking the respective check boxes.
   Alternatively, since you want to select all of the available tables, you can also click the `User Tables` check box. This will automatically select all user tables.
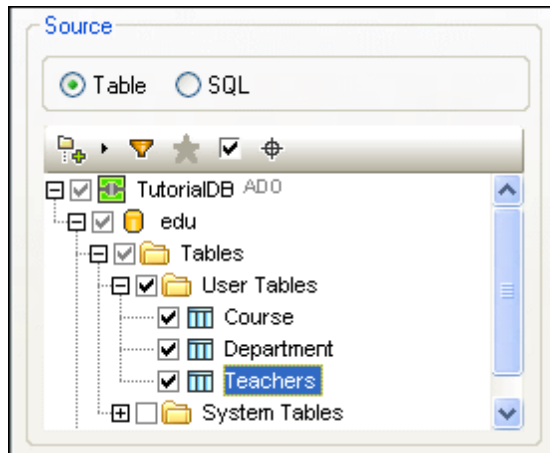
7. In the Destination group box, enter "edu_" in the `File Prefix` field.

8. Enter the path you want the XML files to be exported to.

9. Select the `Course` table in the Source group box and click the **Preview** button in the Preview group box. The contents of the `Course` table appear in the list box below.
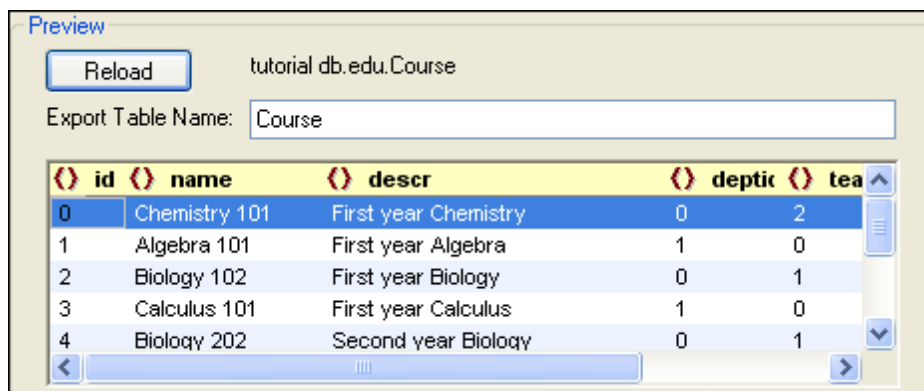
10. To export the `descr` column as an attribute, click the icon next to the "descr" table header ❰❱ until it changes to an equals sign =.

11. To rename the `descr` column, double-click the text "descr" and change it to "Description", press **Enter** to confirm.

12. To exclude the `upperlimit` column from the export, click the scrollbar to scroll to the right, then click the icon next to the "upperlimit" table header until it changes to a ✕ icon.
    This column will not be exported.

= Description	() deptid	() teacherid	✗ upperlin
First year Chemisty	0	2	70
First year Algebra	1	0	65
First year Biology	0	1	70
First year Calculus	1	0	65
Second year calculus	0	0	40

13. Click the **Export** button.

Checking the results

Open the folder that you exported the XML files to. You will see three files: edu_Course.xml,
edu_Department.xml, and edu_Teachers.xml.

Name ▲	Size	Type	Date Modified
edu_Course.xml	2 KB	XML Document	2007-09-20 3:55 PM
edu_Department.xml	1 KB	XML Document	2007-09-20 3:55 PM
edu_Teachers.xml	1 KB	XML Document	2007-09-20 3:55 PM

Open the file `edu_Course.xml`. As can be seen in the file, "Description" appears in the file as
an attribute of each row, and the column "upperlimit" was not exported.

```
10   <Row Description="First year Chemisty">
11     <id>0</id>
12     <name>Chemistry 101</name>
13     <deptid>0</deptid>
14     <teacherid>2</teacherid>
15   </Row>
16   <Row Description="First year Algebra">
17     <id>1</id>
18     <name>Algebra 101</name>
19     <deptid>1</deptid>
20     <teacherid>0</teacherid>
21   </Row>
22   <Row Description="First year Biology">
23     <id>2</id>
24     <name>Biology 102</name>
25     <deptid>0</deptid>
26     <teacherid>1</teacherid>
27   </Row>
```

4.2   **DatabaseSpy Tutorial**

This tutorial takes you through several tasks which provide an overview of how to use DatabaseSpy™ 2017 to its fullest.

The goal of this tutorial is to use DatabaseSpy to create a database that models a zoo. SQL scripts and other files are provided to help you complete this task. You will create the database from scratch, and then use scripts and text files (*.TXT) to populate it with data.

In this tutorial, you will learn how to:

- Use project files to manage the connections and SQL files relevant to your database project
- Use DatabaseSpy to connect to a Microsoft SQL Server database
- Create and add tables to a database using SQL as well as DatabaseSpy's Design Editor
- Create relationships between those tables
- Add constraints to the tables
- Add data to the database using SQL scripts as well as DatabaseSpy's import and editing functionality
- Browse the database using the Online Browser
- Generate SQL statements using features of the SQL Editor such as generating statements based on existing objects, and autocompletion
- Export data to XML files
- Compare two schemas of a database
- Compare your database data with a backup database

Installation and configuration

This tutorial assumes that you have successfully installed DatabaseSpy on your computer and received a free evaluation key-code, or are a registered user of the product. The evaluation version of DatabaseSpy is fully functional but limited to a 30-day period. You can request a regular license from our secure web server or through any one of our resellers.

Tutorial example files

The following files are available in the \Altova\DatabaseSpy2017\DatabaseSpyExamples\Tutorial sub-folder of the (My) Documents folder and are used in the Tutorial:

- create_ZooDB_tables.sql
- create_ZooDB_relationships.sql
- tblAnimalTypes_data.sql
- animalType_queries.sql
- tblZookeepers.txt
- dbo.tblFeedSuppliers.xml
- ZooDB.mdb
- CreateTestSchema.sql

Several other files are available if you want to optionally populate the tutorial Zoo database with data, see Populating the Zoo Database with Data (Optional).

4.2.1 Setting Up a Database Project

Project files organize everything you need in order to work on your database projects. You can store the following items in a project file:

- Database connections
- SQL scripts
- Design files
- Database Data Comparison files
- Database Schema Comparison files
- Favorite database objects

Objective

In this section of the tutorial, you will learn how to create a project in DatabaseSpy and connect to a database. Specifically, you will learn how to do the following:

- Create a new project and add a data source connection
- Add SQL files
- Change the project name
- Define the project startup options

Commands used in this section

Create a Database Connection: This command is located in the Standard toolbar and in the **File** menu. Click this command to open the **Add a Data Source** dialog box that allows you to connect to any database supported by DatabaseSpy.

Add files to the project: This command is located in the toolbar of the Project window. Click this command to open the **Open** dialog box where you can choose the files to be added to and stored with your database project.

Save Project: This command is located in the toolbar of the Project window; alternatively, you can also choose the menu option **File | Save Project As...** or press **Ctrl+S**. Click this command to save an existing project under its project name or to specify a name and path for a new project, respectively.

Options: This command is located in the **Tools** menu. Click this command to open the **Options** dialog box where you can define the project startup options.

Creating the Tutorial Database

Before you can start with this tutorial, a new sandbox database must be created in Microsoft SQL Server (in this tutorial, the database name is "ZooDB"). You can accomplish this task either with Microsoft SQL Server Management Studio, or from Visual Studio. To do this, connect to Microsoft SQL Server, and follow the instructions specific to your SQL Server database administration tool to create the database. For example, the following image illustrates creating a new database on a network SQL Server from Visual Studio 2015:
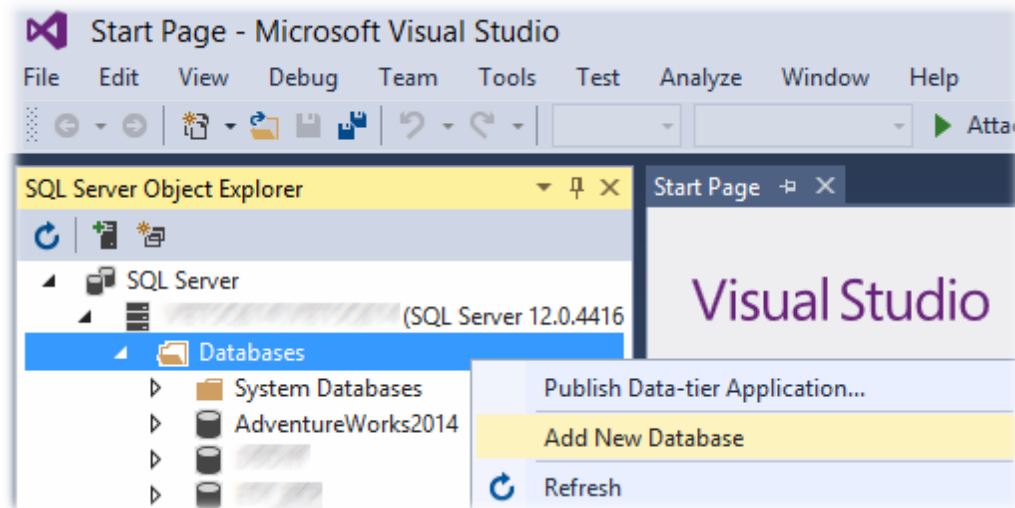
Notes:

- You will need administrator rights to create a database, please contact your database administrator if you do not have the respective rights.
- The new "ZooDB" database must be empty. All tables, relationships, and data will be added to this database in the subsequent steps of this tutorial.

Connecting to the Database

Once the database for this tutorial has been created (see Creating the Tutorial Database), you can connect to it using the Connection Wizard. The Connection Wizard allows you to establish a connection to existing databases of the most commonly used database types, such as Microsoft Access, Microsoft SQL Server, Oracle, MySQL, IBM DB2, Sybase, and PostgreSQL.

To connect to the tutorial database:

1. Select the menu option **File | Create a Database Connection...** or press **Ctrl+Q**. The **Add a Data Source** dialog box appears.

2. Click the **Connection Wizard** button at the top left of the dialog if it is not already selected, and select **Microsoft SQL Server (ADO)**.
3. Click **Next**. The configuration page of the Connection Wizard appears.
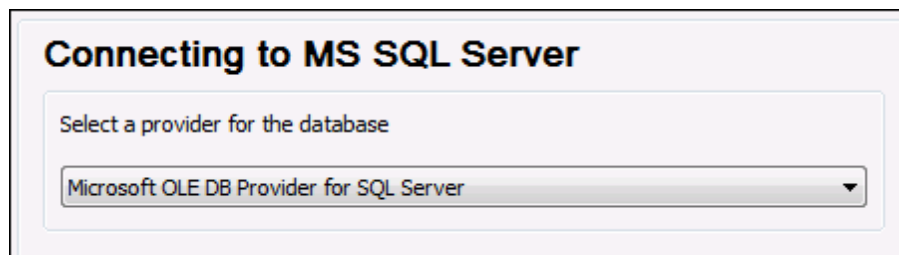
4. Select the database provider **Microsoft OLE DB Provider for SQL Server** from the combo box.
5. Click **Next**. The Connection tab of the **Data Link Properties** dialog box appears.
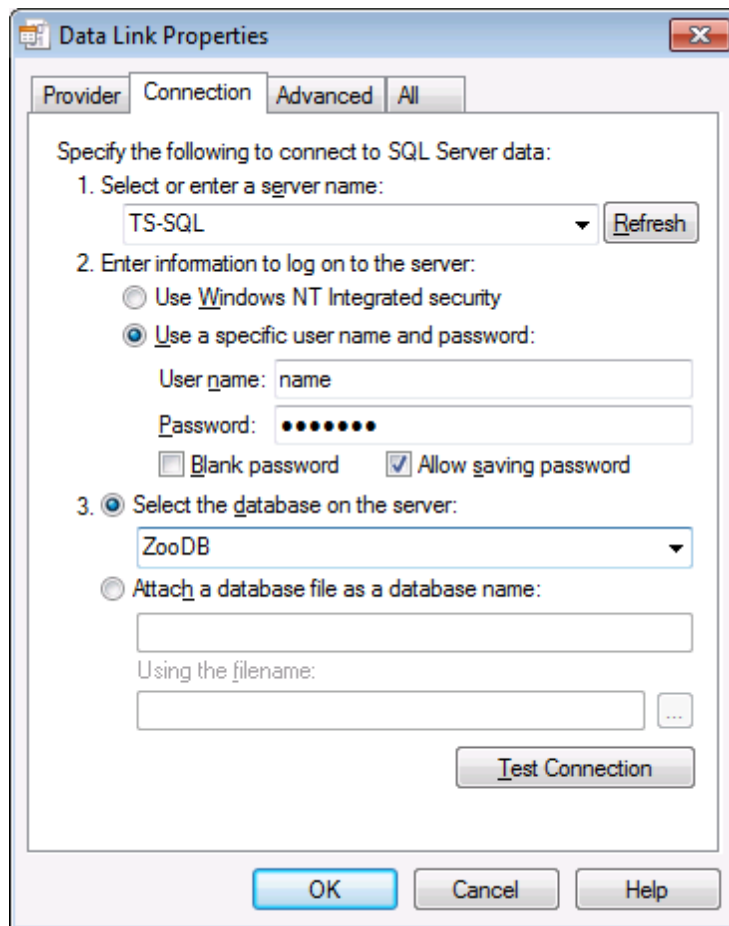
6. Select or enter the name of the server you want to connect to, e.g., `TS-SQL`.
7. Choose the `Use a specific user name and password` radio button.
8. Enter the User name and Password.
9. Activate the `Allow saving password` check box.
10. Select the database you want to connect to from the drop-down list, e.g., `ZooDB`, and click **OK**.
11. In the **Set A Data Source Name** dialog box, set the connection name to "ZooDBConnect" and click **OK**.
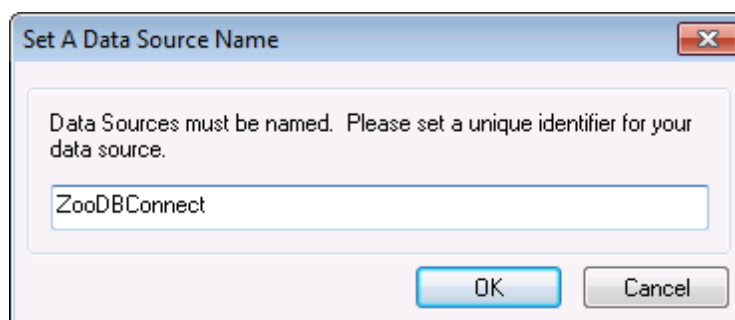
The Project window now contains a connection called "ZooDBConnect" in the Data Sources folder.
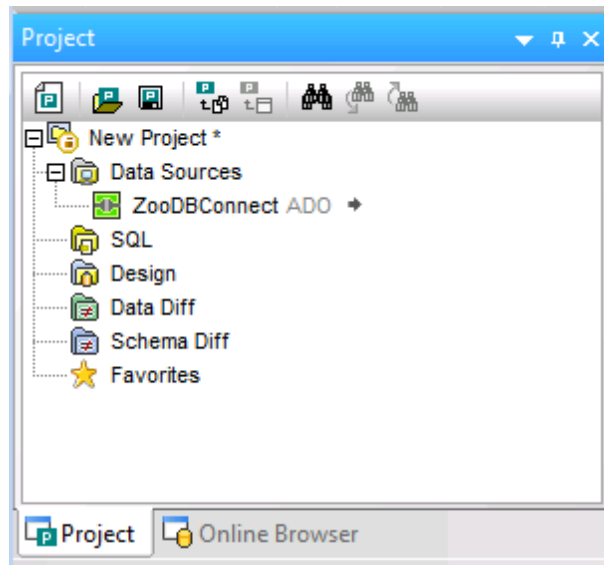
Adding SQL Files

A database project in DatabaseSpy allows you to store SQL files together with the database connection in one project file so that you have all the files and connections you need at hand when you open a project in DatabaseSpy.

In this step, you will add the SQL scripts to the project that you will later use to create the tables for the zoo database and to query the zoo database.

To add the SQL files to the project file:

1. In the Project window, right-click the SQL folder and select **Add Files to Project...** from the context menu. The **Open** dialog appears.
2. Select the files to be added to the SQL folder from the tutorial directory:

    - **animalType_queries.sql**
    - **create_ZooDB_relationships.sql**
    - **create_ZooDB_tables.sql**
    - **tblAnimalTypes_data.sql**

3. Click the **Open** button. The SQL files appear in the SQL folder of the Project window.
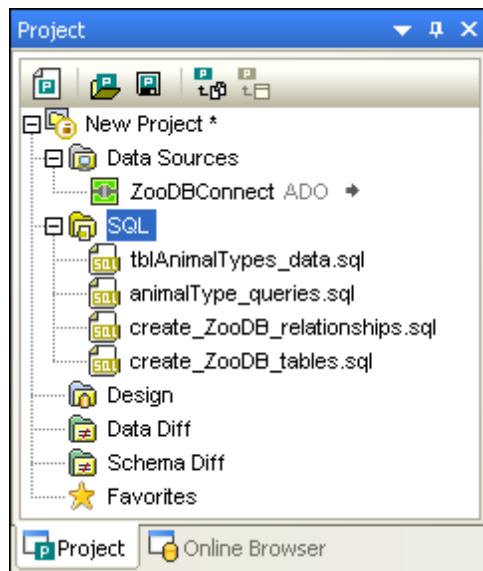
Renaming and Saving the Project

If a project has not been saved yet, `New Project *` is displayed as a project title in the Project window. You can change the project title by saving the project under a descriptive name. DatabaseSpy uses the name of the project file and displays it as project title in the Project window.

In this step of the tutorial, we will change the title of the project to "ZooDBProject" and save the project.
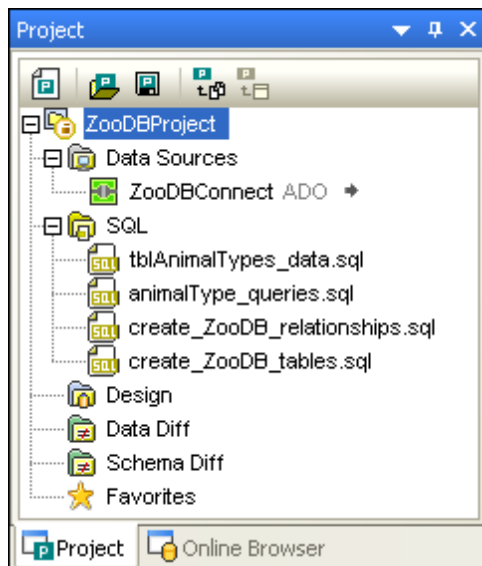
To save the project:

1. Select the menu option **File | Save Project As...**.
2. In the **Save As** dialog box, choose a path for the project and enter "ZooDBProject" as a

file name.

3. Click **Save**. The project is saved as ZooDBProject.qprj and the Project window displays `ZooDBProject` as project name.

Defining the Project Startup Options

In the General options, you can define that the last project be opened by default when DatabaseSpy is started. In this case, the **Add a Data Source** dialog box is suppressed and the project that was open when you last closed DatabaseSpy is opened automatically.
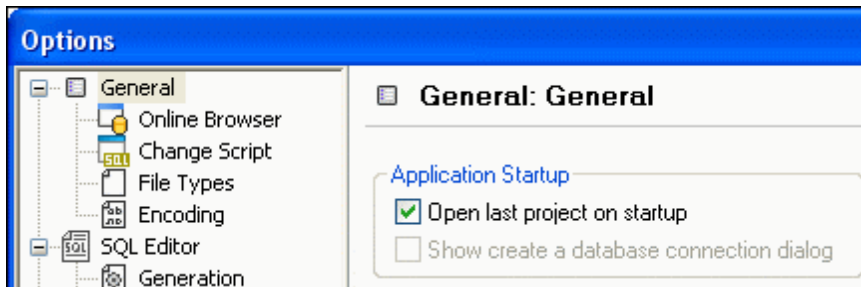
To activate the "Open last project on startup" setting:

1. Select the menu option **Tools | Options** and click **General**.
2. Activate the `Open last project on startup` check box, and click **OK** to confirm.

4.2.2 Adding Tables to the Database

In the previous section, you have created a database project and added a connection to a database to it. However, this database does not contain any tables yet. In DatabaseSpy, you can add tables with only a few mouse clicks.

Objective

In this section of the tutorial, you will learn how to use SQL scripts and the Design Editor to create the tables that make up the zoo database. Specifically, you will learn how to do the following:

- Open and execute an SQL file that is supplied in the tutorial folder
- Add a table using the Design Editor
- Generate and modify a CREATE statement
- Create relationships between the tables

Commands used in this section

Open: This command is located in the context menu that opens when you right-click an SQL file in the SQL folder of the Project window. Alternatively, you can also double-click the SQL file to open the SQL script in an SQL Editor window.

Save Project: This command is located in the toolbar of the Project window; alternatively, you can also choose the menu option **File | Save Project As...** or press **Ctrl+S**. Click this command to save an existing project under its project name or to specify a name and path for a new project, respectively.

Execute: This command is located in the toolbar of an SQL Editor window and in the **SQL Editor** menu. Click this command to execute the SQL in the SQL Editor window.

Refresh: This command is available in the Online Browser as a button, or as an option in the context menu that opens when you right-click a database. Use this command to refresh the database and display new or changed database objects in the Online Browser.

Show in new SQL Editor | **Create**: This command is located in the context menu that opens when you right-click a table in the Online Browser. Click this command to generate a CREATE statement based on the selected table in a new SQL Editor window.

Design Editor: This command is located in the Standard toolbar; alternatively, you can also choose the menu option **File | New | Design Editor** or press **Ctrl+D**. Click this command to open a new Design Editor window which is assigned to the active data source connection.

New Table: This command is located in the Design Editor toolbar; alternatively, you can also choose the menu option **Design Editor | Create New Table** or press **Ctrl+T**. Click this command to add a new table to an open Design Editor window.

Make Primary Key: This command is located in the context menu that appears when you right-click a table in a table design. Click this command to define the primary key for the table.

Make Foreign Key: This command is located in the context menu that appears when you right-click a table in a table design. Click this command to define a new foreign key for the table.

Execute Change Script: This command is located in the toolbar of the Database Structure Change Script window. Click this command to execute the recorded change script and to commit the changes to the database.

Opening and Executing an SQL File

In the previous section of the tutorial, you added the create_ZooDB_tables.sql script to your database project. This script can now be used to add tables to the zoo database. Note that any SQL script must be assigned to a data source connection before it can be executed in DatabaseSpy (see the instructions below).

To add tables to a database by executing an SQL script:

1. Make sure that the ZooDBConnect connection is active.
2. In the Project window, double-click the **create_ZooDB_tables.sql** entry in the SQL folder. The script file opens in an SQL Editor window
3. If necessary, assign a data source to the current script. When the script has no data source assigned, the "Unknown - Offline" status appears at the top of the SQL Editor window.
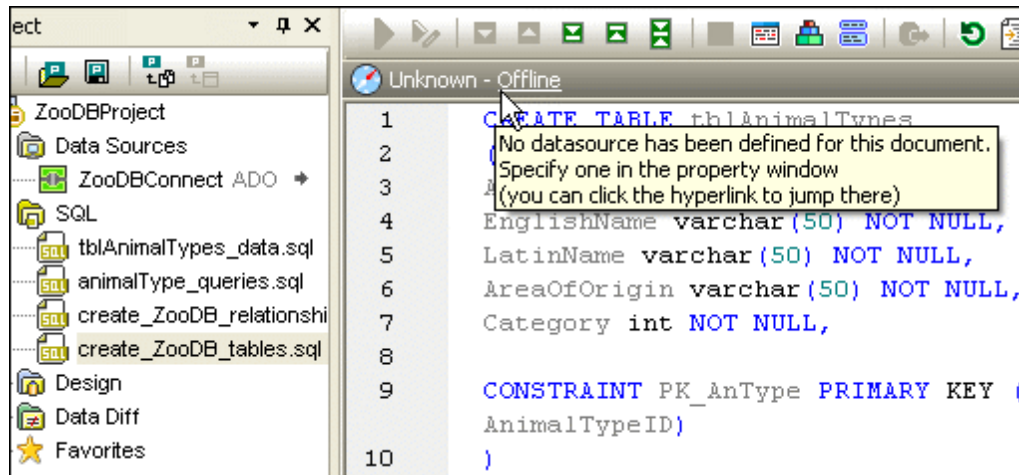
To assign a data source to the script, click the `Offline` hyperlink and, in the Properties window, select "ZooDBConnect" from the `Data Source` list.
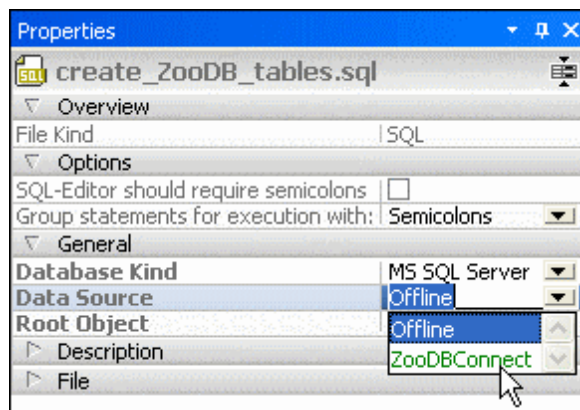
The SQL script is now connected to the ZooDBConnect data source, and the execution

buttons in the SQL Editor are active. Click the **Save Project** button in the Project window or press **Ctrl+S** to save this data source assignment in the project properties.

4.   In the SQL Editor, click the **Execute** button or press **F5**. The tables are added to the ZooDB database.

To see the newly added tables in the Online Browser:

1.   Click the **Online Browser** tab.
2.   Right-click on the name of the database, `ZooDB`, and select **Refresh** from the context menu or click the **Refresh** button in the toolbar of the Online Browser.
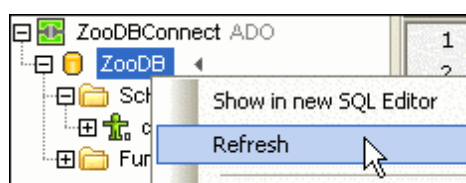
The newly inserted tables are displayed in the User Tables folder. If the script executed successfully, the following tables will have been added to the database:
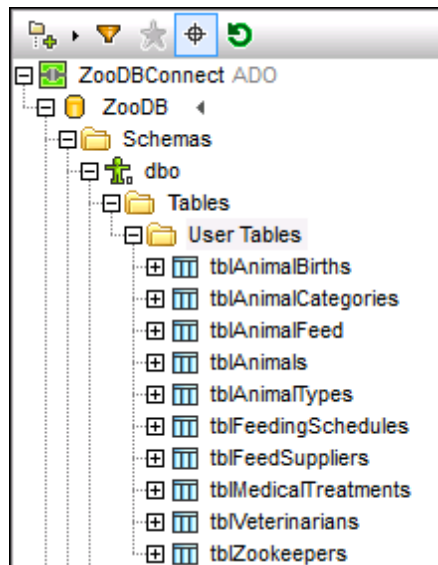
Adding Tables Using Design Editor

The Design Editor can also be used to add tables to databases using a graphical user interface. Use of the Design Editor requires no knowledge of SQL syntax. To illustrate this, we will now add to the database a new table that stores zoo visitor statistics. We will call this table tblVisitorStatistics. The table will consist of the following columns:

- **StatisticID**: int; not null; primary key
- **StatisticDate**: datetime; not null;
- **VisitorCount**: int; not null

To add the table to the database using the Design Editor:

1. Click the **Design Editor** button in the Standard toolbar. A new Design window, called Design1, is opened with the connection name and database name visible in the Status bar.
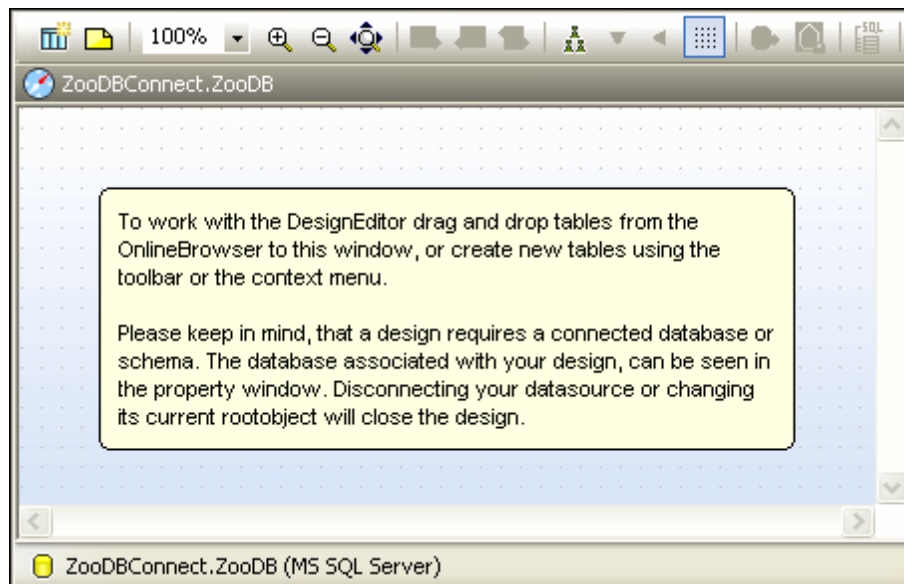
2. Click the **New Table** ⊞ button. A new table appears in the Design Editor window. A change script message appears at this point, and the SQL statements appear in the Database Structure Change Script window. After you complete the table definition, you will need to execute the change script to actually create the table in the database.
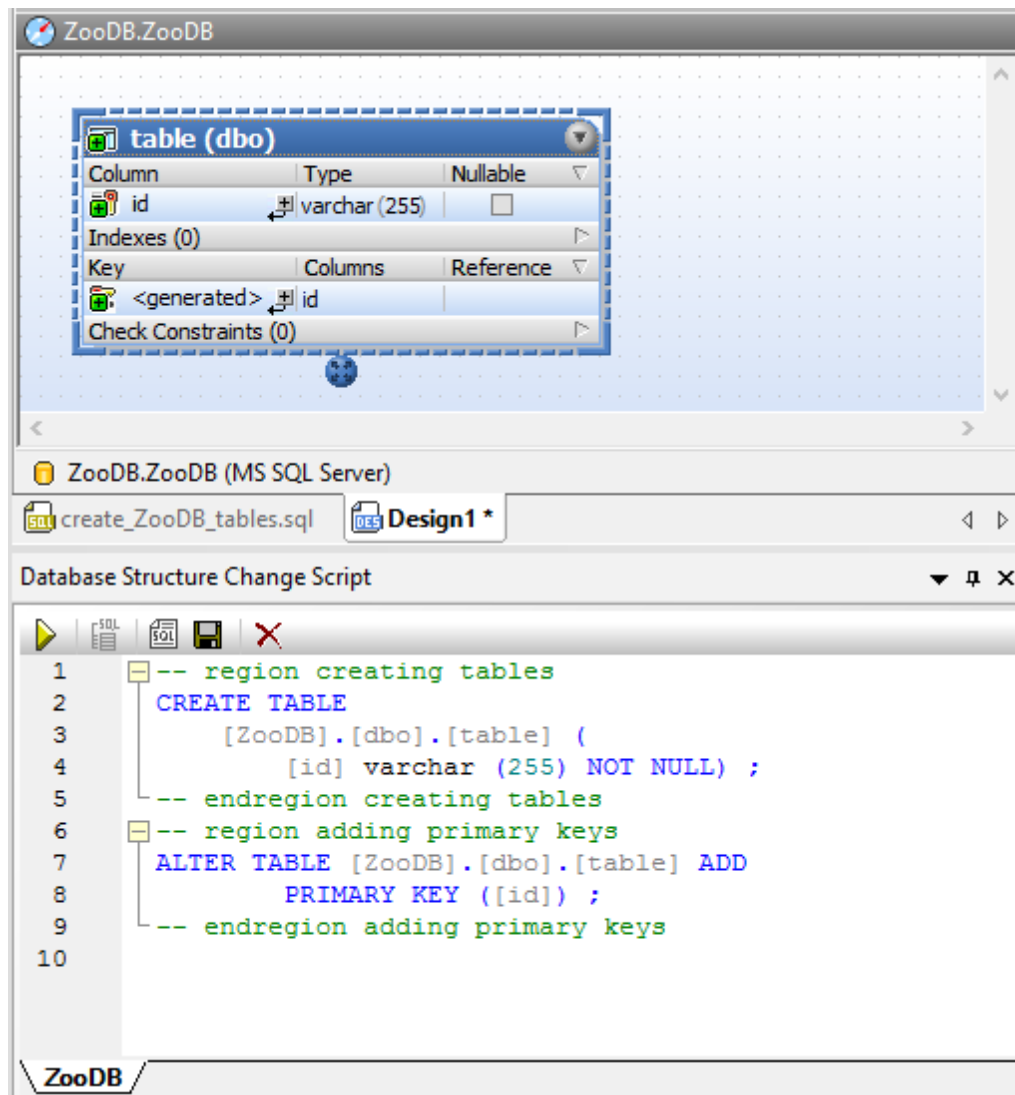
3.   Double-click the table name, "table (dbo)", and enter "tblVisitorStatistics". Press **Enter**.
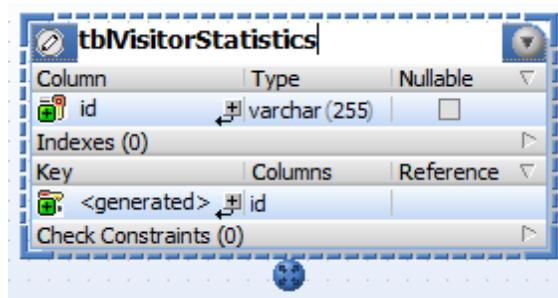
4.   In the table design, double-click the text under "Column". Change the column name to "StatisticID" and press **Enter**.
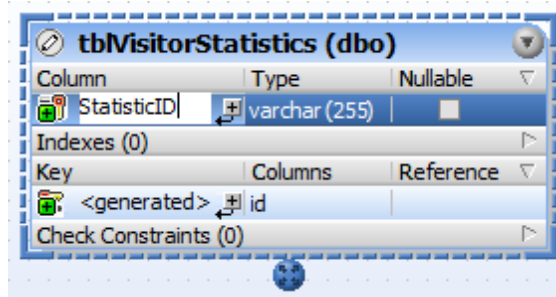
5. Likewise, double-click the text under "Type", replace "varchar(255)" with "int", and press **Enter**. At this stage, the table has one column, and we want to add two more.
6. Click the plus icon next to column StatisticID. A column is added and selected for editing. Type "StatisticDate", press **Tab**, enter type as "date", and press **Enter**.
7. Right-click into the table design and choose **Insert new | Column** from the context menu. Change the column name to "VisitorCount"and press **Enter**. Using the same technique as for previous columns, change the data type of the "VisitorCount" column to "int".
8. Right-click the title bar of the Columns section and activate the **Identity** option in the context menu. The select the **Identity** check box for the StatisticID column.

At this stage, the table design should look as follows:
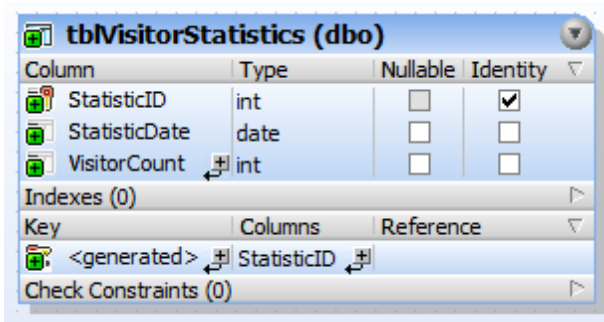
Notice that a primary key having the name <generated> has been generated automatically for the first column in the table. The indicates that the new column will have a primary key with a name automatically assigned by the database, when you run the database update script. Optionally, if you want to change the name of the primary key, replace the text <generated> with a descriptive primary key name. In this case, however, you will need to ensure that the name is not already used in the database.

You are now ready to execute the database script which will actually add the table to the database. In the Database Structure Change Script window, click the **Execute Change Script** button to commit the new table to the database.

Generating and Editing a CREATE Statement

The Online Browser allows you to generate SQL statements based on an existing table or column. In this step, we will generate a CREATE statement from the table `tblZookeepers`, edit the statement, and use it to create a new table `tblEmployees`.

To add a table by editing a CREATE statement generated from an existing table:

1. In the Online Browser, right-click the `tblZookeepers` table.
2. Select the menu option **Show in new SQL Editor | Create** from the context menu. An SQL Editor window appears, displaying the following SQL statement:

3. Edit the statement as shown below to create `tblEmployees`. The parts of the statement that need to be changed are marked using *underlined, italic text*:

```
CREATE TABLE
    [ZooDB].[dbo].[tblEmployees] (
        [EmployeeID] int IDENTITY (1, 1) NOT NULL,
        [FirstName] varchar (50) COLLATE SQL_Latin1_General_CP1_CI_AS
NOT NULL,
        [LastName] varchar (50) COLLATE SQL_Latin1_General_CP1_CI_AS
NOT NULL,
        [Address] varchar (50) COLLATE SQL_Latin1_General_CP1_CI_AS NOT
NULL,
        [City] varchar (50) COLLATE SQL_Latin1_General_CP1_CI_AS NOT
NULL,
        [State] varchar (50) COLLATE SQL_Latin1_General_CP1_CI_AS NOT
NULL,
        [Telephone] varchar (50) COLLATE SQL_Latin1_General_CP1_CI_AS
NOT NULL,
        [DOB] datetime NOT NULL,
        CONSTRAINT [PK_Employee] PRIMARY KEY ([EmployeeID]) ) ;
```

4. Click the **Execute** button or press **F5**. In the Online Browser, right-click the ZooDB database and select **Refresh** from the context menu, or click the **Refresh** button. The newly inserted table, tblEmployees, is displayed along with the tables that you created earlier.

Creating Relationships Between the Tables

Now that you have added the tables to the zoo database, you will add the foreign key relationships to the database using both the Design Editor and an SQL script. A foreign key is a relation between two tables that ensures that data can only be inserted into certain columns in one table if it exists in the second table.

In our example, a new animal type in table tblAnimalTypes can only be added if the Category is existing as CategoryID in table tblAnimalCategories. This way, you can only add animals that belong to an animal category that is already defined in tblAnimalCategories. You will use the Design Editor to add this relation to the database. Furthermore, this tutorial provides an SQL script that adds the remaining foreign key relations to the database.

Defining a foreign key relationship using the Design Editor

1. Make sure that the ZooDBConnect connection is active and click the **Design Editor** button in the Standard toolbar or press **Ctrl+D**.
2. In the Online Browser, click the table tblAnimalTypes and drag it into the Design Editor window.
3. Right-click the table design and select **Insert new | Key | Foreign Key** from the context menu. A foreign key is created in the design, having the name <generated>, and the entry is selected for editing.
4. Optionally, change the key name from <generated> to "FK_CategoryID" and press **Enter**. If you leave the name as <generated>, the key name will be automatically generated by the database.
5. In the Reference column, double-click the "[select the referenced table]" entry, and select "tblAnimalCategories" from the drop-down list that appears. The first column "CategoryID" of the referenced table appears below the table name and there is no need to change this

entry.

6. In the Columns column, double-click the "AnimalTypeID" entry that has been inserted by default, and select "Category" from the drop-down list that appears.
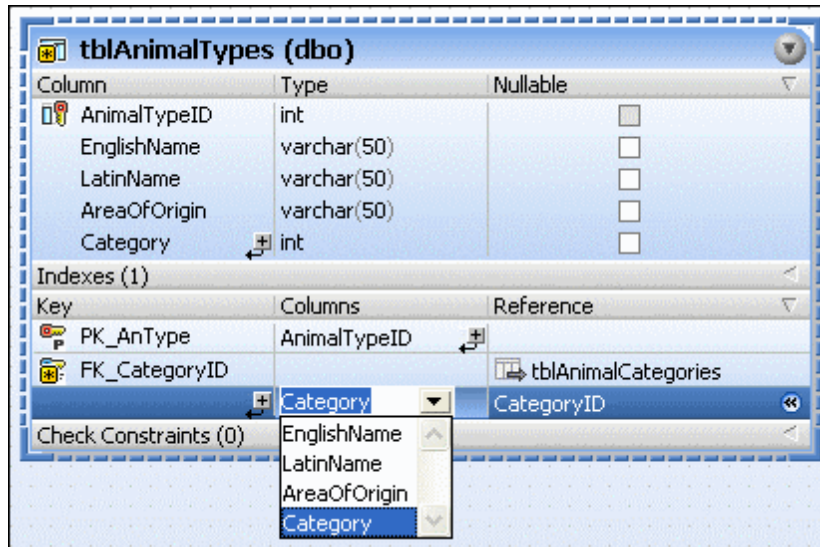
7. In the Database Structure Change Script window, click the **Execute Change Script** button to execute the change script and to commit the new foreign key to the database.
8. Optionally, right click the "FK_CategoryID" entry in the table design and select **Add related tables | Referenced Tables** from the context menu.

Adding foreign key relationships via SQL scripts

To speed up the creation of the remaining foreign key relationships for the zoo database, an SQL script is provided in the tutorial folder. This script has already been added to the project earlier in this tutorial.

1. Make sure that the ZooDBConnect connection is active.
2. In the Project window, double-click the file create_ZooDB_relationships.sql. The file opens in an SQL Editor window.
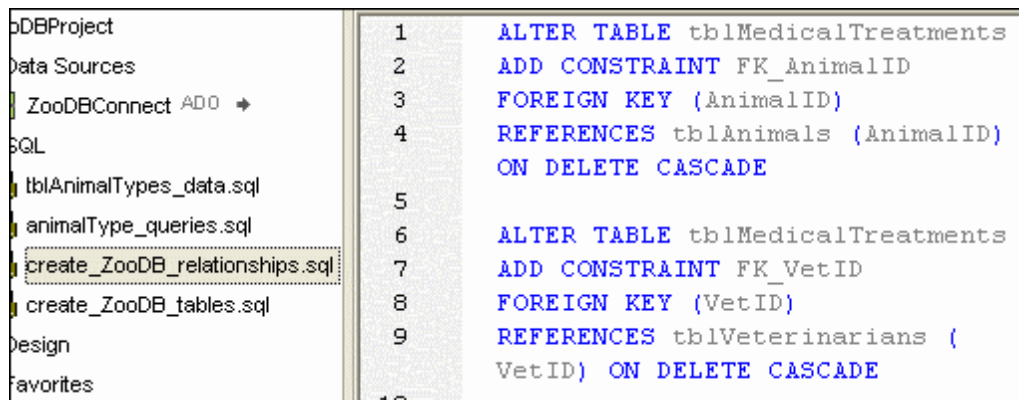
3. In the General section of the Properties window, choose "ZooDBConnect" from the Data Source drop-down list. The SQL script is now connected to the ZooDBConnect data

source, and the execution buttons in the SQL Editor become active.

4. Click the **Save Project** 🖻 button in the Project window or press **Ctrl+S** to save this data source assignment in the project properties.

5. Click the **Execute** ▷ button or press **F5**. Foreign key relationships are created between the tables in the zoo database.

To get a graphical view of the relationships that were just created, right click the "User Tables" folder in the Online browser, and select **Show in new Design Editor**.

4.2.3 Defining Constraints

In the previous section, you have added tables and created relations between them. You may want to control which data is added to your database tables, or simplify data entry by defining default values for certain columns.

Objective

In this section of the tutorial, you will learn how to use the Design Editor to create columns that can contain only unique or limited data, or have a predefined standard value. Specifically, you will learn how to do the following:

- Define a unique constraint for a column
- Add check constraints on column and table level
- Provide columns with default values

Commands used in this section

Show in new Design Editor: This command is located in the context menu that opens when you right-click a database object in the Online Browser. Click this command to open a new Design Editor window and display the selected database object in it.

Make Unique Key: This command is located in the context menu that opens when you right-click a column in a table design. Click this command to define a unique key on the selected column.

Execute Change Script: This command is located in the toolbar of the Database Structure Change Script window. Click this command to execute the recorded change script and to commit the changes to the database.

Insert new | Check Constraint: This command is located in the context menu that opens when you right-click somewhere into a table design. Click this command to define a new check constraint for the table.

Create Default Constraint: This command is located in the context menu that opens when you right-click a column in a table design. Click this command to define a default constraint for the selected column.

Edit Data: This command is located in the **SQL and Data** sub-menu of the context menu that opens when right-clicking a table design. Choose this command to generate an SQL SELECT statement that is automatically executed in the SQL Editor and switches the Result window into the Editing mode. This command works for all supported database kinds for both ADO and ODBC connections.

Options...: This command is located in the **Tools** menu. Click this command to display the **Options** dialog box, where you can change the DatabaseSpy settings.

| + | **Append a new row**: This command is available as a button in the Result window toolbar. Click this button to add a new row to the result grid. Alternatively, you can also right-click anywhere in the result grid and choose **Append a new row** from the context menu. The **Append a new row** command is only available if the Result window is in the Editing mode. |

Defining a Unique Key

By defining a unique key, you can prevent the user from entering duplicate data into a column. In our zoo database, you will define a unique key on the `LatinName` column of table `tblAnimalTypes`. This way, each name can be added to the table only once.

To define a unique key on a column:

1. In the Online Browser, right-click the table `tblAnimalTypes` and select **Design Editor | Show in new Design Editor** from the context menu.
2. In the table design, right-click the column `LatinName` and select **Make Unique key** from the context menu.
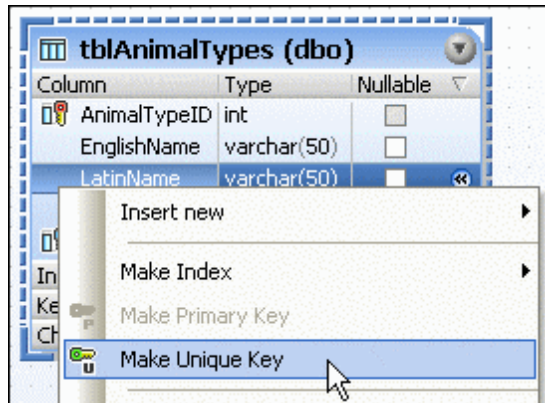
A unique key is created in the design, having the name `<generated>`, and the entry is selected for editing.
3. Optionally, change the key name from `<generated>` to "UK_LatinName" and press **Enter**. If you leave the name as `<generated>`, the key name will be automatically generated by the database.
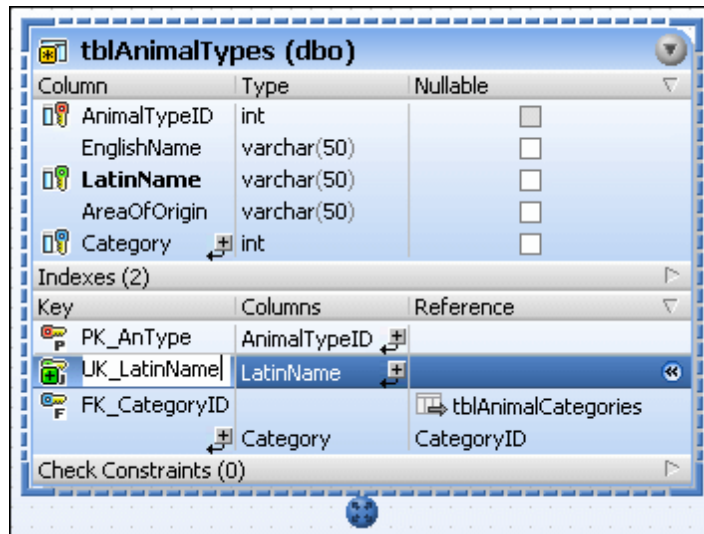
4. In the Database Structure Change Script window, click the **Execute Change Script** button to execute the change script and to commit the new unique key to the database.

Defining a Check Constraint

When a check constraint is defined, data will be validated according to the expression specified in the check constraint, before it is added to a table. Data that fails the validation will not be added.

In this section of the tutorial, you will create a check constraint that makes sure that different animals are entered into the Mother and Father columns of tblAnimalBirths. Since this constraint spans more than one column, it will be displayed in the Check Constraints section of the table design, and in the Constraints sub-folder of the Online Browser, respectively.
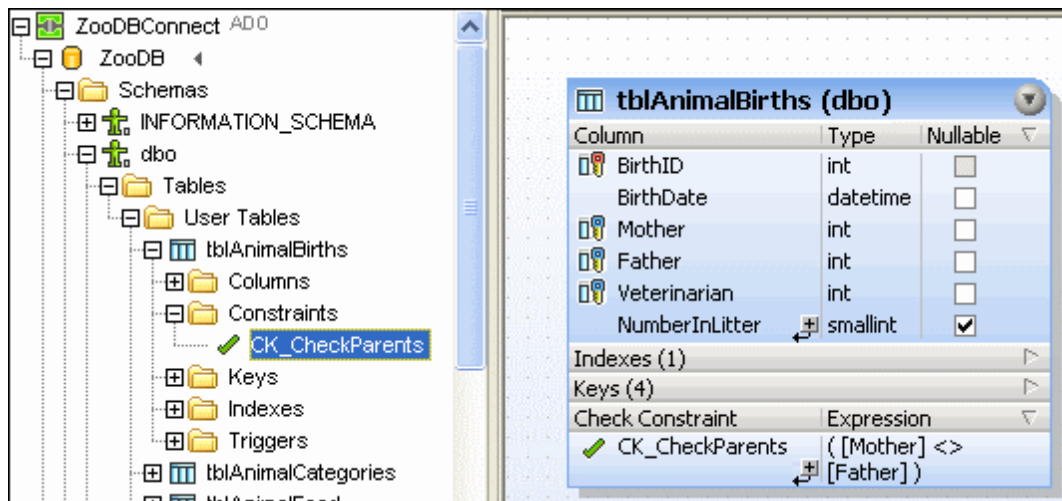
Defining the check constraint

1. In the Online Browser, right-click table tblAnimalBirths and select **Design Editor | Show in new Design Editor** from the context menu.

2. Right-click the table design and select **Insert new | Check Constraint** from the context menu. A check constraint is created in the design, having the name `<generated>`, and the entry is selected for editing.

3. Optionally, change the constraint name from `<generated>` to "CK_CheckParents" and press **Enter**. If you leave the name as `<generated>`, the key name will be automatically generated by the database.

4. In the Expression column, double-click the "[expression required]" entry, enter "Mother <> Father", and press **Enter**.

5. In the Database Structure Change Script window, click the **Execute Change Script** button to execute the change script and to commit the new check constraint to the database.

Testing the check constraint
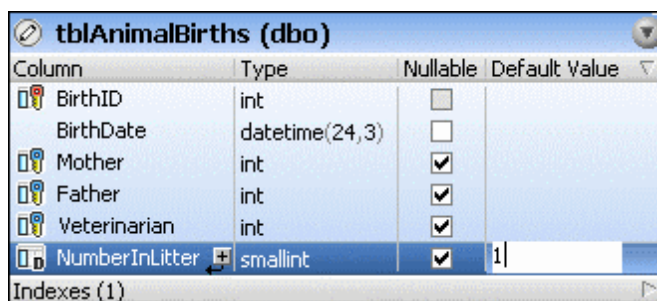
After the check constraint has been committed to the database, you cannot enter the same animal ID into the Mother and Father columns anymore. If both columns contain the same value in a row, DatabaseSpy displays an error in the Message tab.
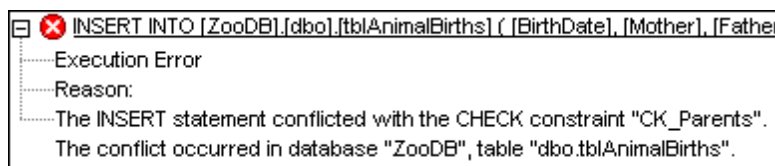
Defining a Default Constraint

For some columns it may be useful to define a standard value. By defining a default constraint and setting the Result view options accordingly, this default is inserted automatically when a new row is added to the table in the Result view.
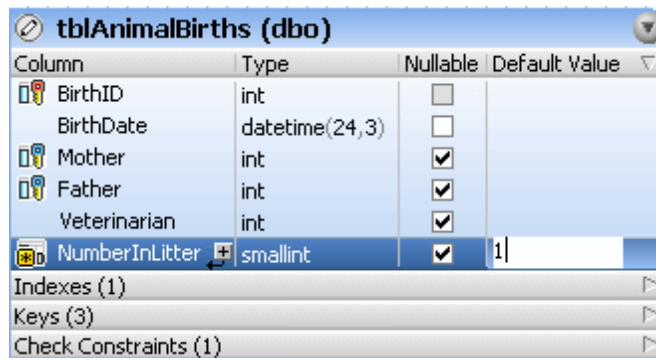
In this section of the tutorial, you will define a default constraint on the tblAnimalBirths table which sets the number in litter automatically to "1" for new rows that are added in the Result view.

To set a default number in litter in tblAnimalBirths:

1. In the table design, right-click column NumberInLitter and select **Create Default Constraint** from the context menu. The Default Value column appears in the table design. (If the Default Value column is not visible, click into the table design and it will resize properly.)

2. Double-click the field in the NumberInLitter row, type "1", and press **Enter**.

3. In the Database Structure Change Script window, click the **Execute Change Script** button to execute the change script and to commit the new default constraint to the database.

Testing the default constraint

When testing the default constraint you have just defined, make sure that the `Initialize cells with default values` option is enabled in the Result View options. Please note that this option is disabled by default.

To create a new row containing a default value:

1. Select the menu option **Tools | Options (Ctrl+Alt+O)** and select the Result View page of the **Options** dialog box.
2. Activate the `Initialize cells with default values` check box and click **OK**.
3. Right-click the table heading in the Design Editor and select **SQL and Data | Edit Data** from the context menu. A message box appears, click **OK**.
4. In the Result view, click the **Append a new row** button. The NumberInLitter column shows "1" in the new row.

	BirthID	BirthDate	Mother	Father	Veterinarian	NumberInLitter
1			*[NULL]*	*[NULL]*	*[NULL]*	1

4.2.4   Inserting Data into the Database

To insert data into the database, you can enter individual INSERT statements into an SQL Editor window or save them in an SQL file and execute all the statements together. More conveniently, DatabaseSpy also allows you to import data from TXT files and add new records in the Result window after executing a SELECT statement for editing.

Objective

In this section of the tutorial, you will learn how to insert data into the tables you have created in the previous section. Specifically, you will learn how to do the following:

*   Execute an SQL script that populates the zoo database with data
*   Import data provided in a TXT file
*   Import data provided in an XML file

- Create a new SQL script and save it in the project
- Add new records in the Result window of the SQL Editor

Commands used in this section

Execute SQL: This command is located in the context menu that opens when you right-click an SQL file in the SQL folder of the Properties window. Use this command to immediately execute the SQL statements contained in the file.

Row Count | Show/Update: This command is available in the context menu that opens when you right-click a table, view or its parent folder in the Online Browser. Click this command to display the number of rows for the selected table or view, or the number of rows for all tables or views, respectively, that are contained in the selected folder.

Import: This command is located in the Tools toolbar; alternatively, you can also choose the menu option **Tools | Import data to the database...** or press **Ctrl+I**. Click this command to open the **Import data to the database** dialog box.

Execute: This command is located in the toolbar of an SQL Editor window and in the **SQL Editor** menu. Click this command to execute the SQL in the SQL Editor window.

Retrieve data | All rows: This command is located in the context menu that opens when you right-click a table or column in the Online Browser; alternatively, you can also select the table and press **Ctrl+Alt+R**. Click this command to have all rows of data displayed in a Result tab.

SQL Editor: This command is located in the Standard toolbar; alternatively, you can also choose the menu option **File | New | SQL Editor** or press **Ctrl+N**. Click this command to open a new SQL Editor window that is assigned to the active data source connection.

Save: This command is located in the Standard toolbar; alternatively, you can also choose the menu option **File | Save** or press **Ctrl+S**. Click this command to save the active file.

Add the Active File: This command is located in the toolbar of the Project window. Clicking this command adds the active file to the project under its corresponding folder.

Save Project: This command is located in the toolbar of the Project window; alternatively, you can also choose the menu option **File | Save Project As...** or press **Ctrl+S**. Click this command to save an existing project under its project name or to specify a name and path for a new project, respectively.

Edit Data: This command is located in the context menu that opens when you right-click a table or column in the Online Browser; alternatively, you can also select the table and press **Ctrl+Alt+E**. Choose this command to generate an SQL SELECT statement that is automatically executed in the SQL Editor and switches the Result window into the Editing mode. This command works for all supported database kinds for both ADO and ODBC connections.

Append a new row: This command is available as a button in the Result window toolbar. Click this button to add a new row to the result grid. Alternatively, you can also right-click anywhere in the result grid and choose **Append a new row** from the context menu. The **Append a new row** command is only available if the Result window is in the Editing mode.

Commit : This command is available as a button in the Result window toolbar. Click this button to commit the changes you have made in the result grid to the database. The **Commit** button is only visible if the Result window is in the Editing mode.

Using a Script to Add Data to the Database

Earlier in this tutorial, you added a script to the ZooDBProject that you can now use to populate the `tblAnimalTypes` and `tblAnimalCategories` tables with data. The file tblAnimalTypes_data.sql contains all the INSERT statements that are necessary to add the data to the `tblAnimalTypes` and `tblAnimalCategories` tables.

```
1      USE zooDB
2
3      INSERT INTO tblAnimalCategories (Category)
       VALUES ('marsupial')
4
5      DECLARE @CategoryID As int
6
7      SET @CategoryID = (SELECT CategoryID FROM
       tblAnimalCategories WHERE Category =
       'marsupial')
8
9      INSERT INTO tblAnimalTypes (EnglishName,
       LatinName, AreaOfOrigin, Category)
10     VALUES ('Parma wallaby', 'Macropus parma',
       'Australia', @CategoryID)
11
12     INSERT INTO tblAnimalCategories (Category)
13     VALUES ('bonytongue')
```

To avoid the manual data source assignment to the individual SQL files in your project in the future, you will also define a default data source (i.e., your tutorial database) for all files that are contained in the SQL folder of the Project window.

To set a default data source for the SQL folder:

1. Click the SQL folder in the Project window.
2. In the Properties window below, select "ZooDBConnect" from the `Data Source` drop-down list in the General section.
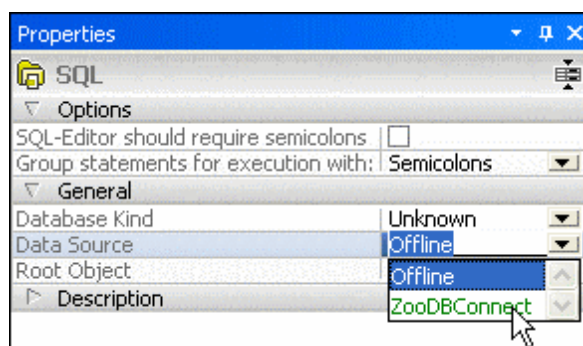
3.   Click the **tblAnimalTypes_data.sql** file in the SQL folder and note that "ZooDBConnect" now appears as data source in the properties.
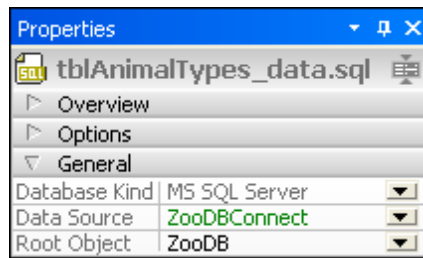
To execute tblAnimalTypes_data.sql:

1.   Make sure that the ZooDBConnect connection is active.
2.   In the Project window, right-click the file **tblAnimalTypes_data.sql** in the SQL folder and choose **Execute SQL** from the context menu. The file opens in an SQL Editor window and executes automatically. The tables `tblAnimalTypes` and `tblAnimalCategories` are populated with data.

Checking the result

To do a quick check as to whether the tables have been successfully populated with data, you can display the row count for all tables in the Online Browser.
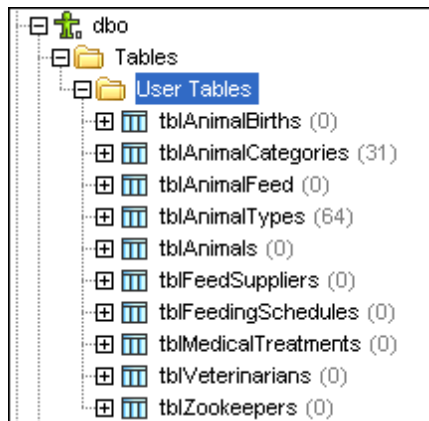
To display the row count of all tables in the Online Browser:

•   Right-click the User Tables folder and select **Row Count | Show/Update** from the context menu.

Importing Data from TXT Files

DatabaseSpy provides an import function which allows you to insert data provided in text files into a database. Example text files for the zoo database are included in the tutorial folder of your DatabaseSpy installation. How to import data from XML files will be explained in the next step of this tutorial.

In this step, you will import data from the **tblZookeepers.txt** file into the database table

`tblZookepers`.

To import data from text files into the database:

1. Click the **Import** ![import icon] button in the Tools toolbar, or select the menu option **Tools |** **Import data to the database...**, or press **Ctrl+I**.
2. Click the **CSV** button on the left side of the dialog box.
3. In the Selection tab, click ![button] next to the **Path**, and select the **tblZookeepers.txt** file from the \Altova\DatabaseSpy2017\DatabaseSpyExamples\Tutorial sub-folder of the [(My) Documents](#) folder.
4. Under **Database Generation Options**, select **Update existing tables**.
5. Under **Import Mode**, select **Generate SQL in SQL Editor**.
6. In the Preview group box, click the **Apply Options** button. This displays table data in the Preview window. To view the table definition, choose the `Show Definition` radio button.
7. Optionally, click inside the `Target Table Name` column to change the target table name.
8. Click the **ZookID** table icon ![icon] until it changes to the symbol for a column not to be imported ![icon]. The column header allows you to define the fields key type, and also if it should be included during the import process. We will not import the ZookID column since this is an identity column and the primary key will be generated automatically during import.

   At this stage, if you followed all the steps above, the import options must be as shown below:
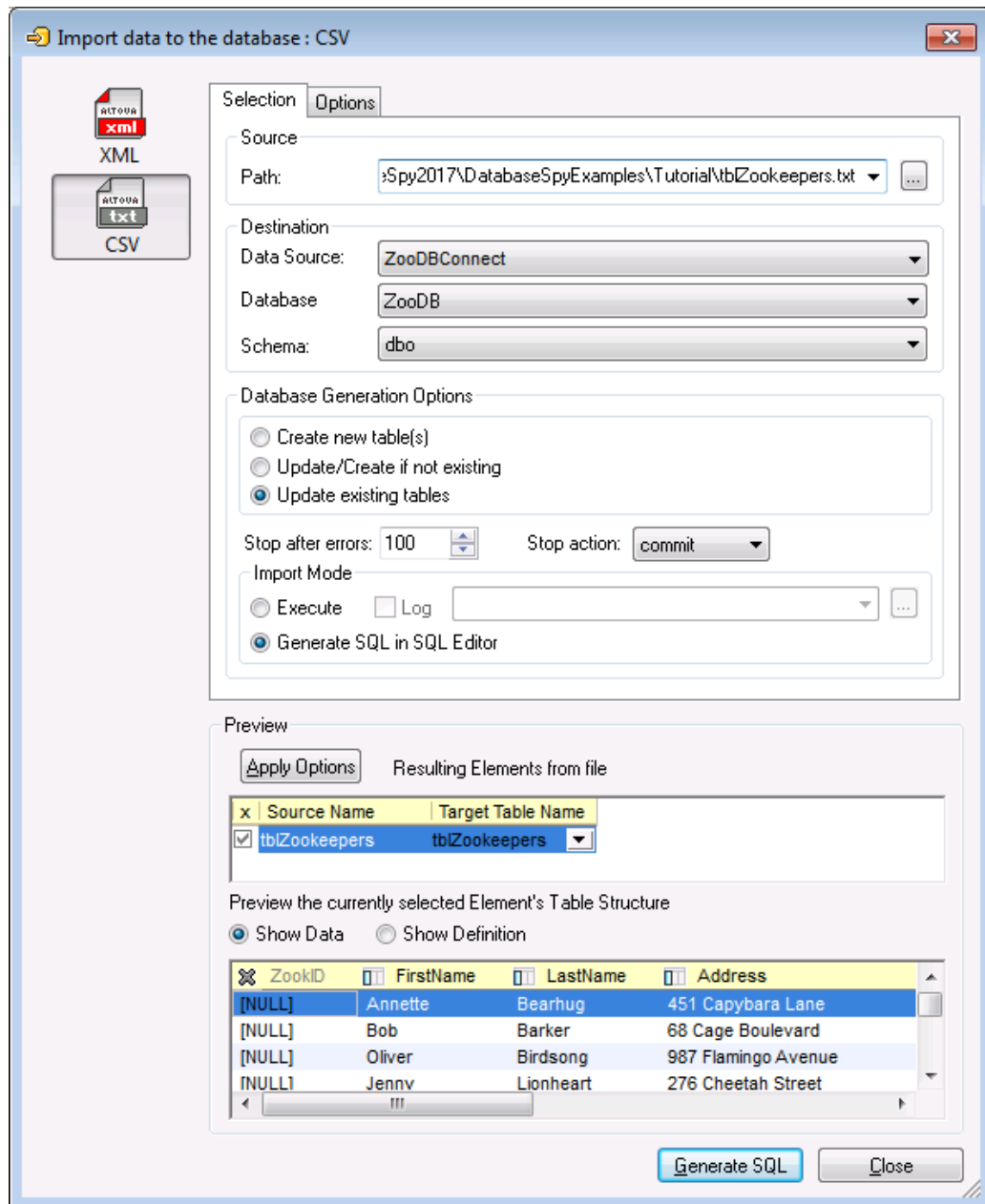
9. Click the **Generate SQL** button. This creates an SQL script which can be executed to complete the import process. Click **Generate SQL** in the message box to view the script in the SQL Editor.

10. Click the **Execute** button or press **F5**.

You can now view the imported data as follows:

- In the Online Browser, right-click the tblZookeepers table and select **Retrieve data | All rows** from the context menu (alternatively, press **Ctrl+Alt+R**).

Importing Data from XML Files

In the previous lesson, you learned how to insert data into a database using text files. In DatabaseSpy, you can also use XML files to export data from, or import data into, a database. The tutorial folder of your DatabaseSpy installation includes also example XML files for this purpose.

In this step, you will import data from **dbo.tblFeedSuppliers.xml** into the **tblFeedSupplier** database table. This assumes you have not already populated this table by other means (for example, by importing data from **tblFeedSuppliers.txt** file into it, see Importing Data from TXT Files).

To import data from an XML file into a database table:

1. Click the **Import** button in the Tools toolbar, or select the menu option **Tools | Import data to the database...**, or press **Ctrl+I**. The **Import data to the database** dialog box opens.
2. Click the **XML** button on the left side of the dialog box.
3. In the Selection tab, click next to the `Path` field. Select **dbo.tblFeedSuppliers.xml** from the \Altova\DatabaseSpy2017\DatabaseSpyExamples\Tutorial sub-folder of the (My) Documents folder.
4. Under **Database Generation Options**, select the **Update existing tables**.
5. Under **Import Mode**, select **Generate SQL in SQL Editor**.
6. Click the **Options** tab.
7. Under **Start point of import**, in the **Starting from list**, select "Row". (This specifies the name of the XML element from which data will be imported).
8. Under **Automatic fields**, deactivate the **Create primary/foreign keys** check box.
9. Under **Other options**, select **Exclude namespace name**.
10. Under **Preview**, click the **Apply Options** button. This displays table data in the Preview window. Note that the Import element now is not an available choice anymore. To view the table definition, choose the `Show Definition` radio button.
11. Click inside the `Target Table Name` column and choose **dbo.tblFeedSuppliers.xml** from the drop-down list.
12. Click the TreatmentID table icon until it changes to the symbol for a column not to be imported . The column header allows you to define whether it should be included during the import process. We will not import the TreatmentID column since this is an identity column and the primary key will be generated automatically during import.

At this stage, if you followed all the steps above, the import options must be as shown below (the two images below correspond to the **Selection** tab, and **Options** tab, respectively).
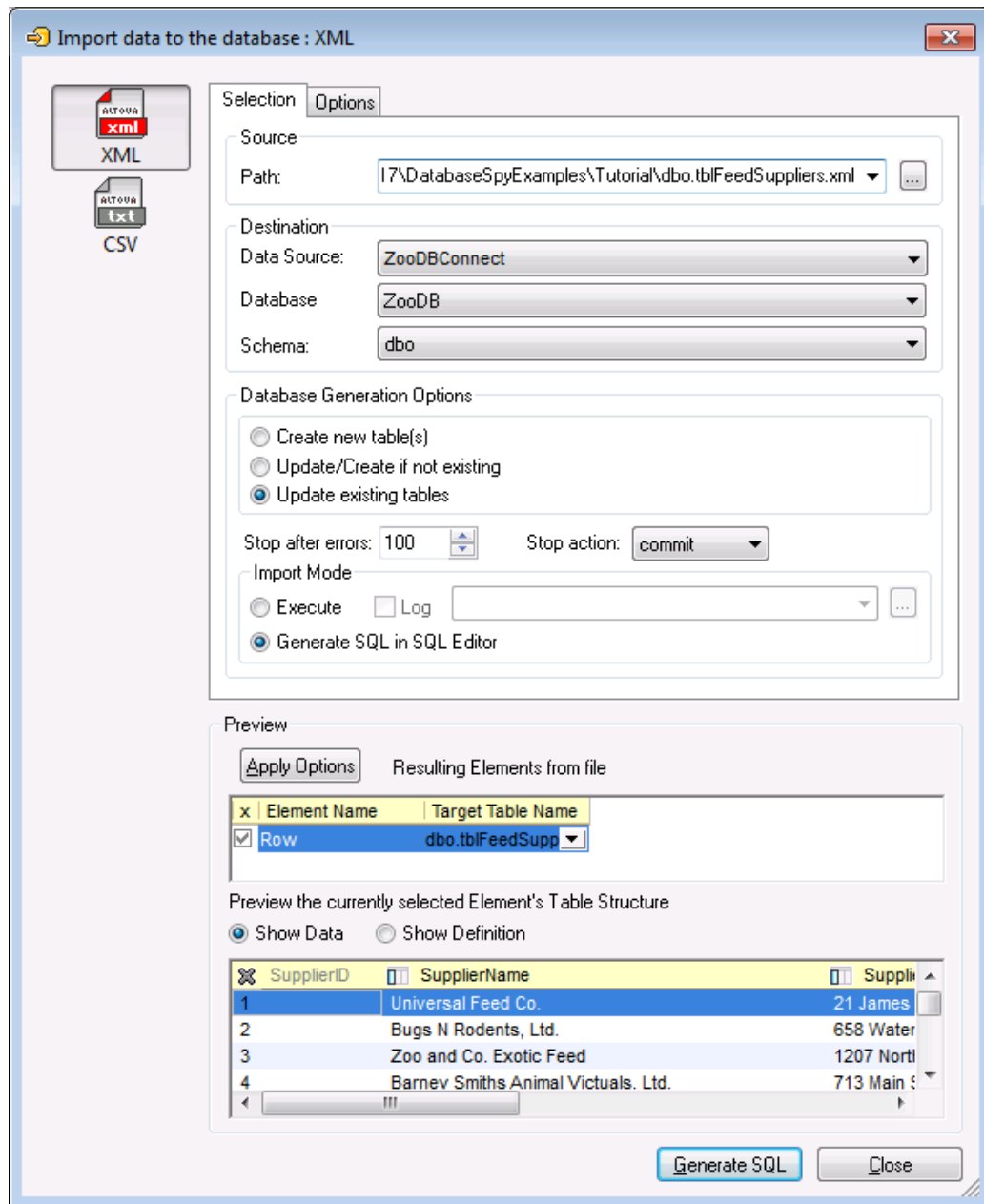
XML Import Options on the Selection tab
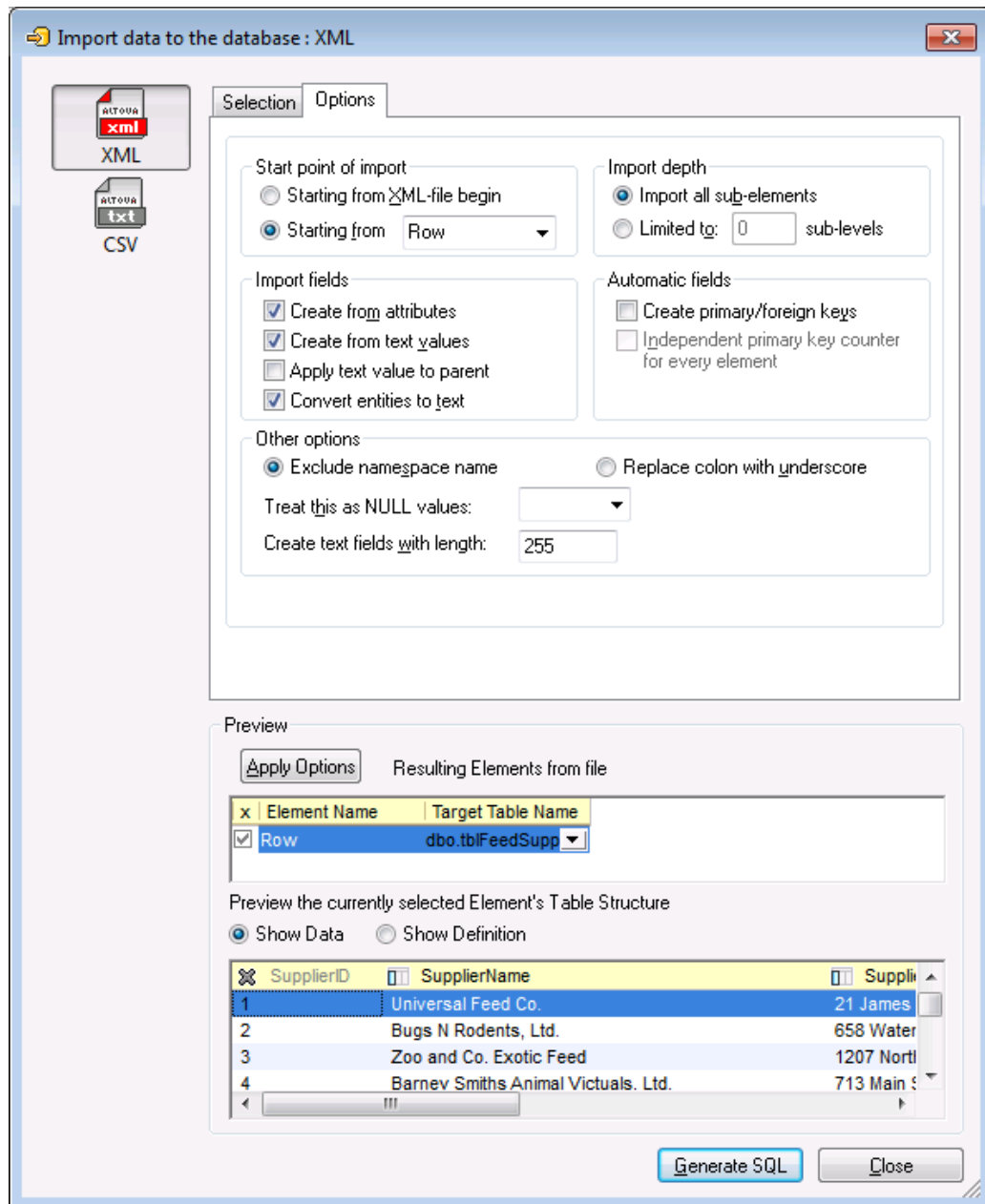
XML Import Options on the Options tab

13. Click the **Generate SQL** button. This creates an SQL script which can be executed to complete the import process. Click **OK** in the message box to view the script in the SQL Editor.

14. Click the **Execute** button or press **F5**.

15. In the Online Browser, right-click the `ZooDB` database and select **Refresh** from the context menu.

16. Right click the `tblMedicalTreatments` table and select **Retrieve data | All rows** from the context menu or press **Ctrl+Alt+R**.

Populating the Zoo Database with Data (Optional)

In the previous steps of this tutorial, you have imported data into the database from a text file (see Importing Data from TXT Files) as well as from an XML file (see Importing Data from XML Files).

Optionally, you can also import data from several text or XML files available in the \Altova \DatabaseSpy2017\DatabaseSpyExamples\Tutorial sub-folder of the (My) Documents folder. In this case, make sure to import files in the order shown below, to avoid problems with foreign key relationships.

Text files	XML files
1. tblZookeepers.txt 2. tblFeedSuppliers.txt 3. tblAnimalFeed.txt 4. tblFeedingSchedules.txt 5. tblAnimals.txt 6. tblVeterinarians.txt 7. tblAnimalBirths.txt 8. tblMedicalTreatments.txt	1. dbo.tblZookeepers.xml 2. dbo.tblFeedSuppliers.xml 3. dbo.tblAnimalFeed.xml 4. dbo.tblFeedingSchedules.xml 5. dbo.tblAnimals.xml 6. dbo.tblVeterinarians.xml 7. dbo.tblAnimalBirths.xml 8. dbo.tblMedicalTreatments.xml

Note that, for the import of all these files to succeed, the foreign key values defined in the source file (text or XML) must exist in the target database tables. If any of the import operations fails due to any reason (for example, due to data already present in the database tables), it is likely that other subsequent imports will fail. In this case, it is recommended to perform the import against a fresh copy of the database.

To set up a fresh copy of the ZooDB database:

1. Create a new database (see Creating the Tutorial Database).
2. Using the SQL Editor, run the following two scripts against it:
   o create_ZooDB_tables.sql
   o create_ZooDB_relationships.sql

Creating and Saving an INSERT Script

The context menu that opens when you right-click a table and drag it into an SQL Editor window provides the possibility to generate an INSERT statement based on an existing table.

Assuming that the zoo in our tutorial has just hired two new zookeepers, you will create a script that adds these two new employees to the table `tblZookeepers`, and save that script in your project.

To create a script that inserts two new zookeepers into tblZookeepers:

1. Click the **SQL Editor** icon in the Standard toolbar or select the menu option **File | New | SQL Editor (Ctrl+N)** to open a new SQL Editor window.

2. Using the **right** mouse button, drag and drop the `tblZookeepers` table from the Online Browser into the SQL Editor window.
   A context-sensitive menu appears.

3. Select **Insert** from the context menu.
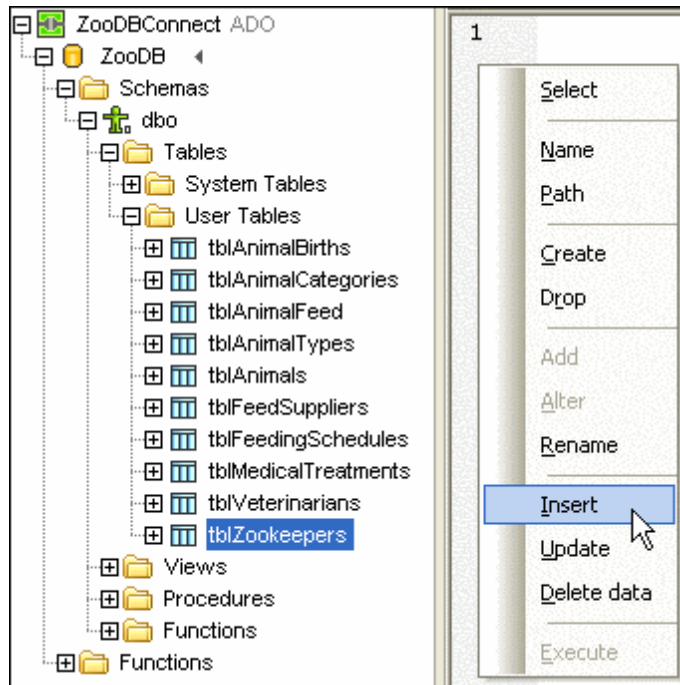
The following statement appears in the SQL window:

```
INSERT INTO [ZooDB].[dbo].[tblZookeepers] ( [ZookID],
[FirstName], [LastName], [Address], [City], [State],
[Telephone], [DOB] ) VALUES ( , '', '', '', '', '', '', '' );
```

4. Repeat steps 2 and 3 to create a second INSERT statement.

5. Alter the **first INSERT statement** as follows:
```
INSERT INTO [ZooDB].[dbo].[tblZookeepers] ([FirstName],
[LastName], [Address], [City], [State], [Telephone], [DOB] )
VALUES ('June', 'Bug', '1345 Gorilla Highway', 'Newark', 'NJ',
'(515)555-7217', '1963-11-26');
```
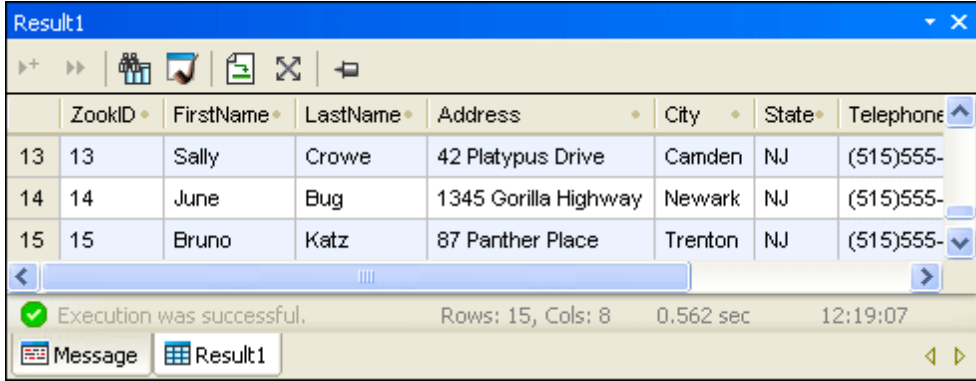
Please note: No value is being inserted for the column ZookID, because it is an identity

column and its value will be automatically generated. You must therefore delete the
`[ZookID]` entry and the first comma from the statement.

6.   Alter the **second INSERT statement** to add the following values:
     `INSERT INTO [ZooDB].[dbo].[tblZookeepers] ([FirstName],`
     `[LastName], [Address], [City], [State], [Telephone], [DOB] )`
     `VALUES ('Bruno', 'Katz', '87 Panther Place', 'Trenton', 'NJ',`
     `'(515)555-7667', '1969-09-13');`

7.   Click the **Execute** button or press **F5**.

8.   To check the results, open a new SQL Editor window and enter the statement `SELECT *`
     `FROM tblZookeepers`. Execute the statement. Optionally, click the **Maximize**
     button to extend the size of the Result window. Notice that two new rows have been
     inserted into the table.

	ZookID	FirstName	LastName	Address	City	State	Telephone
13	13	Sally	Crowe	42 Platypus Drive	Camden	NJ	(515)555-
14	14	June	Bug	1345 Gorilla Highway	Newark	NJ	(515)555-
15	15	Bruno	Katz	87 Panther Place	Trenton	NJ	(515)555-

Execution was successful.   Rows: 15, Cols: 8   0.562 sec   12:19:07

Message   Result1

To save the SQL script and add it to the project:

1.   Make the SQL Editor window that contains the INSERT statements the active window
     and click the **Save** button in the Standard toolbar.
     The **Save As** dialog box opens.

2.   Enter a name for the script, e.g., New_keepers (.sql), select a folder and click the **Save**
     button.
     The **Add to Project** dialog box opens (if the dialog box does not pop up, check whether
     the `Do not show dialog` option is checked in the Newly saved files group box of the
     General options).

Add to Project

Do you wish to add the file to the project?

☐ Always add newly saved files to the project
☐ Do not ask this question again!

[Add to Project]   [Skip Project]
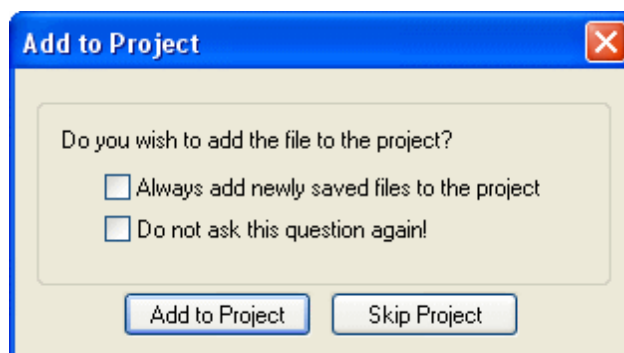
3.   Click the **Add to Project** button.

     Alternatively, or if the **Add to Project** dialog box is not displayed, you can also click the

Skip Project button and do one of the following:

- Click the Project tab, right-click the SQL folder, and select **Add Active File to Project** from the context menu.

- Click the **Add the Active File** icon in the toolbar of the Project window.
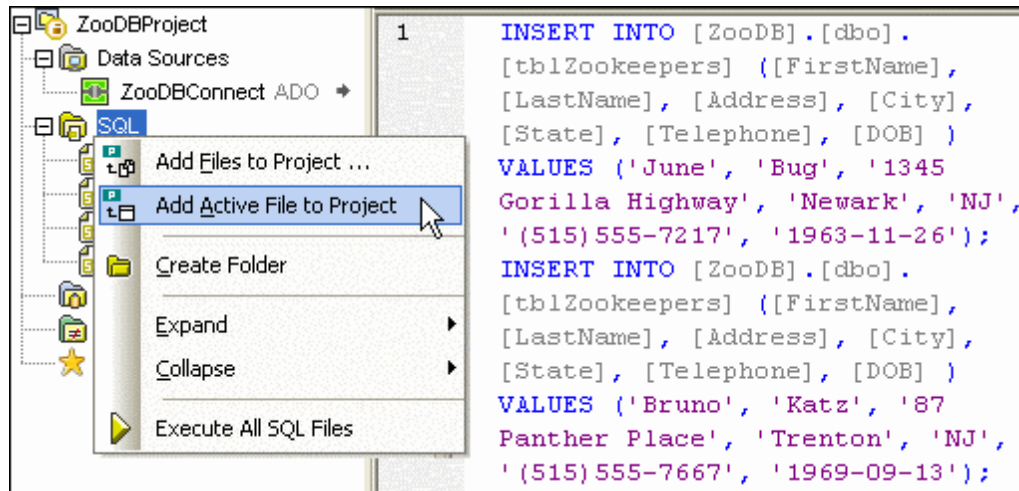
- Right-click the name tab in the SQL Editor and select **Add to Project** from the context menu that opens.
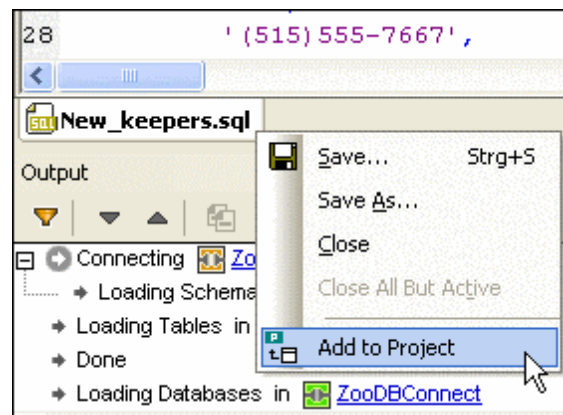
4. Right-click the project's title, `ZooDBProject`, and select **Save Project** from the context menu, or click the **Save Project** icon in the toolbar of the Project window.

Adding Data in the Result Window

A more convenient way to insert data into the database is to add data directly in the Result window.

	SupplierID	SupplierName	SupplierAddress	SupplierCity	SupplierState	SupplierTelephone
7	7	The Pampered Pet	245 Greenaway Street	Chicago	IL	(515)555-0706
8	8	ZooChow Central	3130 Ninth Avenue	NY	NY	(515)555-3345
9		Food4Animals	67 Ella Drive	Boston	MA	(515)555-8647

Note that the Result window has to be enabled for data editing before you can add or delete rows

or edit the data displayed in the result grid. For this purpose, the **Edit Data** command is available in the context menu that opens when you right-click a table or column in the Online Browser.

To add a new food supplier in the Result window:

1. In the Online Browser, right-click the `tblFeedSuppliers` table and select **Edit Data** from the context menu; alternatively, select the table and press **Ctrl+Alt+E**.
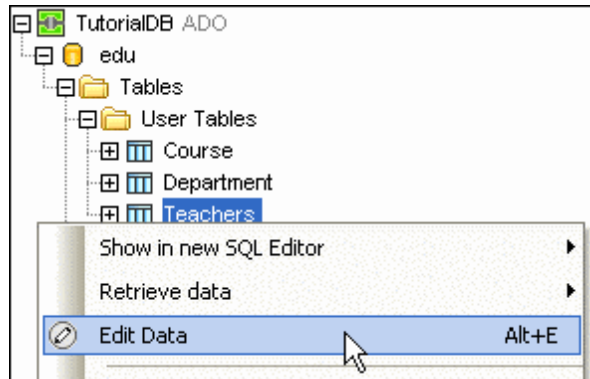
DatabaseSpy generates a SELECT statement in a new SQL Editor window, automatically executes it, and enables the Result window for editing.

2. A popup appears, informing you that editing of data is limited for this table.
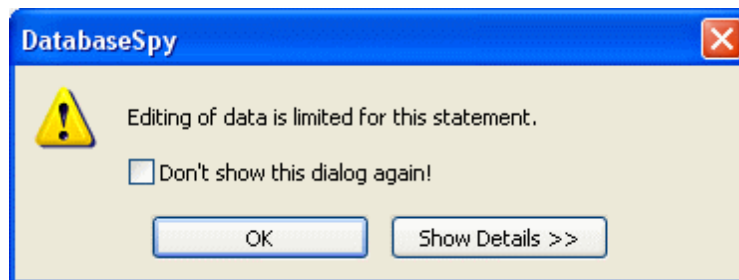
3. Click the **Show Details** button to learn which columns cannot be edited.
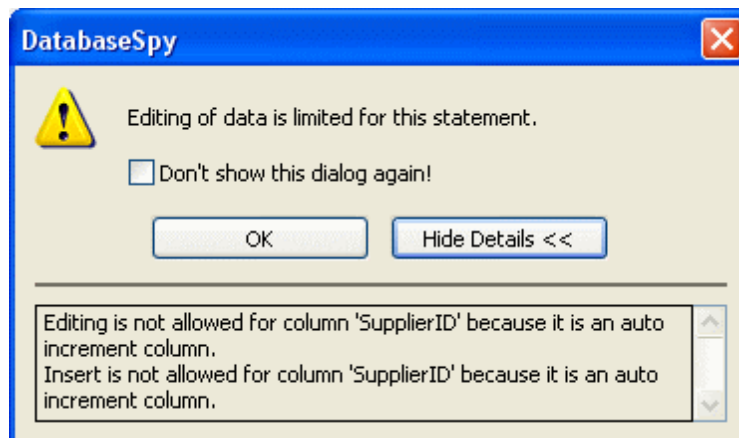
4. Optionally, if you want to suppress this popup in the future, click the `Don't show this dialog again!` check box.

5. Click **OK** to close the popup and display the data in the Result window.

6. In the Result window, click the **Append a new row** icon.

	SupplierID	SupplierName	SupplierAddress	SupplierCity	SupplierState
8	8	ZooChow Central	3130 Ninth Avenue	NY	NY
9					

Please note that the cursor automatically appears in the `SupplierName` column because the value for the `SupplierID` column will be generated automatically when the data is committed to the database.

7. Enter the following data into the newly created row:

    `Food4Animals | 67 Ella Drive | Boston | MA | (515)555-8647`

8. Click the ⬚ Commit ⬚ button to save the changes to the database.

9. Optionally, click the (8) update hyperlink next to tblFeedSuppliers in the Online Browser to update the row count of the table.

4.2.5   Browsing the Database

Now that you populated the tables of the zoo database with data in the previous section of the tutorial, you may want to browse the database and find individual database objects. DatabaseSpy allows you to customize the view of the Online Browser, use filters, and provides the Object Locator if you need to locate a specific object in the database.

Objective

In this section of the tutorial, you will learn how to use filters and how to locate objects in your database project. Specifically, you will learn how to do the following:

- Customize the Online Browser
- Show the row count of tables and views
- Filter database objects in the Online Browser
- Use the Object Locator to locate a specific database object

Commands used in this section

Folders: This command is located in the toolbar of the Online Browser. Click the arrow to the right of the icon to open the drop-down list from where you can select a folder layout for the Online Browser. Please note that the icon changes in accordance with the selected layout.

Row Count | Show/Update: This command is available in the context menu that opens when you right-click a table, view or its parent folder in the Online Browser. Click this command to display the number of rows for the selected table or view, or the number of rows for all tables or views, respectively, that are contained in the selected folder.

Row Count | Clear: This command is available in the context menu that opens that opens when you right-click a table, view or its parent folder in the Online Browser. Click this command to hide the row count for the selected table or view, or for all tables or views, respectively, that are contained in the selected folder.

 Filter folder contents: This command is located in the toolbar of the Online Browser. Clicking this command displays additional filter icons to the right of the folders and allows you to enter filter criteria. The display changes dynamically as you enter the criteria.

 Object Locator: This command is located in the toolbar of the Online Browser. Click this command to display a text box and a drop-down list below the Online Browser.

 Search within/all: This command is located to the right of the drop-down list that appears if the Object Locator is switched on. Click this command to define the focus for the Object Locator. Please note that the icon changes in accordance with the selected focus.

Customizing the Online Browser

The Online Browser window can be configured to show different aspects of the current database. Click the Layouts icon in the icon bar of the Online Browser and select the specific layout you want to use from the drop-down list that opens. The default layout in the Online Browser is **Folders** . Please note that the icon changes in accordance with the selected layout.
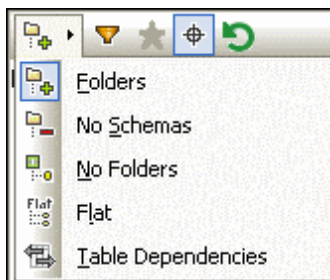
Please note: Filters cannot be defined when either the **No Folders** or the **Table Dependencies** layout is active.

Showing the Row Count

You can get an overview on the number of rows in a table or view without the need to retrieve data first. The row count is displayed directly in the Online Browser and can be updated or cleared at any time, either for single tables or views, respectively, or for all objects in a folder at once.

The row count can be disabled in the [Online Browser options]; it is enabled by default.

To show the row count for a single table:

1. Make sure that the `Optional display of Table and View row counts` check box is activated in the Online Browser options (select the menu option **Tools | Options** or

press **Ctrl+Alt+O** and select the Online Browser page to check).

2. If you hover over tables or views in the Online Browser, the string "(count)" is displayed to the right of any table or view. Select tblAnimals and click on "(count)". Notice how the string turns into a link when you place the mouse cursor over it.

Alternatively, you can also right-click tblAnimals and select **Row Count | Show/Update** from the context menu.

To show the row count for all objects of a folder:

*   With the row counts option activated, right-click the User Tables folder in the Online Browser and select **Row Count | Show/Update** from the context menu.
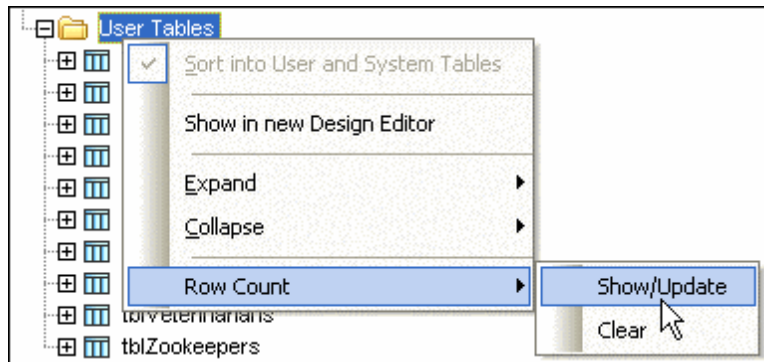
The number of rows is retrieved for all tables that are contained in the User Tables folder.

To hide the row count for tables, views, or folders:

*   Select the database object for which you do not want to show the row count, right-click and select **Row Count | Clear** from the context menu.

Filtering Objects

The application of filters allows you to reduce the number of displayed database objects and thus show only those objects that meet certain filter criteria. You can define different filters for each of the individual folders in the Online Browser.

Please note: Since the filters work on folder level, the filter function is not available in the **No Folders** and **Table Dependencies** layouts.

In this tutorial, you will use a filter to display only tables that contain the string "Animal" in the table name.

To filter for tables containing the string "Animal" in the table name:

1. Make sure you are in **Folders** layout, and expand the folders so you can see the zoo database tables.

2. In the Online Browser, click the **Filter folder contents** icon in the toolbar or press

Ctrl+Alt+F.

Filter icons appear next to the Schemas, Tables and Views folders.
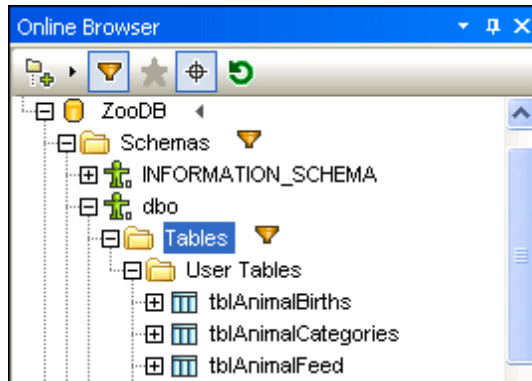
3. Click the Filter icon next to the Tables folder and select **Contains** from the menu that pops up.

A text field appears to the right of the filter icon.

4. Enter "Animal" in the text field. The objects are filtered as you type.

Locating Database Objects

The Object Locator in DatabaseSpy serves for searching for a specific database object by name. When the Object Locator is turned on, you can enter a string in the text box below the Online Browser window and then select a database object from the drop-down list displaying only objects that contain the text entered in the text box. Clicking an entry in the drop-down list selects the corresponding database object in the Online Browser.

Note: To speed up the display of database objects in the Object Locator, only items are displayed that have already been loaded into the Online Browser in the current session. Therefore, if only a few items are displayed or a certain type of database items (e.g., keys or constraints) is missing at all, load these items into the Online Browser by expanding the tree until the desired object is visible. Use the **Expand | Siblings** and **Expand | Children** commands from the context

menu for this purpose.

To display each and every item in the Online Browser:

1. In the Online Browser, expand the schema so that all its subfolders are displayed.

2. Use **Shift+Click** to select all sub-items of the schema.

3. Right-click and select **Expand | Children** from the context menu.

4. Repeat steps 2 and 3 until all subfolders of the schema are expanded.
   Note that in large databases the loading of the items may take a considerable amount of time.

To locate database objects in the Online Browser

1. In the Online Browser, click the **Object Locator** ⊕ icon.
   A text box and a drop-down list appear at the bottom of the Online Browser.

2. Enter the string you want to look for, e.g., "type".
   Clicking the drop-down arrow displays all elements that contain that string.
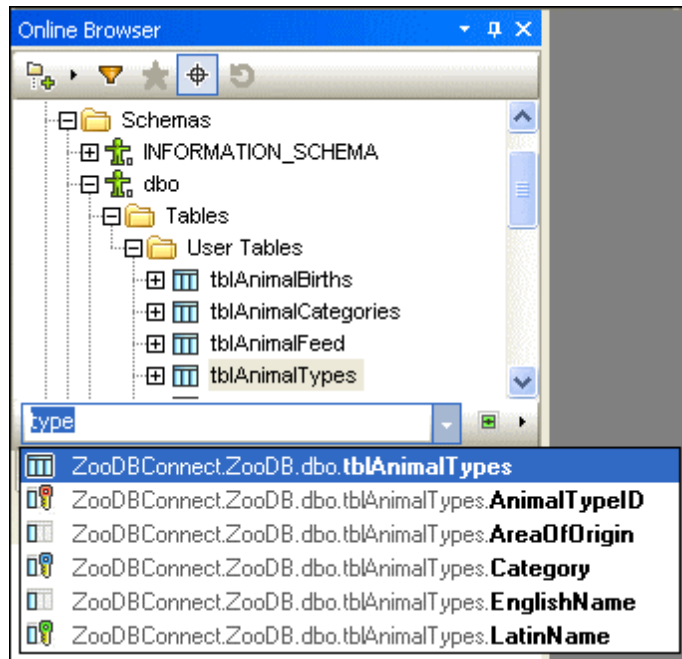
3. Choose an object from the drop-down list to have it selected in the Online Browser.

4. Change the Object Locator context by clicking the arrow icon next to the drop-down list and select one of the options **From current Data Source**, **From focused item**, or **All**.
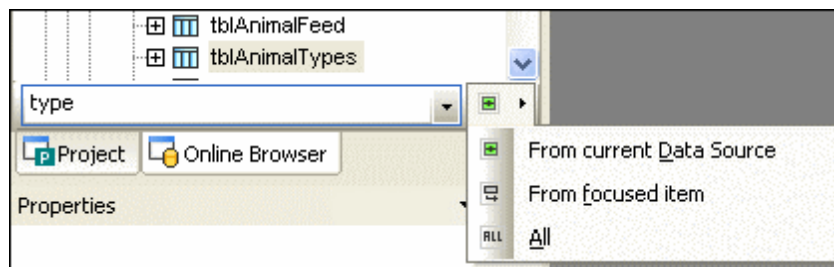
4.2.6   Querying the Database

From time to time, you will want to query your database to see which data is stored in the individual tables. DatabaseSpy provides several possibilities for doing this.

Objective

In this section of the tutorial, you will learn how to query the database. Specifically, you will learn how to do the following:

- Query the zoo database using multiple queries in a script
- Create queries using drag and drop

Commands used in this section

Execute: This command is located in the toolbar of an SQL Editor window and in the **SQL Editor** menu. Click this command to execute the SQL in the SQL Editor window.

Define Target Name: This command is located in the **SQL Script** toolbar and in the **SQL Editor** menu. Clicking this command will insert a comment defining a name for the Result tab before the statement in which the cursor is currently located.

Current statement: This command is located in the toolbar of an SQL Editor window. Click this command to select the statement in which the cursor is currently located.

Using a Script to Query the Database

The SQL file animalType_query.sql that you have added to your database project in one of the previous sections of this tutorial contains a script that uses targets to output the two queries it is composed of to two separate Result tabs and assign a name to each tab. This way, you can execute the script with a single mouse click and have the results displayed in different Result windows.

Targets can be defined by clicking on the **Define Target Name** icon in the SQL Script toolbar and then editing the default target text inserted in the SQL window. Please see the Working with SQL Scripts section for more information.

To query the database using the script animalType_queries.sql:

1. Make sure that the ZooDBConnect connection is active.

2. Click the Project tab.

3. In the SQL folder, double-click the file animalType_queries.sql.
   The file opens in an SQL Editor window.

```
 1        -- target: AnimalTypeCountQuery
 2        SELECT COUNT (ZooDB.dbo.tblAnimalTypes.EnglishName)
 3        FROM ZooDB.dbo.tblAnimalTypes, ZooDB.dbo.
          tblAnimalCategories
 4        WHERE ZooDB.dbo.tblAnimalTypes.Category=ZooDB.dbo.
          tblAnimalCategories.CategoryID AND
 5        ZooDB.dbo.tblAnimalCategories.Category='perch-like';
 6
 7        -- target: AnimalNameQuery
 8        SELECT EnglishName, LatinName
 9        FROM ZooDB.dbo.tblAnimalTypes
10        INNER JOIN ZooDB.dbo.tblAnimalCategories
11        ON ZooDB.dbo.tblAnimalTypes.Category=ZooDB.dbo.
```

4.   In the Properties window, make sure that either "Semicolons" or "SQL Grammar" is
     selected in the `Group statements for execution with` drop-down list.

Properties	▾ ᴗ ✕
🗋 animalType_queries.sql	🎛
▽ Overview	
File Kind	SQL
▽ Options	
SQL-Editor should require semicolons	☐
Group statements for execution with:	Semicolons ▾
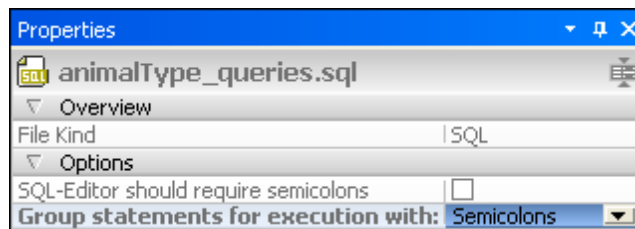
5.   Click the **Execute** ▶ button or press **F5**.
     The results of the two queries appear in separate tabbed windows that use the
     corresponding target names as tab text.

AnimalNameQuery	▾ ✕

	EnglishName	LatinName
1	Night sergeant	Abudefduf concolor
2	Clark's clownfish	Amphiprion clarkii
3	Oscar	Astronotus ocellatus
4	Blue green damselfish	Chromis viridis
5	Full moony	Monodactylus falciformis

✅ Execution was successful Rows: 9, Cols: 2   0.329 sec   11:27:32

🗒 Message 🎞 AnimalTypeCountQuery 🎞 AnimalNameQuery ◁ ▷
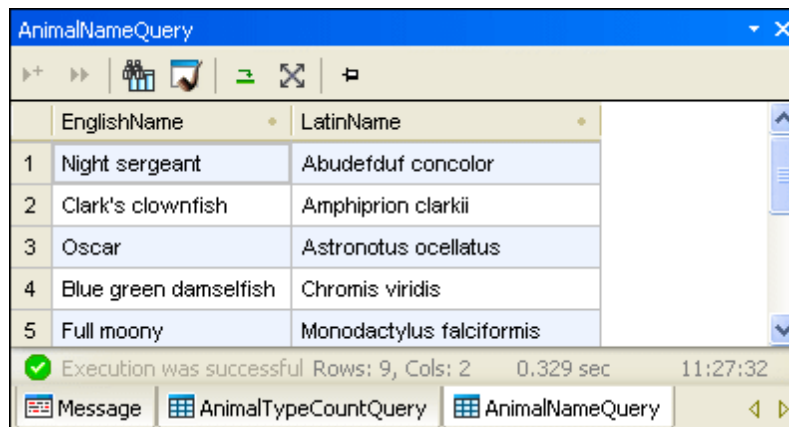
If you do not get separate named result tabs as depicted above, deactivate the **Show
multiple results stacked** 🎛 button in the SQL Editor toolbar.

Using Drag and Drop to Create Queries

DatabaseSpy allows you to generate SELECT statements by dragging a table and dropping it into
an SQL Editor window. To this aim, you should make sure that the default statement generated
for tables is a SELECT statement. Select the menu option **Tools | Options**, go to the **Online
Browser** page and choose **Select** as SQL action for tables.
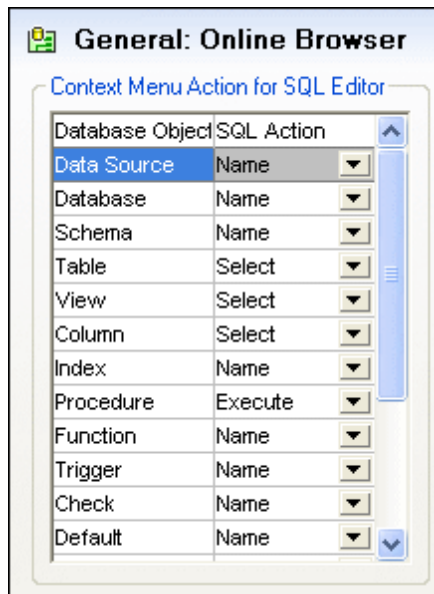
To generate a query based on the table tblZookeepers and specify a target for it:

1. Click the Online Browser tab.

2. Drag and drop the table `tblZookeepers` to the same SQL Editor window where the file animalType_queries.sql is currently open. This generates the following SELECT statement:

   ```
   SELECT [ZookID], [FirstName], [LastName], [Address], [City],
   [State], [Telephone], [DOB] FROM [ZooDB].[dbo].[tblZookeepers];
   ```

3. Place the cursor anywhere in the new SELECT statement. Click the **Current statement** icon in the SQL Editor window. This selects the entire statement.

4. Click the **Define Target name** button in the SQL Script toolbar. The text `--target: Result` appears above the statement.

   ```
   -- target: Result
   SELECT [ZookID], [FirstName], [LastName], [Address],
   [City], [State], [Telephone], [DOB] FROM
   [ZooDB].[dbo].[tblZookeepers];
   ```

5. Change the text "Result" to "Select zookeepers".

   ```
   -- target: Select zookeepers
   SELECT [ZookID], [FirstName], [LastName], [Address],
   [City], [State], [Telephone], [DOB] FROM [ZooDB].
   [dbo].[tblZookeepers];
   ```

6. In the Properties window, make sure that either "Semicolons" or "SQL Grammar" is selected in the `Group statements for execution with` drop-down list.

7. Click the **Execute** button or press **F5**.
   The results of all three queries in the script appear in separate result windows.

Please note: When executing only one of the select statements, make sure that the `--target:` text is selected in addition to the select statement below it, if you want the target window text to be generated. The **Current statement** button will **not** select the `--target:` line!

4.2.7   Using Auto-Completion

In DatabaseSpy, the auto-completion feature assists you when entering SQL statements in the SQL Editor. As you type, you are offered different possible SQL keywords or names of database objects to choose from.

The SQL syntax depends on the specific SQL flavor you are using. This is automatically determined by the connection string you defined using the Connection wizard, i.e., the Create a Database Connection option.

Objective
In this section of the tutorial you will learn how to use the auto-completion feature in DatabaseSpy. Specifically, you will add a new column to the table `tblVeterinarians` while using auto-completion.

Command used in this section

🔲 **SQL Editor**: This command is located in the Standard toolbar; alternatively, you can also choose the menu option **File | New | SQL Editor** or press **Ctrl+N**. Click this command to open a new SQL Editor window that is assigned to the active data source connection.

🔄 **Refresh**: This command is available in the Online Browser as a button, or as an option in the context menu that opens when you right-click a database. Use this command to refresh the database and display new or changed database objects in the Online Browser.

▷ **Execute**: This command is located in the toolbar of an SQL Editor window and in the **SQL Editor** menu. Click this command to execute the SQL in the SQL Editor window.

Adding a New Column to a Table

In the list of veterinarians of our tutorial zoo, currently the only contact information is the phone number, so the zoo authorities also want the e-mail address of the veterinarian. You will add a column, called "Email" of type `varchar(50)`, to the table `tblVeterinarians`.

To create and execute an SQL ALTER statement using auto-completion:

1. Click the **SQL Editor** ⊡ toolbar button to open a new SQL Editor window.
2. Enter "alt". The auto-completion window pops up.
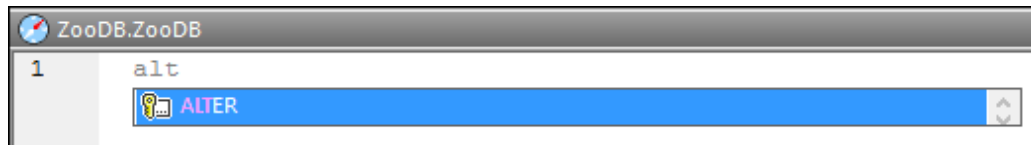
3. Press **Enter**. The word "ALTER" appears in the SQL Editor.
4. Start typing "tab" and add the word "TABLE" after "ALTER". Your SQL statement should now be "ALTER TABLE".
5. Press **Ctrl+Space** to manually trigger auto-completion, select "tblVeterinarians" from the list and press **Space**.
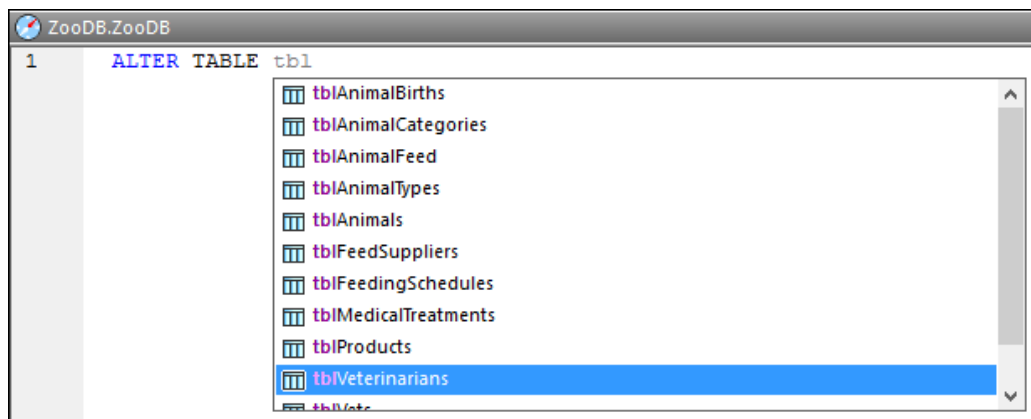
6. Continue creating the SQL statement until you have typed the following text:

```
ALTER TABLE [tblVeterinarians] ADD Email VARCHAR (50)
```

7. Execute the SQL statement by clicking the **Execute** ▷ button or pressing **F5**.

You have now created the new Email column. To view it in the Online Browser, click the **Refresh database** ⟳ button.

4.2.8   Updating Database Data

So far, you have learned how to create tables, insert data and execute queries to retrieve data from these tables. This section of the tutorial shows you how to update data contained in the tables of the Zoo database in the Result window of DatabaseSpy.

Objective

In this section of the tutorial, you will learn how to edit database data in the Result window of the SQL Editor. Specifically, you will learn how to do the following:

- Update a data cell in the Result window
- Delete record sets in the Result window
- Copy a row of records and append it to the result grid as a new row

Commands used in this section

⊘   **Edit Data**: This command is located in the context menu that opens when you right-click a table or column in the Online Browser; alternatively, you can also select the table and press **Ctrl+Alt+E**. Choose this command to generate an SQL SELECT statement that is automatically executed in the SQL Editor and switches the Result window into the Editing mode. This command works for all supported database kinds for both ADO and ODBC connections.

Commit : This command is available as a button in the Result window toolbar. Click this button to commit the changes you have made in the result grid to the database. The **Commit** button is only visible if the Result window is in the Editing mode.

✕   **Delete row**: This command is available as a button in the Result window toolbar. Click this button to delete the row in which the cursor is currently located. Alternatively, you can also right-click a cell in the row to be deleted and choose **Delete row** from the context menu. The **Delete row** command is only available if the Result window is in the Editing mode.

    Select row: The **Selection** command, which is located in the context menu that opens when right-clicking a cell in the result grid, opens a sub-menu from where you can choose the **Row** option. Use this command to select the entire row in which the cursor is currently located.

▤   **Copy selected cells**: This command is located in the context menu that opens when right-clicking a cell in the result grid. Use this command to copy the content of the selected data cells to the clipboard.

▥   **Paste as new rows**: This command is available if the Result window is in the Editing mode and content has been copied to the clipboard beforehand. It is located in the context menu that opens when right-clicking anywhere in the Result window. Choose this command to append a new row to the result grid and automatically insert the clipboard content.

Editing Records in the Result Window

In the previous section of the tutorial, you have added a new column for the e-mail address of the veterinarians to the tblVeterinarians table. Now you will enter the actual e-mail information into this new column.

You can add this information directly in the Result window of the SQL Editor if you first select the tblVeterinarians table for editing.

To edit table records in the Result window:

1. In the Online Browser, right-click the tblVeterinarians table and select **Edit Data** from the context menu. Alternatively, select the table and press **Ctrl+Alt+E**.
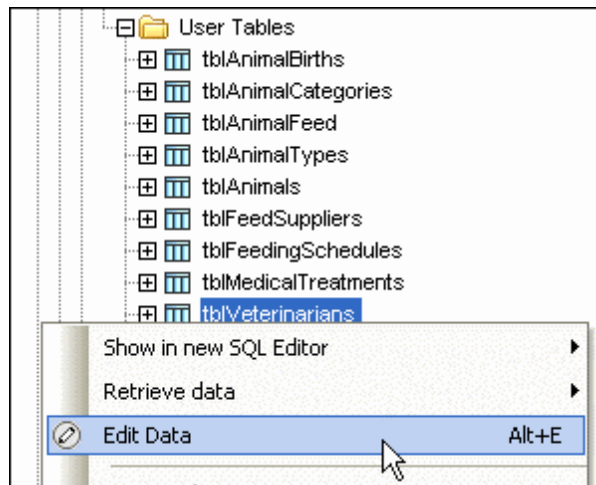
2. A popup appears, informing you that editing of data is limited for this table.
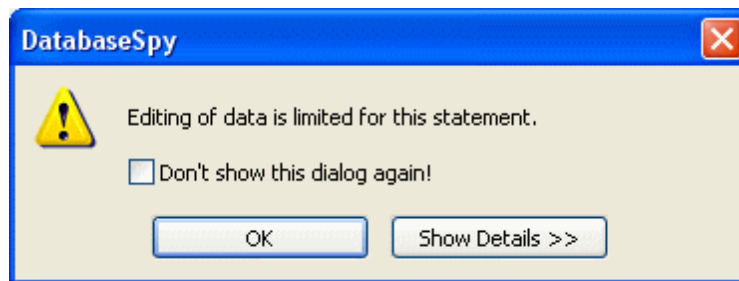
3. Optionally, click the **Show Details** button to learn which columns cannot be edited.

4. Optionally, if you want to suppress this popup in the future, click the `Don't show this dialog again!` check box.

5. Click **OK** to close the popup and display the data in the Result window.

6. In the result grid, double-click the `Email` cell in the first row. A cursor appears in the cell.

7. Double-click the cell again to have the entire content selected and enter "peggy@yahoo.com".

8. Press the **Enter** key. The color of the updated cell changes. To see the full content of the cell, place the mouse cursor over the cell which will then expand.

9. Click the ⬚ Commit ⬚ button to save your changes to the database. The updated cell changes back to its default color and the update is committed to the database.

	Address	City	State	Telephone	DOB	Email
1	12 Bivalve Boulevard	Concord	NH	(515)555-2759	[NULL]	peggy@yahoo.com
2	49 Whaletooth Avenue	Hartford	CT	(515)555-2843	[NULL]	[NULL]
3	31 Marsupial Place	Newark	NJ	(515)555-9352	[NULL]	[NULL]
4	238 Ella Funt Drive	Sacramento	CA	(515)555-7519	[NULL]	[NULL]

Deleting a Table Row in the Result Window
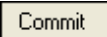
One of the veterinarians will not be available for the zoo anymore and the zoo authorities want to have his records deleted from the database. DatabaseSpy allows you to delete a row from a table with a mouse click, provided that the Result window is in the Editing mode.

To delete a row from a database table:

1. Make sure that the Result window is still in the Editing mode.

2. Click somewhere into the sixth row of the result grid and click the **Delete row** [×] button in the toolbar of the Result window.

	VetID	FirstName	LastName	Address	City	State	Telephone
4	4	Danielle	Batwinger	238 Ella Funt Drive	Sacramento	CA	(515)555-7519
5	5	Robin	Fowler	183 Lionheart Drive	Albuquerque	NM	(515)555-2810
6	6	Jacob	Fisher	87 Bearpaw Circle	Milwaukee	WI	(515)555-7209

The background color of the row as well as the text color change to indicate that this row is to be deleted from the database.

3. Click the [Commit] button. The row is removed from the result grid and deleted from the database.

	VetID	FirstName	LastName	Address	City	State	Telephone
3	3	Victoria	Wormworth	31 Marsupial Place	Newark	NJ	(515)555-9352
4	4	Danielle	Batwinger	238 Ella Funt Drive	Sacramento	CA	(515)555-7519
5	5	Robin	Fowler	183 Lionheart Drive	Albuquerque	NM	(515)555-2810

Copying Table Rows

In the `tblMedicalTreatments` table, the zookeepers in our tutorial zoo keep records about the medical treatments of the zoo animals. If a disease occurs repeatedly, you can copy a previous entry in this table and append it to the result grid as a new row.

	TreatmentID	TreatmentDate	AnimalID	Problem	VetID
1	1	2006-01-09T00:00:00	5000	Heartworms	1
2	2	2005-02-28T00:00:00	5001	Equine Infectious Anemia	4
3	3	2005-12-13T00:00:00	5003	Erysipelas	6
4	4	2005-06-17T00:00:00	5004	Newcastle Disease	2
5	5	2005-06-30T00:00:00	5004	Newcastle Disease	6
6	6	2005-12-31T00:00:00	5005	Bird Flu	6
7	7	2006-02-01T00:00:00	5006	tetanus vaccination	1
8	8	2006-02-01T00:00:00	5007	tetanus vaccination	1

In the screenshot above, you can see that the zoo's hill mynah (AnimalID 5004) suffered from the Newcastle Disease already twice. In this section of the tutorial, you will copy a previous treatment and add it to the medical treatments record.

To copy a row and append it to the result grid:

1. In the Online Browser, right-click the `tblMedicalTreatments` table and select **Edit Data** from the context menu.

2. In the Result window, right-click into the fifth row and select **Selection | Row** from the context menu.

3. Right-click again and select **Copy selected cells** from the context menu.

4. Right-click anywhere in the result grid and select **Paste as new rows** from the context menu.

5. Double-click the `TreatmentDate` cell of the new row and enter the new date of the treatment.

	TreatmentID	TreatmentDate	AnimalID	Problem	VetID
6	6	2005-12-31T00:00:00	5005	Bird Flu	6
7	7	2006-02-01T00:00:00	5006	tetanus vaccination	1
8	8	2006-02-01T00:00:00	5007	tetanus vaccination	1
9		2007-06-30T00:00:00	5004	Newcastle Disease	6

6. If required, change the `VetID` table if a different veterinarian is responsible for the treatment.

7. Click the   Commit   button to actually add the new row to the database.

4.2.9   Designating Database Objects as Favorites

DatabaseSpy's Favorites feature allows you to put database objects into the **Favorites** folder so that you have quick access to them.

Objective

In this section of the tutorial, you will learn how to add tables that need to be updated frequently to the Favorites folder of a database project.

Commands used in this section

> **Add to/Remove from Favorites**: This command is located in the context menu that opens when you right-click a table or column in the Online Browser. Click this command to add the selected item to the Favorites folder in the Project window.

> ⭐ **Show Favorites**: This command is located in the toolbar of the Online Browser. Clicking this command toggles the Favorites view on and off. Note that this icon is unavailable if no items have been added to the Favorites folder yet.

Adding Objects to the Favorites Folder

Some of the tables in the zoo database need to be queried or to have their data updated more often than others. For example, `tblAnimalFeed` needs to be updated every time the amount of a type of feed that the zoo keeps in storage changes. `tblMedicalTreatments` needs to be updated whenever a veterinarian visits. Adding these tables to the Favorites folder in the Project

window provides you with quick access to these tables.

To add tables to the Favorites folder:

1. In the Online Browser, right-click table `tblAnimalFeed` and select **Add to/Remove from Favorites** from the context menu, or select the table and press **Ctrl+F2**.

2. Select the menu option **Tools | Options** and, on the Online Browser page, make sure that the `Enable full row selection` check box is activated. Click **OK** to confirm your change, if applicable.

3. In the Online Browser, move the mouse over table `tblMedicalTreatments` and click the favorites icon that appears to the right of the table name.
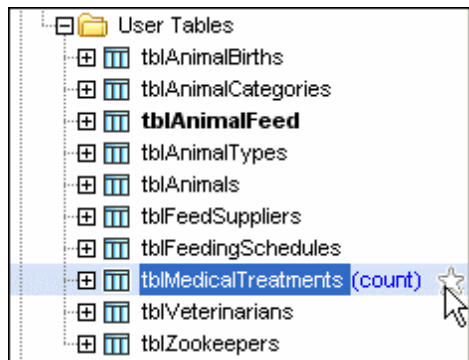
Tables that have been added to the Favorites folder are shown in bold type in the Online Browser.

4. Click the **Show Favorites** icon to switch to the favorites view.
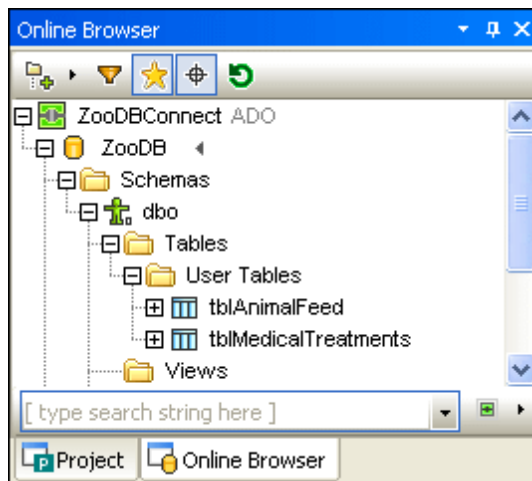
Note that the **Show Favorites** icon is now active and that only favorite items are shown in the Online Browser.

5. Click the **Show Favorites** icon again to switch back to the normal Online Browser view.

6. Click the Project tab to change to the Project window.
   Note that the favorite items have been added to the Favorites folder in the Project window.
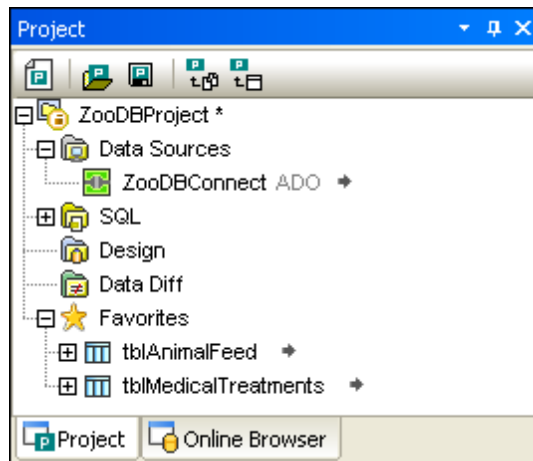
7. Click the **Show in Online Browser** ⬆ icon to the right of an item in the Favorites folder to display the corresponding database object in the Online Browser.

4.2.10 Exporting Database Data

Objective

In this section of the tutorial, you will learn how to export data from the zoo database. Specifically, you will learn how to do the following:

- Select tables for the export to XML files
- Preview the data to be exported
- Specify the path for the export file

Commands used in this section

Export: This command is located in the Tools toolbar; alternatively, you can also choose the menu option **Tools | Export database data...** or press **Ctrl+E**. Clicking this command opens the **Export database data** dialog box.

Exporting Tables to XML

The **Export database data** dialog box allows you to select individual tables for export and choose the desired export format. Furthermore, you can specify a folder where the export files are to be stored, and preview the data before actually exporting it.
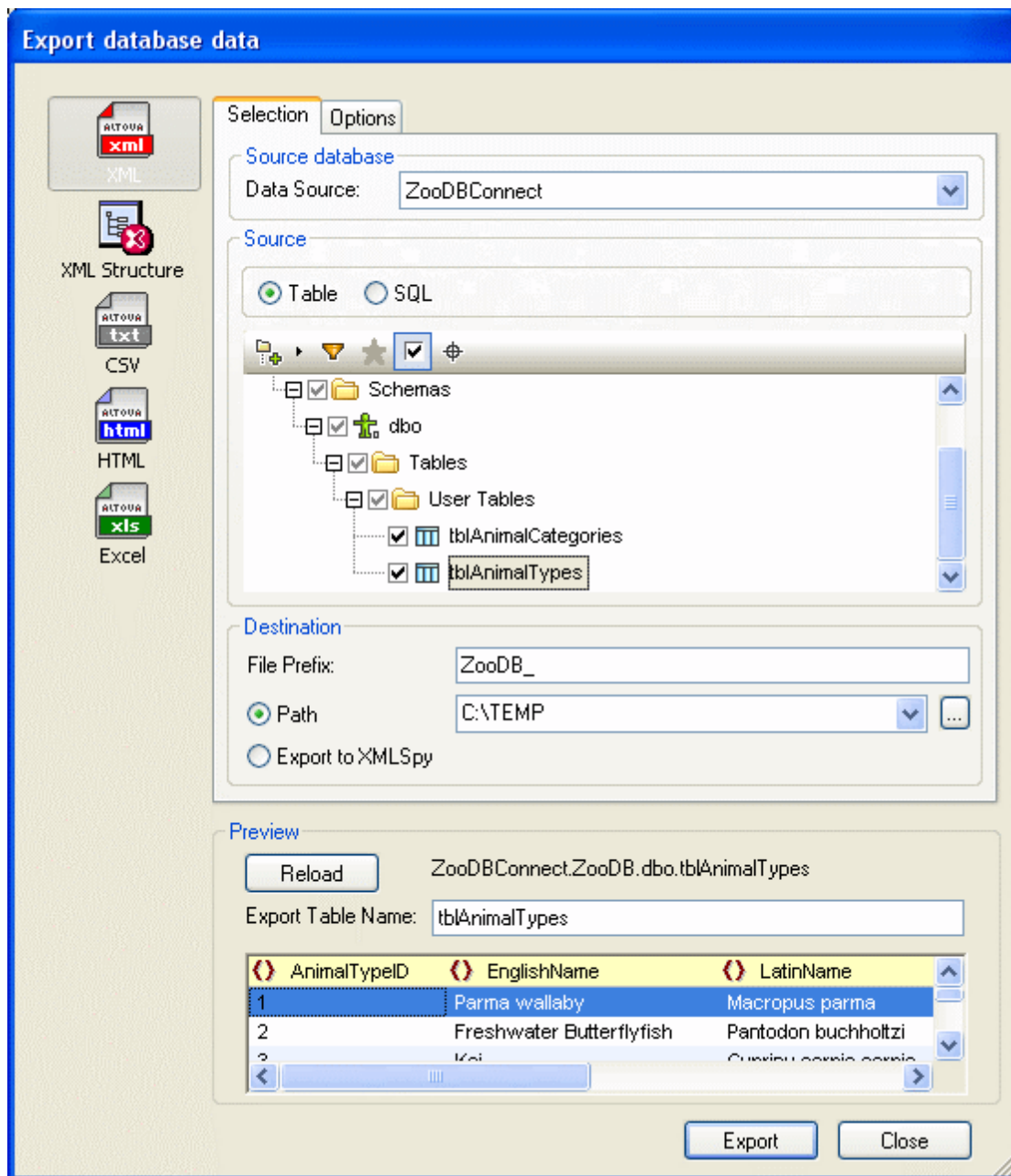
To export table data to XML:

1. Click the **Export** ⬚ button in the Tools toolbar. The **Export database data** dialog box opens.

2. Click the **XML** icon at the left side.

3. In the Source group box, click the respective check boxes of `tblAnimalCategories` and `tblAnimalTypes`.
   Note that Filters, Favorites, and Locators can be used to limit the amount of tables in this tree.

4. Optionally, activate the **Show checked objects only** ⬚ check box to hide objects not

selected for export.

5. In the Destination group box, enter "ZooDB_" as a prefix that will be added to the exported tables.

6. Select the destination path of the XML files, C:\TEMP in this case.

7. Select one of the tables to be exported and click the **Preview** button in the Preview group box to preview the table before exporting it.

8. Click **Export**.
    A separate XML file for each of the selected tables is created at the specified location.

9. Double click one of the exported XML files to check its contents.

```
 1     <?xml version="1.0" encoding="UTF-8"?>
 2     <Import name="tblAnimalTypes" query="SELECT [AnimalTypeID],
 3     ">
 4        <!--MS SQL Server-->
 5        <!--Column name:AnimalTypeID type:int-->
 6        <!--Column name:EnglishName type:varchar maxlength:50-->
 7        <!--Column name:LatinName type:varchar maxlength:50-->
 8        <!--Column name:AreaOfOrigin type:varchar maxlength:50-->
 9        <!--Column name:Category type:int-->
10        <Row>
11           <AnimalTypeID>1</AnimalTypeID>
12           <EnglishName>Parma wallaby</EnglishName>
13           <LatinName>Macropus parma</LatinName>
14           <AreaOfOrigin>Australia</AreaOfOrigin>
15           <Category>1</Category>
16        </Row>
```

4.2.11   Comparing Database Schemas

The schema comparison feature in DatabaseSpy allows you to compare and merge the structure of different schemas of a database or schemas in different database types. In our zoo tutorial, you will compare the live database with a test environment that runs on the same database server.

Objective

In this section of the tutorial, you will learn how to compare the structure of two schemas of a database and merge differences between the schemas. Specifically, you will learn how to do the following:

* Add a schema to the zoo database and add tables using an SQL script
* Set up a database schema comparison and select items for comparison
* Review and correct the mapping
* Compare the structure of two schemas of the zoo database
* Show and examine the differences
* Merge the differences between the two schemas

Commands used in this section

Open File...: This command is located in the **File | Open** menu and opens the Windows **Open** dialog box. Alternatively, you can also use the keyboard shortcut **Ctrl+O**.

Execute: This command is located in the toolbar of an SQL Editor window and in the **SQL Editor** menu. Click this command to execute the SQL in the SQL Editor window.

Refresh: This command is available in the Online Browser as a button, or as an option in the context menu that opens when you right-click a database. Use this command to

refresh the database and display new or changed database objects in the Online Browser.

Schema Comparison: This command is located in the Standard toolbar; alternatively, you can also choose the menu option **File | New | Schema Comparison**. Click this command to open a new Schema Comparison window and select database items for comparison. Note that at least one active connection to a data source is required to open a new schema comparison.

Start Comparison: This command is located in the toolbar of the Schema Comparison window; alternatively, you can also choose the menu option **Schema Comparison | Compare items** or choose **Compare items** from the context menu that opens when you right-click the title bar of a component. Click this command to run a comparison of all mapped items in the comparison components.

Toggle Message Window: This command is located in the Schema Comparison window toolbar. Use this command to switch the display of the Message window on and off.

Collapse items: This command is available in the **Schema Comparison** menu, or as an option in the context menu that opens when you right-click the title bar of a comparison component. Use this command to collapse all items in both components.

Show merge script: Left to Right: This command is available in the **Schema Comparison** menu, or as an option in the context menu that opens when you right-click the title bar of, or a database item in, a comparison component. Use this command to show the merge script that copies the changes in the left component into the right component. If called from the title bar of a component or when no item is selected, this command will generate the merge script for all items in the component.

Show merge script: Right to Left: This command is available in the **Schema Comparison** menu, or as an option in the context menu that opens when you right-click the title bar of, or a database item in, a comparison component. Use this command to show the merge script that copies the changes in the right component into the left component. If called from the title bar of a component or when no item is selected, this command will generate the merge script for all items in the component.

Adding a Schema

The tutorial folder of you DatabaseSpy installation contains an SQL script that will create a new schema in your zoo database and add all zoo tables to it. You will open and execute this script in the SQL Editor.

To add schema "test" to the zoo database:

1. Select the menu option **File | Open | Open File...** or press **Ctrl+O** to open the Windows **Open** dialog box.

2. Navigate to the tutorial folder (\Altova\DatabaseSpy2017\DatabaseSpyExamples\Tutorial sub-folder of the (My) Documents folder), select CreateTestSchema.sql and click **Open**. The script opens in a new SQL Editor window.

3. In the Properties window, select "ZooDBConnect" from the `Data Source` drop-down list.

4. Click the **Execute** button in the SQL Editor toolbar or press **F5**.

5. In the Online Browser, click the **Refresh** button in the toolbar or right-click ZooDB

and select **Refresh** from the context menu.
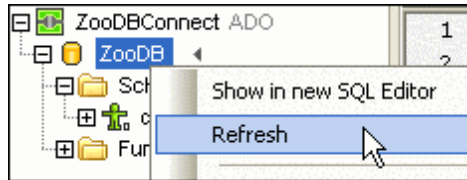
Setting Up a Database Schema Comparison

DatabaseSpy provides a Schema Comparison window which is used to select schemas and tables, and to map database items, start a comparison, and display the merge script for the compared items. Opening a Schema Comparison window and selecting the schemas and database items to be compared is usually done in a single step.

To select schemas for comparison:

1.   Click the **Schema Comparison** button in the Standard toolbar or select the menu option **File | New | Schema Comparison**. A new Schema Comparison window is opened and the **Select Database Objects for Comparison** dialog box appears, where the first data source connection, ZooDBConnect, is preselected in the `Data Source` drop-down list, and the **Left Side** button is activated by default.

2.   Expand the tree structure in the selection dialog box until the schemas are displayed, expand schema dbo and activate the `User Tables` check box. This selects all tables in the User Tables folder.
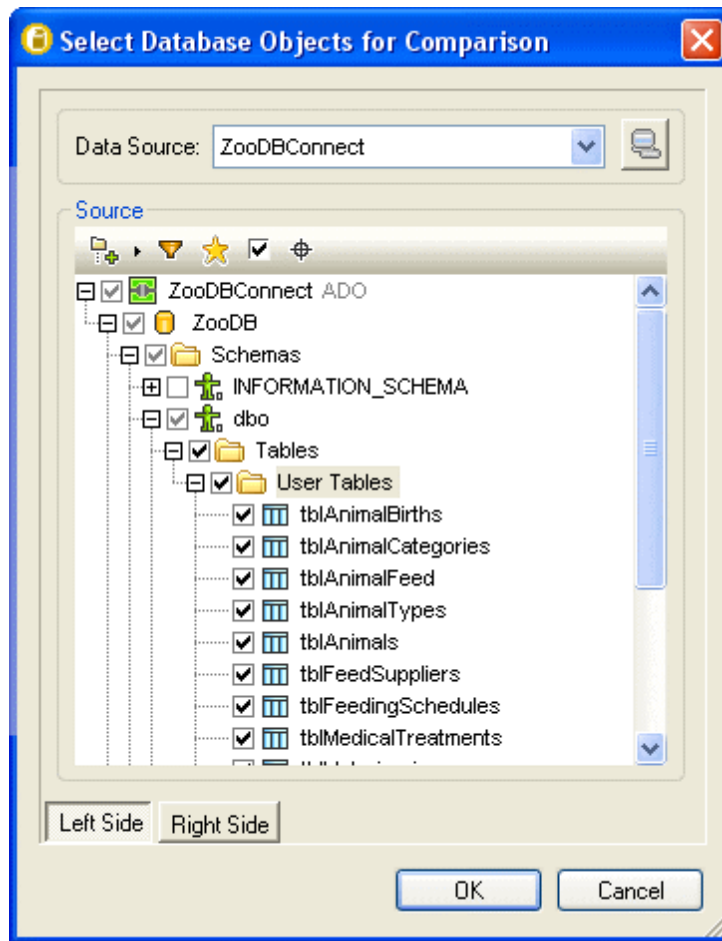
3. Click the **Right Side** button, expand schema test and also activate the `User Tables` check box.
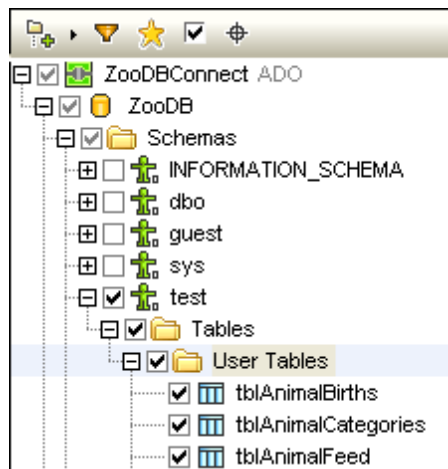
4. Click **OK**. The tables appear in two separate components of the Schema Comparison window, and matching items are mapped automatically.

5. Select the menu option **Schema Comparison | Collapse items** or select this command from the context menu that opens when you right-click the header of either component in the Schema Comparison window.
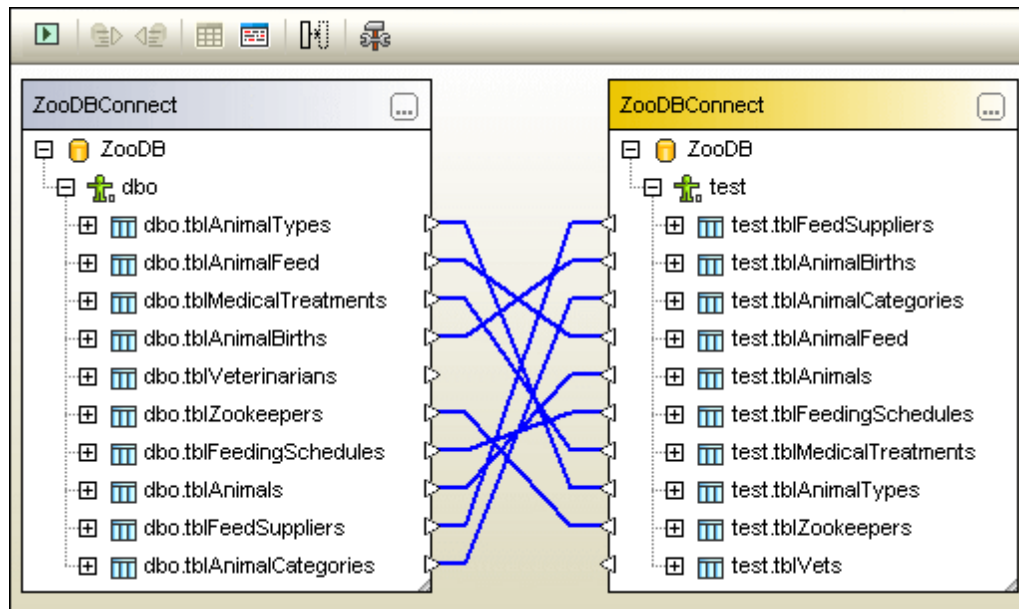
6. Optionally, if the sort order is different in the left and right component, select the menu option **Schema Comparison | Sort items | Ascending** or select this command from the context menu that opens when you right-click the header of either component in the Schema Comparison window.

Mapping Items

Normally, corresponding items are mapped automatically in DatabaseSpy, however if, for example, different naming conventions are applied on the compared databases, automatic mapping might fail. In this case, you can still map corresponding items manually, by drawing a line between the respective items.

If you check the sorted tables in your database schema comparison, you will notice that item dbo.tblVeterinarians is not mapped to test.tblVets. Obviously, the difference in the name of the two items was too big for the automatic mapping to identify these two items as a comparison pair.
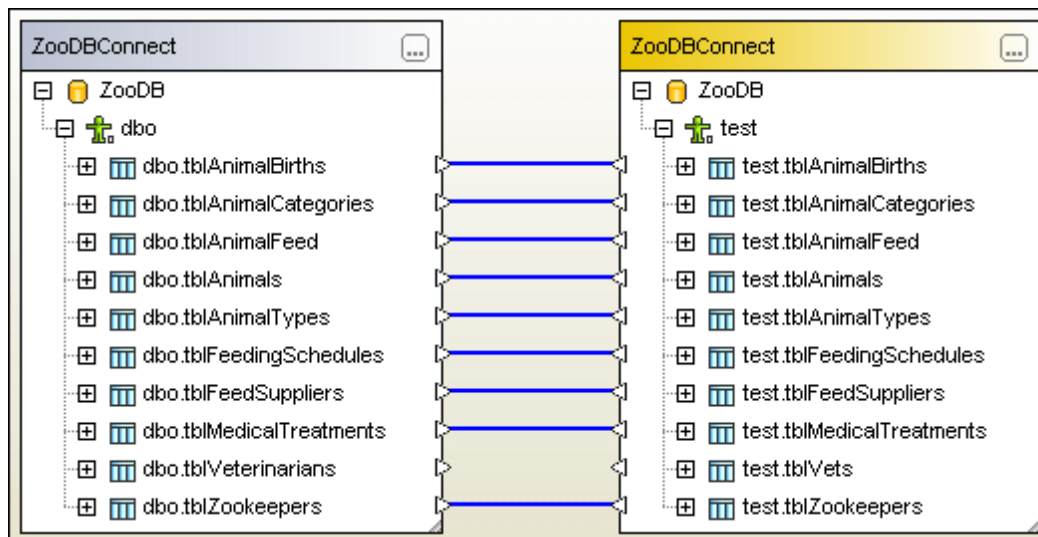
We will now take a closer look on these two items to determine whether or not they can be

considered corresponding. Double-click the items or click the plus sign to expand the table in the comparison components. Since the columns that are contained in the table are pretty similar in the left and right components, we can map them and hence include them in the comparison. Note that all child items of the tables that are equal on both sides will be mapped automatically when you draw the connection line between the parent.
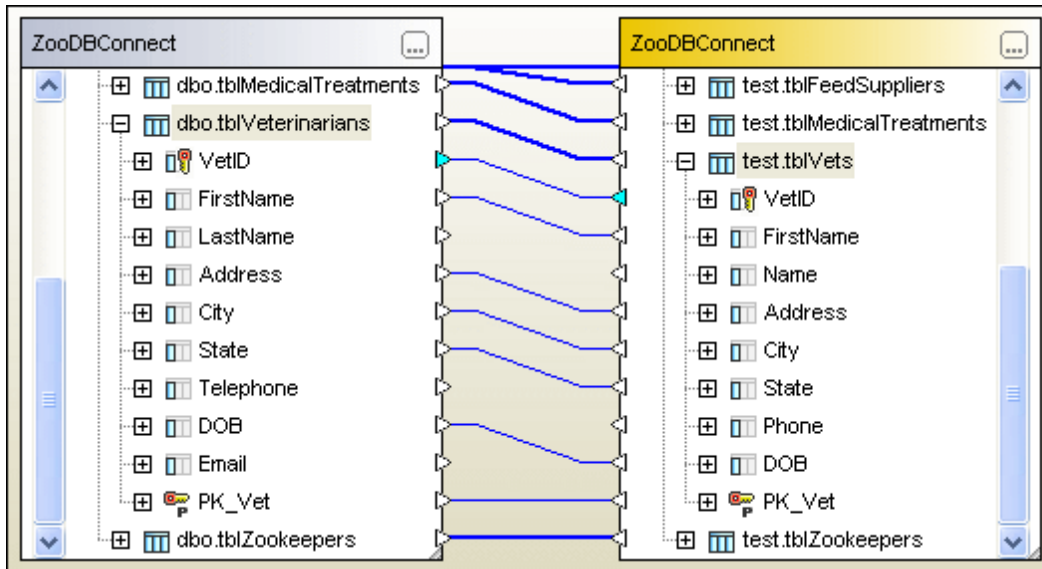
Two column pairs, LastName/Name and Telephone/Phone, are still unmapped, if you further expand these items, you will see that data types as well as Nullable constraints are equal on both sides and that these items can also be mapped safely.
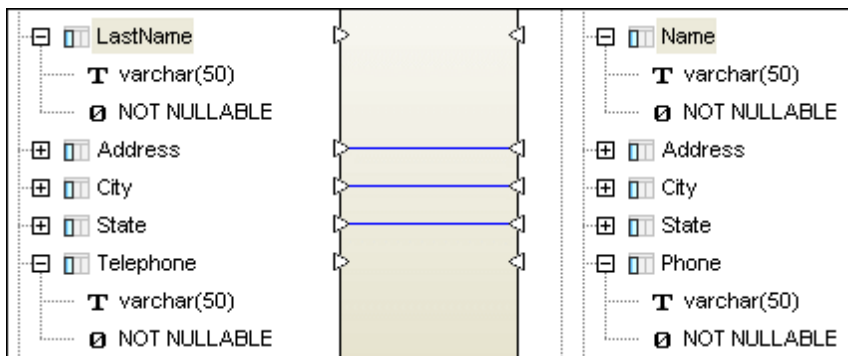
To manually map dbo.tblVeterinarians and test.tblVets:

1. In the Schema Comparison window, click the triangle next to dbo.tblVeterinarians and, keeping the mouse button pressed, draw a line to test.tblVets. Release the mouse button when the cursor changes its shape.
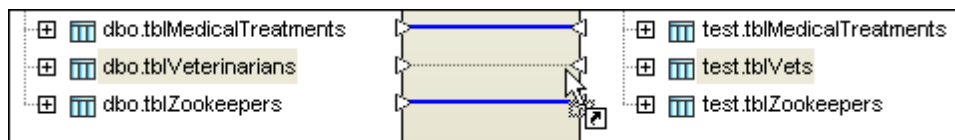
2. Double-click either dbo.tblVeterinarians or test.tblVets to expand the table in both components and map column LastName to column Name as described in step 1.
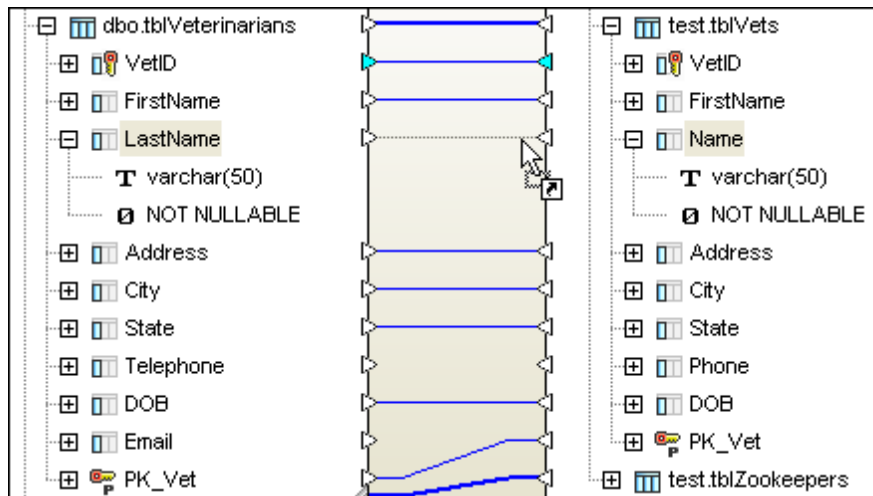
3. In the same way, map column Telephone to column Phone.

Starting a Comparison

You can start a comparison from the **Schema Comparison** menu or by clicking the **Start Comparison** button in the toolbar of the Schema Comparison window. Alternatively, you can also right-click the title bar of either component and choose **Compare items** from the context menu. This way, you run a comparison for all mapped items in the database comparison components. DatabaseSpy displays comparison result icons that indicate whether a pair of mapped items is equal or contains differences.
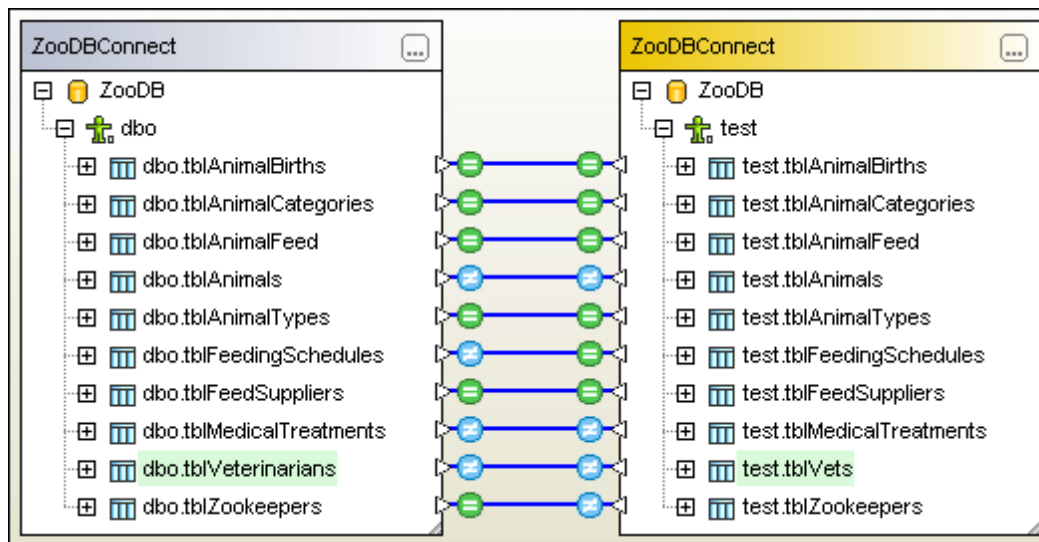
Please note that there are three different kinds of differences displayed in the above screenshot:

- tblAnimals and tbl MedicalTreatments show a Different sign on both sides of the comparison and the table names are not highlighted:
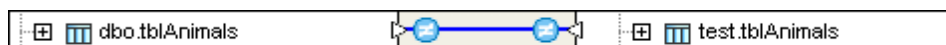
This indicates that the table name itself is equal, however, one or more child items are different. Expand the table to learn where the differences occur.
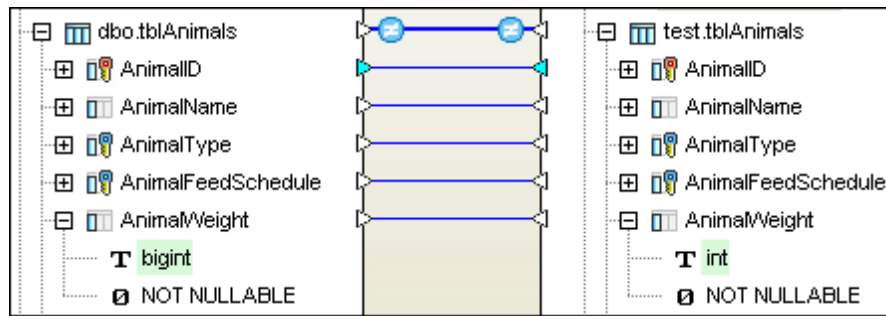
- tblFeedingSchedules and tblZookeepers show a Different sign on one side and an Equal sign on the other side of the comparison; the table names are not highlighted:

  This indicates that the tables are equal but there is at least one additional item in the table that shows the Different sign. If you expand this table, these additional items are not mapped and indicated with a question mark .
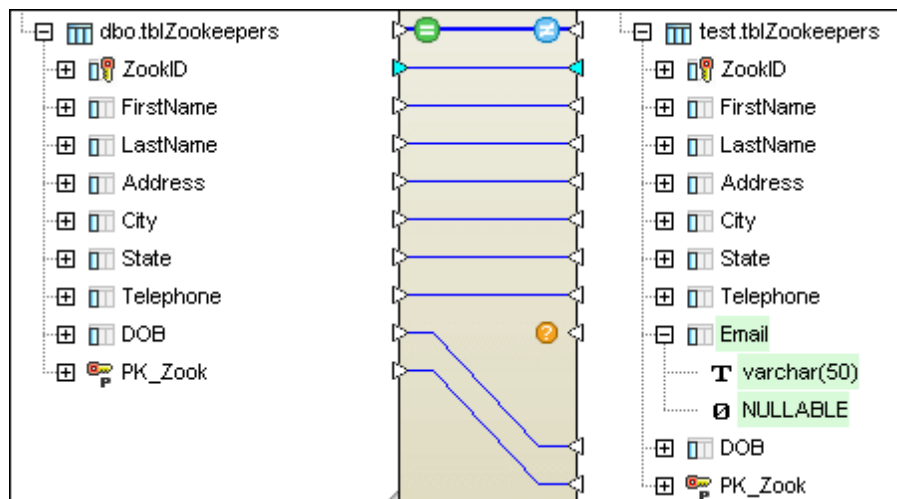
- dbo.tblVeterinarians and test.tblVets, respectively, show a Different sign and their table names are highlilghted:
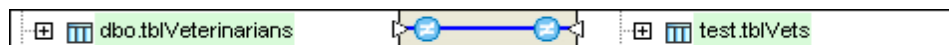
  This indicates that the table names are different but the tables have nevertheless been (manually) mapped. Expand the tables to see if any child items are also different.
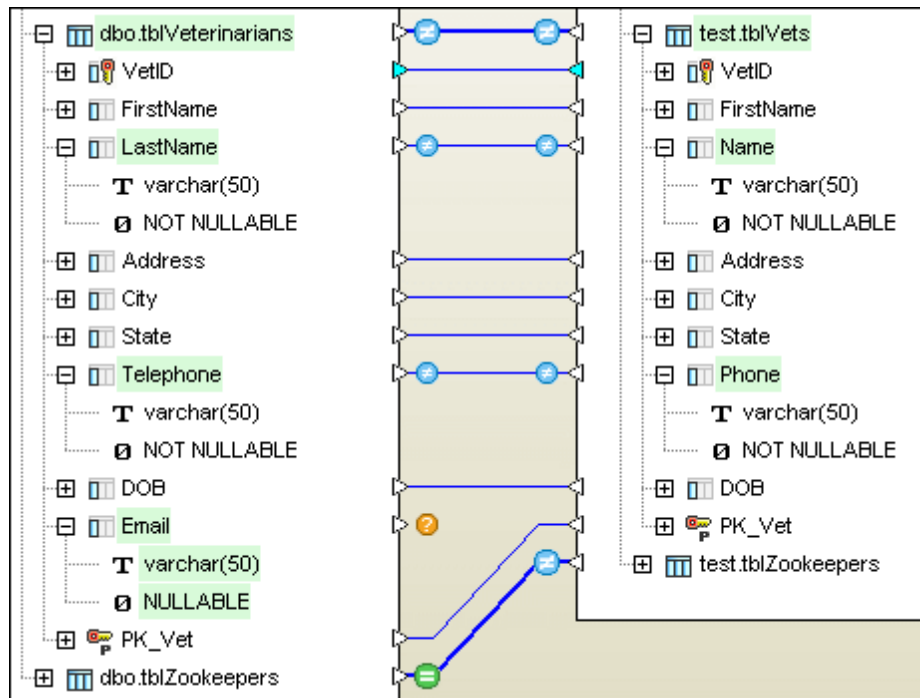
A summary of the comparison result on table level is displayed in the Message window. Click the

Toggle Message Window 🖼 button if the Message window is not displayed.
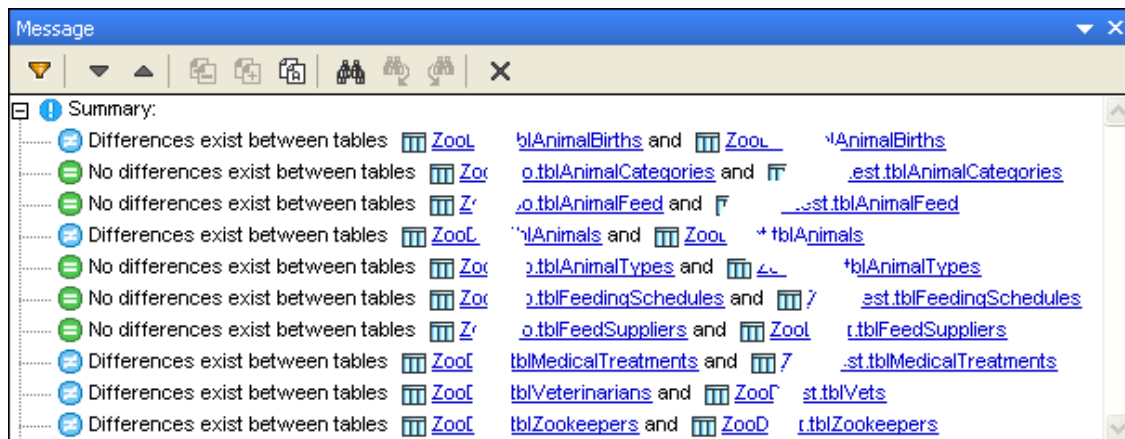
To view the differences in more detail—that is, on item level—you have to examine the comparison results in the components. All items that contain differences are automatically expanded in the components so as to allow for easy identification of differences.

To compare all mapped items:

Do one of the following:

- Select the menu option **Schema Comparison | Compare items**.

- Click the **Start Comparison** 🖼 button in the toolbar of the Schema Comparison window.

- Right-click the title bar in either comparison component and choose **Compare items**

from the context menu.

Examining the Differences

When you start a schema comparison in DatabaseSpy, all items in the comparison components are expanded and scrollbars are displayed so that you are able to scroll through the items and check for differences. DatabaseSpy uses icons to identify tables that contain differences, and color coding to indicate the particular difference itself. So, if you are examining your comparison result for differences, we recommend a top-down approach: Start at the top and collapse any table that does not contain differences. If a table with differences appears do the same on item level, that is, collapse all the items that are equal on both sides of the comparison. Eventually, only different items in tables that contain differences are displayed.
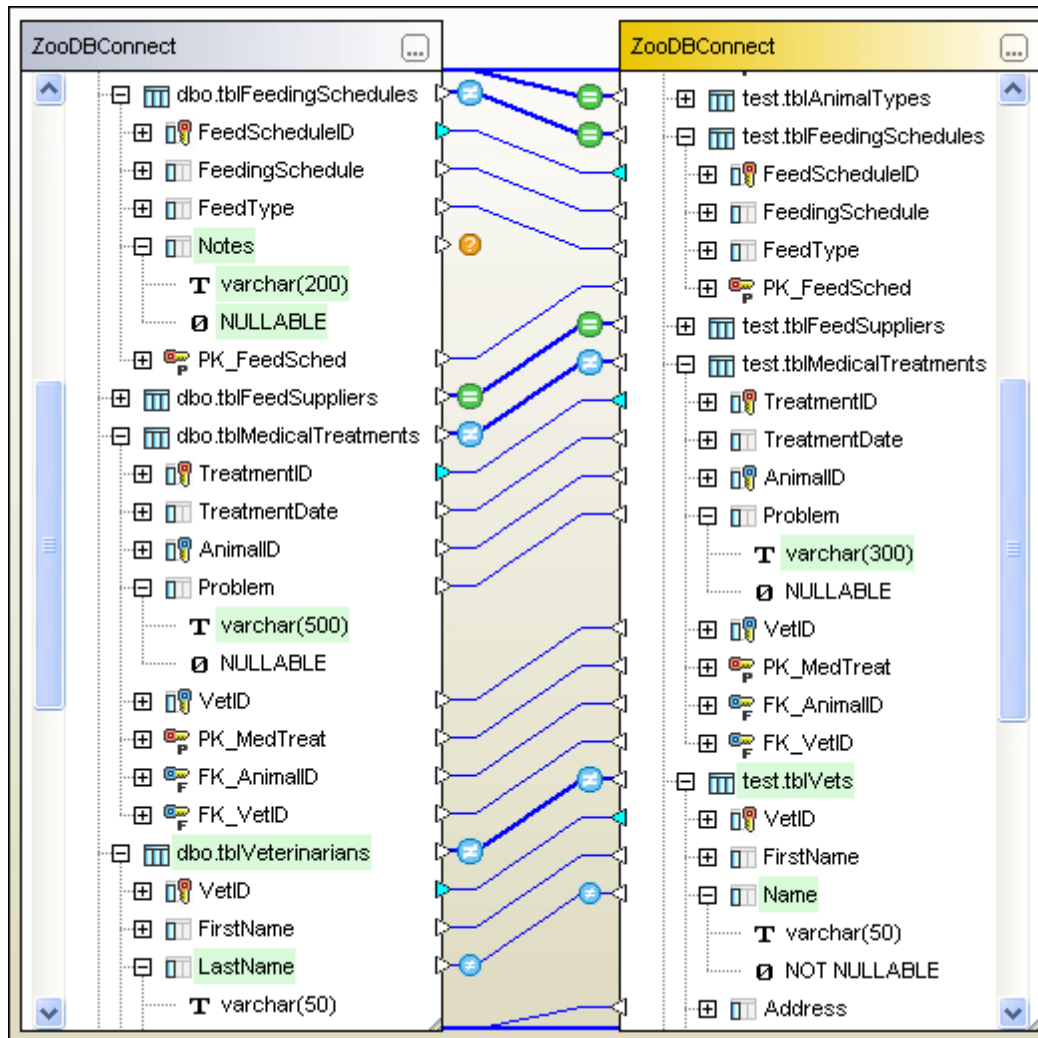
Merging Schemas

After you have started a comparison and examined the differences between the schemas, you can merge the two schemas. Since changes to the database structure cannot be easily undone, DatabaseSpy will not directly perform a merge but display the merge script in an SQL Editor window where you can review the script and execute it if you are sure that the changes to the schema structure should be committed to the database.

You can either merge all differences at once (i.e., create one merge script that contains all the changes to the schema structure) or choose one or more particular differences to merge (i.e., create a merge script that contains selected changes). The merge itself can be performed in both directions, that is, from left to right or from right to left. You can therefore also decide to mix the schema structure by merging some changes from left to right and others from right to left.

The screenshot below shows the merge script for copying the structure of schema "dbo" to schema "test" (merge left to right).

```
1      SP_RENAME '[test].[tblVets]','tblVeterinarians';
2
3      SP_RENAME '[test].[tblVeterinarians].[Name]',
       'LastName','COLUMN';
4
5      SP_RENAME '[test].[tblVeterinarians].[Phone]',
       'Telephone','COLUMN';
6
7      ALTER TABLE [ZooDB].[test].[tblAnimals] ALTER
       COLUMN [AnimalWeight] bigint NOT NULL;
8
9      ALTER TABLE [ZooDB].[test].[tblMedicalTreatments]
       ALTER COLUMN [Problem] varchar (500) COLLATE
       SQL_Latin1_General_CP1_CI_AS NULL;
10
11     ALTER TABLE [ZooDB].[test].[tblVeterinarians] ADD
       [Email] varchar (50) COLLATE
       SQL_Latin1_General_CP1_CI_AS NULL;
12
13     ALTER TABLE [ZooDB].[test].[tblFeedingSchedules]
       ADD [Notes] varchar (200) COLLATE
       SQL_Latin1_General_CP1_CI_AS NULL;
```

A different merge script is generated if you decide to merge the changes from right to left:

```
1      ALTER TABLE [ZooDB].[dbo].[tblAnimalBirths] DROP CONSTRAINT
       [FK_VetID2];
2
3      SP_RENAME '[dbo].[tblVeterinarians]','tblVets';
4
5      SP_RENAME '[dbo].[tblVets].[LastName]','Name','COLUMN';
6
7      SP_RENAME '[dbo].[tblVets].[Telephone]','Phone','COLUMN';
8
9      ALTER TABLE [ZooDB].[dbo].[tblAnimals] ALTER COLUMN
       [AnimalWeight] int NOT NULL;
10
11     ALTER TABLE [ZooDB].[dbo].[tblMedicalTreatments] ALTER COLUMN
       [Problem] varchar (300) COLLATE SQL_Latin1_General_CP1_CI_AS
       NULL;
12
13     ALTER TABLE [ZooDB].[dbo].[tblZookeepers] ADD [Email] varchar
       (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL;
14
15     ALTER TABLE [ZooDB].[dbo].[tblAnimalBirths]
16         ADD CONSTRAINT [FK_VetID2]
17             FOREIGN KEY ([Veterinarian])
18                 REFERENCES [ZooDB].[dbo].[tblVets] ([VetID]) ;
```

In this section of the tutorial, you will rename test.tblVets to test.tblVeterinarians (merge left to right) and add the e-mail column to both dbo.tblZookeepers (merge right to left) and test.tblVeterinarians (merge left to right).

To merge schemas in the zoo database:

1. Select the menu option **Schema Comparison | Collapse items** or right-click the title bar of either component and choose **Collapse items** from the context menu.

2. Expand database and schema and select either dbo.tblVeterinarians or test.tblVets.

3. Select the menu option **Schema Comparison | Show merge script: Left to Right** or right-click the table in the component and choose **Show merge script: Left to Right** from the context menu. The merge script is displayed in a new SQL Editor window.

ZooDBConnect.ZooDB

```
1      SP_RENAME '[test].[tblVets]','tblVeterinarians';
2
```
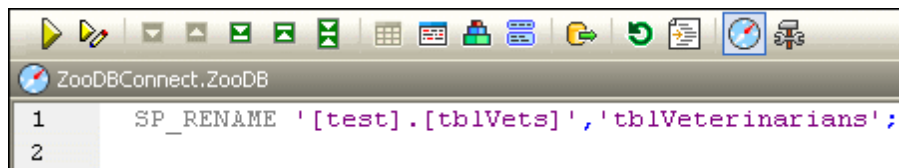
Note that the script contains only SQL for renaming the table itself; remember that there were also two columns with different names in this table. To generate a merge script that not only renames the table but also the two columns, you have to select the columns as well before you generate the merge script. Close the SQL Editor window without executing or saving the script, and get back to the Schema Comparison window.

4. Expand tblVeterinarians and select the table name as well as columns LastName, Telephone, and Email (hold down the **Ctrl** key to select multiple items).
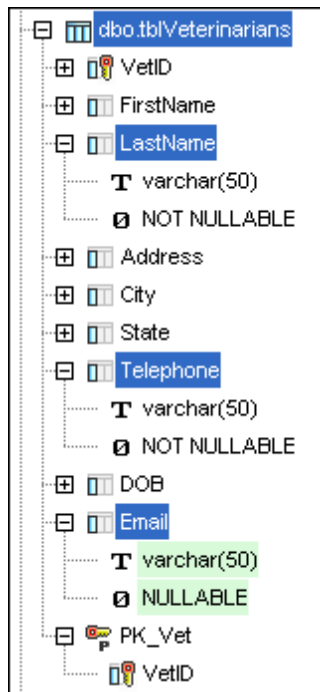
5. Select the menu option **Schema Comparison | Show merge script: Left to Right** and check the merge script in the SQL Editor.
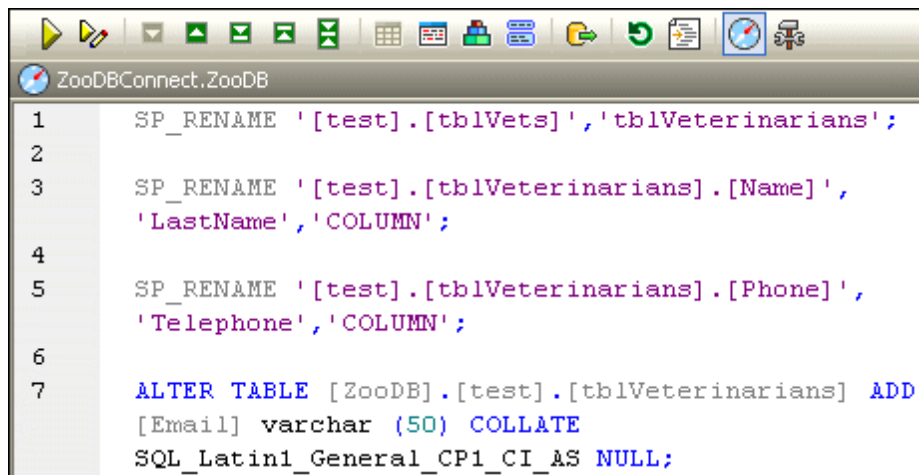
Now both table and columns will be renamed and the column Email will be added to schema test.

6. Make sure that the SQL Editor is still connected to ZooDB and click **Execute** .

7. In the Online Browser, click the **Refresh** button in the toolbar or right-click ZooDB and select **Refresh** from the context menu. Note that dbo.tblVeterinarians now appears unmapped in the Schema Comparison window.

8. Double-click the title bar of the right comparison component and select the renamed table

tblVeterinarians in the **Select Database Objects for Comparison** dialog box.

9. Click the **Compare** ▣ button in the comparison window toolbar. No differences now exist between dbo.tblVeterinarians and test.tblVeterinarians.

10. Expand test.tblZookeepers, right-click column Email and select **Show merge script: Right to Left** from the context menu.

```
1       ALTER TABLE [ZooDB].[dbo].[tblZookeepers] ADD [Email]
        varchar (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL;
```

11. Execute the merge script, refresh the database, and run a comparison to see the changes in the Schema Comparison window. No differences now exist between dbo.tblZookeepers and test.tblZookeepers.

4.2.12   Comparing Database Data

It may happen that you have to restore data from a backup database, or that you migrate your database tables to a different database provider. DatabaseSpy provides a database data comparison and merging feature that can assist you in these tasks.

Objective

In this section of the tutorial, you will learn how to compare database data and show the comparison results. Specifically, you will learn how to do the following:

- Set up a data comparison and select tables for comparison
- Start a comparison
- Show the results in the Comparison Result window
- Change the automatic mapping and remove tables from comparison components
- Merge database data

Commands used in this section

Add a New Data Source...:This command is located in the context menu that opens when you right-click the Data Sources folder in the Project window. Click this command to add an additional data source to your project.

🖳 **Save Project**: This command is located in the toolbar of the Project window; alternatively, you can also choose the menu option **File | Save Project As...** or press **Ctrl+S**. Click this command to save an existing project under its project name or to specify a name and path for a new project, respectively.

🗐 **Data Comparison**: This command is located in the Standard toolbar; alternatively, you can also choose the menu option **File | New | Data Comparison**. Click this command to open a new Data Comparison window and select tables for comparison. Note that at least one active connection to a data source is required to open a new data comparison.

▣ **Start Comparison**: This command is located in the toolbar of the Data Comparison window; alternatively, you can also choose the menu option **Data Comparison | Compare tables** or choose **Compare tables** from the context menu that opens when you right-click the title bar of a component. Click this command to run a comparison of all mapped tables in the comparison components.

▦ **Show selected results**: This command is located in the context menu that opens when you right-click a table in a component of a Data Comparison window; alternatively, you

can also select a table in a component and click the **Show result** icon in the toolbar of the Data Comparison Result window. Click this command to display the detailed comparison results for the selected table.

Show/Hide all equal rows: This command is located in the context menu that opens when you right-click into the Data Comparison Result window; alternatively, you can also use the **Show/Hide all equal rows** toggle button in the toolbar of the Data Comparison window. Deactivate this option to hide all rows that are equal in both of the compared tables.

Show/Hide rows that are only on the left: This command is located as a toggle button in the toolbar of the Data Comparison Result window; alternatively, you can also choose the this option from the context menu that opens when you right-click into the Data Comparison Result window. Deactivate this option to hide rows that exist only in the table that is on the left side of the comparison.

Show/Hide columns which don't have any differences: This command is located as a toggle button in the toolbar of the Data Comparison Result window. Deactivate this option to hide all columns that do not contain differences.

Find: This command is located as a toolbar button in the Data Comparison Result window. Click this command to open the **Find** dialog box and enter the string that you search for in the comparison result grid.

Merge data from right to left: This command is located as a toolbar button in the Data Comparison Result window. Click this command to copy data from the table in the right comparison component to the mapped table in the left component.

Execute: This command is located in the toolbar of an SQL Editor window and in the **SQL Editor** menu. Click this command to execute the SQL in the SQL Editor window.

Refresh: This command is available in the Online Browser as a button, or as an option in the context menu that opens when you right-click a database. Use this command to refresh the database and display new or changed database objects in the Online Browser.

SQL and Data | Retrieve data | All rows: This command is located in the context menu that opens when you right-click a table or column in a component of the Data Comparison window. Click this command to have all rows of data displayed in a Result tab.

SQL and Data | Edit Data: This command is located in the context menu that opens when you right-click a table or column in a component of the Data Comparison window. Choose this command to generate an SQL SELECT statement that is automatically executed in the SQL Editor and switches the Result window into the Editing mode. This command works for all supported database kinds for both ADO and ODBC connections.

Delete row: This command is available as a button in the Result window toolbar. Click this button to delete the row in which the cursor is currently located. Alternatively, you can also right-click a cell in the row to be deleted and choose **Delete row** from the context menu. The **Delete row** command is only available if the Result window is in the Editing mode.
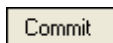
Commit : This command is available as a button in the Result window toolbar. Click this button

to commit the changes you have made in the result grid to the database. The **Commit** button is only visible if the Result window is in the Editing mode.

Setting Up a Database Data Comparison

The tutorial folder of you DatabaseSpy installation contains a Microsoft Access 2003 database which serves as a backup database for the tutorial "zoo" database. In this step of the tutorial, you will add this database to your tutorial project and select the tables in both databases for comparison.

To add the ZooDB-Backup data source to your project:

1. In the Project window, right-click the Data Sources folder and select **Add a New Data Source...** from the context menu.
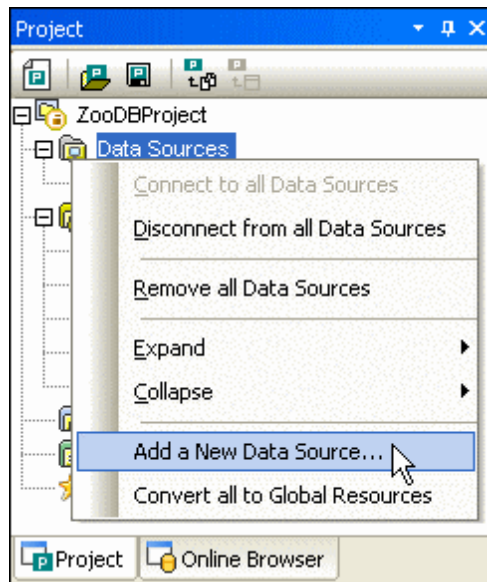
2. In the **Add a Data Source** dialog box, choose the `Microsoft Access ( ADO )` option button and click **Next**.

3. On the **Connect to MS Access** page, click the **Browse** button and navigate to the tutorial folder (\Altova\DatabaseSpy2017\DatabaseSpyExamples\Tutorial sub-folder of the (My) Documents folder).

4. Select the ZooDB.mdb database and click **Next**.

5. In the **Set A Data Source Name** dialog box, enter "ZooDB-Backup" and click **OK**. The data source is added to the project and a connection to it is automatically established.

6. Click the **Save Project** 🖬 button in the Project window toolbar or press **Ctrl+S**.

Opening a Data Comparison window

DatabaseSpy provides a Data Comparison window which is used to select and map tables, start a comparison, and merge data of the compared tables. Opening a Data Comparison window and selecting the tables (and columns) to be compared is usually done in a single step.

To select tables for comparison:

1. Click the **Data Comparison** 🔲 button in the Standard toolbar or select the menu

option **File | New | Data Comparison**. A new Data Comparison window is opened and
the **Select Database Objects for Comparison** dialog box appears, where the first data
source connection, ZooDBConnect, is preselected in the `Data Source` drop-down list,
and the **Left Side** button is activated by default.

2.   Expand the tree structure in the selection dialog box until the tables are displayed and
     activate the `User Tables` check box. This selects all tables in the User Tables folder.
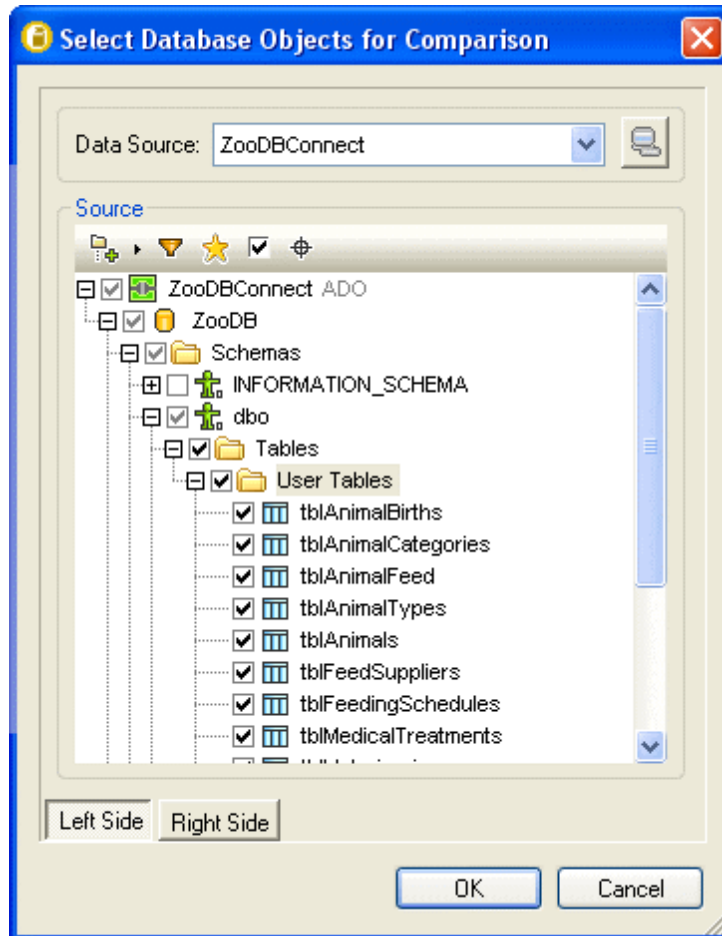
3.   Click the **Right Side** button and select "ZooDB-Backup" from the `Data Source` drop-
     down list.
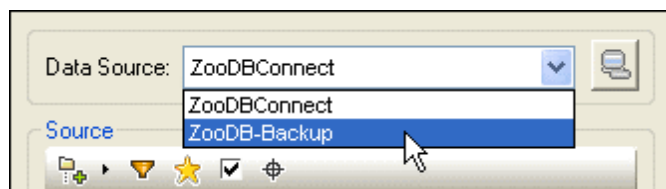
4.   Expand the tree structure in the selection dialog box until the tables are displayed and
     activate the `User Tables` check box. This selects all tables in the User Tables folder.
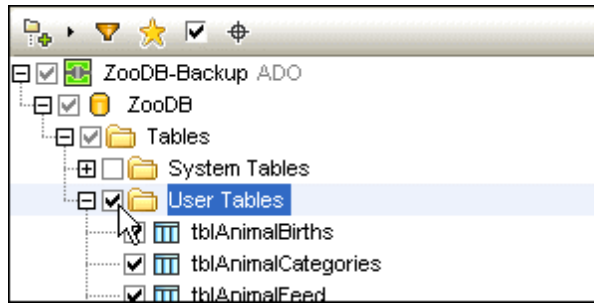
5.   Click **OK**. The tables appear in two separate components of the Data Comparison
     window, and matching tables and columns are mapped automatically.

6.   Select the menu option **Data Comparison | Collapse tables** or select this command
     from the context menu that opens when you right-click the header of either component in
     the Data Comparison window.
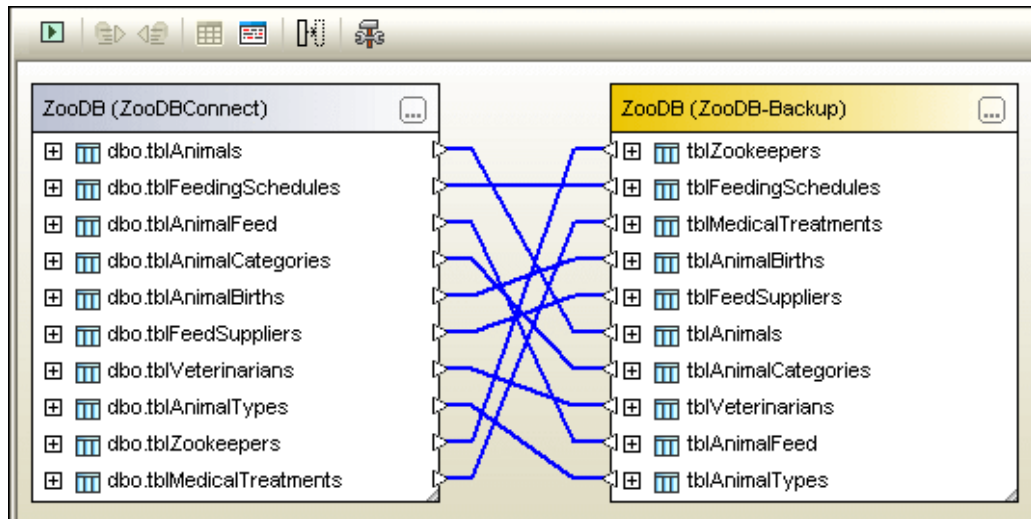
7.   Optionally, if the sort order is different in the left and right component, select the menu
     option **Data Comparison | Sort tables | Ascending** or select this command from the
     context menu that opens when you right-click the header of either component in the Data
     Comparison window.

Starting a Comparison

You can start a comparison from the **Data Comparison** menu or by clicking the **Start
Comparison** [▶] button in the toolbar of the Data Comparison window. This way, you run a
comparison for all mapped tables in the database comparison components. DatabaseSpy
displays the row count for the compared tables and shows comparison result icons that indicate
whether a pair of mapped tables is equal or contains differences. Place your mouse cursor over a
comparison result icon to display a brief summary of the comparison result.
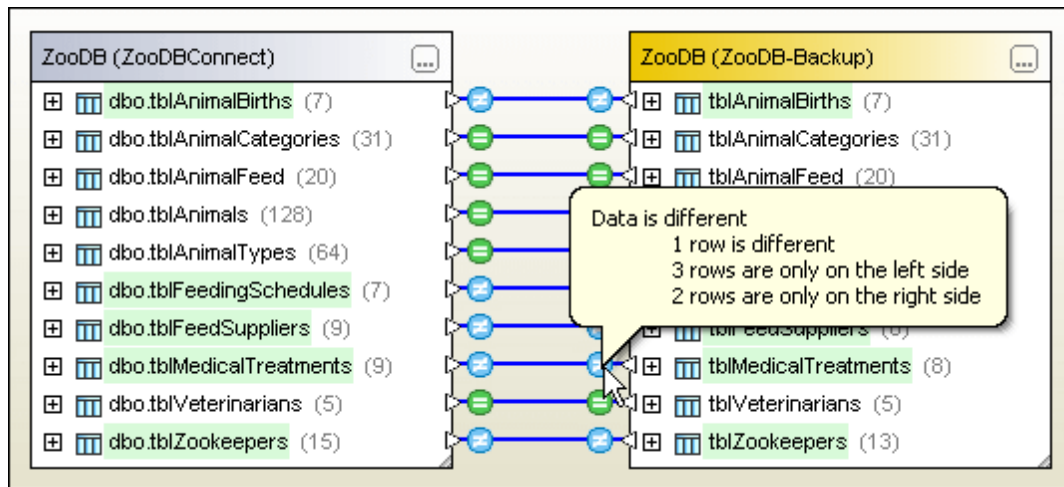
To compare all mapped tables:

- Select the menu option **Data Comparison | Compare tables**.

- Click the **Start Comparison** ▶ button in the toolbar of the Data Comparison window.

- Right-click the title bar in either comparison component and choose **Compare tables** from the context menu.

To show a brief comparison result for tblMedicalTreatments:

- Place the mouse cursor over the comparison result icon 🔵 of table tblMedicalTreatments. A balloon help appears with an overview of the comparison result (*see screenshot above*).

Results in the Message window

The Message window below the Data Comparison window logs for which tables a comparison has been started and displays brief summaries of the individual comparison results. The message window also provides hyperlinks to the tables in the Data Comparison windows and to the detailed results of a specific comparison in the Comparison result window.
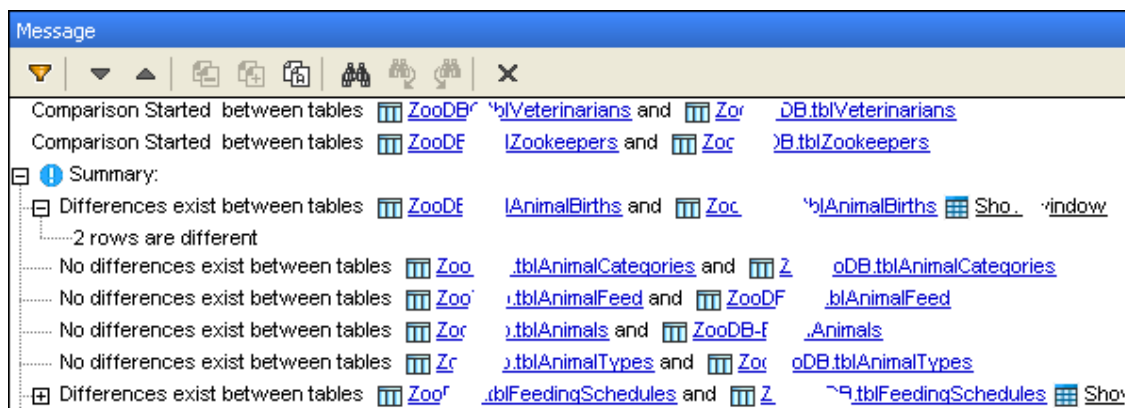
Showing the Results

In the previous step of the tutorial, you have learned how to start a comparison and display an overview of a comparison result directly in the Data Comparison window or in the Message

window. Now you will have a more detailed look on the differences between the two versions of table tblAnimalBirths that you have compared earlier in this tutorial.

If you display the comparison result overview (either by placing the mouse cursor over the comparison result icon of tblAnimalBirths in the Data Comparison window, or by expanding the relevant row in the Summary section of the Message window), you learn that
  • two rows are different and
  • three rows are only on the left side
of your comparison of tblAnimalBirths. You can inspect this comparison result more closely in the Data Comparison Result window.

	BirthID	BirthID	BirthDate	BirthDate	Mother	Mother	Father	Father	Vet...	Vet...	Num...	Num...
1	1	1	2005-01-16...	2005-01-16...	5055	5056	5056	5055	4	4	3	3
2	2	2	2006-03-23...	2006-03-23...	5061	5062	5063	5063	2	2	1	1
3	3		2005-11-07...		5055		5054		6		4	
4	4		2006-02-17...		5048		5047		6		2	
5	5	5	2006-10-04...	2006-10-04...	5127	5127	5002	5002	4	4	5	5
6	6	6	2006-05-24...	2006-05-24...	5026	5026	5025	5025	4	4	1	1
7	7	7	2005-09-14...	2005-09-14...	5051	5051	5050	5050	1	1	1	1
8	8		2006-05-23...		5036		5035		6		6	
9	9	9	2005-12-03...	2005-12-03...	5068	5068	5069	5069	2	2	1	1
10	10	10	2006-02-05...	2006-02-05...	5073	5073	5072	5072	4	4	1	1

To show the detailed comparison result for tblAnimalBirths:

Do one of the following:

  • Click the comparison result icon [icon] next to tblAnimalBirths.
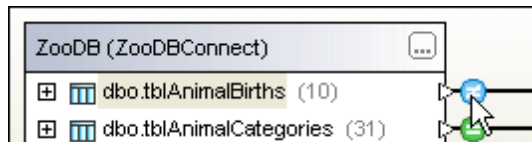
  • Right-click tblAnimalBirths and select **Show selected results** from the context menu.
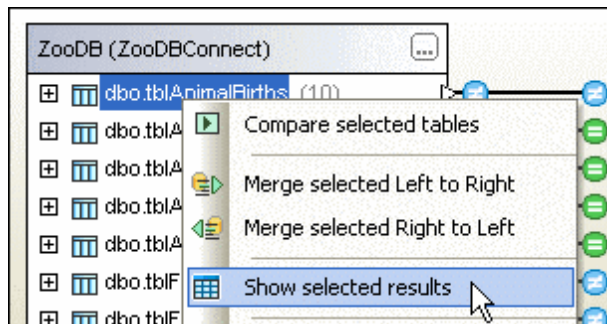
  • In the Result window, click the `Show result window` hyperlink for tblAnimalBirths.

Showing all results

You can also open all comparison results in the Data Comparison Result window with a single click. At the bottom of the Message window, DatabaseSpy provides two hyperlinks, `Show only results with differences` and `Show all results`, which open the results in separate tabs of the Data Comparison Result window.

Configuring the Result view

The Data Comparison Result window provides several buttons which allow you to show or hide rows with a certain comparison result. For example, you can hide all rows that are equal on the left and right side of the comparison or show only non-equal rows but hide rows that exist only on the left side of the comparison.

To hide all equal rows in the Data Comparison Result window:

- Right-click into the Data Comparison Result window and deactivate the context menu option **Show/Hide all equal rows**.

Only non-equal rows are displayed in the Data Comparison Result window.

In the next step, you will also hide all rows that occur only on the left side of the comparison and display only those rows that exist in both tables and are different on the left and right side.

To hide all rows that exist only on the left side:

- In the Data Comparison Result window, deactivate the **Show/Hide rows that are only on the left** button in the toolbar.

Now only rows are displayed that exist in both tables and are different on the left and on the right side.

Since a lot of columns are now displayed where no differences occur, you can also hide these columns and restrict the display to exact those columns and rows which are different in the two tables.

To hide all columns where no differences occur:

- In the Data Comparison Result window, deactivate the **Show/Hide columns which don't have any differences** button in the toolbar.

Only two rows and two columns are now displayed in the Data Comparison Result window – exactly those data sets where the differences occur.

Unmapping and Removing Tables

You have seen in the previous step of the tutorial that some of the tables are equal in both databases. When examining the differences, you can unmap these tables and remove them from the comparison components.

To unmap and remove equal tables:

1. Right-click tblAnimalCategories and choose **Unmap selected** from the context menu.
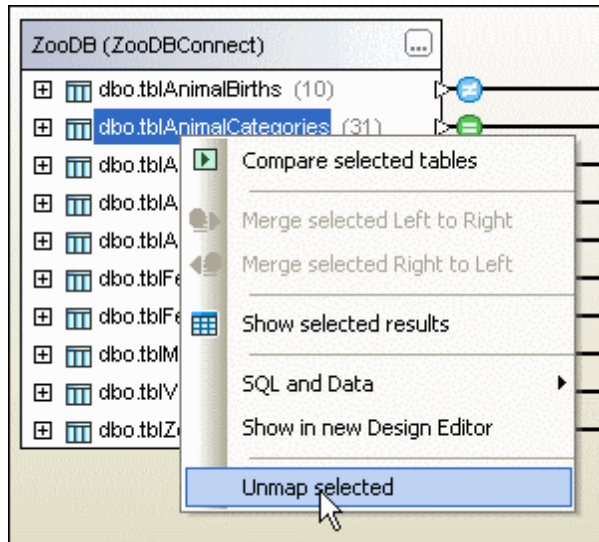
2. Click the connection line between the two versions of tblAnimalFeed and hit the **Del** key.
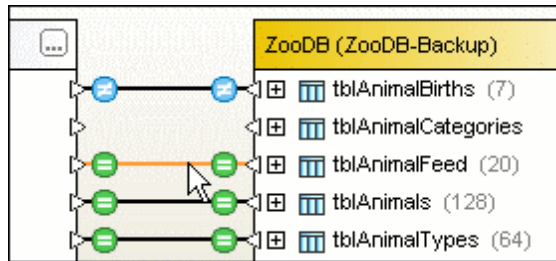
3. Right-click the connector between the two versions of tblAnimals and choose **Unmap selected** from the context menu.
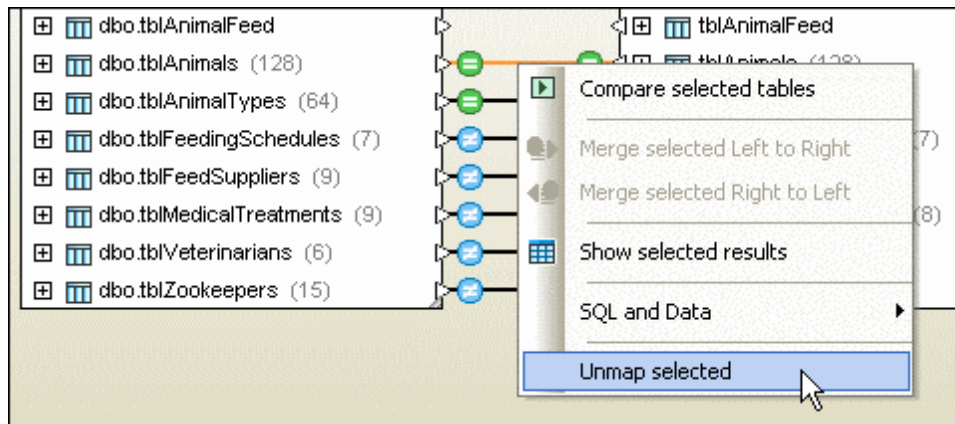
4. Click tblAnimalTypes in the left component and hit the **Del** key. This deletes the mapping and removes tblAnimalTypes from the left component.
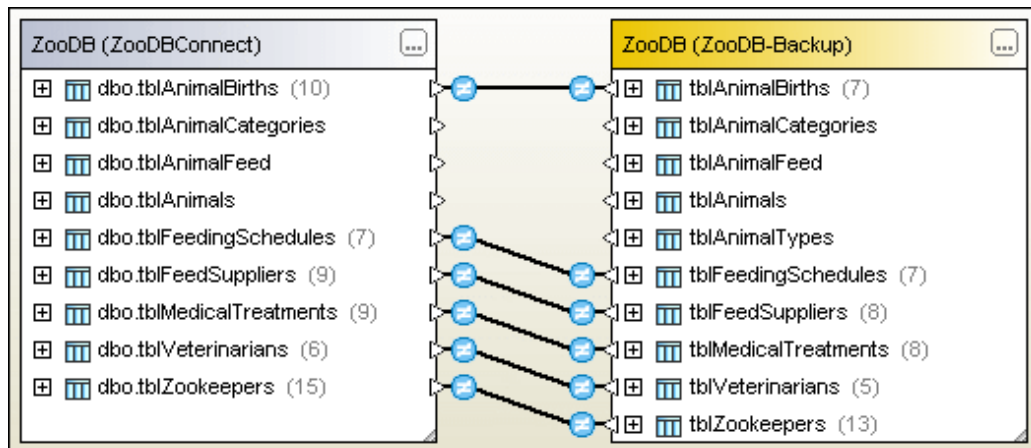
5. Double-click the header of ZooDB-Backup in the right component and deselect tblAnimalCategories, tblAnimalFeed, tblAnimalTypes, and tblAnimals.
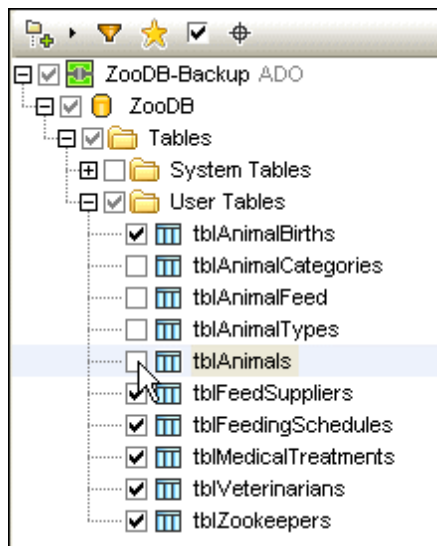
6. Select tblAnimalCategories, tblAnimalFeed, and tblAnimals in the left component and hit the **Del** key.

Merging Tables

Let's pretend that one of our recently hired zookeepers, Bruno Katz, has a degree on veterinary medicine and now joins the team of veterinarians of our tutorial zoo. In this step of the tutorial, you will transfer Bruno's record from tblZookeepers to tblVeterinarians using DatabaseSpy's merging functionality.

To transfer a record set to another table:

1. Click the **Data Comparison** button in the Standard toolbar to open a new Data Comparison window.

2. In the **Select Database Objects for Comparison** dialog box, choose "ZooDBConnect" in the `Data Source` drop-down list, and activate the `tblVeterinarians` and `tblZookeepers` check boxes in the Source group box.

3. Click **OK**. Both tables are added to the left comparison component; the right component

remains empty with no data source assigned.

4.   Click dbo.tblZookeepers and, keeping the mouse button pressed, drag the table into the right component. Note that both components now have the same data source assigned.

5.   Click the triangle next to dbo.tblVeterinarians in the left component and, keeping the mouse button pressed, draw a line to the triangle next to dbo.tblZookeepers in the right component. Release the mouse button when the shape of the cursor changes.
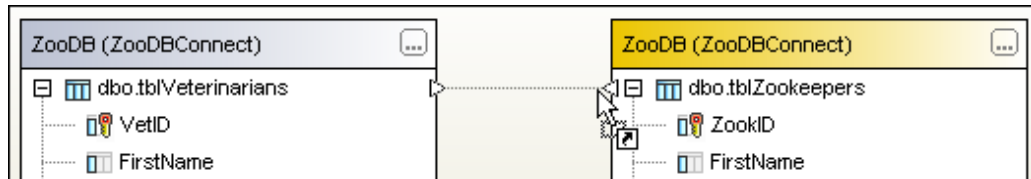
All columns except VetID and ZookID are also mapped automatically.
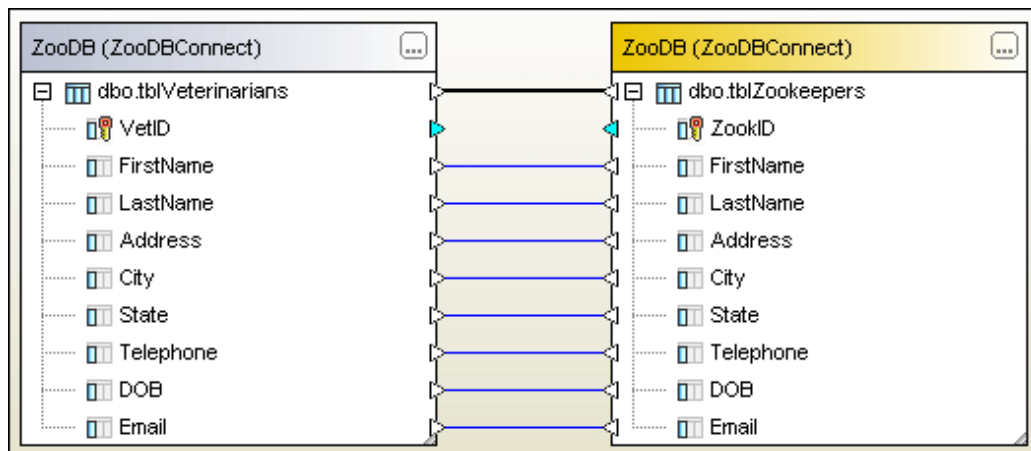
6.   In the same way, map column VetID with column ZookID.

7.   Click the **Start Comparison** [▶] button in the toolbar of the Data Comparison window.

8.   Select the menu option **Data Comparison | Show results**.

9.   In Data Comparison Result window, click the **Find** [⊞] button in the toolbar, enter "Katz" in the `Find what` field, and click **Find next**.

10. Close the **Find** dialog box and click the **Merge data from right to left** [◀≡] button in the toolbar. The **Merge data (Right to Left)** dialog box appears.
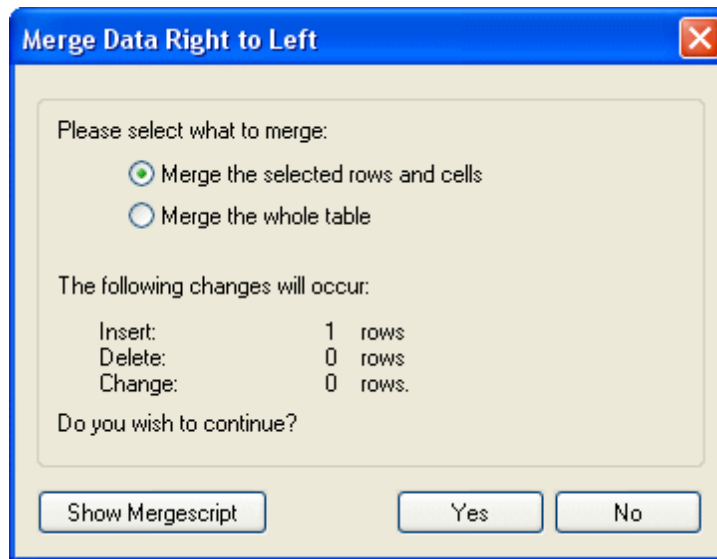
11. Click the **Show Merge script** button and edit the merge script as follows (delete the first column `[VetID]`, and the first value `15,`):

```
INSERT INTO [ZooDB].[dbo].[tblVeterinarians]([FirstName],
[LastName],[Address],[City],[State],[Telephone],[DOB])
VALUES('Bruno','Katz','87 Panther Place','Trenton','NJ','(515)
555-7667','1969-09-13 00:00:00.000');
```

12. Click the **Execute** button or press **F5**.

13. In the Online Browser, select the ZooDB database in the ZooDBConnect data source and click the **Refresh** button in the Online Browser toolbar.

14. In the Data Comparison window, right-click tblVeterinarians and select **SQL and Data | Retrieve data | All rows** from the context menu. Bruno Katz has been added to the list of veterinarians.

	VetID	FirstName	LastName	Address	City	State	Telephone	D
5	5	Robin	Fowler	183 Lionheart Drive	Albuquerque	NM	(515)555-2810	{/
6	6	Jacob	Fisher	87 Bearpaw Circle	Milwaukee	WI	(515)555-7209	{/
7	7	Bruno	Katz	87 Panther Place	Trenton	NJ	(515)555-7667	1

15. In the Data Comparison window, right-click tblZookeepers and select **SQL and Data | Edit data** from the context menu.

16. Click **OK** in the message box that appears and select Bruno Katz in the result grid.

17. Click **Delete row** in the toolbar and click the button to save the change to the database.

Chapter 5

DatabaseSpy Projects

5   DatabaseSpy Projects

Altova web site: 🌐 database projects

A DatabaseSpy Project combines everything that you need when working with your databases in a single project file. It includes data source connections, SQL files, database design files, and comparison files for data and schemas. You can also define favorites for your most frequently used files or database objects.

The content of your project is displayed in the Project window.

Please note: As of version 2008, projects are stored in a different file format. Therefore, as of DatabaseSpy 2008 projects that have been saved **cannot be opened in earlier releases**.

Managing projects

Every time you start DatabaseSpy (and provided that you have not changed the default setting of the `Open last project on startup` check box in the General options), an empty project is displayed in the Project window. You must at least add a data source to it and save it in your file system.
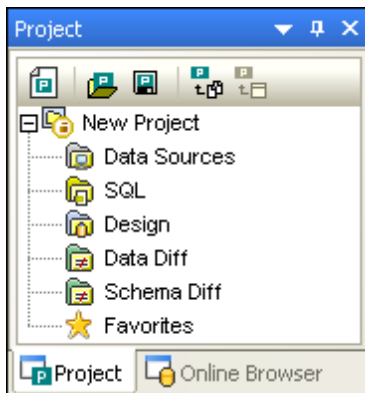
If DatabaseSpy starts with the last project, you can create a new project by proceeding as follows:

- In the Project window, click the **Create a new project** 🔲 button or select the menu option **File | New | Project**.

  If there is an unsaved project already open in the Project window, you are prompted to save this project. After you have decided what to do with the unsaved project, a new, empty project appears in the Project window.

If you want to open an already existing DatabaseSpy Project do one of the following:

- In the Project window, click the **Open a project** 🔲 button or select the menu option **File | Open | Open Project** and choose a project file from the **Open** dialog box.
- Select the menu option **File | Open | Open File...**, select "Altova DatabaseSpy Project (*.qprj)" from the `Files of type` drop-down list, and choose a project file from the **Open** dialog box.

If there is an unsaved project already open in the Project window, you are prompted to save this project.

A project is displayed with its file name in DatabaseSpy, so if you save a project make sure that you choose a descriptive name for the *.qprj file. You can rename a project only by changing its file name.

To save a project:

- In the Project window, click the **Save Project** button or select the menu option **File | Save | Save Project**.

- To save the project under a new name and change its title:

  1. Select the menu option **File | Save | Save Project As...**.

  2. Select the folder where you want to save the project file and enter a name for the file.

  3. Click **Save**.

To close a project:

- Open an existing or a new project.

To delete a project:

- Delete the respective *.qprj file from your hard disk.

5.1 Altova Global Resources

Altova Global Resources is a collection of aliases for file, folder, and database resources. Each alias can have multiple configurations, and each configuration maps to a single resource (*see screenshot below*). Therefore, when a global resource is used as an input, the global resource can be switched among its configurations. This is done easily via controls in the GUI that let you select the active configuration. For example, an SQL file or data source connection can be changed with a couple of mouse-clicks, and a query can be executed without having to change the data source or SQL statement.
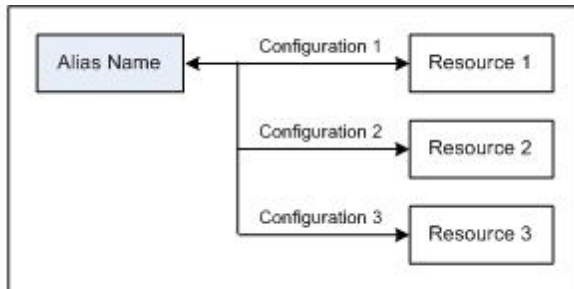
Using Altova Global Resources involves two processes:

- Defining Global Resources: Resources are defined and the definitions are stored in an XML file. These resources can be shared across multiple Altova applications.
- Using Global Resources: Within DatabaseSpy, files can be located via a global resource instead of via a file path. The advantage is that the resource can be switched by changing the active configuration in DatabaseSpy.

Global resources in other Altova products

Currently, global resources can be defined and used in the following individual Altova products: XMLSpy, StyleVision, MapForce, Authentic Desktop, MobileTogether Designer, and DatabaseSpy.

5.1.1 Defining Global Resources

Altova Global Resources are defined in the Manage Global Resources dialog, which can be accessed in two ways:

- Click the menu command **Tools | Global Resources**.
- Click the **Manage Global Resources** icon in the Global Resources toolbar (*screenshot below*).

The Global Resources Definitions file

Information about global resources is stored in an XML file called the Global Resources Definitions file. This file is created when the first global resource is defined in the Manage Global Resources dialog (*screenshot below*) and saved.
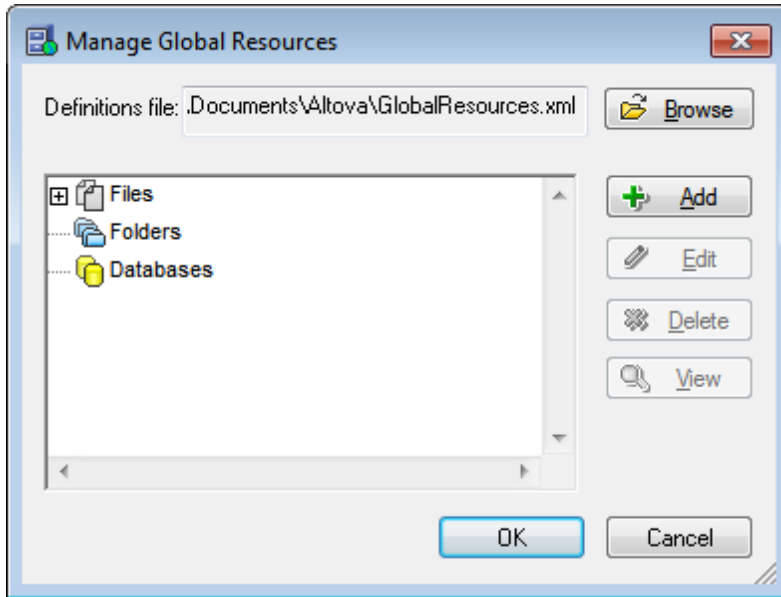
When you open the Manage Global Resources dialog for the first time, the default location and name of the Global Resources Definitions file is specified in the *Definitions File* text box (*see screenshot above*):

```
C:\Users\<username>\My Documents\Altova\GlobalResources.xml
```

This file is set as the default Global Resources Definitions file for all Altova applications. So a global resource can be saved from any Altova application to this file and will be immediately available to all other Altova applications as a global resource. To define and save a global resource to the Global Resources Definitions file, add the global resource in the Manage Global Resources dialog and click **OK** to save.

To select an already existing Global Resources Definitions file to be the active definitions file of a particular Altova application, browse for it via the **Browse** button of the *Definitions File* text box (*see screenshot above*).

Note: You can name the Global Resources Definitions file anything you like and save it to any location accessible to your Altova applications. All you need to do in each application, is specify this file as the Global Resources Definitions file for that application (in the *Definitions File* text box). The resources become global across Altova products when you use a single definitions file across all Altova products.

Note: You can also create multiple Global Resources Definitions files. However, only one of these can be active at any time in a given Altova application, and only the definitions contained in this file will be available to the application. The availability of resources can therefore be restricted or made to overlap across products as required.

Managing global resources: adding, editing, deleting, saving

In the Manage Global Resources dialog (*screenshot above*), you can add a global resource to the selected Global Resources Definitions file, or edit or delete a selected global resource. The Global Resources Definitions file organizes the global resources you add into groups: of files, folders, and databases (*see screenshot above*).

To *add a global resource*, click the **Add** button and define the global resource in the appropriate **Global Resource** dialog that pops up (*see the descriptions of files, folders, and databases in the sub-sections of this section*). After you define a global resource and save it (by clicking **OK** in the Manage Global Resources dialog), the global resource is added to the library of global definitions in the selected Global Resources Definitions file. The global resource will be identified by an alias.

To *edit a global resource*, select it and click **Edit**. This pops up the relevant **Global Resource** dialog, in which you can make the necessary changes (*see the descriptions of files, folders, and databases in the sub-sections of this section*).

To *delete a global resource*, select it and click **Delete**. If the global resource you delete has been added as a data source to the project that is currently open in DatabaseSpy, a message box appears and you can decide whether the data source should be deleted also in the project or whether it should be converted into a local data source.

After you finish adding, editing, or deleting, make sure to click **OK** in the Manage Global Resources dialog to *save your modifications* to the Global Resources Definitions file.

Relating global resources to alias names via configurations

Defining a global resource involves mapping an alias name to a resource (file, folder, or database). A single alias name can be mapped to multiple resources. Each mapping is called a configuration. A single alias name can therefore be associated with several resources via different configurations (*screenshot below*).
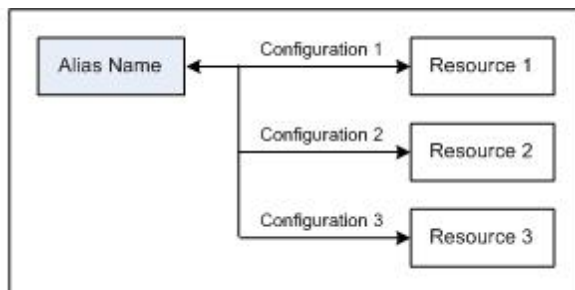
In an Altova application, you can then assign aliases instead of files. For each alias you can switch between the resources mapped to that alias simply by changing the application's active Global Resource configuration (active configuration). For example, in Altova's XMLSpy application, if you wish to run an XSLT transformation on the XML document `MyXML.xml`, you can assign the alias `MyXSLT` to it as the global resource to be used for XSLT transformations. In XMLSpy you can then change the active configuration to use different XSLT files. If `Configuration-1` maps `First.xslt` to `MyXSLT` and `Configuration-1` is selected as the active configuration, then `First.xslt` will be used for the transformation. In this way multiple configurations can be used to

access multiple resources via a single alias. This mechanism can be useful when testing and comparing resources. Furthermore, since global resources can be used across Altova products, resources can be tested and compared across multiple Altova products as well.

Files

The Global Resource dialog for Files (*screenshot below*) is accessed via the **Add | Files** command in the Manage Global Resources dialog. In this dialog, you can define configurations of the alias that is named in the *Resource Alias* text box. After specifying the properties of the configurations as explained below, save the alias definition by clicking **OK**.

After saving an alias definition, you can add another alias by repeating the steps given above (starting with the **Add | Files** command in the Manage Global Resources dialog).

Global Resource dialog

An alias is defined in the Global Resource dialog (*screenshot below*).
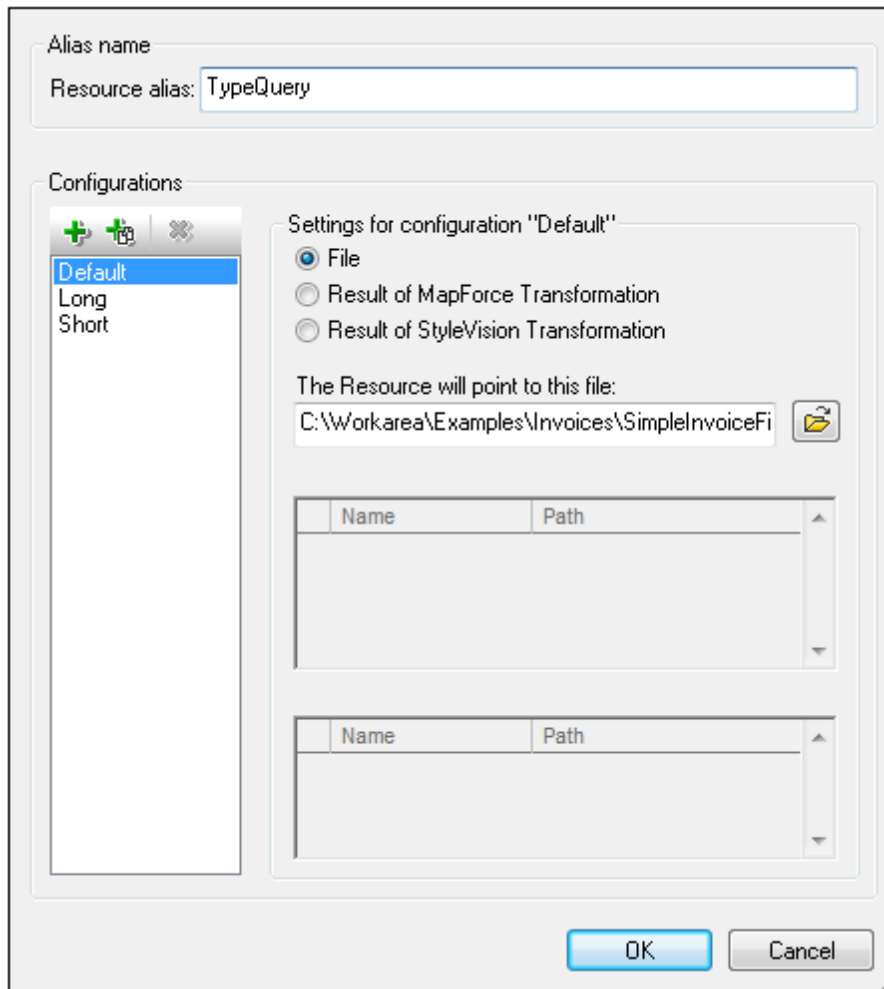
Global Resource dialog icons

![icon] *Add Configuration:* Pops up the Add Configuration dialog in which you enter the name of the configuration to be added.

![icon] *Add Configuration as Copy:* Pops up the Add Configuration dialog in which you can enter the name of the configuration to be created as a copy of the selected configuration.

![icon] *Delete:* Deletes the selected configuration.

![icon] *Open:* Browse for the file to be created as the global resource.

Defining the alias

Define the alias (its name and configurations) as follows:

1. *Give the alias a name:* Enter the alias name in the *Resource Alias* text box.
2. *Add configurations:* The Configurations pane will have, by default, a configuration named `Default` (*see screenshot above*), which cannot be deleted or renamed. You can add as many additional configurations as you like by: (i) clicking the **Add Configuration** or **Add Configuration as Copy** icons, and (ii) giving the configuration a name in the dialog that pops up. Each added configuration will be shown in the Configurations list. In the screenshot above, two additional configurations, named `Long` and `Short`, have been added to the Configurations list. The Add Configuration as Copy command enables you to copy the selected configuration and then modify it.
3. *Select a resource type for each configuration:* Select a configuration from the Configurations list, and, in the *Settings for Configuration* pane, specify a resource for the configuration: (i) File, (ii) Output of an Altova MapForce transformation, or (iii) Output of an Altova StyleVision transformation. Select the appropriate radio button. If a MapForce or StyleVision transformation option is selected, then a transformation is carried out by MapForce or StyleVision using, respectively, the `.mfd` or `.sps` file and the respective input file. The result of the transformation will be the resource.
4. *Select a file for the resource type:* If the resource is a directly selected file, browse for the file in the *Resource File Selection* text box. If the resource is the result of a transformation, in the *File Selection* text box, browse for the `.mfd` file (for MapForce transformations) or the `.sps` file (for StyleVision transformations). Where multiple inputs or outputs for the transformation are possible, a selection of the options will be presented. For example, the output options of a StyleVision transformation are displayed according to what edition of StyleVision is installed (*the screenshot below shows the outputs for Enterprise Edition*).
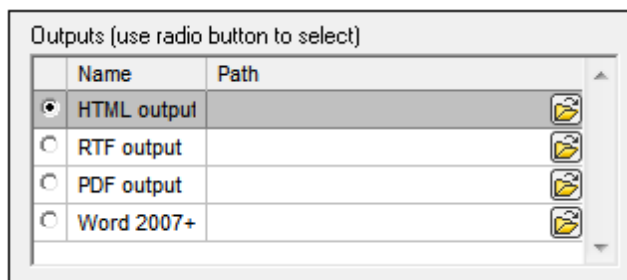
Select the radio button of the desired option (in the screenshot above, 'HTML output' is selected). If the resource is the result of a transformation, then the output can be saved

as a file or itself as a global resource. Click the [icon] icon and select, respectively, Global Resource (for saving the output as a global resource) or Browse (for saving the output as a file). If neither of these two saving options is selected, the transformation result will be loaded as a temporary file when the global resource is invoked.

5. *Define multiple configurations if required:* You can add more configurations and specify a resource for each. Do this by repeating Steps 3 and 4 above for each configuration. You can add a new configuration to the alias definition at any time.

6. *Save the alias definition:* Click **OK** to save the alias and all its configurations as a global resource. The global resource will be listed under Files in the Manage Global Resources dialog.

Result of MapForce transformation

Altova MapForce maps one or more (existing) input document schemas to one or more (new) output document schemas. This mapping, which is created by a MapForce user, is known as a MapForce Design (MFD). XML files, text files, databases, etc, that correspond to the input schema/s can be used as data sources. MapForce generates output data files that correspond to the output document schema. This output document is the *Result of MapForce Transformation* file that will become a global resource.

If you wish to set a MapForce-generated data file as a global resource, the following must be specified in the Global Resource dialog (*see screenshot below*):
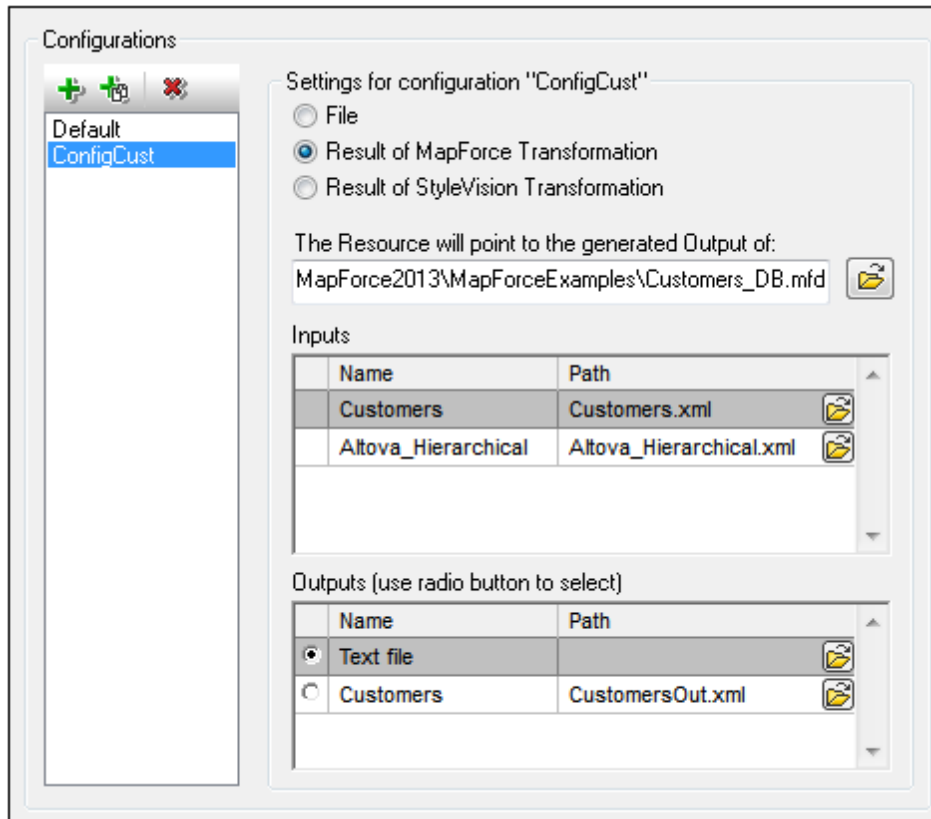
- **A `.mfd` (MapForce Design) file.** You must specify this file in the *Resource will point to generated output of* text box (*see screenshot above*).
- **One or more input data files.** After the MFD file has been specified, it is analysed and, based on the input schema information in it, default data file/s are entered in the *Inputs* pane (*see screenshot above*). You can modify the default file selection for each input schema by specifying another file.
- **An output file.** If the MFD document has multiple output schemas, all these are listed in the *Outputs* pane (*see screenshot above*) and you must select one of them. If the output file location of an individual output schema is specified in the MFD document, then this file location is entered for that output schema in the *Outputs* pane. From the screenshot above we can see that the MFD document specfies that the `Customers` output schema has a default XML data file (`CustomersOut.xml`), while the `Text file` output schema does not have a file association in the MFD file. You can use the default file location in the Outputs pane or specify one yourself. The result of the MapForce transformation will be saved to the file location of the selected output schema. This is the file that will be used as the global resource

Note: The advantage of this option (Result of MapForce transformation) is that the transformation is carried out at the time the global resource is invoked. This means that the global resource will contain the most up-to-date data (from the input file/s).

Note: Since MapForce is used to run the transformation, you must have Altova MapForce installed for this functionality to work.

Result of StyleVision transformation

Altova StyleVision is used to create StyleVision Power Stylesheet (SPS) files. These SPS files generate XSLT stylesheets that are used to transform XML documents into output documents in various formats (HTML, PDF, RTF, Word 2007+, etc). If you select the option *Result of StyleVision Transformation*, the output document created by StyleVision will be the global resource associated with the selected configuration.

For the *StyleVision Transformation* option in the Global Resource dialog (*see screenshot below*), the following files must be specified.
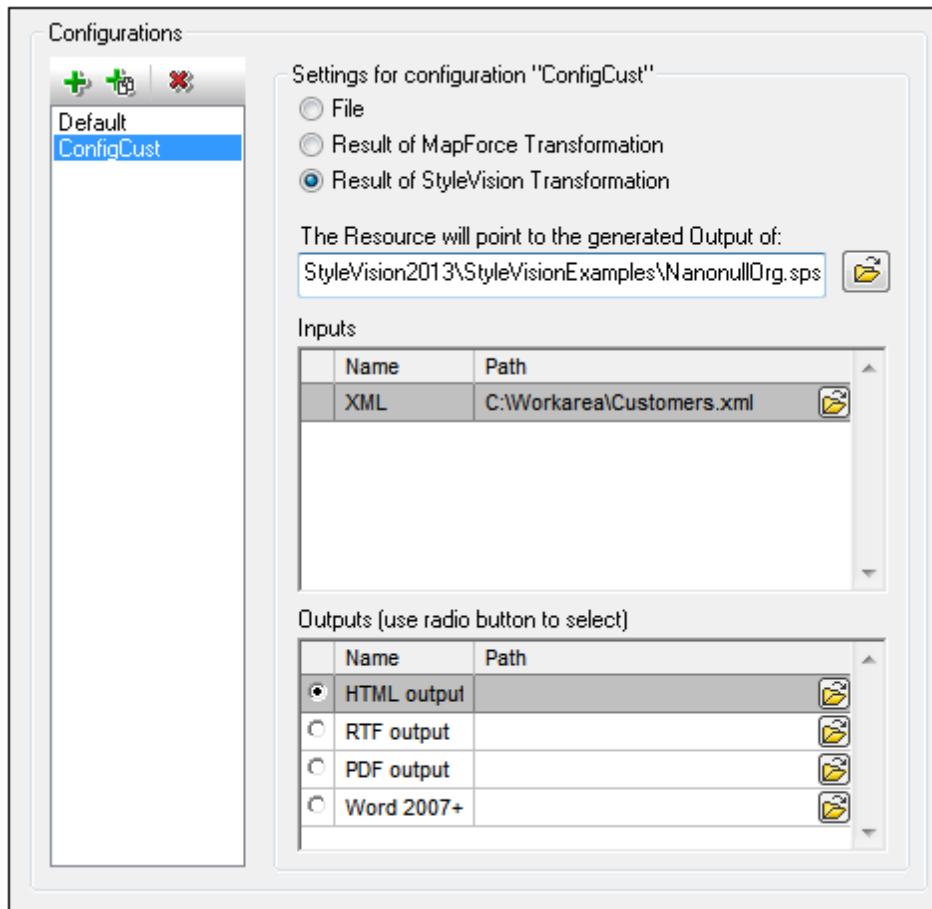
- *A* `.sps` *(SPS) file.* You must specify this file in the *Resource will point to generated output of* text box (*see screenshot above*).
- *Input file/s.* The input file might already be specified in the SPS file. If it is, it will appear automatically in the *Inputs* pane once the SPS file is selected. You can change this entry. If there is no entry, you must add one.
- *Output file/s.* Select the output format in the *Outputs* pane, and specify an output file location for that format.

Note: The advantage of this option (Result of StyleVision transformation) is that the transformation is carried out when the global resource is invoked. This means that the global resource will contain the most up-to-date data (from the input file/s).

Note: Since StyleVision is used to run the transformation, you must have Altova StyleVision installed for this functionality to work.

Folders

In the Global Resource dialog for Folders (*screenshot below*), add a folder resource as described below.
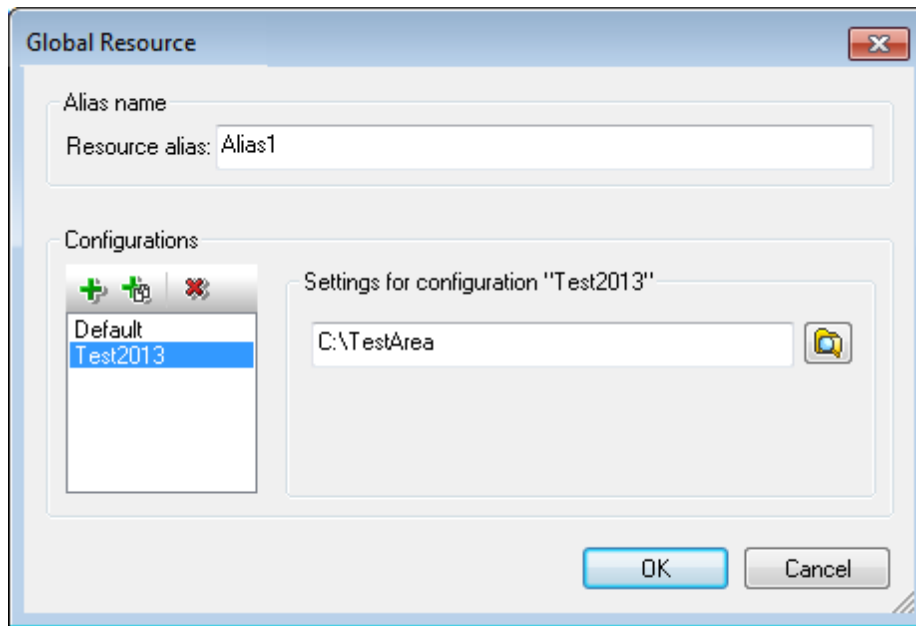
Global Resource dialog icons

  Add Configuration: Pops up the Add Configuration dialog in which you enter the name of the configuration to be added.

  Add Configuration as Copy: Pops up the Add Configuration dialog in which you can enter the name of the configuration to be created as a copy of the selected configuration.

  Delete: Deletes the selected configuration.

  Open: Browse for the folder to be created as the global resource.

Defining the alias

Define the alias (its name and configurations) as follows:

1. *Give the alias a name:* Enter the alias name in the *Resource Alias* text box.
2. *Add configurations:* The Configurations pane will have a configuration named Default (*see screenshot above*). This Default configuration cannot be deleted nor have its name changed. You can enter as many additional configurations for the selected alias as you like. Add a configuration by clicking the **Add Configuration** or **Add Configuration as Copy** icons. In the dialog which pops up, enter the configuration name. Click **OK**. The new configuration will be listed in the Configurations pane. Repeat for as many configurations as you want.
3. *Select a folder as the resource of a configuration:* Select one of the configurations in the Configurations pane and browse for the folder you wish to create as a global resource.
4. *Define multiple configurations if required:* Specify a folder resource for each configuration you have created (that is, repeat Step 3 above for the various configurations you have created). You can add a new configuration to the alias definition at any time.
5. *Save the alias definition:* Click **OK** in the Global Resource dialog to save the alias and all its configurations as a global resource. The global resource will be listed under Folders in

the Manage Global Resources dialog.

Databases

In the Global Resource dialog for Databases (*screenshot below*), you can add a database resource as follows:
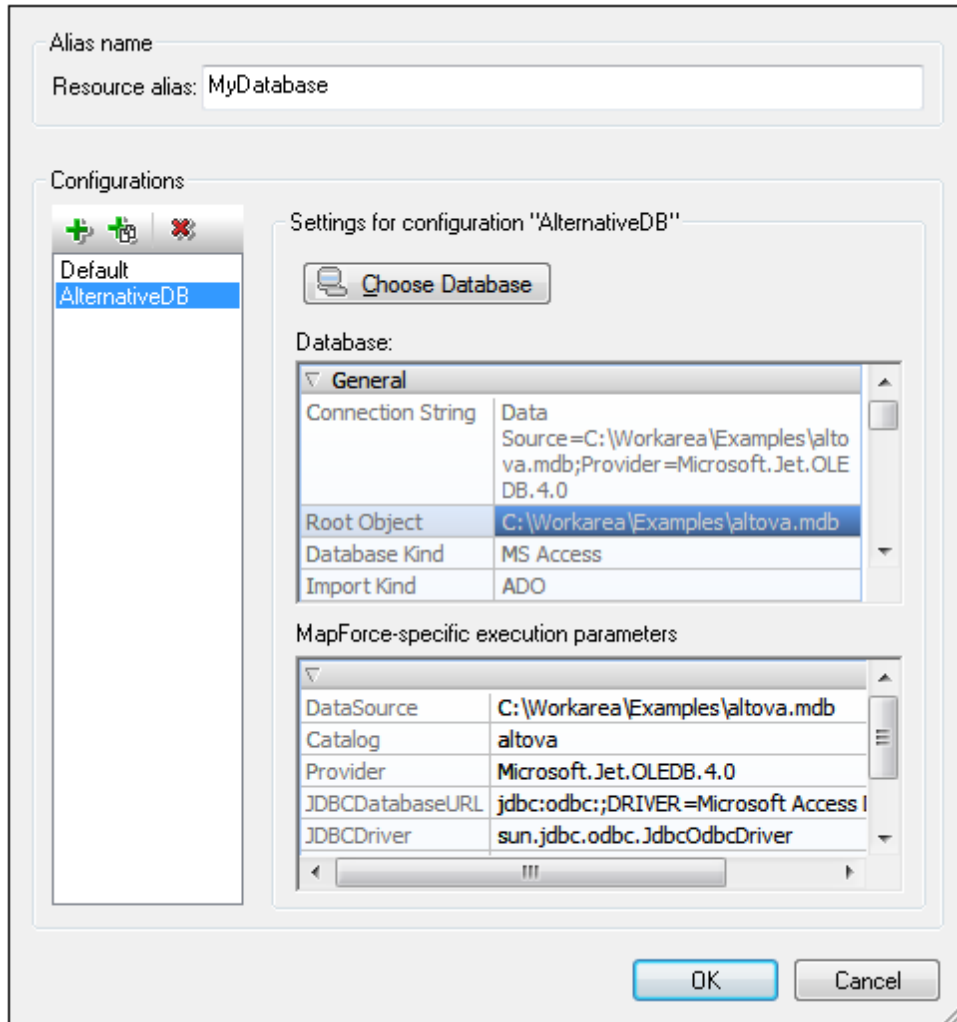
Global Resource dialog icons

➕	*Add Configuration:* Pops up the Add Configuration dialog in which you enter the name of the configuration to be added.
➕	*Add Configuration as Copy:* Pops up the Add Configuration dialog in which you can enter the name of the configuration to be created as a copy of the selected configuration.
✖	*Delete:* Deletes the selected configuration.

Defining the alias

Define the alias (its name and configurations) as follows:

1. *Give the alias a name:* Enter the alias name in the *Resource Alias* text box.
2. *Add configurations:* The Configurations pane will have a configuration named Default (*see screenshot above*). This Default configuration cannot be deleted nor have its name changed. You can enter as many additional configurations for the selected alias as you like. Add a configuration by clicking the **Add Configuration** or **Add Configuration as Copy** icons. In the dialog which pops up, enter the configuration name. Click **OK**. The new configuration will be listed in the Configurations pane. Repeat for as many configurations as you want.
3. *Start selection of a database as the resource of a configuration:* Select one of the configurations in the Configurations pane and click the **Choose Database** icon. This pops up the Create Global Resources Connection dialog.
4. *Connect to the database:* Select whether you wish to create a connection to the database using the Connection Wizard, an ADO Connection, an ODBC Connection, or JDBC Connection. Complete the definition of the connection method as described in the section Connecting to a Database. You can use either the Connection Wizard, ADO Connections, or ODBC Connections.
5. *Select the root object:* If you connect to a database server where a root object can be selected, you will be prompted, in the Choose Root Object dialog (*screenshot below*), to select a root object on the server. Select the root object and click **Set Root Object**. The root object you select will be the root object that is loaded when this configuration is used.
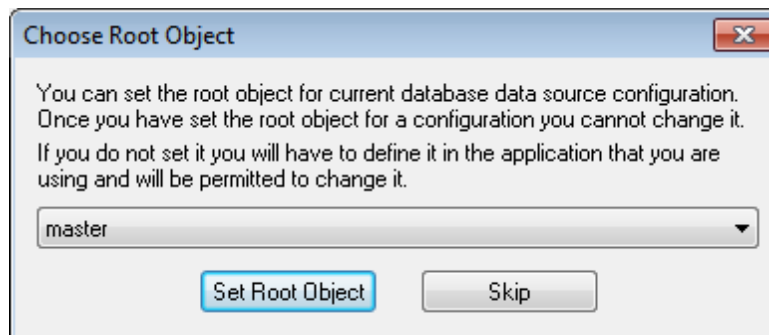
If you choose not to select a root object (by clicking the **Skip** button), then you can select the root object at the time the global resource is loaded.

6. *Define multiple configurations if required:* Specify a database resource for any other configuration you have created (that is, repeat Steps 3 to 5 above for the various configurations you have created). You can add a new configuration to the alias definition at any time.
7. *Save the alias definition:* Click **OK** in the Global Resource dialog to save the alias and all its configurations as a global resource. The global resource will be listed under databases in the Manage Global Resources dialog.

5.1.2   Using Global Resources

There are several types of global resources (file-type, folder-type , and database-type). Some scenarios in which you can use global resources in DatabaseSpy are listed here: Global SQL Statements and Data Sources in DatabaseSpy Projects.

Selections that determine which resource is used

There are two application-wide selections that determine what global resources can be used and which global resources are actually used at any given time:

- *The active Global Resources XML File* is selected in the Global Resource dialog. The global-resource definitions that are present in the active Global Resources XML File are available in the application. Only the definitions in the active Global Resources XML File are available. The active Global Resources XML File can be changed at any time, and the global-resource definitions in the new active file will immediately replace those of the previously active file. The active Global Resources XML File therefore determines: (i) what global resources can be assigned, and (ii) what global resources are available for look-up (for example, if a global resource in one Global Resource XML File is assigned but there is no global resource of that name in the currently active Global Resources XML File, then the assigned global resource (alias) cannot be looked up).

- *The active configuration* is selected via the menu item **Tools | Active Configuration** or via the Global Resources toolbar. Clicking this command (or drop-down list in the toolbar) pops up a list of configurations across all aliases. Selecting a configuration makes that configuration active application-wide. This means that wherever a global resource (or alias) is used, the resource corresponding to the active configuration of each used alias will be loaded. The active configuration is applied to all used aliases. If an alias does not have a configuration with the name of the active configuration, then the default configuration of that alias will be used.

Global SQL Statements

SQL statements that have been saved as *.SQL files can be used as file-type global resources in DatabaseSpy. For example, if you use a development and a production database with different names and/or location, you would need two different SQL statements to query the currently active database. By defining these SQL statements as global resources, you can not only switch between the two databases with a mouse click, but also have the corresponding SQL query displayed in the SQL Editor. All you have to do, is to synchronize the SQL Editor with the data source and execute the query.

Displaying file-type global resources in the SQL Editor

File-type global resources can be viewed, edited, and executed like any other SQL in the SQL Editor. When clicking the **View** button in the **Manage Global Resources** dialog, the file is displayed in the SQL Editor in the background and the **Manage Global Resources** dialog remains open so that you can edit or delete the global resource, if required.
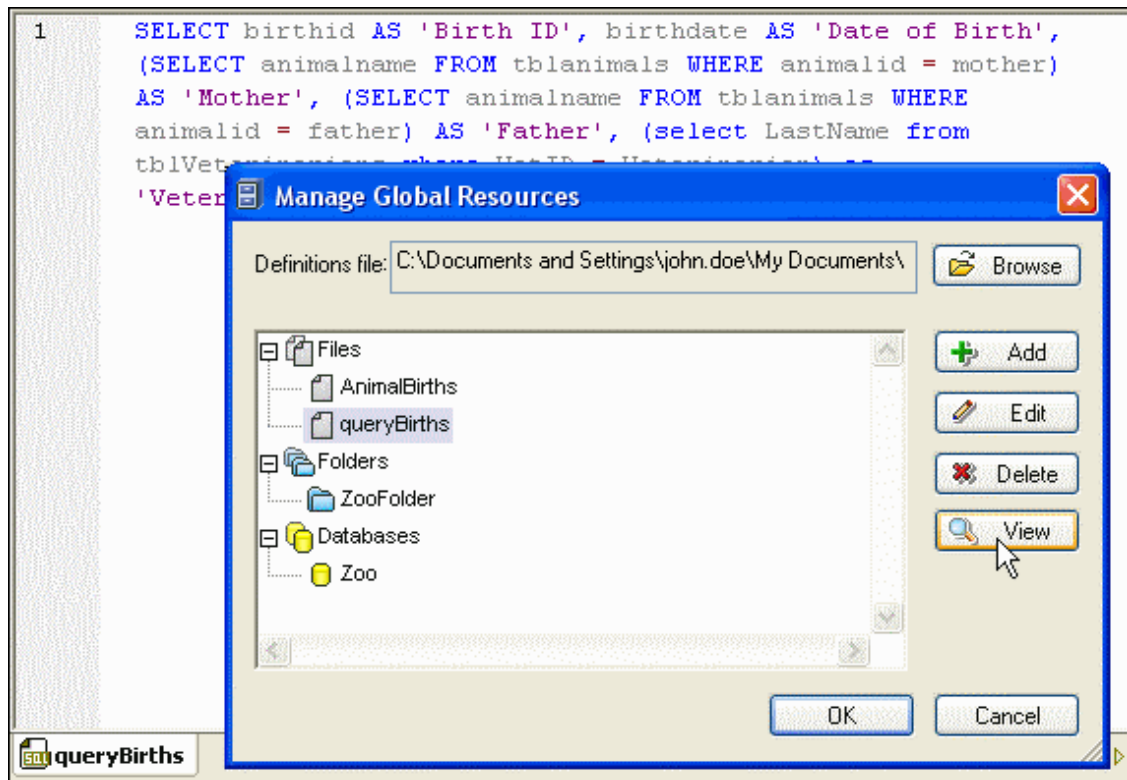
Alternatively you can also use the **File** menu to open a global resource.

To display a file-type global resource in the SQL Editor:

1. Select the menu option **File | Open Global Resource...** to display the **Choose Global Resource** dialog box.

2. Select a file-type global resource under the Files directory and click **OK**. The file is displayed under its Alias name in a new SQL Editor window.

Alternatively, do the following:

1. Select the menu option **Tools | Global Resources...** or click the **Manage Global Resources** toolbar button to open the **Manage Global Resources** dialog.

2. In the Files folder, select the Alias name of the SQL file you want to display and click the **View** button.

3. Click **Cancel** to close the **Manage Global Resources** dialog.

Data Sources in DatabaseSpy Projects

When working with DatabaseSpy, you can make use of database-type global resources in several ways:
- Add a global resource to a project as a data source
- Convert an existing data source connection to a global resource
- Copy a global resource into a project (and thus make it a local data source connection)

Adding a data source connection from a global resource
You can add a data source from a global resource either via the **File | Create a Database Connection...** menu or by right-clicking the Data Sources folder in the Project window and

choosing **Add a New Data source** from the context menu. Both options bring up the **Create a Database Connection** dialog, where you can choose a global resource from the Global Resources page.
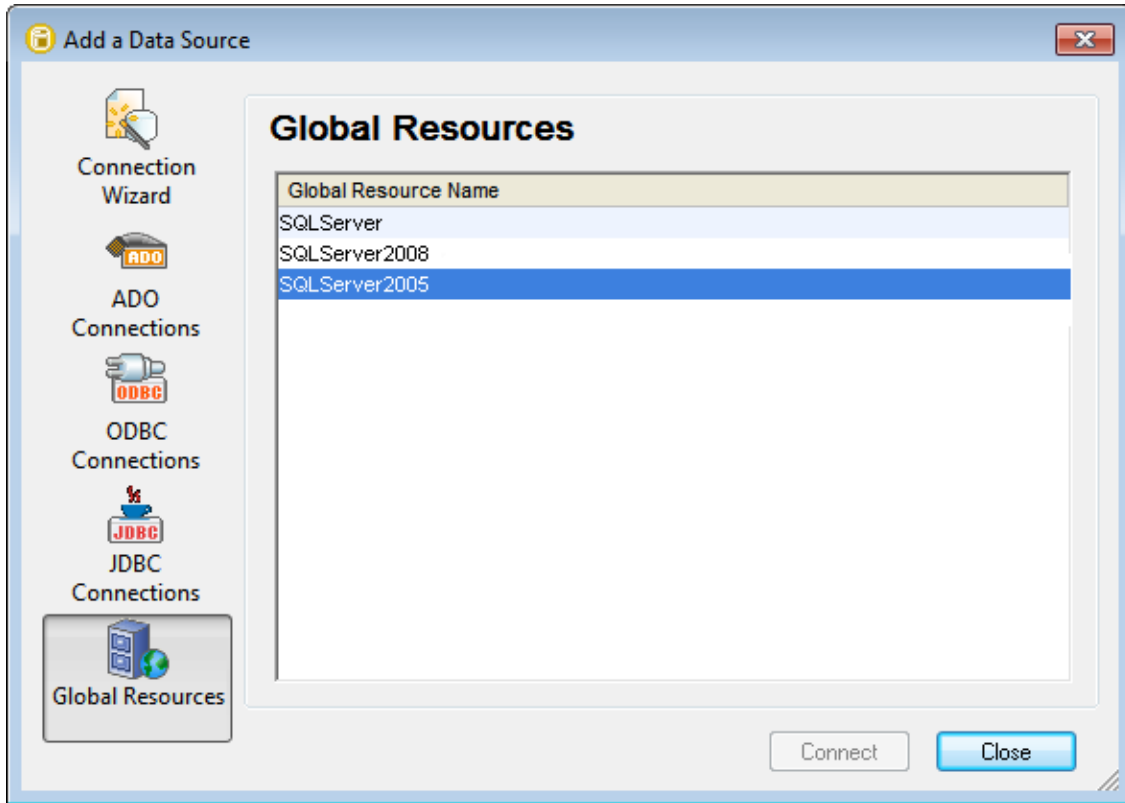
Note that a balloon help pops up when you hover over the individual entries, showing information about the different configurations in the global resource.

Converting an existing data source connection to a global resource

Any data source connection that appears in a DatabaseSpy project can be converted to a global resource with a single mouse click.
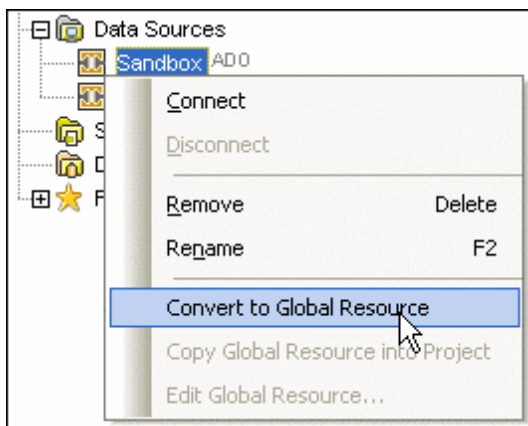
The data source is added to the Global Resources XML file and hence available in all Altova applications. Note that the icon in the Project window changes to indicate that the data source connection is now a global resource.
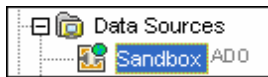
When you right-click the Data Sources folder in the Project window, you also have the option to convert all data sources in the project to global resources.

Copying a global resource into a project

If you want to use only a certain configuration of a global resource, you can add the global resource to your project, then copy it to the project as a local data source connection and remove the global resource from the project again. Make sure that you have selected the appropriate configuration before copying the global resource into the project. Only the active configuration will be available once the data source is local.

To copy a global resource into a project:

1. In the Project window, right-click a data source that has been added to the project as global resource and select **Copy Global Resource into Project** from the context menu.

2. Enter a name in the **Set a Data Source Name** dialog and click **OK**.

3. Optionally, remove the global resource from the project.

Changing the Active Configuration

One configuration of a global resource can be active at any time. This configuration is called the active configuration, and it is active application-wide. This means that the active configuration is active for all global resources aliases in all currently open files and data source connections. If an alias does not have a configuration with the name of the active configuration, then the default configuration of that alias will be used. As an example of how to change configurations, consider the case in which a file has been assigned via a global resource with multiple configurations. Each configuration maps to a different file. So, which file is selected depends on which configuration is selected as the application's active configuration.

Switching the active configuration can be done in the following ways:

• Via the menu command **Tools | Active Configuration**. Select the configuration from the command's submenu.
• In the combo box of the Global Resources toolbar (*screenshot below*), select the required configuration.

DatabaseSpy displays a dialog where you can choose whether or not the files and/or data source connections should be reloaded.
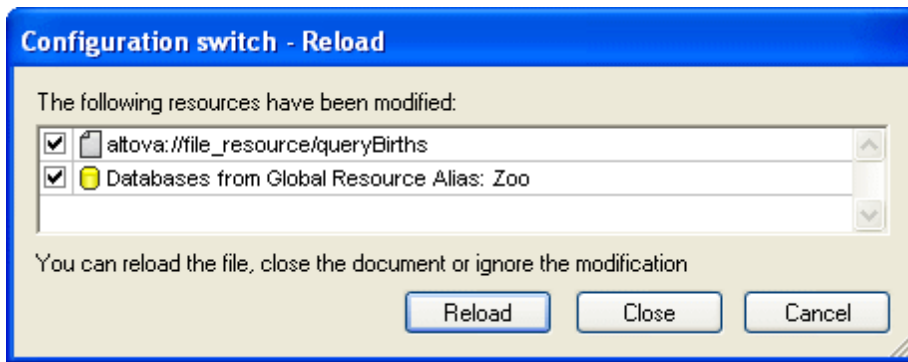
In this way, by changing the active configuration, you can change source files that are assigned via a global resource.

5.2   Connecting to a Database

In the most simple case, a database can be a local file such as a Microsoft Access or SQLite database file. In a more advanced scenario, a database may reside on a remote or network database server which does not necessarily use the same operating system as the application that connects to it and consumes data. For example, while DatabaseSpy runs on a Windows operating system, the database from which you want to access data (for example, MySQL) might run on a Linux machine.

To interact with various database types, both remote and local, DatabaseSpy relies on the data connection interfaces and database drivers that are already available on your operating system or released periodically by the major database vendors. In the constantly evolving landscape of database technologies, this approach caters for better cross-platform flexibility and interoperability.

The following diagram illustrates, in a simplified way, data connectivity options available between DatabaseSpy (illustrated as a generic client application) and a data store (which may be a database server or database file).
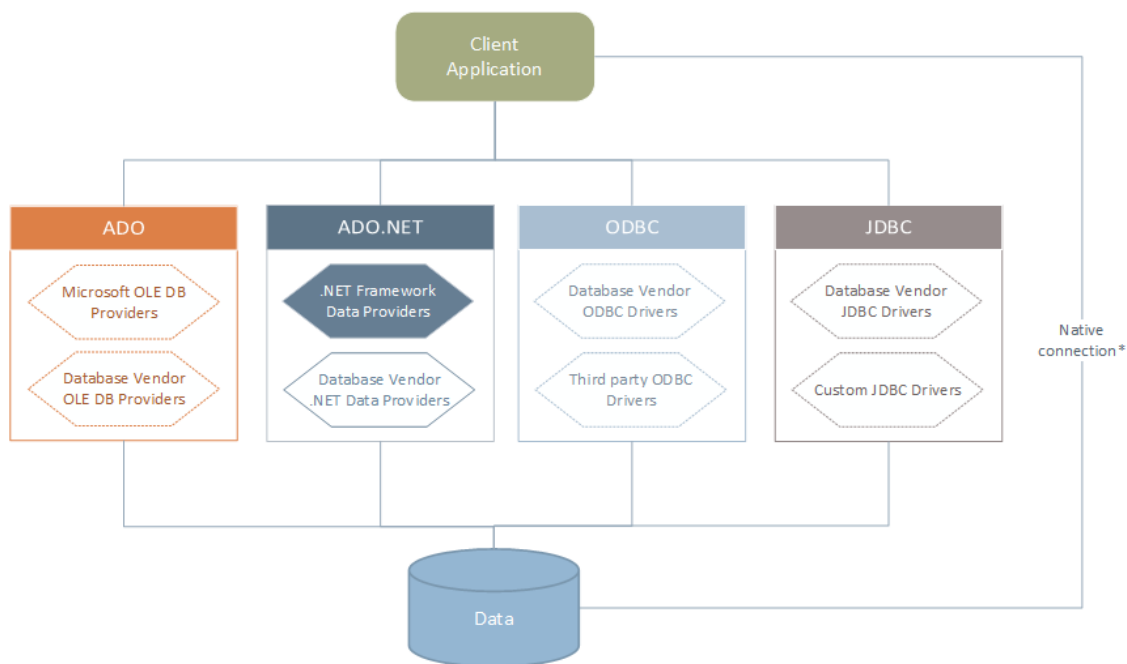
* Direct native connections are supported for SQLite and PostgreSQL databases. To connect to such databases, no additional drivers are required to be installed on your system.

As shown in the diagram above, DatabaseSpy can access any of the major database types through the following data access technologies:

- ADO (Microsoft® ActiveX® Data Objects), which, in its turn, uses an underlying OLE DB (Object Linking and Embedding, Database) provider
- ADO.NET (A set of libraries available in the Microsoft .NET Framework that enable interaction with data)
- JDBC (Java Database Connectivity)
- ODBC (Open Database Connectivity)

Some ADO.NET providers are not supported or have limited support. See ADO.NET Support Notes.

The data connection interface you should choose largely depends on your existing software infrastructure. You will typically choose the data access technology and the database driver which integrates tighter with the database system to which you want to connect. For example, to connect to a Microsoft Access 2013 database, you would build an ADO connection string that uses a native provider such as the **Microsoft Office Access Database Engine OLE DB Provider**. To connect to Oracle, on the other hand, you may want to download and install the latest JDBC, ODBC, or ADO.NET interfaces from the Oracle website.

While drivers for Windows products (such as Microsoft Access or SQL Server) may already be available on your Windows operating system, they may not be available for other database types. Major database vendors routinely release publicly available database client software and drivers which provide cross-platform access to the respective database through any combination of ADO, ADO.NET, ODBC, or JDBC. In addition to this, several third party drivers may be available for any of the above technologies. In most cases, there is more than one way to connect to the required database from your operating system, and, consequently, from DatabaseSpy. The available features, performance parameters, and the known issues will typically vary based on the data access technology or drivers used.

5.2.1   Starting the Database Connection Wizard

Whenever you take an action that requires a database connection, a wizard appears that guides you through the steps required to set up the connection.

Before you go through the wizard steps, be aware that for some database types it is necessary to install and configure separately several database prerequisites, such as a database driver or database client software. These are normally provided by the respective database vendors, and include documentation tailored to your specific Windows version. For a list of database drivers grouped by database type, see Database Drivers Overview.

To start the database connection wizard:

- On the File menu, click **Create a Database Connection**.
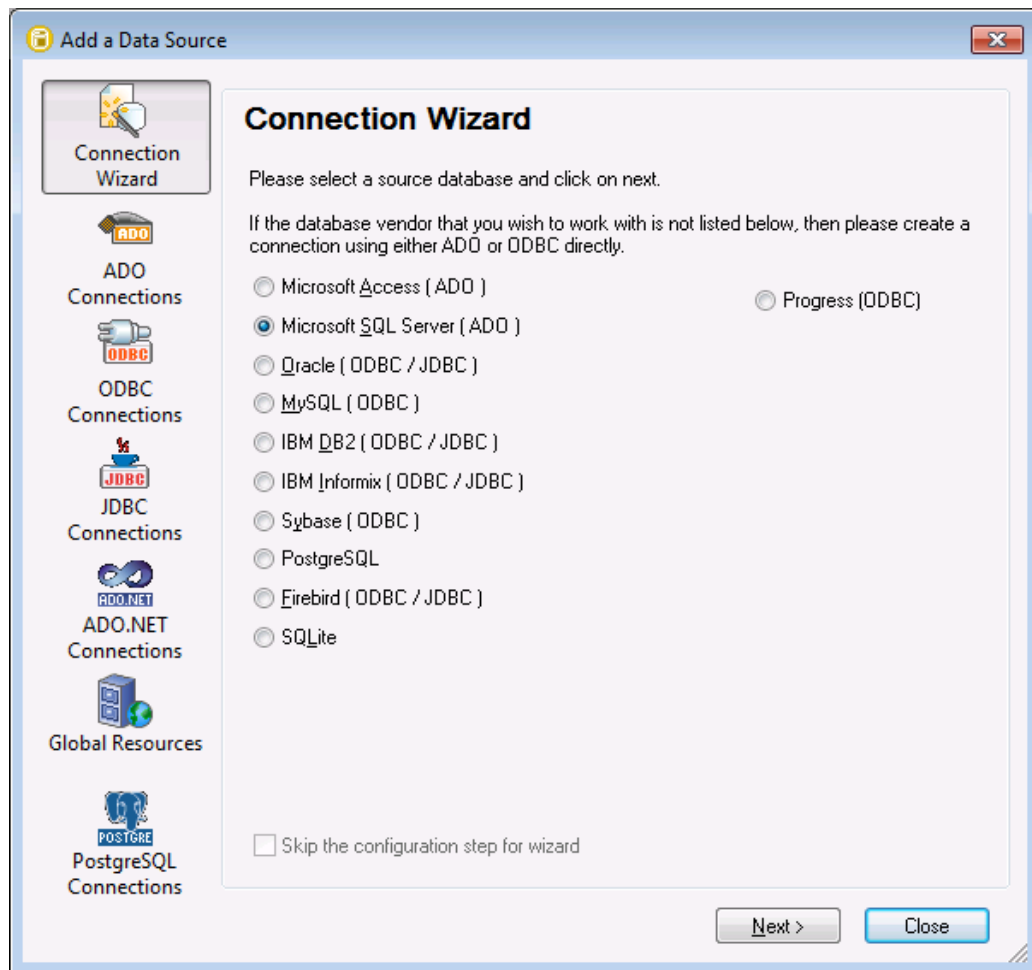
After you select a database type and click **Next**, the on-screen instructions will depend on the database kind, technology (ADO, ODBC, JDBC) and driver used.

For examples applicable to each database type, see Database Connection Examples. For instructions applicable to each database access technology, refer to the following topics:

- Setting up an ADO Connection
- Setting up an ADO.NET Connection
- Setting up an ODBC Connection
- Setting up a JDBC Connection

5.2.2   Database Drivers Overview

The following table lists common database drivers you can use to connect to a particular database through a particular data access technology. Note that this list does not aim to be either exhaustive or prescriptive; you can use other native or third party alternatives in addition to the drivers shown below.

Even though a number of database drivers are available by default on your Windows operating system (highlighted in bold in the table below), you may still want or need to download an alternative driver. For some databases, the latest driver supplied by the database vendor is likely

to perform better than the driver that shipped with the operating system.

With some exceptions, most database vendors provide database client software which normally includes any required database drivers, or provide you with an option during installation to select the drivers and components you wish to install. Database client software typically consists of administration and configuration utilities used to simplify database administration and connectivity, as well as documentation on how to install and configure the database client and any of its components.

Configuring the database client correctly is crucial for establishing a successful connection to the database. Before installing and using the database client software, it is strongly recommended to read carefully the installation and configuration instructions of the database client, since they typically vary for each database version and for each Windows version.

Database	Interface	Drivers
Firebird	ADO.NET	Firebird ADO.NET Data Provider (http://www.firebirdsql.org/en/additional-downloads/)
	JDBC	Firebird JDBC driver (http://www.firebirdsql.org/en/jdbc-driver/)
	ODBC	Firebird ODBC driver (http://www.firebirdsql.org/en/odbc-driver/)
IBM DB2	ADO	IBM OLE DB Provider for DB2
	ADO.NET	IBM Data Server Provider for .NET
	JDBC	IBM Data Server Driver for JDBC and SQLJ
	ODBC	IBM DB2 ODBC Driver
IBM DB2 for i	ADO	• IBM DB2 for i5/OS IBMDA400 OLE DB Provider • IBM DB2 for i5/OS IBMDARLA OLE DB Provider • IBM DB2 for i5/OS IBMDASQL OLE DB Provider
	ADO.NET	.NET Framework Data Provider for IBM i
	JDBC	IBM Toolbox for Java JDBC Driver
	ODBC	iSeries Access ODBC Driver
IBM Informix	ADO	IBM Informix OLE DB Provider
	JDBC	IBM Informix JDBC Driver
	ODBC	IBM Informix ODBC Driver
Microsoft Access	ADO	• **Microsoft Jet OLE DB Provider** • **Microsoft Access Database Engine OLE DB Provider**
	ADO.NET	**.NET Framework Data Provider for OLE DB**
	ODBC	• **Microsoft Access Driver**
Microsoft SQL Server	ADO	• **Microsoft OLE DB Provider for SQL Server** • **SQL Server Native Client**

Database	Interface	Drivers
	ADO.NET	• **.NET Framework Data Provider for SQL Server** • **.NET Framework Data Provider for OLE DB**
	JDBC	• Microsoft JDBC Driver for SQL Server (http://msdn.microsoft.com/en-us/data/aa937724.aspx)
	ODBC	• **SQL Server Native Client**
MySQL	ADO.NET	Connector/NET (http://dev.mysql.com/downloads/connector/net/)
	JDBC	Connector/J (http://dev.mysql.com/downloads/connector/j/)
	ODBC	Connector/ODBC (http://dev.mysql.com/downloads/connector/odbc/)
Oracle	ADO	• Oracle Provider for OLE DB • **Microsoft OLE DB Provider for Oracle**
	ADO.NET	Oracle Data Provider for .NET (http://www.oracle.com/technetwork/topics/dotnet/index-085163.html)
	JDBC	• JDBC Thin Driver • JDBC Oracle Call Interface (OCI) Driver These drivers are typically installed during the installation of your Oracle database client. Connect through the OCI Driver (not the Thin Driver) if you are using the Oracle XML DB component.
	ODBC	• **Microsoft ODBC for Oracle** • Oracle ODBC Driver (typically installed during the installation of your Oracle database client)
PostgreSQL	JDBC	Postgre JDBC Driver (https://jdbc.postgresql.org/download.html)
	ODBC	psqlODBC (https://odbc.postgresql.org/)
	Native Connection	Yes
Progress OpenEdge	JDBC	JDBC Connector (https://www.progress.com/jdbc/openedge)
	ODBC	ODBC Connector (https://www.progress.com/odbc/openedge)
SQLite	Native Connection	Yes
Sybase	ADO	Sybase ASE OLE DB Provider
	JDBC	jConnect™ for JDBC
	ODBC	Sybase ASE ODBC Driver

* The drivers highlighted in bold are Microsoft-supplied. If not already available on Windows system,

they can be downloaded from the official Microsoft website.

> Some ADO.NET providers are not supported or have limited support. See ADO.NET Support Notes.

To understand the capabilities and limitations of each data access technology with respect to each database type, refer to the documentation of that particular database product and also test the connection against your specific environment. To avoid common connectivity issues, consider the following general notes and recommendations:

- Since 32-bit and 64-bit drivers may not be compatible, make sure to install and configure the driver version applicable to your Altova application. For example, if you are using a 32-bit Altova application on a 64-bit operating system, set up your database connection using the 32-bit driver version.
- The latest driver versions may provide features not available in older editions.
- When setting up an ODBC data source, it is generally recommended to create the data source name (DSN) as System DSN instead of User DSN.
- When setting up a JDBC data source, ensure that JRE (Java Runtime Environment) is installed and that the CLASSPATH environment variable of the operating system is configured.
- For the support details and known issues applicable to Microsoft-supplied database drivers, refer to the MSDN documentation.
- For the installation instructions and support details of any drivers or database client software that you install from a database vendor, check the documentation provided with the installation package. Whether you are using an official or third party database driver, the most comprehensive information and the configuration procedures applicable to that specific driver on your specific operating system is normally part of the driver installation package.

5.2.3   Setting up an ADO Connection

Microsoft ActiveX Data Objects (ADO) is a data access technology that enables you to connect to a variety of data sources through OLE DB. OLE DB is an alternative interface to ODBC or JDBC; it provides uniform access to data in a COM (Component Object Model) environment. ADO is the typical choice for connecting to Microsoft native databases such as Microsoft Access or SQL Server, although you can also use it for other data sources.

To set up an ADO connection:

1. Start the database connection wizard.
2. Click **ADO Connections**.
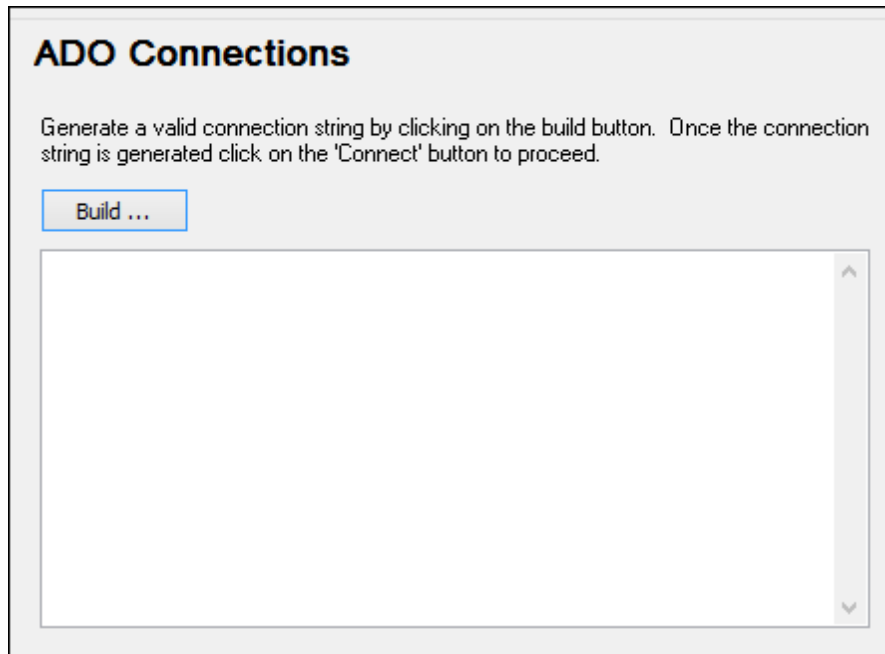
3. Click **Build**.
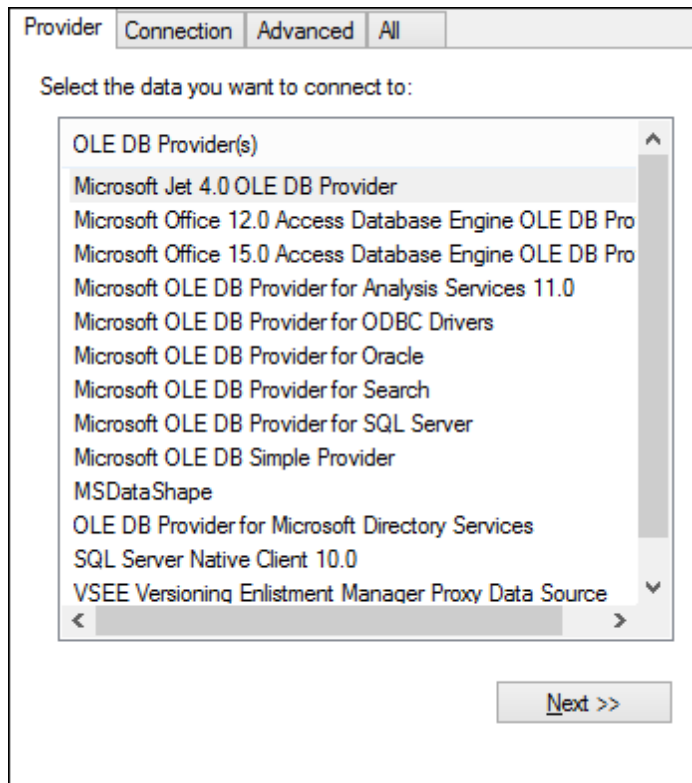
4. Select the data provider through which you want to connect. The table below lists a few common scenarios.

To connect to this database...	Use this provider...
Microsoft Access	• **Microsoft Office Access Database Engine OLE DB Provider** When connecting to Microsoft Access 2003, you can also use the **Microsoft Jet OLE DB Provider**.
SQL Server	• **SQL Server Native Client** • **Microsoft OLE DB Provider for SQL Server**
Other database	Select the provider applicable to your database. If an OLE DB provider to your database is not available, install the required driver from the database vendor (see Database Drivers Overview). Alternatively, set up an ODBC or JDBC connection. If the operating system has an ODBC driver to the required database, you can also use the **Microsoft OLE DB Provider for ODBC Drivers**.

5.   Click **Next** and complete the wizard.

The subsequent wizard steps are specific to the provider you chose. For SQL Server, you will need to provide or select the host name of the database server, as well as the database username and password. For Microsoft Access, you will be asked to browse for or provide the path to the database file.

The complete list of initialization properties (connection parameters) is available in the **All** tab of the connection dialog box—these properties vary depending on the chosen provider. The following sections provide guidance on configuring the basic initialization properties for Microsoft Access and SQL Server databases:

- Setting up the SQL Server Data Link Properties
- Setting up the Microsoft Access Data Link Properties

Connecting to an Existing Microsoft Access Database

This approach is suitable when you want to connect to a Microsoft Access database which is not password-protected. If the database is password-protected, set up the database password as shown in Connecting to Microsoft Access (ADO).

To connect to an existing Microsoft Access database:

1.   Run the database connection wizard (see Starting the Database Connection Wizard).
2.   Select **Microsoft Access (ADO)**, and then click **Next**.
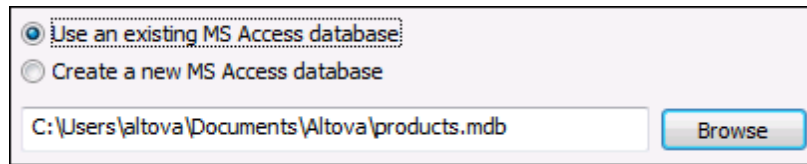
3. Select **Use an existing MS Access database**.
4. Browse for the database file, or enter the path to it (either relative or absolute) .
5. Click **Connect**.

Creating a New Microsoft Access Database

As an alternative to connecting to an existing database file, you can create a new Microsoft
Access database file (.accdb, .mdb) and connect to it, even if Microsoft Access is not installed on
the computer. The database file created by DatabaseSpy is empty. To create the required
database structure, use Microsoft Access or the Design Editor of DatabaseSpy..

To create a new Microsoft Access database:

1. Run the database connection wizard (see Starting the Database Connection Wizard).
2. Select **Microsoft Access (ADO)**, and then click **Next**.
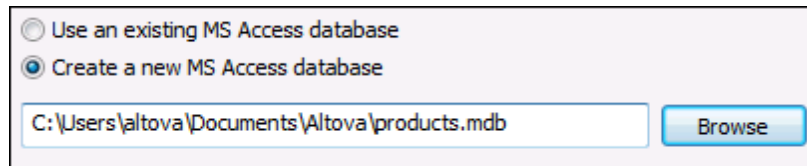
3. Select **Create a new MS Access database**, and then enter the path (either relative or
    absolute) of the database file to be created (for example, **c:\users\public
    \products.mdb**). Alternatively, click **Browse** to select a folder, type the name of the
    database file in the "File name" text box (for example, **products.mdb**), and click **Save**.

> **Notes**
> *   Make sure that you have write permissions to the folder where you want to
>   create the database file.
> *   The database file name must have the **.mdb** or **.accdb** extension.

4. Click **Connect**.

Setting up the SQL Server Data Link Properties

When you connect to a Microsoft SQL Server database through ADO (see Setting up an ADO
Connection), ensure that the following data link properties are configured correctly in the **All** tab of
the Data Link Properties dialog box.
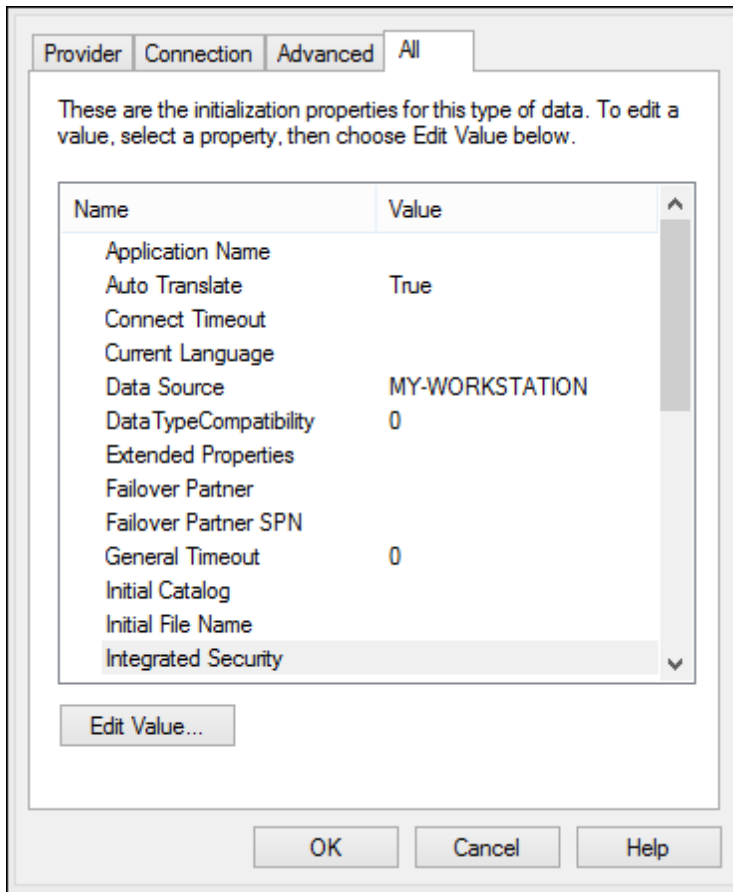
Data Link Properties dialog box

Property	Notes
Integrated Security	If you selected the **SQL Server Native Client** data provider on the **Provider** tab, set this property to a space character.
Persist Security Info	Set this property to **True**.
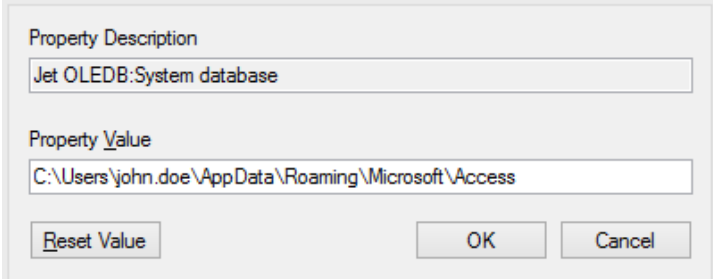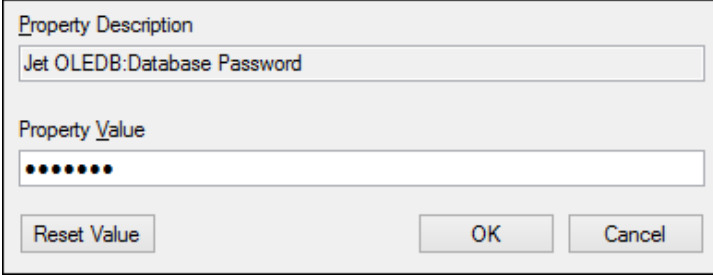
Setting up the Microsoft Access Data Link Properties

When you connect to a Microsoft Access database through ADO (see Setting up an ADO Connection), ensure that the following properties are configured correctly in the **All** tab of the Data Link Properties dialog box.

Data Link Properties dialog box

Property	Notes
Data Source	This property stores the path to the Microsoft Access database file. To avoid database connectivity issues, it is recommended to use the UNC (Universal Naming Convention) path format, for example: `\\anyserver\share$\filepath`
Jet OLEDB:System Database	This property stores the path to the workgroup information file. You may need to explicitly set the value of this property before you can connect to a Microsoft Access database. If you cannot connect due to a "workgroup information file" error, locate the workgroup information file (**System.MDW**) applicable to your user profile (see http://support.microsoft.com/kb/305542 for instructions), and set the property value to the path of the **System.MDW** file.

Property	Notes
	Property Description Jet OLEDB:System database Property Value C:\Users\john.doe\AppData\Roaming\Microsoft\Access Reset Value OK Cancel
Jet OLEDB:Database Password	If the database is password-protected, set the value of this property to the database password. Property Description Jet OLEDB:Database Password Property Value ••••••• Reset Value OK Cancel

5.2.4   Setting up an ADO.NET Connection

ADO.NET is a set of Microsoft .NET Framework libraries designed to interact with data, including data from databases. To connect to a database from DatabaseSpy through ADO.NET, Microsoft .NET Framework 4 or later is required. As shown below, you connect to a database through ADO.NET by selecting a .NET provider and supplying a connection string.

A .NET data provider is a collection of classes that enables connecting to a particular type of data source (for example, a SQL Server, or an Oracle database), executing commands against it, and fetching data from it. In other words, with ADO.NET, an application such as DatabaseSpy interacts with a database through a data provider. Each data provider is optimized to work with the specific type of data source that it is designed for. There are two types of .NET providers:

1. Supplied by default with Microsoft .NET Framework.
2. Supplied by major database vendors, as an extension to the .NET Framework. Such ADO.NET providers must be installed separately and can typically be downloaded from the website of the respective database vendor.

Note: Certain ADO.NET providers are not supported or have limited support. See ADO.NET Support Notes.

To set up an ADO.NET connection:

1. Start the database connection wizard.
2. Click **ADO.NET Connections**.
3. Select a .NET data provider from the list.

> The list of providers available by default with the .NET Framework appears in the "Provider" list. Vendor-specific .NET data providers are available in the list only if they are already installed on your system. To become available, vendor-specific .NET providers must be installed into the GAC (Global Assembly Cache), by running the .msi or .exe file supplied by the database vendor.

4. Enter a database connection string. A connection string defines the database connection information, as semicolon-delimited key/value pairs of connection parameters. For example, a connection string such as `Data Source=DBSQLSERV;Initial Catalog=ProductsDB;User ID=dbuser;Password=dbpass` connects to the SQL Server database `ProductsDB` on server `DBSQLSERV`, with the user name `dbuser` and password `dbpass`. You can create a connection string by typing the key/value pairs directly into the "Connection String" dialog box. Another option is to create it with Visual Studio (see Creating a Connection String in Visual Studio).

> The syntax of the connection string depends on the provider selected from the "Provider" list. For examples, see Sample ADO.NET Connection Strings.
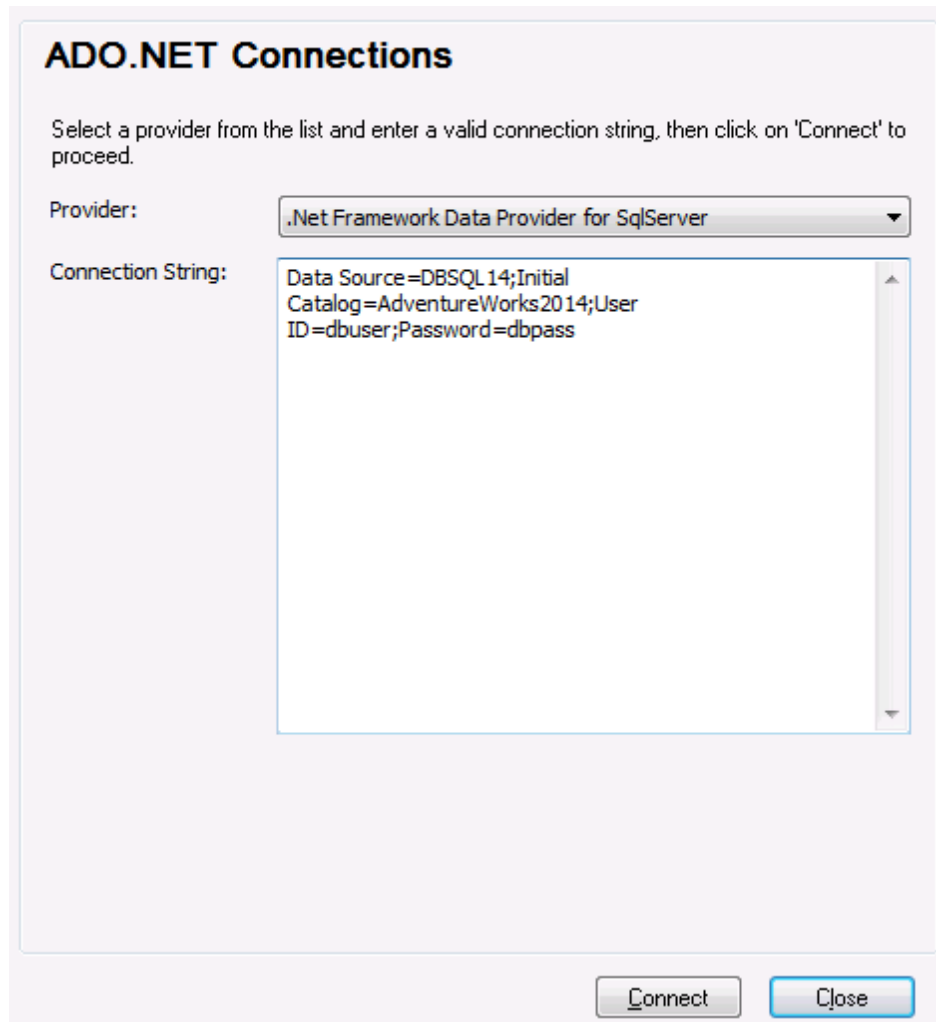
5.   Click **Connect**.

Creating a Connection String in Visual Studio

In order to connect to a data source using ADO.NET, a valid database connection string is required. The following instructions show you how to create a connection string from Visual Studio.

To create a connection string in Visual Studio:

1.   On the **Tools** menu, click **Connect to Database**.
2.   Select a data source from the list (in this example, Microsoft SQL Server). The Data Provider is filled automatically based on your choice.
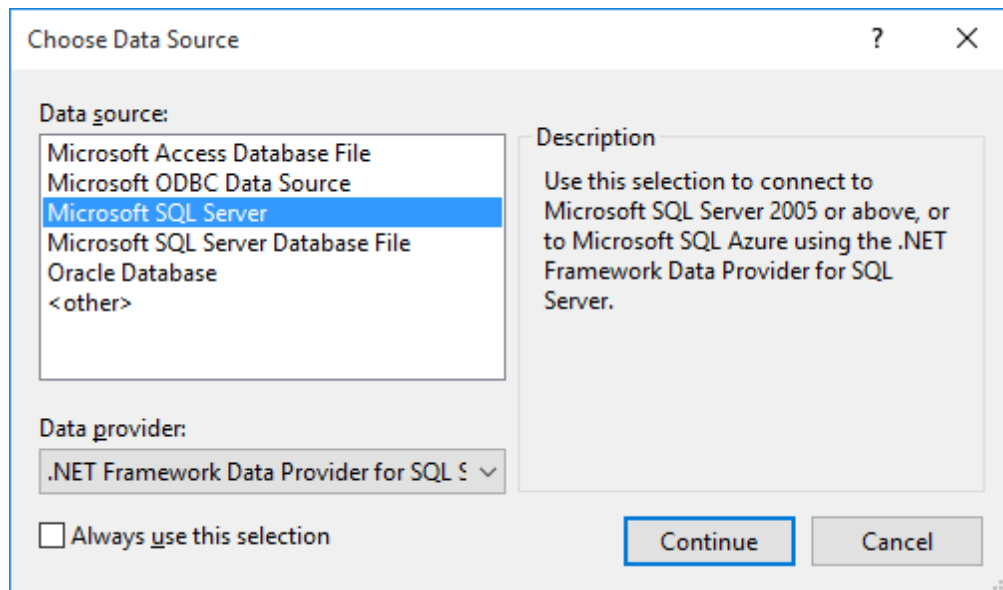
3. Click **Continue**.

4. Enter the server host name and the user name and password to the database. In this example, we are connecting to the database `ProductsDB` on server `DBSQLSERV`, using SQL Server authentication.

5. Click **OK**.

If the database connection is successful, it appears in the Server Explorer window. You can display the Server Explorer window using the menu command **View | Server Explorer**. To obtain the database connection string, right-click the connection in the Server Explorer window, and select **Properties**. The connection string is now displayed in the Properties window of Visual Studio. Note that, before pasting the string into the "Connection String" box of DatabaseSpy, you will need to replace the asterisk (*) characters with the actual password.

Sample ADO.NET Connection Strings

To set up an ADO.NET connection, you need to select an ADO.NET provider from the database connection dialog box and enter a connection string (see also Setting up an ADO.NET Connection). Sample ADO.NET connection strings for various databases are listed below under the .NET provider where they apply.

.NET Framework Data Provider for IBM i

This provider is installed as part of *IBM i Access Client Solutions - Windows Application Package*. A sample connection string looks as follows:

```
DataSource=ServerAddress;UserID=user;Password=password;DataCompression=True;
```

For more information, see the ".NET Provider Technical Reference" help file included in the installation package above.

.NET Framework Data Provider for MySQL

This provider can be downloaded from MySQL website (https://dev.mysql.com/downloads/connector/net/). A sample connection string looks as follows:

```
Server=127.0.0.1;Uid=root;Pwd=12345;Database=test;
```

See also: https://dev.mysql.com/doc/connector-net/en/connector-net-programming-connecting-connection-string.html

.NET Framework Data Provider for SQL Server

A sample connection string looks as follows:

```
Data Source=DBSQLSERV;Initial Catalog=ProductsDB;User
ID=dbuser;Password=dbpass
```

See also: https://msdn.microsoft.com/en-us/library/ms254500(v=vs.110).aspx

IBM DB2 Data Provider 10.1.2 for .NET Framework 4.0

```
Database=PRODUCTS;UID=user;Password=password;Server=localhost:50000;
```

Note: This provider is typically installed with the IBM DB2 Data Server Client package. If the provider is missing from the list of ADO.NET providers after installing IBM DB2 Data Server Client package, refer to the following technical note: http://www-01.ibm.com/support/docview.wss?uid=swg21429586.

See also: http://www.ibm.com/support/knowledgecenter/en/SSEPGG_10.1.0/

com.ibm.swg.im.dbclient.adonet.ref.doc/doc/DB2ConnectionClassConnectionStringProperty.html

Oracle Data Provider for .NET (ODP.NET)

The installation package which includes the ODP.NET provider can be downloaded from Oracle website (see http://www.oracle.com/technetwork/topics/dotnet/downloads/index.html). A sample connection string looks as follows:

```
Data Source=DSORCL;User Id=user;Password=password;
```

Where DSORCL is the name of the data source which points to an Oracle service name defined in the **tnsnames.ora** file, as described in Connecting to Oracle (ODBC).

To connect without configuring a service name in the **tnsnames.ora** file, use a string such as:

```
Data Source=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=host)
(PORT=port)))(CONNECT_DATA=(SERVER=DEDICATED)(SERVICE_NAME=MyOracleSID)));User
Id=user;Password=password;
```

See also: https://docs.oracle.com/cd/B28359_01/win.111/b28375/featConnecting.htm

ADO.NET Support Notes

The following table lists known ADO.NET database drivers that are currently not supported or have limited support in DatabaseSpy.

Database	Driver	Support notes
All databases	**.Net Framework Data Provider for ODBC**	Limited support. Known issues exist with Microsoft Access connections. It is recommended to use ODBC direct connections instead. See Setting up an ODBC Connection.
	.Net Framework Data Provider for OleDb	Limited support. Known issues exist with Microsoft Access connections. It is recommended to use ADO direct connections instead. See Setting up an ADO Connection.
Firebird	**Firebird ADO.NET Data Provider**	Limited support. It is recommended to use ODBC or JDBC instead. See Connecting to Firebird (ODBC) and Connecting to Firebird (JDBC).
Informix	**IBM Informix Data Provider for .NET Framework 4.0**	Not supported. Use **DB2 Data Server Provider** instead.
IBM DB2 for i (iSeries)	**.Net Framework Data Provider for i5/OS**	Not supported. Use **.Net Framework Data Provider for IBM i** instead, installed as part of the *IBM i Access*

Database	Driver	Support notes
		Client Solutions - Windows Application Package.
Oracle	**.Net Framework Data Provider for Oracle**	Limited support. Although this driver is provided with the .NET Framework, its usage is discouraged by Microsoft, because it is deprecated.
PostgreSQL	-	No ADO.NET drivers for this vendor are supported.
Sybase	-	No ADO.NET drivers for this vendor are supported.

5.2.5   Setting up an ODBC Connection

ODBC (Open Database Connectivity) is a widely used data access technology that enables you to connect to a database from DatabaseSpy. It can be used either as primary means to connect to a database, or as an alternative to OLE DB- or JDBC-driven connections.

To connect to a database through ODBC, first you need to create an ODBC data source name (DSN) on the operating system. This step is not required if the DSN has already been created, perhaps by another user of the operating system. The DSN represents a uniform way to describe the database connection to any ODBC-aware client application on the operating system, including DatabaseSpy. DSNs can be of the following types:

- System DSN
- User DSN
- File DSN

A *System* data source is accessible by all users with privileges on the operating system. A *User* data source is available to the user who created it. Finally, if you create a *File DSN*, the data source will be created as a file with the .dsn extension which you can share with other users, provided that they have installed the drivers used by the data source.

Any DSNs already available on your machine are listed by the database connection dialog box when you click **ODBC connections** on the ODBC connections dialog box.
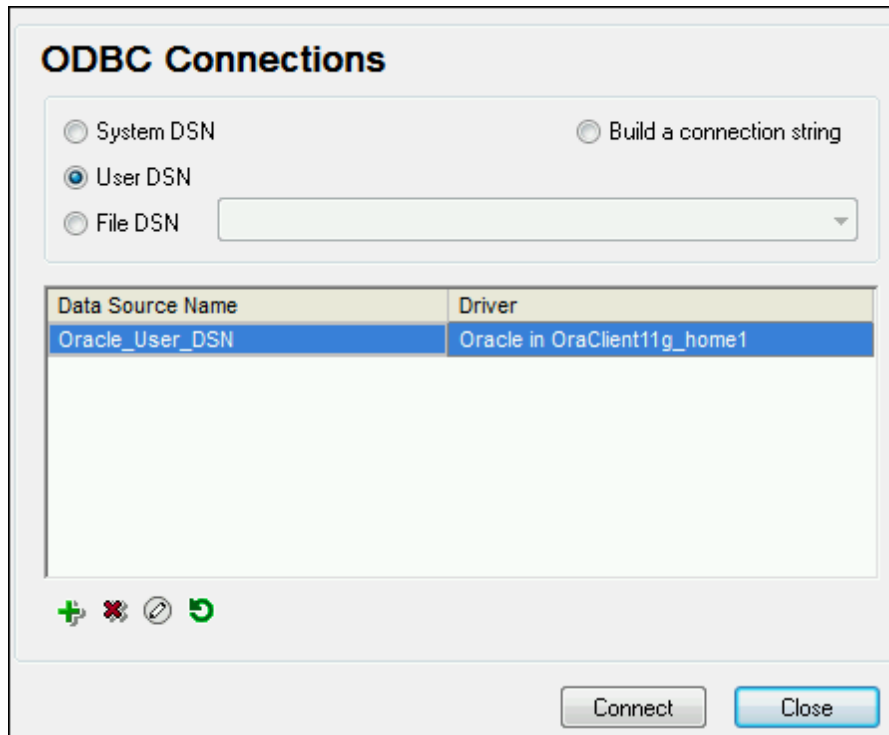
ODBC Connections dialog box

If a DSN to the required database is not available, the DatabaseSpy database connection wizard will assist you to create it; however, you can also create it directly on your Windows operating system. In either case, before you proceed, ensure that the ODBC driver applicable for your database is in the list of ODBC drivers available to the operating system (see Viewing the Available ODBC Drivers).

To connect by using a new DSN:

1. Start the database connection wizard.
2. On the database connection dialog box, click **ODBC Connections**.
3. Select a data source type (User DSN, System DSN, File DSN).

> To create a System DSN, you need administrative rights on the operating system.

4. Click **Add** .
5. Select a driver, and then click **User DSN** or **System DSN** (depending on the type of the DSN you want to create). If the driver applicable to your database is not listed, download it from the database vendor and install it (see Database Drivers Overview).
6. On the dialog box that pops up, fill in any driver specific connection information to complete the setup.

For the connection to be successful, you will need to provide the host name (or IP address) of the database server, as well as the database username and password. There may be other optional connection parameters—these parameters vary between database providers. For detailed

information about the parameters specific to each connection method, consult the documentation of the driver provider. Once created, the DSN becomes available in the list of data source names. This enables you to reuse the database connection details any time you want to connect to the database. Note that User DSNs are added to the list of User DSNs whereas System DSNs are added to the list of System DSNs.

To connect by using an existing DSN:

1. [Start the database connection wizard](#).
2. Click **ODBC Connections**.
3. Choose the type of the existing data source (User DSN, System DSN, File DSN).
4. Click the existing DSN record, and then click **Connect**.

To build a connection string based on an existing .dsn file:

1. [Start the database connection wizard](#).
2. Click **ODBC Connections**.
3. Select **Build a connection string**, and then click **Build**.
4. If you want to build the connection string using a File DSN, click the **File Data Source** tab. Otherwise, click the **Machine Data Source** tab. (System DSNs and User DSNs are known as "Machine" data sources.)
5. Select the required .dsn file, and then click **OK**.

**To connect by using a prepared connection string:**

1. [Start the database connection wizard](#).
2. Click **ODBC Connections**.
3. Select **Build a connection string**.
4. Paste the connection string into the provided box, and then click **Connect**.

Viewing the Available ODBC Drivers

You can view the ODBC drivers available on your operating system in the ODBC Data Source Administrator. You can access the ODBC Data Source Administrator (**Odbcad32.exe**) from the Windows Control Panel, under **Administrative Tools**. On 64-bit operating systems, there are two versions of this executable:

- The 32-bit version of the **Odbcad32.exe** file is located in the **C:\Windows\SysWoW64** directory (assuming that **C:** is your system drive).
- The 64-bit version of the **Odbcad32.exe** file is located in the **C:\Windows\System32** directory.

Any installed 32-bit database drivers are visible in the 32-bit version of ODBC Data Source Administrator, while 64-bit drivers—in the 64-bit version. Therefore, ensure that you check the database drivers from the relevant version of ODBC Data Source Administrator.
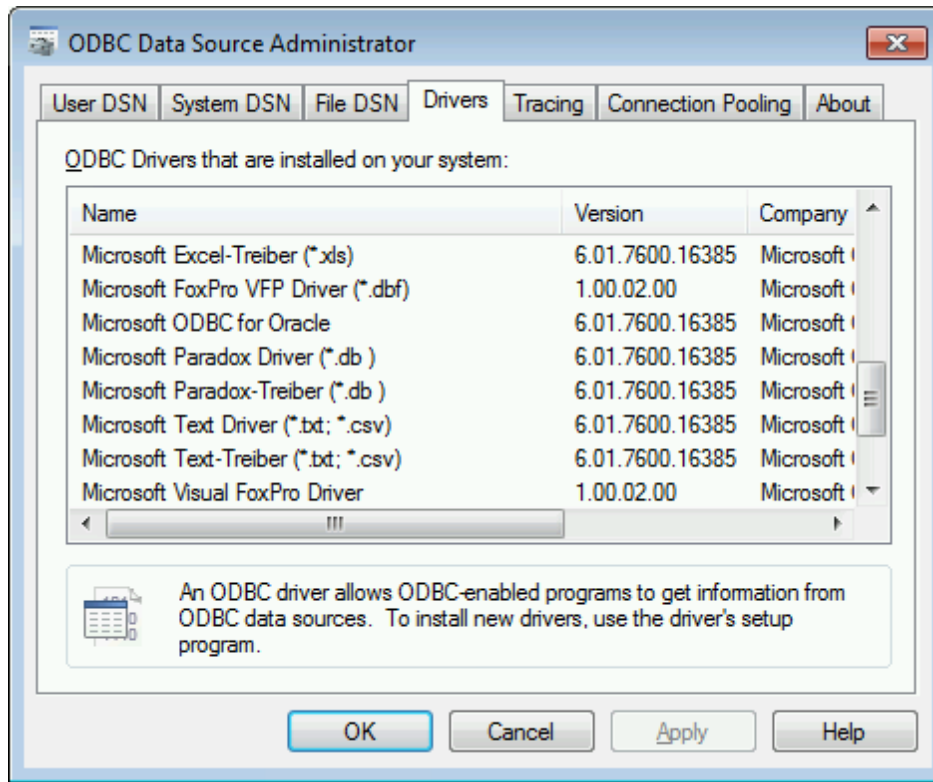
ODBC Data Source Administrator

If the driver to your target database does not exist in the list, or if you want to add an alternative driver, you will need to download it from the database vendor (see Database Drivers Overview). Once the ODBC driver is available on your system, you are ready to create ODBC connections with it (see Setting up an ODBC Connection).

5.2.6   Setting up a JDBC Connection

JDBC (Java Database Connectivity) is a database access interface which is part of the Java software platform from Oracle. JDBC connections are generally more resource-intensive than ODBC connections but may provide features not available through ODBC.
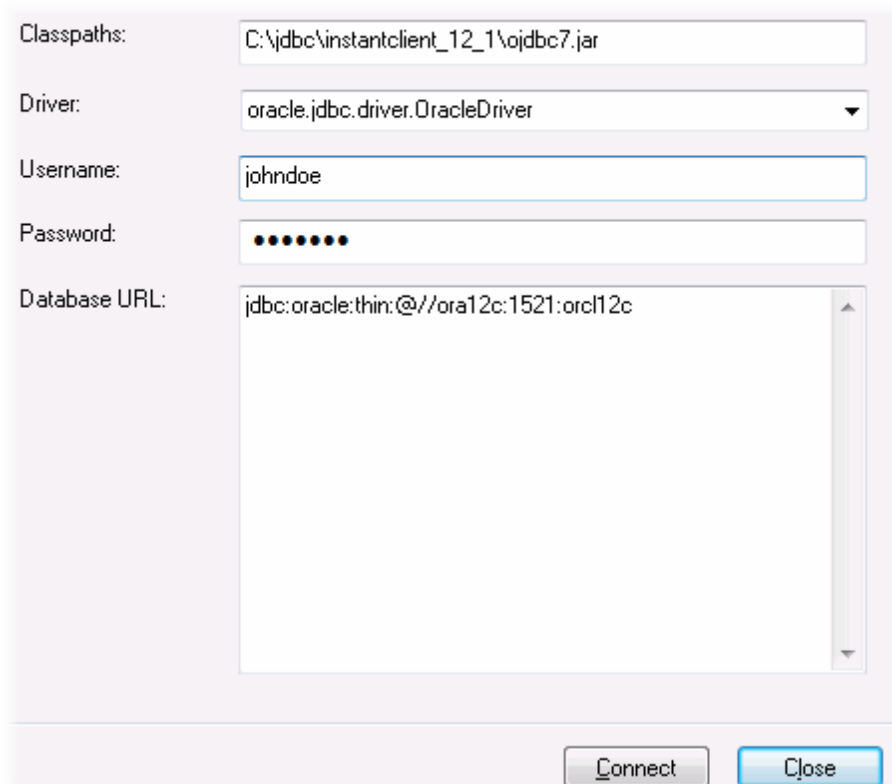
Prerequisites

- JRE (Java Runtime Environment) or Java Development Kit (JDK) must be installed. If you have not installed it already, check the official Java website for the download package and installation instructions.
- The JDBC drivers from the database vendor must be installed. If you are connecting to an Oracle database, note that some Oracle drivers are specific to certain JRE versions and may require additional components and configuration. The documentation of your Oracle product (for example, the "Oracle Database JDBC Developer's Guide and Reference") includes detailed instructions about the configuration procedure for each JDBC driver.
- The operating system's `PATH` environment variable must include the path to the `bin` directory of the JRE or JDK installation directory, for example `C:\Program Files (x86)\Java\jre1.8.0_51\bin`.

- The CLASSPATH environment variable must include the path to the JDBC driver (one or several .jar files) on your Windows operating system. When you install some database clients, the installer may configure this variable automatically. The documentation of the JDBC driver will typically include step-by-step instructions on setting the CLASSPATH variable (see also Configuring the CLASSPATH).

Setting up a JDBC connection

1. Start the database connection wizard.
2. Click **JDBC Connections**.
3. Optionally, enter a semicolon-separated list of .jar file paths in the "Classpaths" text box. The .jar libraries entered here will be loaded into the environment in addition to those already defined in the CLASSPATH environment variable. When you finish editing the "Classpaths" text box, any JDBC drivers found in the source .jar libraries are automatically added to the "Driver" list (see the next step).

Classpaths:	C:\jdbc\instantclient_12_1\ojdbc7.jar
Driver:	oracle.jdbc.driver.OracleDriver ▼
Username:	johndoe
Password:	•••••••
Database URL:	jdbc:oracle:thin:@//ora12c:1521:orcl12c

[Connect] [Close]

4. Next to "Driver", select a JDBC driver from the list, or enter a Java class name. Note that this list contains any JDBC drivers configured through the CLASSPATH environment variable (see Configuring the CLASSPATH), as well as those found in the "Classpaths" text box.

> The JDBC driver paths defined in the CLASSPATH variable, as well as any .jar file paths entered directly in the database connection dialog box are all supplied to the Java Virtual Machine (JVM). The JVM then decides which drivers to use in order to establish a connection. It is recommended to keep track of Java classes loaded into

> the JVM so as not to create potential JDBC driver conflicts and avoid unexpected results when connecting to the database.

5. Enter the username and password to the database in the corresponding boxes.
6. In the Database URL text box, enter the JDBC connection URL (string) in the format specific to your database type. The following table describes the syntax of JDBC connection URLs (strings) for common database types.

Database	JDBC Connection URL
Firebird	`jdbc:firebirdsql://`**`<host>`**`[:`**`<port>`**`]/`**`<database path or alias>`**
IBM DB2	`jdbc:db2://`**`hostName`**`:`**`port`**`/`**`databaseName`**
IBM Informix	`jdbc:informix-sqli://`**`hostName`**`:`**`port`**`/`**`databaseName`**`:INFORMIXSERVER=`**`myserver`**
Microsoft SQL Server	`jdbc:sqlserver://`**`hostName`**`:`**`port`**`;`**`databaseName`**`=name`
MySQL	`jdbc:mysql://`**`hostName`**`:`**`port`**`/`**`databaseName`**
Oracle	`jdbc:oracle:thin:@//`**`hostName`**`:`**`port`**`:`**`service`**
Oracle XML DB	`jdbc:oracle:oci:@//`**`hostName`**`:`**`port`**`:`**`service`**
PostgreSQL	`jdbc:postgresql://`**`hostName`**`:`**`port`**`/`**`databaseName`**
Progress OpenEdge	`jdbc:datadirect:openedge://`**`host`**`:`**`port`**`;databaseName=`**`db_name`**
Sybase	`jdbc:sybase:Tds:`**`hostName`**`:`**`port`**`/`**`databaseName`**

Note: Syntax variations to the formats listed above are also possible (for example, the database URL may exclude the port or may include the username and password to the database). Check the documentation of the database vendor for further details.

7. Click **Connect**.

Configuring the CLASSPATH

The CLASSPATH environment variable is used by the Java Runtime Environment (JRE) to locate Java classes and other resource files on your operating system. When you connect to a database through JDBC, this variable must be configured to include the path to the JDBC driver on your operating system, and, in some cases, the path to additional library files specific to the database type you are using.

The following table lists sample file paths that must be typically included in the CLASSPATH variable. Importantly, you may need to adjust this information based on the location of the JDBC driver on your system, the JDBC driver name, as well as the JRE version present on your operating system. To avoid connectivity problems, check the installation instructions and any pre-installation or post-installation configuration steps applicable to the JDBC driver installed on your operating system.

Database	Sample CLASSPATH entries
Firebird	`C:\Program Files\Firebird\Jaybird-2.2.8-JDK_1.8\jaybird-full-2.2.8.jar`
IBM DB2	`C:\Program Files (x86)\IBM\SQLLIB\java\db2jcc.jar;C:\Program Files (x86)\IBM\SQLLIB\java\db2jcc_license_cu.jar;`
IBM Informix	`C:\Informix_JDBC_Driver\lib\ifxjdbc.jar;`
Microsoft SQL Server	`C:\Program Files\Microsoft JDBC Driver 4.0 for SQL Server\sqljdbc_4.0\enu\sqljdbc.jar`
MySQL	`mysql-connector-java-version-bin.jar;`
Oracle	`ORACLE_HOME\jdbc\lib\ojdbc6.jar;`
Oracle (with XML DB)	`ORACLE_HOME\jdbc\lib\ojdbc6.jar;ORACLE_HOME\LIB\xmlparserv2.jar;ORACLE_HOME\RDBMS\jlib\xdb.jar;`
PostgreSQL	`<installation directory>\postgresql.jar`
Progress OpenEdge	`%DLC%\java\openedge.jar;%DLC%\java\pool.jar;` Note: Assuming the Progress OpenEdge SDK is installed on the machine, `%DLC%` is the directory where OpenEdge is installed.
Sybase	`C:\sybase\jConnect-7_0\classes\jconn4.jar`

- Changing the CLASSPATH variable may affect the behavior of Java applications on your machine. To understand possible implications before you proceed, refer to the Java documentation.
- Environment variables can be user or system. To change system environment variables, you need administrative rights on the operating system.
- After you change the environment variable, restart any running programs for settings to take effect. Alternatively, log off or restart your operating system.

To configure the CLASSPATH on Windows 7:

1. Open the **Start** menu and right-click **Computer**.
2. Click **Properties**.
3. Click **Advanced system settings**.
4. In the **Advanced** tab, click **Environment Variables**,
5. Locate the CLASSPATH variable under user or system environment variables, and then click Edit. If the CLASSPATH variable does not exist, click **New** to create it.
6. Edit the variable value to include the path on your operating system where the JDBC driver is located. To separate the JDBC driver path from other paths that may already be in the CLASSPATH variable, use the semi-colon separator (;).

To configure the CLASSPATH on Windows 8, Windows 10:

1. Right-click the Windows Start button, and then click **System**.
2. Click **Advanced System Settings**.
3. Click **Environment Variables**.
4. Locate the CLASSPATH variable under user or system environment variables, and then click **Edit**. If the CLASSPATH variable does not exist, click **New** to create it.
5. Edit the variable value to include the path on your operating system where the JDBC driver is located. To separate the JDBC driver path from other paths that may already be in the CLASSPATH variable, use the semi-colon separator (;).

5.2.7   Setting up a PostgreSQL Connection

Connections to PostgreSQL databases can be set up either as native connections, or connections via ODBC, JDBC, and other drivers. The advantage of setting up a native connection is that no drivers are required to be installed on your system.

If, for any reason, you prefer to establish a connection by means of a driver, see the Database Drivers Overview. For information about connections through JDBC, see Setting up a JDBC Connection. For an example of connecting to PostgreSQL through ODBC, see Connecting to PostgreSQL (ODBC).

Otherwise, if you want to set up a native connection to PostgreSQL, follow the steps below. To proceed, you need the following prerequisites: host name, port, database name, username, and password.

To set up a native PostgreSQL connection:

1. Start the database connection wizard.
2. Click **PostgreSQL Connections**.
3. Enter the host (*localhost*, if PostgreSQL runs on the same machine), port (typically 5432, this is optional), the database name, username, and password in the corresponding boxes.

Connect to Postgre

Please enter required parameters for Postgre database. Then click on next to connect to the database.

Host:	DBSERV
Port:	5432 [optional]
Database:	zoo
Username:	dbuser
Password:	••••••••

[< Back]   [Connect]   [Close]

4.   Click **Connect**.

If the PostgreSQL database server is on a different machine, note the following:

- The PostgreSQL database server must be configured to accept connections from clients. Specifically, the **pg_hba.conf** file must be configured to allow non-local connections. Secondly, the **postgresql.conf** file must be configured to listen on specified IP address(es) and port. For more information, check the PostgreSQL documentation (https://www.postgresql.org/docs/9.5/static/client-authentication-problems.html).
- The server machine must be configured to accept connections on the designated port (typically, 5432) through the firewall. For example, on a database server running on Windows, a rule may need to be created to allow connections on port 5432 through the firewall, from **Control Panel > Windows Firewall > Advanced Settings > Inbound Rules**.

5.2.8   Setting up a SQLite Connection

SQLite (http://www.sqlite.org) is a file-based, self-contained database type, which makes it ideal in scenarios where portability and ease of configuration is important. Since SQLite databases are

natively supported by DatabaseSpy, you do not need to install any drivers to connect to them.

Connecting to an Existing SQLite Database

To connect to an existing SQLite database:

1. Run the database connection wizard (see Starting the Database Connection Wizard).
2. Select **SQLite**, and then click **Next**.
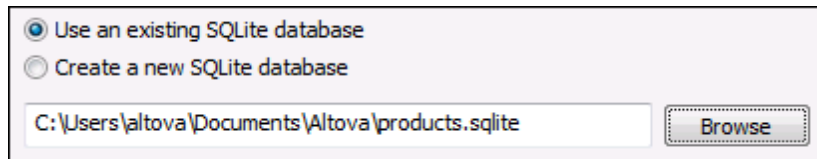
3. Select **Use an existing SQLite database**, and then browse for the SQLite database file, or enter the path (either relative or absolute) to the database. The **Connect** button becomes enabled once you enter the path to a SQLite database file.
4. Click **Connect**.

Creating a New SQLite Database

You can create a new SQLite database file and connect to it, as an alternative to connecting to an existing database file. The database file created by DatabaseSpy is empty; use queries or scripts to create the required database structure and populate it with data.

To create a new SQLite database:

1. Run the database connection wizard (see Starting the Database Connection Wizard).
2. Select **SQLite**, and then click **Next**.
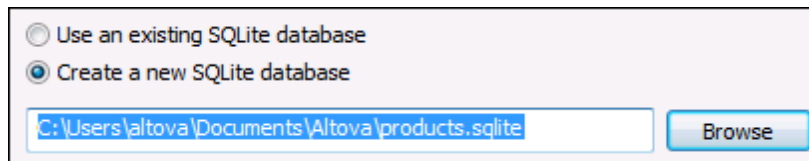
3. Select **Create a new SQLite database**, and then enter the path (either relative or absolute) of the database file to be created (for example, **c:\users\public \products.sqlite**). Alternatively, click **Browse** to select a folder, type the name of the database file in the "File name" text box (for example, **products.sqlite**), and click **Save**.

> Make sure that you have write permissions to the folder where you want to create the database file.

4. Click **Connect**.

5.2.9   Using a Connection from Global Resources

If you have previously configured a database connection to be available as a global resource, you can reuse the connection at any time (even across different Altova applications).

To use a database connection from Global Resources:

1. Start the database connection wizard.
2. Click **Global Resources**. Any database connections available as global resources are listed.
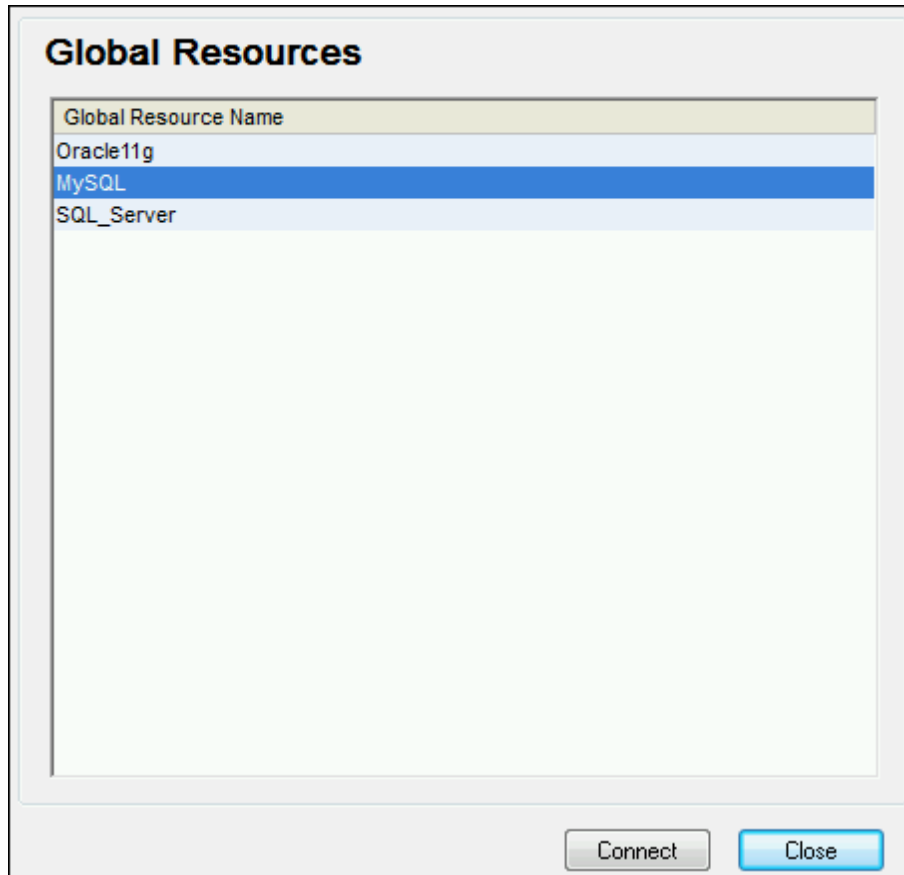
3. Select the database connection record, and click **Connect**.

Tip: To get additional information about each global resource, move the mouse cursor over the global resource.

5.2.10 Database Connection Examples

This section includes sample procedures for connecting to a database from DatabaseSpy. Note that your Windows machine, the network environment, and the database client or server software is likely to have a configuration that is not exactly the same as the one presented in the following examples.

Note: For most database types, it is possible to connect using more than one data access technology (ADO, ODBC, JDBC) or driver. The performance of the database connection, as well as its features and limitations will depend on the selected driver, database client software (if applicable), and any additional connectivity parameters that you may have configured outside DatabaseSpy.

Connecting to Firebird (ODBC)

This topic provides sample instructions for connecting to a Firebird 2.5.4 database running on a Linux server.

Prerequisites:

- The Firebird database server is configured to accept TCP/IP connections from clients.
- The Firebird ODBC driver must be installed on your operating system. This example uses the Firebird ODBC driver version 2.0.3.154 downloaded from the Firebird website (http://www.firebirdsql.org/).
- The Firebird client must be installed on your operating system. Note that there is no standalone installer available for the Firebird 2.5.4 client; the client is part of the Firebird server installation package. You can download the Firebird server installation package from the Firebird website (http://www.firebirdsql.org/), look for "Windows executable installer for full Superclassic/Classic or Superserver". To install only the client files, choose "**Minimum client install - no server, no tools**" when going through the wizard steps.

> **Important:**
>
> - The platform of both the Firebird ODBC driver and client (32-bit or 64-bit) must correspond to that of DatabaseSpy.
> - The version of the Firebird client must correspond to the version of Firebird server to which you are connecting.

- You have the following database connection details: server host name or IP address, database path (or alias) on the server, user name, and password.

To connect to Firebird via ODBC:

1. Start the database connection wizard.
2. Click **ODBC Connections**.
3. Select **User DSN** (or **System DSN**, if you have administrative privileges), and then click **Add** 🔁 .
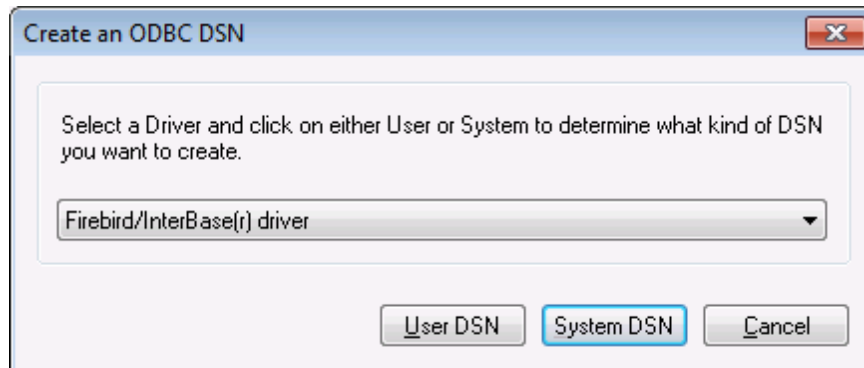
4. Select the Firebird driver, and then click **User DSN** (or **System DSN**, depending on what you selected in the previous step). If the Firebird driver is not available in the list, make sure that it is installed on your operating system (see also Viewing the Available ODBC

Drivers).
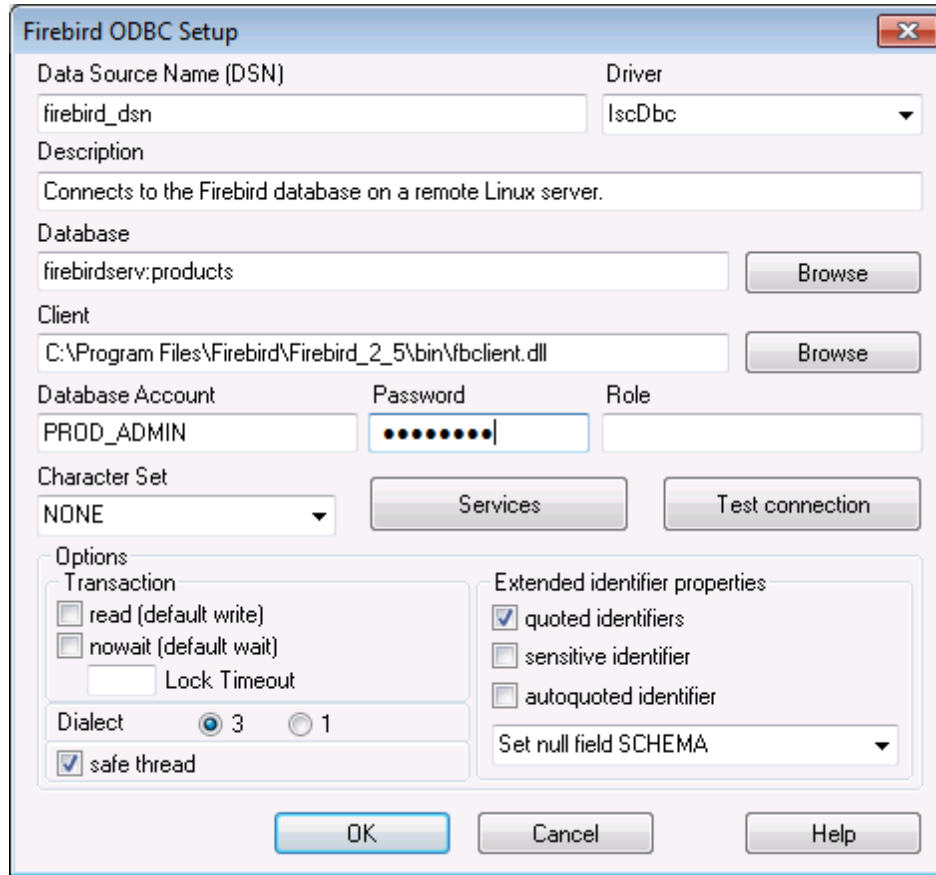
5. Enter the database connection details as follows:

Data Source Name (DSN)	Enter a descriptive name for the data source you are creating.

Database	Enter the server host name or IP address, followed by a colon, followed by the database alias (or path). In this example, the host name is `firebirdserv`, and the database alias is `products`, as follows:
	
	`firebirdserv:products`
	
	Using a database alias assumes that, on the server side, the database administrator has configured the alias *products* to point to the actual Firebird (.fdb) database file on the server (see the Firebird documentation for more details).
	
	You can also use the server IP address instead of the host name, and a path instead of an alias; therefore, any of the following sample connection strings are valid:
	
	`firebirdserver:/var/Firebird/databases/`

	butterflies.fdb 127.0.0.1:D:\Misc\Lenders.fdb If the database is on the local Windows machine, click **Browse** and select the Firebird (.fdb) database file directly.
Client	Enter the path to the **fbclient.dll** file. By default, this is the `bin` subdirectory of the Firebird installation directory.
Database Account	Enter the database user name supplied by the database administrator (in this example, `PROD_ADMIN`).
Password	Enter the database password supplied by the database administrator.

6. Click **OK**.

Connecting to Firebird (JDBC)

This topic provides sample instructions for connecting to a Firebird database server through JDBC.

Prerequisites:

- Java Runtime Environment (JRE) or Java Development Kit (JDK) must be installed on your operating system.
- The operating system's `PATH` environment variable must include the path to the `bin` directory of the JRE or JDK installation directory, for example `C:\Program Files (x86) \Java\jre1.8.0_51\bin`.
- The Firebird JDBC driver must be available on your operating system (it takes the form of a .jar file which provides connectivity to the database). The driver can be downloaded from the Firebird website (http://www.firebirdsql.org/). This example uses *Jaybird 2.2.8*.
- You have the following database connection details: host, database path or alias, username, and password.

To connect to Firebird through JDBC:

1. Start the database connection wizard.
2. Click **JDBC Connections**.
3. Next to "Classpaths", enter the path to the .jar file which provides connectivity to the database. If necessary, you can also enter a semicolon-separated list of .jar file paths. In this example, the required .jar file is located at the following path: **C:\jdbc\firebird \jaybird-full-2.2.8.jar**. Note that you can leave the "Classpaths" text box empty if you have added the .jar file path(s) to the CLASSPATH environment variable of the operating system (see also Configuring the CLASSPATH).
4. In the "Driver" box, select **org.firebirdsql.jdbc.FBDriver**. Note that this entry is available if a valid .jar file path is found either in the "Classpath" text box, or in the operating system's CLASSPATH environment variable (see the previous step).

5. Enter the username and password to the database in the corresponding text boxes.
6. Enter the connection string to the database server in the Database URL text box, by replacing the highlighted values with the ones applicable to your database server.

```
jdbc:firebirdsql://<host>[:<port>]/<database path or alias>
```
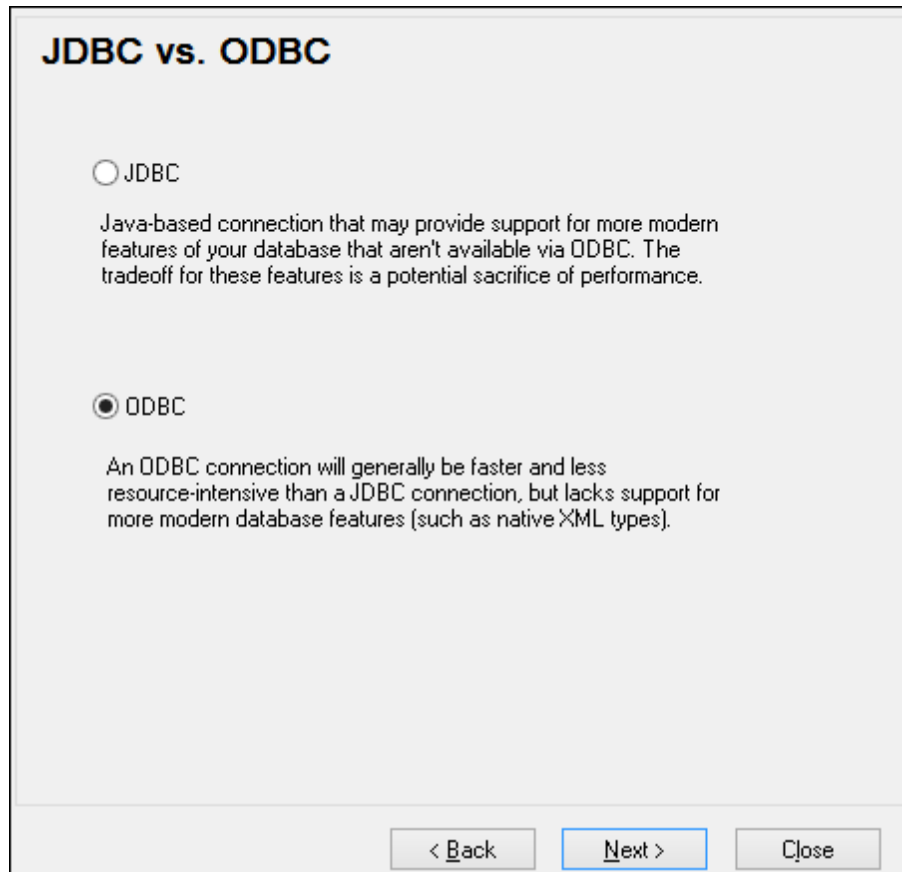
7. Click **Connect**.

**Connecting to IBM DB2 (ODBC)**

This topic provides sample instructions for connecting to an IBM DB2 database through ODBC.

Prerequisites:

- IBM Data Server Client must be installed and configured on your operating system (this example uses IBM Data Server Client 9.7). For installation instructions, check the documentation supplied with your IBM DB2 software. After installing the IBM Data Server Client, check if the ODBC drivers are available on your machine (see Viewing the Available ODBC Drivers).
- Create a database alias. There are several ways to do this:
  - From IBM DB2 Configuration Assistant
  - From IBM DB2 Command Line Processor
  - From the ODBC data source wizard (for this case, the instructions are shown below)
- You have the following database connection details: host, database, port, username, and password.

To connect to IBM DB2:

1. Start the database connection wizard and select **IBM DB2 (ODBC/JDBC)**.
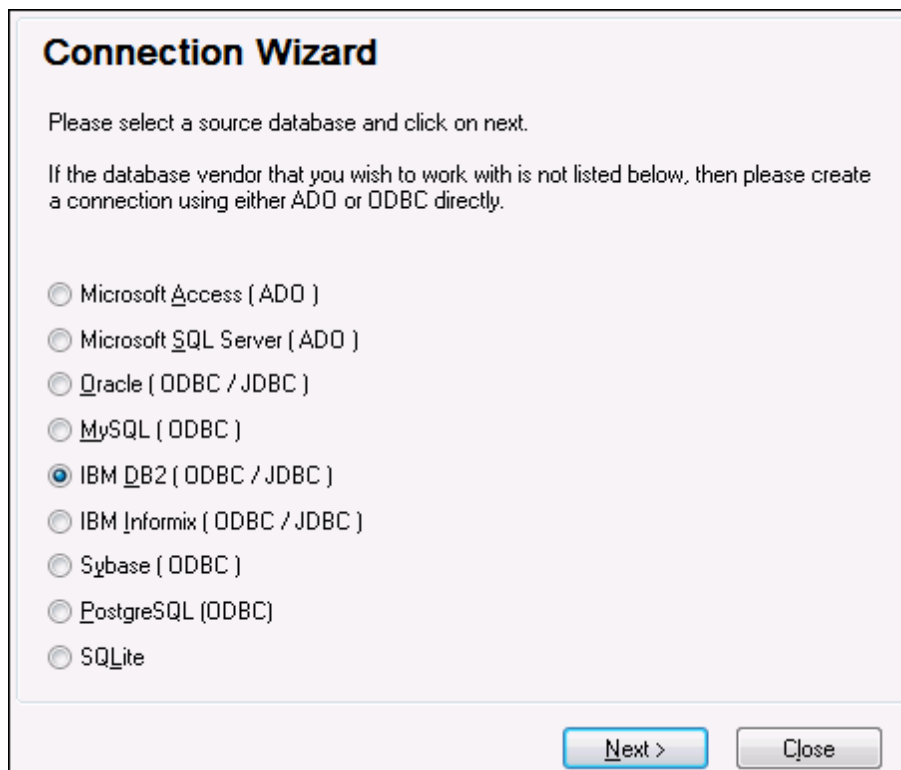
2. Click **Next**.

3. Select **ODBC**, and click **Next**. If prompted to edit the list of known drivers for the database, select the database drivers applicable to IBM DB2 (see Prerequisites), and click **Next**.

Select Drivers:

Edit the list of known drivers for the database.

Driver
☐ SQL Server
☐ Microsoft ODBC for Oracle
☐ Microsoft Access Driver (*.mdb)
☐ Microsoft Access-Treiber (*.mdb)
☐ Driver do Microsoft Access (*.mdb)
☐ Microsoft dBase Driver (*.dbf)
☐ Microsoft dBase-Treiber (*.dbf)
☐ Driver do Microsoft dBase (*.dbf)
☐ Microsoft Excel Driver (*.xls)
☐ Microsoft Excel-Treiber (*.xls)

[< Back]   [Next >]   [Close]
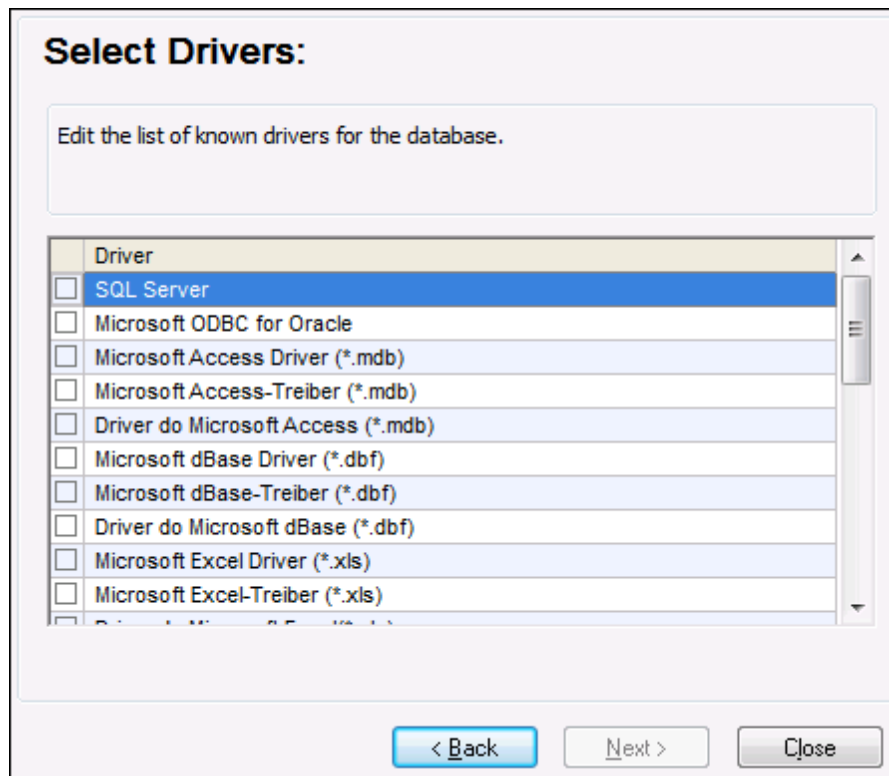
4. Select the IBM DB2 driver from the list, and then click **Connect**. (To edit the list of available drivers, click **Edit Drivers**, and then check or uncheck the IBM DB2 drivers you wish to add or remove, respectively.)

5. Enter a data source name (in this example, **DB2DSN**), and then click **Add**.
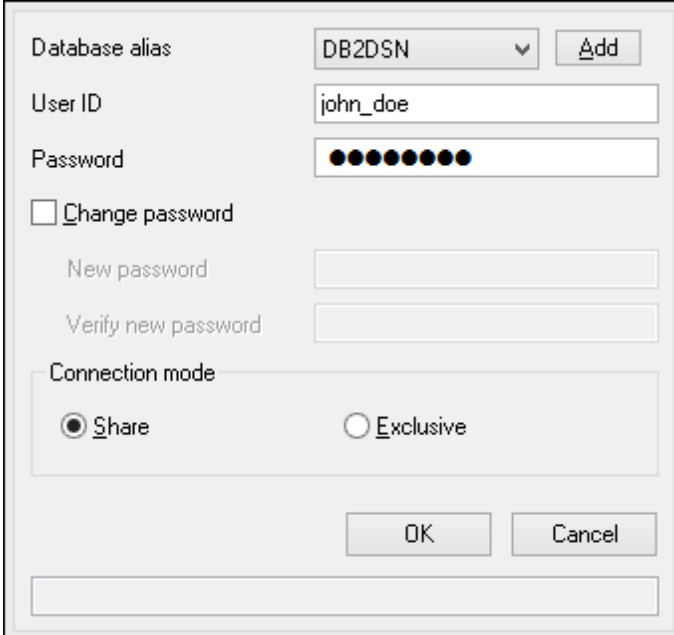
6. On the **Data Source** tab, enter the user name and password to the database.
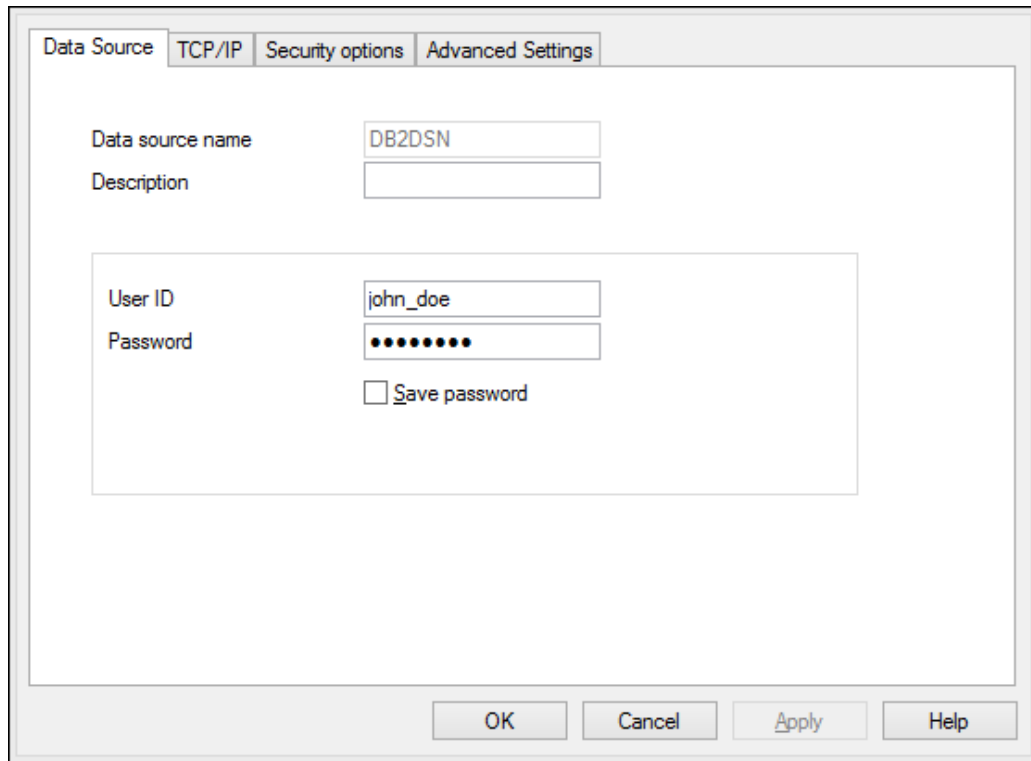
7. On the **TCP/IP** tab, enter the database name, a name for the alias, the host name and the port number, and then click OK.
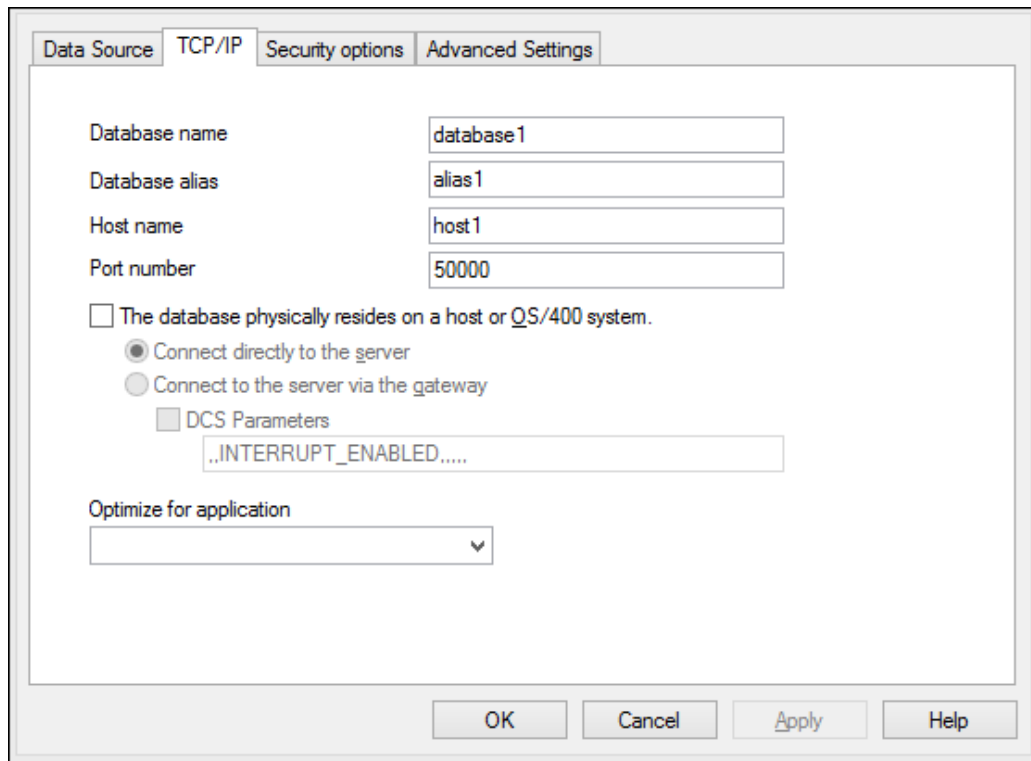
8.   Enter again the username and password, and then click **OK**.

Connecting to IBM DB2 for i (ODBC)

This topic provides sample instructions for connecting to an *IBM DB2 for i* database through ODBC.

Prerequisites:

- *IBM System i Access for Windows* must be installed on your operating system (this example uses *IBM System i Access for Windows V6R1M0*). For installation instructions, check the documentation supplied with your *IBM DB2 for i* software. After installation, check if the ODBC driver is available on your machine (see Viewing the Available ODBC Drivers).
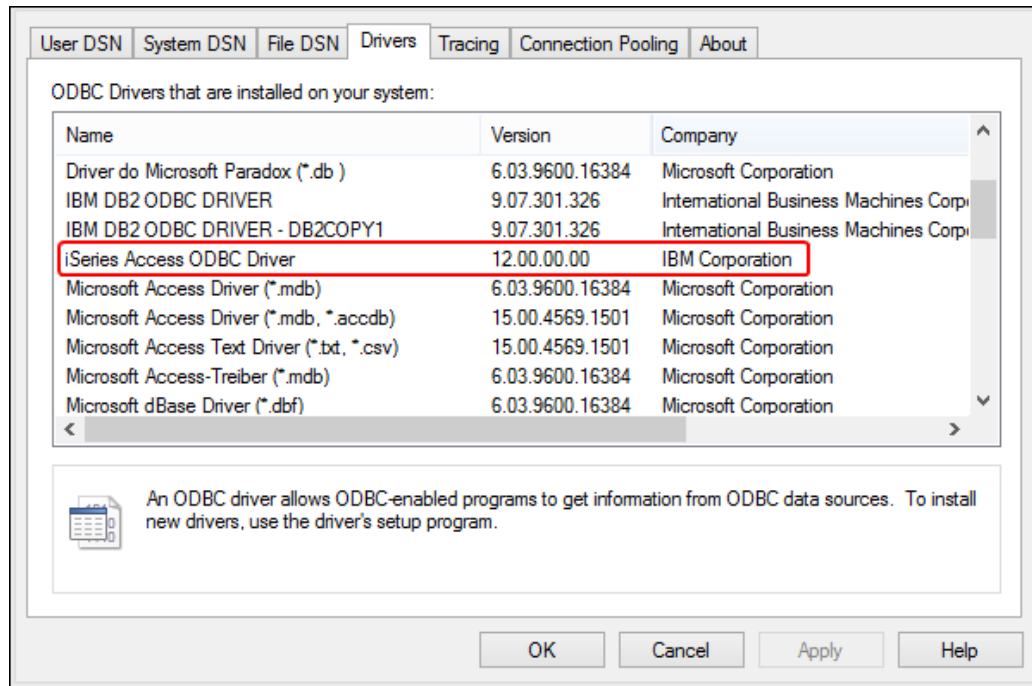
- You have the following database connection details: the I.P. address of the database server, database user name, and password.
- Run *System i Navigator* and follow the wizard to create a new connection. When prompted to specify a system, enter the I.P. address of the database server. After creating the connection, it is recommended to verify it (click on the connection, and select **File** > **Diagnostics** > **Verify Connection**). If you get connectivity errors, contact the database server administrator.
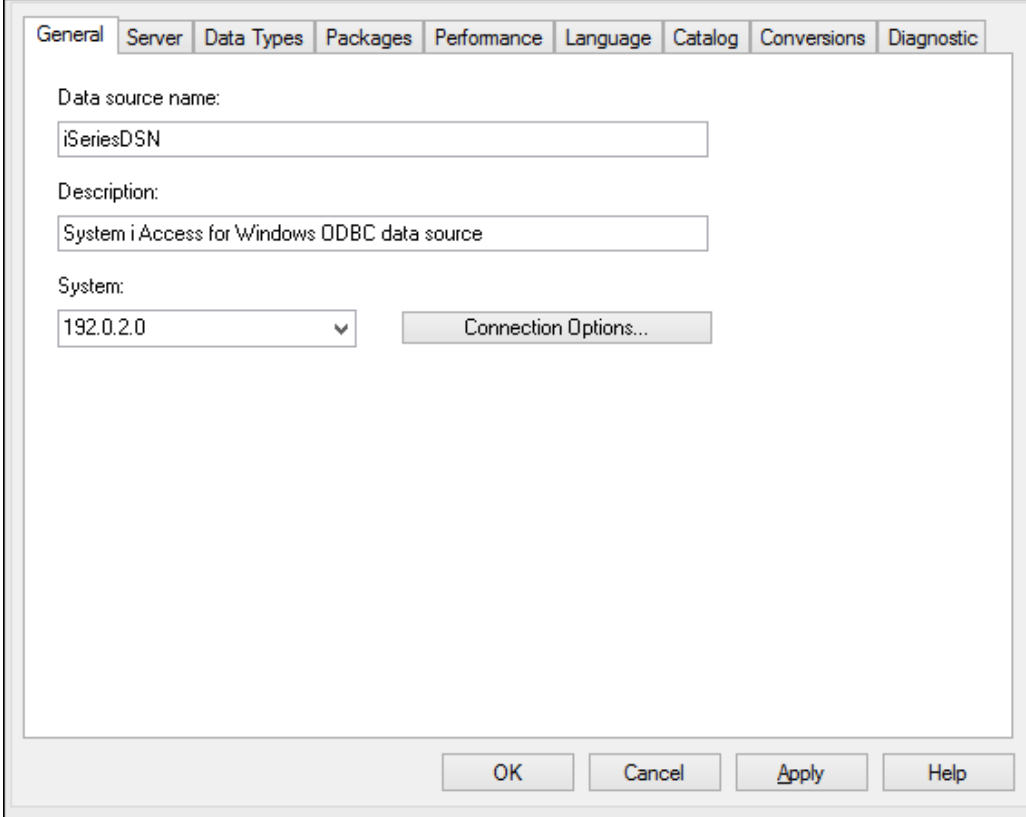
To connect to IBM DB2 for i:

1. [Start the database connection wizard](#).
2. Click **ODBC connections**.
3. Click **User DSN** (alternatively, click **System DSN**, or **File DSN**, in which case the subsequent instructions will be similar).
4. Click **Add** .
5. Select the **iSeries Access ODBC Driver** from the list, and click **User DSN** (or **System DSN**, if applicable).
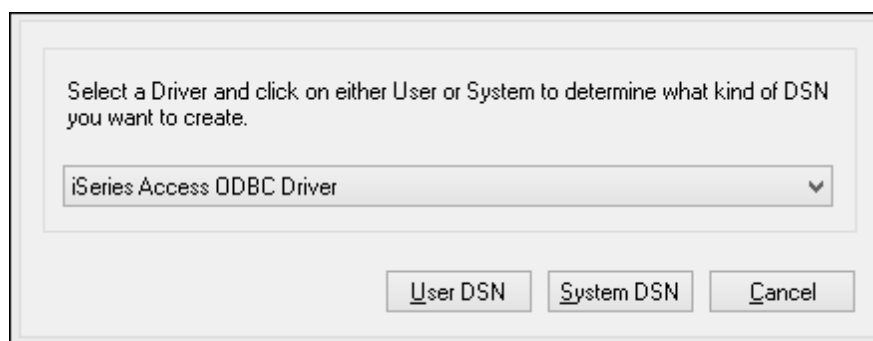
6. Enter a data source name and select the connection from the System combo box. In this

example, the data source name is **iSeriesDSN** and the System is **192.0.2.0**.

7. Click Connection Options, select **Use the User ID specified below** and enter the name
    of the database user (in this example, **DBUSER**).

8. Click **OK**. The new data source becomes available in the list of DSNs.
9. Click **Connect**.
10. Enter the user name and password to the database when prompted, and then click **OK**.

Connecting to IBM Informix (JDBC)

This topic provides sample instructions for connecting to an IBM Informix database server through JDBC.

Prerequisites:

- Java Runtime Environment (JRE) must be installed on your operating system.
- The JDBC driver (one or several .jar files that provide connectivity to the database) must be available on your operating system. In this example, IBM Informix JDBC driver version 3.70 is used. For the driver's installation instructions, see the documentation accompanying the driver or the "IBM Informix JDBC Driver Programmer's Guide").
- You have the following database connection details: host, name of the Informix server, database, port, username, and password.

To connect to IBM Informix through JDBC:

1. Start the database connection wizard.
2. Click **JDBC Connections**.
3. Next to "Classpaths", enter the path to the .jar file which provides connectivity to the database. If necessary, you can also enter a semicolon-separated list of .jar file paths. In this example, the required .jar file is located at the following path: **C:\Informix_JDBC_Driver\lib\ifxjdbc.jar**. Note that you can leave the "Classpaths" text

box empty if you have added the .jar file path(s) to the CLASSPATH environment variable of the operating system (see also Configuring the CLASSPATH).

4. In the "Driver" box, select **com.informix.jdbc.IfxDriver**. Note that this entry is available if a valid .jar file path is found either in the "Classpath" text box, or in the operating system's CLASSPATH environment variable (see the previous step).

Classpaths:	C:\jdbc\Informix_JDBC_Driver\lib\ifxjdbc.jar;
Driver:	com.informix.jdbc.IfxDriver
Username:	dbuser
Password:	•••••••
Database URL:	jdbc:informix-sqli://host:port/MyDatabase:INFORMIXSERVER=MyServerName

Connect   Close

5. Enter the username and password to the database in the corresponding text boxes.
6. Enter the connection string to the database server in the Database URL text box, by replacing the highlighted values with the ones applicable to your database server.

```
jdbc:informix-sqli://hostName:port/
databaseName:INFORMIXSERVER=myserver;
```

7. Click **Connect**.

Connecting to Microsoft Access (ADO)

A simple way to connect to a Microsoft Access database is to follow the wizard and browse for the database file, as shown in Connecting to an Existing Microsoft Access Database. An alternative approach is to set up an ADO connection explicitly, as shown in this topic. This approach is useful if your database is password-protected.

It is also possible to connect to Microsoft Access through an ODBC connection, but there are some limitations in this scenario, so it is best to avoid it.

To connect to a password-protected Microsoft Access database:

1. [Start the database connection wizard](#).
2. Click **ADO Connections**.
3. Click **Build**.
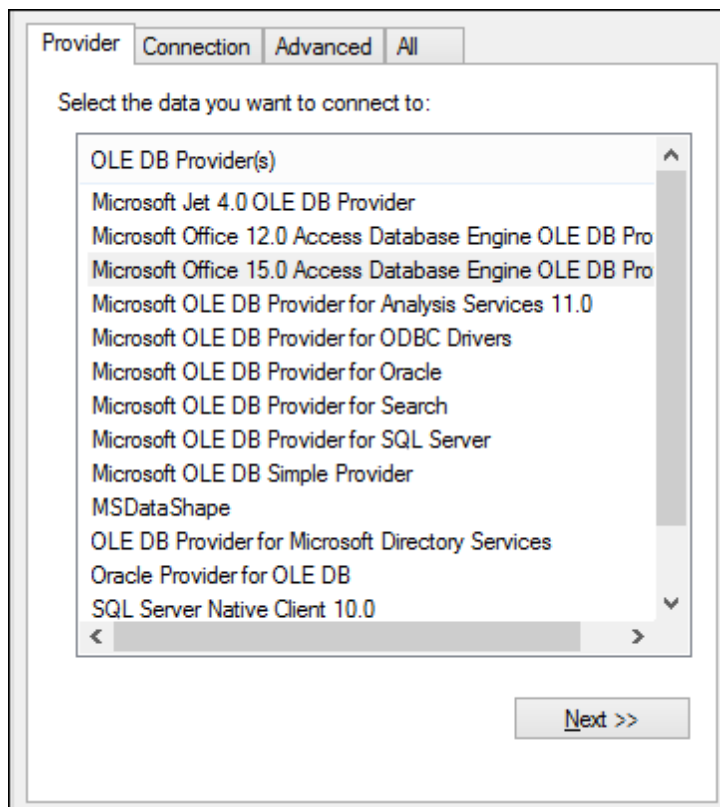
4. Select the **Microsoft Office 15.0 Access Database Engine OLE DB Provider**, and then click **Next**.

5. In the Data Source box, enter the path to the Microsoft Access file. Because the file is on the local network share **U:\Departments\Finance\Reports\Revenue.accdb**, we will convert it to UNC format, and namely **\\server1\\dfs\Departments\Finance\Reports \Revenue.accdb**, where **server1** is the name of the server and **dfs** is the name of the network share.
6. On the **All** tab, double click the **Jet OLEDB:Database Password** property and enter the database password as property value.

Note: If you are still unable to connect, locate the workgroup information file (**System.MDW**) applicable to your user profile (see http://support.microsoft.com/kb/305542 for instructions), and set the value of the **Jet OLEDB: System database** property to the path of the **System.MDW** file.

Connecting to Microsoft SQL Server (ADO)

To connect to SQL Server using the Microsoft OLE DB Provider:

1. [Start the database connection wizard](#).
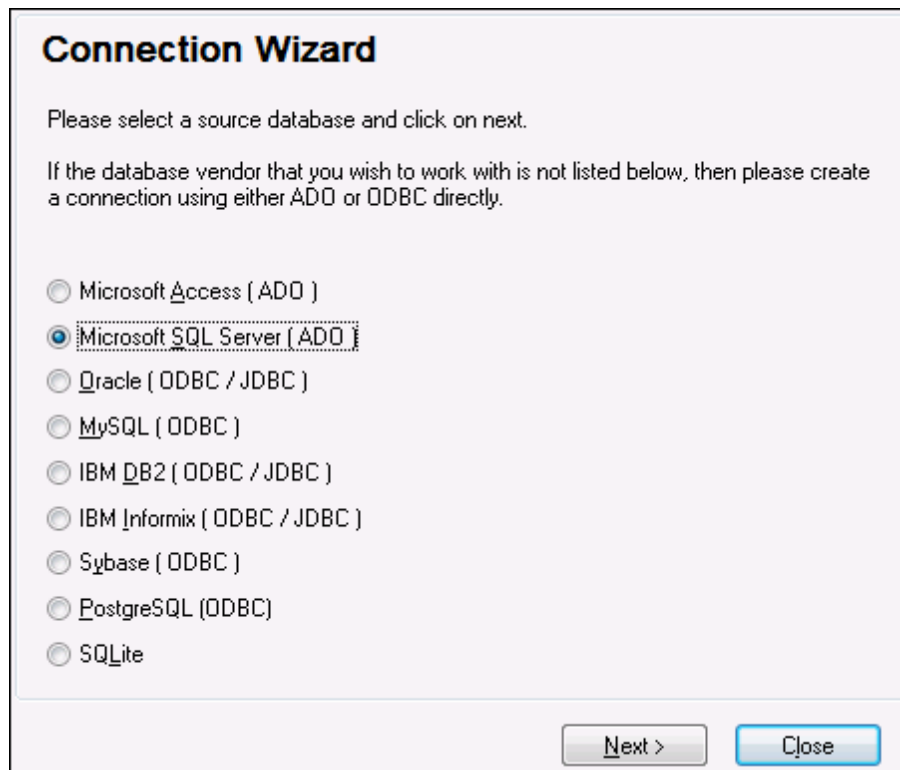
2. Select **Microsoft SQL Server (ADO)**, and then click **Next**. The list of available ADO drivers is displayed.
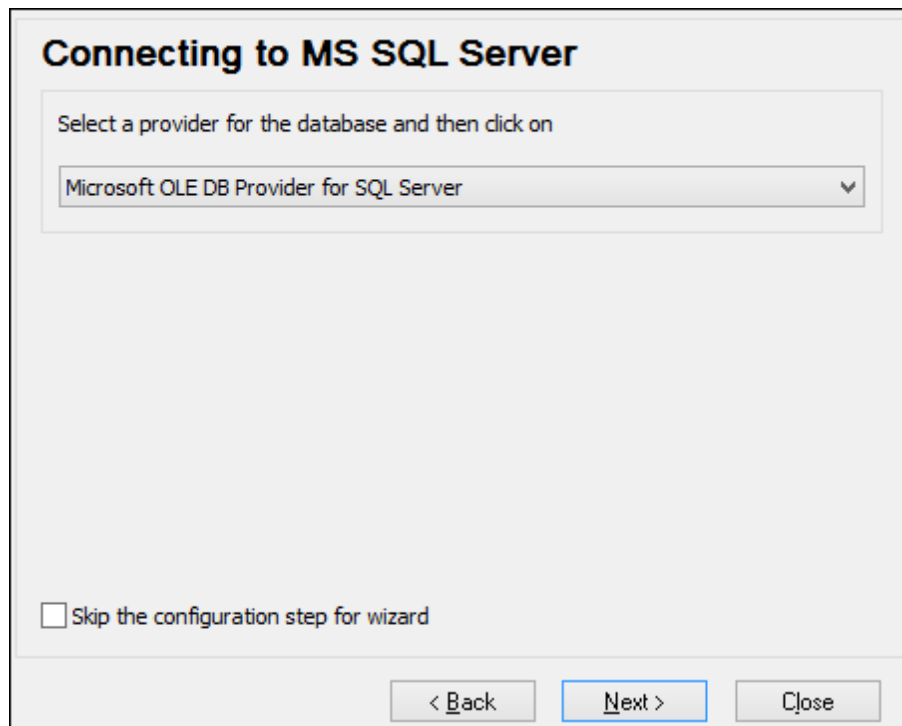
3. Select **Microsoft OLE DB Provider for SQL Server**, and then click **Next**.

4. Select or enter the name of the database server (in this example, **SQLSERV01**). To view the list of all servers on the network, expand the drop-down list.
5. If the database server was configured to allow connections from users authenticated on the Windows domain, select **Use Windows NT integrated security**. Otherwise, select **Use a specific user name and password**, and type them in the relevant boxes.
6. Select the database to which you are connecting (in this example, **NORTHWIND**).
7. To test the connection at this time, click **Test Connection**. This is an optional, recommended step.
8. Do one of the following:
   a. Select the **Allow saving password** check box.
   b. On the **All** tab, change the value of the **Persist Security Info** property to **True**.
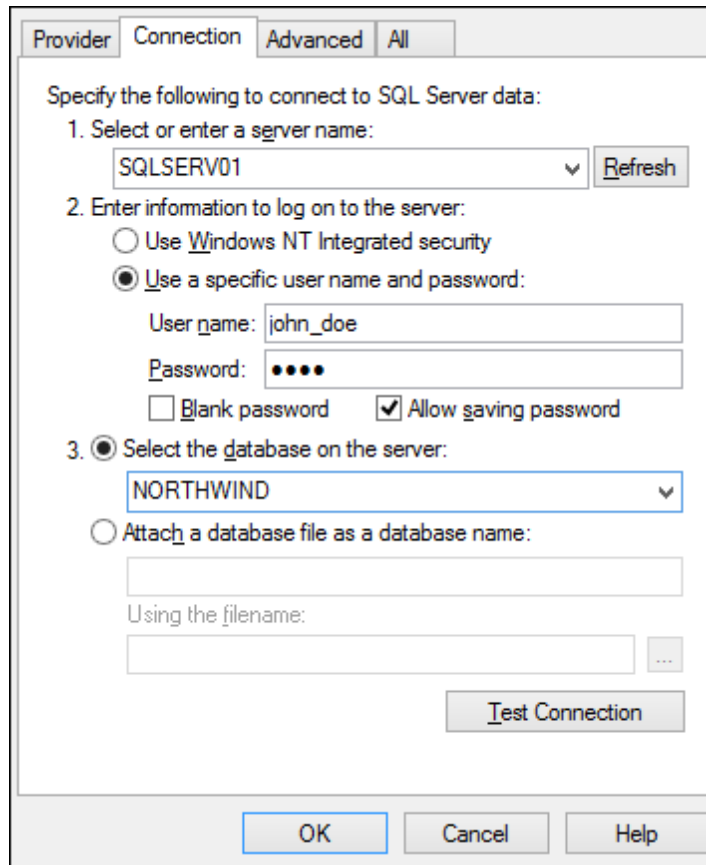
9. Click **OK**.

Connecting to Microsoft SQL Server (ODBC)

To connect to SQL Server using ODBC:

1. .
2. Click **ODBC Connections**.
3. Select **User DSN** (or **System DSN**, if you have administrative privileges), and then click
    Add .
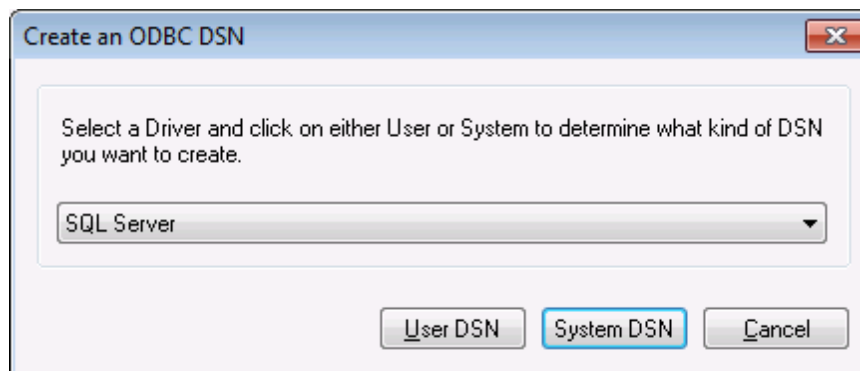
4. Select **SQL Server** (or **SQL Server Native Client**, if available), and then click **User
   DSN** (or **System DSN** if you are creating a System DSN).
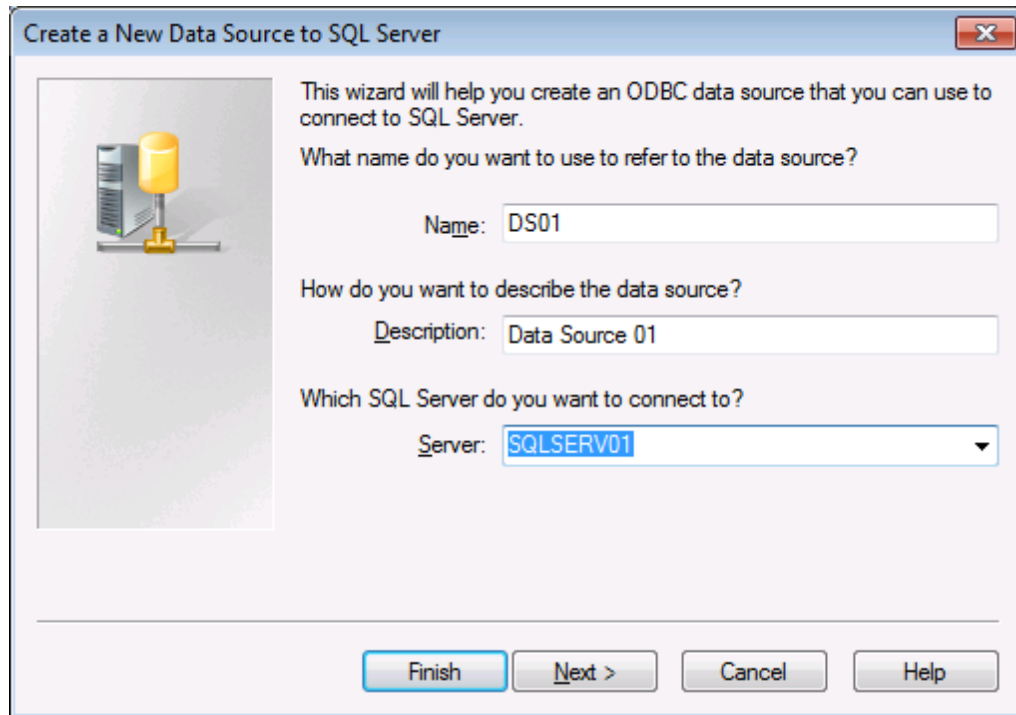
5. Enter a name and description to identify this connection, and then select from the list the SQL Server to which you are connecting (**SQLSERV01** in this example).
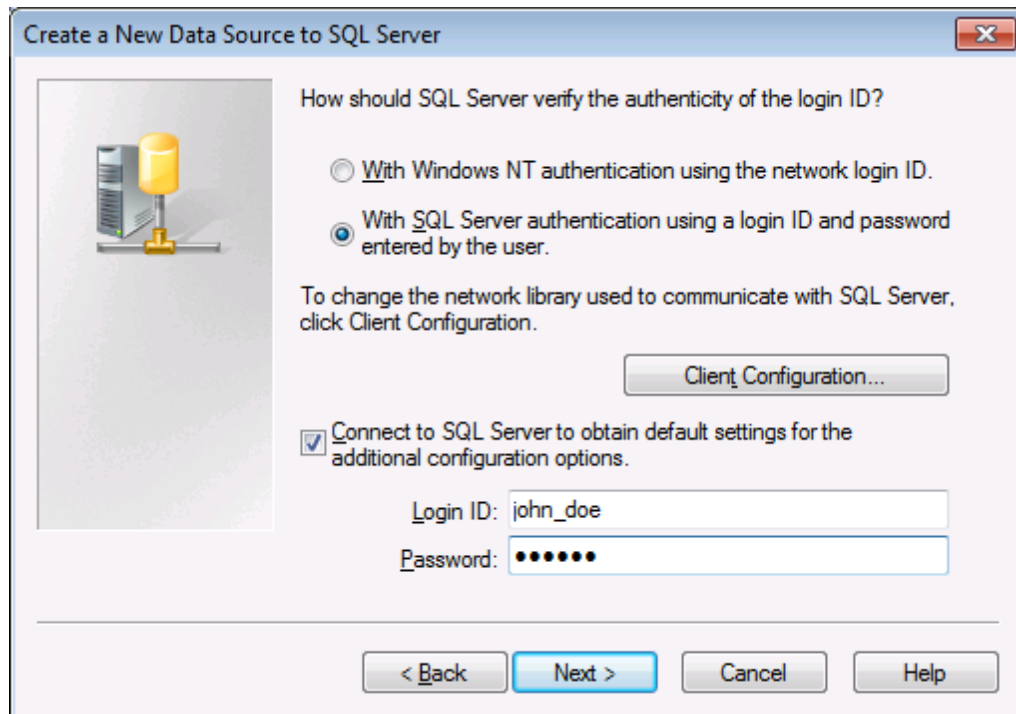
6. If the database server was configured to allow connections from users authenticated on the Windows domain, select **With Windows NT authentication**. Otherwise, select **With SQL Server authentication...** and type the user name and password in the relevant

boxes.
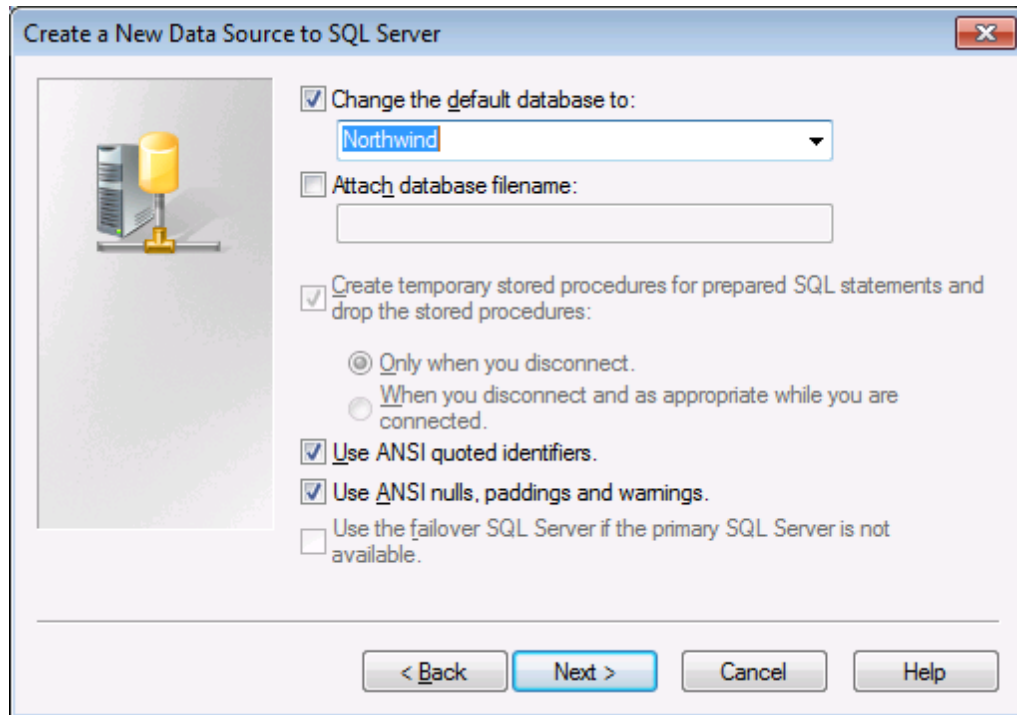
7. Select the name of the database to which you are connecting (in this example, **Northwind**).
8. Click **Finish**.

Connecting to MySQL (ODBC)

This topic provides sample instructions for connecting to a MySQL database server from a Windows machine through the ODBC driver. The MySQL ODBC driver is not available on Windows, so it must be downloaded and installed separately. This example uses MySQL ODBC driver version 5.3.4 downloaded from the official website (see also Database Drivers Overview).

Prerequisites:

*   MySQL ODBC driver must be installed on your operating system (for installation instructions, check the documentation supplied with the driver).
*   You have the following database connection details: host, database, port, username, and password.

To connect to MySQL via ODBC:

1. Start the database connection wizard.

Connection Wizard

Please select a source database and click on next.

If the database vendor that you wish to work with is not listed below, then please create a connection using either ADO or ODBC directly.

- ◉ Microsoft Access (ADO)
- ○ Microsoft SQL Server (ADO)
- ○ Oracle (ODBC / JDBC)
- ○ MySQL (ODBC)
- ○ IBM DB2 (ODBC / JDBC)
- ○ IBM Informix (ODBC / JDBC)
- ○ Sybase (ODBC)
- ○ PostgreSQL (ODBC)
- ○ SQLite

[Next >]   [Close]
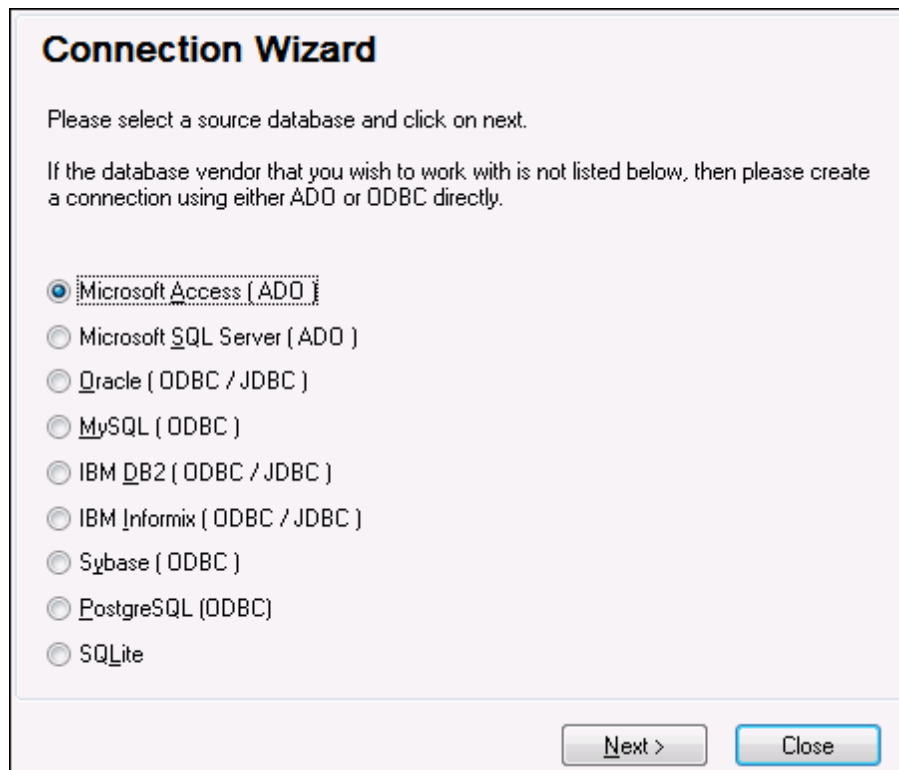
2. Select **MySQL (ODBC)**, and then click **Next**.

3. Select **Create a new Data Source Name (DSN) with the driver**, and select a MySQL driver. If no MySQL driver is available in the list, click **Edit Drivers**, and select any available MySQL drivers (the list contains all ODBC drivers installed on your operating system).
4. Click **Connect**.
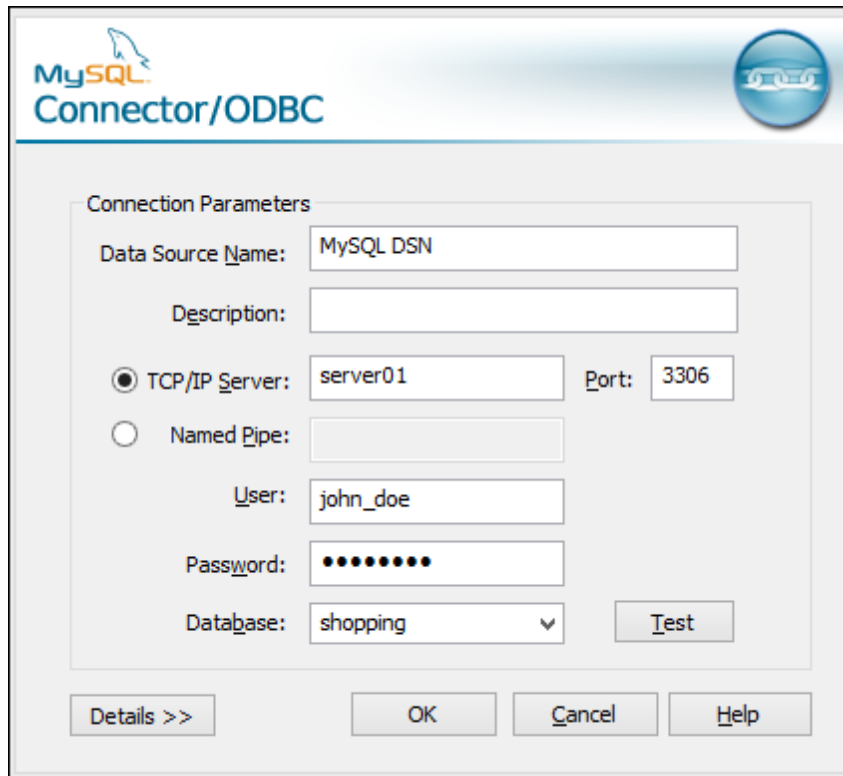
5. In the Data Source Name box, enter a descriptive name that will help you identify this ODBC data source in future.
6. Fill in the database connection credentials (TCP/IP Server, User, Password), select a database, and then click **OK**.

Note: If the database server is remote, it must be configured by the server administrator to accept remote connections from your machine's IP address. Also, if you click **Details>>**, there are several additional parameters available for configuration. Check the driver's documentation before changing their default values.

Connecting to Oracle (ODBC)

This example illustrates a common scenario where you connect from DatabaseSpy to an Oracle database server on a network machine, through an Oracle database client installed on the local operating system.

The example includes instructions for setting up an ODBC data source (DSN) using the database connection wizard in DatabaseSpy. If you have already created a DSN, or if you prefer to create it directly from ODBC Data Source administrator in Windows, you can do so, and then select it when prompted by the wizard. For more information about ODBC data sources, see Setting up an ODBC Connection.

Prerequisites:

*   The Oracle database client (which includes the ODBC Oracle driver) must be installed and configured on your operating system. For instructions on how to install and configure an Oracle database client, refer to the documentation supplied with your Oracle software.
*   The **tnsnames.ora** file located in Oracle home directory contains an entry that describes

the database connection parameters, in a format similar to this:

```
ORCL =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP)(HOST = server01)(PORT = 1521))
    )
    (CONNECT_DATA =
      (SID = orcl)
      (SERVER = DEDICATED)
    )
  )
```

The path to the **tnsnames.ora** file depends on the location where Oracle home directory was installed. For Oracle database client 11.2.0, the default Oracle home directory path could be as follows:

```
C:\app\username\product\11.2.0\client_1\network\admin\tnsnames.ora
```

You can add new entries to the **tnsnames.ora** file either by pasting the connection details and saving the file, or by running the Oracle *Net Configuration Assistant* wizard (if available).

To connect to Oracle using ODBC:

1. [Start the database connection wizard](#).
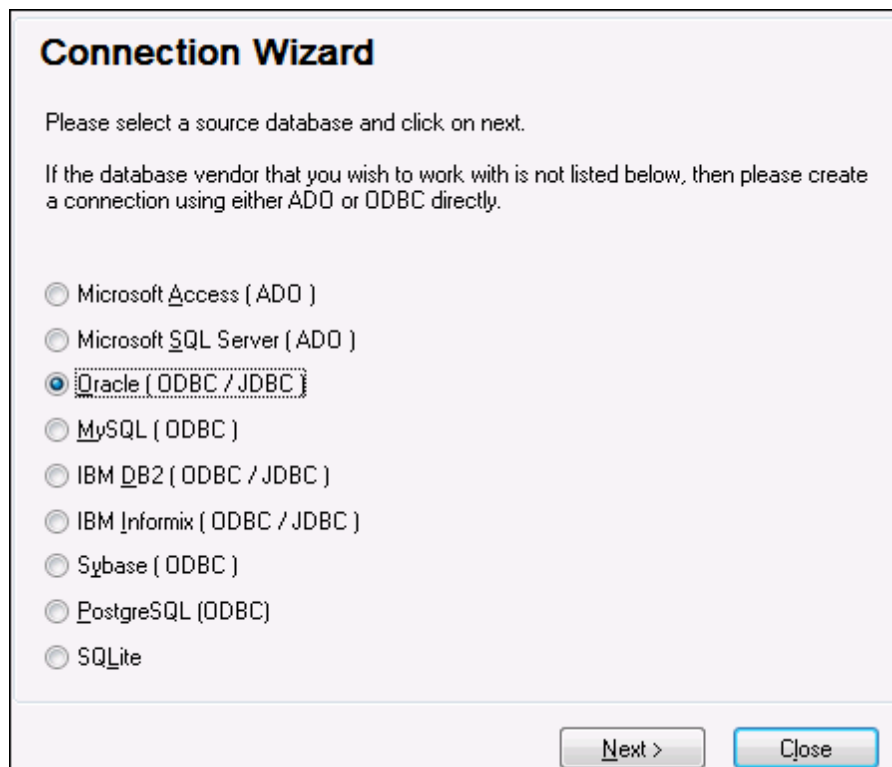
2. Select **Oracle (ODBC / JDBC)**, and then click **Next**.
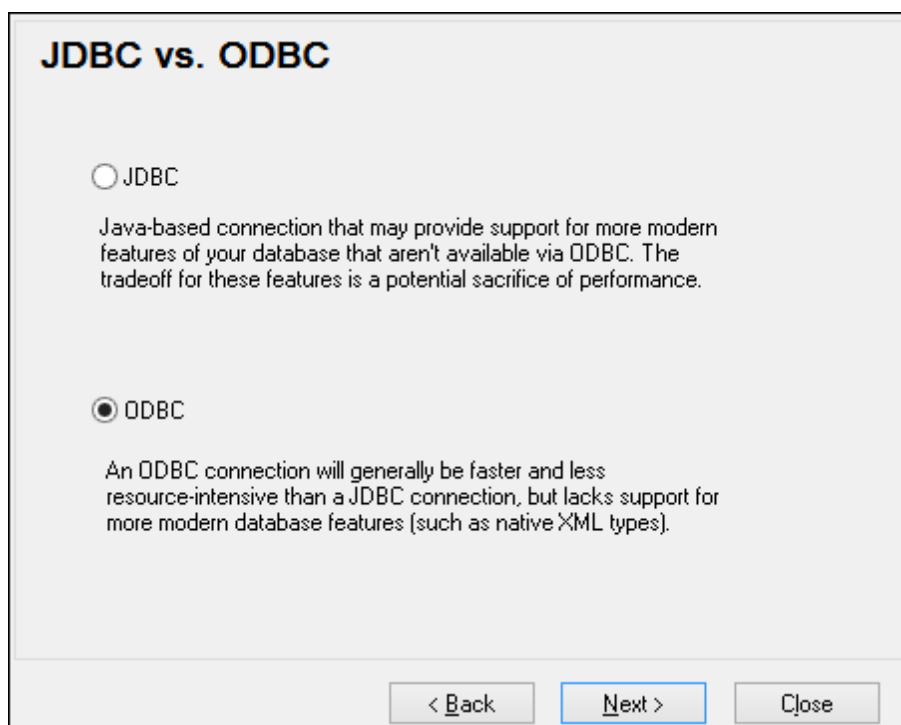
3.   Select **ODBC**.
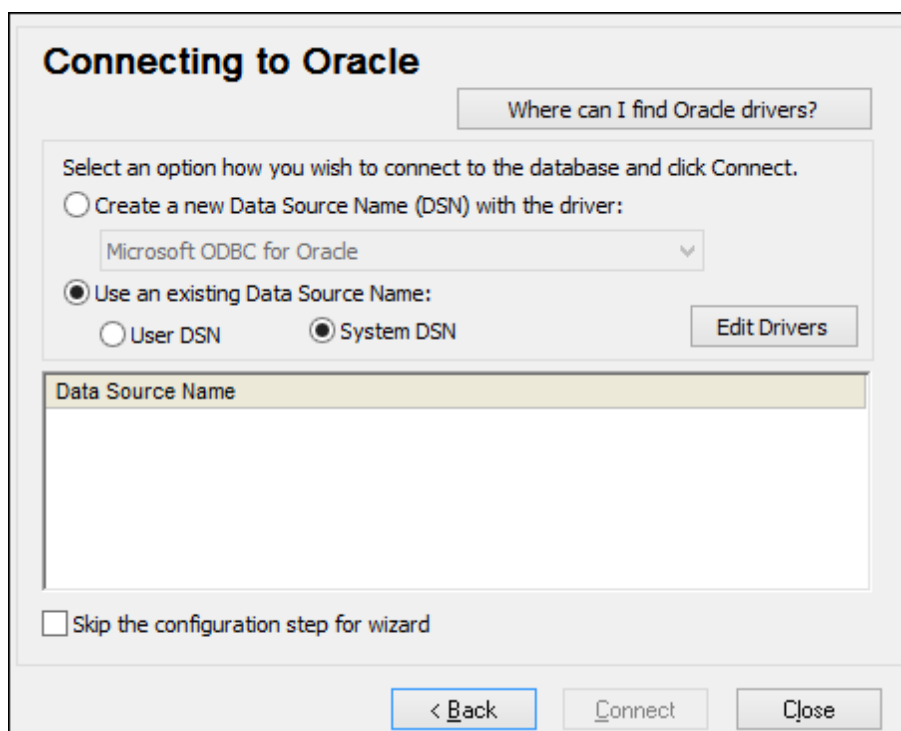
4.   Click **Edit Drivers**.
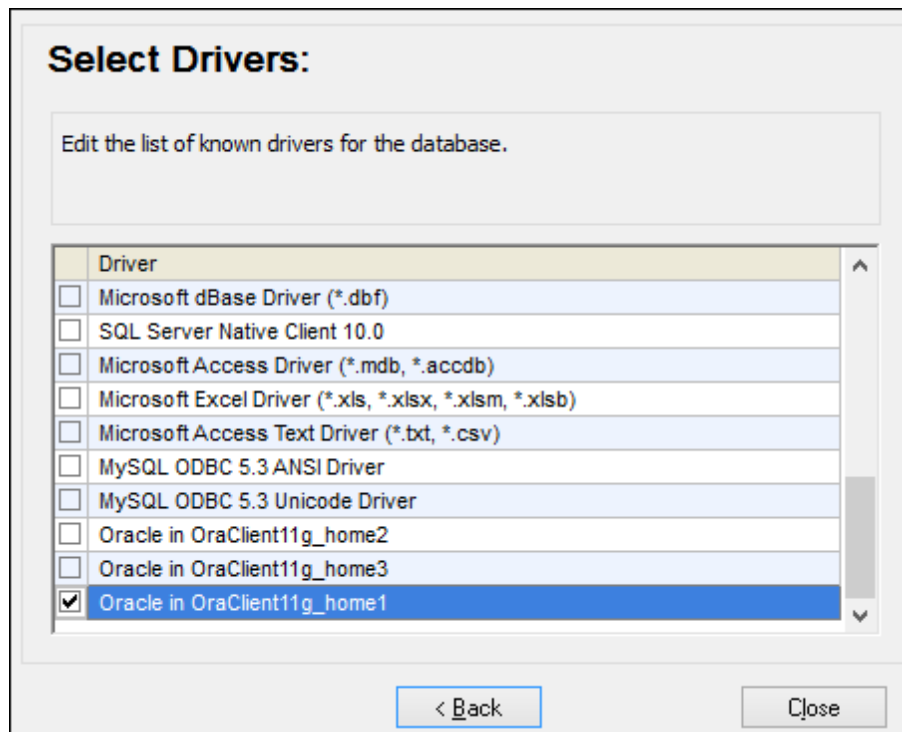
5. Select the Oracle drivers you wish to use (in this example, **Oracle in OraClient11g_home1**). The list displays the Oracle drivers available on your system after installation of Oracle client.
6. Click **Back**.
7. Select **Create a new data source name (DSN) with the driver**, and then select the Oracle driver chosen in step 4.

Avoid using the Microsoft-supplied driver called **Microsoft ODBC for Oracle** driver. Microsoft recommends using the ODBC driver provided by Oracle (see http://msdn.microsoft.com/en-us/library/ms714756%28v=vs.85%29.aspx)

8. Click **Connect**.

9. In the Data Source Name text box, enter a name to identify the data source (in this example, **Oracle DSN 1**).
10. In the TNS Service Name box, enter the connection name as it is defined in the **tnsnames.ora** file (see prerequisites). In this example, the connection name is **ORCL**.
11. Click **OK**.

12. Enter the username and password to the database, and then click OK.
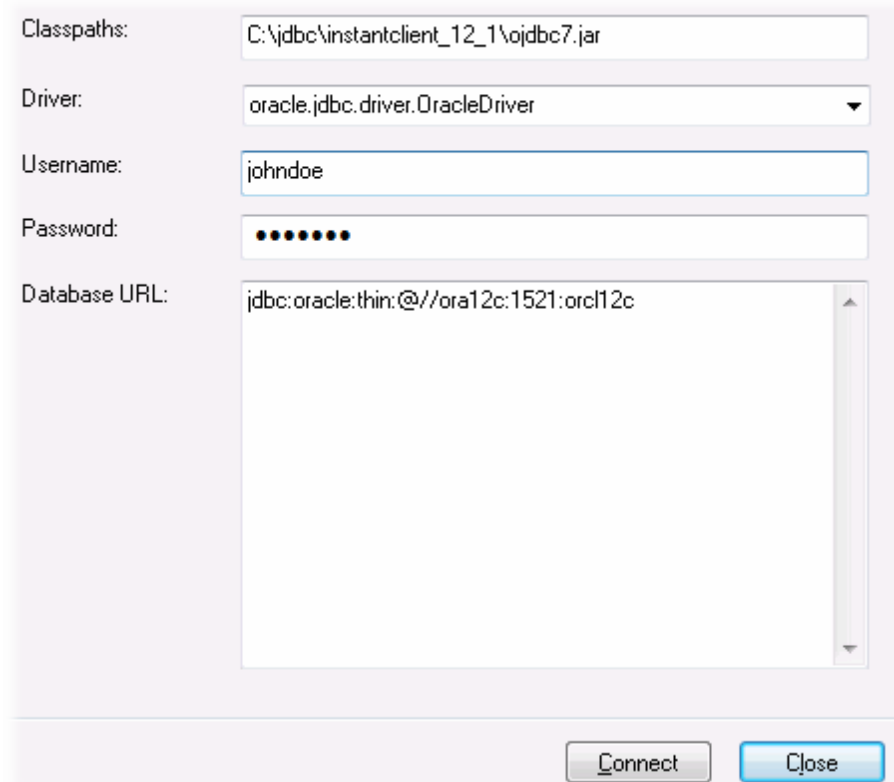
Connecting to Oracle (JDBC)

This example shows you how to connect to an Oracle database server from a client machine, using the JDBC interface. The connection is created as a pure Java connection, using the **Oracle Instant Client Package (Basic)** available from the Oracle website. The advantage of this connection type is that it requires only the Java environment and the .jar libraries supplied by the Oracle Instant Client Package, saving you the effort to install and configure a more complex database client.

Prerequisites:

- Java Runtime Environment (JRE) or Java Development Kit (JDK) must be installed on your operating system.
- The operating system's `PATH` environment variable must include the path to the `bin` directory of the JRE or JDK installation directory, for example `C:\Program Files (x86)\Java\jre1.8.0_51\bin`.
- The **Oracle Instant Client Package (Basic)** must be available on your operating system. The package can be downloaded from the official Oracle website. This example uses Oracle Instant Client Package version 12.1.0.2.0, for Windows 32-bit.
- You have the following database connection details: host, port, service name, username, and password.

To connect to Oracle through the Instant Client Package:

1. Start the database connection wizard.
2. Click **JDBC Connections**.
3. Next to "Classpaths", enter the path to the .jar file which provides connectivity to the database. If necessary, you can also enter a semicolon-separated list of .jar file paths. In this example, the required .jar file is located at the following path: **C:\jdbc \instantclient_12_1\odbc7.jar**. Note that you can leave the "Classpaths" text box empty if you have added the .jar file path(s) to the CLASSPATH environment variable of the operating system (see also Configuring the CLASSPATH).
4. In the "Driver" box, select either **oracle.jdbc.OracleDriver** or **oracle.jdbc.driver.OracleDriver**. Note that these entries are available if a valid .jar file path is found either in the "Classpath" text box, or in the operating system's CLASSPATH environment variable (see the previous step).
5. Enter the username and password to the database in the corresponding text boxes.

6.   Enter the connection string to the database server in the Database URL text box, by replacing the highlighted values with the ones applicable to your database server.

```
jdbc:oracle:thin:@//host:port:service
```

7.   Click **Connect**.

Connecting to PostgreSQL (ODBC)

This topic provides sample instructions for connecting to a PostgreSQL database server from a Windows machine through the ODBC driver. The PostgreSQL ODBC driver is not available on Windows, so it must be downloaded and installed separately. This example uses the psqlODBC driver (version 09_03_300-1) downloaded from the official website (see also Database Drivers Overview).

Note: You can also connect to a PostgreSQL database server directly (without the ODBC driver), see Setting up a PostgreSQL Connection.

Prerequisites:

- *psqlODBC* driver must be installed on your operating system (for installation instructions, check the documentation supplied with the driver).
- You have the following database connection details: server, port, database, user name, and password.

To connect to PostgreSQL using ODBC:

1. [Start the database connection wizard](#).
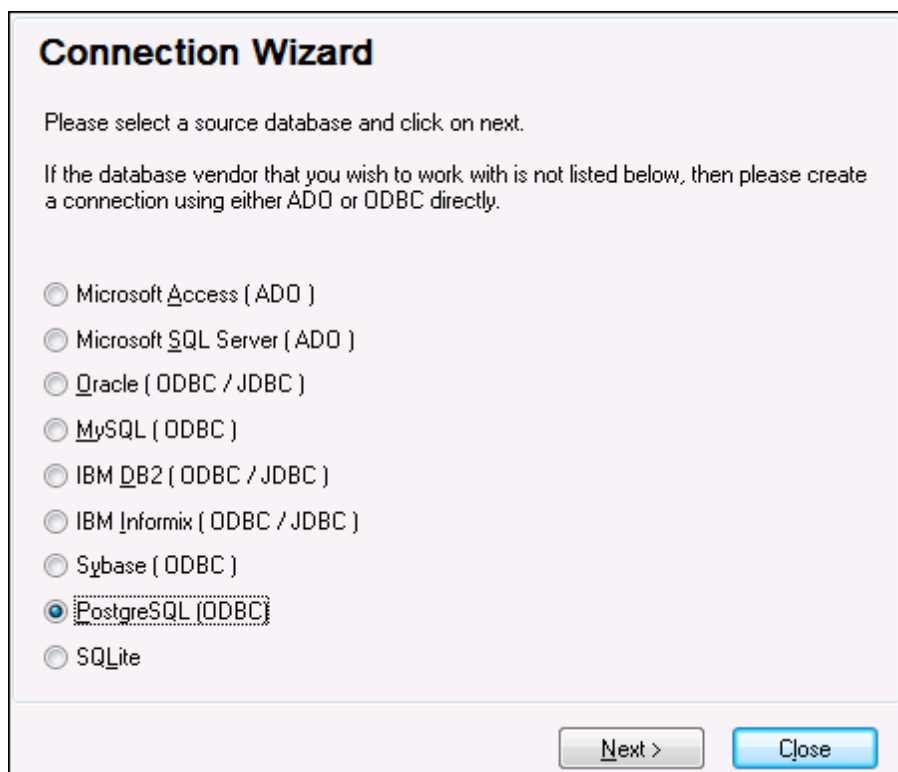
2. Select **PostgreSQL (ODBC)**, and then click **Next**.

3. Select **Create a new Data Source Name (DSN) with the driver**, and select the
    PostgreSQL driver. If no PostgreSQL driver is available in the list, click **Edit Drivers**, and
    select any available PostgreSQL drivers (the list contains all ODBC drivers installed on
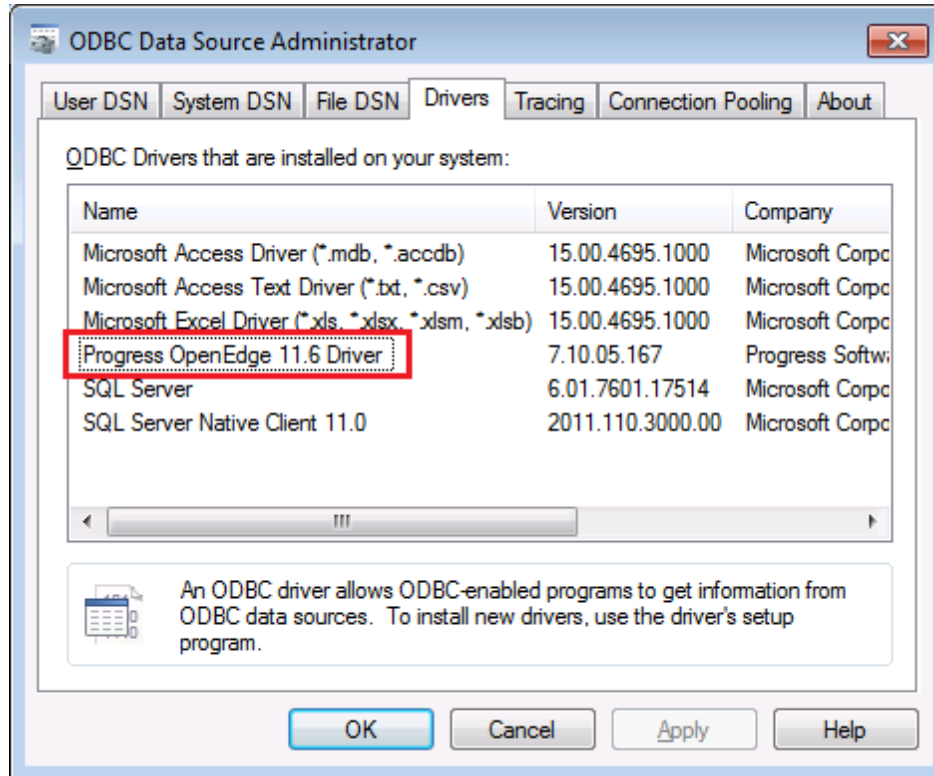   your operating system).
4. Click **Connect**.

5. Fill in the database connection credentials (Database, Server, Port, User Name,
    Password), and then click **OK**.

Connecting to Progress OpenEdge (ODBC)

This topic provides sample instructions for connecting to a Progress OpenEdge database server
through the Progress OpenEdge 11.6 ODBC driver.

Prerequisites

- The *ODBC Connector for Progress OpenEdge* driver must be installed on your operating system. The Progress OpenEdge ODBC driver can be downloaded from the vendor's website (see also Database Drivers Overview). Make sure to download the 32-bit driver when running the 32-bit version of DatabaseSpy, and the 64-bit driver when running the 64-bit version. After installation, check if the ODBC driver is available on your machine (see also Viewing the Available ODBC Drivers).


- You have the following database connection details: host name, port number, database name, user ID, and password.

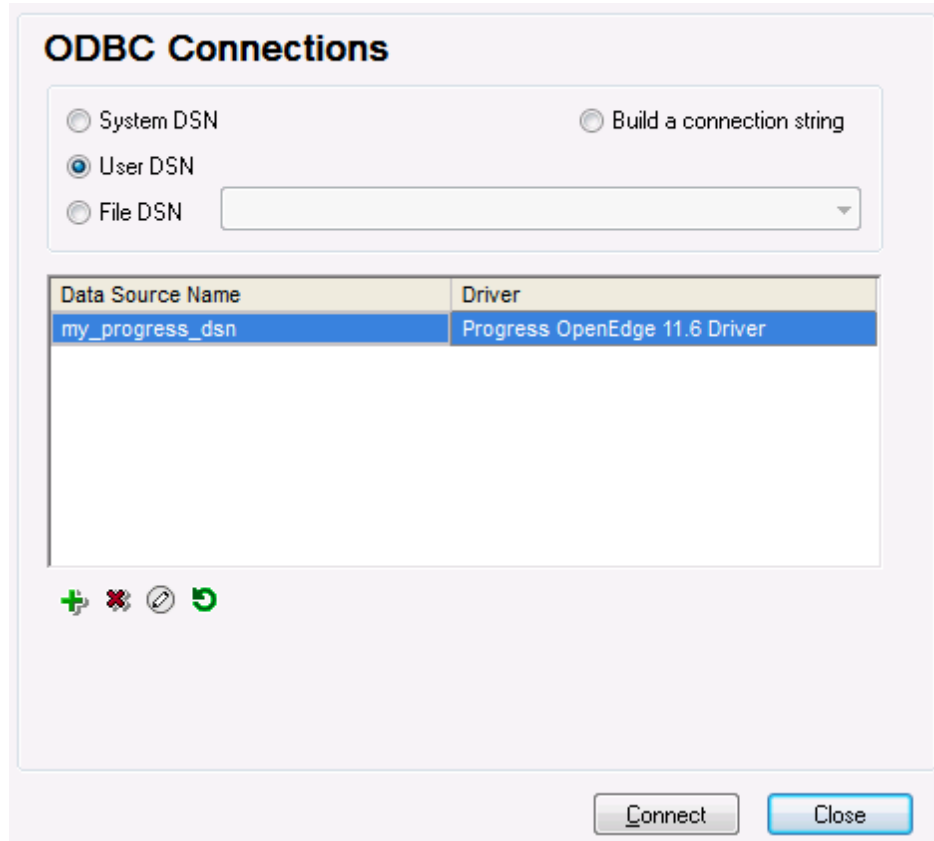**Connecting to Progress OpenEdge through ODBC**

1. Start the database connection wizard.
2. Click **ODBC Connections**.
3. Click **User DSN** (alternatively, click **System DSN**, or **File DSN**, in which case the subsequent instructions will be similar).
4. Click **Add** ➕ .
5. Select the **Progress OpenEdge Driver** from the list, and click **User DSN** (or **System DSN**, if applicable).
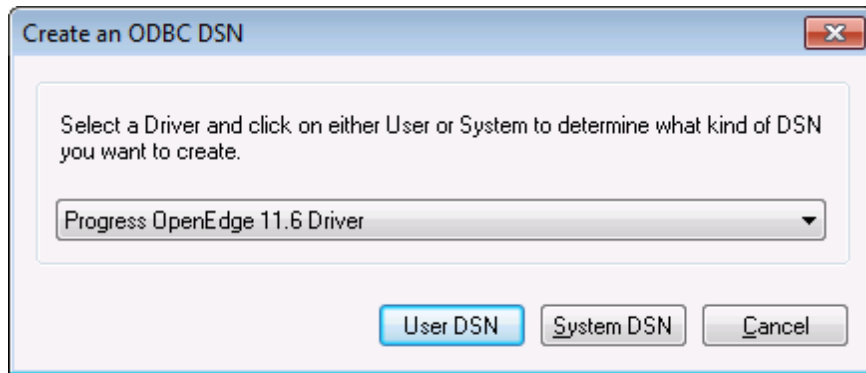
6. Fill in the database connection credentials (Database, Server, Port, User Name, Password), and then click **OK**. To verify connectivity before saving the entered data, click **Test Connect**.
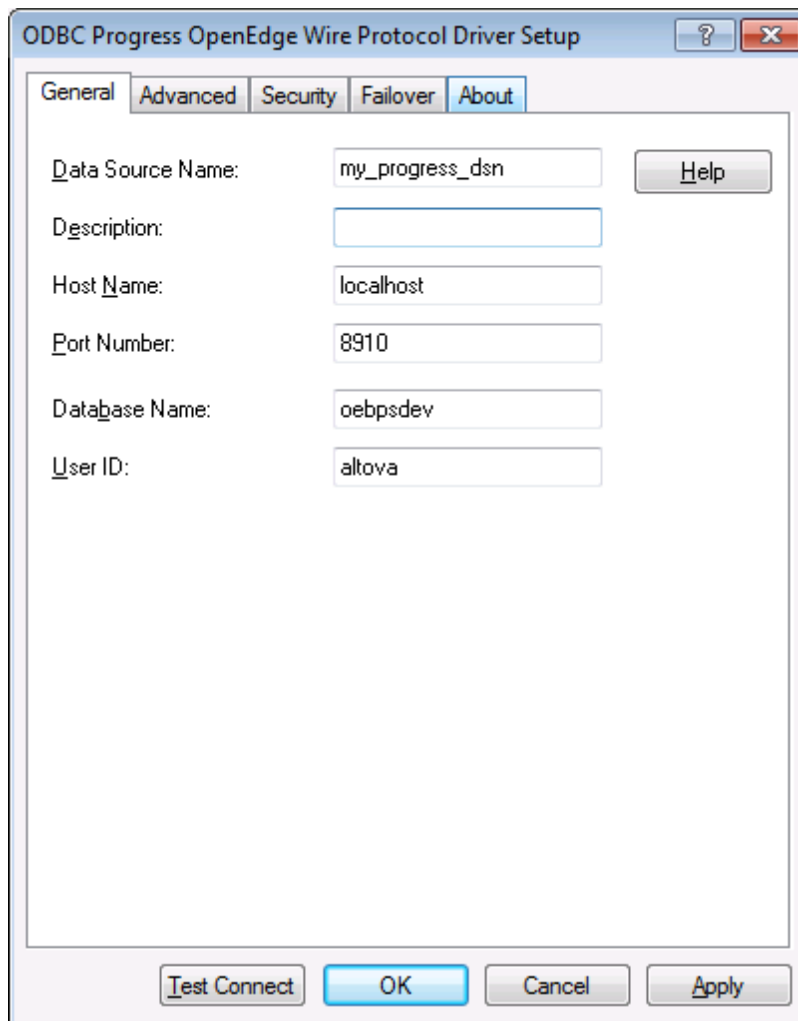
7. Click OK. The new data source now appears in the list of ODBC data sources.

8.   Click **Connect**.

Connecting to Progress OpenEdge (JDBC)

This topic provides sample instructions for connecting to a Progress OpenEdge 11.6 database server through JDBC.

Prerequisites

- Java Runtime Environment (JRE) or Java Development Kit (JDK) must be installed on your operating system. Make sure that the platform of DatabaseSpy (32-bit, 64-bit) matches that of the JRE/JDK.
- The operating system's PATH environment variable must include the path to the bin directory of the JRE or JDK installation directory, for example `C:\Program Files (x86) \Java\jre1.8.0_51\bin`.
- The Progress OpenEdge JDBC driver must be available on your operating system. In this example, JDBC connectivity is provided by the **openedge.jar** and **pool.jar** driver component files available in **C:\Progress\OpenEdge\java** as part of the OpenEdge SDK installation.
- You have the following database connection details: host, port, database name, username, and password.

Connecting to OpenEdge through JDBC

1. Start the database connection wizard.
2. Click **JDBC Connections**.
3. Next to "Classpaths", enter the path to the .jar file which provides connectivity to the database. If necessary, you can also enter a semicolon-separated list of .jar file paths. In this example, the required .jar file paths are: `C:\Progress\OpenEdge\java\openedge.jar;C:\Progress\OpenEdge\java\pool.jar;`. Note that you can leave the "Classpaths" text box empty if you have added the .jar file path(s) to the CLASSPATH environment variable of the operating system (see also Configuring the CLASSPATH).
4. In the "Driver" box, select **com.ddtek.jdbc.openedge.OpenEdgeDriver**. Note that this entry is available if a valid .jar file path is found either in the "Classpath" text box, or in the operating system's CLASSPATH environment variable (see the previous step).

Classpaths:	C:\Progress\OpenEdge\java\openedge.jar;C:\Progress\OpenEd;
Driver:	com.ddtek.jdbc.openedge.OpenEdgeDriver
Username:	dbuser
Password:	●●●●●●
Database URL:	jdbc:datadirect:openedge://localhost:8910;databaseName=o ebpsdev

Connect   Close

5. Enter the username and password to the database in the corresponding text boxes.
6. Enter the connection string to the database server in the Database URL text box, by replacing the highlighted values with the ones applicable to your database server.

```
jdbc:datadirect:openedge://host:port;databaseName=db_name
```

7. Click **Connect**.

Connecting to Sybase (JDBC)

This topic provides sample instructions for connecting to a Sybase database server through JDBC.

Prerequisites:

- Java Runtime Environment (JRE) must be installed on your operating system.
- Sybase *jConnect* component must be installed on your operating system (in this example, *jConnect 7.0* is used, installed as part of the *Sybase Adaptive Server Enterprise PC Client* installation). For the installation instructions of the database client, refer to Sybase documentation.
- You have the following database connection details: host, port, database name, username, and password.

To connect to Sybase through JDBC:

1. Start the database connection wizard.
2. Click **JDBC Connections**.
3. Next to "Classpaths", enter the path to the .jar file which provides connectivity to the database. If necessary, you can also enter a semicolon-separated list of .jar file paths. In this example, the required .jar file path is: **C:\sybase\jConnect-7_0\classes\jconn4.jar**. Note that you can leave the "Classpaths" text box empty if you have added the .jar file path(s) to the CLASSPATH environment variable of the operating system (see also Configuring the CLASSPATH).
4. In the "Driver" box, select **com.sybase.jdbc4.jdbc.SybDriver**. Note that this entry is available if a valid .jar file path is found either in the "Classpath" text box, or in the operating system's CLASSPATH environment variable (see the previous step).

Classpaths:	C:\sybase\jConnect-7_0\classes\jconn4.jar;
Driver:	com.sybase.jdbc4.jdbc.SybDriver
Username:	dbuser
Password:	••••••
Database URL:	jdbc:sybase:Tds:SYBASE12:2048/PRODUCTSDB

Connect   Close

5. Enter the username and password to the database in the corresponding text boxes.
6. Enter the connection string to the database server in the Database URL text box, by replacing the highlighted values with the ones applicable to your database server.

```
jdbc:sybase:Tds:hostName:port/databaseName
```

7.   Click **Connect**.

5.3   Adding Data Sources

The data sources (i.e., database or database schema, respectively) you want to connect to in DatabaseSpy have to be added to the Data Sources folder in the Project window. DatabaseSpy offers a Connection Wizard for the most commonly used database types and dialog boxes for creating ADO and ODBC connections as well as for choosing an Altova Global Resource. All these options are combined in the **Add a Data Source** dialog box which can be called via the **File** menu, the toolbar, a keyboard shortcut, or a context menu option in the Project window.

To add a data source to your project:

1. Do one of the following:

    *   Choose the menu option **File | Create a Database Connection...**.

    *   Press the keyboard shortcut **Ctrl+Q**.

    * Click the **Connect to a database** 	🖥 button in the Standard toolbar.

        The **Add a Data Source** dialog box opens.

    *   In the Project window, right-click the Data Sources folder and select **Add a New Data Source...** from the context menu.

        The **Add a Data Source** dialog box opens.

2. Choose one of the following buttons on the left side of the dialog box and follow the instructions for the respective connection type:
    *   Connection Wizard
    *   ADO Connections
    *   ODBC Connections
    *   JDBC Connections
    *   Global Resources

Data source options

When you right-click a data source name in the Project window, you can choose from several options depending on the current state (connected or disconnected) of the data source and whether or not the data source is a global resource. The corresponding keyboard shortcuts are shown to the right of the options if they are available.

Connect		Connects to a data source if it is disconnected.
Disconnect		Disconnects from a data source if it is connected.
Remove	**Del**	Removes a data source from the project.
Rename	**F2**	Renames a data source. Note that only disconnected data sources can be renamed.
Convert to Global Resource		Converts the data source to an Altova Global Resource and adds a database alias with the data source name to the GlobalResources.xml definition file. If an alias with that name already exists, a warning message will be displayed and the operation will fail.
Copy Global Resource into Project		Generates a copy of the global resource and adds it to the project as stand-alone data source. You will have to specify a name for the new data source. The original global resource remains in the project.

Edit Global Resource...	Opens the **Global Resource** dialog box where you can edit the configuration of the database alias.

To copy a global resource into a project:

1. In the Project window, right-click a data source that has been added to the project as global resource and select **Copy Global Resource into Project** from the context menu.

2. Enter a name in the **Set a Data Source Name** dialog and click **OK**.

3. Optionally, remove the global resource from the project.

To remove a data source from a project:

Select the data source you want to remove and do one of the following:

- Right-click the data source and choose **Remove** from the context menu.
- Press **Del.**

   If the data source is connected, DatabaseSpy displays a warning that removing the data source will close the connection.

Renaming data sources

Data sources in projects can be renamed either in the Properties window or via the context menu directly in the Project window.

Please note: You can only rename data sources that are currently disconnected. Both the **Rename** command in the context menu and the title bar in the Properties window are grayed out for connected data sources.

To rename a data source in a project:

1. Make sure that the data source is disconnected.

2. Select the data source you want to rename and do one of the following:

   - In the Properties window, double-click the title bar.

   - In the Project window, right-click the data source and select **Rename** from the context menu.

   - Press **F2**.

3. Enter the new name and press **Enter**.

4. Save the project.

5.4   Adding Files

You can add existing SQL scripts to your DatabaseSpy Project, or you can save designs, data comparisons, or schema comparisons in files and add them. The Project window provides folders for the different file types and you can define properties (e.g., data source to connect to, execution options for SQL, etc.) on file or folder level. This way, you have all the file that you may need at hand, even if they are stored in different locations. If you need to find the exact path to a file that has been added to a project, DatabaseSpy provides the following context menu option:

- Right-click a file in the Project window and select **Locate File...** from the context menu. DatabaseSpy opens the folder that contains the file in Windows Explorer.

Adding files to the project

Existing files (e.g., advanced SQL Scripts) can be included into a project with a few mouse clicks.

To add files to a project:

1. In the Project window, right-click a folder and select **Add Files to Project...** from the context menu. Alternatively, click the **Add files to the Project** button in the Project window toolbar. The **Open** dialog appears.

2. Browse for the file that you want to add to the project.

3. Click **Open**. The selected file is added to the corresponding folder of the project.

4. Save the project file.

In addition, you can also save the content of the SQL Editor, Design Editor, Data Comparison or Schema Comparison window to a file and add this file to the project at the same time. For every file that is saved for the first time or under a new name, DatabaseSpy pops up the **Add to Project** dialog box which allows you to also add the file to the project.
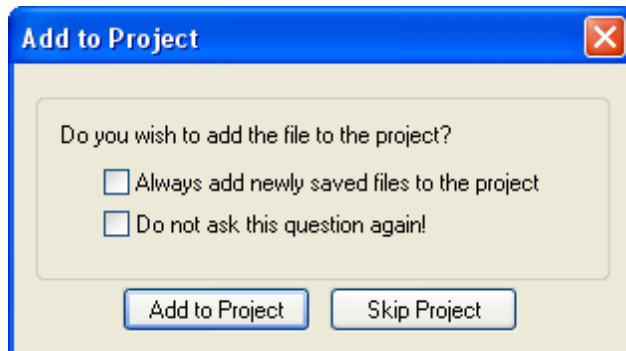
The **Add to Project** dialog box also serves as a shortcut to the General options, allowing you to define that newly saved files be always added to the project automatically. In addition, you can also suppress the display of this dialog box in the future.

You can use the **File** menu, the toolbar, or the context menu that opens when you right-click the name tab of an SQL Editor, Design Editor, or Comparison window to save a file.

To save a file:

1. Make the SQL Editor, Design Editor, or Comparison window that contains the script, design, or comparison to be saved the active window and do one of the following:

- Click the **Save** button in the Standard toolbar.
- Select the menu option **File | Save**.
- Press **Ctrl+S**.
- Right click the naming tab at the bottom of the document window and select **Save...** from the context menu.

The **Save As** dialog box opens.

2. Enter a name for the new file, select a folder and click the **Save** button.

The **Add to Project** dialog box opens (if the dialog box does not pop up, check whether the `Do not show dialog` option is checked in the Newly saved files group box of the General options).

3. Do one or more of the following:
- Click the **Add to Project** button to add the file the currently active project.
- Click **Skip Project** to close the dialog box without adding the file to the project.
- Activate the `Always add newly saved files to the project` check box.
- Activate the `Do not ask this question again!` check box to suppress the display of the dialog box in the future.

To add the active file to the project:

1. Make the window that contains the file you want to add to the project the active window.
2. Do one of the following:
- In the Project window, click the **Add the Active File** icon in the toolbar, or right-click the respective folder and select **Add Active File to Project** from the context menu.
- In the SQL Editor, Design Editor, or Comparison window, right-click the tab at the bottom of the Editor window and select **Add to project** from the context menu. If the file has not been saved yet, the Windows **Save As** dialog box opens where you can enter a name and path for the file.

The file appears below the respective folder in the Project window.

3. Save the project file.

To remove a file from a project:

Do one of the following:

- Right-click the file and select **Remove** from the context menu.
- Select the file and press **Del**.

5.5   Favorites

Frequently used database objects that have been added to your favorites are displayed in the Favorites folder. All options for an individual object are available in the context menu and you can directly open the Online Browser by clicking the arrow to the right of the object name.

Defining favorites

DatabaseSpy's Favorites feature allows you to mark frequently used objects in your project as favorites. These items then show up in the Favorites folder of the Project window and are shown bold in the Online Browser so that you have always quick access to them.
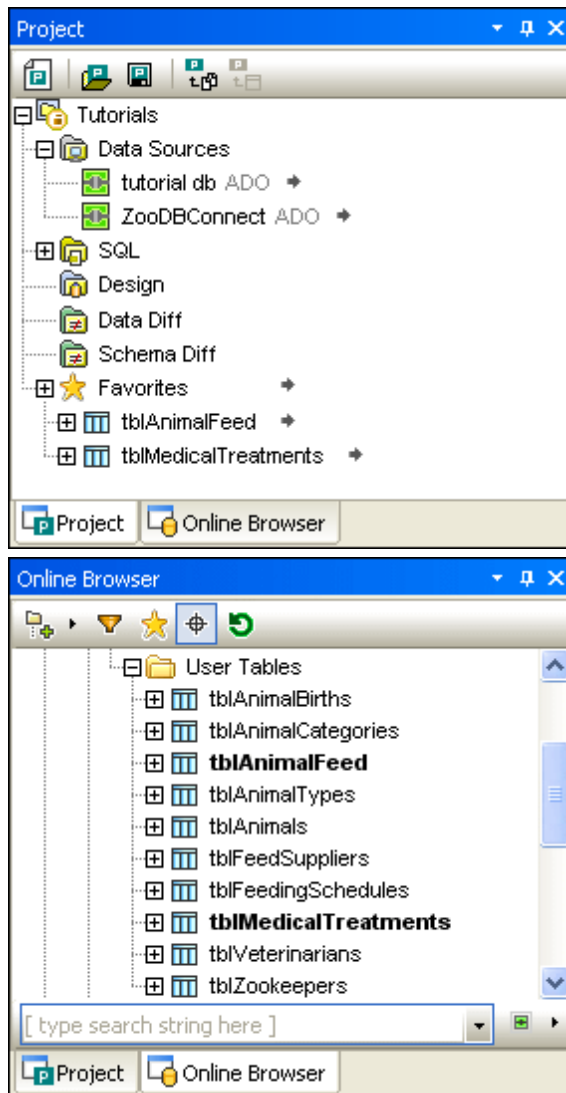
In the Project window, an arrow is shown in the Favorites folder next to the item name. When you click on this arrow, the Online Browser opens with the favorite item selected. Note that you can also add databases and schemas or individual table columns to the Favorites folder.

For every item in the Favorites folder, the relevant options are available in the context menu. For example, you can connect a data source directly from within the favorites folder.

To add items to the Favorites folder:

- In the Online Browser, right-click the respective tables or other database objects and select **Add to/Remove from Favorites (Ctrl+F2)** from the context menu.

- Alternatively, if you have activated the `Enable full row selection` check box in the [Online Browser](#) options, click the favorites icon that appears to the right of a database object if you move the mouse cursor over it in the Online Browser.

  The tables or objects are added to the Favorites folder and are shown in bold type in the Online Browser. In addition, the favorites icon beneath the object name appears yellow when full row selection is enabled.

To view only the favorites in the Online Browser:

1. Click the **Favorites** icon to switch to the favorites view.
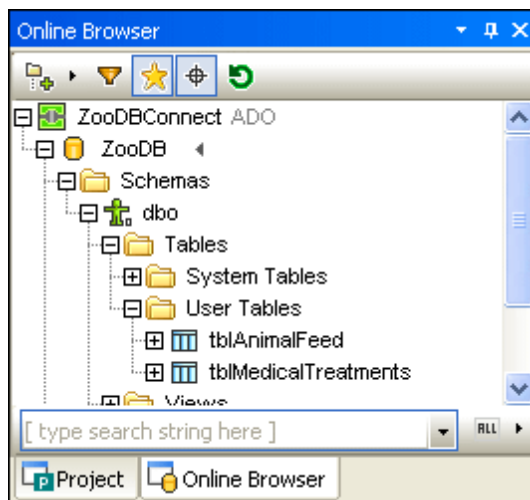
  Note that the **Favorites** icon is now active and only database objects that are contained in the Favorites folder are displayed in the Online Browser.

2. Click the **Favorites** icon again to switch back to the normal Online Browser view.

To rename database objects in the Favorites folder:

1. Make sure that the corresponding data source of the favorites item is connected.

2. Select the database object you want to rename in the favorites folder and do one of the following:

   - In the Properties window, double-click the title bar.

   - In the Project window, right-click the database object in the favorites folder and select **Rename** from the context menu.

   - Press **F2**.

3. Enter the new name and press **Enter**.
   A change script is generated in the Database Structure Change Script window.

4. Execute the change script.

To remove a database object from the Favorites folder:

Do one of the following:

- In the Project tab, right-click the object in the Favorites folder and select **Remove from Favorites (Del)**.

- In the Online Browser, right-click the object in its folder and select **Add to/Remove from Favorites (Ctrl+F2)**.

- When full row selection is enabled, in the Online Browser, move the mouse cursor over a favorite item an click the yellow favorites item to the right of its name.

To remove all database objects from the Favorites folder:

- In the Project tab, right-click the Favorites folder and select **Remove all Favorites**.

5.6   Renaming and Deleting Objects

You have several options to rename a database object:

- • In the Properties window of the Online Browser
- • In the Online Browser, by using the context menu
- • In the table display of the Design Editor

A change script is generated in the Database Structure Change Script window and must be executed to commit the change to the database. DatabaseSpy checks whether you enter a valid name and displays a tooltip if you choose a name that already exists for an object in the database and ignores the input.

Please note: For MS Access, the names of existing tables or columns cannot be changed. If you try to rename an Access table or column name, DatabaseSpy pops up a warning message and rejects the change.

You can rename the following database objects directly in the Online Browser:

- • Tables
- • Columns
- • Keys
- • Check constraints
- • Indexes
- • Views
  Whether or not a view can be renamed, depends on the database kind you are working with. Renaming is not possible for databases such as MS Access, IBM DB2, MySQL 5, or Oracle.
- • Stored procedures

To rename a database object:

1. Do one of the following:

   • In the Online Browser, select a database object and press **F2** or double-click its title bar in the Properties window.

   • In the Online Browser, right-click a database object and select **Rename** from the context menu.

   • In a Design Editor window, double-click the title bar of a table or the database object, or right-click and select **Rename Table** or **Rename Column** or **Rename Key** or **Rename Index** or **Rename Check Constraint**, respectively, from the context menu.

2. Change the name as desired and press **Enter**.
   The changed database object icon appears to the left of its name.

3. If applicable, click the **Update Change Script** button to update the change script.

4. In the Database Structure Change Script window, click the **Execute Change Script** button.

Deleting database objects

In DatabaseSpy's Online Browser, you have several possibilities to delete a database object from a database:

- • Right-click a database object and choose **Delete** from the context menu.

- Select a database object and press **Del**.
- Right-click a database object and select **Show in new SQL Editor | Drop** from the context menu.

In a Design Editor window, you can do the following to delete a database object:

- Right-click a database object and choose **Delete selected Objects** from the context menu.
- Select a database object and press **Del**.

All the options mentioned above are also valid if you have selected more than one database object. If you select one or more database objects of a table as well as the table itself and press the **Delete** key in a Design Editor window, DatabaseSpy prompts you to choose whether you want to delete the entire table or only its objects.

In order to actually delete the database object(s) from the database, you have to execute the change script or the SQL statement, respectively.

5.7   Properties

In DatabaseSpy you can view properties by selecting objects in (i) the Project window, (ii) the Online Browser, (iii) the Design Editor, or (iv) the Data or Schema Comparison window . In addition, you can define properties for designs, SQL files, and comparison files that have been added to a project. The properties are displayed in the Properties window.

Changing a property in the Property Window or in the Design Editor automatically generates an SQL change script which appears in the in the Database Structure Change Script window either automatically or after you have clicked the **Update Change Script** ![icon] button, depending on how you have configured the Change Script options. Properties that cannot be edited are grayed out.

In the **Online Browser**, properties for all database objects as well as for any active data source connection, its database and/or schema(s) can be viewed in the Properties window. When folders are displayed, clicking a folders shows an overview of the folders content in the Properties window.

The properties that are displayed in the Properties window depend on the database object that is currently selected in the Online Browser, Design Editor, or Comparison window. If an object has child objects, the content of the Properties window changes in accordance with the level to which you expand the object, that is, only properties for expanded items are displayed in the Properties window.

For the data source connection, DatabaseSpy displays the same properties like in the Project window. The properties that are displayed when you click on a database in the Online Browser are for information purposes only and cannot be changed.

Object count

For every database object that contains child objects, an overview section is displayed. This section provides hyperlinks for counting the number of the individual children of the database object as well as a hyperlink for counting all children of the object. Note that the count is retrieved automatically if you expand the object in the Online Browser or show a table in the Design Editor.

To retrieve the number of child objects in the Overview section of the Properties window, you have therefore the following options:

- Click the "Update all Counts" hyperlink to retrieve the count for all child objects.
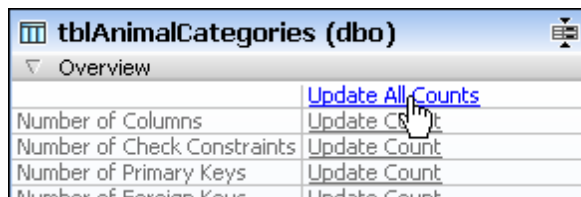
- Click the "Update Count" hyperlink for an individual database object to retrieve the count for this specific item.
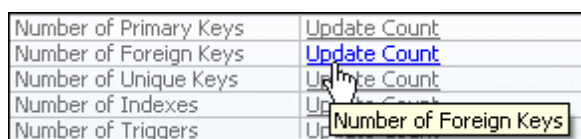
- Show a table in the Design Editor; this automatically retrieves the count for all child

objects of the table.

- Right-click a table and select **Expand | Children** from the context menu. This expands all sub-folders of the table in the Online Browser and thus shows the number of child objects in the table properties.

- Expand one or more sub-folders in the Online Browser and then click the table again. The number of items contained in the folder is displayed in the table properties.

Tables

Table properties can be viewed (i) in the Properties window and (ii) to some extent in the table design. If properties are displayed in the Properties window, it makes no difference whether a database object is selected in the Online Browser or in the Design Editor.

The Overview section lists the child objects of the table and you can retrieve the number of the individual children separately. In the Constraints section, you can see at a glance which constraints are defined on which columns for the table. Check Constraints on table level can even be edited in this section. Whereas the General section provides only read-only information, you can enter a new or change an existing description for the table in the Description section.

A table may contain the following sub-folders:

- **Columns**
  When you expand the Columns sub-folder under a table in the Online Browser or expand the Columns section in a table design in Design Editor, and select a column, the column properties are displayed in the Properties window. Here you can change the data type and the Nullable property in the General section, or enter a description in the Description section. The remaining sections (Overview and Advanced) are read-only.

- **Constraints**
  The constraints properties for check constraints include the name of the table the constraint is assigned to, the expression used for the constraint, and a description. You can change the expression and the description in the Properties window.

- **Keys**
  If you select a primary or unique key in the Online Browser or in the Design Editor, the Properties window displays the number of columns that are used to build the key as well as the table the key is assigned to.

  For foreign keys, also the referenced tables and columns as well as the action on delete and update are displayed in the properties.

- **Indexes**
  Index properties show information such as the number of columns involved, index type, and the column name an index is assigned to. The Options section provides additional read-only information.

Views

In the properties for views, DatabaseSpy displays the number of columns that are included in the view (provided that the object count has been retrieved), and the schema in which the view is defined. The Advanced section shows additional read-only information.

XML Schemas

The only option that you can change directly in the Properties window for XML schemas is the `Decomposition` check box. Any other property is read-only in the Properties window and can

---

only be changed in the **XML Schema management for databases** dialog box.

Procedures

For procedures, only the number of parameters and the schema in which the procedure is defined are displayed in the Properties window. If you expand a procedure to show the parameters, you can also display the data type and the direction mode of a parameter in the Properties window. All properties for procedures are read-only.

Functions

The properties displayed for functions include the number of parameters, the schema, the `Deterministic` check box, and the language. If you expand a function to show the parameters, you can also display the data type and the direction mode of a parameter in the Properties window. All properties for functions are read-only.

Please note: The constraints information can only be displayed if all the table's children objects have already been loaded. This can be achieved in one of the following ways:

- by expanding the table in the Online Browser
- by generating an SQL statement for that table in the SQL Editor
- by showing the table in the Design Editor

Changing table properties

All the table and columns properties that can be edited can be changed either in the Properties window of the Online Browser or in the property window in the Design Editor. In the table display in the Design Editor, you can also edit the data type or check or uncheck the `Nullable` check box.

To change the properties of a column:

1. Do one of the following:

   - In the Online Browser, select a column and change its properties in the Properties window of the Online Browser.

   - In a Design Editor window, select a column name, click the ball at the right edge of the table display and change the properties in the property window that appears.

   The changed column ![icon] icon appears to the left of the column name and the changed table ![icon] icon appears to the left of the table name.

2. If applicable, click the **Update Change Script** ![icon] button to update the change script.

3. In the Database Structure Change Script window, click the **Execute Change Script** ![icon] button.

5.7.1   Project Properties

The project properties are displayed in the Properties window when you click on an object in the Project window. The properties change dynamically when you click on the individual folders or files in a project and can be edited in the Properties window. Properties that cannot be edited are grayed out. All properties that you define for a folder (or sub-folder) are automatically inherited by the files that are contained in that (sub-)folder. You can, however, break that tie by defining separate properties for individual files or sub-folders.

If you have [defined properties for the different folders](#) in the Project window, any file that you add to your project will inherit these properties from its parent folder. You can change the properties for a file at any time. DatabaseSpy displays properties that deviate from the parent folder's values in bold type. Once you have changed a property of a file, the link between the file's property and its parent folder's property is broken and will also not be restored by choosing the identical value in the file's drop-down list. You can, however, [restore the folder-file connection](#) by using the respective command in the context menu.
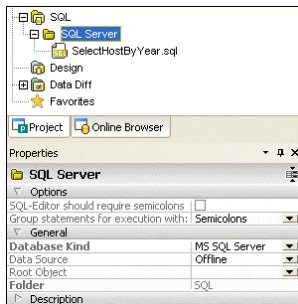
Example: Let's assume you have defined a sub-folder for SQL Server-related files in your Project window, and that you have set the `Database Kind` property to "MS SQL Server" for that sub-folder (*see screenshot below*). Since no properties have been defined for the parent SQL folder, the `Database Kind` property appears in bold type.

The file SelectHostByYear.sql in the SQL Server folder inherits the properties from the SQL Server folder and the `Database Kind` property therefore appears in normal font type. However, the `Data Source` and `Root Object` properties have been changed and the `SQL Editor should require semicolons` check box has been activated, and these three properties are displayed in bold type.
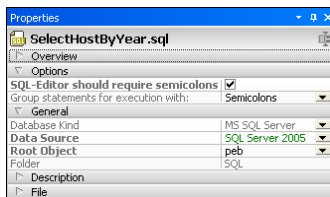
If you deactivate the `SQL Editor should require semicolons` check box again, the property will still be displayed bold although the value is now identical to the parent folder's value (*see screenshot below*). This indicates that the tie between the folder's properties and the file's properties has been broken and—although currently identical—an individual value has been defined for this property.
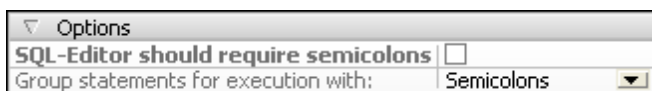
Once a property has been changed for an individual file, the tie between the file and its parent folder can only be re-established via the **Restore value to parent's option** context menu command.

To reset a file's properties to its parent folder's options:

1. In the Project window, select the file and right-click the property you want to reset in the Properties window.

2. Choose **Restore value to parent's option** from the context menu.

   The file inherits the property from the parent folder again.

For the project itself, the path to the project file and, optionally, a description are displayed in the Properties window.

Data source properties

You can define in the data source properties to automatically connect to the data source when the project is opened. To do so, activate the `Connect on Open Project` check box in the Options section of the data source properties.

If you prefer to view data sources in the Online Browser also when they are not connected, you can activate the `Always show in Online Browser` check box which is also located in the Options section.

The `Global Resource` check box is activated if the data source has been added to the projects as a global resource. Note that this property cannot be changed by deactivating the check box. You must copy the global resource into the project and remove the global resource from the project to convert a global resource into a local one.

The `Use ODBC Native Interface` check box is for advanced users who may wish to retrieve structural information about their databases using the API methods provided from ODBC directly rather than DatabaseSpy's internal SQL queries. This might be desired when the user is experiencing difficulties retrieving data, or feels the application is too slow when retrieving this information. The down side to this approach is that the amount of structural information that the user can retrieve via the ODBC interface is limited.

The Driver Information section lists name, version, issuing company, file name, full path, and date of the driver that is used for the selected data source connection.

You can also manually edit the connection details in the Connection section of the data source properties but keep in mind that a project might not be able to establish a connection to the data source if the connection properties have been manipulated. For security reasons, the password for the data source connection cannot be edited directly in the Properties. If you double-click into the `Password` field, an extra dialog box opens where you have to enter the existing password before you can define and confirm a new password.

Detailed properties of the individual database objects can be viewed by selecting the respective database object in the Online Browser. You must connect to a data source before you can view its database objects in the Online Browser.

SQL folder properties

When you select the SQL folder in the Project window, you can edit the properties that apply to the entire folder and are inherited by all files contained in this folder. These properties can be overridden by properties that you define on file or sub-folder level.

The General section provides three drop-down lists for database kind, data source, and root object as well as text fields containing the folder and project names. The Description section shows a text field for entering a description. Please note that the properties for database kind and root object are filled in automatically when you select a data source from the drop-down list. However, you can then change the root object, if applicable.

Design and Favorites properties

For the Design and Favorites folders, the Properties window provides drop-down lists for database kind, data source, and root object. In addition, a text field for entering a description is available. The Design properties include also a text field displaying the project name.

Data Diff properties

In the Data Diff folder, DatabaseSpy displays statistical information on the comparison file, and you have the possibility to edit the compare options as well as the options for string and xml comparison and the merge execution options on file level.

Schema Diff properties

In the Schema Diff folder, DatabaseSpy displays statistical information on the comparison file, and you have the possibility to automatically start a comparison when the document is opened.

5.7.2   SQL Properties

The following properties of SQL files as well as SQL Editor windows (with or without SQL statements) are displayed in the Properties window:

	Overview
File Kind	Always reads "SQL" for SQL files.
	Options
SQL Editor should require semicolons	Activate this check box to improve performance and reliability of all parser-dependent operations. Only statement that are terminated with a semicolon will be considered by the parser.
Group statements for execution with:	Here you can choose how statements should be grouped when the script in the SQL Editor window is executed.
	General
Database Kind	For unsaved SQL Editor windows, this drop-down list lets you define a database type. Note that this field will be adapted automatically if you choose a data source.
Data Source	Here you can choose from the data sources that have been added to the project. As several data sources may be active at one time, it is necessary to be able to assign data sources to SQL files as a group or individually. **Please note**: Assignments made in the project tab apply to the SQL files when they are opened or executed.
Root Object	If a data source has already been selected, you can choose a root object from this drop-down list.
	Description
Description	Here you can enter a description of the file.
	File
Creation Date Modification Date Access Date	Displays the respective creation and modification dates and shows when the file has last been accessed.
Size	Shows the file size.
Read Only Hidden	These properties are read-only and can only be changed outside DatabaseSpy.
Full Path	Displays the full path to the file. To jump directly to this location, right-click the file in the Project window and select **Locate File...** from the context menu.

Default properties for SQL Editor windows

The following scenarios can occur when you open a new SQL Editor window:

- **No project**: If no project has been opened in the Project window, any new SQL Editor window that you open will have the properties Database Kind = "Unknown" and Data Source = "Offline".

- **Data source connected**: If you open an SQL Editor window immediately after you have connected to a data source, the connection properties of this data source will be used for that SQL Editor window. Note that the data source still has to be selected in the Project window.

- **Several data sources defined in one project**: If you have opened a project with several different data sources, you can set the connection properties by clicking on a particular data source before opening an SQL Editor window. If you select a disconnected data source, the Execution Target Bar provides you with a **Connect** button to establish a connection to the data source. Clicking on any other item in the Project window will open an SQL Editor in the Offline mode.

Working with SQL files in the Offline mode

One of the options available to you when selecting from the `Data Source` combo box is "Offline". Designating an SQL file as offline means that it is blocked from being executed. The file cannot be executed when opened in the SQL Editor or when selecting **Execute SQL** from the context menu in the Project window.

All SQL Editor commands and functions are available when editing this file however, you just have the added security of not being able to execute it without having to change the settings in the Properties window.

To define individual settings for project SQL files:

1. In the Project window, select an SQL file in the SQL folder.

2. In the Properties window, select the data source that you want to assign to the file from the `Data Source` field. Active data sources are highlighted in the Properties window as well as in the `Data Source` drop-down list.

3. If required (e.g., because the SQL file includes target keywords), choose either "Semicolons" or "SQL Grammar" from the `Group statements for execution with` drop-down list.

4. If you want the parser to consider only statements that are delimited by semicolons, activate the `SQL Editor should require semicolons` check box and select the desired option from the `Group statements for execution with` drop-down list.

If you have defined properties for the different folders in the Project window, any file that you add to your project will inherit these properties from its parent folder. You can change the properties for a file at any time. DatabaseSpy displays properties that deviate from the parent folder's values in bold type. Once you have changed a property of a file, the link between the file's property and its parent folder's property is broken and will also not be restored by choosing the identical value in the file's drop-down list. You can, however, restore the folder-file connection by using the respective command in the context menu.

To define individual settings of a file opened in the SQL Editor:

1. In the SQL Editor, click the tab of the SQL file whose properties you want to define or change.
   The file settings are displayed in the Properties window.

2. Change the settings according to your needs by double-clicking the `Data Source` or `Root Object` field in the Properties window and choosing an option from the drop-down list.

3. If required (e.g., because the SQL file includes target keywords) choose either

"Semicolons" or "SQL Grammar" from the `Group statements for execution with` drop-down list.

4. If you want the parser to consider only statements that are delimited by semicolons, activate the `SQL Editor should require semicolons` check box and choose the appropriate option from the `Group statements for execution with` drop-down list.

5. Save the SQL file.

5.7.3 Design Properties

In addition to the Properties window, you can show the properties of a column, an index, a key, or a check constraint directly in a Design Editor window. Design Editor's table display shows the data type and the `Nullable` check box for columns, the column and order for indexes, the column and referenced key for keys, and the expression for check constraints.

When you select an index, key, or check constraint in the table display in the Design Editor, the referenced column is automatically printed in bold.

When you click into an empty area of a Design Editor window, the design properties are displayed in the Properties window. Here you can change the data source or schema and view the file properties and path of the design file.

	Overview
File Kind	Always reads "Design" for design files (*.qdes).

	General
Database Kind	For unsaved SQL Editor windows, this drop-down list lets you define a database type. Note that this field will be adapted automatically if you choose a data source.
Data Source	Here you can choose from the data sources that have been added to the project. As several data sources may be active at one time, it is necessary to be able to assign data sources to SQL files as a group or individually. **Please note**: Assignments made in the project tab apply to the SQL files when they are opened or executed.

Root Object	If a data source has already been selected, you can choose a root object from this drop-down list.

Description

Description	Here you can enter a description of the file.

File

Creation Date Modification Date Access Date	Displays the respective creation and modification dates and shows when the file has last been accessed.
Size	Shows the file size.
Read Only Hidden	These properties are read-only and can only be changed outside DatabaseSpy.
Full Path	Displays the full path to the file. To jump directly to this location, right-click the file in the Project window and select **Locate File...** from the context menu.

Please note: The connection properties can only be changed for new designs. As soon as you show a table in a Design Editor window, the connection properties cannot be changed anymore.

To view the properties of a design file:

Do one of the following:

- Click into an empty area of a Design Editor window.

- If the design file has already been added to the project, click the QDES File in the Design folder of the Project window.

    The properties of the design are displayed in the Properties window.

5.7.4   Comparison Properties

When you click into an empty area of a **Data Comparison** or **Schema Comparison** window, the comparison properties are displayed in the Properties window. Here you can change the way tables and columns are compared, how the comparison file is stored in the project, and define several options for string and XML comparison. Properties that apply only to data comparisons are printed in *italics* type.

Overview

File Kind	Reads "Data Comparison" for database data comparison files (*.dbdif) and "Structure Comparison" for database schema comparison files (*.dbsdif).

Compare Options

Compare:	*Changes between native comparison and string comparison. Note that XML comparison is only supported in native comparison mode.*
Keep only different rows in memory	*Stores only rows that contain differences in the database data comparison file.*
Use native comparison for XML columns	*The content of XML columns will be compared as native XML. If this option is activated, additional properties are available in the XML Compare Options section.*
Append trailing zeros to float-numbers	*Floating point numbers will be filled up with trailing zeros.*
Ignore case	*Different case of database data will be ignored during comparison.*

Ignore whitespaces	*Whitespace will not be considered for the comparison.*
Start comparison when opening a document	A comparison will be started automatically if a comparison document is opened in DatabaseSpy.
Processing mode:	*Choose between sequential and parallel processing.*

String Options

Treat [NULL] as empty string	*NULL-values will be regarded as empty string for comparison.*

Merge Execution Options

Use transactions for merge	*Transactions will be used when changes are committed to the database as a result of a merge.*
Rollback on errors	*If an error occurs during the merge, the files will be rolled back to their original content.*

XML Compare Options

Compare whitespaces:	*Choose whether whitespace should remain unchanged, be normalized (i.e., all consecutive whitespace characters are reduced to one whitespace character), or be stripped (i.e., not considered) for comparison.*
Ignore case	*Different case will be ignored during comparison.*
Don't ignore case in node names	*Different case of node names will be regarded as difference.*
Ignore namespace	*Skips the namespace when comparing.*
Ignore prefix	*Ignores prefixes when comparing.*
Ignore order of children	*Compares nodes without considering the order in which child nodes appear.*
Ignore order of attributes	*Does not consider the order in which attributes appear.*
Resolve entities	*When activated, resolves all entities in the document, or compares entities as is when not activated.*
Ignore text	*Differences in corresponding text nodes will not be reported.*
Ignore node depth	*The additional depth of any element (i.e., more levels of descendants) relative to the depth of the corresponding element in the compared file will be ignored. This option must be unselected to enable merging and exporting differences.*
Ignore Attributes	*Nodes of type attribute will be ignored.*
Ignore CDATA	*Nodes of type CDATA will be ignored.*
Ignore Comments	*Nodes of type Comments will be ignored.*
Ignore Processing Instructions	*Nodes of type Processing Instruction will be ignored.*
Ignore Doctype	*Nodes of type DOCTYPE will be ignored.*
Ignore XML declarations	*Nodes of type XML Declaration will be ignored.*

Description

Description	Here you can enter a description of the file.

File

Creation Date Modification Date Access Date	Displays the respective creation and modification dates and shows when the file has last been accessed.

Size	Shows the file size.
Read Only Hidden	These properties are read-only and can only be changed outside DatabaseSpy.
Full Path	Displays the full path to the file. To jump directly to this location, right-click the file in the Project window and select **Locate File...** from the context menu.

Chapter 6

Browsing Data Sources

6   Browsing Data Sources

When you connect to a data source in DatabaseSpy, the data source is also displayed in the **Online Browser**, a powerful tool for analyzing, browsing, and searching for database objects, and for viewing the database structure of multiple databases simultaneously.

The Online Browser gives you an overview of the database objects on the database server(s) to which you are currently connected. Furthermore, you can show the row count for tables and views, generate SQL statements or retrieve data directly from the Online Browser and show them in the SQL Editor, or view the design of a database object in the Design Editor.
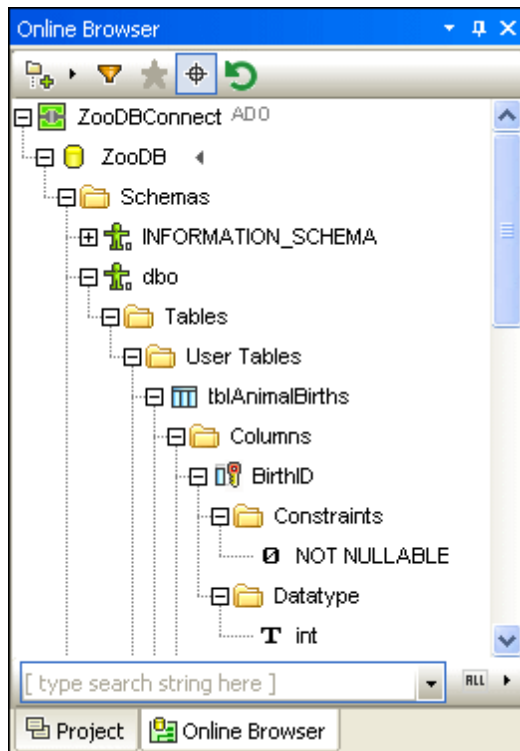
In the current version of DatabaseSpy, the following database objects are supported and displayed in the Online Browser:

- Databases
- Schemas
- Tables
- Views
- Procedures
- Columns
- Constraints
- Keys
- Indexes
- DataTypes
- XML Schemas (for DB2, SQL Server, and Oracle databases)
- Functions
- Triggers

Layouts

Different layouts for the objects in the Online Browser can be selected and you can apply filters to different folders as well as use the Object Locator to locate specific database objects in the Online Browser. Furthermore, you can also define favorites to facilitate quick access to objects that you use frequently.

Connection method

A check box is available in the Online Browser options that, when activated, shows also the connection method in the Project window as well as in the Online Browser together with the data source connection name.

Data comparison

When selecting tables and/or columns for comparison by clicking the **Browse** button in a Data Comparison window, DatabaseSpy displays the **Select Database Objects for Comparison** dialog box, which basically includes a variation of the Online Browser.
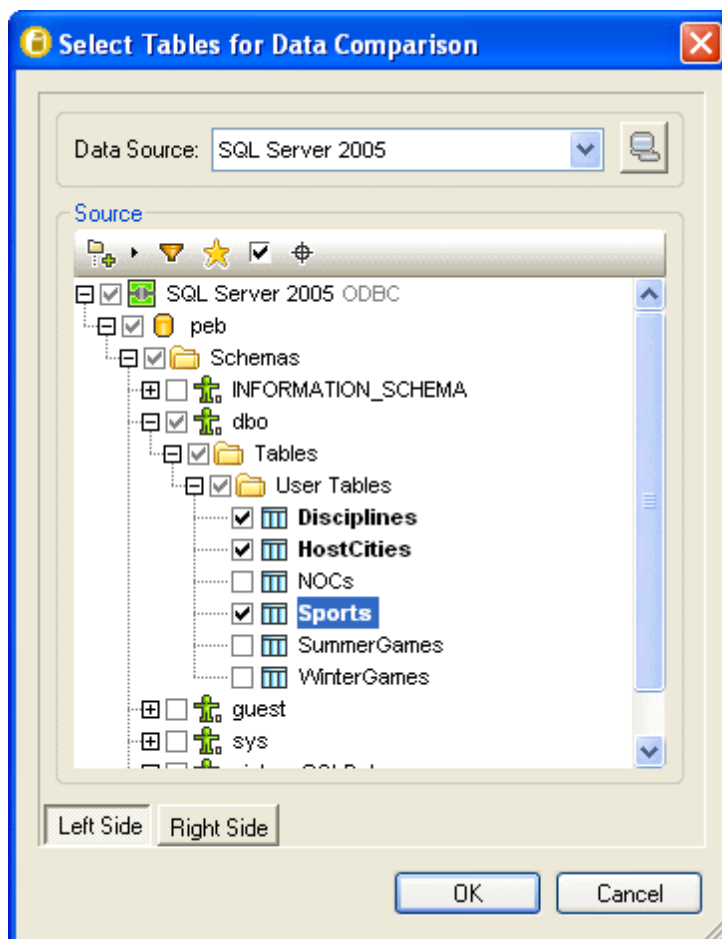
After you have clicked **OK**, DatabaseSpy displays the data source and its selected tables in a Data Comparison window. Each of the two databases is displayed in a component containing a tree structure of the tables that have been selected for comparison. The tables can be collapsed or expanded to show or hide the table columns.

Display options in the Data Comparison window

In the Data Comparison window, the schema of compared tables is displayed by default. In the Data Compare options, you can disable this option, and you can also choose to display the tables' data source name in the header of each component.

6.1   **Expanding and Collapsing Elements**

In the Online Browser, you can expand and collapse either individual elements or use the menu commands to expand or collapse all siblings or children of a certain element.

To expand or collapse a single element:

Do one of the following:

- To expand an element, click the **+** icon to the left of the element.
  Alternatively, you can also select the element and press the right arrow key or the **NUM + ** key.

- To collapse an element, click the **-** icon to the left of the element.
  Alternatively, you can also select the element and press the left arrow key or the **NUM -** key.

To expand or collapse siblings or children:

- Right-click an object in the Online Browser and select one of the following from the context menu:

  - **Expand | Siblings:** Fully expands all contents of the objects that are on the same level as the selected object. For example, if you right-clicked on a table, all sibling tables and the selected table will be expanded.

  - **Expand | Children:** Fully expands the descendent objects of the selected object.

  - **Collapse | Siblings:** Collapses all contents of the objects that are on the same level as the selected object. For example, if you right-clicked on a table, all sibling tables and the selected table will be collapsed.

  - **Collapse | Children:** Collapses the descendent objects of the selected object.

6.2   **Counting Data Rows**

You can display the number of rows contained in a table or view directly in the Online Browser without having to retrieve the data first. The context menu provides an additional **Row Count** sub-menu for this purpose, which contains options for showing/updating and clearing the row count. The **Row Count** sub-menu is also available for the Tables and Views folders and their sub-folders, allowing you to update the row count information for all tables or views with a single click.

When you move the mouse cursor over tables or views in the Online Browser, a hyperlink appears to the right of the object name. This hyperlink reads "(count)" if no row count has been done yet (*see screenshot above*), or "(n) update" if a number of n rows has already been retrieved. Clicking this hyperlink will also update the row count information.

The row count feature can be disabled by deactivating the corresponding check box in the Online Browser options.

To show the row count in the Online Browser:

1. Make sure that the `Optional display of Table and View row counts` check box is activated in the Online Browser options.

2. Do one of the following:

   • Right-click a table, view, or folder containing tables or views in the Online Browser and select **Row Count | Show/Update** from the context menu.

   • Place the mouse cursor over a table or view and click the `(count)` hyperlink that appears.

3. To update the row count information, repeat step 2. Note that the hyperlink reads `(n) update`, where n is the number of rows.

To hide the row count:

• Right-click a table, view, or folder containing tables or views in the Online Browser and select **Row Count | Clear** from the context menu.

6.3   Customizing the Browser Layout

By default, database object are displayed in the Online Browser with minimum vertical spacing, so as to show as much information as possible. You can, however, change this setting in the Online Browser options.

Default vertical spacing   Increased vertical spacing

To increase vertical spacing in the Online Browser:

1.   Select the menu option **Tools | Options** or press **Ctrl+Alt+O**.

2.   Change to the Online Browser page and deselect the `Reduce vertical spacing` check box in the Appearance group box.

3.   Click **OK**.

Layouts

DatabaseSpy provides several predefined layouts for the display of the various database objects in the Online Browser. You can currently select from among the following layouts:

*   The **Folders** layout organizes database objects into folders based on object type in a hierarchical tree. This is the default setting in DatabaseSpy.

*   The **No Schemas** layout is similar to the Folders layout, except that there are no schema folders; tables are therefore not categorized by schema.

*   The **No Folders** layout displays database objects in a hierarchy without using folders. For a better overview, you can activate the `Show labels in browser window` check box in the Display Labels group box of the Online Browser options, so that the name of each database object appears as a prefix.
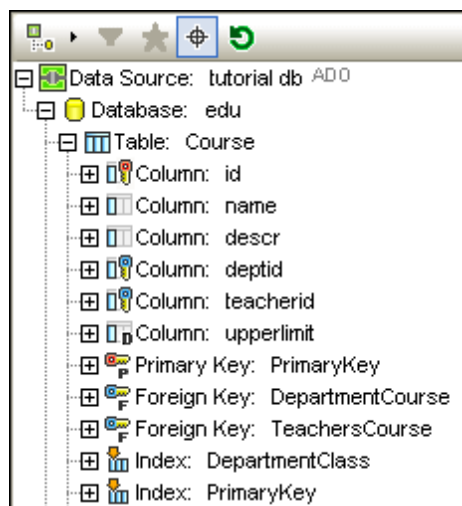
*   The **Flat** layout divides database objects by type in the first hierarchical level. For example, instead of columns being contained in the corresponding table, all columns are displayed in a separate Columns folder.
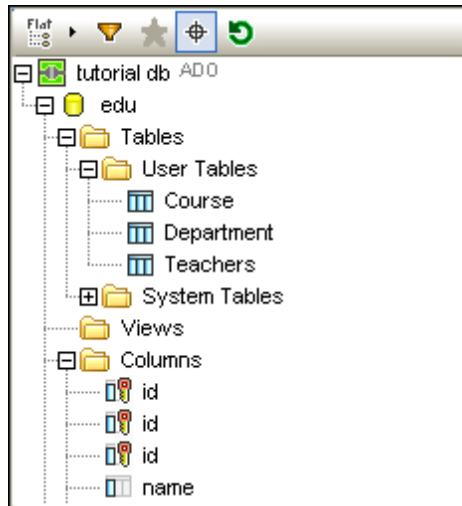
- The **Table Dependencies** layout categorizes tables according to their relationships with other tables. There are categories for tables with foreign keys, tables referenced by foreign keys and tables that have no relationships to other tables.
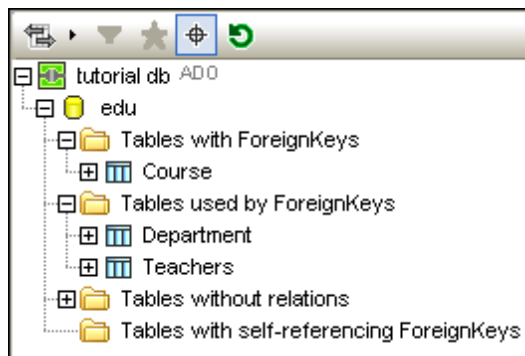
In the Online Browser options, you can define a default layout that DatabaseSpy uses when the Online Browser is displayed. Furthermore, you can decide in the options, whether or not the connection method should be displayed together with the data source connection name.

To select a layout for the Online Browser:

- In the Online Browser, click the layouts [icon] icon and select the desired layout from the drop-down list.
  Please note that the icon changes in accordance with the selected layout.
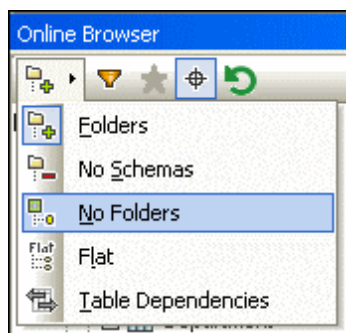
To define the default layout for the Online Browser:

1. Choose the menu option **Tools | Options** and select the Online Browser page of the **Options** dialog box.

2. In the `Default Layout for Browser Window` group box, select the desired layout from the drop-down list.

3. Click **OK**.

To sort tables into User and System tables:

1. In the Online Browser, right-click the Tables folder.
    A context-sensitive menu appears.

2. Select **Sort into User and System Tables**.
    The tables are sorted alphabetically in the User Tables and System Tables folders.

    Please note: You must be in the Folders, No Schemas or Flat layout in order to access this function.

6.4   Selecting the Root Object

The root objects for the currently supported databases are as follows:

Firebird	database
IBM DB2	schema
IBM DB2 for i (iSeries)	schema
IBM Informix	database
MS Access	database
MS SQL Server	database
MySQL	database
Oracle	schema
PostgreSQL	database
Progress OpenEdge	database
SQLite	database
Sybase	database

When you are connected to a data source in the Online Browser, for some of the databases you can select the root object whose contents you want to browse. The following procedure uses MS SQL Server as an example.

To select a root object in the Online Browser:

1. Click the arrow icon ◀ that is next to the currently selected database. A list of the root objects contained in the data source appears.
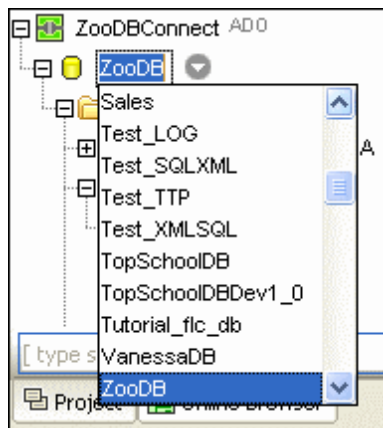
2. Select the desired root object from the list.

6.5   Displaying the Design of an Element

The Online Browser also allows you to view tables and their relationships in the graphical Design Editor. You can display one or more tables in a new Design Editor window or add one or more tables to a Design Editor window that is already open.

To show an object in a new Design Editor window:

* Right-click the object whose design you want to have displayed in the Design Editor and select **Show in new Design Editor** from the context menu.
  A Design Editor window opens and the selected object appears in it.

  Please note: Any number of tables can be selected in the Online Browser, use **Shift +Click** to select a contiguous group, or **Ctrl+Click** to select individual tables.

To add an object to the active Design Editor window:

* Right-click the object and select **Add to Design Editor** from the context menu.

6.6   Locating Objects

To find a specific database item by its name, you can either use DatabaseSpy's filtering functions or the Object Locator which appears as a drop-down list at the bottom of the Online Browser tab.

Showing referenced tables

The Online Browser provides an option in the context menu for constraints that allows you to show the table that is referenced by the selected foreign key. This way, you can identify the referenced table with a single mouse click. Both the foreign key and the referenced table are then selected in the Online Browser and you can display the reference in a Design Editor window, if required.
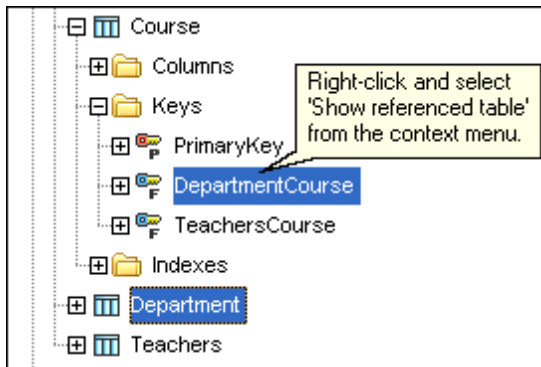
To show referenced tables in the Online Browser:

1.   In the Online Browser, right-click a foreign key in the Constraints folder of a table.

2.   Select **Show referenced table** from the context menu.
     The foreign key as well as the table it references are selected in the Online Browser.

6.6.1   Applying Filters

In DatabaseSpy's Online Browser, it is possible to filter schemas, tables, and views by name or part of a name. Objects are filtered as you type in the name. Filtering is case-insensitive by default, however you can also use case-sensitive filters by activating the corresponding check-box in the General options.

Please note: The filter function does not work if you are using No Folders layout.

To filter objects in the Online Browser:

1.   Click the **Filter Folder contents** [icon] icon in the toolbar or select a database object and press **Ctrl+Ctrl+Alt+F** to activate filtering. Filter icons appear next to all folders that appear in the currently selected layout.
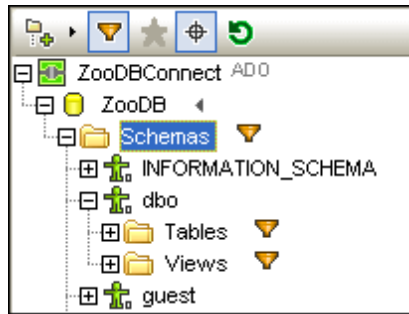
2. Click the filter icon that is next to the folder you want to filter. Select the desired filtering option from the popup menu that appears.

3. Next to the folder, the icon for the chosen filter type appears with an empty field next to it.
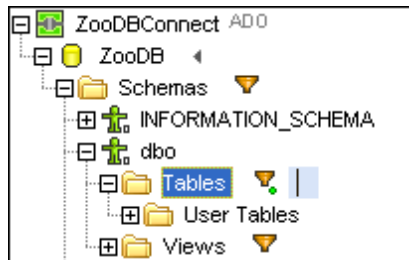
4. Expand the folder you are filtering to display the objects it contains.

5. In the field, enter the string you want to search for. The results are adjusted as you type.
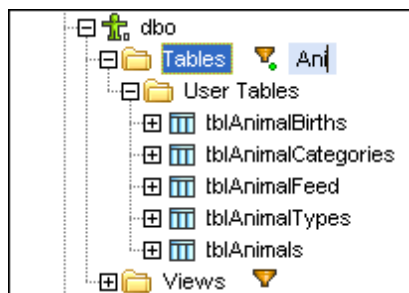
6.6.2   Using the Object Locator

You can use the Object Locator in different ways. Either use the drop-down list to select one of the objects contained herein, or type a string in the text field of the drop-down list and further filter the list of objects contained in the list. Filtering is case-insensitive by default, however you can also use case-sensitive filters by activating the corresponding check-box in the General options.
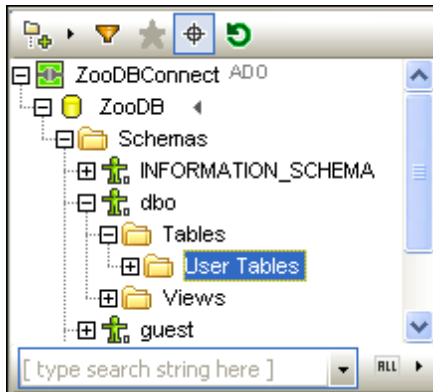
The icon next to the Object Locator drop-down list opens a menu allowing you to further define the displayed database objects.
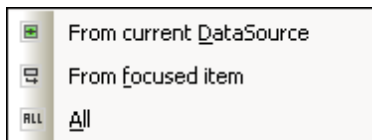
Using this menu, you can restrict the display of objects as follows:

▪   **From current Data Source**: Displays only objects from the currently selected data source.

▪   **From Focused Item**: Changes the display dynamically, depending on the folder or object that is currently selected.

▪   **All**: Displays all objects from all data sources that are currently connected.

To find database elements using the Object Locator:

1.   If the Object Locator is not enabled, click the **Object Locator** icon in the Online Browser or press **Ctrl+L**.

2.   Optionally, change the Object Locator context by clicking the arrow icon next to the drop-down list and select one of the options.

3.   Enter the string you want to look for, e.g., "type".
       Clicking the drop-down arrow displays all elements that contain that string.
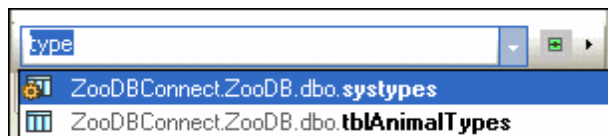
4.   Click the desired object to have it selected in the Online Browser.

6.7   **Hiding Unselected Items**

For a better overview in large databases, the **Select Database Objects for Comparison** dialog box allows you to show only selected database items in the individual Source group boxes. This saves you from scrolling through the list of tables and columns and you can easily check whether all items you want to compare are properly selected.

The `Show checked objects only` check box can be activated separately for both databases to be compared. When this option is checked, all unselected items are hidden in the group box and only the items that have been selected for comparison are displayed.

Chapter 7

Designing Databases

7 Designing Databases

The DatabaseSpy Design Editor lets you view and edit the structures of all your databases through one graphical user interface where you can examine tables and relationships in an existing database to understand them more easily.

You can also edit the database or even add entire tables and specify all their column attributes from scratch. In this case, an SQL script that can subsequently create this design in your database is automatically generated in the Database Structure Change Script window ready for execution.

Designs that have been created in DatabaseSpy's Design Editor can be saved for later re-use or can be sent to a printer.

The only precondition for working with the Design Editor is a data source connection to an existing database, i.e., you need at least an empty database to which you can connect. Design Editor's graphical user interface allows you to add tables or columns to a database, edit their properties, and delete columns or entire tables. In addition, you can also create a primary key, foreign keys, unique keys, as well as indexes and check constraints.

Any changes you make to the database structure are not implemented immediately but recorded in a change script which is displayed in the Database Structure Change Script window below the Design Editor window. You must execute the change script in order to actually change the database design.

To be able to create new designs from scratch, you must first open a new **Design Editor** window.

You can start the Design Editor by clicking the **Design Editor** 🔳 icon in the Standard toolbar, or selecting the menu option **File | New | Design Editor**, or pressing **Ctrl+D**. As soon as a Design Editor window is active, also the Design Editor menu is available. Each Design Editor window furthermore provides a toolbar of its own which contains various layout functions.
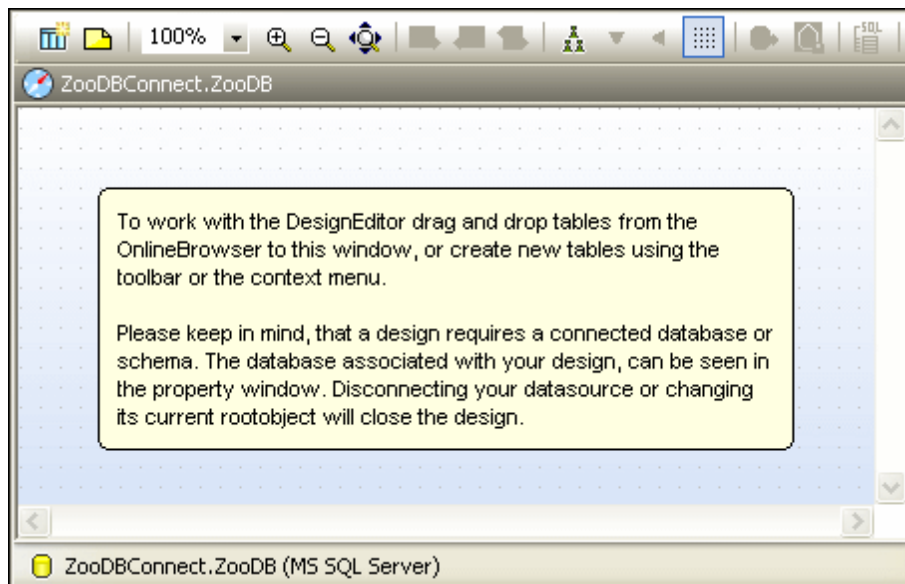
To open a new Design Editor from the Standard toolbar:

- Making sure that you have a data source connected, click the **Design Editor** ⬚ icon in the Standard toolbar, or select the menu option **File | New | Design Editor**, or press **Ctrl +D**.
  A new, empty Design Editor window opens.

Note: You require a connection to a data source to be able to start the Design Editor. The data source to which the active design window is assigned is displayed in the bottom left area of the Design Editor window and in the Execution Target Bar (if visible). You can change the data source of the active Design Editor from the Properties window, provided that no objects have been added to the active Design Editor. Only connected data sources are available in the drop-down list. Working with the Design Editor in the Offline mode is not possible.
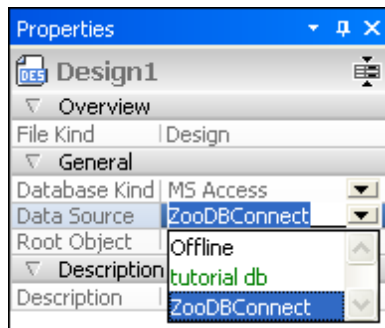
If you have more than one data source connected in your project, a new Design Editor window will always be connected to the active data source, i.e., the data source that is selected in the Online Browser.

To change the data source assigned to a Design Editor window:

1. Click into the Design Editor window whose properties you want to change.
2. In the Properties window, select the data source and, if applicable, default schema from the respective drop-down lists.

   Alternatively, if the Execution Target Bar is visible in the Design Editor, click the data source or root object in the Execution Target Bar to jump directly to the corresponding field in the Properties window.

   Note that only connected data sources are available in the drop-down list.

7.1   Adding Tables

When the Design Editor is connected to a data source, you can either show the design of existing tables by dragging them into the Design Editor window or by using a context menu command to do so, or you can create an entirely new table from scratch. Design Editor's table design shows the most important properties of the table and you can add indexes, keys, and check constraints with a few mouse clicks.

To show an existing table in the Design Editor:

Do one of the following:

- Right-click the table you want to examine in the Online Browser and select **Design Editor | Show in new Design Editor** from the context menu.

- If a Design Editor window is already open, you can also right-click the table and drag it into the Design Editor window or use the context menu command **Design Editor | Add to Design Editor** to add the table to the design.

To create a new table in Design Editor:

1. Optionally, choose the menu option **File | New | Design Editor** or press **Ctrl+D** to open a new Design Editor window. Alternatively, click the **Design Editor** icon in the Standard toolbar.

2. Click the **New Table** icon in the Design Editor toolbar or press **Ctrl+T**. Alternatively, right-click into the Design Editor and choose **Create new Table** from the context menu.

3. DatabaseSpy displays a message informing you that a change script has been created. Optionally, activate the `Don't show this dialog again!` check box and click **OK**.

4. In the table design, double-click the title bar to edit the table name.

5. To complete the table definition, modify the column properties, add columns, define a primary key, add a foreign key relationship, or define an index.

6. In the Database Structure Change Script window, click the **Execute** button to commit the changes to the database-

7.2   Selecting Tables

Dragging on the Design Editor background creates a marquee which selects all the tables it encloses. The context menu of the Design Editor also contains several options in the **Select** sub-menu which allow you to select specific sets of tables.

To select specific sets of tables:

To select only user tables, do one of the following:

- Right-click somewhere in the Design Editor window and choose **Select | Select User Tables** from the context menu.
- Click somewhere in the Design Editor window and press **Alt+T**.
- Select the menu option **Design Editor | Select User Tables**.

To select only system tables, do one of the following:

- Right-click somewhere in the Design Editor window and choose **Select | Select System Tables** from the context menu.
- Click somewhere in the Design Editor window and press **Alt+S**.
- Select the menu option **Design Editor | Select System Tables**.

To select all tables, do one of the following:

- Right-click somewhere in the Design Editor window and choose **Select | Select All** from the context menu.
- Click somewhere in the Design Editor window and press **Ctrl+A**.
- Select the menu option **Design Editor | Select All**.

To select several individual tables:

- Click a table and hold down the **Ctrl** or **Shift** key while selecting additional tables.

7.3   Opening, Saving, and Printing Design Files

You have several possibilities to open an existing design file in a Design Editor window:

- Any file that is stored in the file system can be opened via the **File | Open | Open File...** menu (**Ctrl+O**) and will be displayed under its file name in a new Design Editor window.
  You may have to select or connect to the data source to which the design is assigned or change the data source's root object prior to opening the design.

- A design file that has already been added to the project can be opened from the Project window and will be displayed under its file name in a new Design Editor window.

To open a design file assigned to a connected data source:

1. Select the menu option **File | Open | Open File...** or press **Ctrl+O** to display the standard Windows **Open** dialog box.

2. Select a design file and click **Open**.

3. The design file is opened in a new Design Editor window under its file name.

To open a design file assigned to a disconnected data source:

1. Select the menu option **File | Open | Open File...** or press **Ctrl+O** to display the standard Windows **Open** dialog box.

2. Select a design file and click **Open**.

3. If the data source connection is available in the project but not connected, a message box is displayed.
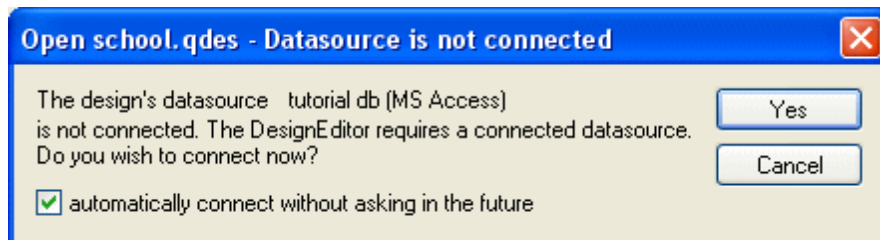
4. Optionally, activate the `Automatically connect without asking in the future` check box.
   This also activates the `Automatically connect data sources` check box in the Design Editor options.

5. Click **Yes** to open the design.

6. The design file is opened in a new Design Editor window under its file name.

To open a design file assigned to a missing data source:

1. Select the menu option **File | Open | Open File...** or press **Ctrl+O** to display the standard Windows **Open** dialog box.

2. Select a design file and click **Open**.

3. If the data source connection is not included in the project, a dialog box appears allowing you to choose or add a connection for the design.
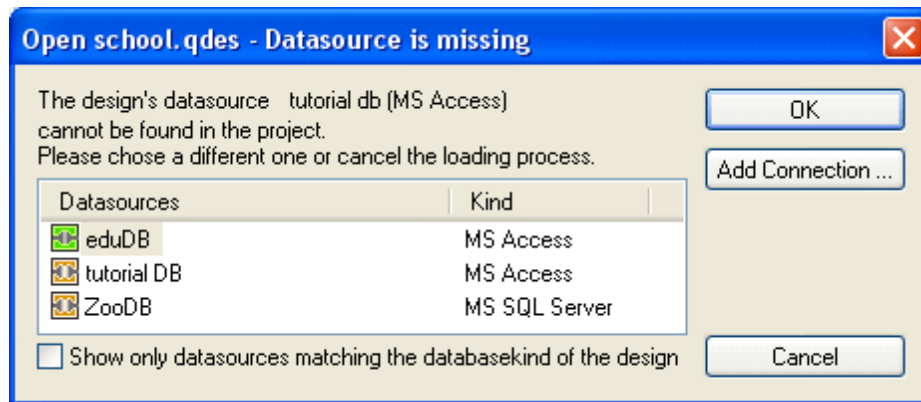
4. Optionally, activate the `Show only data sources matching the database kind of the design` check box to limit the display to matching data sources.

5. Do one of the following:

    - Select a matching connected data source and click **OK**.

    - Select a matching disconnected data source and do the following:

        1. In the **Open *.qdes - Data source is not connected** dialog box click **OK**.

        2. Optionally, activate the `Automatically connect without asking in the future` check box.
           This also activates the `Automatically connect data sources` check box in the Design Editor options.

        3. Click **Yes**.

    - Click the **Add connection** button to add a data source to the project.

6. The design file is opened in a new Design Editor window under its file name.

To open a design file assigned to a different root object:

1. Select the menu option **File | Open | Open File...** or press **Ctrl+O** to display the standard Windows **Open** dialog box.

2. Select a design file and click **Open**.

3. If applicable, select and connect to a data source.

4. In the **Open *.qdes - Root object is not active** dialog box click **Yes** to change the root object in the data source connection to the root object the design is assigned to.
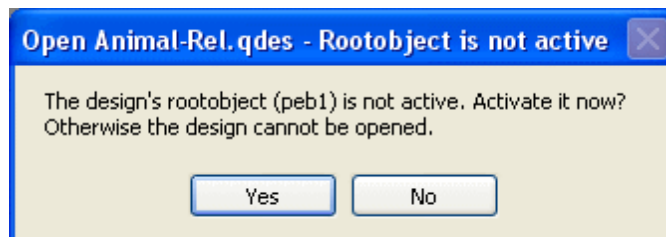
5. The design file is opened in a new Design Editor window under its file name.

To open a design file from the Project window:

- In the Project window, in the Design folder do one of the following:

    - Select a design file and double-click the file name.

- Right-click the file and select **Open** from the context menu.

You may have to select or connect to the data source to which the design is assigned or change the data source's root object in order to display the design in a new Design Editor window.

Saving design files

DatabaseSpy provides several options for saving designs in the Design Editor:

- **Design file (*.qdes)**: These files can only be opened in DatabaseSpy.

- **Image (*.png, *.emf)**: Designs that have been saved as image can be opened with any picture viewer.
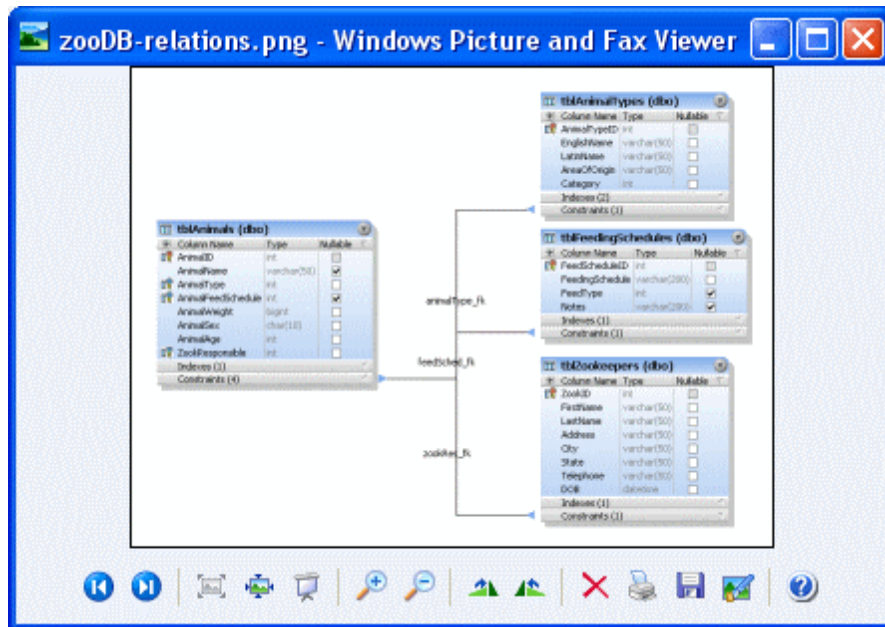
Saved design files can also be added to the project and appear then in the Design folder of the Project window.

To save a design as design file:

1. Make the Design Editor window that you want to save the active window.

2. Do one of the following:

   - Right-click the tab of the Design Editor window and choose **Save** or **Save As...** if you want to save a design file under a new name or path.

   - Press **Ctrl+S**.

   - Click the **Save** button in the Standard toolbar.

To save a design as image:

1. If you have maximized the Design Editor window, click the **Restore** button in the upper right corner of the Design Editor.

2. Resize the window and arrange the table(s) as desired.
   The borders of the window define the white space that surrounds the image.

3. To save the image, do one of the following:

- Click the **Save Diagram as Image** icon in the toolbar of the Design Editor window.

- Choose the menu option **Design Editor | Save Diagram as Image**.

    The standard Windows **Save As** dialog box appears.

4. Enter name and path for the image.

5. In the Save as type drop-own list, select whether you want save the file as portable network graphic (PNG) or enhanced metafile (EMF).

6. Click **Save**.

Printing Designs

Designs in the Design Editor can also be printed. You can print out an entire design or select a number of objects to print only the selection. In the General options, you can define whether or not the program logo is to be printed on top of the page.

A print preview is available for you to check the layout of the printed page before sending it to the printer.
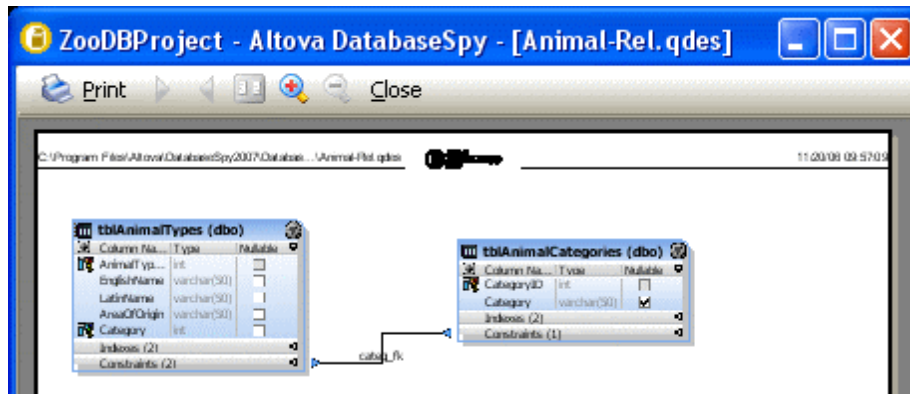
To print a design file:

1. Click in the Design Editor window to make it active.

2. Select the menu option **File | Print...**, or press **Ctrl+P**, or click the **Print** icon in the Standard toolbar.
    The **Print** dialog box appears.
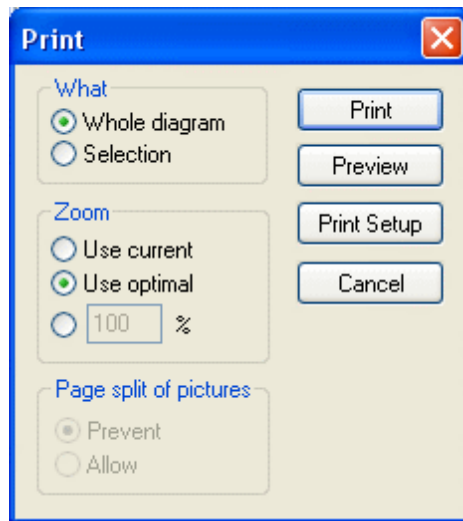
3. In the What group box, define whether to print the whole diagram or only the selected objects.

4. Choose the appropriate zoom factor in the Zoom group box.
    The `Use optimal` option ensures that the design is printed on one page.

5. Optionally, define the printer setup by clicking the **Print Setup** button.

6. Optionally, click the **Preview** button to preview the print-out before actually sending it to the printer.

7. Click **Print**.

To preview a design file before printing:

1. Click in the Design Editor window to make it active.

2. Select the menu option **File | Print Preview**, or press **Ctrl+P**, or click the **Print** icon in the Standard toolbar.
    The **Print** dialog box appears.

3. In the **Print** dialog box, click **Preview**.

4. In the print preview do one of the following:

    • Click the **Print** button to print the file.

    • Click **Close** to return to the **Print** dialog box.

Chapter 8

Retrieving and Editing Data

8   Retrieving and Editing Data

In DatabaseSpy, you can retrieve database data directly from the Online Browser or from a table design in the Design Editor using the **Retrieve data** option in the context menu. The required query is generated and executed with one single mouse click and the data is displayed in a Result tab of the SQL Browser. You can choose to retrieve all rows at once or only a particular number of rows at a time.

If you need to edit data after retrieving it, DatabaseSpy provides the **Edit Data** option in the context menu of the Online Browser for this purpose. The Result window is switched into the Editing mode in this case, indicating this condition in its status bar.
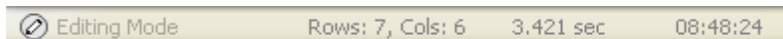
You can also select individual columns of a table when using the **Retrieve data** and **Edit data** commands. Only data from these columns will be retrieved in this case. Note, however, that all columns may be retrieved if you have deactivated the **Generate SELECT statements with full column list** check box in the SQL Generation options.

To retrieve data:

- In the Online Browser, right-click the object(s) you want to retrieve data from and select **Retrieve data | All rows** (**Ctrl+Alt+R**) or **Retrieve data | First n rows** (**Ctrl+Alt+T**) from the context menu.
- In the Design Editor, right-click either the title bar or one or several individual columns (using Ctrl+click) in a table design and select **Retrieve data** | **All rows (Ctrl+Alt+R)** or **Retrieve data** | **First n rows (Ctrl+Alt+T)** from the context menu.

   An SQL Editor window opens which displays the query for the data retrieval. If you have selected objects from different tables, separate SELECT statements for the individual tables are created. The results of the query are displayed in the Result window below.

To retrieve data for editing:

- Provided that direct editing of database data is supported for the respective data source connection, right-click a table or column in the Online Browser and select **Edit Data** from the context menu.

Partial retrieval

In the SQL Editor options, you can define the number n of rows to be retrieved if you select the **Retrieve data | First n rows** option from the context menu. You can then use the toolbar icons of the Result window to retrieve the data in fractions.

Stopping the retrieval

While data is being retrieved, the status bar of the Result tab counts up the rows already retrieved and displays the execution time. In addition, the tab of the respective SQL Editor window shows an icon at the left side, indicating that data is still being retrieved.

While data is being retrieved, the **Execute** button is replaced by the **Stop Retrieval** button in the SQL Editor toolbar. You can stop the retrieval process at any time by clicking the **Stop Retrieval** button. The status bar indicates that the retrieval has been stopped.

To stop the retrieval of database data:

- Click the **Stop Retrieval** button which is visible in the SQL Editor toolbar while data is being retrieved.

To resume the interrupted retrieval of database data:

Do one of the following:

- To retrieve the next n rows of data, click the **Retrieve next n rows** icon or right-click anywhere in the Result tab and select **Retrieve next n rows** from the context menu.
- To retrieve all outstanding data, click the **Retrieve outstanding rows** icon or right-click anywhere in the Result tab and select **Retrieve outstanding rows** from the context menu.

Editing database data in the Result window

To be able to edit the data contained in your database tables, DatabaseSpy provides two different approaches: You can either generate the corresponding SQL statements and execute them in the SQL Editor or, more conveniently, retrieve data and edit it directly in the Result window.

The Result window has to be enabled for editing before you can make any changes to the data displayed in the result grid. For this purpose, the context menu of the Online Browser as well as the context menu that opens when you right-click into a table design in the Design Editor provides the **Edit Data** command which is available on table and column level. This command generates a

SELECT statement which appears in a new SQL Editor window and is executed immediately, thus switching the Result window into the Editing mode.

In addition, also the **Execute for Data Editing** button is available in the SQL Editor toolbar, allowing you to execute a SELECT statement and, at the same time, switch the Result window into the Editing mode. This is particularly useful if you have already retrieved data and now want to update some of the records. The table below shows the editing options for the individual data source connections.

To select data for editing in the Result window:

Do one of the following:

- In the Online Browser or in a Design Editor window, right-click the table or column you want to edit, and choose **Edit Data** from the context menu.
  A SELECT statement is created and executed automatically in the SQL Editor.
- Generate a SELECT statement in a new SQL Editor window and click the **Execute for Data Editing** button, if available for the currently used data source connection.

Please note: Not all editing options (update, insert, or delete row) may be available, depending on the data source connection you are currently using, and on whether the table you are editing has or has not a primary key defined.

The following restrictions apply when editing data in the Result window:

- Only SELECT statements may be present in the SQL Editor window if the **Execute for Data Editing** command is used. If you click the **Execute for Data Editing** button while any other statement is in the SQL Editor, the statement will be executed without switching the Result window into the Editing mode.

- Tables that do not have a primary key defined are only partly supported for editing. DatabaseSpy displays a warning dialog box if you select the **Edit Data** command for such a table.
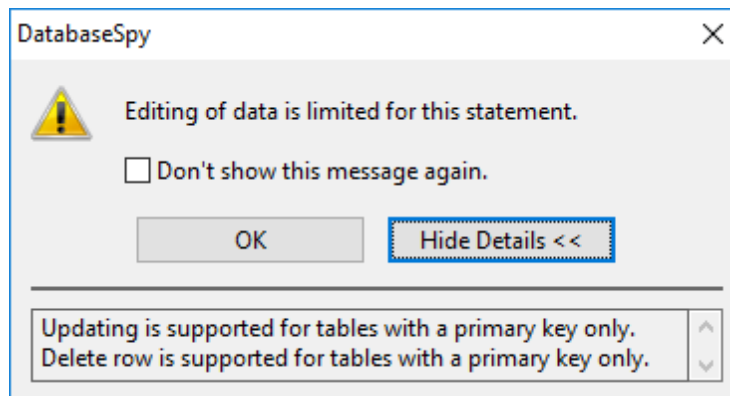
The **Show Details >>** button allows you to display the reason why the editing command has failed. If you do not want to have this dialog box displayed, select the **Don't show this dialog again!** check box or disable this feature in the SQL Editor options. Upon clicking **OK** in the dialog box, DatabaseSpy executes the statement and disables the editing functions in the Result window. You can then, however, insert new rows into such

tables or save the content of XML fields as XML files.

- All columns that form part of the table's primary key must be included in the statement that is used for populating the result grid with records. A warning dialog box similar to the one above ("Editing of data is limited for this statement") will be displayed if you omit such a column in your SELECT statement.

- Columns that are the result of an equation or have certain properties such as autoincrement, timestamp, etc., are excluded from editing. If such columns are present, a warning dialog box similar to the one above appears before the results are displayed. You can, however, still edit the remaining columns of the table.

8.1   Viewing Results

When an SQL Editor window includes more than one statement, the results of a query are by default displayed in separate result tabs with consecutive numbers: Result1, Result2, etc. To change this default display mode, you can activate the **Show multiple results stacked** button in the SQL Editor toolbar. When executing a script containing more than one query, only one result tab is displayed, and the results of the individual queries are displayed in separate panes of that tab. Note that you can change the display mode also after the query has been executed.

To display multiple queries in one result tab:

1. In the SQL Editor toolbar, activate the **Show multiple results stacked** button.

2. Click the **Execute** button or press **F5**.
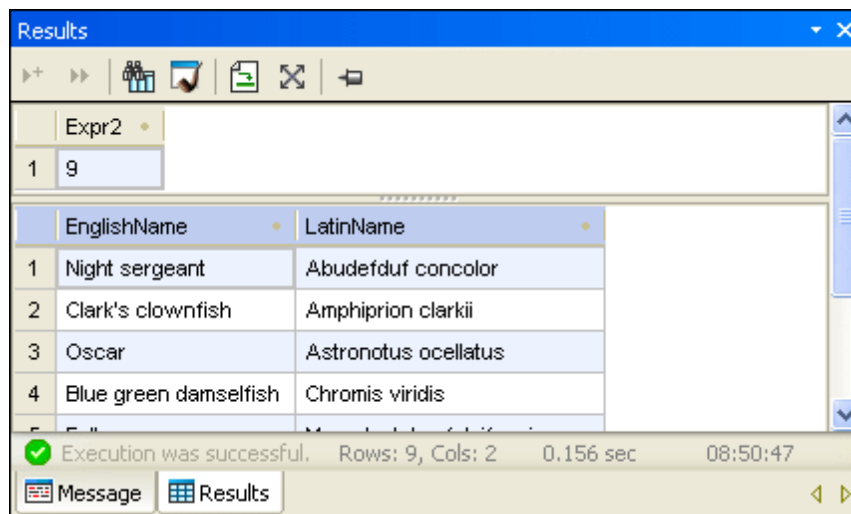
Only one result tab has been created. It consists of two panes however, each containing the result of the individual queries.

Viewing statistical data

The status bar of the Result window can display statistical information about any cells selected in the result grid, which includes:

• Number of selected cells (in the status bar, this statistic is shown as **Count**)
• Number of selected cells that have a numerical data type (**Numeric count**)
• The average value of all selected numeric cells (**Average**)
• The minimum and maximum value of all selected numeric cells (**Min** and **Max**)
• The sum of all selected numeric cells (**Sum**)
• The data type of a cell.

To view statistical information in the Result window, select one or more cells in the result grid. The values for the activated statistical fields are displayed in the status bar.

Not all statistics are enabled by default. To enable or disable a particular statistic, right-click the status bar and select the corresponding option from the context menu.

Note that the **Count** option displays the number of all selected cells, while the **Numerical Count** omits cells that do not have a numerical data type. Naturally, the **Average**, **Min**, **Max**, and **Sum** values are based on the numerical count.

When a single cell is selected, its data type is additionally displayed in the status bar.

Preserving results

If you want to keep a particular result, you can pin the corresponding tab in the Result window.

DatabaseSpy offers a toolbar button as well as several options in the context menu that opens when you right-click a tab in the Result window for this purpose. When the button is toggled on, the Result tab remains visible even if different queries are executed and new Result tabs are generated. Pinned result tabs show a different icon.
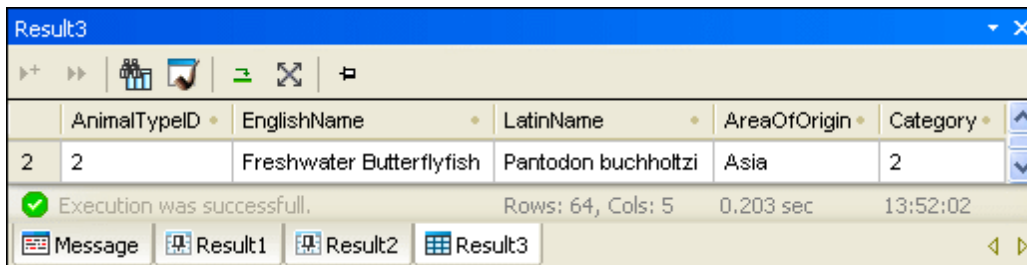
Note that pinned results show a different icon in the tab. When viewing pinned results, please bear in mind that the SQL statement displayed in the SQL Editor window above the Result tab might have been changed since the query has originally been executed and that re-executing the query may display a different result.

To pin a result tab:

1. Execute a query and display its results in the Result window.

2. In the Result window, do one of the following:

- To pin an individual result, click the **Keep result visible** button in the toolbar of the respective Result window or right-click the tab and select **Keep result visible** from the context menu.
- To pin all results that are currently displayed in the Result window, right-click any result tab and select **Keep all results visible** from the context menu.

Any pinned tab is kept in the Result window even if new queries are executed, and new Result tabs are generated.

To release a pinned result tab:

Do one of the following:
- To unpin a result, click the **Keep result visible** icon again, or right-click a pinned result tab and choose **Remove permanent visibility for result** from the context menu.
- To unpin all results in the Result window, right-click any result tab and select **Remove permanent visibility for all results** from the context menu.

8.2   Viewing Large Data Cells

In data cells containing a large amount of text, the Result tab displays only the first part of the text followed by an ellipsis (...). In this case, also the **Auto Size Columns** option in the context menu is limited to displaying 250 pixels. You can, however, always drag the column border to show more text.

Expanding cells temporarily

If cell content is larger than the currently visible cell width (e.g., in tables with a large number of columns), the width of a certain cell is temporarily extended if you move the mouse cursor over the data cell (*see screenshot below*). This applies also to column headings.

	Ani...	AnimalN...	Ani...	A...	Ani...	Animal...	A...	Z...
45	5044	Penny	44	2	1	female	4	7
46	5045	Auntie	31	2	105	female	10	9
47	5046	Snuffelupigus		2	134	male	11	10
48	5047	Walter	1	3	12	male	6	8
49	5048	Maggie	1	3	9	female	3	8

To expand single data cells temporarily:

- Place the mouse cursor over the cell you want to expand.

Using the Data Inspector

Where the temporary expansion of long data cells is insufficient, for example if you want to check the content of XML columns, DatabaseSpy provides the Data Inspector to display data cells containing a large amount of data.
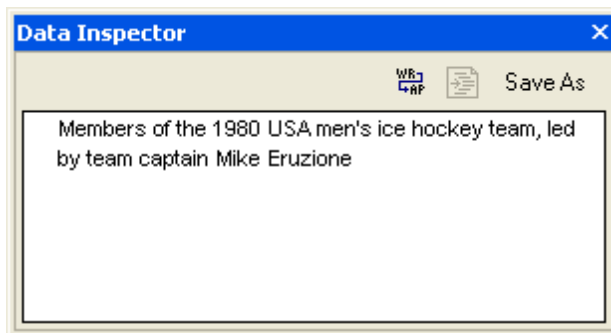
The Data Inspector window can be displayed by clicking the **Show the Data Inspector window** icon in the toolbar of the Result window or Database Data Comparison Result window, respectively, or by choosing the menu option **View | Data Inspector window**.
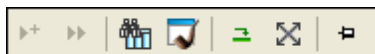
To view data using the Data Inspector:

1. In the Result window or Database Data Comparison Result window, select a data cell and click the **Show the Data Inspector window** icon in the toolbar. Optionally, select the menu option **View | Data Inspector Window**. The Data Inspector window opens and shows the content of the selected data cell.

2. Optionally, click the **Word wrap** ⊞ button to wrap the text in the Data Inspector window.

3. Optionally, click the **Pretty Print** ⊞ button to display the text in a hierarchical way.

4. Optionally, click the **Save As** button to save the content as text or XML.

8.3   Searching and Sorting

Data that has been retrieved from a database is displayed in DatabaseSpy's Result window. If you are searching for a particular string in your data, you can either create an appropriate SELECT statement using a WHERE clause to retrieve only rows that include that string, or, you can make use of DatabaseSpy"s search functionality directly in the Result window. This allows you to find also parts of words in various columns included in the result of a database query.
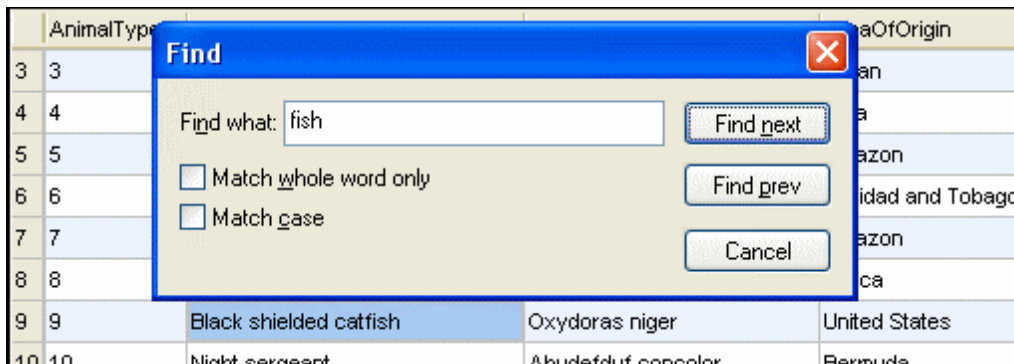
You can call the **Find** dialog box by clicking the **Find** button in the Result window toolbar, or by pressing the standard shortcut **Ctrl+F** when the cursor is located in the Result window.

Please note that the **Find** button is also available when the Result window is in the Editing mode.

To find a string in the Result window:

1.   Retrieve data from the table you want to search.

2.   In the Result window, click the **Find** button to display the **Find** dialog box. Alternatively, you can also click into the result grid and select the menu option **Edit | Find...** or press **Ctrl+F**.

3.   Type the string you are looking for into the `Find what` field.

4.   Optionally, activate the `Match whole word only` check box if you want to restrict your search to entire words that match the search term.

5.   Optionally, activate the `Match case` check box if you want to consider the case when searching.

6.   Click **Find next** to start the search.
      The first occurrence of the search term is highlighted in the result grid.

7.   Optionally, click **Find next** again to jump to the next occurrence of search term or click **Find prev** to jump to the previous occurrence of the search term, if available.

8.   To quit the search, click **Cancel**.

Sorting results

By default, the data is sorted according to its primary key column in ascending order. You can change the sort order in the result grid using the context menu, or by clicking the sorting arrows in the column headers, provided that this option has been activated in the SQL Editor options.

If sorting symbols are displayed in the column headers, you can click them to toggle the sorting order in the sequence ascending – descending – default. The shape of the respective sorting symbol changes accordingly.

	BirthID	BirthDate	Mother	Father	Veterinarian	NumberInLitter
1	1	2005-01-16	5056	5055	4	3
2	7	2005-09-14	5051	5050	1	1
3	3	2005-11-07	5053	5054	6	4

In the example illustrated in the screenshot above, data is sorted in ascending order according to the `BirthDate` column.

To sort data in the result grid:

Do one of the following:

- Right-click anywhere in the column to be sorted and select either **Ascending** or **Descending** from the **Sorting** sub-menu of the context menu.

- Click the sorting symbol in any of the column headers to sort the data.
  The data is sorted according to the contents of that column in ascending order. Click the same column again to sort in descending order. A third click restores the default sort order.

To restore the default sort order:

- Right-click anywhere in the table and choose **Sorting | Restore default** from the context menu.

8.4   Printing Results

Data cells that you select in the result grid can be printed via the menu command **File | Print Preview**. This opens the selected cells in a separate pane, where you can zoom into or out of the grid and send the data to the printer. Alternatively, you can also use the **File | Print...** command, which opens the **Print** dialog box with the `Selection` option preselected in the Print range group box. Note that in the latter case no preview or zooming options are available.

	Employee ID	Supervisor ID	Last Name	First Name	Position	Birth Date
1	1	5	Davolio	Nancy	Sales Representative	1972-12-08
2	2	[NULL]	Fuller	Andrew	Vice President, Sales	1969-02-19
3	3	5	Leverling	Janet	Sales Representative	1971-08-30
4	4	5	Peacock	Margaret	Sales Representative	1973-09-19
5	5	2	Buchanan	Steven	Sales Manager	1975-03-04
6	6	5	Suyama	Michael	Sales Representative	1963-07-02
7	7	5	King	Robert	Sales Representative	1972-05-29
8	8	5	Callahan	Laura	Inside Sales Coordinator	1974-01-09
9	9	5	Dodsworth	Anne	Sales Representative	1976-01-27
10	10	2	Hellstern	Albert	Business Manager	1968-03-13
11	11	10	Smith	Tim	Mail Clerk	1973-06-06
12	12	10	Patterson	Caroline	Receptionist	1979-09-11
13	13	2	Brid	Justin	Marketing Director	1977-10-08
14	14	13	Martin	Xavier	Marketing Associate	1975-11-30

The example above shows the result of a retrieval of an employees table. You can select particular names in this result (e.g., all the managers) and print only their names and positions (*see screenshot below*). Only selected rows and columns will be printed.

	Last Name	First Name	Position
1	Fuller	Andrew	Vice President, Sales
2	Buchanan	Steven	Sales Manager
3	Hellstern	Albert	Business Manager
4	Brid	Justin	Marketing Director

8.5   Updating Data

To update data in the Result window, first select data for editing. This switches the Result window into the Editing mode, which is indicated by the "Editing Mode" message in the status bar.

To edit records, double-click the cell you want to edit and proceed like in any spreadsheet application. You can use the context menu to copy, cut, paste, delete, or select all content or to undo your editing action. You can also paste into the Result grid cell ranges copied from another Result grid or from external files such as Microsoft Excel spreadsheets (see also Adding and Copying Rows).

Result grid

The updated data is not immediately committed to the database but only displayed in the Result grid. To actually commit the update to the database, the Result window provides a **Commit** button in its toolbar.

Data cells that have been edited are indicated by a different background color. The color informs you if there are potential problems with the entered data. For example, if the entered text is not suitable for the data type of the column, the background color of the cell turns red.

If XML columns are present in the Result window, an additional button for each row appears inside the XML column: 🔳. Clicking this button allows you to load an XML file into this XML cell or to save the XML content of the respective cell as XML file. Furthermore, you can assign an XML schema to the XML cell, provided that the current data source connection supports this feature. In databases where XML columns are currently not supported by DatabaseSpy, this icon changes to 🔳 and you can only save the XML content of the respective cell to a file.

In the Editing mode, the toolbar shows the following additional commands:

+	Appends a new row. This command is also available in the context menu, when your right-click a cell. See also Adding and Copying Rows.
×	Deletes an existing row. This command is also available in the context menu, when your right-click a cell.

⬚	Undoes all changes made to the Result grid (provided that they have not been committed to the database yet). To undo the editing of a particular cell, right-click it, and select **Undo Changes for this Cell** from the context menu.
Def	Sets the currently selected cell(s) to the default database value. This command is also available in the context menu, when your right-click a cell. Note that the command (and its corresponding button) is enabled only if the database column provides a default value.
∅	Set the currently selected cell(s) to a `NULL` value. This command is also available in the context menu, when your right-click a cell. Note that the command (and its corresponding button) is enabled only if the corresponding database column allows nulls.

When you finished editing data, click **Commit** to send the changes to the database. DatabaseSpy automatically creates the required SQL statements and executes them. If the execution fails, you can view the statements in the Output window and check why they have not been executed successfully.
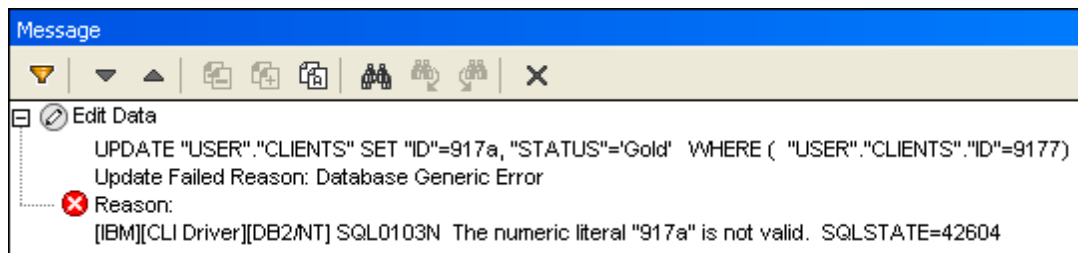
In such a case, the edited cell remains marked modified in the Result window, however if you choose to retrieve data once again, the (incorrect) update will be lost if you confirm the corresponding warning that pops up in a message box.

Using SQL

If you prefer to update data by means of SQL statements, DatabaseSpy can generate the required SQL statement with a click of the mouse (see also Generating SQL Statements). You can either update an entire row or only particular columns of a table. The corresponding SQL statement appears in an SQL Editor window where you can enter the new values.

```
1      UPDATE [Department] SET [id]=, [name]=''  ;
2
```

Take a look at the `Department` table in your Access tutorial database.

Let's assume you want to change the "Art" department to "Music" and the ID from "2" to "3", you would have to edit the SQL statement as follows:

```
1      UPDATE [Department] SET [id]=3, [name]='Music'
       WHERE [id]=2 ;
```

To update a table by means of SQL statements:

1. Connect to the database and show the respective data source in the Online Browser.
2. Optionally, right-click the table you want to update and choose **Retrieve data | All rows** to see the data that is contained in the table.
3. Do one of the following:
   - Select the respective table you want to update.
   - Expand the respective table to see its columns and select the column(s) you want to update.
4. Right-click and select **Show in new SQL Editor | Update** from the context menu or, holding down the right mouse button, drag the selected objects into an open SQL Editor window and choose **Update** from the popup.
   An UPDATE statement for all the selected objects is generated in the SQL Editor.
5. Enter the new values after the equals sign of the respective column.
6. Add a WHERE statement to select only a particular row.
   Note that **all** rows of the table will be updated with the same values if the WHERE statement is omitted.
7. Click the **Execute** button to execute the statement and to update the data in the database.

8.6   **Editing XML Columns**

Columns of type XML cannot be edited directly, however you can use the context menu to change the XML instance of an individual data cell. Clicking the Browse ▦ button that is located inside the XML column shows a popup menu allowing you to save the XML instance or load a different XML instance from the file system. It is also possible to [assign an XML schema](#) to the selected data cell.

	ID	NAME	STATUS	CONTACTINFO
1	3227	Ella Kimpton	Gold	▦ <?xml version="1.0" encoding="UTF-8" ?><Clie...
2	8877	Chris Bontempo	Gold	▦ <?xml version="1.0" encoding="UTF-8" ?><Clie...
3	9077	Lisa Hansen	Silver	▦ Load XML Document from File... ?><Clie...
4	9177	Rita Gomez	Silver	▦ Save XML Document to File... ?><Clie...
5	5681	Paula Lipenski	Standard	▦ Assign XML Schema... ?><Clie...
6	4309	Tina Wang	Silver	▦ <?xml version="1.0" encoding="UTF-8" ?><Clie...

The XML instances stored in an XML data cell in the database can be changed by selecting an XML file from the file system. DatabaseSpy presents the standard Windows **Open** dialog box for this purpose.

To change the XML instance of a database data cell:

1. Click the Browse ▦ button inside the data cell you want to edit.

2. From the menu that appears, select **Load XML Document from File...** or **Save XML Document to File...**.

3. Do one of the following:

    • In the **Open** dialog box, browse to the desired XML instance and click **Open**.

    • In the **Save As** dialog box, select the folder where you want to save the XML file, enter a name for the file, and click **Save**.

4. To commit your changes to the database, click the Commit button in the Result window toolbar.

8.7   Editing Binary Columns

Binary content can be stored in databases using the following data types:

Oracle	BLOB, LONG RAW, RAW, BFILE
MS SQL Server	binary, varbinary, image
MS Access	BINARY, IMAGE
MySQL	binary, tinyblob, varbinary, blob, mediumblob, longblob
IBM DB2	BLOB
IBM DB2 for i	BLOB, BINARY LARGE OBJECT, BINARY, VARBINARY
SQLite	BLOB
Sybase	binary, varbinary, image
PostgreSQL	byte array

In DatabaseSpy you can choose to (i) save binary content from the result grid to a file, or (ii) load binary content from a file and store it in the database. The respective menu entries are available when you click the 🔲 button inside a data cell of the result grid.

	id ▾	name ▾	photo ▾	status ▾
1	3227	Ella Kimpton	🔲 0xFFD8FFE00010...	Gold
2	8877	Chris Bontempo	🔲 *[NULL]*	Gold
3	9077	Lisa Hansen	🔲 0xFFD8FFE00010...	Silver
4	9177	Rita Gomez	🔲 Load Binary from File...	
5	5681	Paula Lipneski	🔲 Save Binary to File...	Standard
6	4309	Tina Wang	🔲 *[NULL]*	Silver

 DatabaseSpy presents the standard Windows **Open**, respectively the **Save As**, dialog box for this purpose. Binary content can be saved to the following formats BIN, BMP, JPG, PNG, MP3, AVI, and PDF.

To change the binary content of a database data cell:

1.   Click the Browse 🔲 button inside the data cell you want to edit.

2.   From the menu that appears, select **Load Binary from File...** or **Save Binary to File...**.

3.   Do one of the following:

   • In the **Open** dialog box, browse to the desired binary file and click **Open**.

   • In the **Save As** dialog box, select the folder where you want to save the binary file, enter a name for the file, and click **Save**.

4.   To commit your changes to the database, click the ⌈ Commit ⌉ button in the Result window toolbar.

Please note: MS Access does not support the loading of a binary file in a binary column.

Please note: In PostgreSQL, you have to activate the `bytea as LO` check box on the second page of the advanced data source options of the PostgreSQL driver settings to be able to store

binary data.
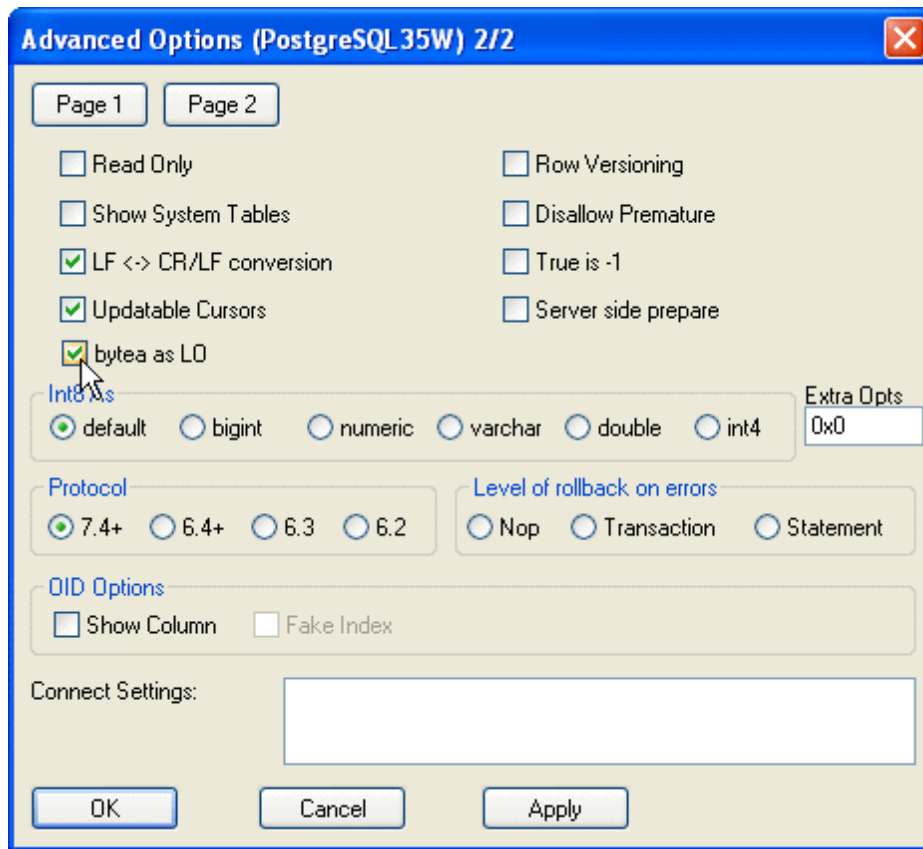
8.8   Inserting Data

In DatabaseSpy, you have several possibilities to populate your tables with data: You can import database data from CSV files, or generate SQL statements in the Online Browser and fill in the desired values, or you can select data for editing and add new rows of data in the Result window.

Importing data

This option is recommended when you need to write a large amount of data into the database. The data can be conveniently prepared in a spreadsheet application and is then imported via the **Import data to the database** dialog box.

Generating SQL

When you want to add only several rows of data or when small tables with only a few columns are concerned, you can also generate an SQL script in the SQL Editor and enter the values manually. You can enter data for a particular column or for the entire row of a table.

To insert data into a table:
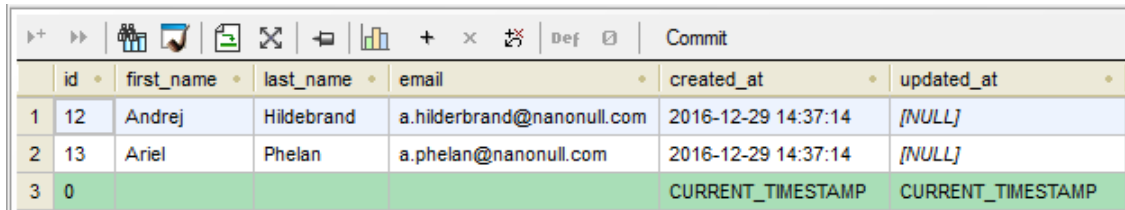
1. Connect to the database and show the respective data source in the Online Browser.

2. Right-click the table and select **Show in new SQL Editor | Insert** from the context menu or, holding down the right mouse button, drag the table into an open SQL Editor window and choose **Insert** from the popup.
   An INSERT INTO statement is generated in the SQL Editor.

3. Enter the desired values in the VALUES (, , ,) part of the statement (the number of commas depends on the number of columns in the respective table).
   If required, check the data type of the individual columns in the Properties window of the Online Browser.

4. Click the **Execute** ▶ button to execute the statement and to insert the data into the database.

5. Optionally, save the SQL statement if you need to insert the same kind of data frequently.

To insert data only into specific columns of a table:

1. Connect to the database and show the respective data source in the Online Browser.

2. In the Online Browser, expand the table you want to edit to and select the column(s) into which you want to insert data.

3. Right-click and select **Show in new SQL Editor | Insert** from the context menu or, holding down the right mouse button, drag the column(s) into an open SQL Editor window and choose **Insert** from the popup.
   An INSERT INTO statement is generated in the SQL Editor.

4. Enter the desired value(s) in the VALUES ( ) part of the statement.
   You may also have to enter the primary key for the new table row. Add the name and the value of the primary key to the statement in this case.
   Please note: If your table contains columns that have the Nullable check box deactivated, you must also enter values for these columns.

5. Click the **Execute** ▶ button to execute the statement and to insert the data into the database.

6. Optionally, save the SQL statement if you need to insert the same kind of data frequently.

8.9   **Adding and Copying Rows**

New rows can be added in the result grid of the SQL Editor, provided that data was retrieved in Editing mode, see Retrieving and Editing Data. The toolbar of the Result window provides the **Append a new row** command for this purpose. This command inserts an empty row where you can enter data and save the record using the **Commit** button.

	id	first_name	last_name	email	created_at	updated_at
1	12	Andrej	Hildebrand	a.hilderbrand@nanonull.com	2016-12-29 14:37:14	*[NULL]*
2	13	Ariel	Phelan	a.phelan@nanonull.com	2016-12-29 14:37:14	*[NULL]*
3	0				CURRENT_TIMESTAMP	CURRENT_TIMESTAMP

You can define in the SQL Editor options that the respective default values of the individual columns be inserted into new rows. In the example above, only the columns `created_at` and `updated_at` have a default value defined; DatabaseSpy hence populates them automatically. If the default property has not been set accordingly, a blank row will be inserted into the result grid.

In addition to editing rows manually, you can also paste into the result grid entire cell ranges (rows, columns, or both) from external files such as Microsoft Excel spreadsheets. The external cell ranges can be pasted over existing cells of the Result grid.

To paste clipboard data from external files into new rows, first click **Append a new row** ( ) to create the required number of rows, and then paste the clipboard cells at the desired location. The **Paste** command is available both as a keyboard shortcut (**Ctrl+V**) or from the **Edit | Paste** menu.

If you are pasting data from a different result grid of DatabaseSpy, it is possible to create the required number of new rows automatically as you paste data. To do this, right-click the Result window and select the **Paste as new rows** (**Ctrl+Shift+V**) command from the context menu.

Adding new records to incomplete selections of data
When not the entire table but a collection of individual columns is selected for data editing, the following restrictions apply when a new row is to be inserted into the result grid:

- The column(s) containing the table's primary key must be included to make data editing possible.
- All columns with the "Nullable" property not set must be included unless they have a default value defined.

Using the table below as example, let's assume that both the `NAME` and the `STATUS` columns are not nullable, but only the `STATUS` column has a default value ("Standard") defined.

We could further assume that each new customer is automatically assigned status "Standard" and thus omit the STATUS column when creating new customers. Adding new rows is possible in this case since the default value is automatically assigned when the record is committed to the database.

When checking the results of the entire table, you will see that the default value "Standard" has been automatically inserted into the Status column of the new record.

However, if you omit, for instance, the Name column, a warning message ("Editing of data is limited for this statement") is displayed when you select the **Edit Data** command. If you do not want to have this dialog box displayed, select the **Don't show this dialog again!** check box or disable this feature in the SQL Editor options. After clicking **OK**, you can still edit data in the

Result window but the **Append a new row** button and context menu option are disabled.

Copying rows

If you need to duplicate some of the data stored in your database tables, DatabaseSpy provides a feature allowing you to copy data from the result grid and append that content to the grid in a new row. You can either copy an entire row or individual cells. If you copy data from more than one row and append it as new rows, DatabaseSpy will create as many new rows as there were to host the original data.

To copy a row and append it to the result grid:

1. Make sure that the Result window is in the Editing mode.
2. In the Result window, right-click into the row you want to copy and choose **Selection | Row** from the context menu. Alternatively, click several individual data cells using **Ctrl +Click**.
3. Right-click anywhere into the Result window and choose **Copy selected cells** from the context menu or click **Ctrl+C**.
4. Right-click anywhere into the Result window and choose **Paste as new rows** from the context menu or click **Ctrl+Shift+V**.

Copying records and using data outside DatabaseSpy

You can use the clipboard to export selected records to any other application that supports the clipboard. It might be helpful to include also the headers of the selected data, even if only a fraction of a record set is copied. The context menu in the Result tab provides the **Copy selected cells with header** command for this purpose. The corresponding header will be copied to the clipboard together with each selected cell.

8.10   Deleting Data

To delete a row from a table, DatabaseSpy provides an option in the context menu of the Online Browser. Selecting this option generates a DELETE statement in a new or existing SQL Editor window. Be sure to edit this statement in order to delete the row(s) that are obsolete since executing the statement as is will not delete any rows from the selected table (the default statement selects rows where the primary key is NULL)!

To delete data from a table:

1. Connect to the database and show the respective data source in the Online Browser.

2. Optionally, right-click the table you want to update and choose **Retrieve data | All rows** to see the data that is contained in the table.

3. In the Online Browser, right-click the respective table and select **Show in new SQL Editor | Delete data** from the context menu or, holding down the right mouse button, drag the table into an open SQL Editor window and choose **Delete data** from the popup. A DELETE statement for the selected table is generated in the SQL Editor and a default WHERE statement is added.

4. Edit the WHERE statement accordingly to select only a particular row for deletion. **Please note**: All rows of the table will be deleted if the WHERE statement is omitted.

5. Click the **Execute** ![Execute button] button to execute the statement and to delete the data from the database.

Deleting data in the Result window

You have two options to delete records in the Result window: Either set a particular database field to NULL or delete an entire row. Both options are provided in the context menu that is available when the Result window is in the Editing mode. The affected fields or rows are marked modified until the changes are committed to the database. This enables you to also undo the editing if the need should arise.

	ID	NAME	STATUS	CONTACTINFO
1	3227	Ella Kimpton	Gold	`<?xml version="1.0" encoding="UTF-8" ?><Clie...`
2	8877	Chris Bontempo	Gold	`<?xml version="1.0" encoding="UTF-8" ?><Clie...`
3	9077	Lisa Hansen	Gold	*NULL*
4	9177	Rita Gomez	Silver	`<?xml version="1.0" encoding="UTF-8" ?><Clie...`
5	5681	Paula Lipenski	Standard	`<?xml version="1.0" encoding="UTF-8" ?><Clie...`
6	4309	Tina Wang	Silver	`<?xml version="1.0" encoding="UTF-8" ?><Clie...`

In the screenshot above, the CONTACTINFO field has been set to "NULL" for the third record, and the entire fifth record has been deleted. Note that the changes have not yet been committed to the database and can still be undone.

To set a database field to NULL:

1. Right-click the database field you want to change and select **Set Null** from the context menu.

2. Click the ![Commit button] button.

To delete a row of records in the result grid:

1. Right-click one of the fields in the row you want to delete and select **Delete Row** from the context menu or click anywhere into the row and click the **Delete row** ⊠ button. The row is marked for deletion (*see screenshot above*).

2. Click the Commit button.

To restore deleted rows in the database:

- Provided that the modified fields in the result grid have not been committed to the database yet, right-click into the deleted row and select **Undo changes for this Cell** from the context menu.
  The entire row of records is restored in the result grid.

Chapter 9

Working with SQL

9   Working with SQL

For working with SQL scripts, DatabaseSpy provides the **SQL Editor** which you can use to create, display, edit, and execute the SQL statements that are required for your database work. You can save SQL scripts that you have created in the SQL Editor and add them to your project. In addition, you can also open existing SQL files and display them in the SQL Editor.

The SQL Editor is started automatically when you open an SQL file or generate a statement from an existing database object. Apart from that, you can always open a new SQL Editor window as follows:

- Click the **SQL Editor** ![SQL icon] icon in the Standard toolbar, or select the menu option **File | New | SQL Editor**, or press **Ctrl+N**.

As soon as an SQL Editor window is active, the **SQL Editor** menu and toolbars are available. In addition, each SQL Editor window has a toolbar of its own which provides buttons for executing SQL as well as various text selection and display options discussed further in this chapter.

Note: A new SQL Editor window is automatically connected to the active data source, that is, the data source assigned to the window where the cursor was located before you opened the SQL Editor. If there is no data source connected, the SQL Editor window is in the Offline state which allows you to create SQL statements independent from any database connection. You can easily change the data source to which the SQL Editor window is connected, as shown below.

To assign a data source to an SQL Editor window:

1. Click into the SQL Editor window whose properties you want to change.
2. In the Properties window, select the data source, and, if applicable, the root object.
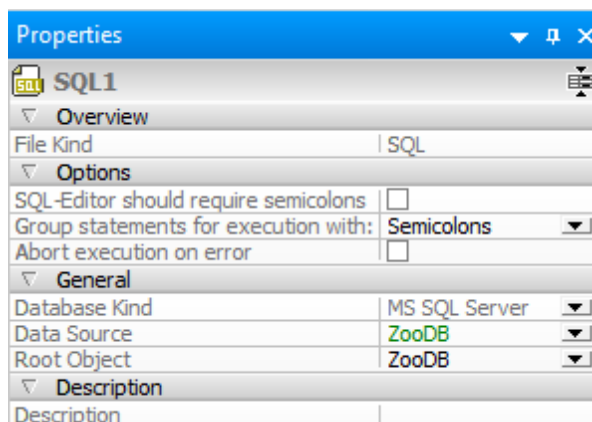
Note: To jump to the Properties window, click the **Unknown** or **Offline** hyperlinks in the

Execution Target Bar (provided that the **Show/Hide Execution Target Bar** button ![button] is toggled on). In the image below, no data source is assigned yet to the SQL Editor window, which is why the Execution Target bar displays "Unknown - Offline" as execution target.
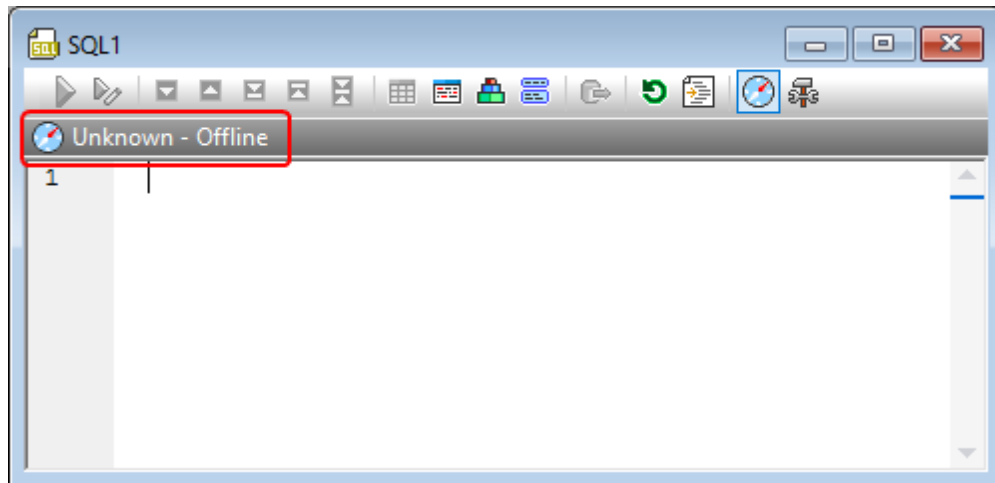
You can edit SQL statements in the SQL Editor like in any other text editor using the standard text manipulation commands such as **Copy** (**Ctrl + C**), **Paste** (**Ctrl + V**), and **Cut** (**Ctrl + X**). In addition, the SQL Editor provides features such as syntax coloring, comments, bookmarks, regions, and text highlighting (see SQL Editor Features). Also, the SQL Editor provides advanced Find and Replace functionality, including the ability to search and replace using regular expressions (see Finding and Replacing Text).

The auto-completion feature can assist you with entering statements into SQL Editor (see Auto-completion). You can also drag existing database objects into the SQL Editor window to create SQL statements automatically (see Generating SQL Statements).

When SQL Editor contains multiple statements, you can choose to execute them in bulk, if supported by the database type or driver, or sequentially (see Executing SQL). It is also possible to give custom descriptive names to the resulting tabs (see Naming Result Tabs).

The appearance of SQL statements in the SQL Editor can be configured in the **SQL Formatting Configurations** dialog box. For example, you can configure to generated SELECT statements so that a line break should always occur before each FROM keyword. The formatting syntax can be fine-tuned not only for each database kind, but also at statement level (see Formatting SQL).

When you need to export only a subset of data from the database, you can write or generate the required SELECT statement in the SQL Editor, and then proceed to exporting results (see Selecting Data for Export). SQL Editor can also be used to create views directly from SQL statements (see Views).

The text displayed in the SQL Editor can be printed and also previewed before printing (see Printing SQL).

9.1   Generating SQL Statements

In DatabaseSpy, you can automatically generate SQL statements based on existing objects displayed in the Online Browser. You can either drag a database object from the Online Browser into an SQL Editor window, or right-click a database object in the Online Browser and use the context menu. The objects from which you can generate SQL statements (or extract information such as name and path) are as follows:

- Tables ⊞
- Columns ⊞
- Keys 🔑
- Indexes 📑
- Triggers 🔺
- Views 👓
- Procedures 📇
- Functions *f*(x)

The following table shows which SQL statements can be generated (or information be extracted into the SQL Editor) for each database object type.

Select	⊞	⊞	-	-	-	👓	-	-
Name	⊞	⊞	🔑	📑	🔺	👓	📇	*f*(x)
Path	⊞	⊞	🔑	📑	🔺	👓	📇	*f*(x)
Create	⊞	-	-	📑	🔺	👓	📇	*f*(x)
Drop	⊞	⊞	🔑	📑	🔺	👓	📇	*f*(x)
Add	-	⊞	🔑	-	-	-	-	-
Alter	-	⊞	🔑	📑	🔺	👓	📇	*f*(x)
Rename	⊞	⊞	-	-	-	👓	-	-
Insert	⊞	⊞	-	-	-	👓	-	-
Update	⊞	⊞	-	-	-	👓	-	-
Delete data	⊞	-	-	-	-	👓	-	-
Execute	-	-	-	-	-	-	📇	-

Note: The syntax of the statements may vary depending on the database kind you are using.

Select	Creates a SELECT statement that retrieves data from (i) all columns of the source table, (ii) the selected column(s) of the parent table, or (iii) all columns that are included in a view. Basically, this is the same procedure as when selecting **Retrieve data** from the context menu, however the query is not automatically executed in this case.
Name	Returns the name of the selected database object. You can also select several objects. The names are printed in individual lines, separated by commas.
Path	Returns the full path of the selected database object (for example, `Database.Schema.Table.Column`). You can also select several objects. The paths are printed in individual lines, separated by commas.
Create	Generates a CREATE statement based on the selected database object.

	Depending on the selected database object, you have to edit this statement as follows: • Edit the **index** name as well as the properties, and enter the appropriate column to create a new index for the parent column of the selected index. • Edit the **trigger** name as well as the table that invokes the trigger, and enter the action that is to be triggered. • Edit procedure name and parameters, and adapt the body accordingly to create a new **stored procedure**. • Edit the **function** name as well as the body of the function. Define additional or remove parameters, if required.
Drop	Creates a DROP or ALTER statement, respectively, that deletes the selected database object from the database.
Add	Generates an ALTER statement that (i) adds a new column to the parent table of the selected column and uses the name and definition of the selected column as default values, or (ii) adds a constraint and uses the properties of the selected constraint as default. The default values have to be edited.
Alter	Generates an ALTER statement that allows you to edit the properties of the selected database object.
Rename	Allows you to rename the selected table, column, or view. Change this parameter to the desired new table name. **Please note**: This command is not available for Microsoft Access databases.
Insert	Allows you to insert data into the selected table, column, or view. By default, no default values specified. You must edit the statement and specify the values that are to be inserted into the table. When inserting data into individual columns, or if the primary key is not included in the view, you must also specify a value for the primary key unless an IDENTITY statement has been used for auto-generation of a primary key when the table was created.
Update	Creates an UPDATE statement that updates (i) all columns of the selected table, (ii) the selected column, or (iii) the columns that are included in the selected view. No default is specified for the new values. Note that this command will update **all** rows with the same values if no WHERE statement is added to select a specific row.
Delete data	Generates a DELETE statement that deletes data (i) from the selected table and adds a WHERE clause using the table's primary key to specify the row to be deleted, or (ii) from the parent table of the selected view. Note that this command will delete **all** rows of the table if no WHERE statement is added.
Execute	Declares the parameters needed for execution and creates an EXEC statement that executes the stored procedure with the required parameters.

Generating SQL statements with drag and drop

When you drag a table or column from the Online Browser and drop it into an SQL Editor window, a SELECT statement is generated by default. The kind of statement that should be generated for any database object is configurable in the <u>Online Browser options</u>. For example, you can configure that dropping a table into the SQL editor should generate an UPDATE statement, not a

SELECT.

To generate SQL statements using the default configured statement type:

- Drag and drop one or several database objects from the Online Browser into an existing SQL Editor window. An SQL statement appears in the SQL Editor. Note that one statement per table will be generated if you select objects from different tables.

You can also right-click a database object and drag it into an open SQL Editor window. In this case, when you release the mouse button, a context menu opens from which you can choose the desired statement type to be generated (for example, SELECT, INSERT, UPDATE, etc.).

To generate SQL statements of specific type:

1. Right-click one or several database objects in the Online Browser and use drag right to drop the object(s) into an existing SQL window.
2. Select from the context menu the type of SQL statement you want to generate.

You can define several additional options that determine how the generated SQL statement appears in the SQL Editor. In the example below, the **Append semi-colons to statement end** has been activated in the SQL Generation options. This causes the semi-colon character to appear at the end of the statement. Also, the **Enable SQL Formatting for the SQL Editor** has been deactivated in the SQL Formatting options. This causes the FROM keyword to appear on the same line as the SELECT keyword (if SQL formatting had been enabled, a line break would have been automatically inserted before the FROM keyword).
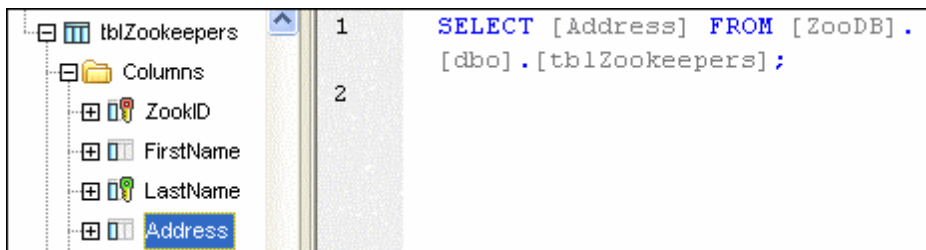
Generated SELECT statement

Generating statements in a new SQL Editor window
If you want your SQL statements to be generated in a new SQL Editor window, do the following:

1. Right-click a database object in the Online Browser and select **Show in new SQL Editor** from the context menu.
2. From the sub-menu, select the appropriate statement you want to use (for example, SELECT, INSERT, etc).

9.2   **Generating Complete DDL Scripts**

DatabaseSpy enables you to generate DDL (Data Definition Language) statements for all objects of a database or schema, as one SQL script. Generating the complete DDL scripts of a database or schema may be useful, for example, if you want to re-create the structure of that schema or database on another server or environment, or if you want to back up the current database structure, or if you want to create a text snapshot of the current database structure for comparing it later to another snapshot.

To generate complete DDL scripts from a schema or database:

1. In Online Browser, identify the folder that is the immediate parent of all "Tables", "Views", "Stored Procedures", etc folders of the current database or schema. For example, if you want to generate the complete DDL scripts of the "dbo" schema in the **ZooDB** tutorial database, this folder is "dbo". For more information about creating the tutorial **ZooDB** database, see Creating the Tutorial Database.
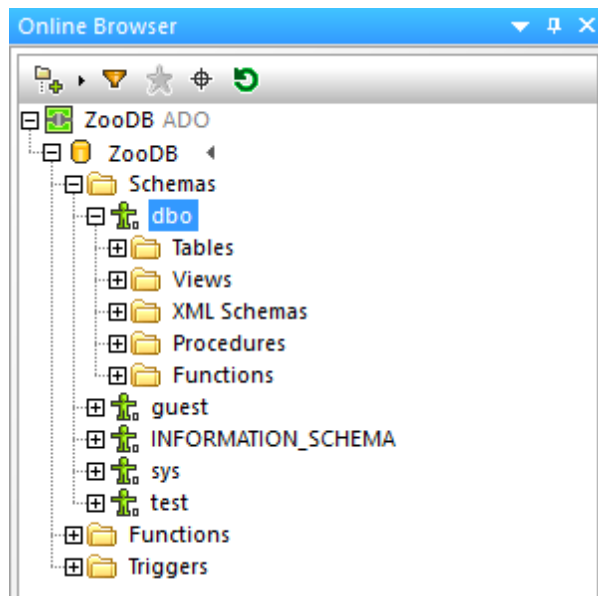
If the current database kind supports multiple schemas (like the **ZooDB** database above), it is possible to generate DDL scripts for multiple schemas in one operation (each schema in a separate file). To select multiple schemas, click the required schemas while holding the **Shift** key pressed.

Note that, in this example, the immediate parent of all tables, views, procedures etc is a schema. However, it may be the root database object (and thus have a different appearance) in other database kinds (since the structure displayed in the Online Browser depends on the kind of the currently connected database).

2. Right-click the folder identified in step 1 ("dbo", in this example), and select the command **Show in new SQL Editor | Complete DDL Script** from the context menu.

When you run the **Complete DDL Script** command, DatabaseSpy analyzes all database objects (tables, views, stored procedures, and so on), and then generates their corresponding DDL statements in a new SQL Editor window. If multiple schemas were selected, the corresponding DDL scripts will be in a new SQL Editor window. While DDL statements are being generated, you

are informed of the progress in the Output window as well as through a dialog box from where you can also cancel the operation.

Once DDL statement generation is complete, you can save the script to a file (see Opening, Saving, and Closing SQL Files).

9.3   Opening, Saving, and Closing SQL Files

Any file that is stored in the file system can be opened via the **File | Open** menu (**Ctrl+O**) and will be displayed under its file name in a new SQL Editor window.

```
1        -- target: AnimalTypeCountQuery
2        SELECT COUNT (ZooDB.dbo.tblAnimalTypes.EnglishName)
3        FROM ZooDB.dbo.tblAnimalTypes, ZooDB.dbo.
         tblAnimalCategories
4        WHERE ZooDB.dbo.tblAnimalTypes.Category=ZooDB.dbo.
         tblAnimalCategories.CategoryID AND
5        ZooDB.dbo.tblAnimalCategories.Category='perch-like';
6
7        -- target: AnimalNameQuery
8        SELECT EnglishName, LatinName
9        FROM ZooDB.dbo.tblAnimalTypes
10       INNER JOIN ZooDB.dbo.tblAnimalCategories
11       ON ZooDB.dbo.tblAnimalTypes.Category=ZooDB.dbo.
```

If an SQL file that has already been added to the project, it can be opened from the Project window and will be displayed under its file name in a new SQL Editor window. A file that has been added to the Global Resources can be opened via the **File | Open | Open Global Resource...** menu and will be displayed under its Alias name in a new SQL Editor window.

To open an SQL file:

Do one of the following:

- Select the menu option **File | Open | Open File...** or press **Ctrl+O**. The standard Windows **Open** dialog box appears. Select an SQL (*.sql) file and click **Open**.
- In the Project window, in the SQL folder do one of the following:

  - Select an SQL file and double-click the file name.
  - Right-click the file and select **Open** from the context menu.

The SQL file is displayed in a new SQL Editor window.

To open a global resource:

- Select the menu option **File | Open | Open Global Resource...** to display the **Choose Global Resource** dialog box and do one of the following:

  - Select a file-type global resource under the Files directory and click **OK**. The file is displayed under its Alias name in a new SQL Editor window.
  - Select a folder-type global resource under the Folders folder and click **OK**. The standard Windows **Open** dialog box appears and displays the content of the directory that is defined for the active configuration. Select a file and click **Open**. The file is displayed under its file name in a new SQL Editor window.

Alternatively, do the following:

1. Select the menu option **Tools | Global Resources** or, in the Global Resources toolbar,

    click the **Manage Global Resources** [image] button.
2. Under the Files folder, click the Alias name of the SQL file you want to open.

3. Click the **View** button. The file is opened with the configuration that is selected in the
    drop-down list of the Global Resources toolbar. If the selected configuration is not
    applicable, the SQL file opens with the Default configuration.

Reparsing

It may occur that a script contains parsing errors when it is first displayed in the SQL Editor. If, for
example, the execution blocks are not set correctly, such a script would produce an error when
being executed. (To check the execution blocks, toggle on the **Show groupings for execution**

[image] icon.) Also the syntax highlighting may not be displayed correctly when the script is initially
parsed or edited by the user. To correct such errors, the SQL Editor toolbar provides the **Reparse**

the whole document [image] button.

To correct a SQL script with parsing errors:

- In the SQL Editor toolbar, click the **Reparse the whole document** [image] button.

Saving and closing

DatabaseSpy offers several options for saving SQL statements in, or closing one or several
windows of, the SQL Editor:

- Save: saves the contents of the active window to the file from which it has been opened.
- Save As: pops up the familiar Windows **Save As** dialog box, in which you enter the name
  and location of the file you wish to save the active file as.
- Save All: saves all modifications that have been made to any open documents.
- Close: closes the active window.
- Close All: closes all open windows.
- Close All But Active: closes all open windows but leaves the active file open in the SQL
  Editor.

For all Close commands, if the file was modified (indicated by an asterisk * after the file name in
the title bar), you will be asked if you wish to save the file first.

All commands except **Close All But Active** are available in the **File** menu, and all commands
that affect only one particular window are included in the context menu that opens when you right-
click the tab of the respective SQL Editor window.
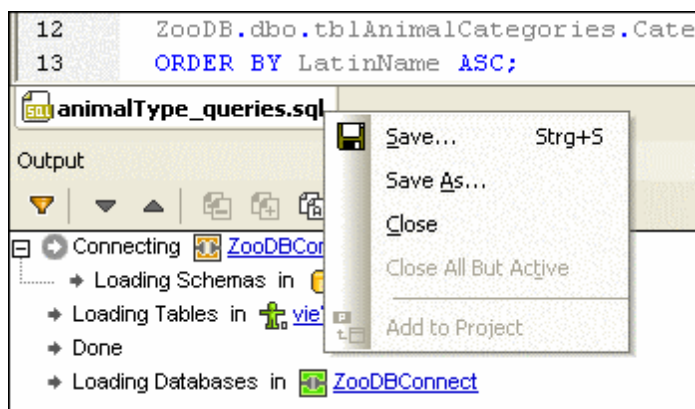
In addition to the menu commands listed above, you can also use the keyboard shortcut **Ctrl+S** to save the active SQL file.

Saving SQL scripts

You can save any SQL that appears in an SQL Editor window and re-use the script later on. If the content of an SQL Editor window has already been saved, the file name appears in the title bar of the SQL Editor window. Unsaved files can be identified by the asterisk to the right of the generic title in the title bar.

A saved SQL script can be made available for all Altova applications by defining it as a global resource. Global resources can be used in all Altova applications that share the same Global Resources XML File.

To save the content of an SQL Editor window to a file:

1. Do one of the following:

    - Right-click the tab of the respective SQL Editor window and choose **Save...** from the context menu.
    - Make the SQL Editor window you want to save the active window and press **Ctrl+S**.

2. If you save the file for the first time, choose a name and path for the file in the **Save As...** dialog box and click **Save**.

To save an SQL file under a new name:

1. Right-click the tab of the respective SQL Editor window and choose **Save As...** from the context menu.
2. In the **Save As...** dialog box, choose a new name and path for the file and click **Save**.

9.4   **Formatting SQL**

The appearance of SQL code can be customized separately for each database kind that is supported by DatabaseSpy. In the SQL Formatting page of the **Options** dialog box, you can furthermore disable this formatting for the SQL Editor and/or the Database Structure Change Script window. Both check boxes are activated by default.

Any changes you make to the default formatting are recorded in a configuration file, at the following path: **..\Documents\Altova\DatabaseSpy2017\SQLFormattingConfigurations.qfmt**. Please note that the configuration file is not automatically transferred during upgrades if the installation folder changes. After upgrading to a new major version, you have to copy the configuration file to the new installation folder (**..\Documents\Altova\DatabaseSpy<Version>**) yourself.

Note: As of version 2009, a new version number is used for configuration files. Therefore, configurations that have been saved in DatabaseSpy 2009 cannot be opened in earlier releases.

The formatting is applied when you generate SQL (see Generating SQL Statements). In addition, you can always click the **Apply SQL Formatting to the active SQL Editor** ▣ button.

You can format the way SQL code is displayed separately for a variety of database kinds:

- MS SQL Server 2005
- MS SQL Server 2008 and greater
- MS Access 2003 and greater
- MySQL 5 and greater
- Oracle 9 and 10
- Oracle 11 and greater
- IBM DB2 8 and greater
- IBM iSeries 5.4 and greater
- IBM Informix 11.1 and greater
- Sybase 12 and greater
- SQLite 3 and greater
- PostgreSQL 8.3 and greater
- Firebird
- Progress OpenEdge
- Any Other Database (ANSI 2003 syntax)

The **SQL Formatting Configurations** dialog box displays a tree structure which allows you to define either the general formatting options on root level or specify formatting options more specifically for the individual statement types.
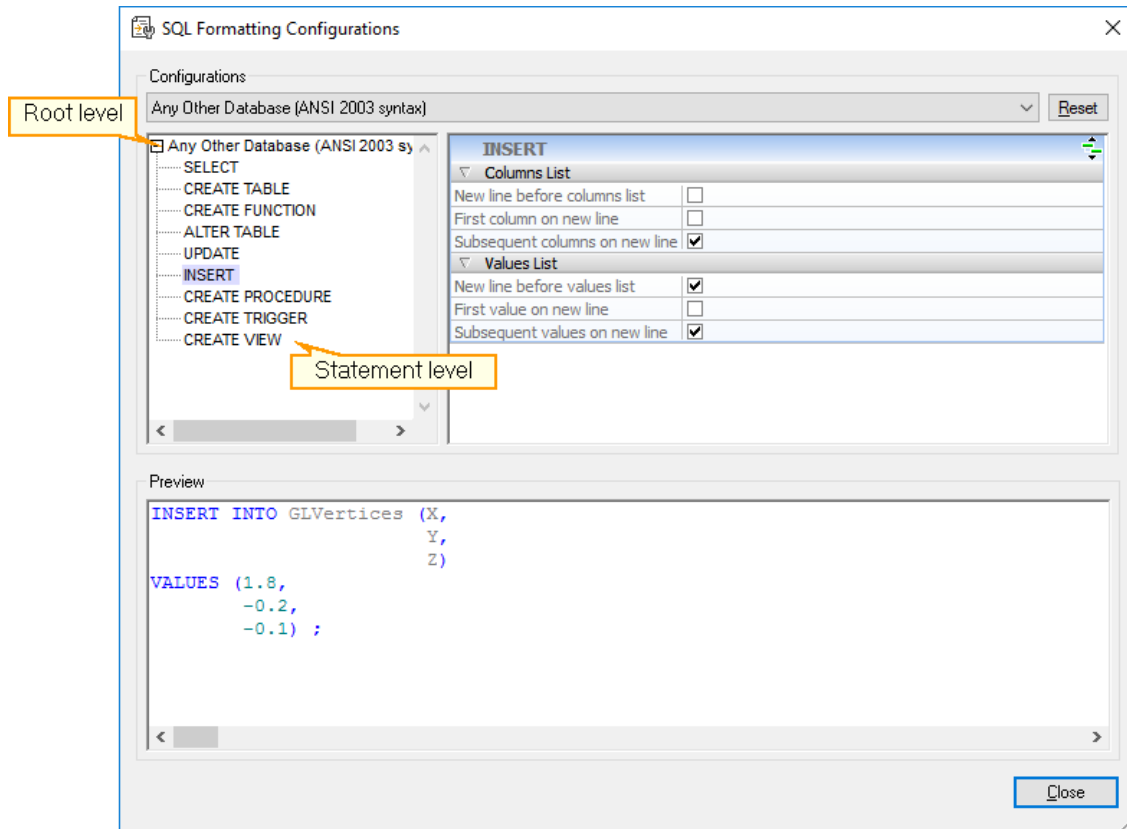
In the Preview group box, the options are immediately applied to a sample statement. Selecting a specific statement type in the tree structure displays the relevant part of the sample statement in the Preview group box.

Options on root level
If you select the root level of a database type, you can either deactivate the formatting options for this individual database type (without deactivating the SQL formatting feature as a whole), or define which case should be used for identifiers, functions, and keywords, when to add spacing and blank lines, and how far text should be indented.

To deactivate SQL formatting for a certain database kind:

1. Select the menu option **Tools | Options** or press **Ctrl+Alt+O** to open the **Options** dialog box and click the **SQL Formatting Configurations...** button on the SQL Formatting page.
2. In the **SQL Formatting Configurations** dialog box, select the respective database kind from the drop-down list in the Configurations group box.
3. In the Options section, select the **Ignore All** check box.

Options for the individual statement types
When you select one of the statement types in the tree structure, the corresponding formatting options are displayed in the window on the right side of the dialog box. Different options are available for the individual statement types.

The header of the options list provides an icon on the right edge that displays the properties

inherited from the root level. If you click the **Show inherited properties** ⬍ icon, the options
from the root level are displayed and you can change the settings for the selected statement type.
Note that these changes apply only to the selected statement type; for all other parts of the
statements, the settings defined on the root level are still valid! Options that differ from those
defined on root level appear in bold type. These options remain also visible if you click the **Hide**

inherited properties ⬍ button.

You can customize the display of the following statement types:

- **Select**

Statement
New line before ORDER BY
Select Items List
First item on new line
Subsequent select list items on new lines
FROM Clause
New line before FROM
First table source on new line
Subsequent table sources on new lines
WHERE Clause
New line before WHERE
First search condition on new line
Subsequent search conditions on new lines

- **Create Table**

Statement
First column/key/index/etc. on new line
Subsequent columns/keys/indices/etc. on new lines
New line before the table definition

- **Create Function**

Statement
Indent contents

- **Alter Table**

Statement
New line before (ADD) CONSTRAINT
New line before FOREIGN KEY
New line before REFERENCES
New line before PRIMARY KEY
New line before UNIQUE

- **Alter Function**

Statement
Place BEGIN and END on new lines
Indent contents

- **Update**

Columns List
New line before columns list
First column on new line
Subsequent columns on new line
WHERE Clause

New line before WHERE
First search condition on new line
Subsequent search conditions on new lines

- **Insert**

Columns List
New line before columns list
First column on new line
Subsequent columns on new line
Values List
New line before values list
First value on new line
Subsequent values on new line

- **Create Procedure**

General Formatting
Disable all formatting
New line before definition

- **Create Trigger**

General Formatting
Disable all formatting
New line before definition

- **Alter Procedure**

General Formatting
Disable all formatting
New line before definition

- **Create View**

General Formatting
Disable all formatting
New line before definition

- **Alter View**

General Formatting
Disable all formatting
New line before definition

Note: Some statement types listed above are not available for configuration when they are not applicable to the selected database kind.

Removing formatting and comments

You can remove all additional whitespace and line feeds that have been added by the **Format SQL** command or because of the default setting for formatting SQL in the SQL Formatting options. The menu option **SQL Refactoring | Remove Comments and Formatting** not only deletes all these characters but also removes any text that is located between block comments or after line comments.

The left pane shows a SELECT statement when SQL formatting is enabled in the SQL Formatting options and the `Whitespace markers` and `End-of-Line markers` options are activated in the Text View Settings. The right pane shows the same statement with the same options after the **Remove Comments and Formatting** command has been selected.

Removing comments

If you want to get rid of all that text in your statement that has been commented out, you can use the **Remove comments** option in the **SQL Refactoring** menu. Any text that is enclosed by block comment characters or appears after a line comment will be deleted from the active SQL Editor window (*see screenshot below*).

9.5   **SQL Editor Features**

The SQL Editor has multiple visual aids to make the display, editing and navigation of text easier. These include:

- Line numbers
- Syntax coloring
- SQL statement selection and navigation
- Bookmarks
- Regions
- Source folding
- End-of-Line and whitespace markers
- Zooming
- Word wrapping
- Text highlighting

Where applicable, you can toggle or customize the features above from the **Text View Settings** dialog box. Settings in the **Text View Settings** dialog box apply to the entire application—not only to the active document.
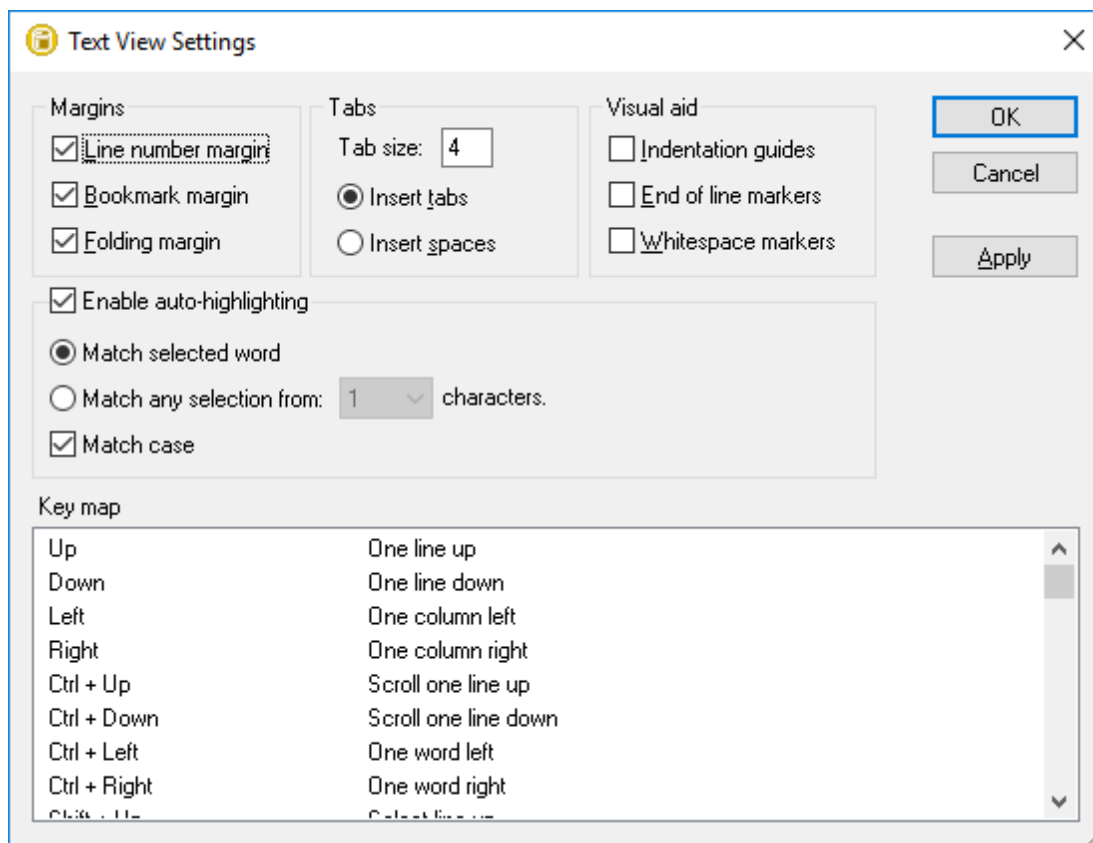
Text View Settings dialog box

To open the **Text View settings** dialog box, do one of the following:

- On the **SQL Editor** menu, select **Text View Settings**.

· Right-click the SQL Editor, and select **Text View Settings** from the context menu.

Some of the navigation or viewing aids can also be toggled from the application menu, toolbars, or keyboard shortcuts. For reference to all applicable shortcuts, see the "Key Map" section of the **Text View Settings** dialog box illustrated above.

Line numbers

Line numbers are displayed in the line numbers margin, which can be toggled on and off in the **Text View Settings** dialog box. When a section of text is collapsed, the line numbers of the collapsed text are also hidden.

Syntax coloring

Syntax coloring is applied according to the semantic value of the text. You can enable or disable syntax coloring from the **Tools | Options** menu, **SQL Editor** group.

Statement selection and navigation

The following buttons in the SQL Editor toolbar allow you to select text, depending on where the cursor is positioned.

Next statement (Alt+Down): Selects the statement after the statement where the cursor is currently located.

Previous statement (Alt+Up): Selects the statement before the statement where the cursor is currently located.

Last statement (Alt+End): Selects the last statement in the SQL Editor.

First statement (Alt+Home): Selects the first statement in the SQL Editor.

Current statement (Shift+Alt+Enter): Selects the statement where the cursor is currently located. Alternatively, you can also triple-click into a statement or click into any of the margins, if visible.

Bookmarks

Lines in the document can be bookmarked for quick reference and access. If the bookmarks margin is toggled on, bookmarks are displayed in the bookmarks margin.

Otherwise, bookmarked lines are highlighted in cyan.

```
1   SELECT [id], [name], [descr], [deptid], [teacherid],
    [upperlimit] FROM [Course];
2   SELECT [id], [name] FROM [Department];
3   SELECT [id], [name] FROM [Teachers];
```

The bookmarks margin can be toggled on or off in the **Text View Settings** dialog box.

You can edit and navigate bookmarks using the following commands:

Insert/Remove Bookmark

Go to Next Bookmark

Go to Previous Bookmark

Delete All Bookmarks

The commands above are available in the **SQL Editor | Bookmarks** menu and the SQL Editor toolbar. Bookmark commands are also available through the context menu, when you right-click the SQL Editor.

Regions
In the SQL Editor, regions are sections of text that you mark and declare as a unit to structure your SQL scripts. The benefit of regions is that their enclosed text can be easily collapsed or expanded, and thus simplify navigation in long scripts (see also "Source folding" below). It is also possible to nest regions within other regions.

To create a region:

1. In the SQL Editor, select the statements you want to make into a region.
2. Click the **Add Region** R button in the SQL Script toolbar or select the menu option **SQL Editor | Insert Region**.

```
1   -- region Department
2   INSERT INTO [Department] ( [id], [name] ) VALUES ( 0, 'Science' );
3   INSERT INTO [Department] ( [id], [name] ) VALUES ( 1, 'Math' );
4   INSERT INTO [Department] ( [id], [name] ) VALUES ( 2, 'Art' );
5   -- endregion
```
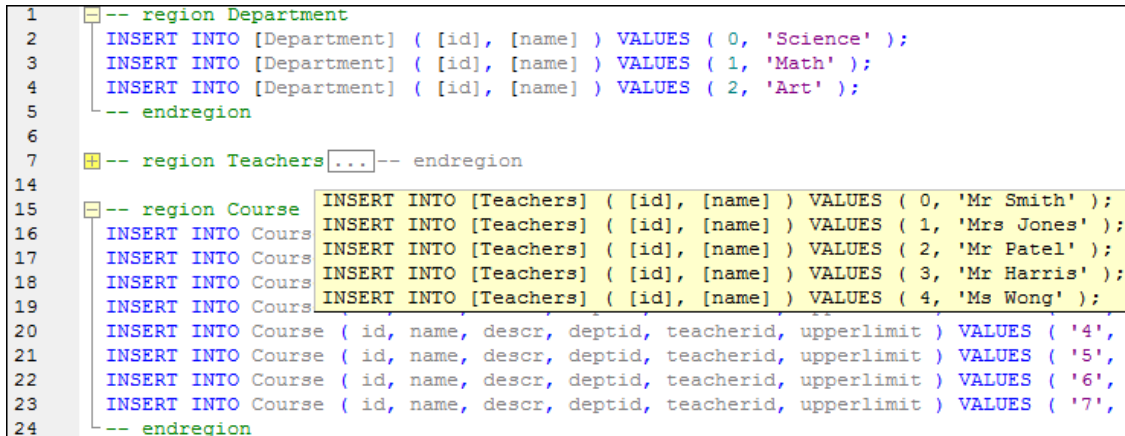
To remove a region:

*   Delete the -- region and -- endregion comments.

Source folding
Source folding refers to the ability to expand and collapse regions and is displayed in the source folding margin. The margin can be toggled on and off in the Text View Settings dialog box. To expand or collapse portions of text, click the "+" and "-" nodes at the left side of the window. Any portions of collapsed code are displayed with an ellipsis symbol. To preview the collapsed code

without expanding it, move the mouse cursor over the ellipsis. This opens a tooltip that displays the code being previewed, as shown in the image below. Note that, if the previewed text is too big to fit in the tooltip, an additional ellipsis appears at the end of the tooltip.

The **Toggle All Folds** command toggles all regions to their expanded or collapsed forms.

End-of-line markers, whitespace markers

End-of-line (EOL) markers and whitespace markers can be toggled on in the **Text View Settings** dialog box. The image below shows a document where both end-of-line and whitespace markers are visible. An arrow represents a tab character, a "CR" is a carriage return, and a dot represents a space character.

Zooming in and out

You can zoom in and out by scrolling (with the scroll-wheel of the mouse) while holding the **Ctrl** key pressed. Alternatively, press the "-" or "+" keys while holding the **Ctrl** key pressed.

Word wrapping

To toggle word wrapping in the currently active document, select the **SQL Editor | Word Wrap** menu command, or click the **Word Wrap** toolbar button.

Text highlighting

When you select text, all matches in the document of the text selection that you make are highlighted automatically. The selection is highlighted in pale blue, and matches are highlighted in pale orange. The selection and its matches are indicated in the scroll bar by gray marker-squares. The current cursor position is given by the blue cursor-marker in the scroll bar.

To switch text highlighting on, select **Enable auto-highlighting** in the Text View Settings dialog

box. A selection can be defined to be an entire word or a fixed number of characters. You can also specify whether casing should be taken into account or not.

For a character selection, you can specify the minimum number of characters that must match, starting from the first character in the selection. For example, you can choose to match two or more characters. In this case, one-character selections will not be matched, but a selection consisting of two or more characters will be matched. So, in this case, if you select `t`, then no matches will be shown; selecting `ty` will show all `ty` matches; selecting `typ` will show all `typ` matches; and so on.

9.6   Executing SQL

SQL statements that have been created in the SQL editor or SQL files that have been opened in the SQL Editor can be executed directly from the SQL Editor. The Execute command can be called via the **SQL Editor** menu, via a keyboard shortcut or by clicking a toolbar icon.

To execute SQL in an SQL Editor window:

1. Make the SQL Editor window in which you want to execute SQL the active window.
2. Optionally, select the statement or statements you want to execute.
3. Click the **Execute** ▶ button, or select the menu option **SQL Editor | Execute**, or press **F5**.
   If the data source is not connected, a popup message is displayed asking whether you would like to connect to the data source (provided that the SQL file is not set to Offline).
4. If applicable, click **Yes** in the message box to connect to the data source.
   All SQL statements that are in the SQL Editor, or the selected statements, respectively, are executed. The results appear in separate result windows, one for each statement.
5. Click the respective tabs to access the individual results.

Alternatively, you can also execute SQL scripts in the Project window if the SQL files are included in the project.

To execute an SQL file:

1. In the Project window, select an SQL file that you want to execute.
2. Right-click the file and choose **Execute SQL** from the context menu.
   If the data source is not connected, a popup message is displayed asking whether you would like to connect to the data source.
3. If applicable, click **Connect** in the message box to connect to the data source.
   The SQL script opens in a new SQL Editor window and is executed immediately.

Execute for Data Editing

When supported in the active data source connection, you can also execute a SELECT statement for data editing in the Result window. In this case, the **Execute for Data Editing** button will be enabled in the toolbar of the SQL Editor and the **Execute for Data Editing** command will be available in the **SQL Editor** menu.

To execute SELECT statements for data editing:

1. Generate a SELECT statement in an SQL Editor window that is connected to a data source that supports data editing from within the SQL Editor.

2. Click the **Execute for Data Editing** button or select the menu option **SQL Editor | Execute for Data Editing**.
   The Result window is switched into the Editing mode and you can edit the database data directly in the result grid.

Asynchronous execution and retrieval

DatabaseSpy uses asynchronous execution and retrieval when a query is started in the SQL Editor or from within a Design Editor or Data Comparison window. While the execution is in

progress, DatabaseSpy displays a message in the Message window.
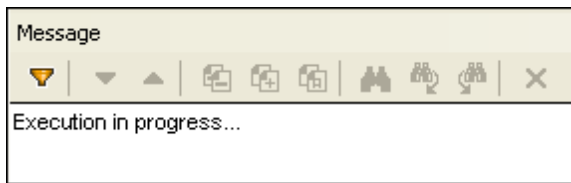
In large databases, you can cancel the execution by clicking the **Stop Execution/Retrieval** ■ button in the SQL Editor toolbar while the execution is in progress and retrieval has not yet been started. This way, you can abort an execution when you notice that the retrieval would take too much time, and refine the query.

After you have canceled an execution or retrieval, an "Execution was cancelled" message is displayed in the status bar of the Result window, and you can edit your query as required and restart the execution.

Defining the execution mode

You can choose whether you want to send the SQL that is contained in the active SQL Editor window to the database engine as a whole, or as separate batches. The Properties window provides the options **SQL Editor should require semicolons** and **Group statements for execution with** for this purpose.

The **Group statements for execution with** list provides the following options:

- **Semicolons**: The parser uses semicolons as statement separator. Any SQL code that is terminated by a semicolon is considered a statement and sent to the server separately. If semicolons are missing from your SQL script, a dialog box may prompt you to add them automatically.
- **No Grouping**: The script is sent to the server as a whole, without any modification. If the script contains multiple statements, and if the driver is known not to support executing multiple statements in bulk, this option is not meaningful. In this case, a dialog box may prompt you about this limitation.
- **SQL Grammar**: The respective flavor of SQL grammar is used to separate the statements when sending them to the server.
- **GO keyword**: A GO keyword must be present in the SQL script to separate execution blocks.

Showing execution groups

The toolbar in the SQL Editor provides the **Show groupings for execution** 📧 button which allows for a graphical illustration of the statement blocks that will be sent to the database engine for execution.

```
 1        -- target: AnimalTypeCountQuery
 2        SELECT COUNT (ZooDB.dbo.tblAnimalTypes.EnglishName)
 3        FROM ZooDB.dbo.tblAnimalTypes, ZooDB.dbo.
          tblAnimalCategories
 4        WHERE ZooDB.dbo.tblAnimalTypes.Category=ZooDB.dbo.
          tblAnimalCategories.CategoryID AND
 5        ZooDB.dbo.tblAnimalCategories.Category='perch-like';
 6
 7        -- target: AnimalNameQuery
 8        SELECT EnglishName, LatinName
 9        FROM ZooDB.dbo.tblAnimalTypes
10        INNER JOIN ZooDB.dbo.tblAnimalCategories
11        ON ZooDB.dbo.tblAnimalTypes.Category=ZooDB.dbo.
          tblAnimalCategories.CategoryID AND
12        ZooDB.dbo.tblAnimalCategories.Category='perch-like'
13        ORDER BY LatinName ASC;
```

In the screenshot above, the groupings for execution are displayed when the "Semicolons" option is selected in the **Group statements for execution with** drop-down list.

Execution timeout

You can define a timeout for the execution of queries in the Retrieval settings group box of the SQL Editor options. DatabaseSpy will attempt to stop retrieving data after the specified amount of time has elapsed. Note that the database kind and database driver must support setting a command timeout; otherwise, the query will continue running beyond the timeout period. In this case, you can stop execution using the **Stop Execution/Retrieval** 🔴 button; however, be aware that some drivers might not support cancelling the execution.

Abort execution on error

When the SQL Editor window is open, the Properties window includes a property called **Abort execution on error**. This option may be useful when you execute SQL scripts that contain multiple statements, and, if any statement fails, you would like the execution to stop at the corresponding statement. To enable this behavior, select the **Abort execution on error** check box.
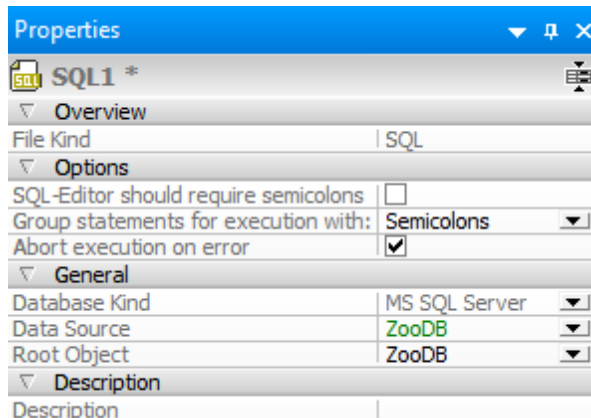
Properties window

Consider the following example: you would like to run against the database a script which contains multiple INSERT statements. If the **Abort execution on error** check box is enabled, and an INSERT statement fails, the script will stop running at the INSERT statement which failed. In this case, any subsequent INSERT statements will not be executed. If the **Abort execution on error** check box is disabled, and an INSERT statement fails, the script will proceed to executing the rest of the INSERT statements, until the end of the script.

The execution status of the script (including the number of affected rows) can be viewed in the Message window, along with additional information about the error which occurred (if any). Click the underlined text in the Messages window to jump to the corresponding line in the SQL Editor.

9.7   Auto-Completion

When entering an SQL statement in the SQL Editor, auto-completion helps you by offering lists of appropriate keywords, data types, identifiers, separators, and operators depending on the type of statement you are entering. By default, auto-completion is invoked automatically in certain contexts (for example, when you start typing a statement).
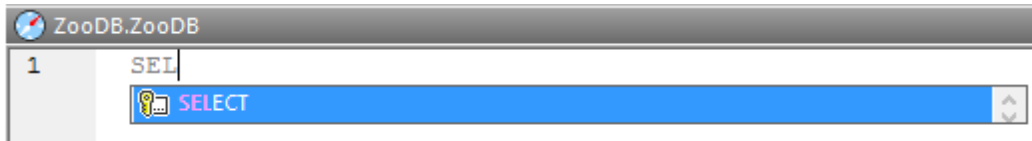
Auto-completion

Use the **Up** and **Down** keyboard keys to navigate through the list of suggestions. To pick a suggested entry, click it or press **Enter**.

To disable auto-completion suggestions:

1. On the **Tools** menu, click **Options** (or press **Ctrl+Alt+O**).
2. Click **SQL Editor**.
3. Under **Entry Helpers**, clear the **Automatically open** check box.

To invoke auto-completion suggestions manually as and when required:

- In the SQL Editor, press **Ctrl+Space**.

For example, in the image below, auto-completion has been invoked manually by pressing **Ctrl +Space**. The goal here is to quickly select a table available in the [dbo] schema of a SQL Server database.
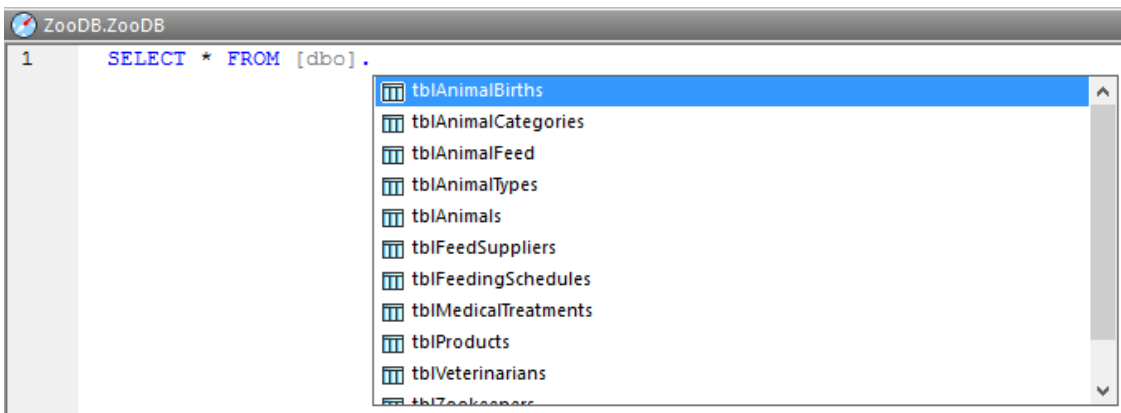
Auto-completion invoked manually

The object types that may appear in the auto-completion window can be identified by icon type, as follows:

Schemas

Tables

Views

Columns

Functions

Data types

Procedures

Keywords

Separators

9.8 Commenting Out Text

In the SQL Editor it is possible to comment out statements, parts of statements, or groups of statements. These statements, or the respective parts of them, will be skipped when the SQL script is being executed.

To comment out a section of text:

1. Select a statement or part of a statement.

```
-- target: AnimalNameQuery
SELECT EnglishName, LatinName
FROM ZooDB.qa_usr.tblAnimalTypes
INNER JOIN ZooDB.dbo.tblAnimalCategories
ON ZooDB.qa_usr.tblAnimalTypes.Category=ZooDB.dbo.tblAnimal
AND
ZooDB.dbo.tblAnimalCategories.Category='perch-like'
ORDER BY LatinName ASC
```

2. Click the **Comment/Uncomment selected block** /* button in the SQL Script toolbar or select **Insert | Insert/Remove Block Comment** from the **SQL Editor** menu. The statement is commented out.

```
-- target: AnimalNameQuery
SELECT EnglishName, LatinName
FROM ZooDB.qa_usr.tblAnimalTypes
INNER JOIN ZooDB.dbo.tblAnimalCategories
ON ZooDB.qa_usr.tblAnimalTypes.Category=ZooDB.dbo.tblAnimal
/*AND
ZooDB.dbo.tblAnimalCategories.Category='perch-like'*/
ORDER BY LatinName ASC
```

To comment out text line by line:

1. Put the cursor in a statement at the point from which you want to comment it out.

```
-- target: AnimalNameQuery
SELECT EnglishName, LatinName
FROM ZooDB.qa_usr.tblAnimalTypes
INNER JOIN ZooDB.dbo.tblAnimalCategories
ON ZooDB.qa_usr.tblAnimalTypes.Category=ZooDB.dbo.tblAnimal
 AND
ZooDB.dbo.tblAnimalCategories.Category='perch-like'
ORDER BY LatinName ASC
```

2. Click the **Comment/Uncomment selected lines** -- button in the SQL Script toolbar or select **Insert | Insert/Remove Line Comment** from the **SQL Editor** menu. The statement is commented out from the current position of the cursor to the end of the line.

```
-- target: AnimalNameQuery
SELECT EnglishName, LatinName
FROM ZooDB.qa_usr.tblAnimalTypes
INNER JOIN ZooDB.dbo.tblAnimalCategories
ON ZooDB.qa_usr.tblAnimalTypes.Category=ZooDB.dbo.tblAnimalC
 AND
ZooDB.dbo.tblAnimalCategories.Category='perch-like'
--ORDER BY LatinName ASC
```

To remove a block comment or a line comment:

1. Select the part of the statement that is commented out. If you want to remove a line comment, it is sufficient to select only the comment marks ‑‑ before the comment.
2. Do one of the following:

   - Click the **Comment/Uncomment selected block** button in the SQL Script toolbar or select **Insert | Insert/Remove Block Comment** from the **SQL Editor** menu.
   - Click the **Comment/Uncomment selected lines** button in the SQL Script toolbar or select **Insert | Insert/Remove Line Comment** from the **SQL Editor** menu.

9.9   **Naming Result Tabs**

When you execute a script that contains more than one SELECT query, the results of each query are displayed in separate result windows. By default, these windows are called "Result1", "Result2", etc. Using *targets*, you can specify more meaningful names for query result windows.

Note: The target keywords are only recognized if either "Semicolons" or "SQL Grammar" is selected in the **Group statements for execution with** drop-down list in the Properties window. If you execute a query containing target keywords without having selected the appropriate options, the query is still executed, but output to unnamed result tabs. A warning is displayed in the **Message** tab of the Result window.

To name a target result window:

1. Place the cursor anywhere in the SELECT statement. Click the **Current statement** ![icon] button (**Shift+Alt+Enter**) in the SQL Editor window. This selects the entire statement.

2. Click the **Define Target Name** ![icon] button in the SQL Script toolbar. The text --target: Result appears above the statement.

```
-- target: Result
SELECT [ZookID], [FirstName], [LastName], [Address],
[City], [State], [Telephone], [DOB] FROM
[ZooDB].[dbo].[tblZookeepers];
```

3. Change the text "Result" as desired.

```
-- target: zookepers
SELECT [ZookID], [FirstName], [LastName], [Address],
[City], [State], [Telephone], [DOB] FROM
[ZooDB].[dbo].[tblZookeepers];
```

4. In the Properties window, make sure that either "Semicolons" or "SQL Grammar" is selected in the **Group statements for execution with** drop-down list.

5. Click the **Execute** ![icon] button or press **F5**.
    The results appear in a Result window that has a tab with the text that you specified as target, e.g., zookeepers.

9.10   Finding and Replacing Text

To start a search in the SQL Editor, press **Ctrl+F** (or select the menu command **Edit | Find**). You can then search in the entire document or within a text selection for a search term that you enter in the dialog.

- Enter a string to find, or use the combo box to select a string from one of the last 10 strings.
- When you enter or select a string to find, all matches are highlighted and the positions of the matches are indicated by beige markers in the scroll bar.
- The currently selected match has a different highlight color than the other matches, and its position is indicated in the scroll bar by the dark blue cursor-marker.
- The total number of matches is listed below the search term field, together with the index position of the currently selected match. For example, `2 of 4` indicates that the second of four matches is currently selected.
- You can move from one match to the next, in both directions, by selecting the **Previous** ◀ **(Shift+F3)** and **Next** ▶ **(F3)** buttons at bottom right.
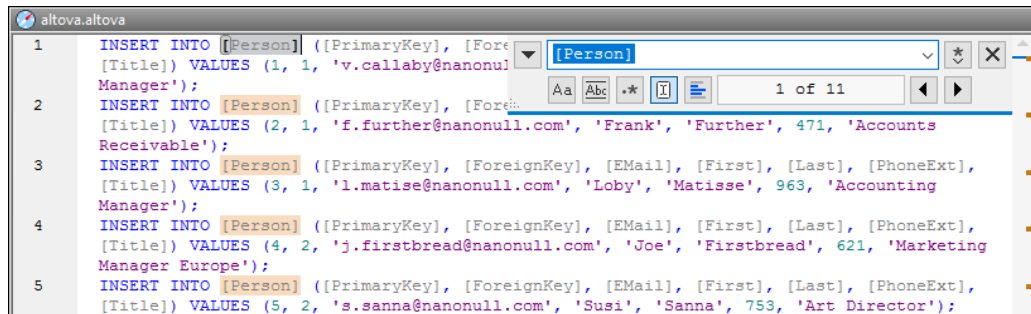
- To switch between the Find and Replace modes, click the down-arrow button ▼ at top left.
- To close the Find dialog, click the **Close** ✕ button at top right, or press **Esc**.

Note the following points:

- The Find dialog is *modeless*. This means that it can remain open while you continue to use Text View.
- If text is selected prior to opening the dialog box, then the selected text is automatically inserted into the search term field.
- To search within a selection, do the following: (i) Mark the selection; (ii) Toggle on the *Find in Selection* ▤ option to lock the selection; (iii) Enter the search term. To search within another selection, unlock the current selection by toggling off the *Find in Selection* ▤ option, then make the new selection and toggle on the *Find in Selection* ▤ option.
- After the Find dialog is closed, you can repeat the current search by pressing **F3** for a forward search, or **Shift+F3** for a backward search. The Find dialog will appear again in this case.

Find options

Find criteria can be specified via buttons located below the search term field. When an option is toggled on, its button color changes to blue. You can select from the following options:

Option	Icon	Description
Match case	Aa	Performs a case-sensitive search when toggled on ("Address" is not the same as "address").
Match whole word	Abc	Only the exact words in the text will be matched. For example, for the input string *fit*, with **Match whole word** toggled on, only the word *fit* will match the search string; the *fit* in *fitness*, for example, will not.
Regular expression	.*	If toggled on, the search term will be read as a regular expression. See "Using regular expressions" below.
Find anchor	I	When a search term is entered, the matches in the document are highlighted and one of these matches will be marked as the current selection. The **Find anchor** toggle determines whether that first current selection is made relative to the cursor position or not. If **Find anchor** is toggled on, then the first currently selected match will be the next match from the current cursor location. If **Find anchor** is toggled off, then the first currently selected match will be the first match in the document, starting from the top.
Find in selection	☰	When toggled on, locks the current text selection and restricts the search to the selection. Otherwise, the entire document is searched. Before selecting a new range of text, unlock the current selection by toggling off the Find in Selection option.

Using regular expressions

You can use regular expressions (regex) to find a text string. To do this, first, switch the *Regular expression* .* option on. This specifies that the text in the search term field is to be evaluated as a regular expression. Next, enter the regular expression in the search term field. For help with building a regular expression, click the **Regular Expression Builder** ☆ button, which is located to the right of the search term field. Click an item in the Builder to enter the corresponding regex metacharacter/s in the search term field. The screenshot below shows a simple regular expression to find email addresses.

The following custom set of regular expression metacharacters are supported when finding and replacing text.

.	Matches any character. This is a placeholder for a single character.
\(abc\)	The \(and \) metacharacters mark the start and end of a tagged expression. Tagged expressions may be useful when you need to tag ("remember") a matched region for the purpose of referring to it later (back-reference). Tagged expressions are similar to matched subexpressions (indexed groups) in the .NET flavour of regular expressions. Up to nine sub-expressions can be tagged (and then back-referenced later). For example, **\(the\) \1** matches the string the the. This expression can be literally explained as follows: match the string "the" (and remember it as a tagged region), followed by a space character, followed by a back-reference to the tagged region matched previously.
\n	Where n is 1 through 9 , n refers to the first through ninth tagged region (see above). For example, if the search string was **Fred\([1-9]\)XXX** and the replace string was **Sam\1YYY**, when applied to **Fred2XXX** this would generate **Sam2YYY**.
\<	Matches the start of a word.
\>	Matches the end of a word.
\	Escapes the character following the backslash. In other words, the expression \x allows you to use the character x literally. For example, \[would be interpreted as [and not as the start of a character set.
[...]	Matches any characters in this set. For example, **[abc]** matches any of the characters a, b or c. You can also use ranges: for example **[a-z]** for any lower case character.
[^...]	Matches any characters not in this set. For example, **[^A-Za-z]** matches any character except an alphabetic character.
^	Matches the start of a line (unless used inside a set, *see above*).
$	Matches the end of a line. For example, **A+$** matches one or more A's at end of line.

*	Matches zero or more occurrences of the preceding expression. For example, `Sa*m` matches `Sm`, `Sam`, `Saam`, `Saaam` and so on.
+	Matches one or more occurrences of the preceding expression. For example, `Sa+m` matches `Sam`, `Saam`, `Saaam` and so on.

Replacing text

To replace text, do the following:

1. Press **Ctrl+H** (or select the menu command **Edit | Replace**) to open the Replace dialog. (Alternatively, you can switch to Replace mode of the Find/Replace dialog by clicking the down-arrow button ▼ at the top left of the dialog.)
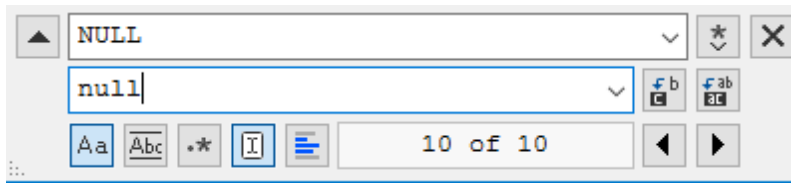
2. Enter the string to be replaced in the Find field, and enter the new string in the Replace field. The number of text matches to replace and the index of the currently selected match is displayed below the Replace field. Also, the locations of matches are indicated in the scroll bar by beige markers.
3. The **Replace Next** and **Replace All** buttons are located to the right of the Replace field. If you click **Replace Next**, one of the following happens: (i) If the cursor is located adjacent to a match or inside a match, then the match is replaced; (ii) if the cursor is located outside a match, it jumps to the next match. Click **Replace All** to replace all matches.

Note the following points:

- To replace text within a selection—rather than the entire document—do the following: (i) Mark the selection; (ii) Toggle on the *Find in Selection* ☰ option to lock the selection; (iii) Enter the Find and Replace text strings; (iii) Click **Replace Next** or **Replace All** as required. To replace text within another selection, unlock the current selection by toggling off the *Find in Selection* ☰ option, then make the new selection and toggle on the *Find in Selection* ☰ option to lock the new selection.
- To undo a replace action, press **Ctrl+Z** or select **Edit | Undo**.

Finding and replacing special characters

You can search for any the following special characters within text, provided that the **Regular expression** option ⃰ is enabled:

- \t (Tab)
- \r (Carriage Return)
- \n (New line)

- \\ (Backslash)

For example, to find a tab character, press **Ctrl + F**, select the `.*` option, and then enter `\t` in the Find dialog box.

The special characters above can also be used as replacement text. For example, to replace all occurrences of semi-colon by carriage returns within the text "A;B;C;D", use the following find and replace options:
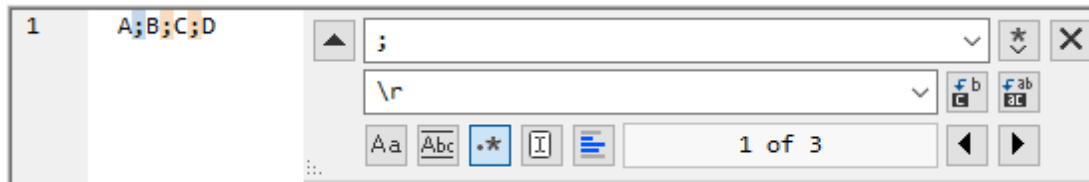
The outcome is:

```
A
B
C
D
```

9.11   Selecting Data for Export

In addition to DatabaseSpy's powerful export function, you can use SELECT statements to retrieve data from a table and export only this particular data to a range of file formats such as

XML or HTML. The SQL Editor window provides an additional **Export** [icon] button to open the **Export database data** dialog box to facilitate this. When you click this button, the dialog box opens with the **SQL** radio button activated and the content of the SQL Editor window inserted into the Source group box.
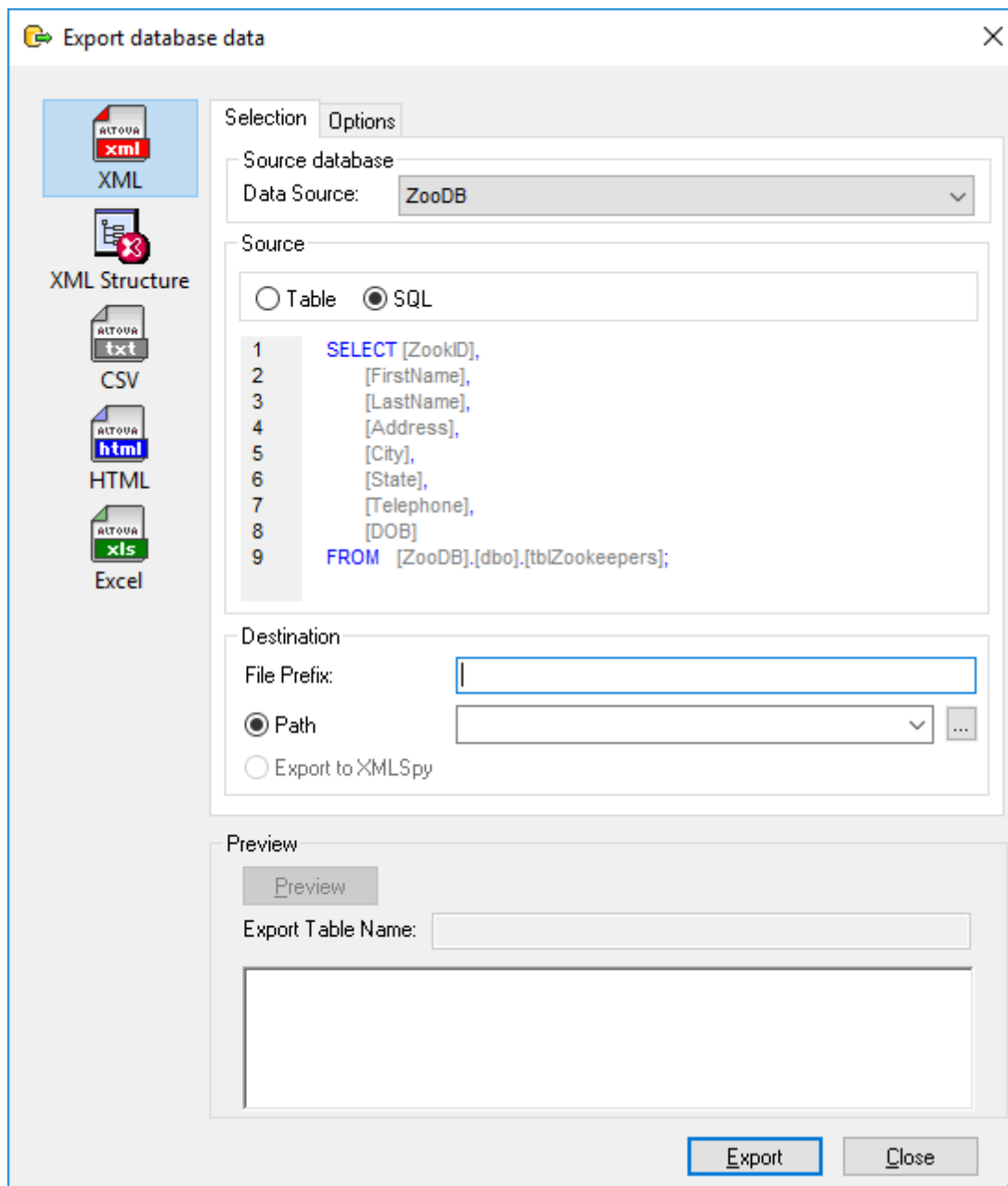
Export database data dialog box

Note: Clicking the **Export** ⬒ button in the Tools toolbar will also open the Export database data dialog box. However, in this case the Table radio button will be preselected and you would have to type in or paste the SQL code manually.

To export data using SQL statements in an SQL Editor window:
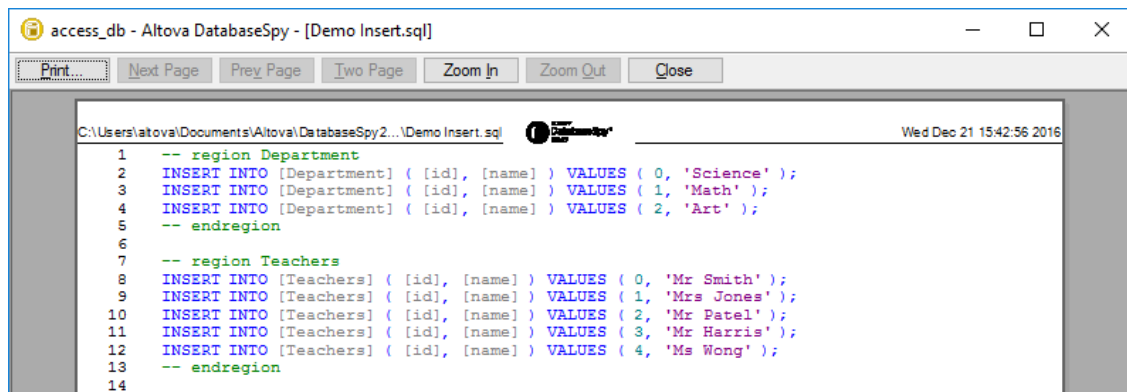
1. Make the SQL Editor window whose SQL statements you want to use for export the active window.
2. Press the **Export** ⬒ button in the SQL Editor window.
    The **Export database data** dialog box opens. The SQL statements that are currently in the SQL Editor appear in the Source group box.
3. Optionally, enter a file prefix that will be added to the file name.
4. Do one of the following:

    - Activate the **Path** radio button and enter the Path you want the destination files to appear in.
    - Activate the **Export to XMLSpy** radio button if you want your exported data to be displayed in XMLSpy. Note that this option is not available for export to Excel files, or if XMLSpy is not installed on your computer

5. Optionally, click the **Options** tab to change the options for XML, XML Structure, CSV, HTML, or Excel.
6. Click the **Export** button when all the settings are correct.
7. Click **OK**. The data is exported as Table1 in the desired format and either stored under the path you specified or opened in XMLSpy for further editing.

9.12 Printing SQL

You can print the content of the SQL Editor like in any other text editor. When turning on the line numbers margin, you can also print out line numbers.

Note: The source folding margin has no effect on the print-out (that is, the expand/collapse icons will not show up in the print-out). If any bookmarks are defined in the bookmark margin, the line where the bookmark occurs will appear highlighted on the printed page.

A print preview is available for you to check the layout of the printed page before sending it to the printer.

To print the content of an SQL Editor window:

1. Click in the SQL Editor window to make it active.
2. Select the menu option **File | Print...**, or press **Ctrl+P**, or click the **Print** icon in the Standard toolbar.
3. In the **Print range** group box, define whether you want to print all or specific pages or only the selected part of the SQL.
4. Optionally, define the printer setup by clicking the **Properties** button.
5. Click **OK**.

To preview SQL before printing:

1. Click in the SQL Editor window to make it active.
2. Select the menu option **File | Print Preview**.
3. In the print preview, do one of the following:
    o Click the **Print** button to print the file.
    o Click **Close** to return to the SQL Editor.

Chapter 10

Comparing Database Data

10   Comparing Database Data

Altova web site: 🌐 compare database contents

To compare database data in DatabaseSpy, the tables that should be compared are added to components in a Data Comparison window, where they are mapped, and where several options for comparing and merging the table data are provided. The two components represent the databases that are compared and are indicated with different colors (grey: left component, dark yellow: right component). These colors will later also appear in the Comparison Result window so that you can easily see which column belongs to which database. The title bar of each component displays the name of the database; if you place the mouse cursor over a title bar, a balloon help appears and displays also the name of the data source that is used to connect to the database.
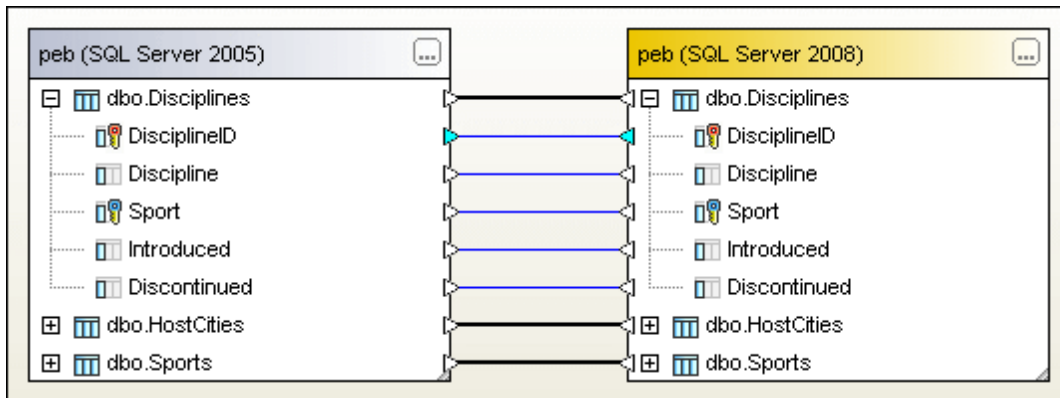
When the requirements for a database data comparison are met, there are several ways to start a comparison of databases in DatabaseSpy:

- Opening a Data Comparison window and adding tables using the **Select Database Objects for Comparison** dialog box. Tables from both databases can be selected in the dialog box.

- Choosing the command **Comparison | Show in new Data Comparison Document** from the context menu that opens when you right-click tables in the Online Browser. The tables from the second database can be added via the **Select Database Objects for Comparison** dialog box or by dragging them from the Online Browser into the right component in the Data Comparison window.

- Opening a data comparison that is included in your DatabaseSpy Project from the Project window.

- Opening a data comparison file that is stored in your file system using the **File | Open | Open File...** command.

You can switch on the display of the Message window by activating the **Toggle Message Window** 🖼 button in the toolbar of the Data Comparison window. The Message window provides a summary of the compared tables and you can click the hyperlinks to jump to the respective table in the Data Comparison window and show its properties in the Properties window (*see screenshot below*).

After a comparison has been started, the Message window shows an overview of the comparison results and provides links to the Comparison Result windows of the individual compared tables (*see screenshot below*).
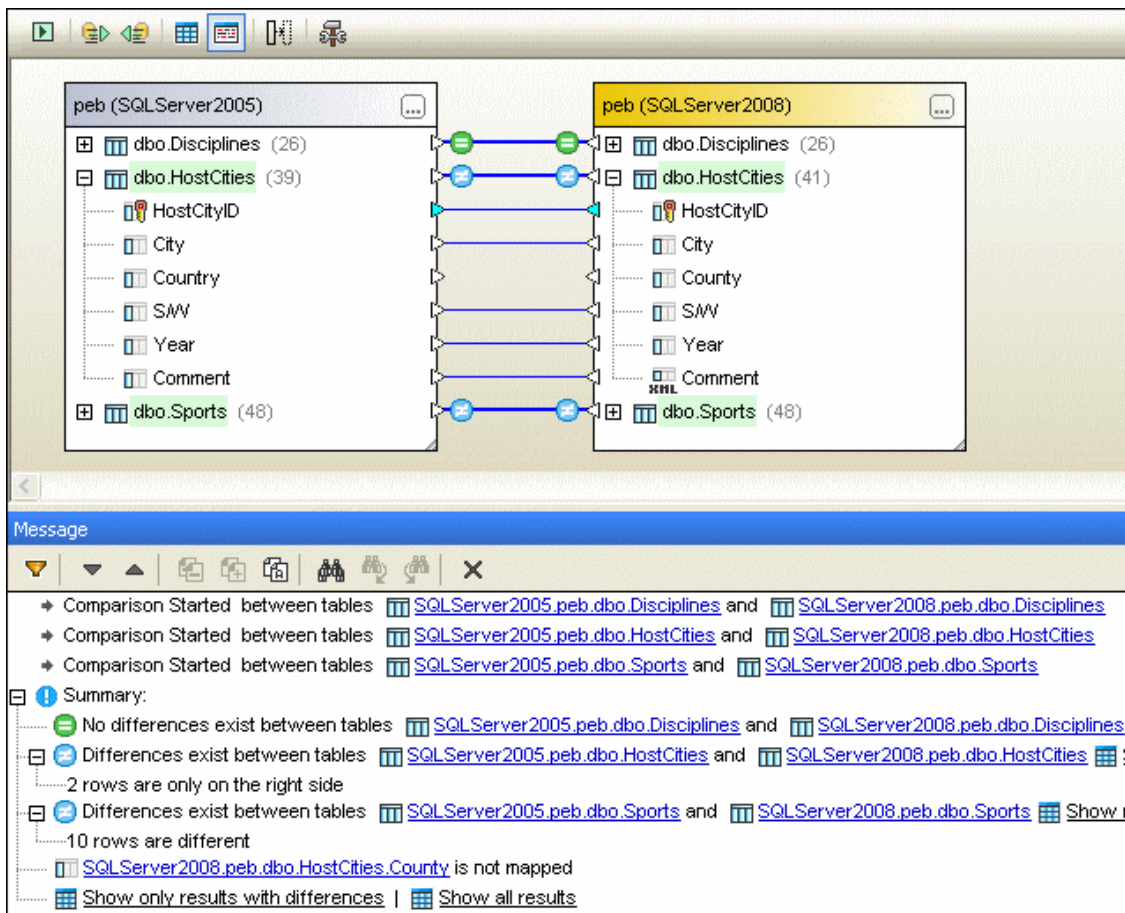
10.1 Prerequisites

There are two prerequisites that are mandatory in order to start a table comparison in DatabaseSpy:

- An active connection to a data source
- A sorting key in every table that is to be compared

Data source connection

At least one connection to a data source must be available in your DatabaseSpy Project. If this is not the case, an information box is displayed which allows you to add a data source.
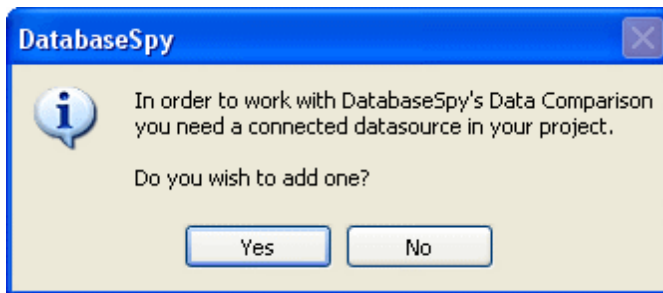
Clicking **Yes** in this dialog box takes you directly to the **Create a Database Connection** dialog box where you can define a data source connection. The new data source is then added to the project, DatabaseSpy connects to it, and the **Select Database Objects for Comparison** dialog box opens.

If none of your project's data sources are connected when you select the **File | New | Data Comparison** menu option, you also cannot start a data comparison.
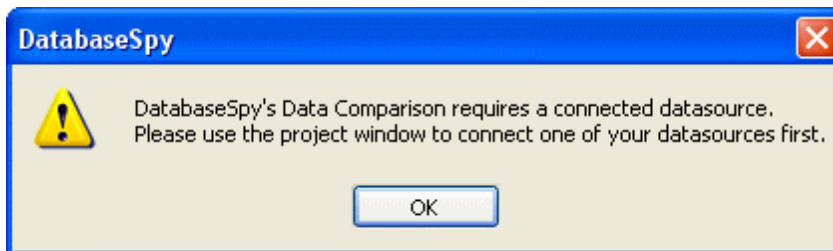
Close the message box by clicking **OK** and connect to one of the data sources that are available in your project.

Key column

At least one column in each table is used as a sort criterion that is needed for comparison. DatabaseSpy uses primary key columns to sort tables for comparison. DatabaseSpy indicates tables that cannot be compared (*see screenshot below*), and these tables cannot be mapped.
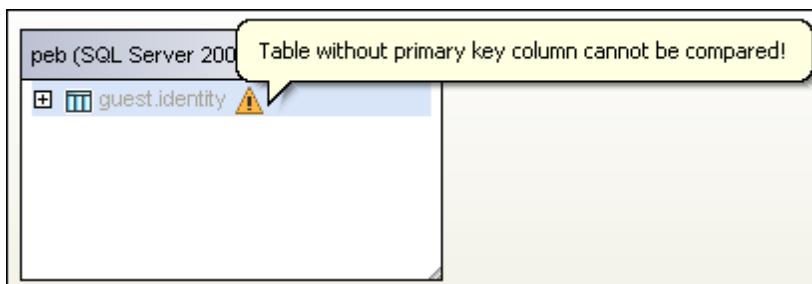
10.2   Opening a Database Data Comparison Window

When opening a new Data Comparison window, the database comparison is given a name of the form `DatabaseDataComparisonX`, where `X` is an integer indicating that database comparison's position in the sequence of database comparisons opened in the currentDatabaseSpy session. This name appears in a tab at the bottom of the window.
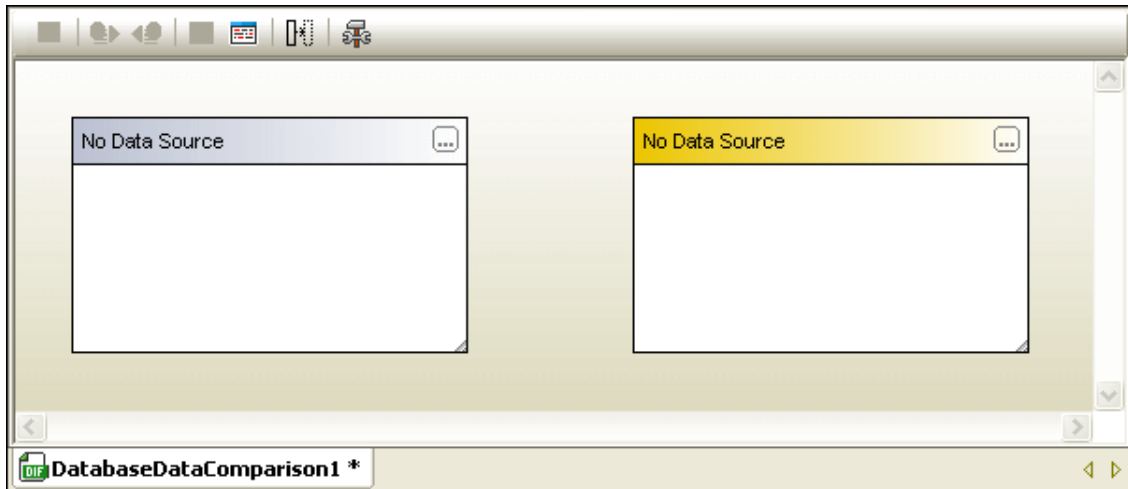
The **File | New | Data Comparison** command opens the Data Comparison window and pops up the **Select Database Objects for Comparison** dialog box, where you must connect to the databases you want to compare (one in each component) and select the required tables. Usually you will also select the first database when opening a new Data Comparison window. However, you could also just open a Data Comparison window and add the tables later. This way, you could, for example, create comparison templates with different options (e.g., ignore whitespace or case, etc.) and save them in a project.

To open a Data Comparison window:

- Select the menu option **File | | New | Data Comparison** or click the **Data Comparison**
  button in the Standard toolbar (Make sure that you are connected to a data source). If the `Show table selection for new documents` check box on the Data Compare page of the **Options** dialog box is activated (default setting), the **Select Database Objects for Comparison** dialog box pops up automatically.

  If you just want to open a Data Comparison window without selecting a database yet, click **Cancel**. The **Select Database Objects for Comparison** dialog box closes and the empty Data Comparison window is displayed in DatabaseSpy.

10.3   Selecting Tables

Tables are selected for comparison (i) in the **Select Database Objects for Comparison** dialog box or (ii) directly in the Online Browser using a context menu command. For both options an active connection to a data source is mandatory, and both options will automatically open a Data Comparison window.

Selecting tables using a selection dialog box

If you start a new database data comparison in DatabaseSpy and want to choose the tables from both databases in a single step, this will be your preferred method. The **Select Database Objects for Comparison** dialog box can be called in various ways:

- Selecting the menu option **File | New | Data Comparison** or clicking the **Data Comparison** button in the Standard toolbar opens a new Data Comparison window and brings up the dialog box.

- If a Data Comparison window is already open, you can either click the **Browse** button in the title bar of either component or double-click one of the title bars. The **Select Database Objects for Comparison** dialog box opens with the pane for the left or right component, respectively, selected.
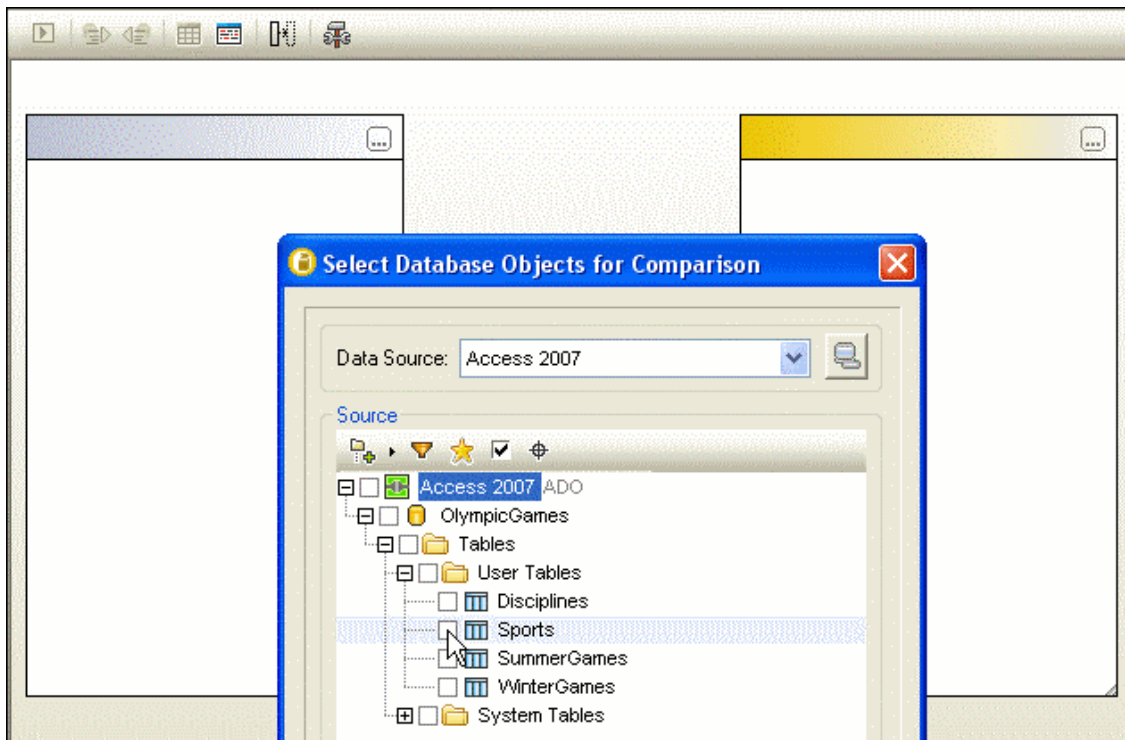
In the **Select Database Objects for Comparison** dialog box, the first connected data source is suggested in the `Data Source` drop-down list. However, all data source connections of the project can be selected in the `Data Source` drop-down list of the dialog box, no matter whether they are currently connected or not. If you choose an unconnected data source, DatabaseSpy automatically establishes a connection.

If the data source containing the tables you want to compare is not included in your project and is therefore not listed in the drop-down list, you can click the **Browse** button in the Data Source

group box to open the **Create a Database Connection** dialog box, where you can create the required data source connection. Note that any data source connection created from within the **Select Database Objects for Comparison** dialog box will be automatically added to your DatabaseSpy Project!

To add tables to a comparison component via selection dialog box:

1. Make sure that a connection to a data source exists in your DatabaseSpy Project.

2. Do one of the following:

   • If you want to open a new Database Comparison window, select the menu option **File |
     New | Data Comparison** or click the **Data Comparison** 0≣ button in the Standard
     toolbar. The **Select Database Objects for Comparison** dialog box opens with the
     Left Side button activated.

   • If a Database Comparison window is already open, clickthe Browse button in the title
     bar, or double-click the title bar of either component.
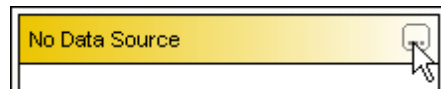
     The **Select Database Objects for Comparison** dialog box opens with either the **Left
     Side** or the **Right Side** button activated, depending on the component you have used
     to call the dialog box.

3. Choose one of your project's data sources from the `Data Source` drop-down list.

4. Expand the data source as required and choose the desired tables by activating the
   respective check boxes. Activating a folder check box selects all the tables that are
   contained in the folder.

5. Click the **Right Side** button (or the **Left Side** button, respectively, if you started from the
   right side) and repeat steps 3 and 4 for the second database.

6. Click **OK**. The selected tables are displayed in the components of the database
   comparison window.

Selecting tables using the Online Browser

If both data sources are connected and the tables to be compared are therefore displayed in the
Online Browser you can use (i) the context menu or (ii) drag and drop to select tables for, or add
tables to, a comparison.

To use the Online Browser to add tables to a comparison component:

1. Select the tables that you want to add to a comparison component in the Online Browser.

2. Right-click and do one of the following:

   • If you want to open a new Database Comparison window, select **Comparison | Show
     in new Data Comparison Document** from the context menu. A new Database
     Comparison window opens and the selected tables are added to the left comparison
     component.

   • If you want to add tables to an existing Database Comparison window, select
     Comparison | Add to Data Comparison Document from the context menu. The
     tables are added to the component that contains the respective data source or, if the
     data source is not yet present in the comparison, to the empty component.

3. Alternatively, if a Database Comparison window is already open, drag and drop tables from the Online Browser to the desired component in the Database Comparison window.

4. Repeat step 2 or 3 for the second database.

10.4 Adding and Removing Tables

If you need to add one or more tables to a database data comparison, you can use the **Select Tables for Comparison** dialog box, the context menu in the Online Browser, or drag-and-drop to do so.

To add tables to an existing data comparison:

- In the Data Comparison window, click the **Browse** button in either component or double-click the title bar of a component to open the **Select Database Objects for Comparison** dialog box. If you have clicked into the left component, the dialog box opens for the left side of the comparison, ditto for the right side.

- Select the tables in the Online Browser, right-click and choose **Comparison | Add to Data Comparison Document** from the context menu. The tables will be added to the component that contains the corresponding data source.

- Select the tables in the Online Browser and drag them to a component that contains the corresponding data source. Note that tables cannot be dragged onto components that contain a data source that does not match the selected tables' data source.

Removing tables from a component

A table is removed from a comparison component when you deselect its check box in the **Select Database Objects for Comparison** dialog box.

To remove a table from a component:

- Open the **Select Database Objects for Comparison** dialog box and deselect the table in the Source group box.

10.5   Mapping Tables and Columns

After you have selected two tables for comparison, DatabaseSpy analyzes the structure of the tables and presents them in the Data Comparison window.

Mapping options

The default comparison options define that tables and columns be mapped automatically according to the table name or column name, respectively, after the second database has been selected. You can change these options on the Data Compare page of DatabaseSpy's **Options** dialog box.
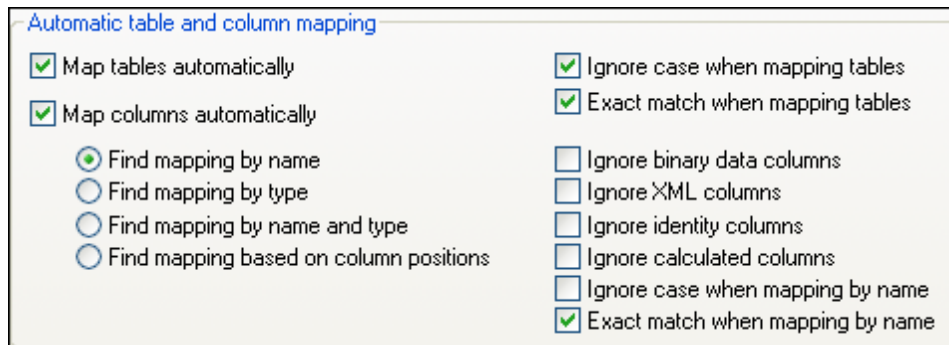
If you do not change these settings, all tables and columns that you include in a component for comparison will be mapped automatically. Mapped tables are connected by bold black lines. You can expand or collapse tables to view or hide the table columns. Mapped columns are connected by blue lines.
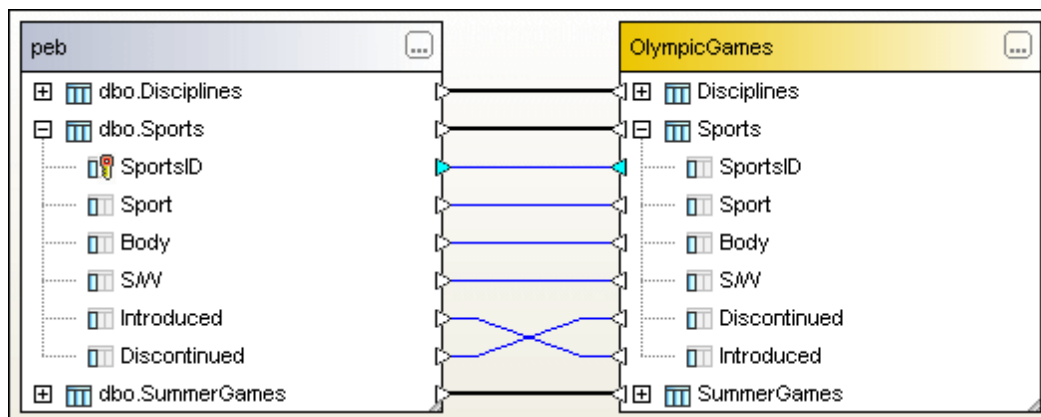
You can disable auto-mapping for the active comparison and for future comparisons by deactivating the `Map tables automatically` check box on the Data Compare page of the **Options** dialog page. When auto-mapping is disabled, you can map columns and/or table in several ways:

- Select the **Map items** option from the context menu
- Draw a mapping line between the two components

To map tables or columns manually:

1. In a Data Comparison window, click the triangle next to a table name and, keeping the mouse button pressed, move the cursor to the corresponding triangle in the second

component. Now triangles are also displayed for the columns of the connected table.

2.   Repeat step 1 for all columns that you want to include in your comparison. Be sure to map the column which is indicated by a cyan triangle since this column is used as a sort criterion and the tables cannot be compared without it.

To map all tables of a component:

•   Right-click the title bar of either the left or the right component and select **Map items** from the context menu.

10.5.1 Mapping Columns

When columns are mapped automatically, you can choose from among the following options on the Data Compare page of the **Options** dialog box:

•   **Column name**: DatabaseSpy checks for matching column names in both tables. Additional options for ignoring case sensitivity or white space are also available on the Comparison Options page of the **Options** dialog box (for all future comparisons) or the Properties window of the active Data Comparison window (for the current comparison), respectively.

•   **Data type**: This option should only be used in small databases where each column has a unique data type to prevent conflicts with multiple columns of the same data type. DatabaseSpy analyzes the tables' data types and assigns them to a set of data type representations which can be compared. This way you could automatically map columns that have a different name but share the same data type.

•   **Name and data type**: Using this option, DatabaseSpy considers first the column name and then its data type when mapping.

•   **Ordinal position**: Columns will only be mapped if the ordinal position at the time of table creation is identical in both tables. This way you can compare tables that are equally designed but use different column names.

Manual mapping

If automatic mapping fails for some reason (e.g., because of a typo in a column name), you can map columns manually by drawing a line between the triangles next to the column names in both components.

In the screenshot below, the "Introduced" column could not be mapped automatically because it is misspelled in the right component. You have to map this column manually.
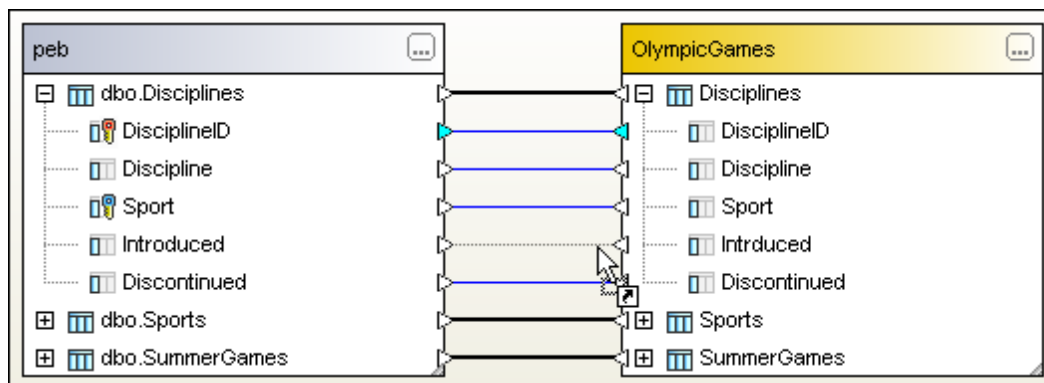
Note that you can only map two columns when their parent tables are also mapped. A mapping

as illustrated in the screenshot below is not allowed and hence impossible.
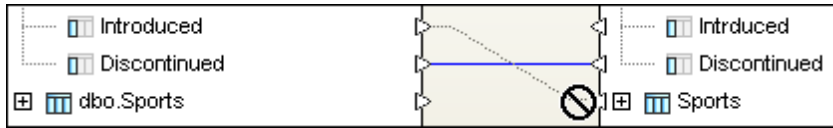
To map columns manually:

1. Click the triangle next to the column name in the first component and, keeping the mouse button pressed, move the cursor to the corresponding triangle in the second component.

2. When the shape of the cursor changes (*see first screenshot above*) release the mouse button. The dotted connection line becomes solid and the two columns are mapped. You can only map columns when the corresponding tables have already be mapped (*see second screenshot above*).

10.5.2   Changing and Deleting Mappings

Mappings that are incorrect or concern columns that should not be compared can be deleted. You can also change the mapping of a table or column and map it to a different table or column in the second comparison component.

DatabaseSpy provides context options for deleting the mapping of the selected object or all objects of the comparison component. In addition, you can use your mouse to change the mapping of individual tables or columns. Note that if you have activated automatic mapping of columns in the Data Compare options, DatabaseSpy tries to map columns automatically when you change the mapping of a table.

To delete mappings:

Do one of the following:

- To delete all mappings of a comparison, right-click the title bar of either component and choose **Unmap items** from the context menu.

- To delete a single table or column mapping, right-click the appropriate object and choose **Unmap selected** from the context menu.

- Click the connection line between two mapped tables or columns and hit the **Del** button.

    Note that unmapping a table will also unmap all columns of that table.

Changing the mapping

If you want to change an existing mapping, you can either delete the incorrect mapping and use one of the methods for manual mapping or simply re-draw the connection line between two tables. Please note that you cannot change the end point of a connection line but have to create a mapping as if no mapping would exist for the table or column in question.
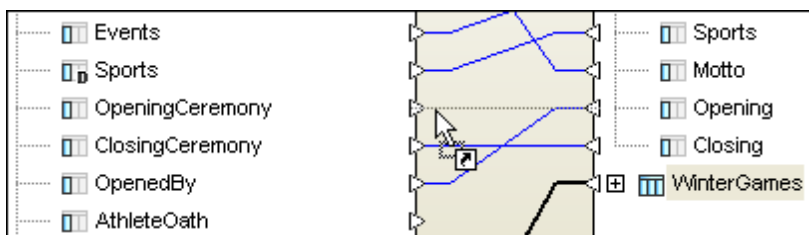
In the example above, the Opening column in the right component has been incorrectly mapped to the OpenedBy column in the left component. To correct this, start at the triangle next to the Opening column on the right side and draw a connection to the OpeningCeremony column on the left side. You could also start at the OpeningCeremony column and connect it with the Opening column. The incorrect mapping between OpenedBy and Opening will disappear when the mouse button is released.

To change an incorrect mapping:

- Ignore the existing mapping and <u>draw a connection</u> between the two tables or columns you want to map.

10.5.3   Sorting Mapped Tables

Tables that are displayed in the components of a Data Comparison window can be sorted either ascending or descending. In addition,DatabaseSpy provides the option to prefer mapped tables in the sort order so that all mapped tables are sorted either ascending or descending, and only then the remaining, unmapped tables are displayed in a component.

The tables in the screenshot below have been sorted using the command **Ascending mapped first**. Note that the column order cannot be changed, only the tables are sorted with the mapped tables being displayed on top.
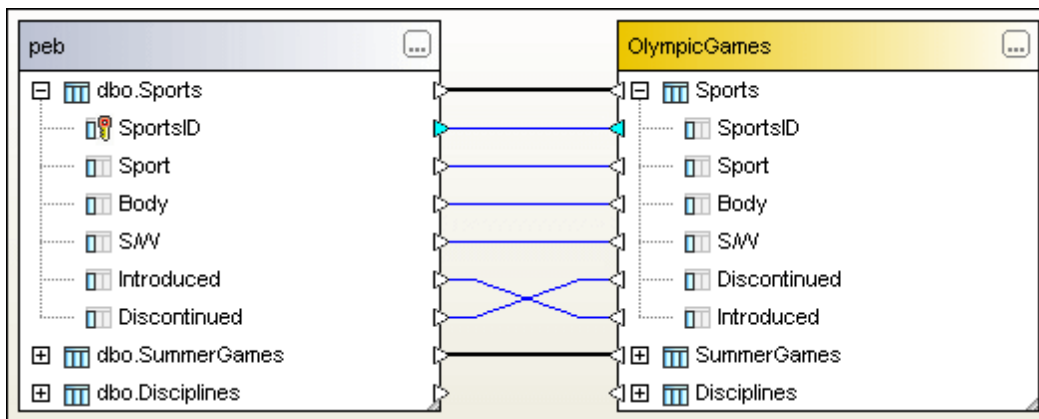
The sort options are available in the **Data Comparison** menu and in the context menu that appears when you right click the title bar of either component or an unoccupied area of the Data Comparison window. Please note, that the sorting commands always apply to both databases, irrespective of which title bar you have clicked.

To sort tables in the Data Comparison window:

1. Do one of the following:
   - Select the menu option **Data Comparison**.
   - Right-click the title bar of either the left or the right component.
   - Right-click into an unoccupied area of the Comparison window.

2. Choose either **Ascending** or **Descending** from the **Sort tables** sub-menu. If you want to exclude unmapped tables from the sort order, choose **Ascending mapped first** or **Descending mapped first**, respectively.

Dragging tables in components

You can always change the position of a table within a comparison component by dragging the table to the desired location. If both components are connected to the same data source, you can also drag a table from one component to the other. The order in which the columns of a table appear in a component is determined by the column's ordinal position and cannot be changed.

10.6   Modifying the Database Comparison Options

The default comparison options are defined on the Comparison Options and XML Compare pages of the **Options** dialog box. These settings apply to all Data Comparison windows that are opened in the future. In the Properties window, you can customize these settings for the current comparison. Any changes you make in the Properties window apply only to the active data comparison and are saved together with the comparison structure in the *-.dbdif file.

To modify the database comparison options:

1. Click into an unoccupied area of the active Data Comparison window or click its tab.

2. Change the comparison options as required in the Properties window.

3. Optionally, save the data comparison file (*.dbdif).

10.6.1   Options for String Comparison

The Comparison Options page of the **Options** dialog box displays the default comparison options for all future Data Comparison windows. Special comparison options for XML fields are defined on the XML Compare page.

Comparison Mode
You can choose from among two comparison modes: The native type considers the data type of the column that is compared, whereas everything will be converted into strings before comparison if you use the string representation type.

Comparison Options
Deactivate the `Ignore case in comparison` check box if upper and lower casing should not be ignored when checking for differences.
Check the `Ignore whitespaces in comparison` option if you do not want to consider whitespaces when comparing database data. Whitespace characters are space, tab, carriage return, and line feed.
If you do not want to make a difference between an empty field and a field containing the NULL

value, activate the `Treat [NULL] as empty` string check box.

Numeric Options

Floating point numbers can be filled up with trailing zeros by activating the `Append trailing zeros to floating point numbers` check box.

Optimization Options

If you do not care to see rows that are equal in both tables of a comparison, you can check the `Keep only different rows in memory` option. In this case, only different rows are kept in the memory and displayed in the Result window.

Execution Options

You can define whether you want to process comparisons of multiple tables sequentially or in parallel. Changes can be committed with or without the use of transactions and you can decide whether or not you want to rollback on errors.

Comparison Options in the Properties Window
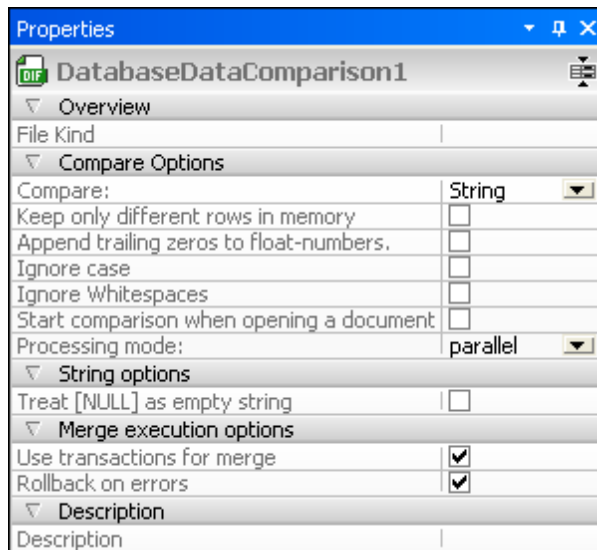
All options described above are also available in the Properties window, if you click the tab of a Data Comparison window. Changes that you make in the Properties window apply only to the active Data Comparison window and are saved together with the data comparison file (*.dbdif).

10.6.2   Options for Comparing XML Fields

The XML Compare page of the **Options** dialog box displays the default comparison options that apply to columns of type XML if native comparison has been selected. The options that are defined on this dialog page are valid for all future Data Comparison windows.

Whitespaces
Whitespace characters are space, tab, carriage return, and line feed. The three options here compare files with whitespace unchanged; with whitespace normalized (i.e., all consecutive whitespace characters are reduced to one whitespace character); and with all whitespace stripped (i.e., not considered for comparison).

Case sensitivity
If the `Ignore case` check box is checked, then case is ignored, and you have the option of ignoring or not ignoring case in node names.

Namespace/Prefix
These are options for ignoring namespaces and prefixes when searching for differences.

Order
If `Ignore order of child nodes` is selected, then the relative position of the child nodes of an element is ignored. As long as an element node with the same name exists in each of the two sets of sibling nodes, the two sets are considered to be equal. Note, however, that if an element node has an attribute, it will always be considered unequal to an element with the same name in the compared sibling set—even if the `Ignore order of child nodes` is selected.

The option of ignoring the order of attributes is also available, and applies to the order of attributes of a single element.

Entities
If `Resolve entities` is selected, then all entities in the document are resolved. Otherwise the files are compared with the entities as is.

Text
If `Ignore text` is selected, then differences in corresponding text nodes are not reported. A different string inside a tag will be ignored since only the XML structure is compared but not the text content of the tags.

Depth

If `Ignore node depth` is selected, then the additional depth of any element (i.e., more levels of descendants) relative to the depth of the corresponding element in the compared file is ignored.

Ignore node types
Check the node types that will **not** be compared in the Compare session. Node types that may be ignored are Attributes, CDATA, Comments, Processing Instructions, DOCTYPE statements, and XML declarations.

XML Comparison Options in the Properties Window
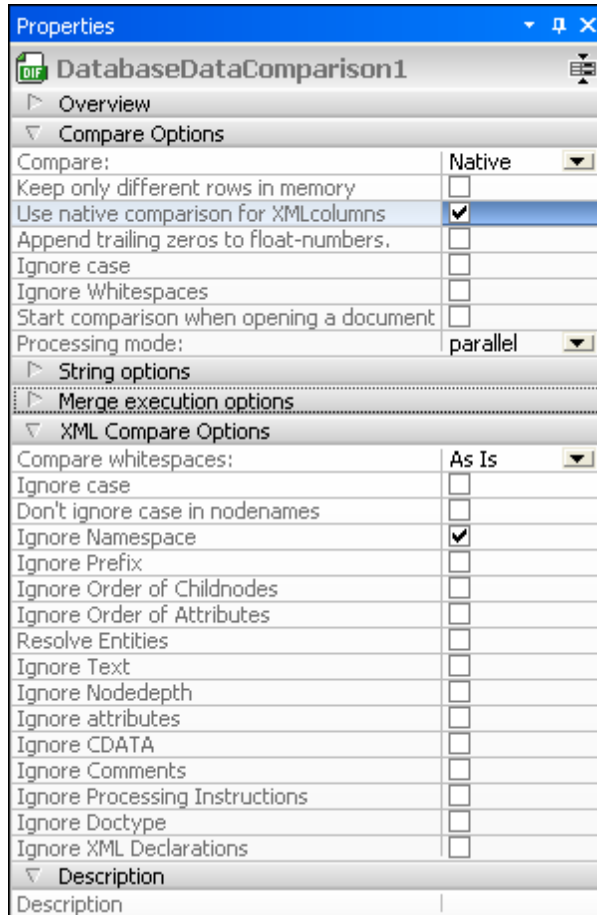
To display the comparison options for XML, you have to activate the `Use native comparison for XML columns` check box, which is only available if you choose "Native" from the `Compare` drop-down list in the Compare section.

10.7   Saving Database Data Comparison Files

After you have [selected]() and [mapped]() tables for comparison and defined the appropriate [comparison options](), you can save the comparison in a data comparison file that can be [added]() to a project. This is either done via the **File | Save As...** command or by right-clicking the tab of a Data Comparison window and choosing the **Save...** options from the context menu. If you do a particular comparison on a regular basis, the use of a data comparison file which can be opened directly and establishes the needed data source connections automatically may save you a considerable amount of working time.

10.8   Running a Database Data Comparison

After you have selected and mapped the tables and columns you want to compare, you can start a database data comparison in DatabaseSpy. Either one or more selected tables or all tables included in the active database data comparison can be compared. A comparison can be started

(i) via the **Data Comparison** menu, (ii) the **Start Comparison** [▶] button in the Data Comparison window, or (iii) the context menu. A comparison starts automatically when a *.dbdif document is opened in DatabaseSpy and the `Start comparison when opening a document` check box is activated in the properties of a data comparison file. In the background, SQL SELECT statements are generated and executed for, and data is retrieved from, both tables.
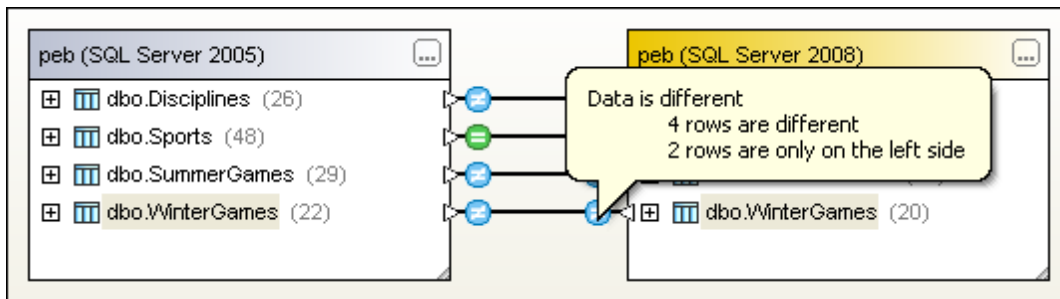
To run a database data comparison:

- To compare all tables in the comparison component, do one of the following:

  - Select the menu option **Data Comparison | Compare tables**.

  - Click the **Start Comparison** [▶] button in the toolbar of the active comparison window.

  - Right-click the title bar of either component and select **Compare items** from the context menu.

  - Press **F5**.

- To restrict comparison to only some of the tables in the component:

  1. Select one or more tables in either component.

  2. Right-click and choose **Compare selected** from the context menu or press **F5**.

- To start a comparison automatically when a *.dbdif file is opened:

  1. Save the database data comparison in to a *.dbdif file.

  2. Activate the `Start comparison when opening a document` check box in the Properties window.

The result of a comparison is indicated with icons in the Data Comparison window. When placing the mouse cursor over an icon, a balloon help appears with a brief description of the comparison result for the selected table (*see screenshot above*). It makes no difference whether you hover over the left or over the right icon. In huge tables which contain a lot of columns, you can collapse and expand individual tables when you are examining the differences. If you double-click a column in one component, both versions of the column are collapsed or expanded, respectively, simultaneously in both components. To collapse all columns in both components with a single click, use the **Collapse tables** command from the **Data Comparison** menu or the context menu that opens when you right click the title bar of a component. Furthermore, the result is displayed in the Message window that automatically opens after the comparison is completed. You can expand the messages to display more detailed information.
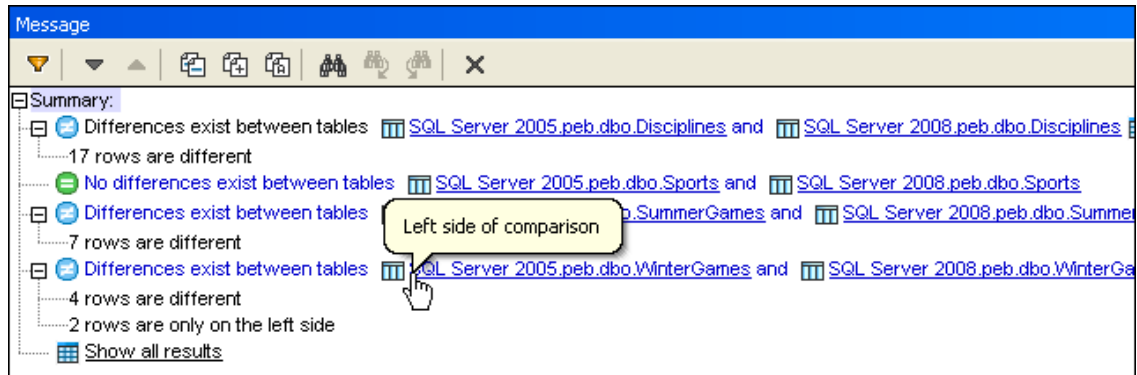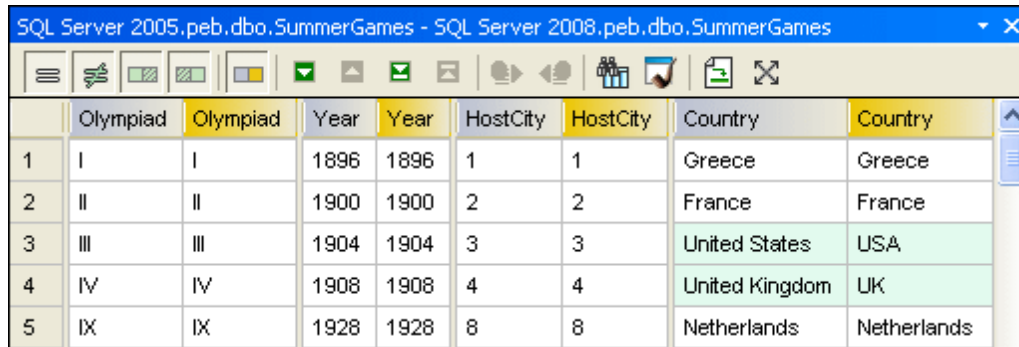
The underlined parts of the messages are hyperlinks. By clicking the links, you can jump either to the individual tables in the left or right component, or open the result view for the individual comparisons. Clicking the `Show all results` hyperlink at the bottom of the Message window will open individual result views for all tables that are included in the comparison.

If you place the mouse cursor over one of the table names, a balloon help appears and indicates whether the tables constitutes the left or the right side of the comparison.

10.9   Displaying Differences in Databases

The result of a database comparison is displayed as an overview immediately in the Message window, and in a balloon help when you hover over one of the comparison icons. A detailed view of the differences is available in the Result window where you can check every table cell for differences.

	Olympiad	Olympiad	Year	Year	HostCity	HostCity	Country	Country	
1	I	I	1896	1896	1	1	Greece	Greece	
2	II	II	1900	1900	2	2	France	France	
3	III	III	1904	1904	3	3	United States	USA	
4	IV	IV	1908	1908	4	4	United Kingdom	UK	
5	IX	IX	1928	1928	8	8	Netherlands	Netherlands	

The Result window for comparisons is a variation of the normal Result window where the columns of the compared tables are displayed next to each other in the result grid. This way, you see the content of the compared columns side by side. Differences are indicated with colors.

Hiding columns without differences

In tables with a large amount of columns where only some of the columns contain differences, you can hide the columns that contain no differences and show only different columns in the result grid. Click the **Show/Hide columns which don't have any differences** button for this purpose.

Searching text

If you search for a particular string in the Comparison Result window, you can use the **Find** button in the toolbar.

Displaying large data cells

If the Data Inspector window is not visible in DatabaseSpy, you can click the **Show the Data Inspector window** button in the toolbar. This pops up the Data Inspector which displays the content of the currently selected cell.

Maximizing the view

The **Maximize the result view** button allows you to maximize the size of the result grid so as to show as many rows of the comparison as possible. If you then would like to check the mapping once again, you can click the **Go to table mapping** button.

To show the results of a table comparison:

1. Run a table comparison.
2. Do one of the following:

   * In the Data Comparison window, click one of the comparison result icons for the

---

compared table.

- In the Message window, click ⊞ Show result window.

- In one of the comparison components, right-click a table and choose **Show selected results** from the context menu.

- In the Data Comparison window, click the **Show result** ⊞ button in the toolbar.

- Right-click the title bar of a comparison component and choose **Show results** from the context menu.

- Select the menu option **Data Comparison | Show results**.

3. Optionally, if you want to hide columns where no differences occur, click the **Show/Hide columns which don't have any differences** ▭ button.

10.9.1 Configuring the Comparison Result View

On the Result View page of the **Options** dialog box, you can customize the appearance of the Result View.
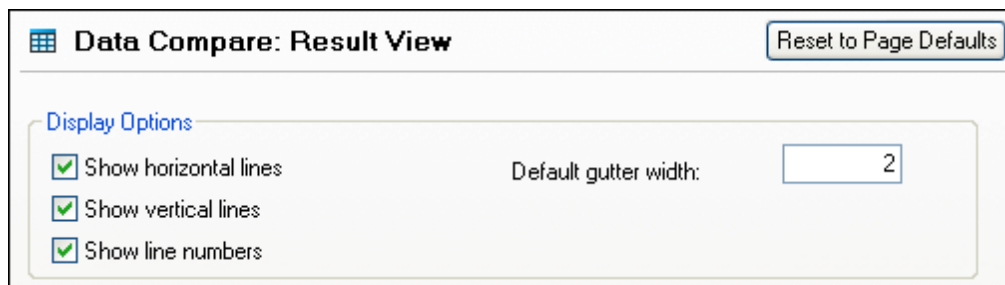
Display Options

You can show horizontal and/or vertical lines as well as line numbers in the result grid by activating the respective check boxes. The default gutter width (i.e., the space between the individual column pairs) can also be defined on this page.

Controlling what rows are displayed

You can configure the view of individual Comparison Result windows to display rows on the basis of their compared status. For example, you can opt to not display all equal rows, or to display rows that exist only in the left component but to not display files that exist only in the right component. To do this, you use a set of simple toggle commands that are available as toolbar icons.

=	**Show/Hide all equal rows**: Shows or hides rows that are equal in the left and right component.
⇶	**Show/Hide all different rows**: Shows or hides rows that are different in the left and right component.
▱	**Show/Hide rows that are only on the left**: Shows or hides rows that exist only in the table that is contained in the left component.
▱	**Show/Hide rows that are only on the right**: Shows or hides rows that exist only in the table that is contained in the right component.

Show/Hide rows that cannot be merged from left to right: Shows or hides rows that are different in the left and right component and can be merged only from the right to the left side and not from the left to the right side.

Show/Hide rows that cannot be merged from right to left: Shows or hides rows that are different in the left and right component and can be merged only from the left to the right side and not from the right to the left side.

In addition, you can also hide all equal columns in the result grid and show only those columns where differences occur.

Show/Hide columns which don't have any differences: A toggle command that toggles the display of columns without differences on and off.

10.9.2   Navigating Differences

You can navigate through the differences by using the following navigation commands in the context menu and toolbar:

Next Difference: Selects the next difference as the current difference.

Previous Difference: Selects the previous difference as the current difference.

Last Difference: Selects the last difference in the document as the current difference.

First Difference: Selects the first difference in the document as the current difference.

Please note: The navigation commands move the selection of the current difference on row level. If there is more than one different cell in a row, you have to use the mouse cursor or the arrow keys to select any other cell than the first in a row.

10.10 Comparing Structure from within a Data Comparison

You can open a schema comparison from within a data comparison if you want to check the structure of the schema that contains a particular column. The selected tables and their corresponding tables in the other component of the comparison open in a new Schema Comparison window and are mapped automatically. If a corresponding table does not exist in the opposite component, no table is opened in the second component.

You can continue to work in the Schema Comparison window as in a regular database schema comparison session (see Comparing Database Schemas).

To compare the structure of selected tables:

1. Select one or more tables in either component of a Data Comparison window.
2. Right-click and choose **Open selected in new Schema Comparison** from the context menu or click the icon in the Standard toolbar.
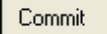
To compare the structure of all tables contained in a component:

- Right-click the title bar of either component and choose **Open in new Schema Comparison** from the context menu.

10.11 Editing Compared Tables

In the Comparison Result view, data **cannot be edited directly**. You can either merge the content of two compared cells (i.e., copy the content of one cell to the other cell) or retrieve data and open a new result view in the editing mode.

To edit compared tables:

1. Click the **Go to table mapping** button to jump to the table mapping of the table you want to edit.

2. In the Data Comparison window, select the column (and the primary key column(s) of the table).

3. Right-click and select **SQL and Data | Edit Data** from the context menu.

4. Make the necessary changes in the new result window that opens in the editing mode.

5. Click the [Commit] button.

6. In the Data Comparison window, right-click the edited table and choose **Compare selected tables** from the context menu.

10.12 Merging Database Data

After you have started a database comparison in DatabaseSpy, you have several options for merging the data between the two tables:

- Merging one or more selected table(s) (left to right or right to left) in the Data Comparison window
- Merging all tables in the Data Comparison window
- Showing and executing the merge script in the SQL Editor
- Merging single cells (left to right or right to left) in the Comparison Result window

The following icons are used for merging (note that the same icons are used for different commands, depending on the menu command you select or the toolbar button you click):

Merge selected Left to Right: This command is available as toolbar button and as a context menu option in a database comparison window when at least one table is selected in the left or right component. Clicking this command replaces the table in the right component with the table in the left component.

Merge selected Right to Left: This command is available as toolbar button and as a context menu option in a database comparison window when at least one table is selected in the left or right component. Clicking this command replaces the table in the left component with table in the right component.

Merge Left to Right: This command is available in the **Data Comparison** menu and in the context menu that opens when you right-click the title bar of either component in a database comparison window. Clicking this command replaces all the tables in the right component with the corresponding tables in the left component.

Merge Right to Left: This command is available in the **Data Comparison** menu and in the context menu that opens when you right-click the title bar of either component in a database comparison window. Clicking this command replaces all the tables in the left component with the corresponding tables in the right component.

Merge data from left to right: This command is available as toolbar button and as a context menu option in the Comparison Result window when at least one cell is selected. Note that it is unimportant whether the left or the right column is selected in the result grid; data will nevertheless be copied to the right table even if the right column is selected.

Merge data from right to left: This command is available as toolbar button and as a context menu option in the Comparison Result window when at least one cell is selected. Note that it is unimportant whether the left or the right column is selected in the result grid; data will nevertheless be copied to the left table even if the left column is selected.

10.12.1 Merging Tables

When you decide to merge tables in your components, DatabaseSpy displays a dialog box providing an overview of the actions to be carried out. You must confirm the merge before the changes are actually committed to the database.
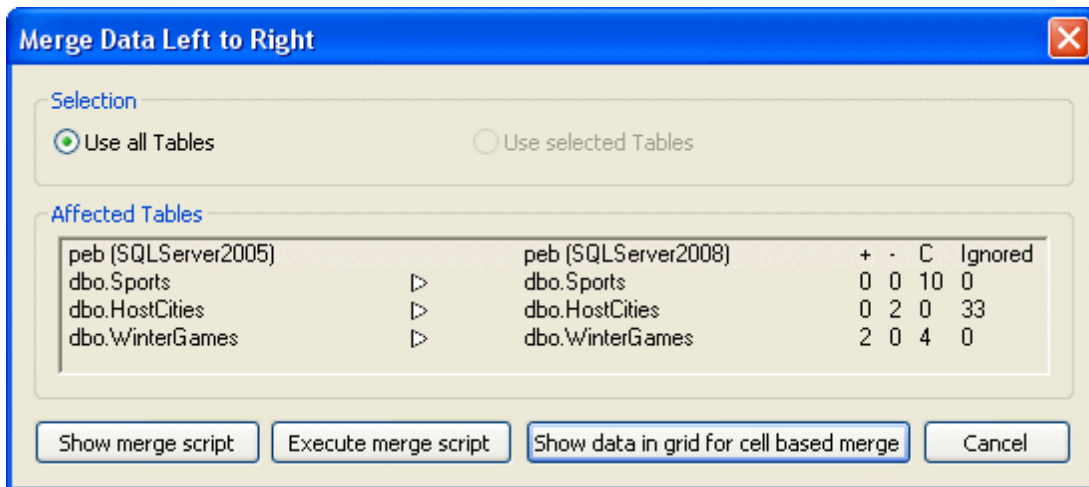
DatabaseSpy displays the databases and the affected tables, and indicates the direction in which the merge will take place. The **Show merge script** button allows you to view the changes in detail before the change script is executed. If you have selected at least one table before calling the merge command, the `Use selected Tables` radio button is selected and you can choose in the dialog box whether you want to display (and merge) only the selected or all tables.
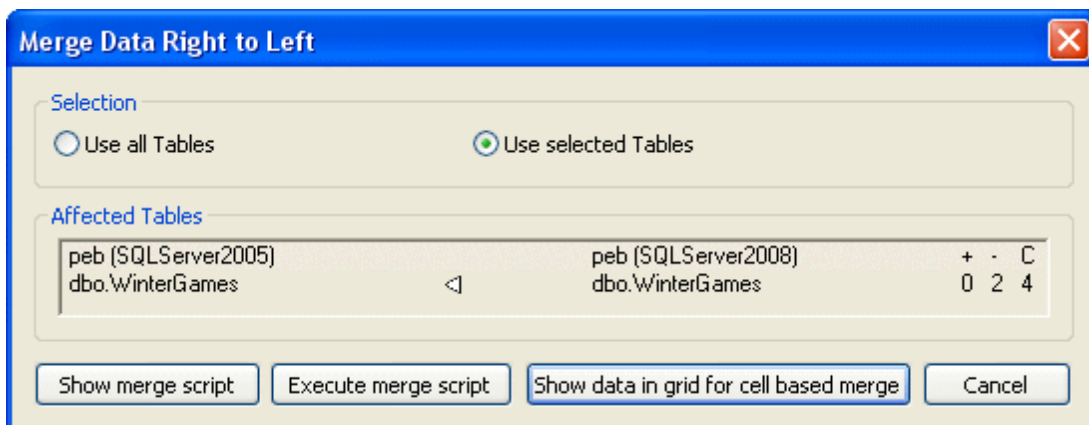
To merge all compared tables:

1. Select and map the required tables and run a database data comparison.

2. Check the comparison result and decide which table(s) should be kept.

3. Select the appropriate menu option from the **Data Comparison** menu:

   • To copy the data from the left table, choose **Merge Left to Right**.

   • To copy the data from the right table, choose **Merge Right to Left**.

4. Do one of the following:

   • Click **Execute merge script** to commit the changes to the database.

   • Click **Show data in grid for cell based merge** if you want to merge only selected data. A Comparison Result view opens for each selected table.

   • Click **Show merge script** to view—and edit, if required—the change script in an SQL Editor window. Execute the change script.

To merge selected compared tables:

1. Select and map the required tables and run a database comparison.

2. Check the comparison result and decide which table(s) should be kept.

3. Select the table(s) that you want to merge.

4. Select the appropriate menu option from the context menu or click the corresponding button in the toolbar of the Data Comparison window:

   • To copy the data from the left table, choose **Merge selected Left to Right** or click the **Merge selected from left to right** button.

   • To copy the data from the right table, choose **Merge selected Right to Left** or click the **Merge selected from right to left** button.

5. Do one of the following:

   • Click **Execute merge script** to commit the changes to the database.

   • Click **Show data in grid for cell based merge** if you want to merge only selected data. A Comparison Result view opens for each selected table.

   • Click **Show merge script** to view—and edit, if required—the change script in an SQL Editor window. Execute the change script.

10.12.2 Displaying the Merge Script

You may want to edit the changes before they are committed to the database. For this purpose, DatabaseSpy provides a merge script that opens in an SQL Editor and can be edited.
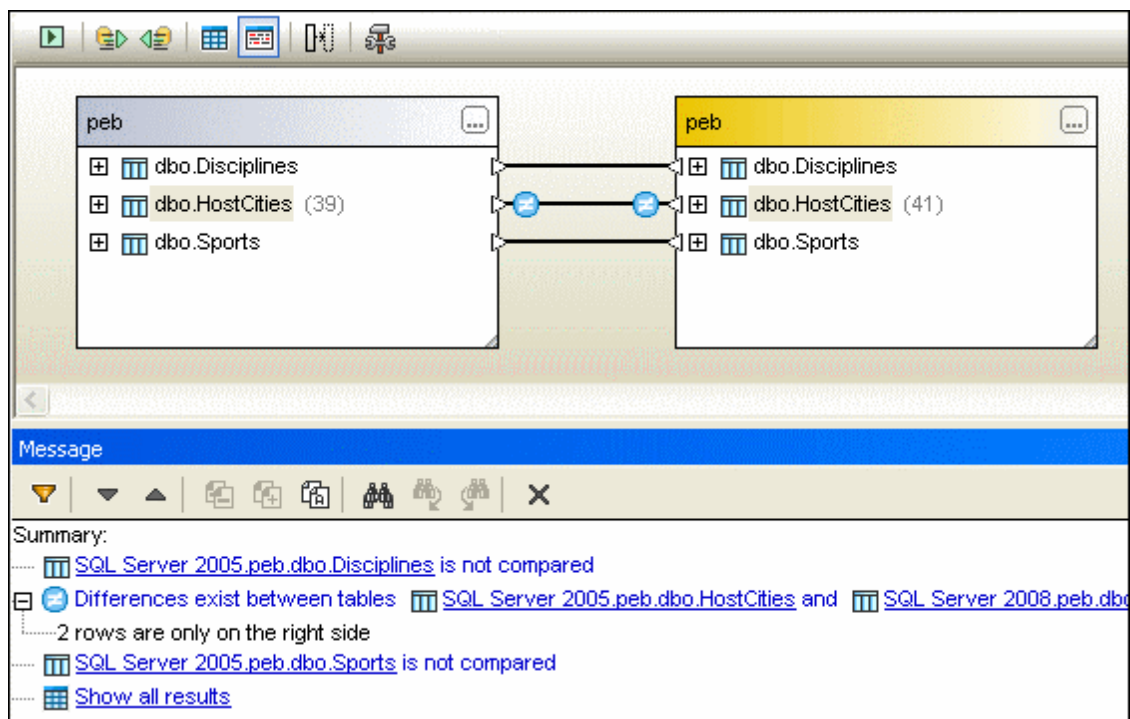
If you choose to merge from right to left in the comparison illustrated above, the merge script would look something like that:

```
1     -- region merge from dbo.HostCities to dbo.HostCities
2     INSERT INTO [peb].[dbo].[HostCities] ([HostCityID], [City],
      [Country], [S/W], [Year], [Comment])
3     VALUES (40, 'Vancouver', 'Canada', 'Winter', '2010', '[NULL]') ;
4     INSERT INTO [peb].[dbo].[HostCities] ([HostCityID], [City],
      [Country], [S/W], [Year], [Comment])
5     VALUES (41, 'Sochi', 'Russia', 'Winter', '2014', '[NULL]') ;
6     -- endregion merge from dbo.HostCities to dbo.HostCities
```

Since the merge script opens in a new SQL Editor window, you can also edit the script before executing it.

To display the merge script for a comparison:

- Right-click a table in a comparison component and do one of the following:

    - Select **Show merge script: Left to Right** or **Show merge script: Right to Left**, respectively, from the **SQL and Data** sub-menu.

    - Select **Merge selected Left to Right** or **Merge selected Right to Left**, respectively, and click **Show merge script**.

- Select a table in a comparison component and choose the menu option **Data Comparison | SQL and Data | Show merge script: Left to Right** or **Show merge script: Right to Left**, respectively.

- Display the results of a table comparison and do the following:

    1. Select a cell that is different in the compared tables and click the **Merge data from left to right** 🔁 or the **Merge data from right to left** 🔁 button, respectively, or right-click and choose the corresponding commands from the context menu.

    2. In the **Merge Data** dialog box that appears, click the **Show merge script** button.

10.12.3 Merging Individual Results

In the Comparison Result view, you can merge single cells in the result grid, that is, copy individual cells from one column of a comparison pair to the other. This enables you to copy data from both sides of a comparison in the same result grid. Note that the changes cannot be undone and that not all rows may be merged in both directions. When rows are present only on one side of a comparison, only the entire row can be copied to the other side or deleted, respectively!

Cells that cannot be merged in both directions are marked with little arrows in red and green in the result grid, where the green arrow indicates the direction in which the merge is possible, and the red arrow indicates the direction in which a merge is not possible. If columns cannot be merged at all, an icon with two red arrows is displayed. In the example below, the left table uses the 3-digit country code and the data type of its Country column is varchar(3). The right table uses the full name of the country and stores it in a Country column of type varchar(100). Since the full names would be truncated when being copied to the 3-digit column in the left table, only merging from left to right is possible in this case.

Country	Country
GRE	Greece
FRA	France
USA	USA
GBR	United Kingdom

If you try to merge tables that contain such incompatible values, the entire row in which such a merge problem occurs is ignored during the merge. DatabaseSpy displays a message box where the changes are listed and that you have to commit before the merge is started.

Merge data (Right to Left)

Please select what to merge:

◉ Merge the selected rows and cells
○ Merge the whole table

The following changes will occur:

Insert:	0	rows
Delete:	0	rows
Change:	3	rows.

WARNING:
2 rows are ignored, because of incompatible values!

Do you wish to continue?

[Show Mergescript]   [Yes]   [No]
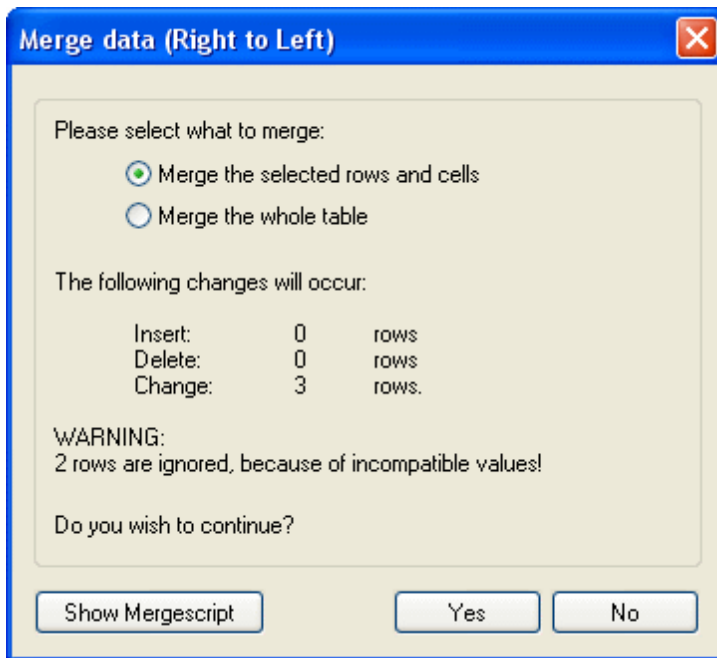
In the **Merge Data** dialog box, you can still decide to merge all rows of the table or to show the merge script prior to merging.

The commands **Merge data from left to right** and **Merge data from right to left** are available as toolbar icons as well as context menu options.

To merge individual cells in the Comparison Result view:

1. In the Comparison Result view, select one or more cells that are different in the left and right table, and do one of the following:

   - Click the **Merge data from left to right** [icon] or **Merge data from right to left** [icon] button, respectively, which are located in the Comparison Result toolbar.

   - Right-click and select **Merge data from left to right** or **Merge data from right to left**, respectively, from the context menu.

2. In the **Merge data** dialog box that appears, click **Yes** to commit the changes to the database.

10.12.4 Reverting a Merge

In addition to merge scripts that allow you to view and edit the changes that occur during a merge before actually committing them to the database, DatabaseSpy also provides restore scripts allowing you to undo a merge.

The restore script corresponding to the merge script that has been used to insert two rows into the HostCities table in chapter "Displaying the Merge Script" is depicted below:

```
1      -- region revert merge from dbo.HostCities to dbo.HostCities
2      DELETE FROM [peb].[dbo].[HostCities] WHERE ([peb].[dbo].
       [HostCities].[HostCityID]=41);
3      DELETE FROM [peb].[dbo].[HostCities] WHERE ([peb].[dbo].
       [HostCities].[HostCityID]=40);
4      -- endregion revert merge from dbo.HostCities to dbo.HostCities
```

Please note: You must generate the restore script **before** you execute the merge script, just like you have to create a backup file before changing the file!

To display the restore script for a comparison:

Do one of the following:

- Right-click a table in a comparison component and select **Show restore script: Left** or **Show restore script: Right**, respectively, from the **SQL and Data** sub-menu.

- Select a table in a comparison component and choose the menu option **Data Comparison | SQL and Data | Show restore script: Left** or **Show restore script: Right**, respectively.

Chapter 11

Comparing Database Schemas

11 Comparing Database Schemas

A database schema comparison enables you to compare objects of two different database schemas, in terms of their structure or size. Database objects that can be compared include tables, columns, views, functions, and stored procedures. Comparison works differently for tables, as opposed to views, functions, or stored procedures. Namely, in case of tables, the comparison results report the structural differences (such as different columns, constraints, data types, and so on). In case of views, functions, and stored procedures, the comparison results inform you if the size of the object is the same or different in database A (left-side of comparison) as compared to database B (right-side of comparison).

Note: Line-by-line text comparison of database object definitions is not supported. To compare two object definitions (for example, database views) line-by-line, first save the definitions to files and then open both files in a differencing application such as DiffDog.

To start a database schema comparison, click the **Schema Comparison** [⬛] toolbar button. Two database components are displayed side by side (a "left" component and a "right" component). Before running the actual comparison, you need to assign a data source to each of the left and right components. Then you can select for comparison either all objects of a schema, or specific schema objects. To change the objects included in comparison, click the **Browse** [⬛] button on the right or left component, and select or clear the required check boxes. For more information, see Running a Database Schema Comparison.
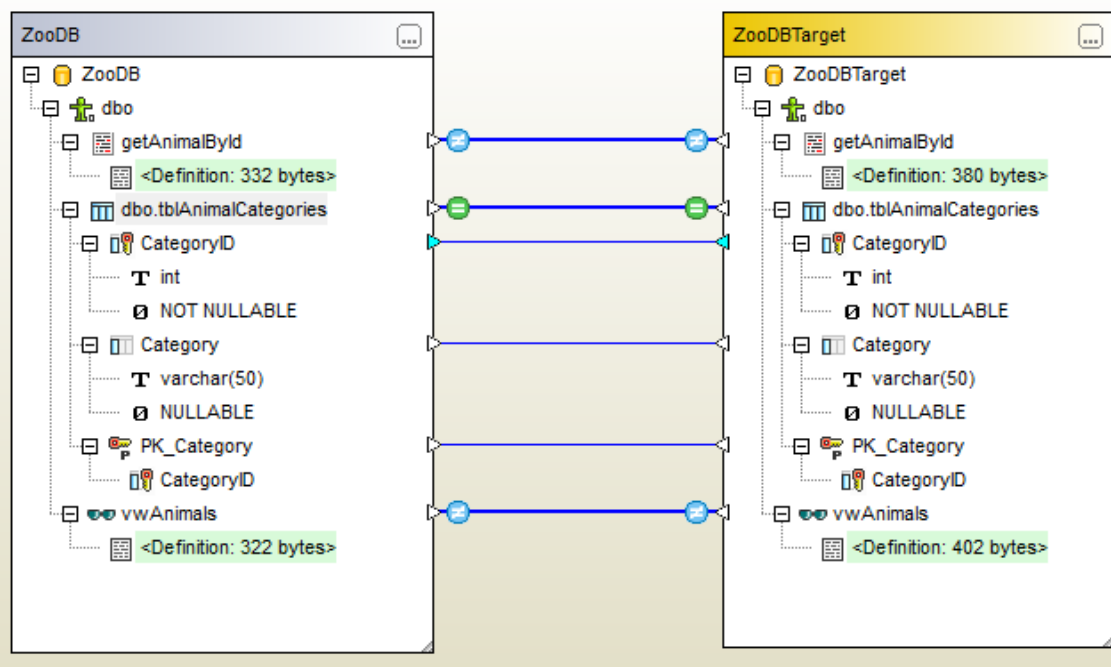
Sample database schema comparison

The sample database schema comparison above illustrates the result of comparing a stored procedure, a table, and a view between two SQL Server databases, "ZooDB" and "ZooDBTarget". The table definitions are equal in both the source and target schema (which is indicated by the [⬛] sign), while the stored procedure and view definition is different (which is indicated by the [⬛]

sign).

A database schema comparison can be saved as a file. Database schema comparison files have the **.dbsdif** extension, and can also be opened with DatabaseSpy and added to DatabaseSpy projects. A comparison file stores the connection details of the database(s) involved in comparison, the database objects selected for comparison, as well as any configuration options you have defined in the "Properties" window (see also Database Schema Comparison Files).

Various database comparison options are configurable. For example, you can have DatabaseSpy automatically match the tables or columns to be compared (by name, data type, or both), or perform the mapping manually. The comparison options can be adjusted from **Tools | Options | Data Compare**.

After running a database schema comparison, you can generate a merge script to update either the left or the right database. You can review and edit the merge script before actually executing it. It is also possible to select specific objects to be included in the merge script (for example, one or several columns of a table). For more information, see Merging Database Schemas.

Note: The merge script synchronizes only differences resulting after comparing structure of tables (this includes changes to columns, keys, or constraints). Merging differences between other object types (such as stored procedures, functions, and views) is not supported.

11.1   Running a Database Schema Comparison

A database schema comparison in DatabaseSpy involves the steps shown below.

Step 1: Connect to the databases that are to be compared

1. On the **File** menu, click **New | Schema Comparison**. (Alternatively, click the **Schema Comparison** ⬚ toolbar button.) At this stage, you may be prompted to add a new data source, if no connected data source already exists. In this case, click **Yes** and follow the wizard steps to create a new data source (see [Connecting to a Database](#)).
2. When prompted to select database objects for comparison, first choose a data source from the Data Source list, and then select the check boxes next to the objects you want to include for comparison. The Data Source list displays any active data sources that you've connected since starting DatabaseSpy.
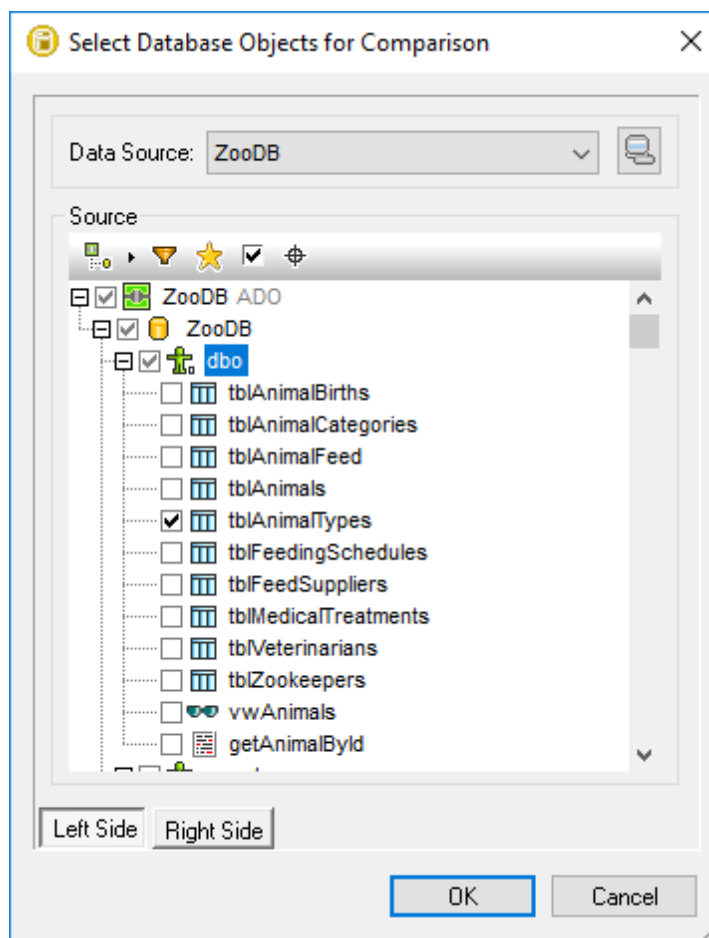
3. Click either **Left Side** or **Right Side** to add the data source to the left or right side of the comparison, respectively, and then click OK. Be patient while DatabaseSpy adds the selected objects to the comparison. At this stage, only one of the comparison components (right or left, depending on what you selected) has an assigned data source.
4. Click the **Browse** ⬚ button of the empty component, and choose the second data source from the Data Source list, if one is available. Otherwise, click **Quick Connect** ⬚

, and follow the wizard steps to connect to the second data source.
5. Select the check boxes next to the objects you want to compare, and then click **Left Side** (or **Right Side**, depending on the case).

Step 2: Map the objects to be compared (if applicable)

To perform a comparison between any two objects that appear in the left and right side of the comparison, the objects must be connected by a line (a "mapping"). The mapping instructs DatabaseSpy to compare these two objects specifically. By default, DatabaseSpy will attempt to match automatically the objects to be compared by their name, and thus create the corresponding mappings. The mapping options can be adjusted from **Tools | Options | Data Compare**. When necessary, you can create mappings manually, or adjust the existing ones. For example, you may need to create a mapping manually between two objects if the object names are different in the "left" as opposed to the "right" database. To create a mapping, click the triangle on the left component and, holding the left mouse button pressed, drag it to a target triangle on the right component.

To delete all mappings of a comparison, right-click the title bar of either component and choose **Unmap items** from the context menu. To delete a single mapping, right-click the appropriate object and choose **Unmap selected** from the context menu. Alternatively, click the connection line between two mapped objects and press **Delete**.

Note: Unmapping a table will also unmap all columns of that table.

Step 3: Run the comparison

Once the mappings between objects are created, you can run the comparison:
* On the **Schema Comparison** menu, click **Compare items**. (Alternatively, click the **Start Comparison** toolbar button.)

The result of a comparison is indicated by the following icons:

The compared objects are equal in both components

Differences exist between the objects in the left and right components

A corresponding item is missing in the opposite component

Schema objects that are different are highlighted with green colour.

To simplify navigation through huge schemas which contain a lot of objects, you can collapse and expand individual objects when you are examining the differences. If you double-click an item in one component, both versions of the item are collapsed or expanded, respectively, simultaneously in both components. To collapse all items in both components, do one of the following:

* On the **Schema Comparison** menu, click **Collapse items**.
* Right-click the title bar of a component, and select **Collapse items** from the context menu.

11.2   Database Schema Comparison Files

After running a database schema comparison, you can save it to a file. This will save you time in future, whenever you want to compare the same database schemas. Database schema comparison (.dbsdif) files store the connection details of the databases involved in comparison, the database objects selected for comparison, as well as any configuration options you have defined in the "Properties" window. (Note that passwords stored in database schema comparison files are encrypted.)

To save a schema comparison to a file:

1. Run a comparison (see Running a Database Schema Comparison).
2. On the **File** menu, click **Save As**, and select "Altova Database Schema Comparison (.dbsdif)" as file type.

To open a schema comparison file:

- On the **File** menu, click **Open**, and browse for the .dbsdif file.

11.3   Merging Database Schemas

After running a schema comparison between two database schemas (the "left" and the "right" one), table differences can be synchronized (merged) between the two databases. Since the merge operation affects the structure of the database, it is not applied directly. Instead, DatabaseSpy generates a merge script. The generated merge script can apply changes either from the "left" database to "right", or the other way round, from "right" to "left".

Note: The merge script synchronizes only differences resulting after comparing structure of tables (this includes changes to columns, keys, or constraints). Merging differences between other object types (such as stored procedures, functions, and views) is not supported.

Once the merge script is generated, you can choose to execute the merge script directly in DatabaseSpy, , or save it to a file for later execution.

Before executing the merge script against the database, it is always recommended to review it. If the structural differences between database schemas are numerous, the generated merge script will be equally big. To avoid a potentially dangerous update of multiple items in the target database through a big script, you could generate smaller scripts and apply them progressively. For example, you can select specific items (columns, constraints) that should be included in the merge, and then generate the script only for selected objects.

To generate and execute a merge script:

1.   Run a database schema comparison (see Running a Database Schema Comparison).
2.   Optionally, if you want to merge only specific items, select these items in a component using **Ctrl+Click**. Note that it has no influence on the direction of the merge whether you select the items in the left or in the right component (for example, you can select items in the right component and still choose to merge from left to right).

     Important: Child elements of a table or column are not automatically included in the merge if you select just the parent. For this reason, when selecting a parent object (for example, a table), make sure to select also any child items that should be merged (such as columns or constraints). For example, it the column name is equal but the data type is different (as shown in the image below), you must select the data type item to create a valid merge script.
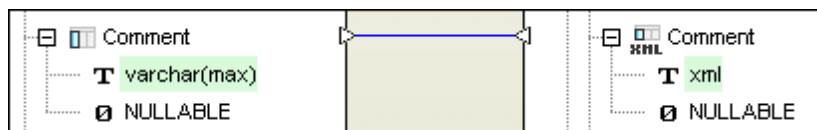
3.   Do one of the following:

     •   On the **Schema Comparison** menu, select **Show merge script: Left to Right**.
     •   Right-click the component, and select **Show merge script: Left to Right** from the context menu

     •   Click the **Merge selected from left to right** toolbar button.
     The commands above assume you want to merge the database structure from left to right. Otherwise, use the corresponding reverse (right to left) commands.

4. Review and edit the merge script as required, and click the **Execute** button to actually merge the schemas.

Chapter 12

XML Schema Management

12   XML Schema Management

Altova web site: XML in databases

If the database supports XML as type, you can load XML instance files into database fields, and save them to local files from the database fields (see Editing XML Columns).

XML support is available for the following databases:

- DB2 9.1, 9.5, 9.7, 10.1, 10.5
- SQL Server 2005, 2008, 2012, 2014, 2016
- Oracle 9i, 10g, 11g, 12c
- PostgreSQL 8.3, 9.0.10, 9.1.6, 9.2.1, 9.4, 9.5

Note: PostgreSQL does not have XML schema management (add schema, drop schema).

If XML schema management is supported by the database, you can register XML schemas with the database. The XML schemas are displayed in the Online Browser in a sub-folder called `XML Schemas`.
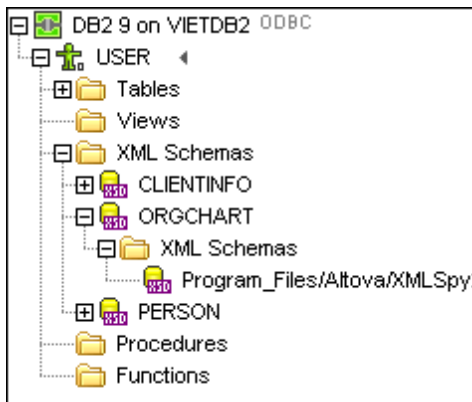
Click the expand button in front of an XML schema name to view dependent/secondary schemas. The properties of the individual XML schemas can be viewed and—where possible—edited in the Properties window.
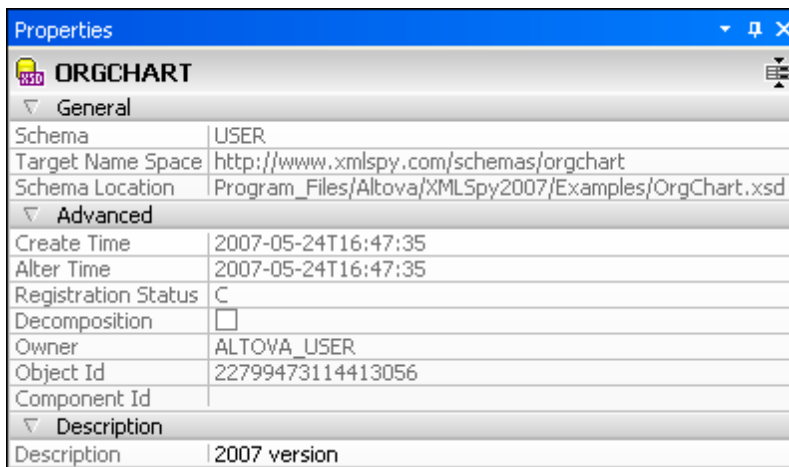
To view an XML schema in detail, you can open it in XMLSpy. For this purpose, a context menu option in the Online Browser as well as a button in the **XML Schema management for databases** dialog box are available. You must have XMLSpy installed on you PC to be able to use this feature.

Viewing XML schemas assigned to a particular column

All the XML schemas that have been assigned to an XML data cell of a table are displayed in the Online Browser in an XML Schemas sub-folder below the respective column. This way, you can see at a glance, which XML schemas are used to validate the content of a particular XML column in the database.
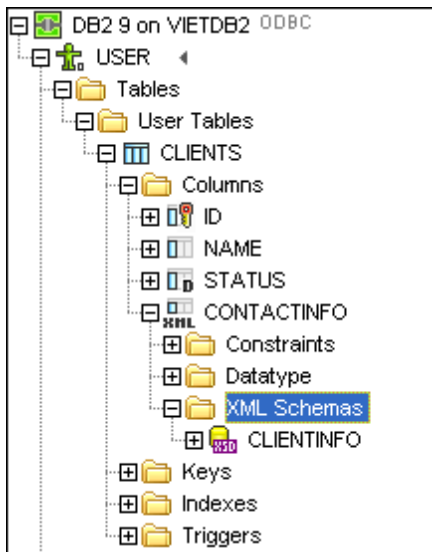
To view an XML schema in XMLSpy:

Do one of the following:

- In the Online Browser, right-click a schema in the XML Schemas folder and select **View in XMLSpy** from the context menu.

- In the **XML Schema management for databases** dialog box, select a schema and click the **View Schema** button, or right-click the schema and choose **View Schema** from the context menu.

The schema opens in XMLSpy.

12.1   Assigning XML Schemas

You can assign an XML schema to a database column or, in DB2 databases, field of type XML. This schema is used for data validation when an INSERT or UPDATE action is performed. The **Choose XML Schema** dialog box lists the XML schema that is currently assigned to an XML column/data cell and allows you to assign an XML schema to the selected column/cell.
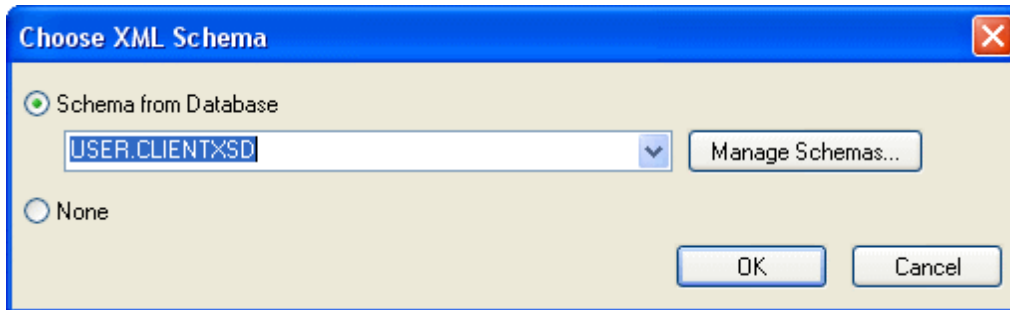
You can choose one of the existing XML schemas that are registered with the database. If you want to add an additional XML schema to the database, click the **Manage Schemas...** button.

After you have assigned an XML schema to an XML column/field, you must commit the changes to the database. At this stage, DatabaseSpy validates the XML file that is contained in the XML column/field against the assigned XML schema and reports any errors to the Message tab. If the validation fails, the XML schema assignment cannot be committed to the database.
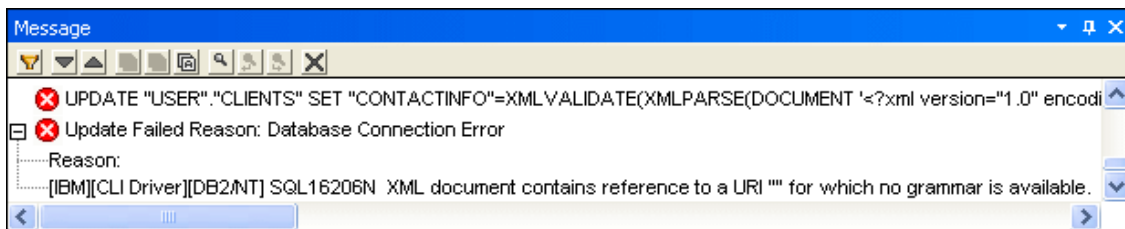
To assign an XML schema to a database column or field of type XML:

1. Depending on the database type, do one of the following:

    - In DB2 databases, and provided that the Result window is in the Editing mode, click the Browse ▦ button in the XML data cell you want to edit, and choose **Assign XML Schema** from the menu that appears.

    - In SQL Server databases, right-click the column in the Online Browser and choose **Assign XML Schema...** from the context menu.

    The **Choose XML Schema** dialog box opens.

2. Do one of the following:

    - Select the `Schema from Database` radio button and choose one of the registered XML schemas from the drop-down list. Optionally, click the **Manage Schemas...** button to add a schema to the database.

    - To remove any XML schema assignment from the column/data cell, select the `None` radio button.

3. Click **OK**.

4.   Depending on the database type, do one of the following:

- In DB2 databases, click the | Commit | button in the Result window.

- In SQL Server databases, execute the change script.

12.2 Managing XML Schemas

To add, modify, or delete XML schemas, DatabaseSpy provides the **XML Schema management for databases** dialog box which can be accessed via a menu command or using a toolbar icon. Unless you are connected to a data source supporting XML, the dialog box appears in the Offline state.
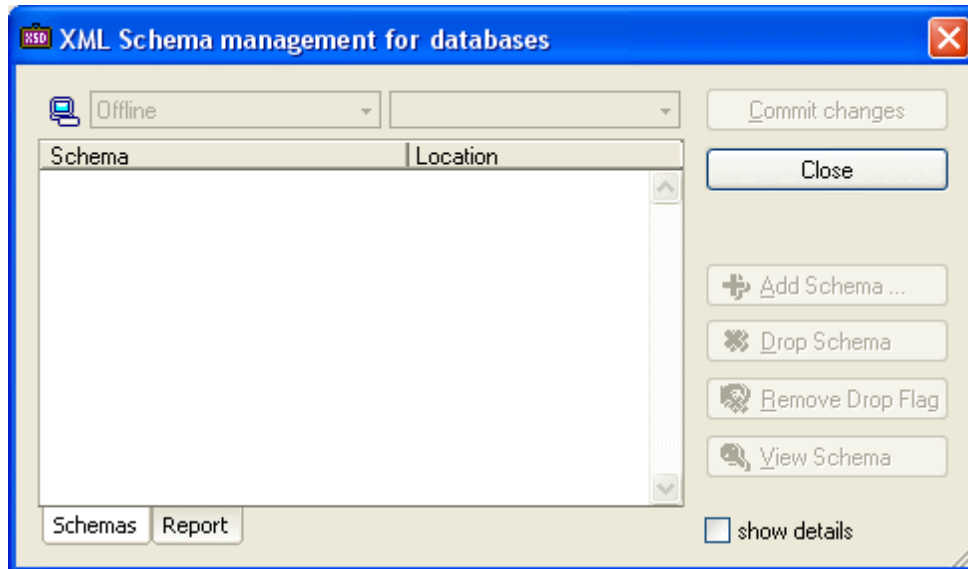
The dialog box provides a **Connect to a database** [icon] button which allows you to call the **Add a Data Source** dialog box and connect to a data source. DatabaseSpy connects to the data source in the background and you can add, delete, or modify XML schemas for this database.

If you are connected to at least one data source, when you open the dialog, you can use the left drop-down list to select a data source connection from the project that is currently open in DatabaseSpy. If you select a connection to a database that does not support XML, or for which XML support is not yet implemented in DatabaseSpy, a corresponding information message is displayed in the list box of the dialog box.

For supported data source connections, DatabaseSpy automatically displays the XML schemas that are registered with the database in the list box of the **XML Schema management for databases** dialog box. You can then add schemas, change them, drop schemas from the database, or select an XML schema and view it in XMLSpy. Any changes you make are not directly committed to the database. Edited objects are marked new (￼), changed (￼), or deleted (￼) in the list box (*see screenshot below*). You must actively commit the changes in order for the changes to become effective.
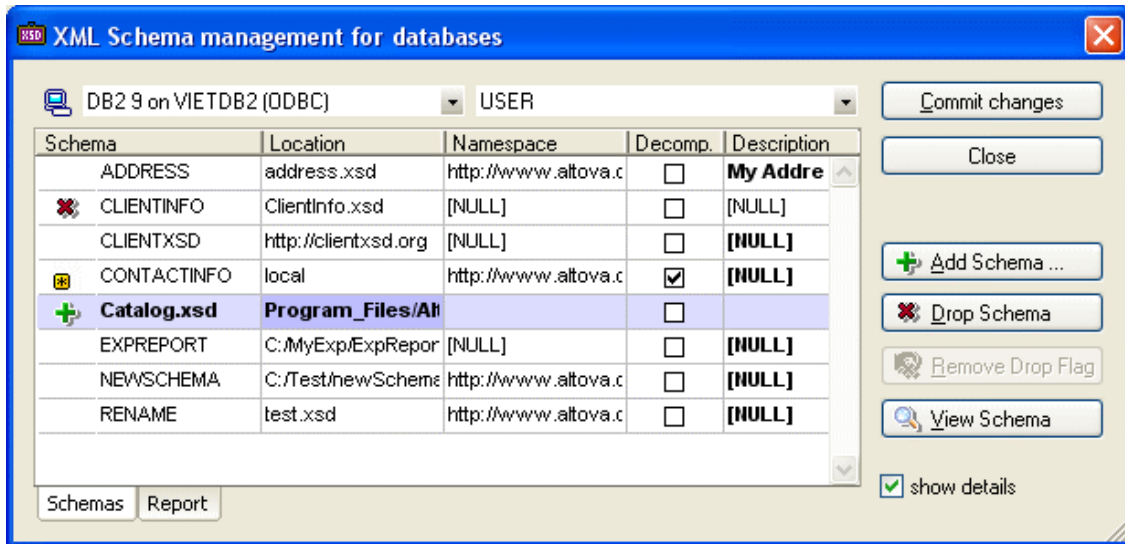
By default, only the `Schema` and `Location` columns are displayed in the dialog box. You can, however, activate the `Show details` check box, which is located at the bottom right of the dialog box, to show also the `Schema name` (if applicable), `Namespace`, `Decomp.` (if applicable), and `Description` columns. If dependent XML schemas are present, you can view these objects by clicking the expand icon in the list box. The dialog box is resizable so as to allow you to conveniently view all the information if the `Show details` check box is activated and additional information is displayed.

All the functions in this dialog box are also available in a context menu that opens when you right-click anywhere in the list box.
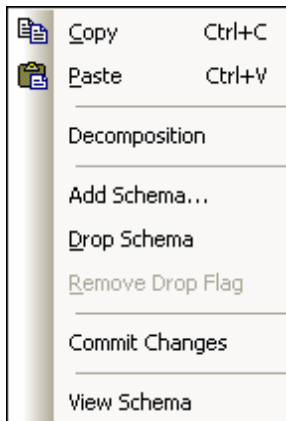
To open the XML Schema management for databases dialog box:

- Select the menu option **Tools | XML Schema Management for Databases**, or click the **XML Schema Management** icon in the Tools toolbar, or select an XML schema in the Online Browser and choose **Manage XML Schemas...** from the context menu.

Alternatively, if the Result window is in the Editing mode:

1. Click the Browse button inside an XML data cell you want to edit and choose **Change XML Schema** from the popup menu.

2. In the **Choose XML Schema** dialog box, select the `Schema from Database` radio button. It is not necessary to select an XML schema from the drop-down list.

3. Click the **Manage Schemas...** button.
    The **XML Schema management for databases** dialog box opens. Please note that the left drop-down list is disabled in this case. Changes can only be made for the active data source.

To connect to a different database:

*   Click the **Connect to a database** icon in the upper left part of the **XML Schema management for databases** dialog box to open the **Add a Data Source** dialog box. The **XML Schema management for databases** dialog box remains open in the background.

To list the available XML schemas of a database connection:

1. In the **XML Schema management for databases** dialog box, select a database from the left drop-down list or connect to a different database as described above.

2. If required, select a database schema from the right drop-down list.

3. Optionally, activate the `Show details` check box to display also namespace, decomposition, and description of the listed XML schemas.

12.3   Adding XML Schemas

You can register additional XML schemas with a database and later assign these schemas to individual XML fields of the database. In the **XML Schema management for databases** dialog box, these newly added XML schemas are marked new ( ) in the list box, and must be confirmed before they are actually registered with the database's XSR repository.
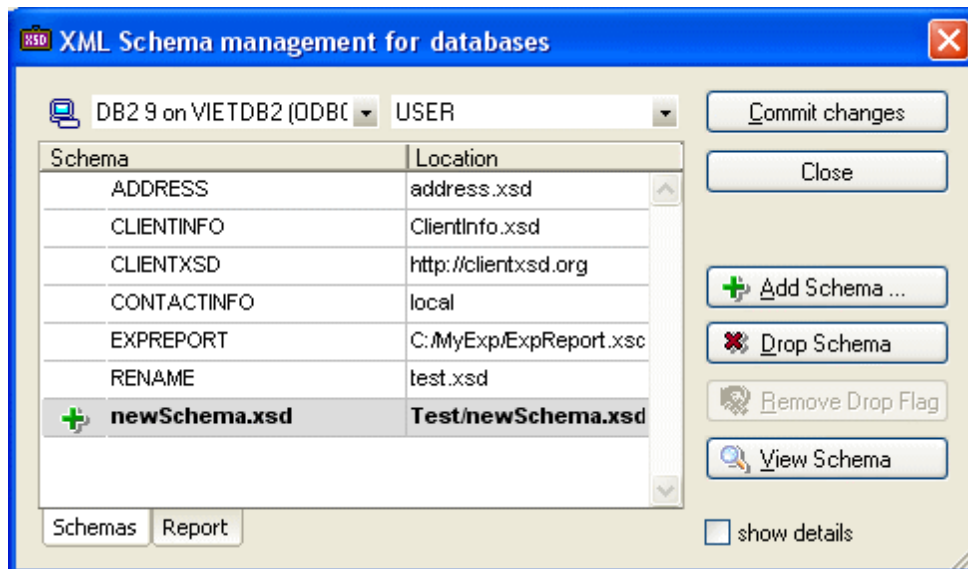
Before you commit the changes to the database, you can view the XML schema in XMLSpy or change its description or decomposition properties in the dialog box.

All actions that you perform in the **XML Schema management for databases** dialog box are reported in the Report tab which works the same way as the Output window in DatabaseSpy.
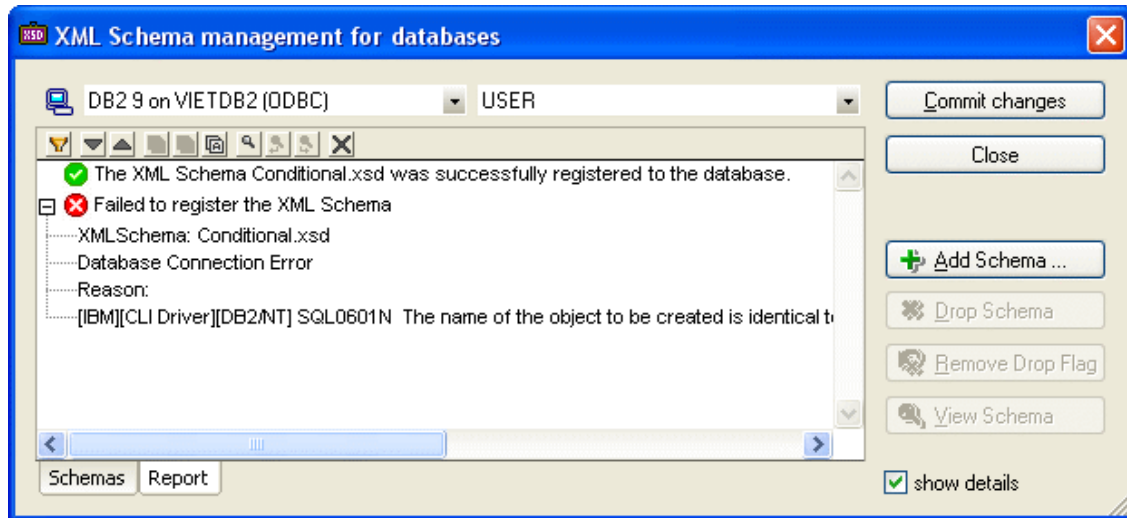
After an XML schema has been successfully registered with the database, it appears in the XML Schemas sub-folder of the Online Browser.

To register an XML schema with the database:

1. In the **XML Schema management for databases** dialog box, select a database connection from the left drop-down list if no data source connection is active.

2. Click the **Add Schema...** button or right-click anywhere into the list box and select **Add Schema...** from the context menu.
    The standard Windows **Open** dialog box appears.

3. Browse to the location of the required XML schema file and click **Open**.
    The XML schema appears in the list of available XML schemas and a **New** ⊹ icon indicates that this XML schema has not been committed to the database yet.

4. Optionally, double-click the file name of the XML schema and edit the name as desired.

5. Optionally, activate the Show details check box and enter a description for the XML file.

6. Optionally, activate the Decomp. check box or right-click the XML schema and activate the **Decomposition** option in the context menu to enable the XML schema for decomposition, if applicable.

7. Optionally, change the database schema in the DB schema column, if applicable.

8. Click the **Commit changes** button.

9. Click **Close** to exit the dialog box. If any changes have not been committed yet, a prompt appears, asking you whether you would like to commit or discard the changes.

12.4 Modifying XML Schemas

DatabaseSpy allows you to change certain parameters of XML schemas that are registered with the database. You may have to activate the `Show details` check box in the **XML Schema management for databases** dialog box to see all the properties.

When you add a new XML schema to the database, you can edit the schema name and the location information in the **XML Schema management for databases** dialog box **before** you commit the new schema to the database.

Once an XML schema has been registered with the database, only the schema description and the decomposition property can be changed. This can be done either in the **XML Schema management for databases** dialog box or in the Properties window.

Please note: Although you can edit an XML schema that you have opened for viewing in XMLSpy, you **cannot save back a schema to the database**. Any changes you make in XMLSpy must be saved in a new XML schema file!

To change XML schema properties:

1. Select the menu option **Tools | XML Schema Management for Databases** or click the

    XML Schema Management 🔲 icon in the Tools toolbar, or right-click an XML schema in the Online Browser and choose **Manage XML Schemas...** from the context menu.

2. In the **XML Schema management for databases** dialog box, select an XML schema in the list box.

3. Optionally, activate the `Show details` check box if the `Namespace`, `Decomp.`, and `Description` columns are not displayed.

4. If applicable, change the status of the `Decomp.` check box as required.
    The XML schema is enabled for decomposition if the decomposition property is checked.

5. If applicable, change the database schema in the `DB schema` column as required.

6. Edit the schema description as required.

7. Click the **Commit changes** button to confirm your changes.

8. Click **Close** to exit the dialog box. If any changes have not been committed yet, a prompt appears, asking you whether you would like to commit or discard the changes.

Alternatively, do the following:

1. Select an XML schema in the Online Browser and edit the properties in the properties window.

2. Execute the change script to commit your changes to the database.

12.5   Deleting XML Schemas

The **XML Schema management for databases** dialog box can also be used to remove an XML schema from a database. Upon clicking the **Drop Schema** button, the schema is marked for deletion (✖) in the list box and you have to confirm the deletion before the schema is actually dropped from the database's XSR repository.
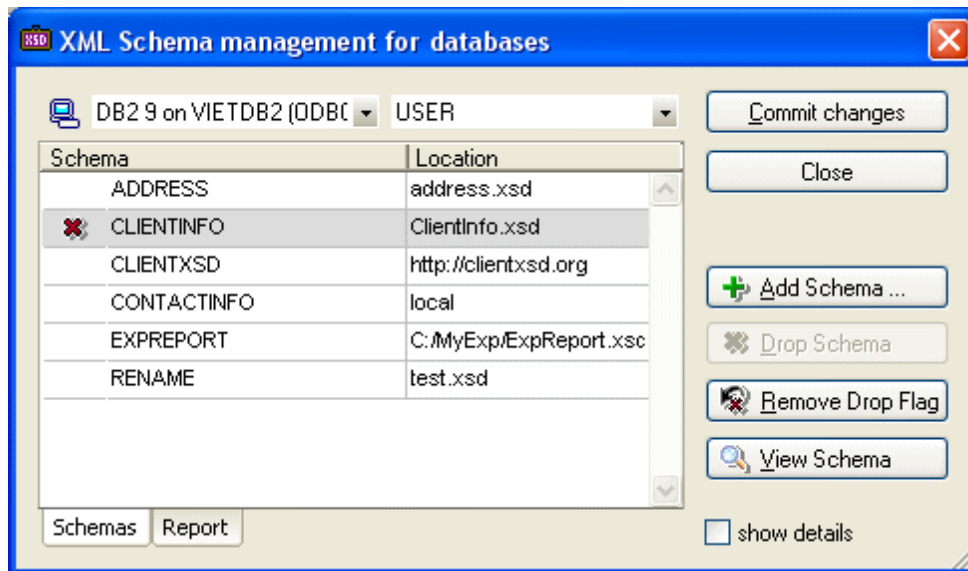
The **XML Schema management for databases** dialog box also provides a **Remove Drop Flag** button that allows you to undo the deletion, provided that the changes have not been committed yet.

Alternatively, you can also use an option in the context menu of the Online Browser to delete XML schemas from the database. The XML schema is marked for deletion in the Online Browser and a database structure change script is generated which must be executed to commit the deletion to the database.

Please note: If you delete an XML schema from the XML Schemas sub-folder of a database column (i.e., from the sub-folder that shows the XML schemas assigned to a certain column of a database table), the XML schema will be deleted from the database's XSR repository and not just removed from the column.

To delete a registered XML schema from the database:

1. In the **XML Schema management for databases** dialog box, select a database connection from the left drop-down list if no data source connection is active.

2. Select the XML schema you want to delete and click the **Drop Schema** button. The XML schema is marked for deletion (✖) in the list box. Alternatively, you can also right-click the schema and select **Drop Schema** from the context menu.

3. Click the **Commit changes** button.

4. Click the **Close** button to exit the dialog box. If any changes have not been committed yet, a prompt appears, asking you whether you would like to commit or discard the changes.

Alternatively, do the following:

1. In the Online Browser, right-click the XML schema you want to delete and select **Delete** from the context menu. Alternatively, you can also select the XML schema and press the **Delete** key.
    A change script is generated in the Database Structure Change Script window and the XML schema is marked for deletion in the Online Browser.

2. In the Database Structure Change Script window, click the **Execute Change Script** button.

To undo the deletion of an XML schema:

1. In the **XML Schema management for databases** dialog box, select an XML schema that is marked for deletion.

2. Click the **Remove Drop Flag** button or select the option **Remove Drop Flag** from the context menu.

3. Click the **Close** button. A prompt appears, asking you whether you would like to commit or discard the changes.

4. Click **Yes**.
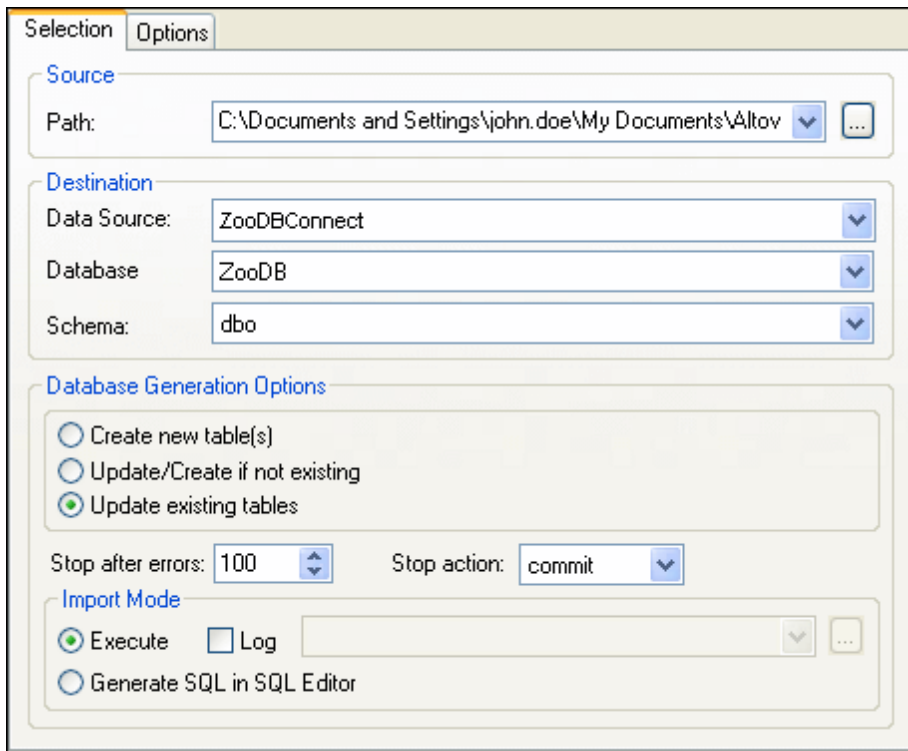
Chapter 13

Importing Database Data

13   Importing Database Data

Altova web site: database import/export

DatabaseSpy currently allows you to import data contained in XML or CSV files to your databases. When you click the **Import** button in the Tools toolbar or select the menu option **Tools | Import data to the database... (Ctrl+I)**, the **Import data to the database** dialog box opens where you can select the XML or CSV file to be imported and define the import options. The **Import data to the database** dialog box contains two tabs, Selection and Options.

The lower part of the dialog box always shows a preview of the selected data, which can be updated to reflect the selected options.
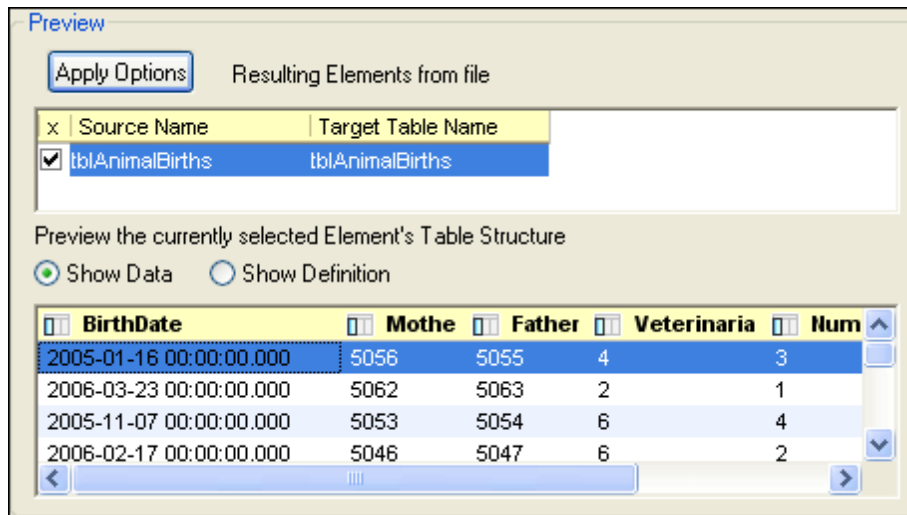
You can choose to preview the data (*screenshot above*) or the definition (*screenshot below*) of the selected element's table structure.
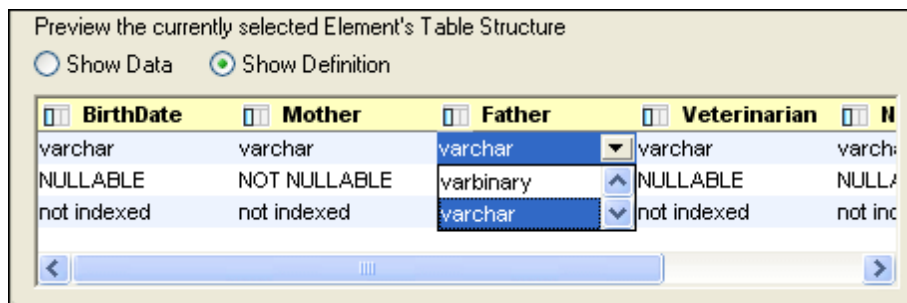
All fields will be created as Text fields with the length specified in the Options tab. You can change the data type separately for every column in the Preview group box if you select the `Show Definition` radio button (*see screenshot above*).

To import database data from a text file:

1. Make sure that a data source is active in the project folder.

2. Select the menu option **Tools** | **Import data to the database**, or press **Ctrl+I**, or click the **Import** button in the Tools toolbar.
   The **Import data to the database** dialog opens.

3. In the Source group box, enter the Path to the text file you want to import.
   You can type it in, select it from the drop-down list, or click the **Browse** button.

4. In the Destination group box, select where you want to import the data to.
   If you previously selected a data source, the three fields are automatically filled in, if not:

   1. Select the data source, database, and schema from the drop-down lists.

   2. Select `Create new table(s)`, `Update/Create if not existing`, or `Update existing tables`, depending on where you want to import the data.

   3. In the `Stop after errors` drop-down list, specify the number of errors after which the import process should be stopped.

   4. In the `Stop action` drop-down list, select the action to perform when import is

stopped: Commit commits the tables that did not contain errors; rollback does not commit any imported data.

5. In the Import Mode group box, do one or more of the following:

   - Activate the Execute radio button if the necessary SQL should be generated and executed in one step.

   - Activate the Generate SQL in SQL Editor radio button if the generated SQL file should be placed in the SQL Editor ready for execution.

   - Check the Log check box to activate the drop-down list where you can enter the name and path of the SQL log, select a previously used log file, or select an existing log file by clicking on the **Browse** button.

6. Click the **Apply Options** button to preview the data and further define the import settings.

7. Click the Show Data or Show Definition radio button to edit the different aspects of the table structure.

8. Double-click a field in the preview window to edit the data type, nullability, or indexing. Define how/whether you want to import each column by clicking the column header to change it.

   Import

   Import as unique key

   Import as primary key

   Do not import

9. Click **Import**.
   If the Execute radio button is selected in the Import Mode group box, the text file is imported into the database and a prompt appears asking if you want to view the log.sql file. You may have to execute the generated SQL script first to import the text file into the database.

13.1   Defining XML Import Options

The Options tab of the **Import data to the database** dialog box allows you to define the options when importing data from files to the database. When the XML icon on the left side of the dialog box is selected, you may change the import options for files of type XML.

Start point of import
You can choose to import the entire XML document or restrict your import to the data hierarchy starting from the currently selected element. The number of sub-levels below the start point that will be imported is specified in the Import Depth option.

Import depth
Specifies the number of sub-levels below the start point that will be imported.

Import fields
Depending on your XML data, you may want to import only elements, attributes, or the textual content of your elements. Note that you can deselect the import of individual elements in the Preview window.

Automatic fields
DatabaseSpy will produce one output file or table for each element type selected. You can choose to automatically create primary/foreign key pairs to link your data in the relational model, or define a primary key for each element.

Other options
The `Exclude namespace name` radio button, together with the `Replace Colon With Underscore` radio button, is an either/or choice. Specifies whether namespace prefixes of elements and attributes should be excluded or whether the colon in the namespace prefix should be replaced with an underscore.

Text strings in the XML document that should be treated as NULL values as well as the length of text fields in the database can be specified.

13.2   Defining CSV Import Options

When the CSV icon on the left side of the dialog box is selected, you may change the import options for files of type CSV on the Options tab of the **Import data to the database** dialog box.

Options

If the files you are importing contain column names, select the `First row contains column names` check box to exclude them from the import. In the `Treat this as NULL values` drop-down list, you can choose the character that should be recognized as a NULL value in the file being imported. The delimiter character and the quotes to be used to identify strings can also be selected from the respective drop-down lists. In the `Encoding` drop-down list, select the encoding of the files you are importing from the drop-down list. If a two- or four-byte encoding is selected as the default encoding (i.e., `UTF-16`, `UCS-2`, or `UCS-4`), you can choose between little-endian and big-endian byte ordering. A fixed length of the text fields being imported can be defined in the `Create text fields with length` field.

Chapter 14

Exporting Database Data

14   Exporting Database Data

Altova web site: database import/export

DatabaseSpy can export database data to a number of file formats such as XML, XML Structure, CSV, HTML, and Excel. When you click the **Export** button in the Tools toolbar or select the menu option **Tools | Export database data... (Ctrl+E)**, the **Export database data** dialog box opens, where you can select the data to be exported and define the export options for the individual export file types. The **Export database data** dialog box contains two tabs, Selection and Options, where the Selection tab is the same for all export types while each export file type has an Option tab of its own. You can change the export file type by clicking on the respective icon in the left area of the dialog box.

The Source group box basically is a depiction of the Online Browser window with all its features and options. In addition, you can select the individual tables for export here. Please note that the default layout in the Source group box for XML, CSV, HTML, and Excel is Folders while it is Table Dependencies for XML structure. You can, however, change the layout by clicking on the **Layouts** button in the Source group box.

Exporting structured XML

The structured XML export option exports table data in an hierarchical fashion.

```
<import>
        <parentTable>
                <name>…</name>
                …
                <childTable>
                        <childname>…
                        <grandchildTable>
                </childTable>
```

```
<?xml version="1.0" encoding="UTF-8"?>
<Import>
 <tblAnimalTypes>
  <AnimalTypeID>1</AnimalTypeID>
  <EnglishName>Parma wallaby</EnglishName>
  <LatinName>Macropus parma</LatinName>
  <AreaOfOrigin>Australia</AreaOfOrigin>
  <Category>1</Category>
 </tblAnimalTypes>
 <tblAnimalTypes>
  <AnimalTypeID>2</AnimalTypeID>
  <EnglishName>Freshwater Butterflyfish</EnglishName>
  <LatinName>Pantodon buchholtzi</LatinName>
  <AreaOfOrigin>Asia</AreaOfOrigin>
  <Category>2</Category>
 </tblAnimalTypes>
```


14.1   Selecting Database Data for Export

The Selection tab of the **Export database data** dialog box is the same for all export file types. The option that has been used last is active by default. If you want to export your database data to a different file format, select the respective icon on the left side of the dialog box and change to the Options tab in order to define the export options for XML, XML Structure, CSV, HTML, or Excel.
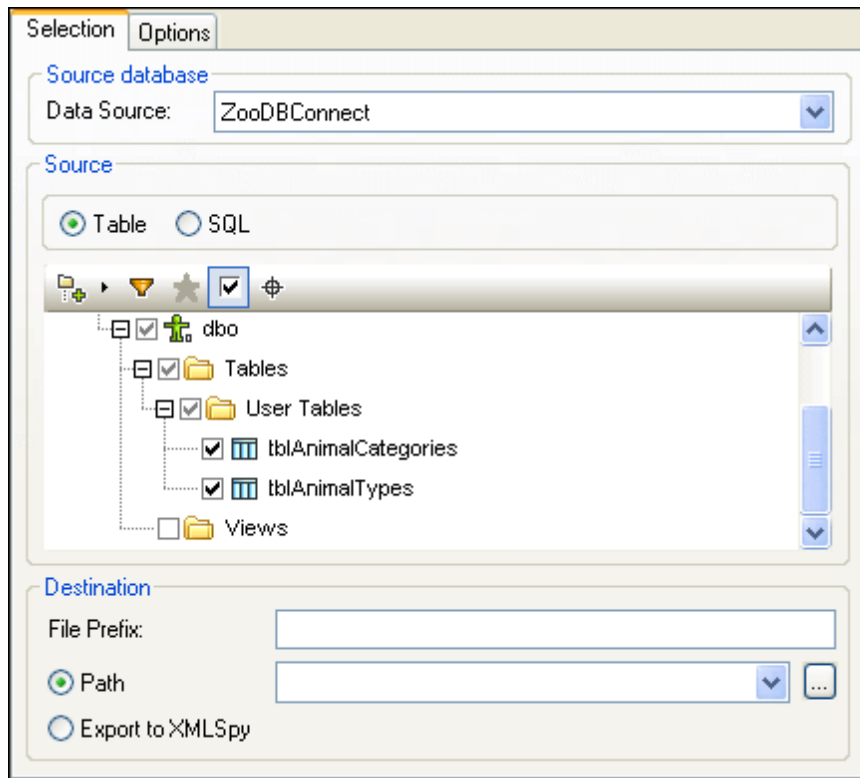
The Source group box of the **Export database data** dialog box provides a toolbar containing the same icons as the Online Browser (layout, filter, favorites, and Object Locator) as well as the

Show checked objects only [icon] icon. The latter is useful if you have a large number of tables in your database and want to export data from only a few of them. After you have activated the

check boxes of the appropriate tables, clicking the **Show checked objects only** [icon] icon displays only the tables that you have selected for export and hides the tables not concerned.

To export database data:

1. Make sure that a data source is active in the project tab.

2. Select the menu option **Tools | Export database data**, or press **Ctrl+E**, or click the

    Export [icon] button in the Tools toolbar.
    The **Export database data** dialog box opens.

3. Click the desired icon on the left side of the dialog.

4. In the Source database group box, select the data source from the drop-down list.

5. Select the data you want to export in the Source group box.
    Do one of the following:

- The `Table` radio button is selected by default. The data source you selected above is displayed as it appears in the Online Browser. The filterand favorites functions as well as the object locator are all available in the toolbar.

- Clicking the `SQL` radio button displays SQL commands or allows you to enter them directly.
  Please note: If the you have opened the **Export database data** dialog box from the toolbar of the SQL Editor window and not from the Tools toolbar, the `SQL` radio button is active by default and the contents of the SQL Editor window appear in the text box.

- If you select tables while the `Table` radio button is selected and then click the `SQL` radio button, the SELECT statements for those tables will be generated automatically and appear in this window.

```
1   SELECT [BirthID], [BirthDate], [Mother], [Father], [Veterinarian],
    [NumberInLitter] FROM [ZooDB].[dbo].[tblAnimalBirths];
2   SELECT [FeedTypeID], [FeedName], [AmountInStock], [Supplier],
    [AmountOrdered] FROM [ZooDB].[dbo].[tblAnimalFeed];
3   SELECT [AnimalTypeID], [EnglishName], [LatinName], [AreaOfOrigin],
    [Category] FROM [ZooDB].[dbo].[tblAnimalTypes];
4
```

6. If required, change the database you want to export the data from by clicking on the arrow to the right of the database name and selecting a database from the drop-down list.

7. Activate the check boxes of the tables you want to export from the database.

8. Specify the Destination to which you want to export the data. Do one of the following:

   - Optionally, enter a prefix that will be added to the table name when the table is exported.

   - Select `Path` and enter a path in the field or use the **Browse** ⊡ button.

   - Select `Export to XMLSpy`.

9. Select the table you want to preview and click the **Preview** button in the Preview group box.
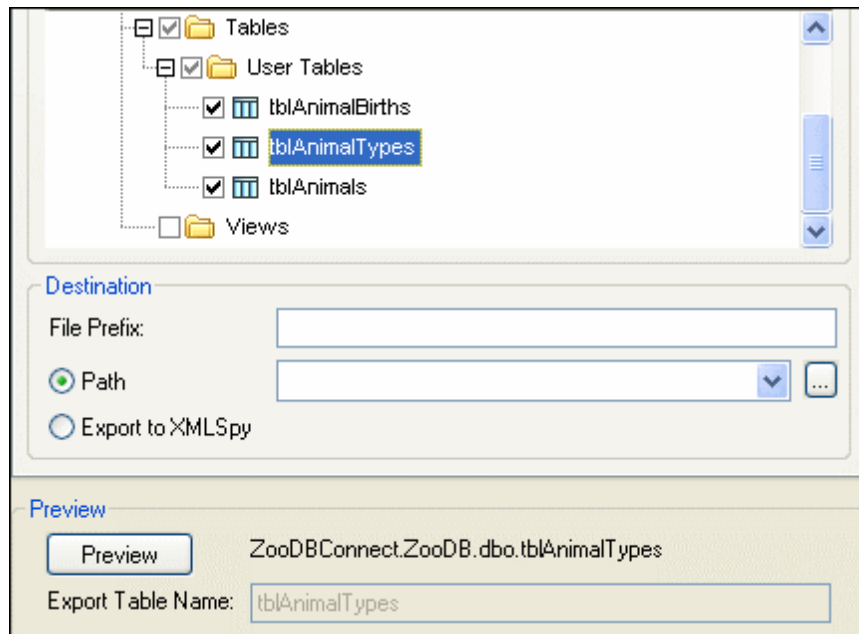
Please note that the **Preview** button is only activated if a table is selected.
The table data now appears in the Preview group box and the **Preview** button changes to **Reload**.
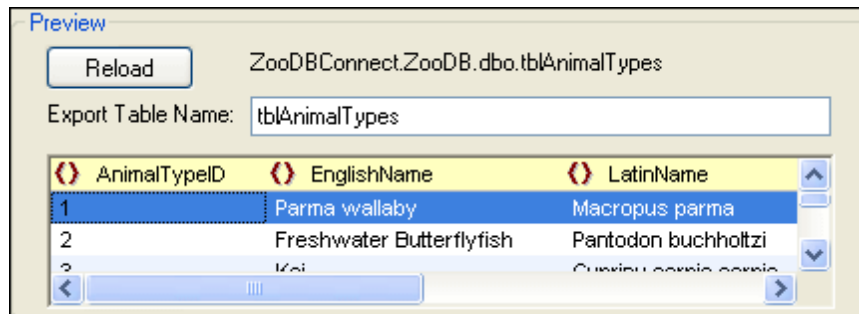
10. For each column, click the icon next to the column name to select whether you want it to be exported to the export file as an element [()], an attribute [=], or not appear at all [✖].

    In order to restore the original setting click the **Reload** button.

11. Change the file name of the selected table in the `Export Table Name` field, if necessary.

12. Click **Export**.
    Each of the selected tables is exported to a separate file. A message appears that informs you about the successful export of your database data.

Exporting data from designs

If you have a specific design open in a Design Editor window, you can easily export the data that is contained in the tables of this particular design. The Design Editor window provides an additional **Export** [🔁] button which is active if at least one table in the Design Editor window is selected. When you click this button, the **Export database data** dialog box opens with the `Table` radio button activated and the check box(es) of the selected table(s) already checked.

You can still add additional tables to the export definition.

Please note: Clicking the **Export** 🔘 button in the Tools toolbar will also open the **Export database data** dialog box. However, in this case no tables will be preselected for export.

To export data from a Design Editor window:

1. Make the Design Editor window whose tables you want to use for export the active window and select at least one table.

2. Press the **Export** 🔘 button in the Design Editor window.
   The **Export database data** dialog box opens. The tables that are currently selected in the Design Editor are already preselected in the Source group box.

3. Do one of the following:

   • Activate the `Path` radio button and enter the Path you want the destination files to appear in.

   • Activate the `Export to XMLSpy` radio button if you want your exported data to be displayed in XMLSpy.
     Note that this option is not available for export to Excel files.

4. In the Source group box, select a table and click the **Preview** button in the Preview group box.

5. Optionally, change the file name in the `Export Table Name` field.

6. If required, exclude columns from export or export them as attributes.

7. Optionally click the Options tab to change the options for XML, XML Structure, CSV, HTML, or Excel.

8. Click the **Export** button when all the settings are correct.
   A message box appears to inform you about the success of the export.

9. Click **OK**.
   The data is exported under the name specified in the `Export Table Name` field in the desired format and either stored under the path you specified or opened in XMLSpy for further editing.

14.2   Defining XML and XML Structure Export Options

In the Options tab for XML and XML Structure you can define the appearance of the exported files.
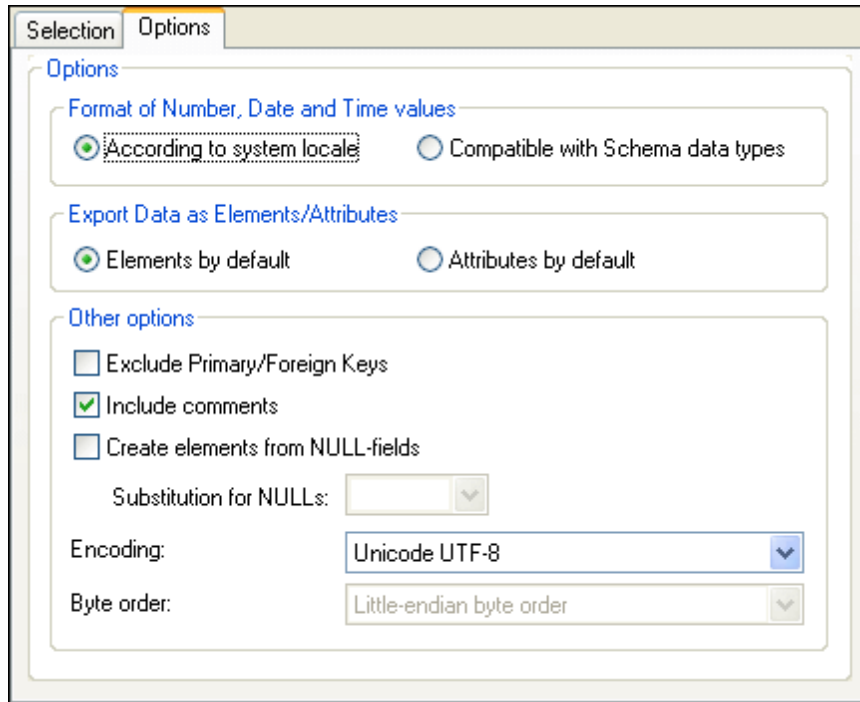
To define the extended options for XML and XML Structure export files:

1. Choose one of the following options for the format of number, date, and time values:

    • According to system locale
    • Compatible with Schema data types

2. Choose one of the following options for the default data type:

    • Select `Elements by default` to export the data as elements
    • Select `Attributes by default` to export the data as attributes

    Please note: You can change individual columns in tables to be exported as elements or attributes in the Preview group box of the dialog.

3. If you want to exclude primary and foreign keys from the export file, activate the `Exclude Primary/Foreign Keys` check box.

4. Activate the `Include comments` check box to include an attribute in the exported XML file that shows the SQL query used to select the data, as well as a list of comments containing one item for each column header in the database table.

5. Activate the `Create elements from NULL-fields` check box to include an empty element for each null value in the data being exported.

6. Optionally select from the `Substitution for NULLs` drop-down list or type in the string that should appear in the XML file when there is a NULL value in the data being exported.
    You can also type in a value that is not contained in the drop-down list. This option is only available if the `Create elements from NULL-fields` check box is activated.

7. Select the desired encoding for the files that are generated during export from the `Encoding` drop-down list.

8. If a two- or four-byte encoding is selected as the default encoding (i.e., `UTF-16`, `UCS-2`, or `UCS-4`), you can choose between little-endian and big-endian byte ordering.

14.3 Defining CSV Export Options

In the Options tab for CSV files, you can choose from among several options concerning the appearance of the CSV file that is created as a result of the export.
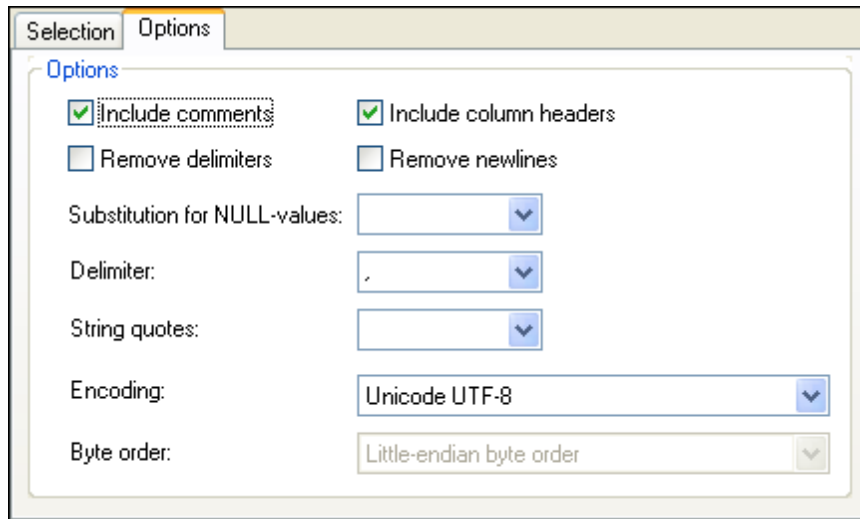
To define the extended options for CSV export files:

1. Activate the `Include comments` check box to include an attribute in the exported CSV file that shows the SQL query used to select the data, as well as a list of comments containing one item for each column header in the database table.

2. Activate the `Remove delimiters` check box to remove delimiters that are contained in text values in the exported data. Set the delimiter you want to remove by using the `Delimiter` drop-down list, the second drop-down list in this tab.
   For example, if this option is activated and you export a table containing the string "Ba'ker", the string will be "Baker" in the exported text.

3. If the exported tables is to include the database column names, activate the `Include column headers` check box.

4. Select the `Remove newlines` check box if you want to remove newlines from exported data.

5. Select from the `Substitution for NULL-values` drop-down list or type in the string that should appear in the CSV file when there is a NULL value in the data being exported. You can also type in a value that is not contained in the drop-down list.

6. Select from the `Delimiter` drop-down list the character you would like to use to delimit columns in the text file or enter the desired character string.

7. If you want to put each string into quotes, select the desired character from the `String quotes` drop-down list.
   You can also type in a character that is not contained in the drop-down list.

8. Select the desired encoding for the files that are generated during export from the `Encoding` drop-down list.

9. If a two- or four-byte encoding is selected as the default encoding (i.e., `UTF-16`, `UCS-2`, or `UCS-4`), you can choose between little-endian and big-endian byte ordering.

14.4 Defining HTML Export Options

The Options tab for HTML files provides some settings that apply to exported HTML files.
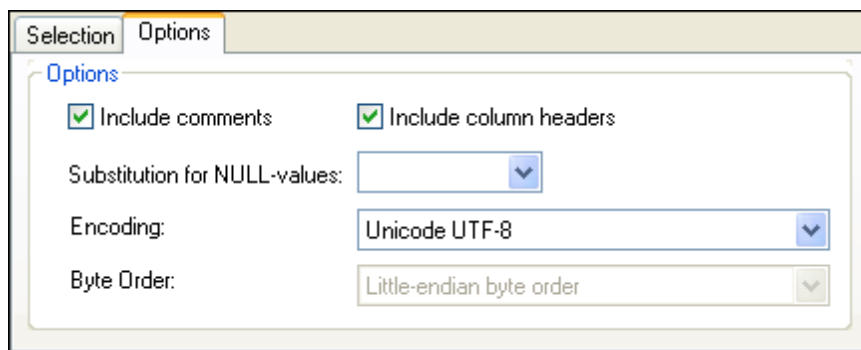
To define the extended options for HTML export files:

1. Activate the `Include comments` check box to include a comment in the HTML code of the exported HTML file that shows the SQL query used to select the data, as well as a list containing one item for each column header in the database table.

2. If the HTML table generated is to include the database column names, activate the `Include column headers` check box.

3. Select from the `Substitution for NULL-values` drop-down list or type in the string that should appear in the exported HTML file when there is a NULL value in the data being exported.
   You can also type in a value that is not contained in the drop-down list.

4. Select the desired encoding for the files that are generated during export from the `Encoding` drop-down list.

5. If a two- or four-byte encoding is selected as the default encoding (i.e., `UTF-16`, `UCS-2`, or `UCS-4`), you can choose between little-endian and big-endian byte ordering.

14.5   Defining Excel Export Options

For Excel export files, you can only define whether comments and column headers are to exported and specify a substitution for NULL values.
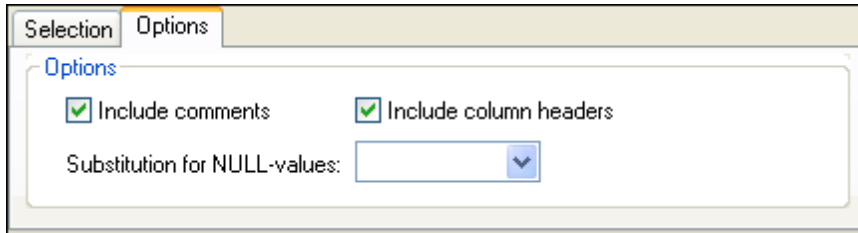
To define the extended options for Excel export files:

1. Activate the `Include comments` check box to include a comment in the Excel code of the exported Excel file that shows the SQL query used to select the data, as well as a list containing one item for each column header in the database table.

2. If the Excel table generated is to include the database column names, activate the `Include column headers` check box.

3. Select from the `Substitution for NULL-values` drop-down list or type in the string that should appear in the exported Excel file when there is a NULL value in the data being exported.
   You can also type in a value that is not contained in the drop-down list.

Chapter 15

DatabaseSpy Settings

15   DatabaseSpy Settings

The **Tools | Options** command enables you to define global application settings. These settings are specified in a dialog box with a navigation pane and saved in the registry. They apply to all current and future document windows. The **Apply** button in the **Options** dialog displays the changes in the currently open documents and fixes the current settings. The changes are seen immediately in the background windows. Each dialog page provides also a **Reset to Page Defaults** button which sets all changes you made on this page back to the default settings.

Each page of the **Options** dialog is described in detail in this section.

To change DatabaseSpy settings:

1. Select **Tools | Options...** or press **Ctrl+Alt+O**.
    The **Options** dialog box appears.

2. In the navigation pane, select the dialog pages for the following options:

    *   General
    *   SQL Editor
    *   Design Editor
    *   Data Compare

3. Make the desired changes and click **OK**.

15.1   General Options

The main page of the General options defines the general appearance of DatabaseSpy as well as general options when working with the application. Related settings are defined in the Online Browser, Change Script, File Types, and Encoding options.

Application Startup
You can choose to automatically open the project file that was open when you last closed DatabaseSpy or to have the **Add a Data Source** dialog box displayed when DatabaseSpy is started. If you do not choose any of the options in this group box, you must open files or projects in DatabaseSpy via the **File** menu.

Program Logo
Choose whether you want the DatabaseSpy logo displayed when the application is started and whether it should appear on top of a print-out you make.

Line Breaks
When you open a file, the character coding for line breaks in it is preserved if **Preserve old** is selected. Alternatively, you can choose to code line breaks in any of three codings: CR&LF (for PC), CR (for MacOS), or LF (for Unix).

Frame Title
The frame title in DatabaseSpy shows the project name and the name of the object that is currently active in the SQL Editor or the Design Editor, respectively. You can choose whether you want to see just the file name or the entire path of the object in the frame title of the application.

Automatic reload of files
If you are working in a multi-user environment, or if you are working on files that are dynamically generated on a server, you can watch for changes to files that are currently open in the interface. Each time DatabaseSpy detects a change in an open document, it will prompt you about whether you want to reload the changed file.

Connection Wizard

If you activate the **Skip Configuration Step in Connection Wizard** check box, the Configure page in the **Add a Data Source** dialog box is skipped and clicking **Next** in the first dialog page of the **Add a Data Source** dialog box will bring you directly to the **Data Link Properties** dialog box.

Newly saved files

You can automatically add files to your project when you save them to the file system or suppress the display of the respective dialog box.

15.1.1 Online Browser

The Online Browser section of the **Options** dialog box lets you configure layout and behaviour aspects of the Online Browser.
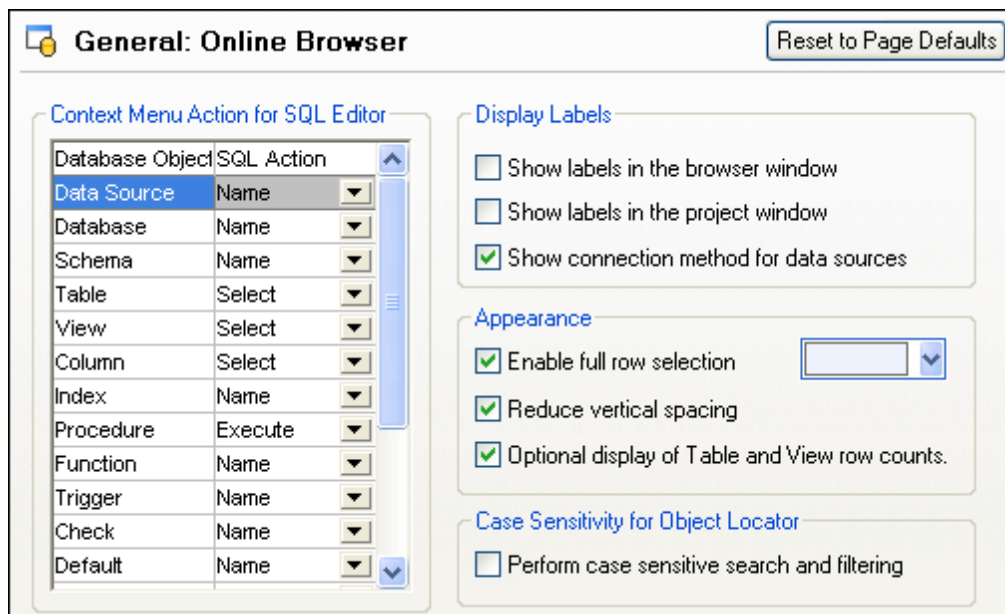
Context Menu Action for SQL Editor

Specify the default action taken when an object is dragged and dropped from the Online Browser to the SQL Editor. You can specify a different default action for each database object type.

Display Labels

You can have DatabaseSpy display labels that are displayed as a prefix in front of the names in the Online Browser or the Project window. For example, table names are prefixed with the word "Table" and columns with the word "Column" in the Online Browser, or data sources are prefixed with the word "Data Source" in the Project window.

To not show the connection method for data sources in the Online Browser and Project windows, deactivate the respective check box.

Show connection method   Hide connection method
in Project window: in Online Browser:
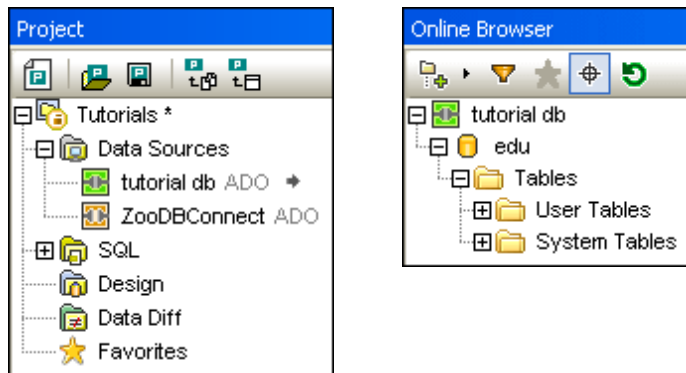
Please note that you can show or hide the connection method only for both the Project window and the Online Browser; this option cannot be set individually for the two of them.

Appearance

When the **Enable full row selection** check box is activated, in the Online Browser the row in which the cursor is currently positioned is highlighted and you can click to select the row without having to move the cursor exactly over the text of the row. From the drop-down list, you can choose a color to be used for selecting.

If you prefer a wider display of you database data, you can deactivate the **Reduce vertical spacing** check box. This option applies to both Project window and Online Browser.
The **Optional display of Table and View row counts** check box is activated by default, allowing you to show the number of data rows of a table or view without having to retrieve data first.

Case Sensitivity for Object Locator

When searching and filtering database objects, you can decide whether or not this should be case sensitive.

15.1.2 Change Script

The Change Script section of the **Options** dialog box lets you configure the script generation and execution of the Database Structure Change Script.
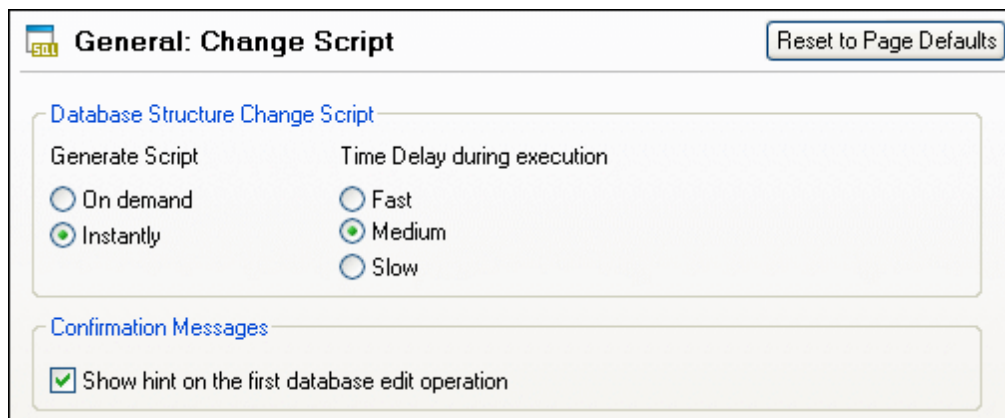
Database Structure Change Script

The Database Structure Change Script can be generated either instantly when a change occurs, or on demand. In the latter case, you must generate the change script manually by clicking on

the **Update Change Script** icon. In this group box, you can also define the time delay during execution of the change script. Choosing **Slow**, for example, increases the time delay and gives you the opportunity to cancel the execution of the change script at a certain stage.

Confirmation Messages

DatabaseSpy can display a popup alarming you that a change script has been generated when a database object is edited for the first time. You can turn off this message by deactivating the **Show hint on the first database edit operation** or by activating the **Don't show this dialog again!** check box in the dialog box.
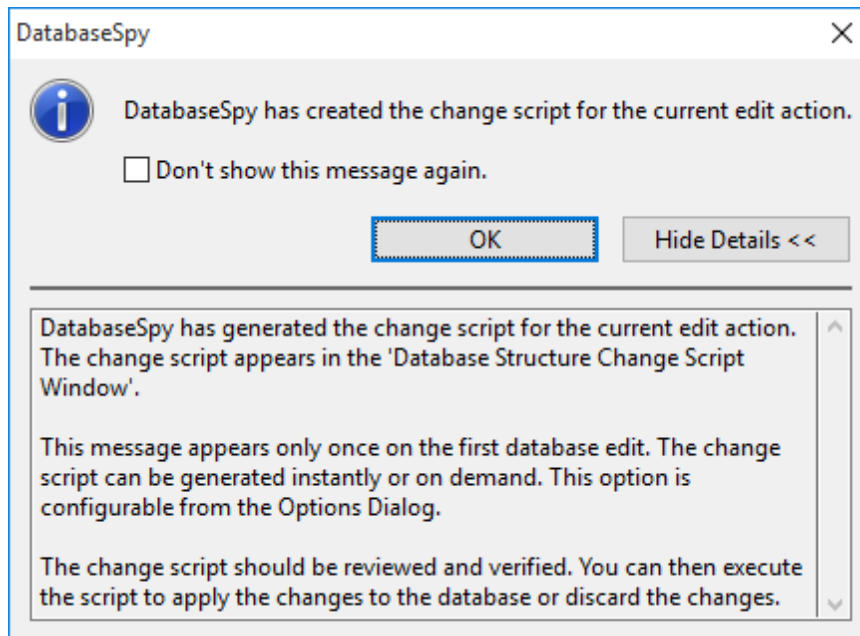
If you want DatabaseSpy to inform you when a change script is created, but do not want to have the details displayed, click the **Hide Details** button and the dialog will appear in a minimized form next time.

15.1.3 File Types

The File Types section of the **Options** dialog box allows you to customize the behavior of DatabaseSpy on a per-file-type basis.
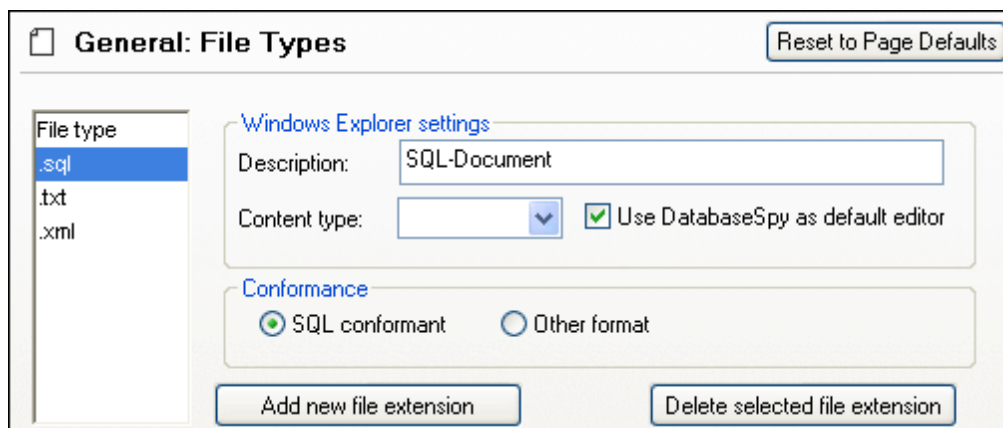
Windows Explorer settings

You can define the file type description and MIME-compliant content type used by Windows Explorer and whether DatabaseSpy is to be the default editor for documents of this file type.

Conformance

DatabaseSpy provides specific editing and other features for various file types. The features for a file type are set by specifying the conformance in this option. DatabaseSpy lets you set file type to conform with SQL and other (text) grammars. A large number of file types is defined with a default conformance that is appropriate for the file type. We recommend that you do not modify these settings unless you are adding a new file type or deliberately wish to set a file type to another kind of conformance.

Add new file extension

Adds a new file type to the File types list. You must then define the settings for this new file type using the other options in this section.

Delete selected file extension

Deletes the currently selected file type and all its associated settings.

15.1.4 Encoding

In the Encoding section of the **Options** dialog box, you can specify several options for file encodings.
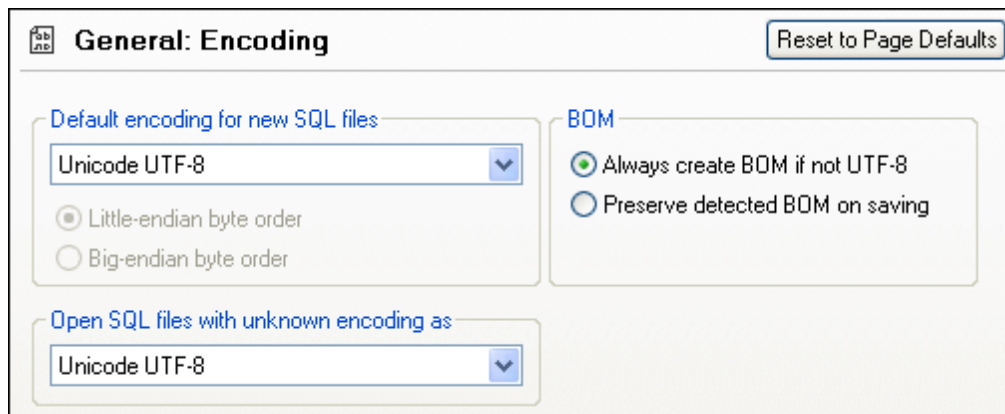
Default encoding for new SQL files

Define the default encoding for new files so that each new document includes the encoding-specification that you specify here. If a two- or four-byte encoding is selected as the default encoding (i.e., UTF-16, UCS-2, or UCS-4), you can also choose between little-endian and big-endian byte ordering for the SQL files.
The encoding for existing files will, of course, always be retained.

Open SQL files with unknown encoding as

You can select the encoding with which to open an SQL file with no encoding specification or where the encoding cannot be detected.
Please note: SQL files which have no encoding specification are correctly saved with a UTF-8 encoding.

BOM (Byte Order Mark)

When a document with two-byte or four-byte character encoding is saved, the document can be

saved either with (i) little-endian byte-ordering and a little-endian BOM (**Always create BOM if not UTF-8**); or (ii) the detected byte-ordering and the detected BOM (**Preserve detected BOM on saving**).

15.2   SQL Editor Options

The main page of the SQL Editor options defines the visual appearance of the editor. Additional SQL Editor-related settings are defined in the SQL Generation, SQL Formatting, Autoinsertion, Result View, and Fonts, options.

General

Syntax coloring emphasizes different elements of SQL syntax using different colors.
By activating the **Connect data source on execute** check box, you can have DatabaseSpy connect to the corresponding data source automatically whenever an SQL file is executed and its data source is not connected.

Retrieval

You can have DatabaseSpy display a dialog box asking you whether you want to change the time out settings when the permissible execution period is exceeded.
Specify the maximum amount of time that is permissible for the execution of retrievals.
The number n you enter into the **Buffered Amount (Rows)** field will appear in the context menu of the Online Browser if you select the **Retrieve data | First n rows** option.

Entry Helpers

To enable auto-completion suggestions as you start typing in SQL Editor, select the **Automatically open** check box.

The entry helper buffer for auto-completion and auto-insertion can be filled either when you connect to a data source or when it is used for the first time. Note that filling the buffer may take some time. Use the **Clear Buffer** button to reset the buffer.

Text View Settings

Text view settings are defined in a separate dialog box, which can be accessed via the **Text View Settings** button. See SQL Editor Features.

15.2.1 Generation

The Generation section of the **Options** dialog box lets you configure how the statements are generated.

Statement generation options

Use the **Select a database** field list to select a database kind and set the statement generation options individually for the different database kinds you are working with. Activating the Apply **to all databases** check box sets the options that are currently selected for all databases.

When the **Append semicolons to statement end** check box is activated, a semicolon is appended when you generate an SQL statement in the SQL Editor. Note that editing of data in Oracle databases and IBM iSeries and DB2 databases via a JDBC connection is possible only if this check box is unchecked.

By default, identifiers in SQL statements are enclosed in database-specific escape characters. You can deactivate the **Surround identifiers with escape characters** check box if you do not want to display these characters.

The **Generate SELECT statements with full column list** check box is also activated by default. If you prefer SELECT * FROM ... when retrieving all columns of a table, deactivate it.

Confirmation Messages

When you have deactivated the **Append semicolons to statement end** check box and activated the **Show warning when editor enforces semicolons and SQL generation does not** check box, DatabaseSpy will display a message box when a statement is generated (see Generating SQL Statements).
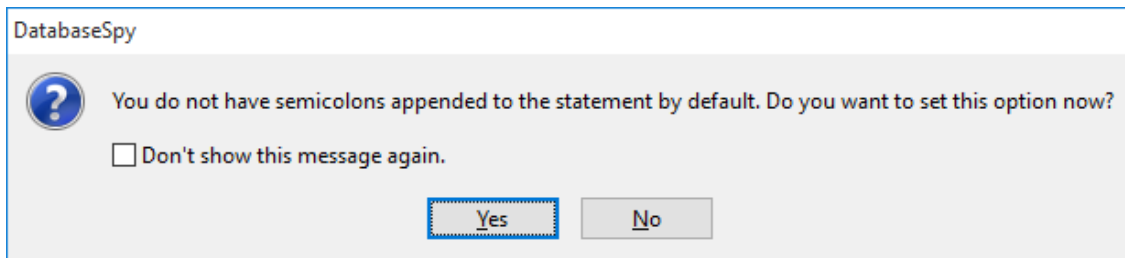
15.2.2 Formatting

The Formatting section of the **Options** dialog box lets you switch off SQL formatting for the SQL
Editor and Change Script windows and configure the behavior of open SQL Editor windows.
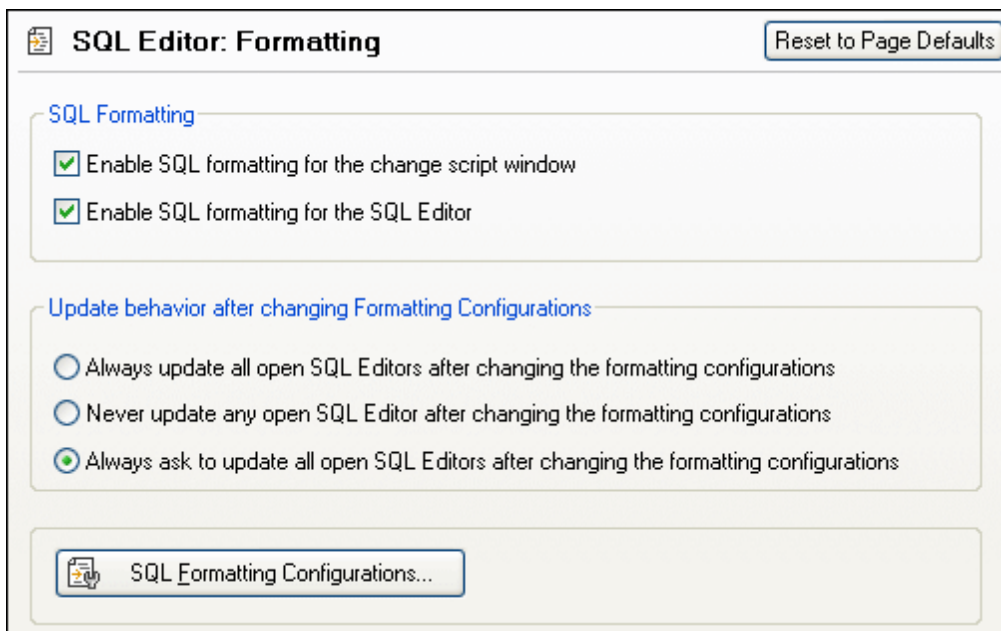
SQL Formatting
You can disable the SQL formatting options for the Change Script window and/or the SQL Editor
by deactivating the respective check boxes. Note that in the Change Script window, SQL
formatting is applied automatically when the next change is recorded, and that you cannot apply
formatting without changing the database structure.

Update behavior after changing Formatting Configurations
This group box allows you to specify whether or not open SQL Editor windows should be
automatically updated after formatting configurations have changed or if a prompt should be
displayed, where you can choose to update the windows or to keep the existing formatting.

SQL Formatting Configurations
The **SQL Formatting Configurations...** button opens the **SQL Formatting Configuration**
dialog box, where you can <u>define the configurations</u> for the different database types or disable
formatting on database or even statement type level.

15.2.3 Autoinsertion

The Autoinsertion section of the **Options** dialog box lets you configure the options that apply for text that is inserted automatically into the SQL Editor.

Closing Characters

You can disable the autoinsertion of closing characters completely by deactivating the **Autoinsert closing characters** check box. If you only want to exclude some of the characters from being inserted automatically, activate the **Autoinsert closing characters** check box and deactivate the corresponding check box for the respective character. If you insert an opening character for which autoinsertion is enabled, the corresponding closing character is automatically inserted and the cursor is placed between the two characters.

When inserting comment marks, please note that you have to highlight a text section before entering a "/" or "-" in order to comment out this text. DatabaseSpy will then automatically insert a "*" after the "/" in front of the text as well as "*/" after the text if you enter a "/", or add a second "-" if you enter a "-" while the text to be commented out is selected.

Autoindent

The **Match previous line** option helps you preserve text indentation after you press the **Enter** key in SQL Editor. For example, if line 1 is indented by a tab character, you may want the next line to have the same indentation as line 1 after you press **Enter**. To enable this behaviour, select the **Match previous line** check box.

Behaviour

When the corresponding option is activated, the closing character is deleted automatically when the opening character is deleted in the SQL Editor.

To place selected text inside a pair of autoinserted characters, activate the **Wrap selected text when autoinserting** check box and then type the opening character. The opening and closing characters are inserted automatically before and after the text.

Star Expansion

DatabaseSpy can automatically replace the "*" in a `SELECT * FROM ...` statement with all of the columns of the respective table if the **Expand columns with tab** option is checked. Please note that the statement has to be valid and that you must place the cursor behind the "*" in the statement and then press the **Tab** key to achieve this.

15.2.4 Result View

The Result View section of the **Options** dialog box lets you configure aspects of the appearance of the Result window in the SQL Editor.
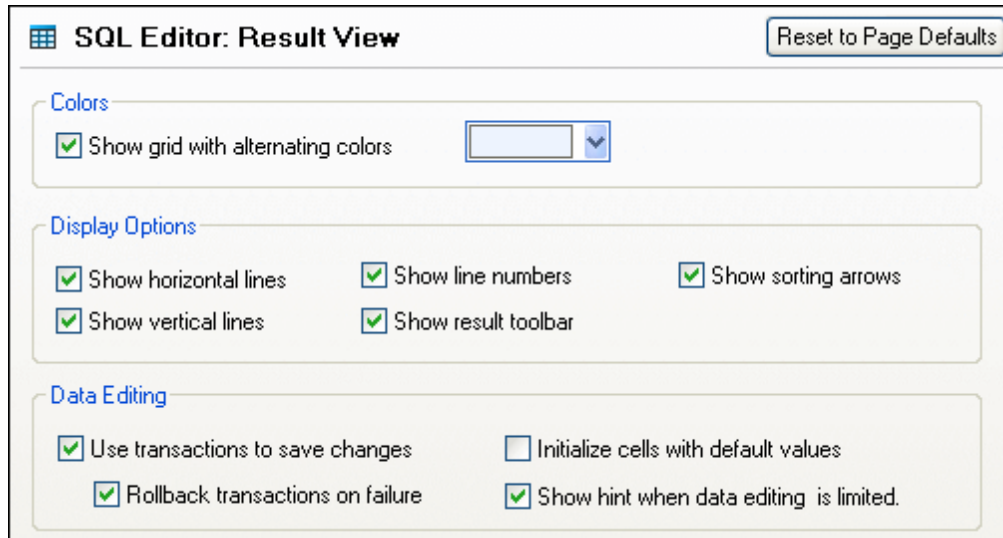
Colors

You can have rows in Result tabs displayed as simple grid or with alternating white and colored rows. The color can be changed using the color chooser below the check box.

Display Options

By default, horizontal and vertical grid lines as well as line numbers and the **Result** toolbar are displayed. You can switch any of these options off by deactivating the respective check box.

DatabaseSpy allows you to sort data contained in the result grid either ascending or descending by repeatedly clicking the column headers. To enable this option, activate the **Show sorting arrows** check box.

Data Editing

When the default setting is applied, DatabaseSpy uses transactions to save changes in the result grid and rolls back these changes if a failure occurs during committing the changes to the database. Any editing is then written into a transaction which is executed as a whole. If the **Rollback transactions on failure** check box is activated, all changes contained are undone upon failure of a single update statement in the transaction.
Please note that transactions are not supported by MS Access databases

If default values are defined for the individual database fields, DatabaseSpy can automatically insert these values when creating a new line in the result grid. Deactivate the **Initialize cells with default values** check box if you want to create empty lines.

When data editing is not possible to the full extent, DatabaseSpy shows a warning message

before the data is displayed in the Result window. You can suppress this hint by deactivating the
Show hint when data editing is limited check box.

15.2.5   Fonts

The Fonts section of the **Options** dialog box lets you configure color and font settings of different
parts of SQL statements. Use the drop-down list below the heading of the dialog page to switch
between Text Editor and Result View settings.

Text Editor Font
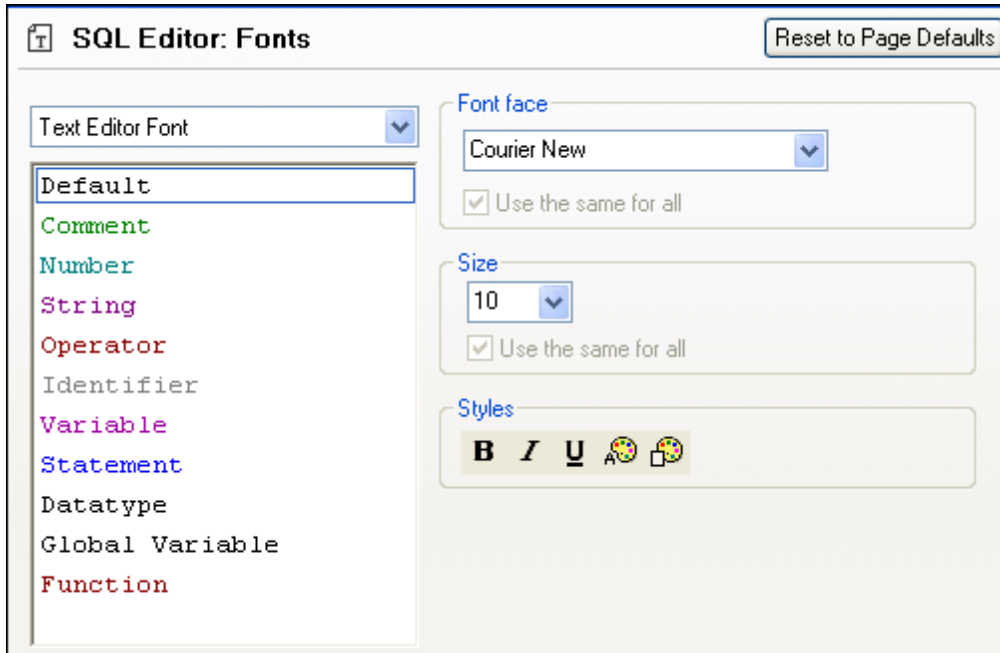
The types listed in the left hand pane are elements of SQL statements. You can choose the
common font face, style, and size of all text that appears in SQL Editor. Note that the same font
and size is used for all text types. Only the style can be changed for individual text types. This
enables the syntax coloring feature.
Click the **Reset to Page Defaults** button to restore the original settings.

Result View Font
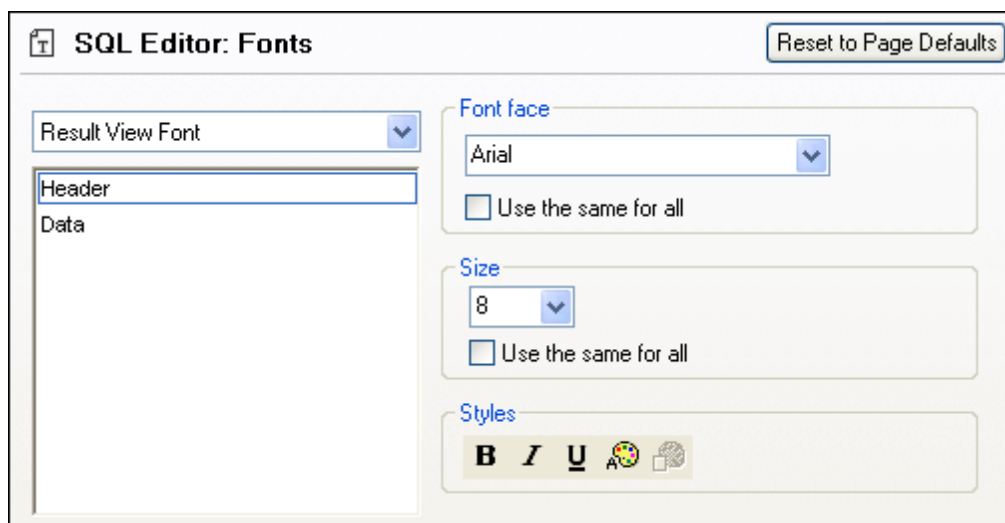
In the left hand pane, you can choose either the result grid header or the data rows. You can choose the common font face, style, and size of all text that appears in the result grid. Note that you can apply the same font and/or size to all text types by activating the respective **Use the same for all** check box.
Click the **Reset to Page Defaults** button to restore the original settings.

15.3   Design Editor Options

The main page of the Design Editor options defines the visual appearance and general behaviour of the editor. Additional Design Editor-related settings are defined in the Element Colors and Fonts options.

General Behavior

If you have several data sources defined in your project but not all are active, you can automatically connect to the related data source when you open a design file in Design Editor. A prompt asking you if you want to connect will appear if this option is not activated.
DatabaseSpy can display tooltips when database objects are dragged from the Online Browser or from different table designs into the Columns or Keys section of a table design.
When creating new database objects, you can automatically display the properties box in the Design Editor so as to fine-tune the definition of the objects without having to change to the properties window. For this purpose, activate the **Automatically expand newly created objects** check box.

Tables

When new tables are created in the Design Editor, DatabaseSpy can automatically create primary keys for them. The first column of a new table is then renamed to "id" and a primary key constraint is added to the Keys section.
You can have the schema name displayed in parentheses in the title bar of a table.

Relations

By default, all connection lines between related tables are labeled in Design Editor. You can, however, hide the labels of unselected lines by deactivating the respective check box. Highlighting active relations objects will gray out tables that do not form part of a relation when you click on the line that connects related tables.

Default options on insertion

The default table layout can be set to compact and/or expanded. In the compact layout only the

column names of a table are visible (check the **Insert with compact layout (single column)**
check box to switch to compact layout). If the **Expand table object when inserted** check box
is not checked, only the title bar of a table will be displayed.

Grid Settings

Adjust the grid size according to your needs and specify whether or not database objects in the
Design Editor should snap to grid.

15.3.1   Element Colors

The Colors section of the **Options** dialog box lets you configure colors for objects in the Design
Editor.
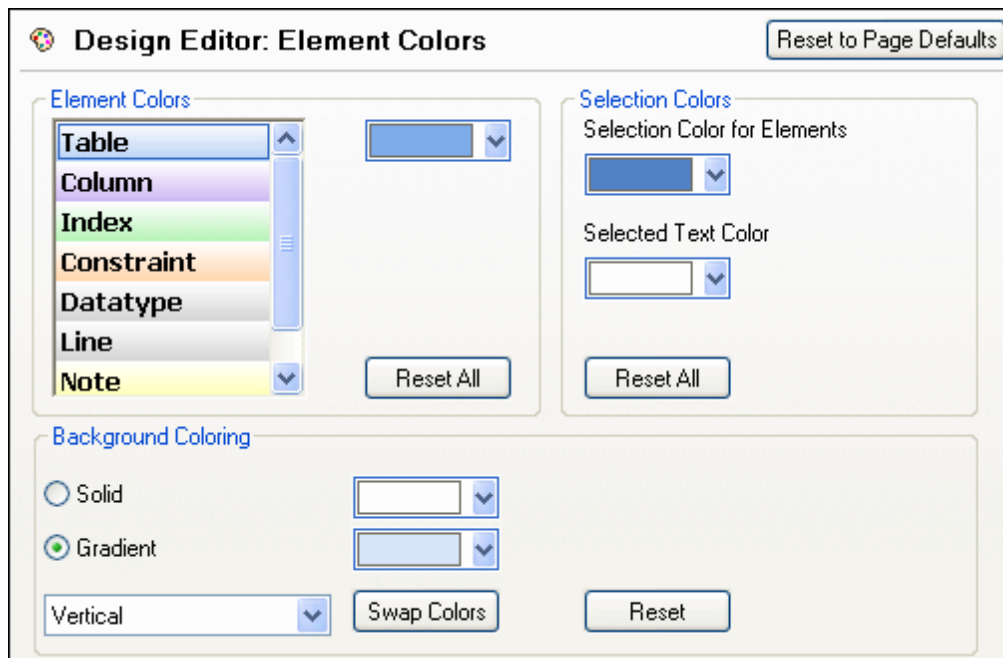
Element Colors

Specify the color that is used for displaying each element in the Properties windows. Select an
element from the list and choose a color for it from the drop-down list. Click the **Reset All** button
to restore the original settings.

Selection Colors

Define the colors for selected elements and text. Click the **Reset All** button to restore the original
settings.

Background Coloring

Defines the appearance of Design Editor's background. Choose between solid and gradient
background and specify the colors and the way they are used in the gradient.

15.3.2   Fonts

The Fonts section of the **Options** dialog box lets you configure fonts for objects in the Design
Editor.
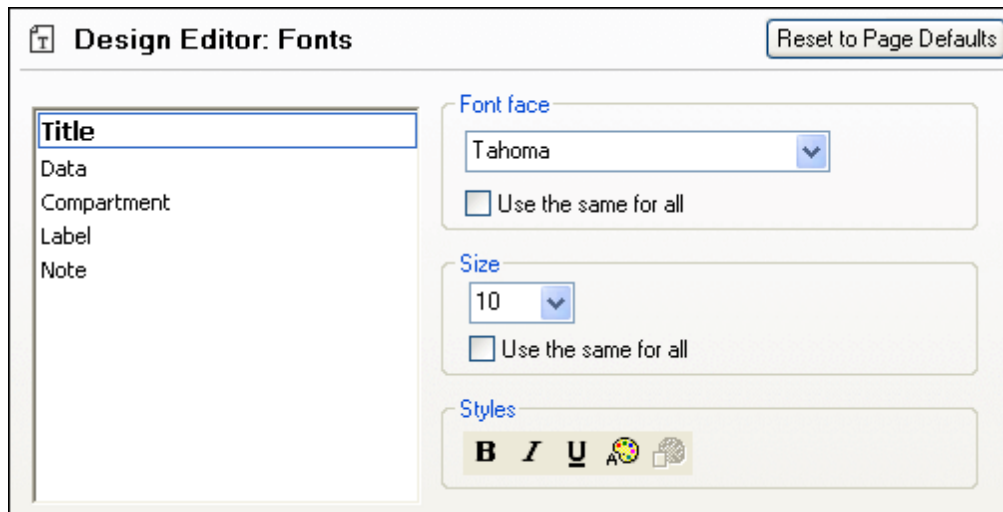
Font settings

You can choose the font face, size, and style for the individual elements displayed in the Design Editor. Note that, unlike in the Text Font section, it is possible to specify different font faces for header and data rows. To apply the same font and/or size to all text types, activate the respective **Use the same for all** check box.

The element names in the list box are defined as follows:

- **Title**: The name of a table in its title bar
- **Data**: The content of the table, i.e., the column names
- **Compartment**: The sub-titles in a table, i.e., Column Name, Indexes, Keys, Check Constraints
- **Label**: The designation of the connection line which links related tables in the Design Editor
- **Note**: The content of sticky notes that you add to the design view

15.4 Data Compare Options

The main page of the Data Compare Options defines how the compared tables in data and schema comparisons are displayed in the comparison components and whether tables and/or columns should be mapped automatically when added to a comparison. Additional settings related to database data comparison are defined on the Comparison Options, XML Compare, and Result View pages.

Display Options

In order to be able to open a comparison window without having to specify data source and tables to be compared, deactivate the Show table selection for new documents check box. If you prefer that all tables be shown with the columns not expanded, deactivate the **Show tables initially expanded** check box. You can deactivate the display of the schema name for tables in components, and show the data source name in the component header.

Automatic table and column mapping

When tables are added to a comparison, they are automatically mapped. You can deactivate the automatic mapping of columns and/or tables and choose how (i.e., by name, by type, by name and type, or based on ordinal position) they should be mapped. In addition, you can specify whether or not binary data columns, XML columns, identity columns, calculated columns, or the case of a column name should be ignored. If you want to map item names that match to a large extent but are not exactly equal (e.g. "expenses2008" and "expenses_09"), deactivate the **Exact match when mapping by name** check box.

Schema comparison

If you activate the **Try to map unconnected items during comparison** check box, DatabaseSpy will try to find corresponding items for unmapped items in the opposite component when a comparison is started.

15.4.1 Comparison Options

The Comparison Options section of the **Options** dialog box displays the options that are used for string-based comparison.

Comparison Mode

You can choose from among two comparison modes: The native type considers the data type of the column that is compared, whereas everything will be converted into strings before comparison if you use the string representation type.

Comparison Options

Deactivate the **Ignore case in comparison** check box if upper and lower casing should not be ignored when checking for differences.
Check the **Ignore whitespaces in comparison** option if you do not want to consider whitespaces when comparing database data. Whitespace characters are space, tab, carriage return, and line feed.
If you do not want to make a difference between an empty field and a field containing the NULL value, activate the **Treat [NULL] as empty string** check box.

Numeric Options

Floating point numbers can be filled up with trailing zeros by activating the **Append trailing zeros to floating point numbers** check box.

Optimization Options

If you do not care to see rows that are equal in both tables of a comparison, you can check the **Keep only different rows in memory** option. In this case, only different rows are retrieved and displayed in the Result window.

Execution Options

You can define whether you want to process comparison of multiple tables sequentially or parallel. Changes can be committed with or without the use of transactions and you can decide whether or not you want to rollback on errors.

15.4.2 XML Compare

The XML section of the **Options** dialog box displays the options that are used for XML-based comparison.

Whitespace

Whitespace characters are space, tab, carriage return, and line feed. The three options here compare files with whitespace unchanged; with whitespace normalized (i.e., all consecutive whitespace characters are reduced to one whitespace character); and with all whitespace stripped (i.e., not considered for comparison).

Case sensitivity

If the **Ignore case** check box is checked, then case is ignored, and you have the option of ignoring or not ignoring case in node names.

Namespace/Prefix

These are options for ignoring namespaces and prefixes when searching for differences.

Ignore node types

Check the node types that will **not** be compared in the Compare session. Node types that may be ignored are Attributes, CDATA, Comments, Processing Instructions, DOCTYPE statements, and XML declarations.

Order

If **Ignore order of child nodes** is selected, then the relative position of the child nodes of an element is ignored. As long as an element node with the same name exists in each of the two sets of sibling nodes, the two sets are considered to be equal. Note, however, that if an element node has an attribute, it will always be considered unequal to an element with the same name in the compared sibling set—even if the **Ignore order of child nodes** is selected. The option of ignoring the order of attributes is also available, and applies to the order of attributes of a single element.

Entities

If **Resolve entities** is selected, then all entities in the document are resolved. Otherwise the files are compared with the entities as is.

Text
If **Ignore text** is selected, then differences in corresponding text nodes are not reported.

Depth
If **Ignore node depth** is selected, then the additional depth of any element (i.e., more levels of descendants) relative to the depth of the corresponding element in the compared file is ignored. This option must be unselected to enable merging and exporting differences.

15.4.3   Result View

The Compare Result View section of the **Options** dialog box provides options for customizing the display of the comparison results.

Display Options
Activate the relevant check boxes if you want to display horizontal or vertical lines, or line numbers in the Compare Result View. You can also adapt the default gutter width if you want to separate the compared columns from each other more clearly.

Advanced Selection Options
When checked, the **Don't auto-select relevant cells** check box allows you to select a single cell containing a difference in the result grid without automatically selecting the corresponding cell in the compared table. Leaving the check box unchecked (default setting) the two different cells are always selected as a pair so as to show you exactly which cells will be affected by a merge.

Chapter 16

User Reference

16   User Reference

16.1   Tables

All the information stored in a database is contained in tables. In DatabaseSpy, you have two options for creating a new table: you can (i) copy an existing table with few mouse clicks by generating and executing the required SQL statements in the SQL Editor, or (ii) use the Design Editor to create a table from scratch or duplicate columns of an existing table.
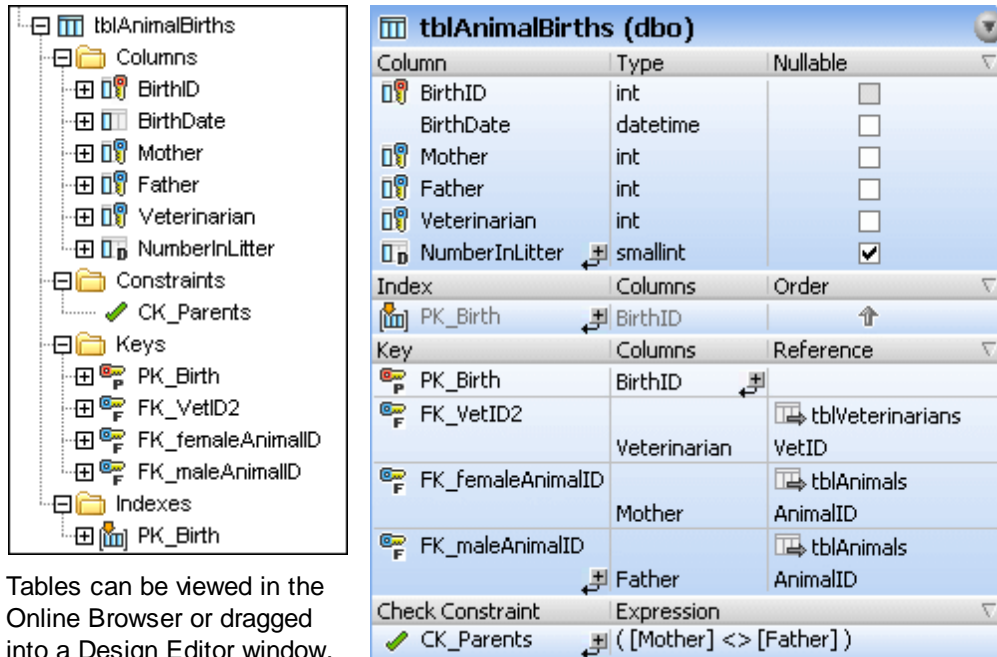
Tables can be viewed in the Online Browser or dragged into a Design Editor window.

16.1.1   Creating Tables

Once you are connected to a data source, you can create a new table in DatabaseSpy in several ways.

- Open an SQL Editor window, enter the required SQL statement (*see example for SQL Server below*), and click the **Execute** button in the SQL Editor toolbar.

```
CREATE TABLE newTable (
      id INT NOT NULL,
      CONSTRAINT PK_newKey PRIMARY KEY (id)
)
```

   You must define at least one column for the new table. Additional columns can be included immediately when the table is created, or added later using drag and drop in the SQL Editor or the Design Editor.

- Open a Design Editor window, click the **New Table** button in the Design Editor toolbar (or select the menu option **Design Editor | Create New Table**), edit the default properties as required, and click the **Execute Change Script** button in the Database Structure Change Script window.
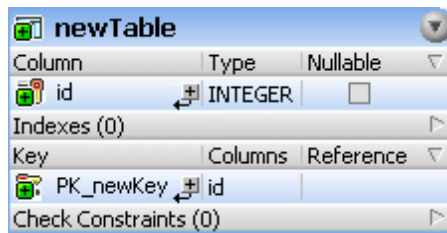
The above screenshot depicts the same example in a DB2 database. The Database Structure Change Script window displays the corresponding change script if the currently used schema in the database is "USER".
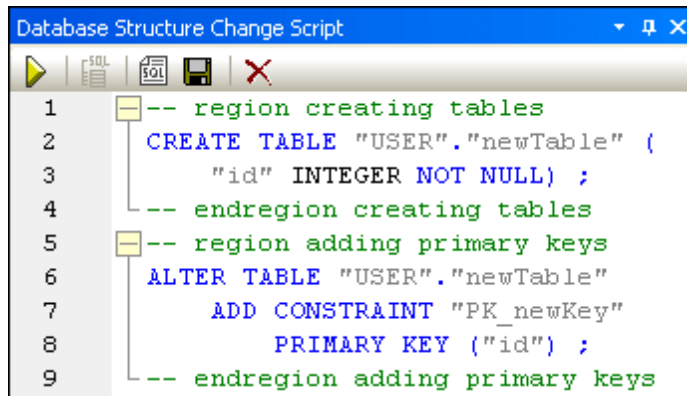
After you have executed either the SQL statement you have entered in the SQL Editor or the change script that is automatically generated when adding a table in the Design editor, you can conveniently add additional columns, keys, indexes, and constraints in the Design Editor, duplicate existing columns from other tables, or create foreign key relationships by dragging a key or index from the Online Browser into a table design.

16.1.2 Duplicating Tables

If you want to re-use the definition of a table that is already present in DatabaseSpy, all you have to do is to display the corresponding CREATE statement in an SQL Editor window, edit it as required, and execute the script. This way, you can even duplicate tables from different databases, provided that you edit the SQL statements so as to comply with the syntax of the target database.

Alternatively, you can also use DatabaseSpy's table structure converting feature.

To duplicate a table definition:

1. Right-click a table in the Online Browser and select **Show in new SQL Editor | Create** from the context menu.

2. Rename the table in the SQL Editor if you want to add the duplicate to the same database.

3. Optionally, edit the column and key properties.

4. If you want to add the table to a different database, change the execution target in the execution target bar.

5. Optionally, edit the SQL syntax to comply with the target database.

6. Click the **Execute** button in the SQL Editor toolbar.

16.1.3 Modifying Column Properties

When a column is selected either in the Online Browser or in a Design Editor window, its properties are displayed in DatabaseSpy's Properties window. In addition, you can show some of the properties (data type, nullability, default value, description, identity, increment, and seed) directly in the Design Editor.

Data type
When you need to change the data type of a column, you can edit the `Type` field directly in the table design. DatabaseSpy assists you by displaying incomplete data type entries in gray and changing the color to black as soon as you have typed in a valid keyword. While you are typing, a drop-down list containing the available data types is displayed. Incorrect syntax is marked in red. For data types requiring a length, you can also type in "max" and thus set the length to the maximum allowed value.

Nullability
The `Nullable` check box can also be changed directly in the table design. Please note that NULL values are not permitted if a primary key is defined on that particular column and that you cannot deactivate the `Nullable` check box if NULL values are already present in the column.

Data length
Every data type has a default length defined that is automatically used when you select that particular data type. For some data types, you can increase or decrease this default length in the Properties window. Please consider that existing values may be truncated if you decrease the data length in an existing table.

Precision
In numeric columns, you can define the maximum number of digits to be used by a data type.

Scale
The Scale property defines the maximum number of digits to the right of the decimal point in numeric and decimal columns.

Description
You can add an individual description for each of the columns.

16.1.4 Designing Tables

Once you have created a table in DatabaseSpy, you have several options for designing the table. You can always use the SQL Editor and write and execute the respective SQL statements to change the design of a table, however, DatabaseSpy's Design Editor provides a series of more convenient options for adding columns, primary, unique, and foreign keys, indexes, as well as check and default constraints, or defining identifier columns.

- Drag and Drop
- Context menu options
- **Create new Object** button
- Menu commands
- Keyboard shortcuts

Drag and Drop

In the Design Editor, you can use drag and drop in two ways:

- Select a table, column, or primary key in the Online Browser and drop it onto a table

design of another table.

- Select a table in the Online Browser and drag it into the Columns section of the table design of a different table to create duplicates of all columns in the target table. Note that only the columns of the source table will be duplicated; to duplicate the table in its entirety, use the context menu in the Online Browser. You can also select one or more individual columns in the Online Browser or a table design in the Design Editor and drag the column(s) into a (different) table design.

Context menu options

Depending on where the cursor is located in a table design when the right mouse button is pressed, two different context menus are provided for creating database objects.
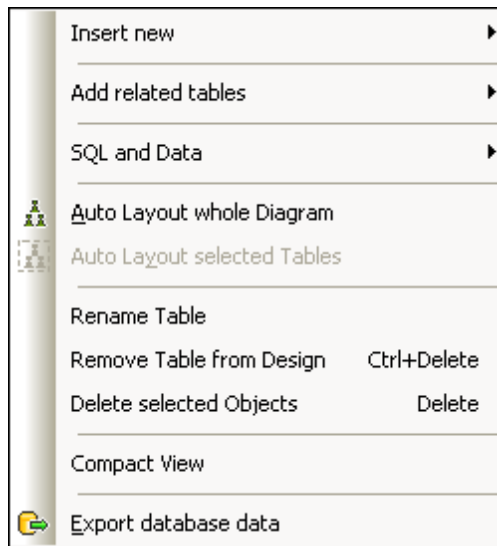
Right-clicking the table header or into any of the Keys, Indexes, or Constraints sections will pop up the context menu depicted above.
If you right-click a column in the Columns section of a table design, the right context menu will be displayed and you can directly define an index, keys, or a default constraint on the selected column.
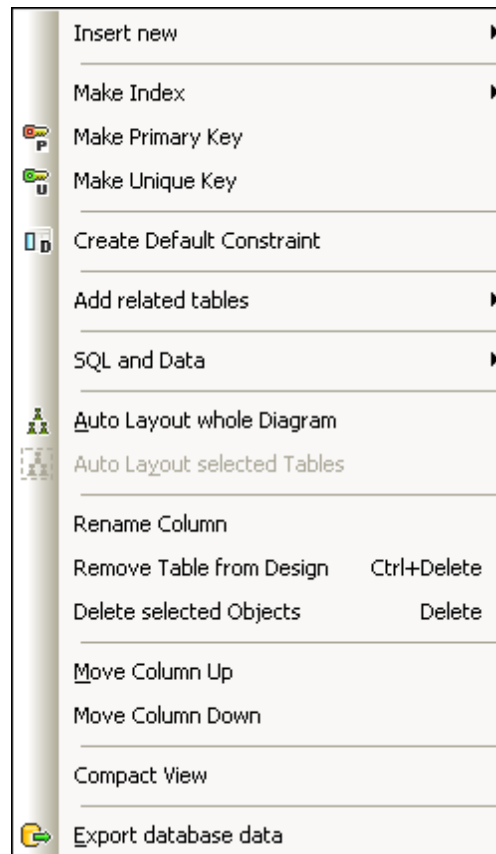
Create new Object button

Each section in a table design provides a **Create new Object** 🔳 button for adding new database objects to a table. Clicking this button will immediately add a new column or check constraint, or pop up a menu where you can choose the type of key or index you want to define.

Menu commands

The command for adding new columns is also available in the **Design Editor** menu. Select the menu option **Design Editor | Create New Column** to add a new column to the active table design in the Design Editor.

Keyboard shortcuts

DatabaseSpy provides also the keyboard shortcut **Alt+C** for adding new columns in the Design Editor.

Adding Columns

DatabaseSpy provides different methods for adding columns to a table: you can either define a new column from scratch in the Design Editor, or re-use an existing column definition from a different table.

To create a new column from scratch:

1. In the Design Editor, do one of the following:

    *   Select the table and choose the menu option **Design Editor | Create New Column**.

    *   Press **Alt+C**.

    *   Right-click the table and select **Insert new | Column** from the context menu.

    *   Click the plus ![icon] icon to the right of the last column name.

    The column is inserted into the table and shows a New ![icon] icon to the left of the column name which is highlighted for editing.

2. Enter a descriptive name for the column and press **Enter**.
    The SQL statement in the Database Structure Change Script window is updated.

3. Optionally, repeat steps 1 and 2 to insert additional columns.

4. Modify the column properties as required.

5. Optionally, change the column order by right-clicking a column and selecting **Move Column Up** or **Move Column Down** from the context menu.

6. Execute the change script.

Re-using existing column definitions

Two options in DatabaseSpy allow you to add columns to a table that are based on existing columns in another table: (1) You can generate an SQL statement that adds the selected columns to a table. By default, the table from which the columns originate is included in the statement and has to be edited so as to add the columns to a different table. (2) When the target table has already been created and is displayed as table design in a Design Editor window, you can drag one or several columns from the Online Browser or from a different table design and drop the column(s) onto the Columns section of the target table. You can even drag an entire table from the Online Browser, this will add all columns of the source table to the target table.

To duplicate columns from other tables:

1. Select the column to be duplicated in the Online Browser or in the Design Editor and, keeping the mouse button pressed, drag it into the Columns section of the table design where you wish to add the column.

2. Optionally, edit the column properties and/or the column name.

3. Execute the change script.

To generate SQL for creating a column using an existing column definition:

1. In the Online Browser, right-click the column that serves as a model for the new column to be created.

2. Select the menu option **Show in new SQL Editor | Add** from the context menu.

3. In the SQL Editor, change the column name and edit the column definition, if required.

4. Click the **Execute** button or press **F5**.

Deleting Columns

Several options are available for deleting an individual column, or several columns in bulk. To delete more than one column in one step, select the columns in the Online Browser or Design Editor while keeping the **Ctrl** key pressed (**Ctrl-Click**).

To delete a column from a table:

1. Do one of the following:

   • In the Online Browser, right-click a column and choose **Delete** from the context menu or press **Del**.

   • In the Design Editor, right-click and choose **Delete selected Columns** from the context menu or press **Shift+Del**.

   A change script is generated and the column is marked for deletion .

   • In the Online Browser, right-click a column and select **Show in new SQL Editor | Drop** from the context menu.

   A DROP statement is generated in a new SQL Editor window.

2. Depending on your choice in step 1, do one of the following:

   • In the Database Structure Change Script window, click the **Execute Change Script** button.

   • In the SQL Editor, click the **Execute** button or press **F5**.

Defining a Primary Key

By default, when creating a new table in DatabaseSpy's Design Editor, a primary key is automatically defined on the first column. This option can, however, be disabled in the Design Editor Options, you will then have to define a primary key manually. The instructions below show how to add a primary key in Design Editor. In addition, you can always enter and execute the appropriate SQL statement in an SQL Editor window.

1. If a table already has a primary key, no primary key may be defined when you show a table in a Design Editor window.
2. The column on which the primary key is defined must not be nullable (that is, the **Nullable** check box next to it must be cleared in the table design). If you attempt to create a primary key on a nullable column, a warning dialog box is displayed.

There are several ways to define a primary key in the Design Editor:

• Select the **Make Primary key** command from the context menu of a column in a table design.
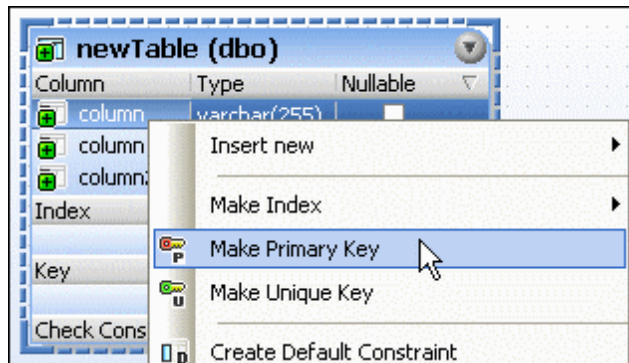
- Drag a column from the Columns section and drop it onto the Key section of a table design. To create a primary key on multiple columns, first select multiple columns while holding the **Shift** key pressed, and then drag them into the Keys section. This menu is also displayed if you click the **Create new Key Constraint** icon in the Key section of the table design.
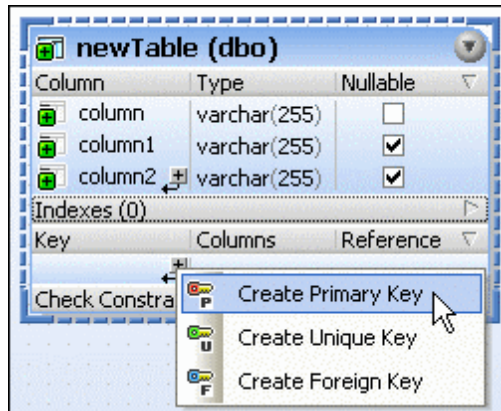
- Right-click anywhere into the table design and select **Insert new | Key | Primary Key** from the context menu. Alternatively, expand the Keys section, click the **Create new Key Constraint** icon at the bottom of the Key column and select **Create Primary Key** from the menu that pops up.

Regardless of the approach chosen above, the primary key is added to the Key section of the table design.
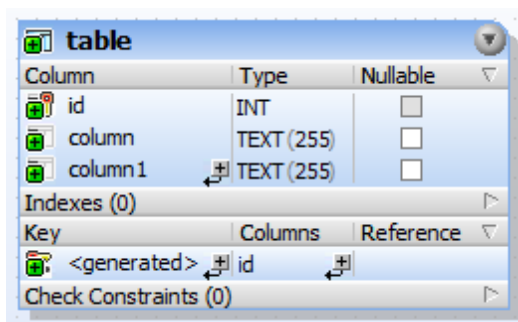
By default, the name of the new primary key is `<generated>`, which means its name will be

automatically generated by the database. This ensures uniqueness of the name in the database. If necessary, you can override the generated name. To do this, enter a descriptive name in place of `<generated>`, and press **Enter**.

> If you override the generated name with a custom name, the new name must be unique; otherwise, errors will occur when the database change script is executed.

Note: In MySQL databases, the name of the primary key constraint is always PRIMARY. In SQLite, custom key names are ignored by the database; therefore, the primary key displayed in the Online Browser of DatabaseSpy appears with the name "<generated>".

To add the primary key on multiple columns, click the **Add Column to Key** 🔲 button (at the bottom of the "Columns" column) and choose a column from the drop-down list that opens. To remove a column from a primary key, right-click the column (in the "Columns" column), and then select **Delete selected objects** from the context menu.

As always when making changes to the database structure, a change script is generated which must be executed in order to actually create the primary key in the database.

Columns that have a primary key assigned are marked with the primary key 🔑 icon in the Design Editor and the Online Browser.

Viewing and Modifying Primary Keys

After you have executed the change script, the primary key 🔑 icon for columns is displayed to the left of the column name in the Columns folder of the Online Browser and when you show the table in the Design Editor. In the Online Browser, the name of the primary key constraint will also show up in the Keys sub-folder of the respective table and will be marked with the primary key 🔑 icon.
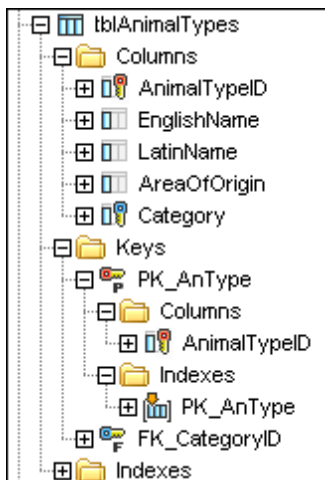
In the Online Browser, the column definition can be viewed in detail by expanding the primary key entry in the Keys folder.
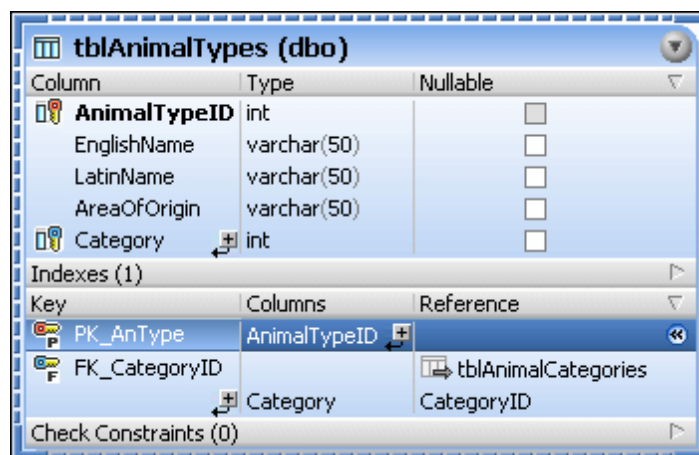
In the Design Editor, the column the primary key is assigned to is displayed in bold when you select the primary key constraint in the Keys section.

Modifying a primary key constraint

You can rename, delete or change the definition of a primary key constraint. Note that the primary key constraint is always deleted first and then recreated with the new definition. The respective SQL statement is generated automatically in the Database Structure Change Script window if you modify the primary key definition in the Design Editor. A primary key cannot be deleted if it is referenced by a foreign key in another table.

Defining and Modifying Unique Keys

This topic shows you how to define a unique key in the Design Editor. You can also define a unique key by entering and executing the appropriate SQL statement in an SQL Editor window.

When defining a unique key for a table, please consider that if the unique key constraint is applied to an existing column, this column must not contain duplicate values. The column may be nullable, however the NULL value can occur only once.

There are several ways to define a unique key in the Design Editor; the procedures are similar to those described for primary keys:

- Select the **Make Unique key** command from the context menu of a column in a table design
- Drag a column from the Columns section and drop it onto the Keys section of a table design
- Click the **Create new Key Constraint** icon in the Keys section of a table design
- Right-click anywhere into the table design and select **Insert new | Key | Unique Key** from the context menu

Regardless of the approach chosen above, the key is added to the Key section of the table design. By default, the name of the new unique key is `<generated>`, which means its name will be automatically generated by the database. This ensures uniqueness of the name in the database. If necessary, you can override the generated name. To do this, enter a descriptive name in place of `<generated>`, and press **Enter**.

> If you override the generated name with a custom name, the new name must be unique; otherwise, errors will occur when the database change script is executed.

As always when making changes to the database structure, a change script is generated which must be executed in order to actually create the unique key in the database. DatabaseSpy performs a check as to whether the content in the constrained column is unique. If the check fails, a dialog box opens displaying a "duplicate key was found" error. The exact contents of the error message depends on the database type.
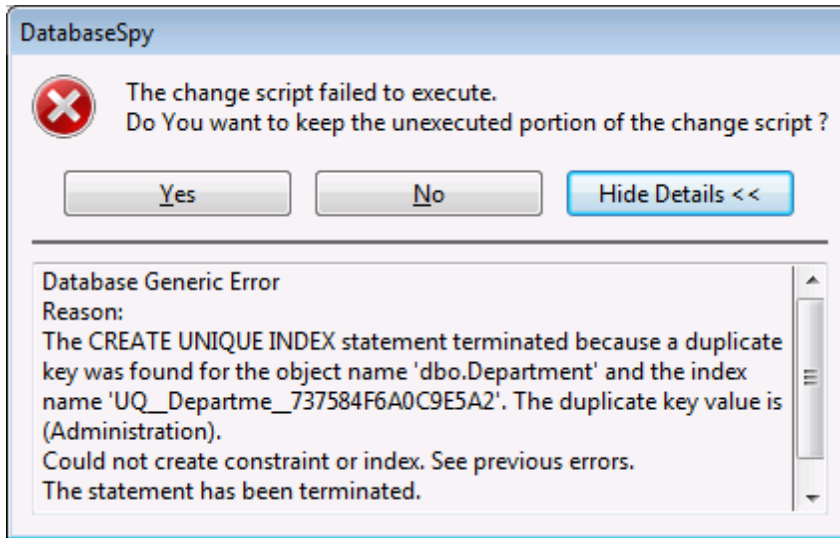
Columns that have a unique key assigned are marked with the unique key icon in the Design Editor and the Online Browser.

Viewing unique key constraints

After you have executed the change script, the unique key icon is displayed to the left of the column name in the Columns folder of the Online Browser and when you show the table in the Design Editor. In the Online Browser, the name of the unique key constraint will also show up in the Keys sub-folder of the respective table and will be marked with the unique key icon.
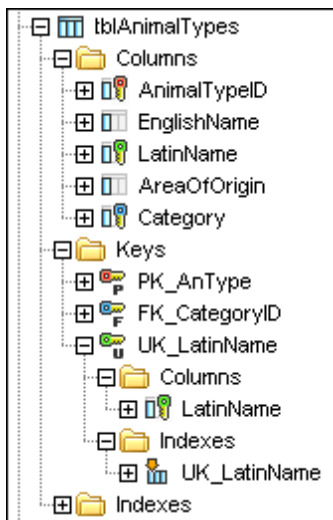
In the Online Browser, the column definition can be viewed in detail by expanding the unique key entry in the Keys folder.
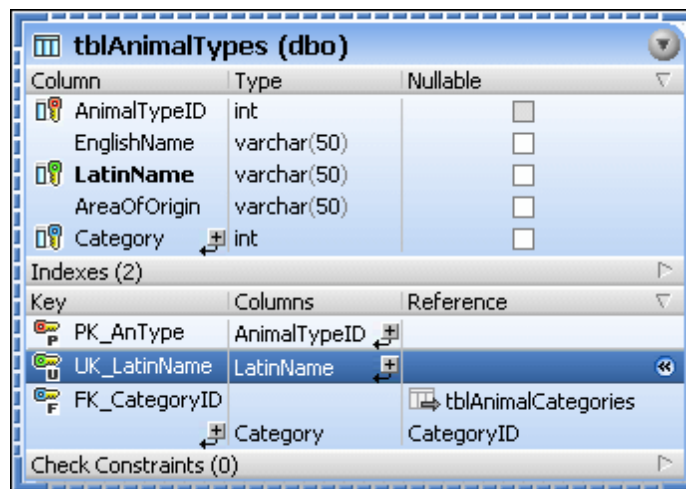
In the Design Editor, the column the unique key is assigned to is displayed in bold when you select the unique key constraint in the Keys section.

Modifying a unique key constraint

You can rename, delete or change the definition of a unique key constraint. Note that the unique key constraint is always deleted first and then recreated with the new definition. The respective SQL statement is generated automatically in the Database Structure Change Script window if you modify the unique key definition in the Design Editor. A unique key cannot be deleted if it is referenced by a foreign key in another table.

Defining Foreign Keys

A foreign key constraint can reference a column in another table if this column has (i) a primary key constraint, (ii) a unique key constraint, or (iii) in SQL Server, a unique index defined on it.

> The column that is referenced by the foreign key in the other table has to be of same data type and length as the referencing column.

There are several ways to define a foreign key in the Design Editor which basically use procedures similar to those described for primary keys.

Approach 1

1. Add to the Design Editor both the table on which the foreign key should be and the referenced table.
2. Do one of the following:
   o Drag a column from the Columns section of a table design (or from the Online Browser) and drop it onto the Keys section of a table design.
   o Drag a primary key, unique key, or (in SQL Server) unique index from a table design (or from the Online Browser) and drop it onto the Keys section of a table design.

By default, a foreign key having the name `<generated>` is automatically created on the first column that has the same data type as the column that is referenced by the dragged key or index, respectively. This may or may not be the column on which you need the foreign key. Therefore, if applicable, double-click the column (in the Keys section, under "Columns") and select the correct column from the list.
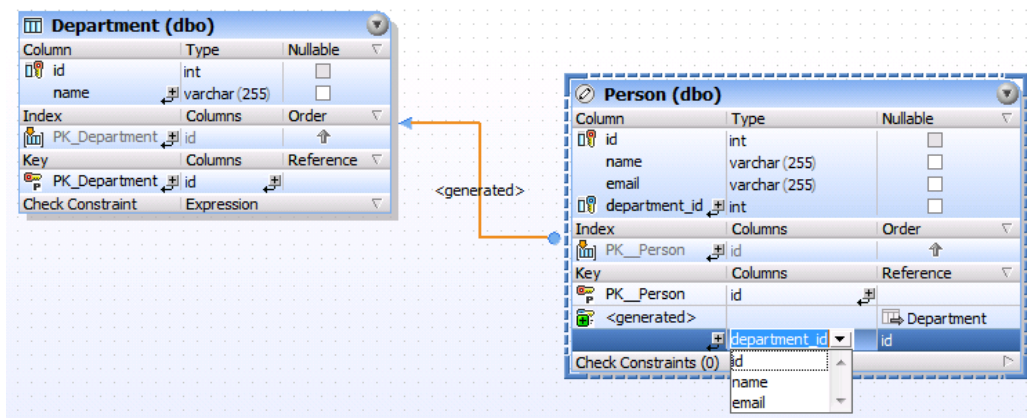
Approach 2

1. Do one of the following:
   o   Click the **Create new Key Constraint** ⊞ icon in the Keys section of a table design.
   o   Right-click anywhere into the table design and select **Insert new | Key | Foreign Key** from the context menu.

   A foreign key having the name <generated> is created automatically, without a reference to a particular table. To choose the referenced table, click **[select the referenced table]** and select the required table from the list.
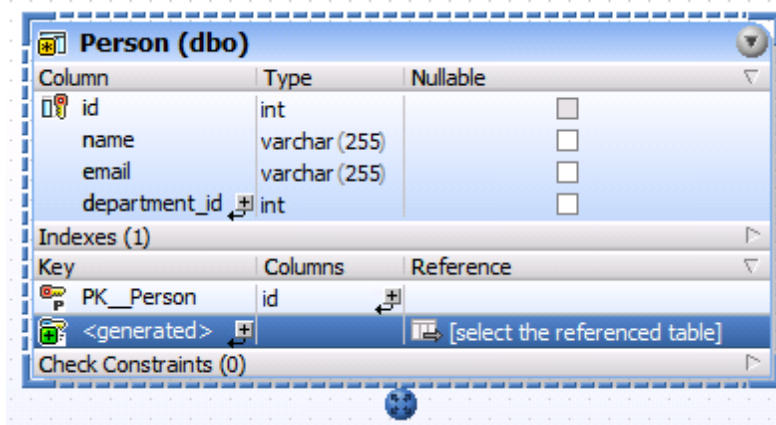
2. If applicable, set the "from" and "to" columns for the foreign key, as follows:

   o   To change the column on which the foreign key is, double-click the column (in the Keys section, under "Columns") and select the appropriate entry from the list.
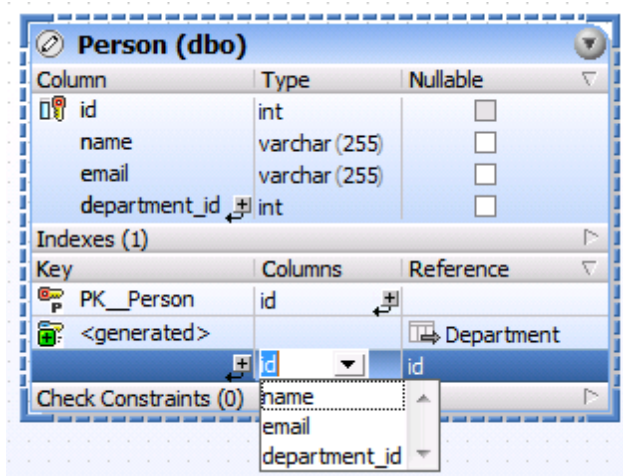
   o   To change the column to which the foreign key points, double-click the column (in the Keys section, under "Reference"), and select the appropriate entry from the list.

Naming foreign keys

By default, the name of the foreign key is `<generated>`, which means its name will be automatically generated by the database. This ensures uniqueness of the name in the database. If necessary, you can override the generated name. To do this, enter a descriptive name in place of `<generated>`, and press **Enter**.

> If you override the generated name with a custom name, the new name must be unique; otherwise, errors will occur when the database change script is executed.

Note: In SQLite, custom key names are ignored by the database. The key name displayed in the Online Browser of DatabaseSpy is the internal key name as provided by the database.

Duplicating foreign keys

If you want to duplicate an existing foreign key, you can generate an SQL statement that adds a foreign key constraint to the table, as follows.

1. In the Online Browser, expand the Keys folder of the table that contains the foreign key you want to duplicate and right-click the foreign key.
2. Select **Show in new SQL Editor | Add** from the context menu.
3. In the SQL Editor, edit table name, column name, as well as the name of the foreign key as required.
4. Click the **Execute** ▷ button to execute the SQL statement and create the new foreign key.

Cross-schema references

Foreign keys can also reference a column in a table that is located in a different schema. When defining this reference using the Design Editor, tables in the same schema are printed black, whereas tables located in a different schema are printed blue in the drop-down list that opens when you double-click a table name in the Reference column of a table design.
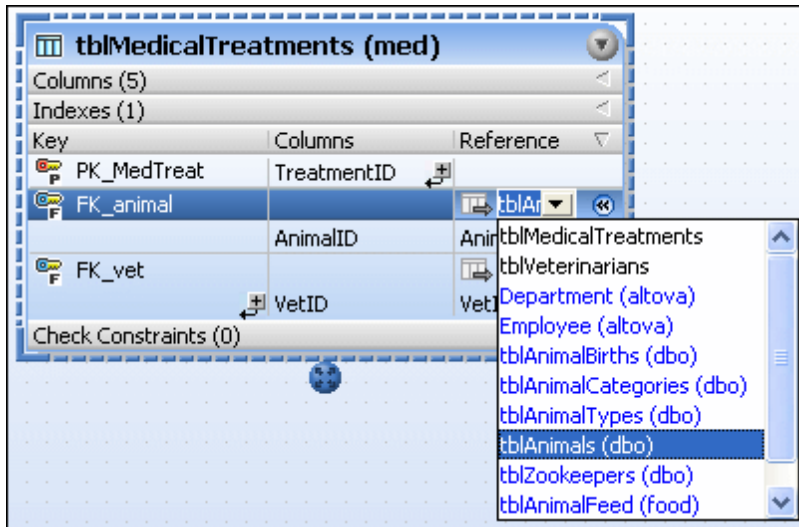
Viewing and Modifying Foreign Keys

After you have executed the change script, the foreign key icon for columns is displayed to the left of the column name in the Columns folder of the Online Browser and when you show the table in the Design Editor. In the Online Browser, the name of the foreign key constraint will also show up in the Keys sub-folder of the respective table and will be marked with the foreign key icon.
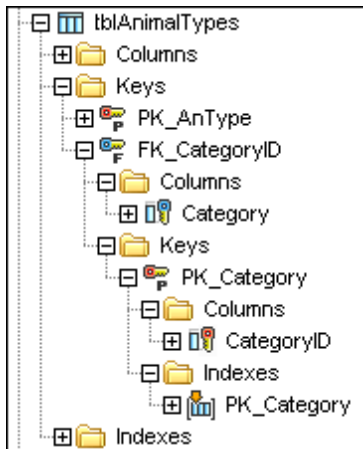
In the Online Browser, the column definition can be viewed in detail by expanding the foreign key entry in the Keys folder.
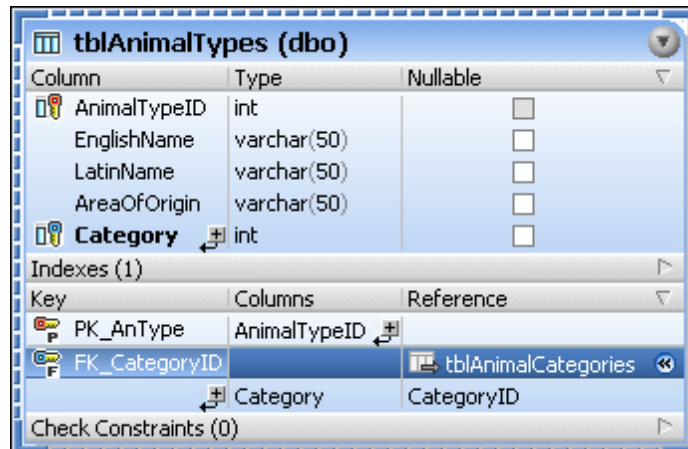
In the Design Editor, the column the foreign key is assigned to is displayed in bold when you select the foreign key constraint in the Keys section.

You can also display the referenced table in the Design Editor window. Right-click the table design and select **Add related tables | Referenced Tables** from the context menu. The table that is referenced by the foreign key constraint is added to the design. Click the connector line or the label to indicate all keys and columns that are involved in the foreign key relation.
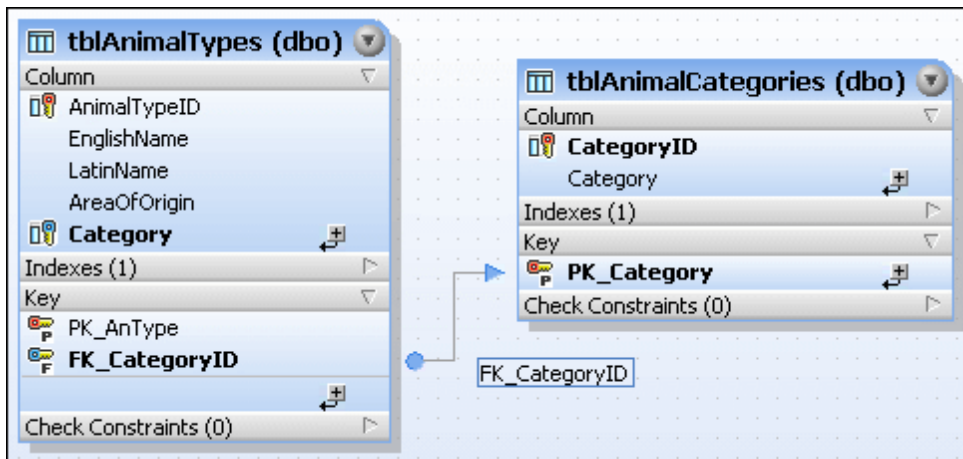
Modifying a foreign key constraint

You can rename, delete or change the definition of a foreign key constraint. Note that the foreign key constraint is always deleted first and then recreated with the new definition. The respective SQL statement is generated automatically in the Database Structure Change Script window if you modify the foreign key definition in the Design Editor.

Defining and Modifying Check Constraints

Check constraints can be defined both on table and on column level and make sure that only valid data is entered into a specific column of a table. If the condition is not met, the respective column cannot be updated and new rows cannot be added *(see screenshot below)*. DatabaseSpy outputs a corresponding message in the Message tab.
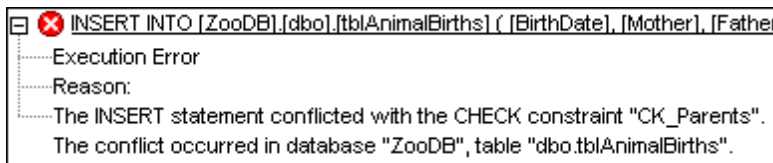
Note: Adding check constraints is not supported for SQLite and MySQL databases.

To define a check constraint in the Design Editor:

1. Do one of the following:

   - Right-click into the table and select **Insert new | Check Constraint** from the context menu.
   - Expand the Check Constraints section by clicking the triangle at the right edge of the section header and click the **Create new Check Constraint** icon in the Check Constraint column.

   A new check constraint is added to the Check Constraints section and highlighted for editing. By default, the name of the foreign key is <generated>, which means its name will be automatically generated by the database. This ensures uniqueness of the name in the database. If necessary, you can override the generated name. To do this, enter a descriptive name in place of <generated>, and press **Enter**.

> If you override the generated name with a custom name, the new name must be
> unique; otherwise, errors will occur when the database change script is executed.

2. In the Expression column, enter the expression that is to be checked when data is
    committed to the database and press **Enter**.
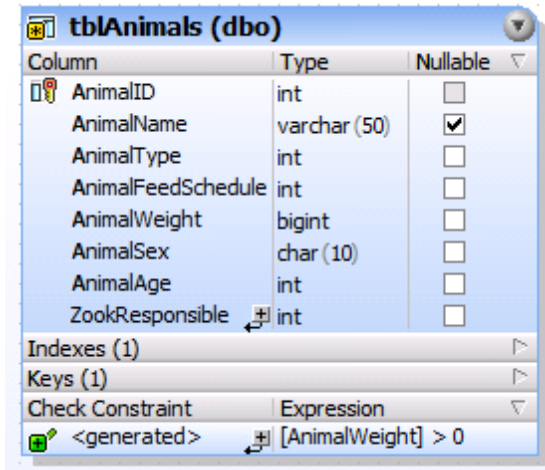
As always when making changes to the database structure, a change script is generated which
must be executed in order to actually create the check constraint in the database. DatabaseSpy
performs a validation of the expression that will be used for the check constraint. While you are
defining a check constraint in the Design Editor, the Database Structure Change Script window
displays comments if the expression is invalid.

```
-- region adding check constraints
-- Error: missing expression for check constraint CK_1.
-- ALTER TABLE [ZooDB].[dbo].[table1] ADD CONSTRAINT [CK_1] CHECK ( )
-- endregion adding check constraints
```

The following issues are validated and corresponding messages are displayed in the change
script:

- Missing expression
- Invalid identifier
- Textual or numeric constant not matching column data type
- Column data types not matching
- Invalid expression
- "IS NOT NULL" clause applied to a nullable column

Viewing check constraints

In the Design Editor, check constraints on table level are displayed in a separate section where
they can be edited and where you can add additional or delete unneeded check constraints.

Columns that have a check constraint assigned are marked with the check constraint ▣ icon in the Design Editor and the Online Browser. Check constraints on column level can be edited in the Properties window of the column to which they are applied. Please note that they do not show up in the Check Constraints section of the table. To see the name and definition of this column's constraint, you have to display the column properties in the Properties window.

In the Online Browser, check constraints appear in the constraints sub-folder of the table or column, depending on whether the constraint has been defined on table or column level.

Check constraints on table level *(above screenshot)* appear in the Constraints folder of the table they are assigned to.

Check constraints on column level *(right screenshot)* are displayed under the Constraints folder of the column. Note that no Constraints folder on table level is available in this case.
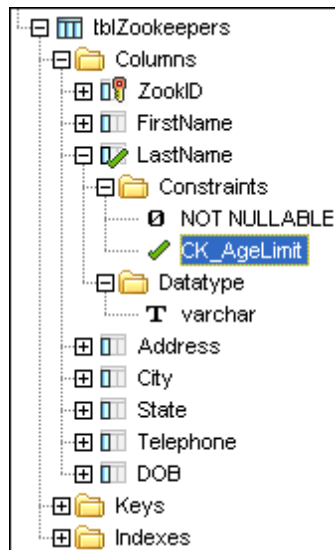
Modifying check constraints

You can rename, delete or change the expression of a check constraint. Note that the check constraint is always deleted first and then recreated with the new definition. The respective SQL statement is generated automatically in the Database Structure Change Script window if you modify the check definition in the Design Editor.

When you rename a column that is used in a check constraint expression, DatabaseSpy automatically updates the check constraint in the change script that is used to rename that column.

If a column is deleted, any check constraints that reference this column are dropped as well.

Adding check constraints using SQL

If check constraints already exist in a table, you can add new constraints to the table also by right-clicking an existing check constraint and generating a statement that uses the properties of the existing item as default. After editing the properties as required and executing the SQL, a new check constraint is added to the database.

To add a check constraint to a table using SQL:

1. Connect to the database and show the respective data source in the Online Browser.
2. Expand the table to which you want to add a check constraint and right-click an existing constraint in the Constraints folder.
3. Select **Show in new SQL Editor | Add** from the context menu or, holding down the right mouse button, drag the constraint into an open SQL Editor window and choose **Add** from the popup. An ALTER TABLE statement is generated in the SQL Editor window.
4. Edit the statement as follows:

    1. Enter the path of the table that is to be checked after the ALTER TABLE part of the statement.
    2. Type the name of the new constraint after the ADD CONSTRAINT part of the statement.
    3. Edit the CHECK (Expression) section; enter the expression that is to be checked when the table is updated.

5. Click the **Execute** button to execute the statement and to add the new check constraint to the database.
6. In the Online Browser, right-click the database and choose **Refresh** from the context menu.

Defining and Modifying Default Constraints

The table design in the Design Editor provides an option in the context menu to define a standard value for a column. This command inserts the Default field into the column properties where you can then enter the actual standard value for the column.

To define a default value for a column:

1. In the Design Editor, right-click the column for which you want to define a default value and select **Create Default Constraint** from the context menu. In the column properties, the Default field is added to the General section and to the table design.

    Alternatively, right-click the header of the Columns section in the table design and activate the Default Value column. Double-click that column in the row of the column for which you want to define a default constraint.

2. Enter the default value into the Default field (either in the Properties window or directly in the table design).

As always when making changes to the database structure, a change script is generated which must be executed in order to actually create the default constraint in the database.

Viewing default constraints

After you have executed the change script and unless a constraint with a higher priority has already been defined for that particular column, the default ⬛ icon is displayed to the left of the column name in the Columns folder of the Online Browser and when you show the table in the Design Editor.

In the Online Browser, the name of the default constraint, which is composed of the column name plus the "Default_" prefix, shows up in the Constraints sub-folder of the respective column and is marked with the default constraint **Def** icon.
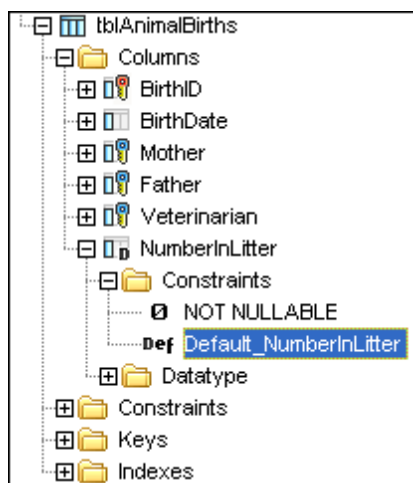
The actual value of the default constraint, that is the default value that will be inserted into the column if a new row is added to the table, can be viewed in the Properties window of the

respective column when displayed in the Design Editor.

Modifying default constraints

You can delete a default constraint or change its default value. Note that the default constraint is always deleted first and then recreated with the new value. The respective SQL statement is generated automatically in the Database Structure Change Script window if you change the default value in the Design Editor or in the Properties window.
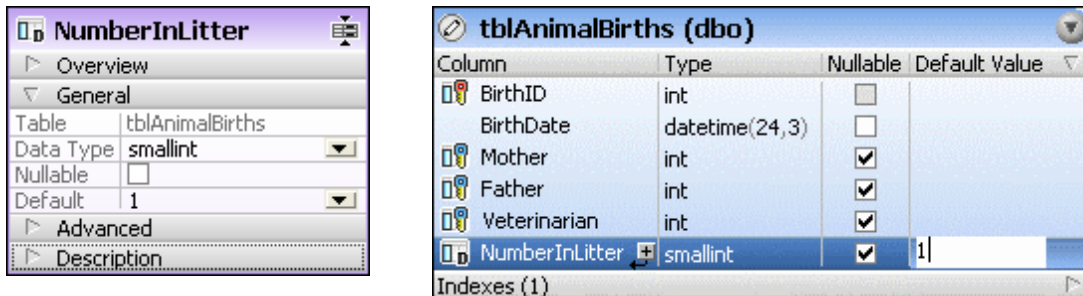
To change the default value of a column:

1. In the Online Browser, select the column and make sure that its properties are being displayed. Alternatively, display the `Default Value` field in the table design.
2. Edit the default value in the `Default` field of the General section, or in the appropriate row of the Default Value column in the table design, respectively.
3. Execute the change script.

To delete a default constraint:

1. In the Online Browser, select the column and expand its Constraints sub-folder.
2. Right-click the default constraint and select **Delete** from the context menu or select the constraint and press **Del**.
3. Execute the change script.

Creating Identifier Columns

The definition of identifier columns is handled differently in the individual database types.

Note: Identifier columns are not supported for the following database types: Firebird, Oracle, PostgreSQL, Progress OpenEdge, and SQLite.

▣ IBM DB2

In IBM DB2 databases, the `Nullable` check box of a column must be deactivated in order to allow for the `Identity` property to be activated. When the `Identity` check box is activated, two additional fields appear in the General section of the column properties: Enter the initial value that will be inserted into the first row that is created for this table into the `Seed` field, and specify the amount by which this value will be incremented for additional rows in the `Increment` field.

The identifier column of a table can defined when a new table is created and the column has to be of data type INTEGER, BIGINT, SMALLINT, or DECIMAL or NUMERIC with a scale of 0.

▣ Microsoft Access

Microsoft Access databases provide a special COUNTER data type that is automatically applied to the column when the `Auto Increment` check box is activated in the Advanced section of the column properties. It is not important which data type is applied to the column when the check box is activated. If the `Nullable` property is checked, a warning is displayed in the Database Structure Change Script window and the `Nullable` property is automatically deactivated when the change script is executed. The initial value of the identity column for the first row to be added is 1, the value will be incremented by 1 for any additional row that you add.

The identifier column of a table can be defined when a new table is created or when a new column is added to an existing table and defined as identity column.

⊟ Microsoft SQL Server

In Microsoft SQL Server databases, the `Nullable` check box of a column must be deactivated in order to allow for the `Identity` property to be activated. When the `Identity` check box is activated, two additional fields appear in the Advanced section of the column properties: Enter the initial value that will be inserted into the first row that is created for this table into the `Seed` field, and specify the amount by which this value will be incremented for additional rows in the `Increment` field.

The identifier column of a table can defined when a new table is created or when a new column is added to an existing table and defined as identity column. The column has to be of data type int, bigint, smallint, tinyint, or decimal or numeric with a scale of 0.

⊟ MySQL

In MySQL databases, the `Nullable` check box of a column must be deactivated in order to allow for the `Auto Increment` property to be activated. The identifier column of a table can defined when a new table is created and the column has to be of a numeric data type.

⊟ Sybase

In Sybase databases, the `Nullable` check box of a column must be deactivated in order to allow for the `Identity` property to be activated.

The identifier column of a table can defined when a new table is created and the column has to be of data type int, bigint, smallint, tinyint, or decimal or numeric with a scale of 0.

As always when making changes to the database structure, a change script is generated which must be executed in order to actually create the identifier column in the database. DatabaseSpy performs a validation of the column definition that will be used for the identifier column. While you are defining an identifier column in the Design Editor, the Database Structure Change Script window displays comments if the definition is invalid.

```
-- region creating tables
-- Cannot create identity column that is nullable.
Please revise the attributes of column 'newColumn'.
-- CREATE TABLE [ZooDB].[guest].[newTable] ( [newColumn]
varchar (255)  IDENTITY (1, 1) NULL )
-- endregion creating tables
```

To create an Identifier column:

1. In the Design Editor or in the Online Browser, select the column you want to use an as identity column.
2. In the Properties Window, activate the `Identity` check box or the `Auto Increment`

check box, respectively. Alternatively, display the Identity column in the table design and activate the check box for the identity column.

3. Optionally change the default values in the `Seed` (initial value) and `Increment` fields, if applicable.

4. Execute the change script.

16.1.5 Viewing Tables

You can view the structure of any existing table in your database by dragging it from the Online Browser into the Design Editor window or by using the corresponding context menu option. The most important table and column details are immediately displayed in the Design Editor's graphical view. Additional sections for indexes, keys, and check constraints can be expanded for the relevant details.

To view a table in a Design Editor window:

- Do one of the following:

  - Open a new Design Editor window and drag a table from the Online Browser into the window.

  - Right-click a table in the Online Browser and select **Show in new Design Editor** from the context menu.

  - To add a table to an existing Design Editor window, right-click the table in the Online Browser and select **Add to Design Editor** from the context menu.

To auto-size a table:

- In a Design Editor window, click the **Autosize** 🔷 icon at the bottom of the table to view all content properly.

To remove a table from a Design Editor window:

- Do one of the following:

  - Right-click the table and select **Remove from Design** from the context menu.

  - Select a table and press **Ctrl+Del**.

  - Select a table and choose the menu option **Design Editor | Remove from Design**.

  Please note that removing a table from a design does not delete it from the connected database.

Table layout

You can define the default table layout in the Design Editor options, however you can also change the layout in the Design Editor itself. Apart from the **optimized view** which shows the fields `Column Name`, `Type`, and `Nullable` for each column, you can choose two more options for displaying tables in the Design Editor.

The **compact view** shows only the column, index, and constraint names as well as the icons that indicate the constraints that have been defined for a column or the index or constraint type, respectively.
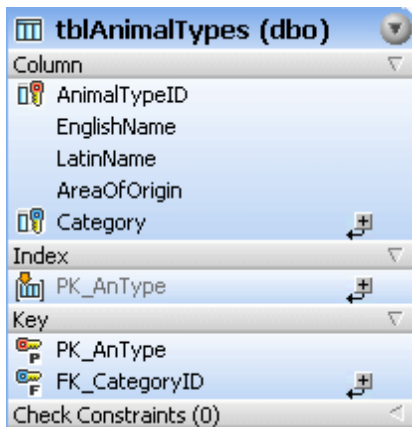
The **collapsed view** shows only the table header.

This view is particularly useful if you want to view all the tables and relations of the entire database and do not need detailed information about the individual tables.

To view a table in the compact view:

- Right-click the table in a Design Editor window and activate **Compact View** in the context menu.

To switch off compact view for a table:

- Right-click the table and deactivate **Compact View** in the context menu.

To collapse or expand a table:

Do one of the following:

- Click the arrow in the upper right corner of a table to toggle between collapsed and expanded view.

- If you want to expand or collapse several tables at a time, select the tables using **Ctrl +Click** and click **Expand selected Tables** or **Collapse selected Tables** , respectively, in the Design Editor toolbar.

Viewing Table Columns

When you drag a table into a Design Editor window or use the **Show in new Design Editor** command from the context menu of the Online Browser, the table appears in the Design Editor in the optimized view: column name, data type, and the Nullable property are displayed and can be edited directly in the table design.

Every column that has a special quality, such as, for example, forming part of a relationship, is marked with an icon in front of its name in Design Editor. If a column has more than one constraint assigned, only the constraint with the highest priority is depicted in the column icon. The priority of the constraints is as follows:

Primary key column: This column is used as the table's primary key.

Unique column: This column has a unique constraint defined and therefore accepts only data that is unique in the context of the column.

Referencing column: This column has a foreign key constraint assigned and references the primary key of a different table.

XML column: The data contained in this column are of type XML.

Checked column: This column has a check constraint defined which ensures that only approved values are present in the respective column.

Default value: A default constraint has been defined for this column.

Displaying additional information

When you right-click the header of the table design's Column section, you can select a number of additional properties (Default Value, Description, Identity, Increment, and Seed) to be displayed in the design. Check the properties you want to display in the context menu or select **Show all additional columns**.

Viewing Indexes

When you expand the Indexes section, the table's indexes are displayed in such a way that each column forming part of a particular index is shown in a row of its own and there is a separate column indicating whether it is an ascending or descending index.

The type of an index is indicated with icons. Non-clustered indexes do not have an icon. The following icons are used:

   Clustered index

   XML index (available only for columns of type XML)

You can rename an index, change, add, or remove columns, or modify the order directly in the table design. Indexes that have been created automatically for primary or unique keys cannot be modified.

To view the index properties in detail, select an index and view the properties in the Properties window.

Viewing Check Constraints

A check constraint is used to define a range of values that are accepted by a column. It uses a logical expression to determine the acceptable values. Check constraints can be defined on table level (i.e., several columns of a table are checked in the expression) or on a single column. Multiple check constraints can be applied to a single column.

Check constraints **on table level** are shown with an icon when you expand the Check Constraints section of a table design. They are displayed together with the expression(s) that are used to validate the table.

Check constraints on table level can be renamed and their expressions can be modified directly in the table design.

A check constraint **on column level** is indicated by the check column icon being displayed with the column the check constraint is defined for in the Columns section of the table design. Please note that this icon is not displayed if either a primary key, unique key, or foreign key is defined for this column, or if the column is of type XML, since all these qualities are of a higher priority and their corresponding icon will be displayed with the column.

You can edit the expression in the Properties window. To rename a check constraint on column level, right-click the constraint in the Online Browser and select **Rename** from the context menu.

Viewing Key Constraints

When you expand the Keys sections, the key constraints that have been defined for the table are displayed in such a way that each column forming part of a particular key is shown in a row of its own. For foreign keys, there is an additional column where the column that is referenced in another table is displayed.

Key	Columns	Reference	▽
🔑P PK_AnType	AnimalTypeID 🔳		
🔑U UK_LatinName	LatinName 🔳		
🔑F FK_CategoryID		🔳 tblAnimalCategories	
	🔳 Category	CategoryID	

The type of a key constraint is indicated with icons. The following icons are used:

🔑P   **Primary key**: This constraint is the primary key of the table. The primary key ensures that no duplicate data and no NULL values can be inserted into the respective column(s). Only one primary key is allowed in a table.

🔑U   **Unique key**: This constraint makes sure that the values entered in a specific column not participating in a primary key are unique. Multiple unique keys can be defined on a table and the NULL value is allowed in the respective table(s). Note, however, that the value NULL can only occur once per column. A unique key can be referenced by a foreign key.

🔑F   **Foreign key**: This constraint creates a foreign key relation where the primary or unique key of a table is referenced by a column in a different table. The primary key column or unique key column, respectively, in the first table is used as a foreign key in the referencing column of the second table. This way, the foreign key ensures that only data that is present in the primary or unique key column of the first table can be inserted into the second table.

You can rename a key constraint, change, add, or remove columns, or modify the reference of a foreign key directly in the table design.

To view the properties of a key constraint in detail, select a key constraint and view the properties in the Properties window.

16.1.6 Viewing Relationships Between Tables

When you drag into the Design Editor two or more tables that are related with each other, the relationships between these tables are shown with lines connecting the related tables. DatabaseSpy uses different colors to distinguish the individual relationships:

- **Green**: Self-reference, i.e. the table uses a key that is assigned to one or more of the table's own columns.
- **Blue**: Incoming reference, i.e. the selected table is referenced by a key used in another table.
- **Orange**: Outgoing reference, i.e. the table has a key defined that references one or several column(s) in another table.

Please note that the color of one and the same relation line may change depending on the table that is currently selected. For example, a foreign key relationship is displayed blue if you select the referenced table, and orange if you select the referencing table. The screenshot below shows an incoming relationship for tblAnimalCategories.
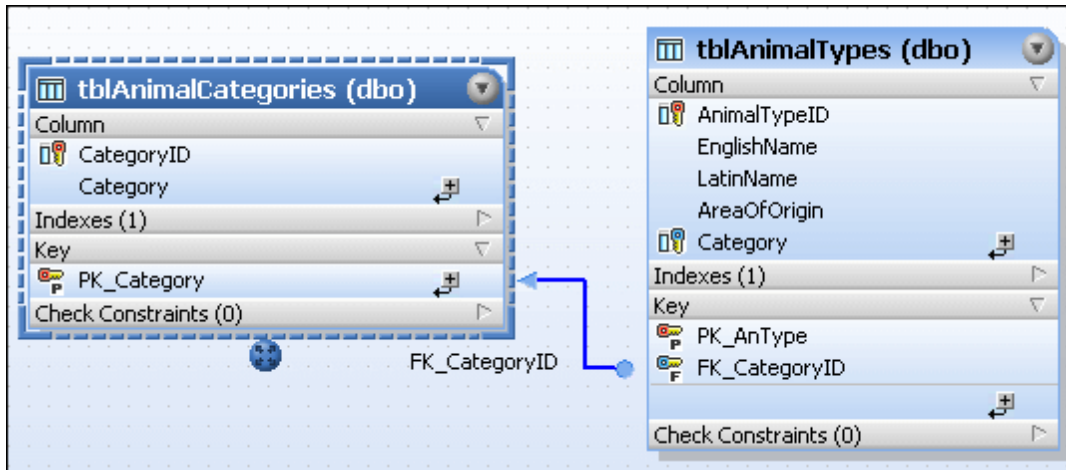
The same relationship, however viewed from the referencing tblAnimalTypes table's point of view, is depicted with an orange relation line (*see screenshot below*).
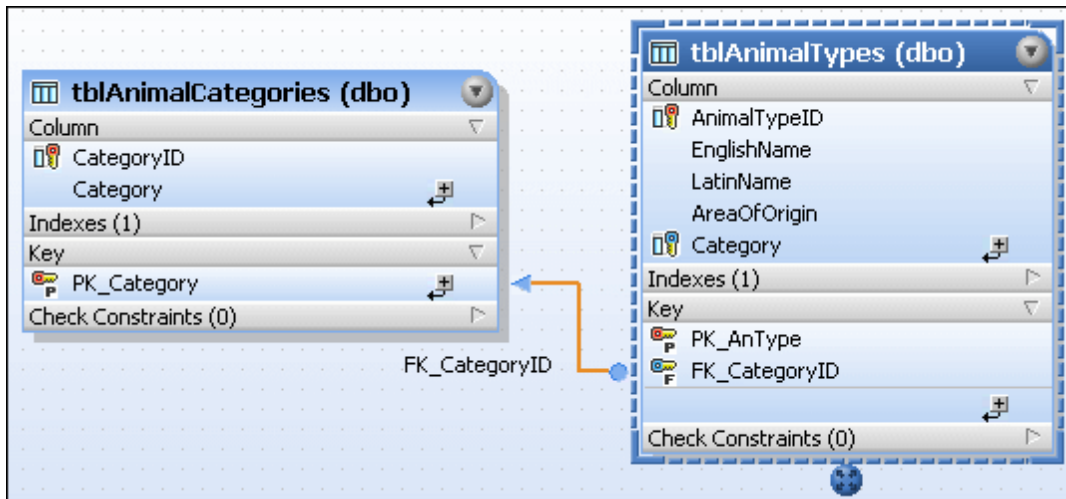
The screenshot below illustrates a self relation that makes sure that the manager number (MgrNo) of an employee's manager is contained in the list of employee numbers (EmpNo).
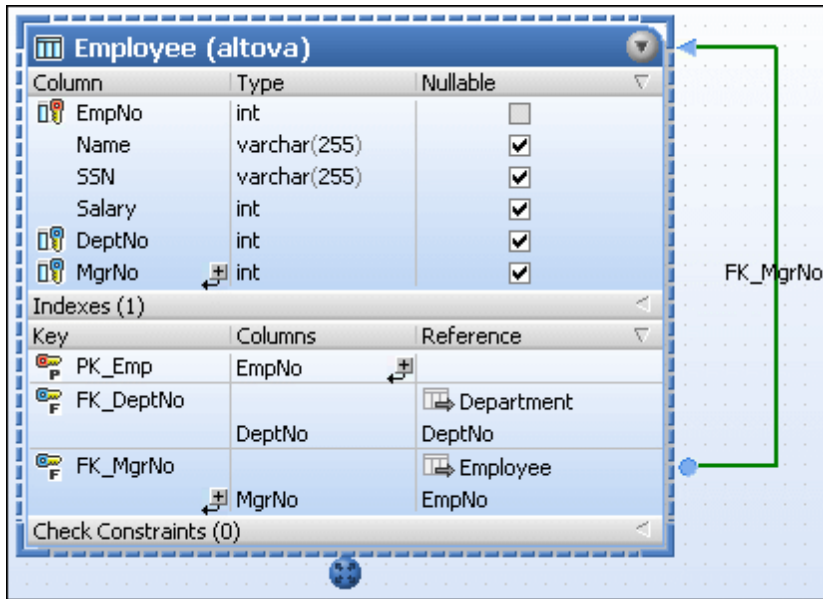
16.1.7 Finding Related Tables

In the Design Editor, you can display the relations between the individual tables. To facilitate this, DatabaseSpy provides an option in the context menu that automatically adds referenced or referencing tables to the active design. You can also save this design in a design file.
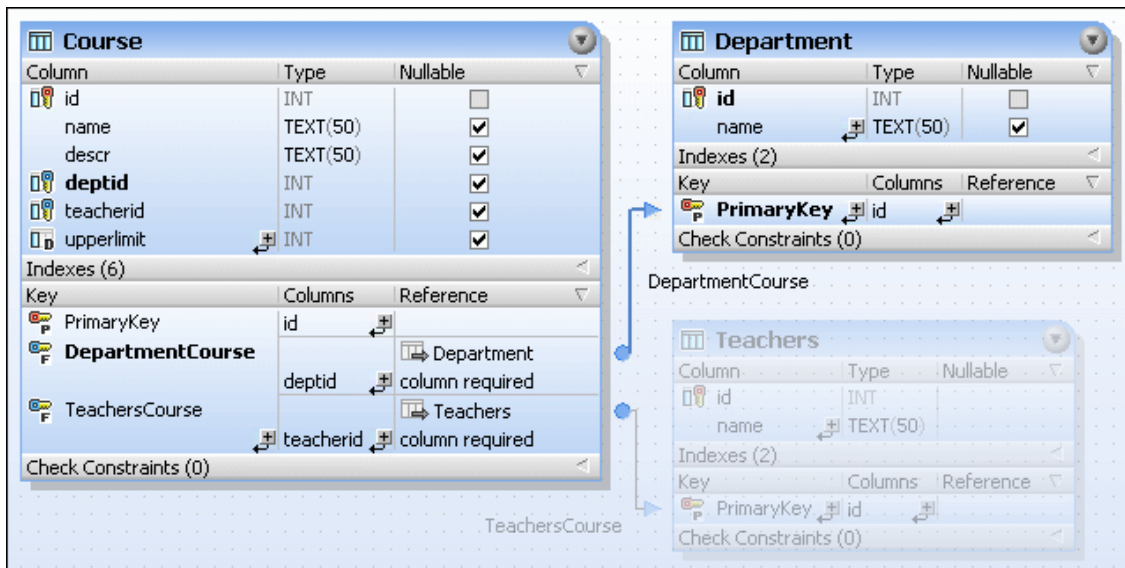
You can also define to highlight objects that form part of the currently selected relation in the Design Editor options. In this case, as depicted in the screenshot above, tables that are not part of the selected relation are grayed out. Also note that the referenced columns are printed bold in both tables so that you can view at a glance which columns are related with each other.

To display related tables in the Design Editor:

1. Right-click a table in the Design Editor and select **AddRelated tables** from the context menu.

2. Choose one of the following options from the sub-menu:
    - Referenced Tables
    - Referencing Tables
    - All Relations

    The respective tables are added to the Design Editor window.

To highlight objects of the currently selected relation:

1. Make sure that the `Highlight active relations objects` check box is activated in the Design Editor options.

2. Click on a connection line.
    The connection line is highlighted and all tables that are not part of this relation are grayed out in the Design Editor window.

16.1.8 Converting Table Structures

Altova web site: convert databases

DatabaseSpy allows you to convert the structure of a selected table so as to fit the syntax requirements of a database kind other than the table was initially created in. This can be done either in the Online Browser or the Design Editor by using Drag and Drop. DatabaseSpy creates a change script that must be executed to commit the changes to the database.

The following restrictions apply to the conversion of database tables:
- Properties that would require parsing (e.g., check constraints or triggers) will not be converted
- Foreign keys will not be converted
- Indexes will not be converted

If database items cannot be converted into the target database, a warning message will be displayed in DatabaseSpy.

To convert the structure of a table to a different database type:

1. Open a Design Editor window for the target database.

2. Optionally, display the source table in a Design Editor window of the source database.

3. Select a table in the Online Browser or Design Editor and drag it into the Design Editor window of the target database.

4. Execute the change script.

16.1.9 Indexes

Indexes can make queries of a database considerably faster since only the indexed columns have to be queried. Indexes are created automatically when you define a primary or unique key on a column. Where supported by the database, DatabaseSpy allows for the definition of clustered, non-clustered, and XML indexes via a context menu option in the Design Editor. For a more fine-tuned definition of indexes in the individual database kinds, specify the appropriate options in the index properties.

In the Design Editor, indexes can be viewed in a separate section of the table display. Here you see at a glance which columns are included in the index definition, and whether they will be queried in ascending or descending order.

Indexes that have been defined automatically, have the same name as the corresponding primary or unique key. These indexes are displayed in the Indexes section but cannot be modified.

Creating and Modifying Indexes in Design Editor

Depending on the kind of database you are using, different types of indexes can be defined.

☐ Microsoft Access

Only non-clustered indexes can be defined. In the index properties, you can activate the `Enforce Data Validation` check box and select "Primary", "Disallow NULL", or "Ignore NULL" as the preferred method of data validation.

☐ Microsoft SQL Server

Clustered, non-clustered, and—as of SQL Server 2005—XML indexes can be created. Only one clustered index can be defined per table, and will normally be reserved for the primary key. If you create an index on a column of type XML in SQL Server 2005, XML is the only available type of index.

You can change the index type in the General section. To ensure that the indexed data is unique, check the `Unique` property. A unique index can also be referenced by a foreign key.

In the Options section of the index properties, you can activate a number of options. A brief description of the individual option is displayed when you hover over an option with the mouse cursor.

☐ Oracle

Only non-clustered indexes can be defined. In the `Uniqueness` check box, you can choose whether the index should be unique or of type "Bitmap". Please note that unique indexes cannot be referenced by a foreign key in Oracle databases.

In the Options section of the index properties, you can activate a number of options. A brief description of the individual option is displayed when you hover over an option with the mouse cursor.

☐ MySQL

Only non-clustered indexes can be defined. In the General section, you can change the index type to "Regular", "Hashed", or "Full text". To ensure that the indexed data is unique, check the `Unique` property. Please note that unique indexes cannot be referenced by a foreign key in MySQL databases.

⊟ IBM DB2

Clustered, non-clustered, and XML indexes can be created. Only one clustered index can be defined per table; note that this does not necessarily has to be the primary key column. If you create an index on a column of type XML, XML is the only available type of index.

You can change the index type in the General section. To ensure that the indexed data is unique, check the Unique property. Please note that unique indexes cannot be referenced by a foreign key in DB2 databases.

In the Options section you can allow reverse scans, and you can define whether the index should be padded or compressed. Activate the `Collect Statistics` check box, if required, and optionally choose sampled or detailed statistics.

⊟ Sybase

Clustered and non-clustered indexes can be defined. Only one clustered index can be defined per table. You can change the index type in the General section. To ensure that the indexed data is unique, check the `Unique` property. Please note that unique indexes cannot be referenced by a foreign key in Sybase databases.

In the General section of the index properties, you can define a number of options. A brief description of the individual option is displayed when you hover over an option with the mouse cursor.

⊟ PostgreSQL

Clustered and non-clustered indexes can be defined. Only one clustered index can be defined per table. You can change the index type in the General section. In the Advanced section, you can change the default access method (which is "B-tree") to "Hash", "GiST", or "GIN". To ensure that the indexed data is unique, check the `Unique` property. Currently, only B-Tree indexes can be declared unique. Please note that unique indexes cannot be referenced by a foreign key in PostgreSQL databases.

To define an index in the Design Editor:

1. Do one of the following:

    *   Select the column you want to include in the index definition and press the right mouse button. Select **Make Index** from the context menu.

    *   Select the column and, keeping the mouse button pressed, drag it into the Index section.

    *   Right-click into the table and select **Insert new | Index** from the context menu.

    *   Expand the Index section by clicking the triangle at the right edge of the section header and click the plus symbol 🔳.

2. Select the appropriate index type (non-clustered, clustered, or XML index) from the sub-menu that opens. The types of indexes that are available depend on the database kind you are using.

    A new index is added, and the prefix `IX_` is inserted into the Index section and highlighted for editing. The first suitable column or the selected column, respectively, is automatically inserted into the Columns column.

3. Optionally, change the index properties in the properties window.

4. In the Index column, enter a descriptive name and press **Enter**.

5. In the Columns column, optionally double-click the column and select a different column from the drop-down list.

6. Click into the Order column and select either ascending or descending sort order and press **Enter**.

7. Optionally, click the plus symbol 🔲 and select a column from the drop-down list and repeat step 7 to add one or more additional columns to the index definition.

8. Execute the change script.

Creating Indexes Based on an Existing Index Definition

If indexes already exist in a table, you can add new indexes to the table also by right-clicking an existing index and generating a statement that uses the properties of the existing item as default. After editing the properties as required and executing the SQL, a new index is added to the database.

To add an index to a table using SQL:

1. Connect to the database and show the respective data source in the Online Browser.

2. Expand the table to which you want to add an index and right-click an existing index in the Index folder.

3. Select **Show in new SQL Editor | Create** from the context menu or, holding down the right mouse button, drag the index into an open SQL Editor window and choose **Create** from the popup.
    An SQL statement is generated in the SQL Editor window.

4. Edit the statement as required.

5. Click the **Execute** ▶ button to execute the statement and to add the new index to the database.

6. In the Online Browser, right-click the database and choose **Refresh** from the context menu.

16.2   Views

Altova web site: database views

In DatabaseSpy you can create a view from a SELECT statement in the SQL Editor, or copy the definition of an existing view via the context menu in the Online Browser. This way, you have frequently used complex queries stored in database and can later execute them with a single mouse click or use them as a basis for more fine-tuned queries.

Views are displayed in the Views folder of the Online Browser. You may have to refresh the data source to view a newly created view in the Online Browser. The context menu provides several options to manage the views in your database (see also Generating SQL Statements).

The SQL syntax in the instructions below may vary depending on the database kind you are working with.

To create a view in SQL Editor:

1. Create a query by typing in SQL statements or opening an SQL file.
2. Right-click in the SQL Editor window and select **Create View As** from the context menu. Alternatively, you can also select the menu option **SQL Refactoring | Create View As**. The statement CREATE VIEW [View1] AS is automatically inserted in front of the query and the view name View1 is automatically highlighted.
3. Change the view name to a more descriptive one. If a view with this name already exists, an error will occur.
4. Click the **Execute** ▷ button or press **F5** to create the view.
5. Refresh the data source to display the view in the Online Browser.

Be aware that, if your SELECT query retrieves data from more than one table, and if identical column names exist in selected tables, then ambiguous column names may occur in queries such as **SELECT * FROM table1, table2**. Creating views from SELECT queries which contain ambiguous columns is not supported and produces an error. The text of the error is different for each database kind and may not necessarily reflect the exact cause. Therefore, before creating views from a SELECT query, make sure that it does not contain ambiguous column names. For example, if the table INVOICE has a column "id" and the table INVOICEDATA has a column "id", an error would occur for a statement such as:

```
CREATE VIEW `View1` AS SELECT * FROM invoice, invoicedata WHERE invoice.id =
invoicedata.id;
```

However, the error would be eliminated with a statement such as:

```
CREATE VIEW `View1` AS SELECT invoice.id AS id1, invoicedata.id AS id2 FROM
invoice, invoicedata WHERE invoice.id = invoicedata.id;
```

Note the syntax in code listings above applies to MySQL.

To create a view based on an existing view:

1. In the Online Browser, right-click an existing view and select **Show in new SQL Editor | Create** from the context menu. A CREATE VIEW statement with the definition of the existing view is displayed in a new SQL Editor window.
2. Type a new name after the CREATE VIEW part of the statement and modify the SELECT part accordingly.
3. Click the **Execute** button or press **F5** to create the view.
4. Refresh the data source to display the view in the Online Browser.

To modify a view:

1. In the Online Browser, right-click an existing view and select **Show in new SQL Editor | Alter** from the context menu. An ALTER VIEW statement with the definition of the existing view is displayed in a new SQL Editor window.
2. Modify the view as required.
3. Click the **Execute** button or press **F5** to modify the view.

16.3   Stored Procedures

Stored procedures can improve the performance of distributed applications by reducing the data traffic between server and client installations. Please note that you must register a stored procedure with the database before it can be called by a client program.

In DatabaseSpy, stored procedures that are registered with the database are displayed in a separate folder in the Online Browser. A context menu provides several maintenance options and, in addition, facilitates the creation of SQL statements for creating, altering, dropping, or executing the stored procedure (see Generating SQL Statements).
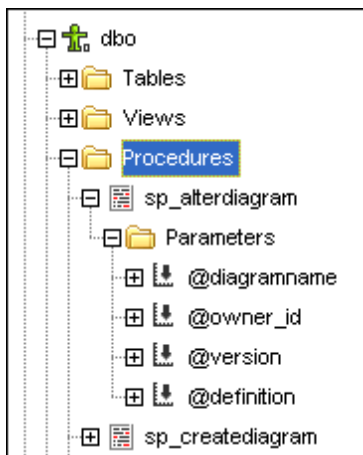
DatabaseSpy provides sub-folders for the procedures where the parameters are displayed. The definitions can be edited and renamed in the Properties window.

You can add stored procedures to the favorites, rename, or delete them. When deleting a stored procedure, a database structure change script is generated which must be executed in order for the deletion to take effect.

16.4   Triggers

Triggers are user-defined functions that are executed automatically after a CREATE, UPDATE, or DELETE statement on tables or views is executed. In SQL Server 2005, triggers can also be fired on database level.

If a trigger is defined for a particular table, view, or—with SQL Server 2005—database, the SQL statement(s) in the trigger definition are executed automatically before, after, or instead of the statement that activates the trigger.

You can use an existing trigger as a basis for creating a new trigger definition or delete a trigger from the database. The context menu in the Online Browser provides several options in this respect (see Generating SQL Statements).

16.5   User-defined Functions

DatabaseSpy supports user-defined functions and displays them in a separate folder on the schema/database level. For each function, a Parameters sub-folder is provided where the function's parameters and their data types are listed.

```
⊟ dbo
   ⊞ 📁 Tables
   ⊞ 📁 Views
   ⊟ 📁 Functions
      ⊟ 📁 User Functions
         ⊟ f(x) GetAge
            ⊟ 📁 Parameters
               ⊟ 📇 return
                  ⊟ 📁 Datatype
                     T  int
               ⊟ 📇 @name
                  ⊟ 📁 Datatype
                     T  varchar
```
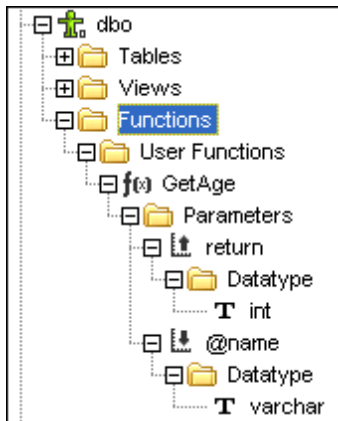
Functions can be added to the favorites. Some maintenance options applicable to functions are available through the context menu (see Generating SQL Statements). Please note that functions are not supported in Access databases.

Be sure to always add the schema prefix to the function name when you select a function.

16.6   Creating Packages

DatabaseSpy supports the definition of PL/SQL packages for Oracle databases. Packages are objects that group related PL/SQL types, items, and subtrees into modules and thus allow you to re-use code.

Packages consist of a package specification and an optional package body. The specification is the interface and the body defines cursors and subprograms.

Packages are defined with the commands:

CREATE [OR REPLACE] PACKAGE package_name

...

[CREATE [OR REPLACE] PACKAGE BODY package_name

...

16.7   Menus

The **Menus** section contains a complete description of all DatabaseSpy menu commands. We've tried to make this user manual as comprehensive as possible. If, however, you have questions which are not covered in the User Reference or other parts of this documentation, please look up the FAQs and Discussion Forums on the Altova website. If you are still not able to have your problem satisfactorily addressed, please do not hesitate to contact us through the Support Center on the Altova website.

Note that in the **File** and **Edit** menus, all standard Windows commands are supported, as well as additional database-related commands.

16.7.1   File Menu

The **File** menu contains all commands relevant to manipulating files, in the order common to most Windows software products.
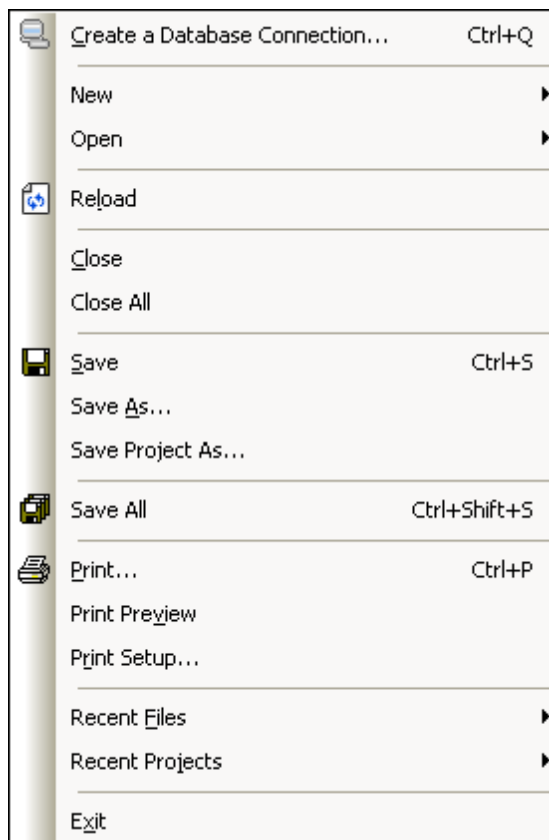
In addition to the standard Open, Save, Print, Print Setup, and Exit commands, DatabaseSpy offers a range of application-specific commands.

Create a Database Connection...

    Ctrl+Q

The **Create Database Connection...** command opens the **Add a Data Source** dialog box allowing you to choose between the generic Connection Wizard, creating ADO or ODBC

---

connections, and Global Resources.

In the General Options, you can specify whether or not the **Add a Data Source** dialog box should be displayed upon startup of the application.

New

The **New** sub-menu contains options for opening new projects, SQL Editor and Design Editor windows, data comparisons, and schema comparisons.

Project

 Ctrl+Shift+N
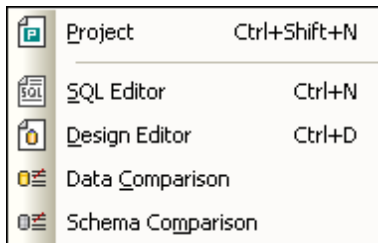
The **Project** command creates a new project in DatabaseSpy. If you are currently working with another project, a prompt appears, asking if you want to close all documents belonging to the current project.

SQL Editor

 Ctrl+N

The **SQL Editor** command opens a new SQL Editor window.

Design Editor

 Ctrl+D

The **Design Editor** command opens a new Design Editor window. Please note that you need an active database connection in order to open a Design Editor window.

Data Comparison

The **Data Comparison** command opens a new Data Comparison window and the **Select Database Objects for Comparison** dialog box, where you can select tables from two databases that you want to compare. Note that you can disable the display of the selection dialog box in the Data Compare options.

When no data source exists in your project, DatabaseSpy displays a warning message which allows you to add a data source immediately.
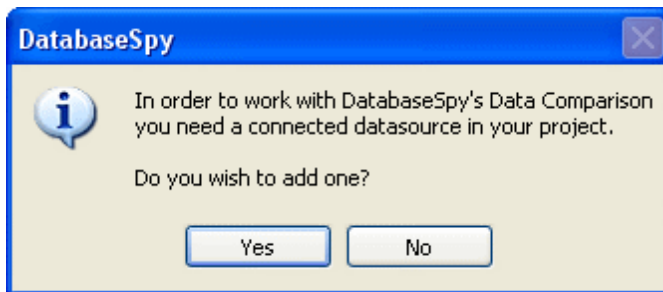
Click **Yes** in the message box to open the **Create a Database Connection** dialog box.

Also make sure that at least one data source is connected before you select the command to avoid the following message.
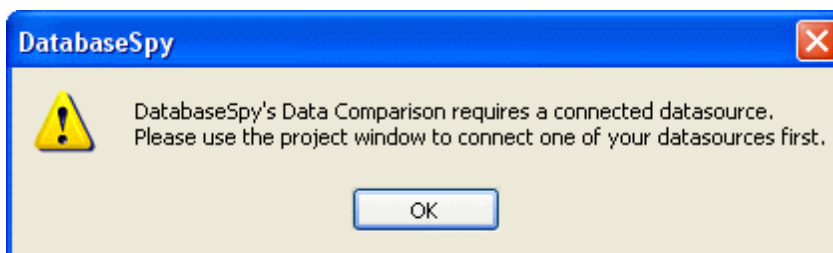
Schema Comparison

The **Schema Comparison** command opens a new Schema Comparison window and the **Select Database Objects for Comparison** dialog box, where you can select tables from two database schemas that you want to compare. Note that you can disable the display of the selection dialog box in the Data Compare options.

When no data source exists in your project, DatabaseSpy displays a warning message which allows you to add a data source immediately.
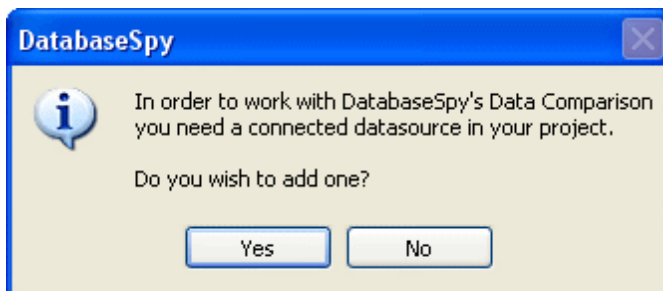
Click **Yes** in the message box to open the **Create a Database Connection** dialog box.

Also make sure that at least one data source is connected before you select the command to avoid the following message.
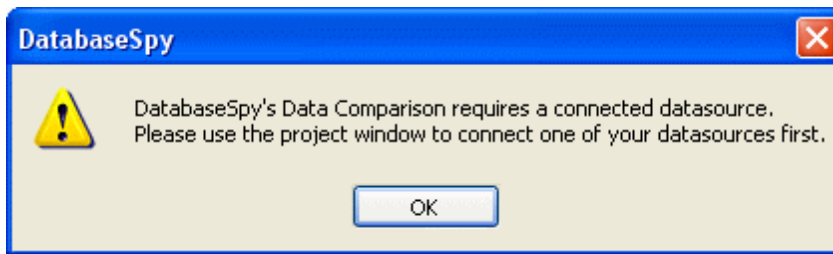
Open

The **Open** sub-menu provides options for opening projects, files, and Altova Global Resources.
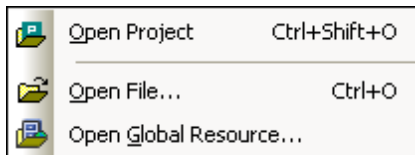
Open Project...

   Ctrl+Shift+O

The **Open Project...** command opens an existing project in DatabaseSpy. The standard Windows **Open** dialog box appears where you can choose a project file (*.qprj). If you are currently working with another project, the previous project is closed first.

Open File...

   Ctrl+O

The **Open...** command pops up the familiar Windows **Open** dialog box and allows you to open any project file (*.qprj), SQL file (*.sql), design file (*.qdes), database data comparison file (*.dbdif), or database schema comparison file (*.dbsdif). In the **Open** dialog box, you can select more than one file to open. Use the `Files of Type` combo box to restrict the kind of files displayed in the dialog box.

Open Global Resource...

The **Open Global Resource...** command opens the **Choose Global Resource** dialog box where you can select a file or folder and display it in the SQL Editor. The **Open Global Resources dialog...** button allows you to display the **Manage Global Resources** dialog box where you can make changes to a resource prior to opening it.
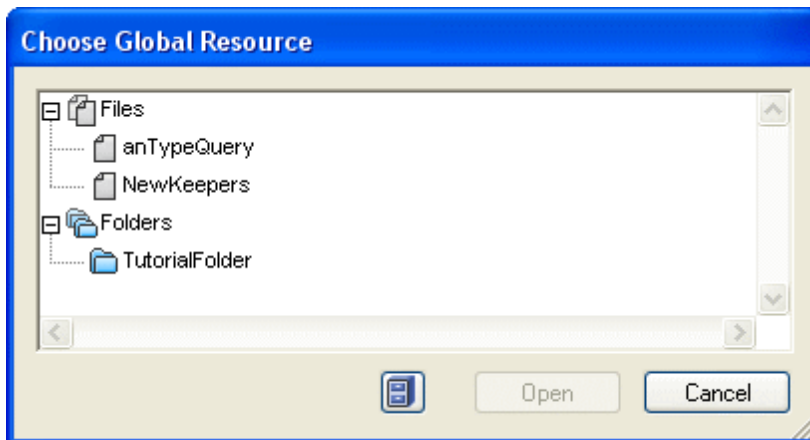
If you select a folder that has been defined as a global resource, then the standard Windows **Open** dialog box appears and displays the content of the directory that is defined for the active configuration.

Reload

The **Reload** command reloads the version of the active file that is saved in your file system. Any changes that you made in DatabaseSpy will be lost. You can still keep the updated version by clicking **No** in the message box that pops up.
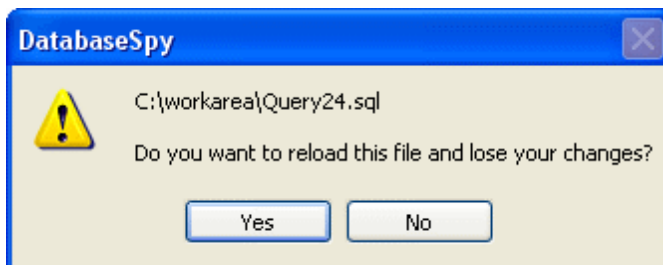
Close

The **Close** command closes the active window. If the file was modified (indicated by an asterisk ***** after the file name in the title bar), you will be asked if you wish to save the file first.

Close All

The **Close All** command closes all open windows. If any document has been modified (indicated by an asterisk ***** after the file name in the title bar), you will be asked if you wish to save the file first.

Please note that the project will not be closed by the **Close All** command. A project can only be closed by opening a new project or by exiting DatabaseSpy.

Save

    Ctrl+S

The **Save** command saves the contents of the active window to the file from which it has been

opened. If the active window contains a new file, the **Save As...** dialog box is opened automatically.

Save As...

The **Save As...** command pops up the familiar Windows **Save As** dialog box, in which you enter the name and location of the file you wish to save the active file as.

Save Project As...

The **Save Project As...** command saves the project that is currently open under a new name or at a different path. The standard Windows **Save As** dialog box appears where you can enter the desired file name and path.

Save All

> **Ctrl+Shift+S**

The **Save All** command saves all modifications that have been made to any open documents. The command is useful if you edit multiple documents simultaneously. If a document has not been saved before (for example, after being newly created), the **Save As...** dialog box is presented for that document.

Print...

🖨   **Ctrl+P**

The **Print...** command opens the Print dialog box, in which you can select printer options and print the active file.

Please note that the **Print...** command behaves differently depending on whether it is called from an SQL Editor window or Design Editor window:

- Choosing the **Print...** command from an **SQL Editor** window brings up the standard windows **Print** dialog box.
- If you select the **Print...** command for a design file in the **Design Editor**, DatabaseSpy presents a **Print** dialog box where you can choose which part of the design file is to be printed and adapt the zoom. Choosing the **Print** button brings up the standard Windows **Print** dialog box. Click the **Preview** button to open the preview window and check the layout before printing or change the printer setup by clicking the **Print Setup** button.
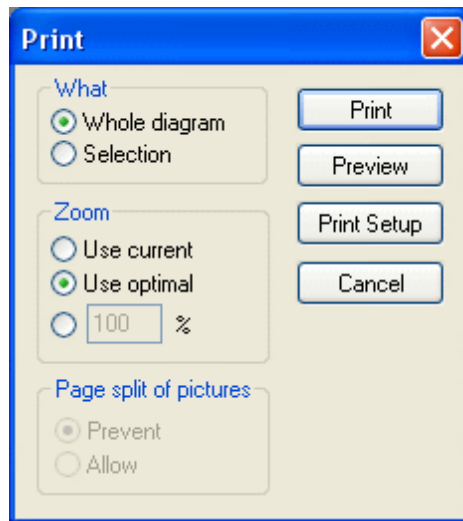
Print Preview

The **Print Preview** command opens a preview window where you can check the layout of your active window before printing.

Please note that the **Print Preview** command behaves differently depending on whether it is called from an SQL Editor window or Design Editor window:

- If you select **File | Print Preview** from an active **SQL Editor** window, a preview window opens immediately where you can browse the pages and zoom in or zoom out using the appropriate buttons.
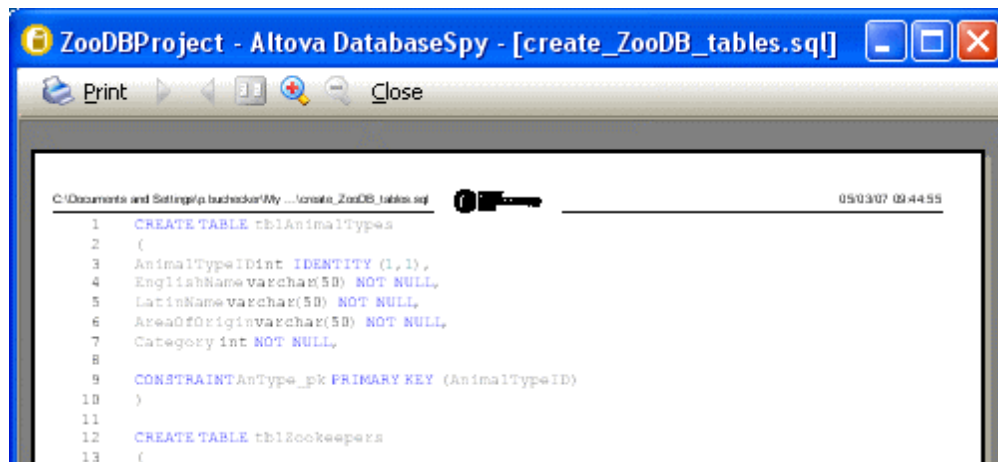
- When called from an active **Design Editor** window, the **Print Preview** command will first open the **Print** dialog box for design files. Click the **Preview** button to open the preview window where you can browse the pages and zoom in or zoom out using the appropriate buttons. If you click the **Close** button, the **Print** dialog box reappears and you can either print the file or cancel the operation.
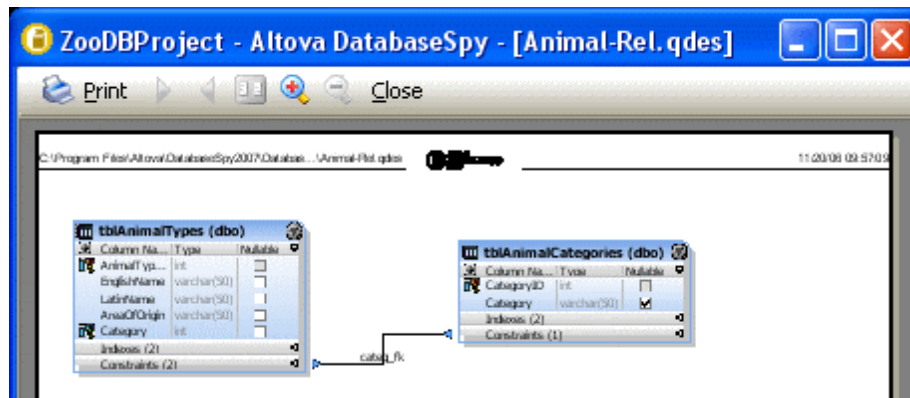
Print Setup...

The **Print Setup...** command displays the printer-specific **Print Setup** dialog box, in which you specify such printer settings as paper format and page orientation. These settings are applied to all subsequent print jobs.

Recent Files

The **Recent Files** command in the **File** menu opens a sub-menu displaying a list of the four most recently used files, with the most recently opened file shown at the top of the list. You can open any of these files by clicking its name. To open a file in the list using the keyboard, press **Alt+F** to open the File menu, and then press the number of the file you want to open.

Recent Projects

The **Recent Projects** command opens a submenu displaying the file name for the four most recently used projects, allowing quick access to these files.

Also note, that DatabaseSpy can automatically open the last project that you used, whenever you start DatabaseSpy. (**Tools | Options | General** | Open last project on startup).

Exit

The **Exit** command is used to quit DatabaseSpy. If you have any open files with unsaved changes, you are prompted to save these changes. DatabaseSpy also saves modifications to program settings and information about the most recently used files.

16.7.2 Edit Menu

The **Edit** menu contains commands for editing text in the SQL Editor window.

↰	<u>U</u>ndo	Alt+Backspace
↱	<u>R</u>edo	Ctrl+Y
✂	Cu<u>t</u>	Shift+Delete
▤	<u>C</u>opy	Ctrl+C
▤	<u>P</u>aste	Ctrl+V
	Select All	Ctrl+A
	<u>F</u>ind…	Ctrl+F
	Find <u>n</u>ext	F3
	R<u>e</u>place…	Ctrl+H

Different options are provided if the active window is a Design Editor.

↰	<u>U</u>ndo	Alt+Backspace
↱	<u>R</u>edo	Ctrl+Y
✂	Cu<u>t</u>	Shift+Delete
▤	<u>C</u>opy	Ctrl+C
▤	<u>P</u>aste	Ctrl+V
▥	Select Us<u>e</u>r Tables	Alt+T
▦	Select <u>S</u>ystem Tables	Alt+S
	Select <u>A</u>ll	Ctrl+A

Undo

   Alt+Backspace

The **Undo** command contains support for unlimited levels of Undo. Every action can be undone and it is possible to undo one command after another. The Undo history is retained after using the **Save** command, enabling you go back to the state the document was in before you saved your changes.

Redo

   Ctrl+Y

The **Redo** command allows you to redo previously undone commands, thereby giving you a complete history of work completed. You can step back and forward through this history using the **Undo** and **Redo** commands.

Cut

   Shift+Delete

The **Cut** command copies the selected text to the clipboard and deletes it from its present

location.

Copy

        Ctrl+C

The **Copy** command copies the selected text to the clipboard. This can be used to duplicate data within DatabaseSpy or to move data to another application.

Paste

        Ctrl+V

The **Paste** command inserts the contents of the clipboard at the current cursor position.

Select User Tables

        Alt+T

The **Select User Tables** command selects all user tables that are contained in the active Design Editor window.

Select System Tables

        Alt+S

The **Select System Tables** command selects all system tables that are contained in the active Design Editor window.

Select All

Ctrl+A

The **Select All** command selects (i) any text in the currently active SQL Editor window or (ii) all tables in the currently active Design Editor window.

Find...

Ctrl+F

The **Find** command pops up the **Find** dialog, in which you can specify the string you want to find and other options for the search.

Find Next

F3

The **Find next** command repeats the last **Find...** command to search for the next occurrence of the requested text.

Replace...

Ctrl+H

The **Replace** command enables you to find and replace one text string with another text string. It features the same options as the **Find...** command. You can replace each item individually or you can use the **Replace All** button to perform a global search-and-replace operation.

16.7.3 View Menu

The **View** menu contains commands for switching the various windows on or off.
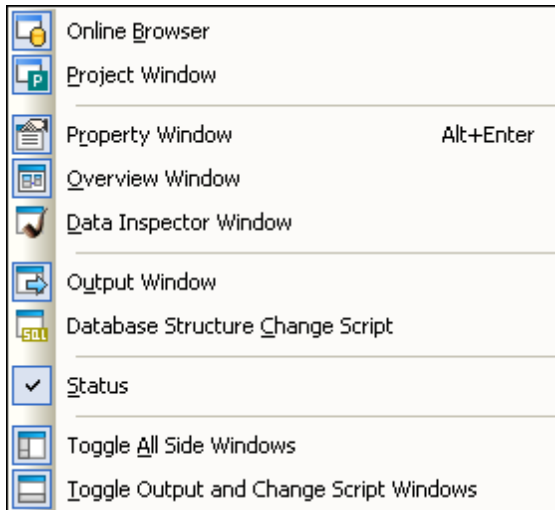
Online Browser

This command lets you switch the Online Browser on or off.

This is a dockable window. Dragging on its title bar detaches it from its current position and makes it a floating window. Click right on the title bar to allow docking or hide the window.

Project Window

This command lets you switch the Project window on or off.

This is a dockable window. Dragging on its title bar detaches it from its current position and makes it a floating window. Click right on the title bar to allow docking or hide the window.

Property Window

This command lets you switch the Property window on or off.

This is a dockable window. Dragging on its title bar detaches it from its current position and makes it a floating window. Click right on the title bar to allow docking or hide the window.

Overview Window

This command lets you switch the Overview window on or off.

This is a dockable window. Dragging on its title bar detaches it from its current position and

makes it a floating window. Click right on the title bar to allow docking or hide the window.

Data Inspector Window

This command lets you switch the <u>Data Inspector window</u> on or off.

This is a dockable window. Dragging on its title bar detaches it from its current position and makes it a floating window. Click right on the title bar to allow docking or hide the window.

Output Window

This command lets you switch the <u>Output window</u> on or off.

This is a dockable window. Dragging on its title bar detaches it from its current position and makes it a floating window. Click right on the title bar to allow docking or hide the window.

Database Structure Change Script

This command lets you switch the <u>Database Structure Change Script window</u> on or off.

This is a dockable window. Dragging on its title bar detaches it from its current position and makes it a floating window. Click right on the title bar to allow docking or hide the window.

Status

The **Status** command toggles on and off the Status bar.

Toggle All Side Windows

This command toggles on and off the display of the Online Browser and the Project, Properties, and Overview windows as well as any other window that has been docked to the side of the application. Note that this command applies also to windows that are docked on the right side of DatabaseSpy.

Toggle Output and Change Script Windows

This command toggles the display of the Output and Change Script windows on and off.

16.7.4   SQL Editor Menu

The **SQL Editor** menu contains commands for selecting, editing, and executing SQL statements.
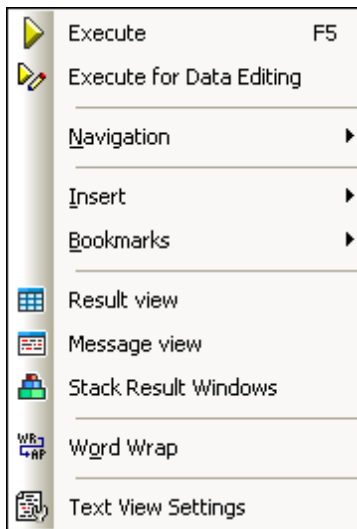
The **SQL Editor** menu is only visible if the active window contains an SQL file.

Execute

    F5

The **Execute** command executes all the SQL statements that are currently in the active window of the SQL Editor. You can customize the behaviour of the Execute command in Properties window of the active SQL Editor options.

The **Stop Retrieval** command stops the retrieval of database data. The data that has been retrieved so far is displayed in the Result tab. Use the **Retrieve next n rows** button or the **Retrieve outstanding rows** button, respectively, or the context menu in the Result view to resume retrieval. This button replaces the **Execute** button while data is being retrieved.

Execute for Data Editing

The **Execute for Data Editing** command executes a SELECT statement in the Online Browser and switches the Result window into the Editing Mode so that you are able to update or even add new data directly in the result grid. This command is only available if data editing from within the SQL Editor is supported for the data source connection you are currently working with.

Please note: This command is supported only for SELECT statements in the SQL Editor.

Navigation

The **Navigation** sub-menu contains options for selecting statements in the active SQL Editor window.

Next Statement

Alt+Down

The **Next Statement** command selects the statement following the statement in which the cursor is currently located.

Previous Statement

Alt+Up

The **Previous Statement** command selects the statement preceding the statement in which the cursor is currently located.

Last Statement

Alt+End

The **Last Statement** command selects the last statement in active window of the SQL Editor.

First Statement

Alt+Home

The **First Statement** command selects the first statement in the active window of the SQL Editor.

Select Entire Statement

Shift+Alt+Enter

The **Select Entire Statement** command selects the SQL statement in which the cursor is currently located.

Insert

The **Insert** sub-menu contains options for inserting block and line comments, targets, and regions into the active SQL Editor window.

The **Insert/Remove Block Comment** and **Insert/Remove Line Comment** options are toggle commands that can also be used to remove a previously inserted comment.

Insert/Remove Block Comment

The **Insert/Remove Block comment** inserts or removes a [block comment](#) around the currently selected text block in the active window of the SQL Editor.

Insert/Remove Line Comment

The **Insert/Remove line comment** command inserts [line comment](#) from the current position of the cursor to the end of the line or removes a selected line comment.

Insert Target

The **Insert Target** command inserts `--Target: Result` before the currently marked SELECT statement. Executing the statement places the result in a tab named "Result". The text "Result" can be edited, allowing you to create your [own named tabs](#).

Please note that in the Properties window of the active SQL Editor either "Semicolons" or "SQL Grammar" must be selected in the `Group statements for execution with` drop-down list in order to support the `target` keyword.

Insert Region

The **Insert Region** command creates a [region](#) for the currently selected block of text. Regions can be folded or expanded by clicking the **+** icon.

Bookmarks

The **Bookmarks** sub-menu provides commands for inserting and removing bookmarks, and allows you to navigate between the bookmarks in an SQL Editor window.
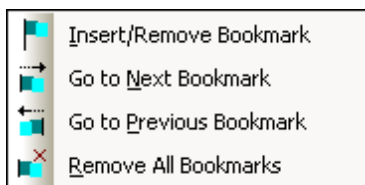
Insert/Remove Bookmark

The **Insert/Remove Bookmark** command inserts a bookmark at the line of the current cursor position, or removes the bookmark if the cursor is in a line that has been bookmarked previously.

Bookmarked lines are displayed in one of the following ways:
- If the [bookmarks margin](#) has been enabled, then a solid cyan ellipse appears to the left of the text in the bookmark margin.

- If the bookmarks margin has not been enabled, then the complete line containing the cursor is highlighted.

Go to Next Bookmark

The **Go to Next Bookmark** command places the text cursor at the beginning of the next bookmarked line.

Go to Previous Bookmark

The **Go to Previous Bookmark** command places the text cursor at the beginning of the previous bookmarked line.

Remove All Bookmarks

The **Remove All Bookmarks** command removes all the defined bookmarks from the SQL Editor window that is currently active.

Please note: The **Undo** command does not undo the effects of this command.

Result View

The **Result view** command toggles all Result tabs in SQL Editor windows on and off.

Message View

The **Message view** command toggles the Message tab in SQL Editor windows on and off.

Stack Result Windows

If the **Stack Result Windows** option is deactivated, results of queries are displayed in separate, sequentially numbered tabs of the Result window (Result1, Result2, etc.). If the option is activated, all results are displayed in a single result tab with multiple panes for the individual queries.

Show Groupings For Execution

The **Show groupings for execution** command toggles the display of execution groups for SQL statements on and off.

Word Wrap

The **Word Wrap** command enables or disables word wrapping in the SQL Editor.

Text View Settings

The **Text View Settings** command opens the **Text View Settings** dialog box where you can define settings for the display of line number, bookmark, and source folding margins, specify the tab size, and define which visual aids are available in the SQL Editor.

16.7.5   SQL Refactoring Menu

The **SQL Refactoring** menu contains commands for changing the layout of SQL code in the SQL Editor.
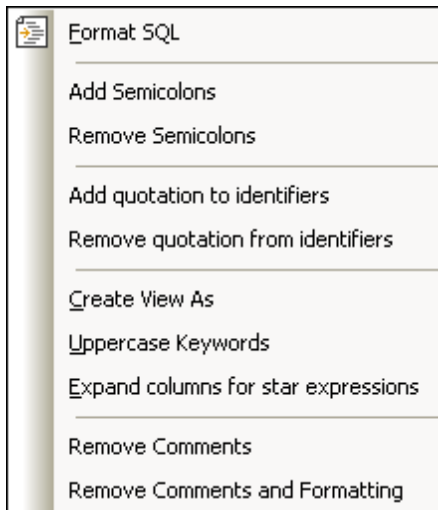
The **SQL Refactoring** menu is only visible if the active window contains an SQL file.

Format SQL

The **Format SQL** command applies SQL formatting as currently defined for the respective database kind to the active SQL Editor Window.

A message is displayed in the Message tab, after the formatting has been completed.
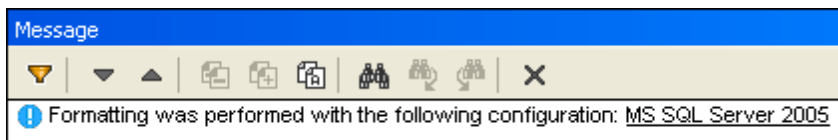
Add Semicolons

The **Add Semicolons** command appends a semicolon to the end of each statement. No semicolon will be added however if it already exists in a statement.

Remove Semicolons

The **Remove Semicolons** command removes all semicolons from the end of statements. It makes no difference whether the semicolons have been entered manually or using the **Add Semicolons** command.

Add Quotations to Identifiers

This command adds database-specific quotations (e.g., ", ' [], etc.) to the table and column identifiers in the SQL Editor.

Remove Quotations from Identifiers

This command removes database-specific quotations (e.g., ", ' [], etc.) from the table and column identifiers in the SQL Editor. It makes no difference whether the identifiers have been entered manually or using the **Add Quotations to Identifiers** command.

Create View As

The **Create View As** command creates a view from a SELECT statement in the active SQL Editor window. A view is a stored query.

Uppercase Keywords

The **Uppercase Keywords** converts all keywords in the active SQL Editor into uppercase.

Expand Columns for Star Expressions

The **Expand columns for star expressions** command inserts the full column list of a table if the cursor is placed after the asterisk in a SELECT statement and the command is executed.

Note that the SQL statement must be complete and valid. If, for example, the `SQL Editor should require semicolons` option is checked, the terminating semicolon must indeed be present in the SQL Editor.

Remove Comments

This command removes all text that is enclosed in comment marks (`*/.../*` and `--`). Please note that the **Remove Comments** command removes the text that has been comment out and **not** the command marks! To remove the comment marks, use the **Insert/Remove Block Comment** and **Insert/Remove Line Comment** commands.

Remove Comments and Formatting

The **Remove Comments and Formatting** command reverts all formatting that has been added by the **Format SQL** command, that is, any additional line feed or whitespace is removed and the SQL is displayed in one line. In addition, this command removes all text that has been commented out.

16.7.6 Design Editor Menu

The **Design Editor** menu contains commands for maintaining database objects in a graphical manner.
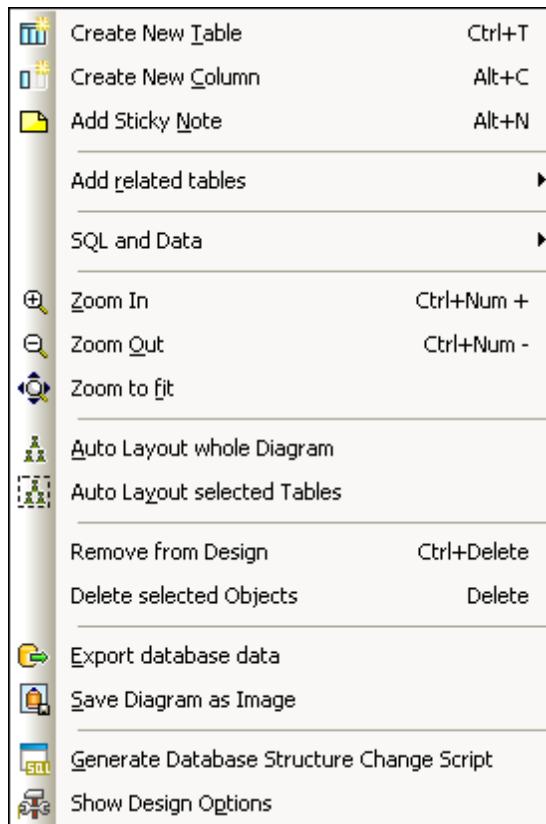
The **Design Editor** menu is only visible if the active window contains a design file.

Create New Table

    Ctrl+T

The **Create New Table** command creates a new table in the currently active Design Editor window and automatically assigns it to the active data source. A Database Structure Change Script is generated and has to be executed to add the newly created table to the database.

Create New Column

    Alt+C

The **Create New Column** command adds a new column to the selected table. Please note that if you have selected more than one table, the column will be added to the table that was selected last.

Add Sticky Note

    Alt+N

The **Add Sticky Note** command adds a Sticky Note to the Design Editor window. Double-click the note to enter text and drag it to the desired position.

Add Related Tables

The **Add Related Tables** command opens a sub-menu which adds tables that are either referenced by or are referencing the selected table.

🖅	Referenced Tables
🖅	Referencing Tables
🖅	All Relations
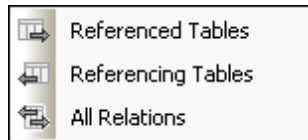
Referenced Tables

🖅

The **Referenced Tables** command adds tables that are referenced by the selected table to the active Design Editor window.

Referencing Tables

🖅

The **Referencing Tables** command adds tables which include a reference to the selected table to the active Design Editor window.

All Relations

🖅

The **All Relations** command adds both referenced and referencing tables to the active Design Editor window.

SQL and Data

The **SQL and Data** command opens a sub-menu which provides options for generating SQL scripts, retrieving, and editing data from within the design view.

	Show in new SQL Editor	▶
	Retrieve data	▶
⊘	Edit Data	Alt+E
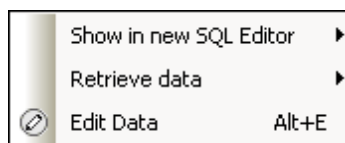
Show in new SQL Editor

The **Show in new SQL Editor** command opens a sub-menu which enables you to create SQL statements from the selected tables in the Design Editor.
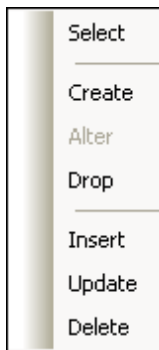
If more than one table is selected, a statement is created for each of the selected tables. Please note that different statements are created depending on what you have selected:

- If you have selected the **entire table**, one statement is created for all of the columns in the table.

```
SELECT [AnimalTypeID], [EnglishName], [LatinName],
[AreaOfOrigin], [Category] FROM [ZooDB].[dbo].[tblAnimalTypes];
```

- If you have selected a **single column**, the statement is created for this particular column only.

```
SELECT [EnglishName] FROM [ZooDB].[dbo].[tblAnimalTypes];
```

- If you have selected **some but not all of the columns**, one statement is created for the selected columns.

```
SELECT [EnglishName], [AreaOfOrigin] FROM
[ZooDB].[dbo].[tblAnimalTypes];
```

Retrieve Data

The **Retrieve Data** command opens a sub-menu which enables you to perform a query on the selected table in the Design Editor.
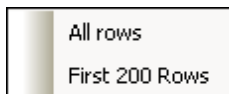
All rows

The **All rows** command generates a SELECT statement for all rows in a new SQL Editor window and executes the query immediately. The result of the query is displayed in the Result tab below the SQL Editor window.

First n rows

The **First n rows** command generates a SELECT statement for the first n rows in a new SQL Editor window and executes the query immediately. The result of the query is displayed in the Result tab below the SQL Editor window.

You can customize the number of rows to be retrieved in the **SQL Editor options**.

Edit Data

⊘   **Ctrl+Alt+E**

The **Edit Data** command retrieves data from the selected table or columns and switches the result grid into the editing mode (if supported for the respective data source).

Zoom In

🔍⊕   **Ctrl+Num +**

The **Zoom In** command increases the zoom factor of the Design Editor window by 10 percent. You can also zoom in by scrolling (with the scroll-wheel of the mouse) while keeping the **Ctrl** key pressed.

Zoom Out

🔍⊖   **Ctrl+Num -**

The **Zoom Out** command decreases the zoom factor of the Design Editor window by 10 percent. You can also zoom out by scrolling (with the scroll-wheel of the mouse) while keeping the **Ctrl** key pressed.

Zoom to Fit

The **Zoom to Fit** command sizes the working area to include all tables in the current Design Editor window.

Auto Layout Whole Diagram

The **Auto Layout Whole Diagram** command arranges all tables in the active Design Editor window in a way that considers relations between the individual tables as well as proper layouting.

Auto Layout Selected Tables

The **Auto Layout Selected Tables** command arranges the selected tables in the active Design Editor window in a way that considers relations between the individual tables as well as a proper layouting.

Remove from Design

Ctrl+Del

The **Remove from Design** command removes the selected table from the Design Editor window. Please note that the table is not deleted from the database in this case. To drop a table from the database use the **Delete Selected Objects** command instead.

Delete Selected Objects

Del

The **Delete Selected Objects** command generates a Database Structure Change Script and marks the selected objects for deletion. You must execute the change script in order to actually delete the objects from the database.

You can select multiple objects for deletion. If you select one or more database objects of a table as well as the table itself and press the **Delete** key, DatabaseSpy prompts you to choose whether you want to delete the entire table or only its objects.

Export Database Data

The **Export Database Data** command opens the **Export database data** dialog box where you can define the export settings. If you have tables selected in the Design Editor window, the check boxes of these tables will already be activated in the Source group box of the **Export database data** dialog box and thus selected for export.

Save Diagram As Image

The **Save Diagram As Image** command opens the standard Windows **Save As** dialog box where you can specify a name and location for the design image. The designs can be stored as Portable Network Graphics (*.png) or Enhanced Meta Files (*.emf).

Generate Database Structure Change Script

The **Generate Database Structure Change Script** command starts the generation of a change script or updates it after changes occurred to the database structure. Please note that this menu option is not available if in the General options you have activated the `Instantly` radio button in the Database Structure Change Script group box.

Show Design Options

The **Show Design Options** command opens the Design Editor page of the Options dialog box where you can customize the settings for the Design Editor.

16.7.7   **Data Comparison Menu**

The **Data Comparison** menu contains all the commands that are related to the comparison of database data.
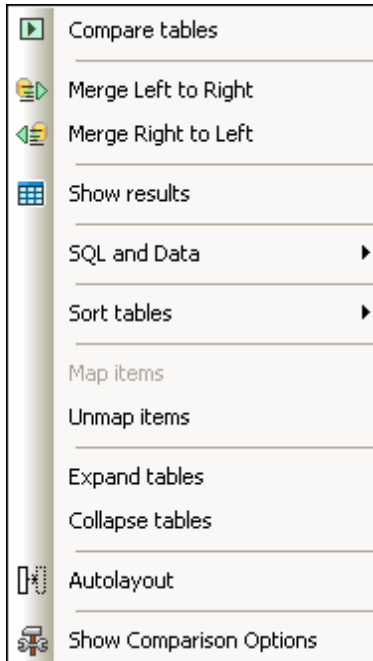
The **Data Comparison** menu is only visible if the active window contains a data comparison file.

Compare Tables

The **Compare Tables** command starts a comparison of the tables that you have selected in the Data Comparison window.

Merge Left to Right

The **Merge Left to Right** command updates the tables in the right comparison component with the data contained in the tables in the left component. Missing table rows are added and obsolete rows are deleted. The required SQL script is generated and executed in the background. DatabaseSpy displays a warning message which has to be confirmed before the changes are actually committed to the database.

Merge Right to Left

The **Merge Right to Left** command updates the tables in the left comparison component with the data contained in the tables in the right component. Missing table rows are added and obsolete rows are deleted. The required SQL script is generated and executed in the background. DatabaseSpy displays a warning message which has to be confirmed before the changes are actually committed to the database.

Show Results

The **Show Results** command displays the Comparison Result window for the selected table(s) or for all mapped tables in the active Data Comparison window, if no table is selected.

SQL and Data

The **SQL and Data** sub-menu of the **Data Comparison** menu provides options for showing the merge and restore scripts for both sides of a comparison.
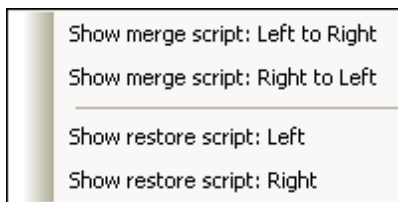
The commands of the **SQL and Data** sub-menu are only available after a comparison has been started.

Show merge script: Left to Right

The **Show merge script: Left to Right** command opens a new SQL Editor window and displays an SQL change script that overwrites the data in the table(s) contained in the right component with the values of the mapped table(s) of the left component.

Show merge script: Right to Left

The **Show merge script: Right to Left** command opens a new SQL Editor window and displays an SQL change script that overwrites the data in the table(s) contained in the left component with the values of the mapped table(s) of the right component.

Show restore script: Left

The **Show restore script: Left** command reverts the changes that have been made to the left table and restores the condition the left table had before the merge script was executed.

Show restore script: Right

The **Show restore script: Right** command reverts the changes that have been made to the right table and restores the condition the right table had before the merge script was executed.

Sort Tables

The **Sort Tables** sub-menu of the **Data Comparison** menu provides options for sorting the tables that are included in the components of a database data comparison.

Please note that the commands always affect both sides of the comparison (i.e., you cannot sort

just one component).

Ascending

The **Ascending** command sorts all tables in a comparison window in ascending order.

Descending

The **Descending** command sorts all tables in a comparison window in descending order.

Ascending mapped first

The **Ascending mapped first** command sorts all tables in a Data Comparison window and displays the mapped tables in ascending order first (*see screenshot*).

Descending mapped first

The **Descending mapped first** command sorts all tables in a Data Comparison window and displays the mapped tables in descending order first (*see screenshot*).

Map items

The **Map items** command tries to map all tables in the Data Comparison window using the settings defined in the Data Compare section of the **Options** dialog box.

Unmap items

The **Unmap items** command deletes all mappings in the active comparison window. If one or more tables are selected in the components, only the mapping of the selected tables will be deleted.

Expand tables

The **Expand tables** command expands all tables in both components of the active comparison window.

Collapse tables

The **Collapse tables** command collapses all tables in both components of the active comparison window.

Autolayout

The **Autolayout** command fits the components into the visible part of the comparison window, tries to avoid horizontal scrollbars in the components, and aligns the top and bottom borders of the components.

Show Comparison Options

The **Show Comparison Options** command opens the Data Compare section of the **Options** dialog page.

16.7.8   Schema Comparison Menu

The **Schema Comparison** menu contains all the commands that are related to the comparison of database schemas.
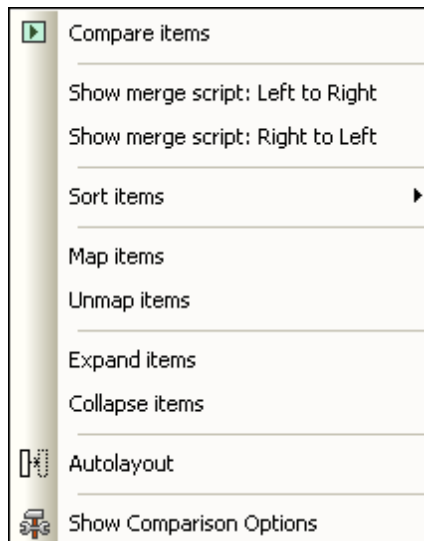
The **Schema Comparison** menu is only visible if the active window contains a schema comparison file.

Compare items

The **Compare items** command starts a comparison of the tables that you have selected in the Schema Comparison window.

Show merge script: Left to Right

The **Show merge script: Left to Right** command opens a new SQL Editor window and displays an SQL change script that overwrites the table structure(s) contained in the right component with those of the mapped table(s) of the left component.

Please note that only selected items are included in the merge script and that child items are not

automatically considered in the merge script when their parents are selected. If you want to merge both parent and children you must select all items.

Consider the following example: Two versions of a database are compared where one column is misspelled and the data type of that column has been changed on purpose between the two versions. If child entries would be automatically merged if a parent entry is selected for merging, you would not be able to correct the type in the column name without setting the data type back to its old setting.

Show merge script: Right to Left

The **Show merge script: Right to Left** command opens a new SQL Editor window and displays an SQL change script that overwrites the table structure(s) contained in the left component with those of the mapped table(s) of the right component.

Please note that only selected items are included in the merge script and that child items are not automatically considered in the merge script when their parents are selected. If you want to merge both parent and children you must select all items.

Consider the following example: Two versions of a database are compared where one column is misspelled and the data type of that column has been changed on purpose between the two versions. If child entries would be automatically merged if a parent entry is selected for merging, you would not be able to correct the type in the column name without setting the data type back to its old setting.

Sort Items

The **Sort items** sub-menu of the **Schema Comparison** menu provides options for sorting the tables that are included in the components of a database schema comparison.

Please note that the commands always affect both sides of the comparison (i.e., you cannot sort just one component).

Ascending

The **Ascending** command sorts all tables in a comparison window in ascending order.

Descending

The **Descending** command sorts all tables in a comparison window in descending order.

Ascending mapped first

The **Ascending mapped first** command sorts all tables in a Schema Comparison window and displays the mapped tables in ascending order first (*see screenshot*).

Descending mapped first

The **Descending mapped first** command sorts all tables in a Schema Comparison window and displays the mapped tables in descending order first (*see screenshot*).

Map items

The **Map items** command tries to map all tables in the Schema Comparison window using the settings defined in the [Data Compare] section of the **Options** dialog box.

Unmap Items

The **Unmap items** command deletes all mappings in the active comparison window. If one or more tables are selected in the components, only the mapping of the selected tables will be deleted.

Expand items

The **Expand items** command expands all items in both components of the active comparison window.

Collapse items

The **Collapse items** command collapses all items in both components of the active comparison window.

Autolayout

The **Autolayout** command fits the components into the visible part of the comparison window, tries to avoid horizontal scrollbars in the components, and aligns the top and bottom borders of

the components.

Show Comparison Options

The **Show Comparison Options** command opens the Data Compare section of the **Options** dialog page.

16.7.9 Tools Menu

The **Tools** menu allows you to:
- Export and import database data
- Generate SQL change scripts
- Manage XML schemas for databases
- Access user-defined tools
- Manage Global Resources and configurations
- Customize menus, toolbars, etc.
- Restore toolbars and windows to their default state
- Define the global program settings
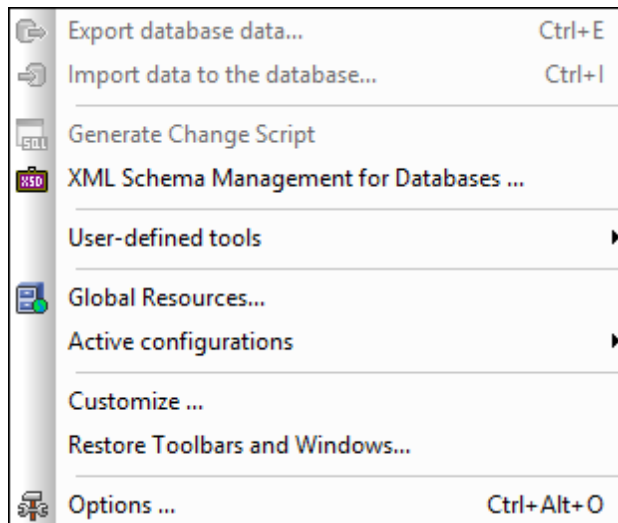
Export Database Data...

   Ctrl+E

The **Export Database Data...** command opens the Export Database Data dialog box where you can define your export options.

Import Data to the Database...

   Ctrl+I

The **Import Data to Database...** command opens the Import data to the database dialog box where you can define your import options.

Generate Change Script

The **Generate Change Script** command starts the generation of a change script or updates it

after changes occurred to the database structure. Please note that this menu option is not available if in the General options you have activated the `Instantly` radio button in the Database Structure Change Script group box.

XML Schema Management for Databases...

The **XML Schema Management for Databases** command opens the **XML Schema management for databases** dialog box which allows you to select a data source connection and add, drop, modify, or view XML schemas for the active data source.

User-defined Tools

The **User-defined Tools** command opens a sub-menu where you can add your personal menu items, e.g., links to other applications.

To add a user-defined tool, select the menu option **Tools | Customize** or right-click into the toolbar and select **Customize** from the context menu. Change to the Tools tab and add name of and path to the EXE file of the application you want to start from within DatabaseSpy. The name you entered in the Menu contents box appears in the **Tools | User-defined Tools** sub-menu.

Global Resources...

The **Global Resources...** command opens the **Manage Global Resources** dialog box, where you can add, edit, or delete global resources.

Active Configurations

The **Active configurations** command opens a sub-menu, where all available configurations are listed and the active configuration is indicated. Selecting a configuration in the sub-menu will make it the active configuration.

Customize...

The **Customize...** command lets you customize DatabaseSpy to suit your personal needs.

Commands

The **Commands** tab allows you customize your menus or toolbars.
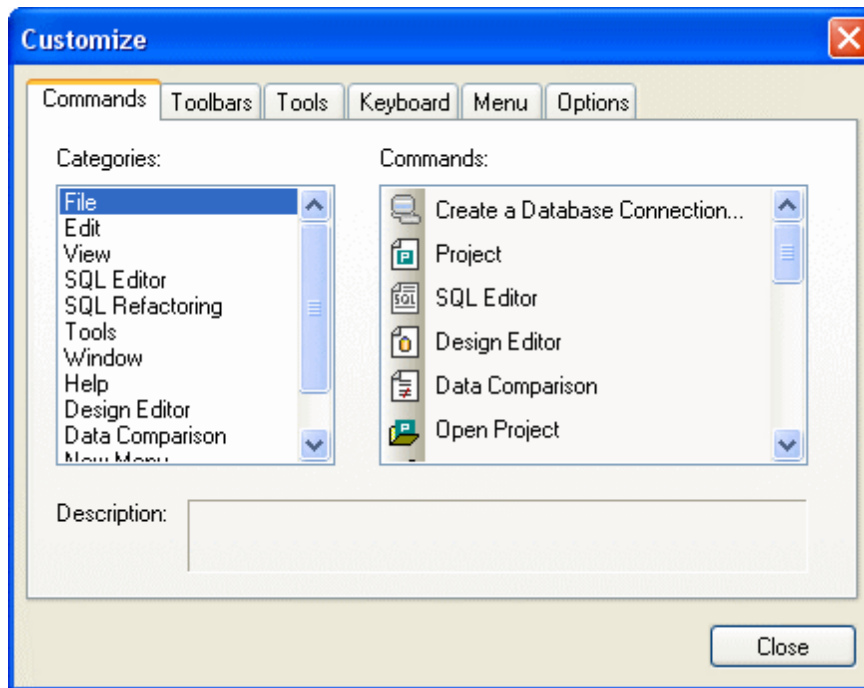
To add a command to a toolbar or menu:

1. Select the menu option **Tools | Customize**.
   The **Customize** dialog box appears.

2. Select the `All Commands` category in the Categories list box.
   The available commands appear in the Commands list box.

3. Click on a command in the Commands list box and drag it to an existing menu or toolbar.
   An **I**-beam appears when you place the cursor over a valid position to drop the command.

4. Release the mouse button at the position you want to insert the command.

   • A small button appears at the tip of mouse pointer when you drag a command. The "x" below the pointer means that the command cannot be dropped at the current cursor position.
   • The "x" disappears whenever you can drop the command (over a tool bar or menu).
   • Placing the cursor over a menu when dragging opens it, allowing you to insert the command anywhere in the menu.
   • Commands can be placed in menus or toolbars. If you created you own toolbar you can populate it with your own commands/icons.

Please note: You can also edit the commands in the context menus (right-click anywhere to open the context menu), using the same method. Click the Menu tab and then select the specific context menu available in the Context Menus combo box.

Toolbars

The **Toolbars** tab allows you to activate or deactivate specific toolbars, as well as create your own specialized ones.

DatabaseSpy toolbars contain symbols for the most frequently used menu commands. For each symbol you get a brief "tool tip" explanation when the mouse cursor is directly over the item and the status bar shows a more detailed description of the command.

You can drag the toolbars from their standard position to any location on the screen, where they appear as a floating window. Alternatively, you can also dock them to the left or right edge of the main window.
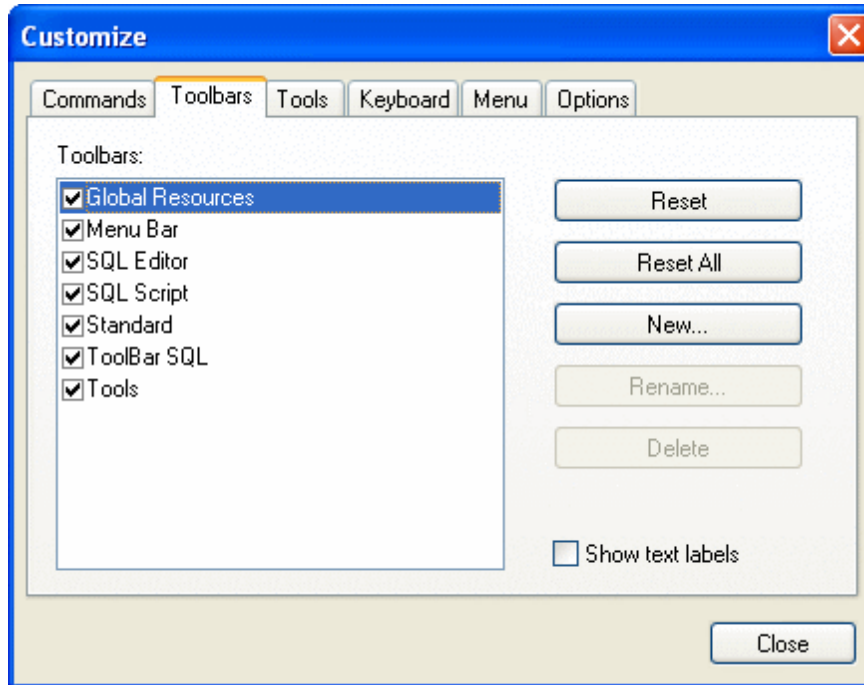
Show text labels:
This option displays explanatory text below toolbar icons when activated. You can activate or deactivate this option for each toolbar individually.

To activate or deactivate a toolbar:

- Click the check box to activate (or deactivate) the specific toolbar.

To create a new toolbar:

1. Click the **New...** button, and give the toolbar a name in the **Toolbar Name** dialog box that appears.
2. Drag commands to the toolbar in the Commands tab of the **Customize** dialog box.

To reset the Menu Bar:

1. Click the Menu Bar entry.
2. Click the **Reset** button, to reset the menu commands to the state they were in when DatabaseSpy was installed.

To reset all toolbar and menu commands:

1. Click the **Reset All** button to reset all the toolbar commands to the state they were when the program was installed.
   A prompt appears stating that all toolbars and menus will be reset.
2. Click **Yes** to confirm the reset.

To change a toolbar name:

- Click the **Rename...** button to edit the name of the toolbar.

This option is available only for user-defined toolbars.

To delete a toolbar:

1. Select the toolbar you want to delete in the Toolbars list box.

2. Click the **Delete** button.
   A prompt appears, asking if you really want to delete the toolbar.

3. Click **Yes** to confirm the deletion.

This option is available only for user-defined toolbars.

Tools

The **Tools** tab allows you to add your personal menu items, e.g., links to other applications, to the **Tools** menu.
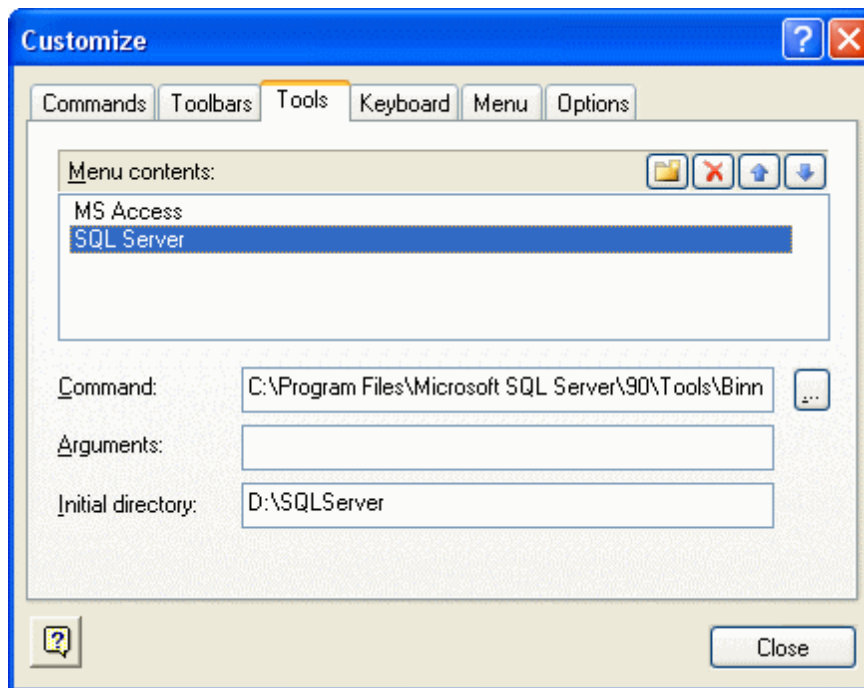
Any user-defined entries will appear in the **User-defined tools** sub-menu.

To add a user-defined tool:

1. Click the **New** icon in the Menu contents title bar or press the **Insert** key.

2. In the text field that appears, enter the string that is to appear as menu entry in the **Tools** menu.

3. In the `Command` field enter the path to the EXE file of the application or use the browse ⬚ button to select the path.

4. If applicable, enter arguments needed to start the application in the `Arguments` field.

5. Enter the path to the initial directory, if required.

To change the sequence of user-defined tools:

*   Select an entry and use the **Move Item Up** ⬆ and **Move Item Down** ⬇ buttons to move the menu item.
    Alternatively, you can also use the shortcut keys **Alt+Up** and **Alt+Down** to move the entry.

To rename a user-defined menu item:

*   Double-click an item in the `Menu contents` field and edit the entry.

To remove a user-defined tool:

*   Select the entry you want to remove from the Tools menu and click the **Delete** ❌ icon in the Menu contents title bar or press the **Delete** key.

Keyboard

The **Keyboard** tab allows you to define (or change) keyboard shortcuts for any DatabaseSpy command.
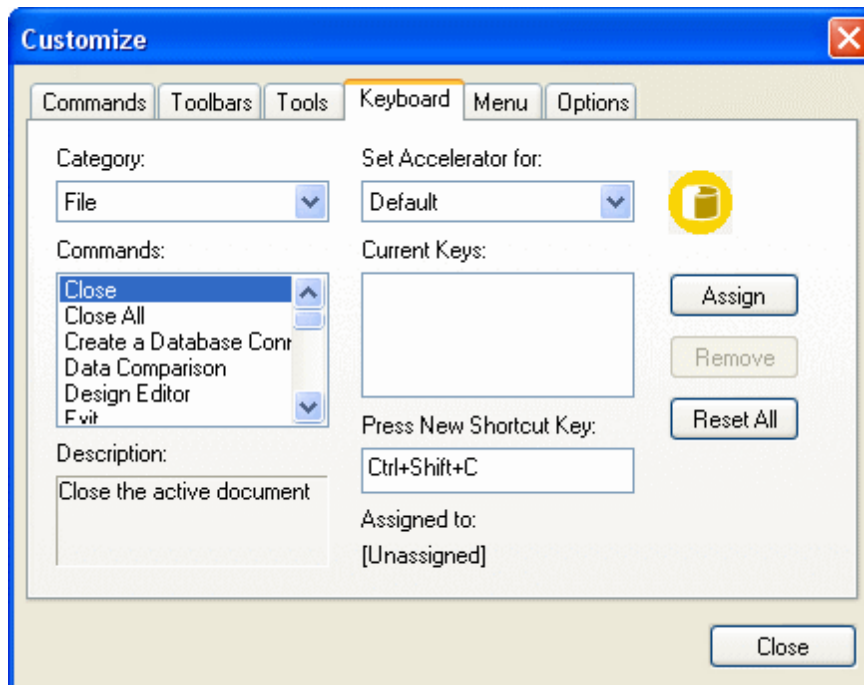
In the `Set accelerator for` drop-down list, you can differentiate between shortcuts that are valid in the Default menu or in the DatabaseSpy Design menu.

To assign a new Shortcut to a command:

1. Select the All Commands category using the `Category` combo box.

2. Select the command you want to assign a new shortcut to, in the `Commands` list box.

3. Click in the `Press New Shortcut Key` text box, and press the shortcut keys that are to activate the command.
    The shortcuts appear immediately in the text box. If the shortcut was assigned previously, then that function is displayed below the text box.

4. Click the **Assign** button to assign the shortcut.
    The shortcut now appears in the `Current Keys` list box.
    (To clear this text box, press any of the control keys, **CTRL**, **ALT** or **SHIFT**).

To de-assign or delete a shortcut:

1. Click the shortcut you want to delete in the `Current Keys` list box.

2. Click the **Remove** button.

3. Click the **Close** button to confirm.

To reset all shortcut keys:

1. Click the **Reset All** button to reset all the shortcut keys to the state they were when the program was installed.
    A prompt appears stating that all toolbars and menus will be reset.

2. Click **Yes** to confirm the reset.

Currently assigned keyboard shortcuts

Hotkeys by key

F1	Table of Contents
F3	Find next
F5	Execute
Alt	Show Paths
Alt+Enter	Property Window
Shift+Alt+Enter	Select entire Statement
Shift+Delete	Cut
Alt+Backspace	Undo
Ctrl+Insert	Copy
Shift+Insert	Paste
Alt+Down	Next Statement
Alt+Up	Previous Statement
Alt+Home	First Statement
Alt+End	Last Statement
Alt+Insert	Append new row
Ctrl+C	Copy/Copy selected cells
Ctrl+D	New Design Editor
Ctrl+E	Export database data...
Ctrl+F	Find...
Ctrl+H	Replace...
Ctrl+I	Import data to the database...

Ctrl+N	New SQL Editor
Ctrl+O	Open...
Ctrl+P	Print...
Ctrl+Q	Create a database connection...
Ctrl+S	Save
Ctrl+S	Save Project
Ctrl+V	Paste
Ctrl+X	Cut
Ctrl+Y	Redo
Ctrl+Z	Undo
Ctrl+Alt+E	Edit data
Ctrl+Alt+O	Options...
Ctrl+Alt+R	Retrieve all rows
Ctrl+Alt+T	Retrieve first n rows
Ctrl+Shift+N	New Project
Ctrl+Shift+O	Open Project
Ctrl+Shift+S	Save All
Ctrl+Shift+V	Paste as new row

Hotkeys by function

Append new row	Alt+Insert	
Copy	Ctrl+C	Ctrl+Insert
Copy selected cells	Ctrl+C	
Cut	Ctrl+X	Shift+Delete
Edit data	Ctrl+Alt+E	
Execute	F5	
Export database data...	Ctrl+E	
Find next	F3	
Find...	Ctrl+F	
First Statement	Alt+Home	
Import data to the database	Ctrl+I	
Last Statement	Alt+End	
New Design Editor	Ctrl+D	
New Project	Ctrl+Shift+N	
New SQL Editor	Ctrl+N	
Next Statement	Alt+Down	
Open...	Ctrl+O	
Open Project	Ctrl+Shift+O	
Options...	Ctrl+Alt+O	
Paste	Ctrl+V	Shift+Insert
Paste as new row	Ctrl+Shift+V	
Previous Statement	Alt+Up	
Print...	Ctrl+P	
Property Window	Alt+Enter	
Create a database connection	Ctrl+Q	
Redo	Ctrl+Y	
Replace...	Ctrl+H	
Retrieve all rows	Ctrl+Alt+R	
Retrieve first n rows	Ctrl+Alt+T	
Save	Ctrl+S	
Save All	Ctrl+Shift+S	
Save Project	Ctrl+S	
Select entire Statement	Shift+Alt+Enter	

Show Paths	Alt	
Table of Contents	F1	
Undo	Ctrl+Z	Alt+Backspace

Menu

The **Menu** tab allows you to customize the main menu bars as well as the context menus.
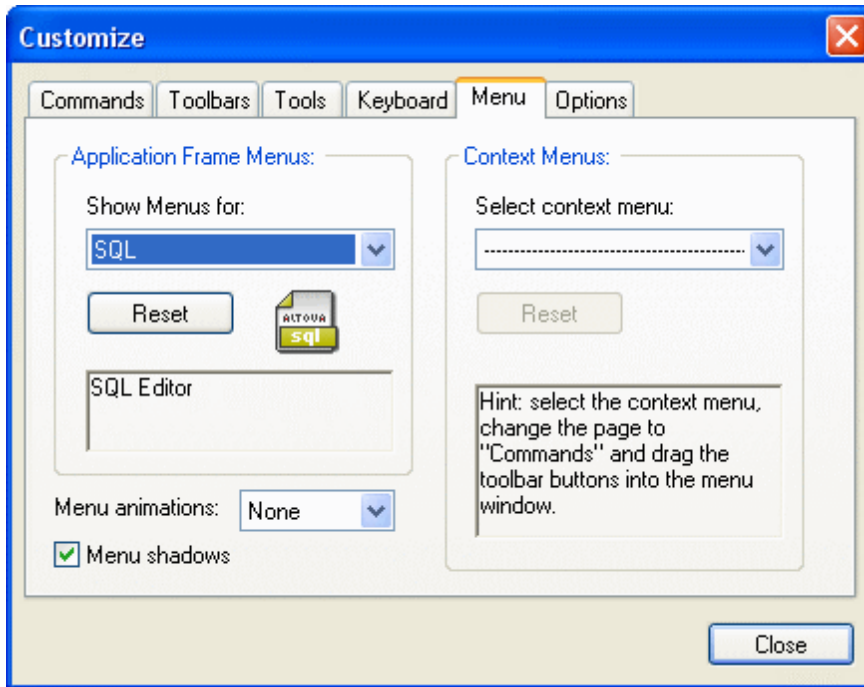
You can customize the Default, the SQL, the Design, the Database Data Comparison, and the Database Schema Comparison menu bars.

The Default menu is the one visible when no SQL Editor or Design Editor or Data Comparison windows are open in DatabaseSpy.
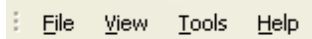
The SQL menu is the menu bar that is displayed when at least one SQL Editor window has been opened.
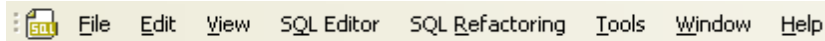
The Design menu is the menu bar visible when at least one Design Editor window has been opened.

The Database Data Comparison menu bar is visible when at least one Data Comparison window has been opened.
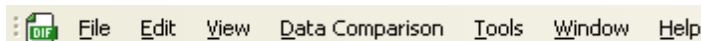
The Database Schema Comparison menu bar is visible when at least one Schema Comparison

window has been opened.

Menu animations and menu shadows

You can choose from among several menu animations if you prefer animated menus. The `Menu animations` drop-down box provides the following options:

- None (default)
- Unfold
- Slide
- Fade

The `Menu shadows` check box, which is checked by default, can be deactivated if you do not want all your menus to have shadows.

To customize a menu:

1. Select the menu bar you want to customize from the `Show Menus for` combo box.

2. Click the Commands tab, and drag the commands to the menu bar of your choice.

To delete commands from a menu:

1. Select the menu option **Tools | Customize** to open the **Customize** dialog box.

2. Do one of the following:

   - Right-click the command or icon representing the command and select the **Delete** option from the context menu.

   - Drag the command away from the menu, and drop it as soon as the check mark icon appears below the mouse pointer.

To reset either of the menu bars:

1. Select either the Default, the SQL, the Design, or the Database Data Comparison entry in the `Show Menus for` combo box.

2. Click the **Reset** button just below the menu name.
   A prompt appears asking if you are sure you want to reset the menu bar.

3. Click **Yes**.

To customize any of the context menus (right-click menus):

1. Select the context menu from the `Select context menu` combo box.
   The context menu you selected appears.

2. Click the Commands tab, and drag the commands to the context menu.

To delete commands from a context menu:

1. Select the menu option **Tools | Customize** to open the **Customize** dialog box.

2. Do one of the following:

   - Right-click on the command or icon representing the command and select the **Delete** option from the context menu.

   - Drag the command away from the context menu, and drop it as soon as the check mark icon appears below the mouse pointer.

To reset any of the context menus:

1. Select the context menu from the combo box.

2. Click the **Reset** button just below the context menu name.
    A prompt appears asking if you are sure you want to reset the context menu.

To close a context menu window:

Do one of the following:

*   Click on the **Close icon** at the top right of the title bar.

*   Click the **Close** button of the **Customize** dialog box.

To change the appearance of menus:

1. If you want animated menus, select the appropriate option from the `Menu animations` drop-down box.

2. If required, deactivate the `Menu shadows` check box.

Options

The Options tab allows you to set general environment settings.
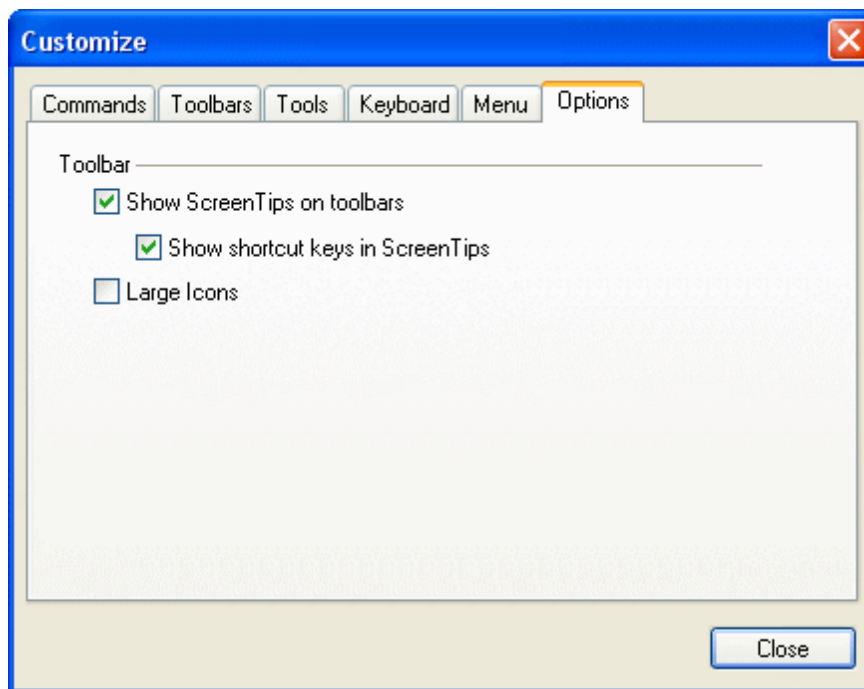
Toolbar

When active, the `Show ScreenTips on toolbars` check box displays a popup when the mouse pointer is placed over an icon in any of the icon bars. The popup contains a short description of the icon function, as well as the associated keyboard shortcut, if one has been assigned.

The `Show shortcut keys in ScreenTips` check box allows you to decide whether or not you want to have the shortcut displayed in the tooltip.

When active, the `Large icons` check box switches between the standard size icons, and larger versions of the icons.

Options...

　　Ctrl+Alt+O

The **Options** command opens the **Options** dialog box where you can set the General options as well as specific settings for SQL or Design Editor or Data Comparison windows.

16.7.10　Window Menu

To organize the individual windows in an DatabaseSpy session, the **Window** menu contains standard commands common to most Windows applications. You can also or switch to an open document window directly from the menu.
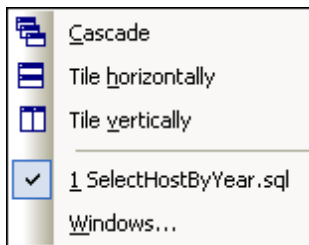
You can cascade the open windows, tile them, or arrange document icons once you have minimized them.

Cascade

The **Cascade** command rearranges all open document windows so that they are all cascaded (i.e., staggered) on top of each other.

Tile horizontally

The **Tile horizontally** command rearranges all open document windows as **horizontal tiles**, making them all visible at the same time.

Tile vertically

The **Tile vertically** command rearranges all open document windows as **vertical tiles**, making them all visible at the same time.

Currently open window list

This list shows all currently open windows and lets you quickly switch between them.

You can also use the **Ctrl+TAB** or **Ctrl+F6** keyboard shortcuts to cycle through the open windows.

Windows...

At the bottom of the list of open windows is the **Windows...** command. Clicking this entry opens the **Windows** dialog box, which displays a list of all open windows and provides commands that can be applied to the selected window(s). A window is selected by clicking on its name.

Warning: To exit the **Windows** dialog box, click **OK**; do **not** click the **Close Window(s)** button in the dialog box. The **Close Window(s)** button closes the window(s) currently selected in the **Windows** dialog box.

16.7.11 Help Menu

▼ **Table of Contents**

☐ *Description*

Opens the onscreen help manual of DatabaseSpy with the Table of Contents displayed in the left-hand-side pane of the Help window. The Table of Contents provides an overview of the entire Help document. Clicking an entry in the Table of Contents takes you to that topic.

▼ **Index**

☐ *Description*

Opens the onscreen help manual of DatabaseSpy with the Keyword Index displayed in the left-hand-side pane of the Help window. The index lists keywords and lets you navigate to a topic by double-clicking the keyword. If a keyword is linked to more than one topic, a list of these topics is displayed.

▼ **Search**

☐ *Description*

Opens the onscreen help manual of DatabaseSpy with the Search dialog displayed in the left-hand-side pane of the Help window. To search for a term, enter the term in the input field, and press **Return**. The Help system performs a full-text search on the entire Help documentation and returns a list of hits. Double-click any item to display that item.

▼ **Software Activation**

☐ *Description*

After you download your Altova product software, you can activate it using either a free evaluation key or a purchased permanent license key.

- *Free evaluation key.* When you first start the software after downloading and installing it, the Software Activation dialog will pop up. In it is a button to request a free evaluation key-code. Enter your name, company, and e-mail address in the dialog that appears, and click Request Now! The evaluation key is sent to the e-mail address you entered and should reach you in a few minutes. Now enter the key in the key-code field of the Software Activation dialog box and click **OK** to start working with your Altova product. The software will be unlocked for a period of 30 days.
- *Permanent license key.* The Software Activation dialog contains a button to purchase a permanent license key. Clicking this button takes you to Altova's online shop, where you can purchase a permanent license key for your product. There are two types of permanent license: single-user and multi-user. Both will be sent to you by e-mail. A *single-user license* contains your license-data and includes your name, company, e-mail, and key-code. A *multi-user license* contains your license-data and includes your company name and key-code. Note that your license agreement does not allow you to install more than the licensed number of copies of your Altova software on the computers in your organization (per-seat license). Please make sure that you enter the data required in the registration dialog exactly as given in your license e-mail.

Note: When you enter your license information in the Software Activation dialog, ensure that you enter the data exactly as given in your license e-mail. For multi-user licenses, each user should enter his or her own name in the Name field.

The Software Activation dialog (*screenshot below*) can be accessed at any time by clicking the **Help | Software Activation** command.

You can activate the software by either:

- Entering the license key information (click **Enter a New Key Code**), or
- Acquiring a license via an Altova LicenseServer on your network (click **Use Altova LicenseServer**, located at the bottom of the Software Activation dialog). The Altova LicenseServer must have a license for your Altova product in its license pool. If a license is available in the LicenseServer pool, this is indicated in the Software Activation dialog (*screenshot below*), and you can click **Save** to acquire the license.
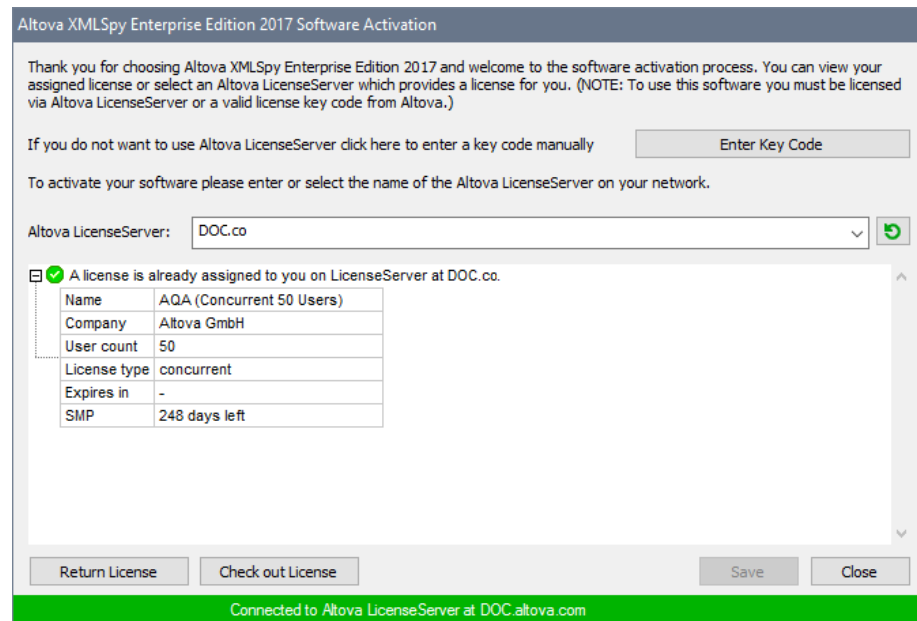
Note that, once a license has been acquired, it cannot be returned to the LicenseServer for a period of seven days. After that time, you can return the license (click **Return License**) so that the license can be acquired by another client. A LicenseServer administrator, however, can unassign an acquired license via the administrator's Web UI of LicenseServer at any time.

Check out license
You can check out a license from the license pool for a period of up to 30 days so that the license is stored on the product machine. This enables you to work offline, which is useful, for example, if you wish to work in an environment where there is no access to your Altova LicenseServer (such as when your Altova product is installed on a laptop and you are traveling). While the license is checked out, LicenseServer displays the license as being in use, and the license cannot be used by any other machine. The license automatically reverts to the checked-in state when the check-out period ends. Alternatively, a checked-out license can be checked in at any time via the **Check in** button of the Software Activation dialog.

To check out a license, do the following: (i) In the Software Activation dialog, click **Check out License** (*see screenshot above*); (ii) In the License Check-out dialog that appears, select the check-out period you want and click **Check out**. The license will be checked out. The Software Activation dialog will display the check-out information, including the time when the check-out period ends. The **Check out License** button in the dialog changes to a **Check In** button. You can check the license in again at any time by clicking **Check In**. Because the license automatically reverts to the checked-in status, make sure that the check-out period you select adequately covers the period during which you will be working offline.

Note: For license check-outs to be possible, it must be enabled on the LicenseServer. If this functionality has not been enabled, you will get an error

message to this effect. In this event, contact your LicenseServer administrator.

Altova LicenseServer provides IT administrators with a real-time overview of all Altova licenses on a network, together with the details of each license, as well as client assignments and client usage of licenses. The advantage of using LicenseServer therefore lies in administrative features it offers for large-volume Altova license management. Altova LicenseServer is available free of cost from the Altova website. For more information about Altova LicenseServer and licensing via Altova LicenseServer, see the Altova LicenseServer documentation.

▼ Order Form

▭ *Description*

When you are ready to order a licensed version of the software product, you can use either the **Order license key** button in the Software Activation dialog (*see previous section*) or the **Help | Order Form** command to proceed to the secure Altova Online Shop.

▼ Registration

▭ *Description*

Opens the Altova Product Registration page in a tab of your browser. Registering your Altova software will help ensure that you are always kept up to date with the latest product information.

▼ Check for Updates

▭ *Description*

Checks with the Altova server whether a newer version than yours is currently available and displays a message accordingly.

▼ Support Center

▭ *Description*

A link to the Altova Support Center on the Internet. The Support Center provides FAQs, discussion forums where problems are discussed, and access to Altova's technical support staff.

▾ FAQ on the Web

⊟ *Description*

A link to Altova's FAQ database on the Internet. The FAQ database is constantly updated as Altova support staff encounter new issues raised by customers.

▾ Download Components and Free Tools

⊟ *Description*

A link to Altova's Component Download Center on the Internet. From here you can download a variety of companion software to use with Altova products. Such software ranges from XSLT and XSL-FO processors to Application Server Platforms. The software available at the Component Download Center is typically free of charge.

▾ DatabaseSpy on the Internet

⊟ *Description*

A link to the Altova website on the Internet. You can learn more about DatabaseSpy and related technologies and products at the Altova website.

▾ About DatabaseSpy

⊟ *Description*

Displays the splash window and version number of your product. If you are using the 64-bit version of DatabaseSpy, this is indicated with the suffix (x64) after the application name. There is no suffix for the 32-bit version.

Chapter 17

License Information

17 License Information

This section contains:

- Information about the distribution of this software product
- Information about software activation and license metering
- Information about the intellectual property rights related to this software product
- The End-User License Agreement governing the use of this software product

Please read this information carefully. It is binding upon you since you agreed to these terms when you installed this software product.

17.1   Electronic Software Distribution

This product is available through electronic software distribution, a distribution method that provides the following unique benefits:

- You can evaluate the software free-of-charge before making a purchasing decision.
- Once you decide to buy the software, you can place your order online at the Altova website and immediately get a fully licensed product within minutes.
- When you place an online order, you always get the latest version of our software.
- The product package includes a comprehensive integrated onscreen help system. The latest version of the user manual is available at www.altova.com (i) in HTML format for online browsing, and (ii) in PDF format for download (and to print if you prefer to have the documentation on paper).

30-day evaluation period

After downloading this product, you can evaluate it for a period of up to 30 days free of charge. About 20 days into this evaluation period, the software will start to remind you that it has not yet been licensed. The reminder message will be displayed once each time you start the application. If you would like to continue using the program after the 30-day evaluation period, you have to purchase an Altova Software License Agreement, which is delivered in the form of a key-code that you enter into the Software Activation dialog to unlock the product. You can purchase your license at the online shop at the Altova website.

Helping Others within Your Organization to Evaluate the Software

If you wish to distribute the evaluation version within your company network, or if you plan to use it on a PC that is not connected to the Internet, you may only distribute the Setup programs, provided that they are not modified in any way. Any person that accesses the software installer that you have provided, must request their own 30-day evaluation license key code and after expiration of their evaluation period, must also purchase a license in order to be able to continue using the product.

For further details, please refer to the Altova Software License Agreement at the end of this section.

17.2   Software Activation and License Metering

As part of Altova's Software Activation, the software may use your internal network and Internet connection for the purpose of transmitting license-related data at the time of installation, registration, use, or update to an Altova-operated license server and validating the authenticity of the license-related data in order to protect Altova against unlicensed or illegal use of the software and to improve customer service. Activation is based on the exchange of license related data such as operating system, IP address, date/time, software version, and computer name, along with other information between your computer and an Altova license server.

Your Altova product has a built-in license metering module that further helps you avoid any unintentional violation of the End User License Agreement. Your product is licensed either as a single-user or multi-user installation, and the license-metering module makes sure that no more than the licensed number of users use the application concurrently.

This license-metering technology uses your local area network (LAN) to communicate between instances of the application running on different computers.

Single license

When the application starts up, as part of the license metering process, the software sends a short broadcast datagram to find any other instance of the product running on another computer in the same network segment. If it doesn't get any response, it will open a port for listening to other instances of the application.

Multi license

If more than one instance of the application is used within the same LAN, these instances will briefly communicate with each other on startup. These instances exchange key-codes in order to help you to better determine that the number of concurrent licenses purchased is not accidentally violated. This is the same kind of license metering technology that is common in the Unix world and with a number of database development tools. It allows Altova customers to purchase reasonably-priced concurrent-use multi-user licenses.

We have also designed the applications so that they send few and small network packets so as to not put a burden on your network. The TCP/IP ports (2799) used by your Altova product are officially registered with the IANA (*see* the IANA website (http://www.iana.org/) *for details*) and our license-metering module is tested and proven technology.

If you are using a firewall, you may notice communications on port 2799 between the computers that are running Altova products. You are, of course, free to block such traffic between different groups in your organization, as long as you can ensure by other means, that your license agreement is not violated.

You will also notice that, if you are online, your Altova product contains many useful functions; these are unrelated to the license-metering technology.

17.3 Intellectual Property Rights

The Altova Software and any copies that you are authorized by Altova to make are the intellectual property of and are owned by Altova and its suppliers. The structure, organization and code of the Software are the valuable trade secrets and confidential information of Altova and its suppliers. The Software is protected by copyright, including without limitation by United States Copyright Law, international treaty provisions and applicable laws in the country in which it is being used. Altova retains the ownership of all patents, copyrights, trade secrets, trademarks and other intellectual property rights pertaining to the Software, and that Altova's ownership rights extend to any images, photographs, animations, videos, audio, music, text and "applets" incorporated into the Software and all accompanying printed materials. Notifications of claimed copyright infringement should be sent to Altova's copyright agent as further provided on the Altova Web Site.

Altova software contains certain Third Party Software that is also protected by intellectual property laws, including without limitation applicable copyright laws as described in detail at http://www.altova.com/legal_3rdparty.html.

All other names or trademarks are the property of their respective owners.

17.4 Altova End User License Agreement

THIS IS A LEGAL DOCUMENT -- RETAIN FOR YOUR RECORDS

ALTOVA® END USER LICENSE AGREEMENT

Licensor:
Altova GmbH
Rudolfsplatz 13a/9
A-1010 Wien
Austria

Important - Read Carefully. Notice to User:

This End User License Agreement ("Agreement") is a legal document between you and Altova GmbH ("Altova"). It is important that you read this document before using the Altova-provided software ("Software") and any accompanying documentation, including, without limitation printed materials, 'online' files, or electronic documentation ("Documentation"). By clicking the "I accept" and "Next" buttons below, or by installing, or otherwise using the Software, you agree to be bound by the terms of this Agreement as well as the Altova Privacy Policy ("Privacy Policy") including, without limitation, the warranty disclaimers, limitation of liability, data use and termination provisions below, whether or not you decide to purchase the Software. You agree that this agreement is enforceable like any written agreement negotiated and signed by you. If you do not agree, you are not licensed to use the Software, and you must destroy any downloaded copies of the Software in your possession or control. You may print a copy of this Agreement as part of the installation process at the time of acceptance. Alternatively, a copy of this Agreement may be found at http://www.altova.com/eula and a copy of the Privacy Policy may be found at http://www.altova.com/privacy.

1. SOFTWARE LICENSE

(a)   License Grant.
         (i)   Upon your acceptance of this Agreement Altova grants you a non-exclusive, non-transferable (except as provided below), limited license, without the right to grant sublicenses, to install and use a copy of the Software on one compatible personal computer or workstation in the same local area network (LAN) up to the Permitted Number of computers. Subject to the limitations set forth in Section 1(c), you may install and use a copy of the Software on more than one of your compatible personal computers or workstations if you have purchased a Named-User license. Subject to the limitations set forth in Sections 1(d) and 1(e), users may use the software concurrently on a network. The Permitted Number of computers and/or users and the type of license, e.g. Installed, Named-Users, and Concurrent-User, shall be determined and specified at such time as you elect to purchase the Software. Installed user licenses are intended to be fixed and not concurrent. In other words, you cannot uninstall the Software on one machine in order to reinstall that license to a different machine and then uninstall and reinstall back to the original machine. Installations should be static. Notwithstanding the foregoing, permanent uninstallations and redeployments are acceptable in limited circumstances such as if an employee leaves the company or the machine is permanently decommissioned. During the evaluation period, hereinafter defined, only a single user may install and use the software on one (1) personal computer or workstation. If you have licensed the Software as part of a suite of Altova software products (collectively, the "Suite") and have not installed each product individually, then the Agreement governs your use of all of the software included in the Suite.

(ii)   If you have licensed SchemaAgent, then the terms and conditions of this Agreement apply to your use of the SchemaAgent server software ("SchemaAgent Server") included therein, as applicable, and you are licensed to use SchemaAgent Server solely in connection with your use of Altova Software and solely for the purposes described in the accompanying documentation.

(iii)   If you have licensed Software that enables users to generate source code, your license to install and use a copy of the Software as provided herein permits you to generate source code based on (i) Altova Library modules that are included in the Software (such generated code hereinafter referred to as the "Restricted Source Code") and (ii) schemas or mappings that you create or provide (such code as may be generated from your schema or mapping source materials hereinafter referred to as the "Unrestricted Source Code"). In addition to the rights granted herein, Altova grants you a non-exclusive, non-transferable, limited license to compile the complete generated code (comprised of the combination of the Restricted Source Code and the Unrestricted Source Code) into executable object code form, and to use, copy, distribute or license that executable. You may not distribute or redistribute, sublicense, sell, or transfer the Restricted Source Code to a third-party in the un-compiled form unless said third-party already has a license to the Restricted Source Code through their separate agreement with Altova. Notwithstanding anything to the contrary herein, you may not distribute, incorporate or combine with other software, or otherwise use the Altova Library modules or Restricted Source Code, or any Altova intellectual property embodied in or associated with the Altova Library modules or Restricted Source Code, in any manner that would subject the Restricted Source Code to the terms of a copyleft, free software or open source license that would require the Restricted Source Code or Altova Library modules source code to be disclosed in source code form. Notwithstanding anything to the contrary herein, you may not use the Software to develop and distribute other software programs that directly compete with any Altova software or service without prior written permission. Altova reserves all other rights in and to the Software. With respect to the feature(s) of UModel that permit reverse-engineering of your own source code or other source code that you have lawfully obtained, such use by you does not constitute a violation of this Agreement. Except as otherwise expressly permitted in Section 1(j) reverse engineering of the Software is strictly prohibited as further detailed therein.

(iv)   In the event Restricted Source Code is incorporated into executable object code form, you will include the following statement in (1) introductory splash screens, or if none, within one or more screens readily accessible by the end-user, and (2) in the electronic and/or hard copy documentation: "Portions of this program were developed using Altova® [name of Altova Software, e.g. MapForce® 2016] and includes libraries owned by Altova GmbH, Copyright © 2007-2016 Altova GmbH (www.altova.com)."

(b)   Server Use for Installation and Use of SchemaAgent. You may install one (1) copy of the Software on a computer file server within your internal network solely for the purpose of downloading and installing the Software onto other computers within your internal network up to the Permitted Number of computers in a commercial environment only. If you have licensed SchemaAgent, then you may install SchemaAgent Server on any server computer or workstation and use it in connection with your Software. No other network use is permitted, including without limitation using the Software either directly or through commands, data or instructions from or to a computer not part of your internal network, for Internet or Web-hosting services or by any user not licensed to use this copy of the Software through a valid license from Altova.

(c)   Named-Use. If you have licensed the "Named-User" version of the software, you may install the Software on up to five (5) compatible personal computers or workstations of which you are the primary user thereby allowing you to switch from one computer to the other as necessary provided that only one (1) instance of the Software will be used by you as the Named-User at any

given time. If you have purchased multiple Named-User licenses, each individual Named-User will receive a separate license key code.

(d)   Concurrent Use in Same Local Area Network (LAN).   If you have licensed a "Concurrent-User" version of the Software, you may install the Software on any compatible computers in a commercial environment only, up to ten (10) times the Permitted Number of users, provided that only the Permitted Number of users actually use the Software at the same time and further provided that the computers on which the Software is installed are on the same local area network (LAN). The Permitted Number of concurrent users shall be delineated at such time as you elect to purchase the Software licenses. Each separate local area network (LAN) requires its own set of separate Concurrent User Licenses for those wishing to use the Concurrent User versions of the Software in more than one location or on more than one network, all subject to the above Permitted Number limitations and based on the number of users using the Software. If a computer is not on the same local area network (LAN), then a locally installed user license or a license dedicated to concurrent use in a virtual environment is required.

(e)   Concurrent Use in Virtual Environment. If you have purchased Concurrent-User Licenses, you may install a copy of the Software on a single host terminal server (Microsoft Terminal Server or Citrix Metaframe), application virtualization server (Microsoft App-V, Citrix XenApp, or VMWare ThinApp) or virtual machine environment within your internal network for the sole and exclusive purpose of permitting individual users within your organization to access and use the Software through a terminal server, application virtualization session, or virtual machine environment from another computer provided that the total number of users that access or use the Software concurrently at any given point in time on such network, virtual machine or terminal server does not exceed the Permitted Number; and provided that the total number of users authorized to use the Software through the terminal server, application virtualization session, or virtual machine environment does not exceed ten (10) times the Permitted Number of users. Key codes for concurrent users cannot be deployed to more than one host terminal server, application virtualization server or virtual machine environment.   You must deploy a reliable and accurate means of preventing users from exceeding the Permitted Number of concurrent users. Altova makes no warranties or representations about the performance of Altova software in a terminal server, application virtualization session, or virtual machine environment and the foregoing are expressly excluded from the limited warranty in Section 5 hereof. Technical support is not available with respect to issues arising from use in such environments.

(f)   Backup and Archival Copies. You may make one (1) backup and one (1) archival copy of the Software, provided your backup and archival copies are not installed or used on any computer and further provided that all such copies shall bear the original and unmodified copyright, patent and other intellectual property markings that appear on or in the Software. You may not transfer the rights to a backup or archival copy unless you transfer all rights in the Software as provided under Section 3.

(g)   Key Codes, Upgrades and Updates.   Prior to your purchase and as part of the registration for the thirty (30) day evaluation period, as applicable, you will receive an evaluation key code. You will receive a purchase key code when you elect to purchase the Software from either Altova GmbH or an authorized reseller. The purchase key code will enable you to activate the Software beyond the initial evaluation period. You may not re-license, reproduce or distribute any key code except with the express written permission of Altova. If the Software that you have licensed is an upgrade or an update, then the latest update or upgrade that you download and install replaces all or part of the Software previously licensed. The update or upgrade and the associated license keys does not constitute the granting of a second license to the Software in that you may not use the upgrade or updated copy in addition to the copy of the Software that it is replacing and whose license has terminated.

(h)   Title. Title to the Software is not transferred to you. Ownership of all copies of the Software and of copies made by you is vested in Altova, subject to the rights of use granted to you in this Agreement. As between you and Altova, documents, files, stylesheets, generated program code (including the Unrestricted Source Code) and schemas that are authored or created by you via your utilization of the Software, in accordance with its Documentation and the terms of this Agreement, are your property unless they are created using Evaluation Software, as defined in Section 4 of this Agreement, in which case you have only a limited license to use any output that contains generated program code (including Unrestricted Source Code) such as Java, C++, C#, VB.NET or XSLT and associated project files and build scripts, as well as generated XML, XML Schemas, documentation, UML diagrams, and database structures only for the thirty (30) day evaluation period.

(i)   Reverse Engineering.   Except and to the limited extent as may be otherwise specifically provided by applicable law in the European Union, you may not reverse engineer, decompile, disassemble or otherwise attempt to discover the source code, underlying ideas, underlying user interface techniques or algorithms of the Software by any means whatsoever, directly or indirectly, or disclose any of the foregoing, except to the extent you may be expressly permitted to decompile under applicable law in the European Union, if it is essential to do so in order to achieve operability of the Software with another software program, and you have first requested Altova to provide the information necessary to achieve such operability and Altova has not made such information available. Altova has the right to impose reasonable conditions and to request a reasonable fee before providing such information. Any information supplied by Altova or obtained by you, as permitted hereunder, may only be used by you for the purpose described herein and may not be disclosed to any third party or used to create any software which is substantially similar to the expression of the Software. Requests for information from users in the European Union with respect to the above should be directed to the Altova Customer Support Department.

(j)   Other Restrictions. You may not loan, rent, lease, sublicense, distribute or otherwise transfer all or any portion of the Software to third parties except to the limited extent set forth in Section 3 or as otherwise expressly provided. You may not copy the Software except as expressly set forth above, and any copies that you are permitted to make pursuant to this Agreement must contain the same copyright, patent and other intellectual property markings that appear on or in the Software. You may not modify, adapt or translate the Software. You may not, directly or indirectly, encumber or suffer to exist any lien or security interest on the Software; knowingly take any action that would cause the Software to be placed in the public domain; or use the Software in any computer environment not specified in this Agreement. You may not permit any use of or access to the Software by any third party in connection with a commercial service offering, such as for a cloud-based or web-based SaaS offering.

You will comply with applicable law and Altova's instructions regarding the use of the Software. You agree to notify your employees and agents who may have access to the Software of the restrictions contained in this Agreement and to ensure their compliance with these restrictions.

(k)   NO GUARANTEE. THE SOFTWARE IS NEITHER GUARANTEED NOR WARRANTED TO BE ERROR-FREE NOR SHALL ANY LIABILITY BE ASSUMED BY ALTOVA IN THIS RESPECT. NOTWITHSTANDING ANY SUPPORT FOR ANY TECHNICAL STANDARD, THE SOFTWARE IS NOT INTENDED FOR USE IN OR IN CONNECTION WITH, WITHOUT LIMITATION, THE OPERATION OF NUCLEAR FACILITIES, AIRCRAFT NAVIGATION, COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL EQUIPMENT, MEDICAL DEVICES OR LIFE SUPPORT SYSTEMS, MEDICAL OR HEALTH CARE APPLICATIONS, OR OTHER

APPLICATIONS WHERE THE FAILURE OF THE SOFTWARE OR ERRORS IN DATA PROCESSING COULD LEAD TO DEATH, PERSONAL INJURY OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE. YOU AGREE THAT YOU ARE SOLELY RESPONSIBLE FOR THE ACCURACY AND ADEQUACY OF THE SOFTWARE AND ANY DATA GENERATED OR PROCESSED BY THE SOFTWARE FOR YOUR INTENDED USE AND YOU WILL DEFEND, INDEMNIFY AND HOLD ALTOVA, ITS OFFICERS AND EMPLOYEES HARMLESS FROM ANY THIRD PARTY CLAIMS, DEMANDS, OR SUITS THAT ARE BASED UPON THE ACCURACY AND ADEQUACY OF THE SOFTWARE IN YOUR USE OR ANY DATA GENERATED BY THE SOFTWARE IN YOUR USE.

2. INTELLECTUAL PROPERTY RIGHTS

You acknowledge that the Software and any copies that you are authorized by Altova to make are the intellectual property of and are owned by Altova and its suppliers. The structure, organization and code of the Software are the valuable trade secrets and confidential information of Altova and its suppliers. The Software is protected by copyright, including without limitation by United States Copyright Law, international treaty provisions and applicable laws in the country in which it is being used. You acknowledge that Altova retains the ownership of all patents, copyrights, trade secrets, trademarks and other intellectual property rights pertaining to the Software, and that Altova's ownership rights extend to any images, photographs, animations, videos, audio, music, text and "applets" incorporated into the Software and all accompanying printed materials. You will take no actions which adversely affect Altova's intellectual property rights in the Software. Trademarks shall be used in accordance with accepted trademark practice, including identification of trademark owners' names. Trademarks may only be used to identify printed output produced by the Software, and such use of any trademark does not give you any right of ownership in that trademark. Altova®, XMLSpy®, Authentic®, StyleVision®, MapForce®, UModel®, DatabaseSpy®, DiffDog®, SchemaAgent®, SemanticWorks®, MissionKit®, Markup Your Mind®, Nanonull™, RaptorXML™, RaptorXML Server™, RaptorXML +XBRL Server™, Powered By RaptorXML™, FlowForce Server™, StyleVision Server™, and MapForce Server™ are trademarks of Altova GmbH. (pending or registered in numerous countries). Unicode and the Unicode Logo are trademarks of Unicode, Inc. Windows, Windows XP, Windows Vista, Windows 7, and Windows 8 are trademarks of Microsoft. W3C, CSS, DOM, MathML, RDF, XHTML, XML and XSL are trademarks (registered in numerous countries) of the World Wide Web Consortium (W3C); marks of the W3C are registered and held by its host institutions, MIT, INRIA and Keio. Except as expressly stated above, this Agreement does not grant you any intellectual property rights in the Software. Notifications of claimed copyright infringement should be sent to Altova's copyright agent as further provided on the Altova Web Site.

3. LIMITED TRANSFER RIGHTS

Notwithstanding the foregoing, you may transfer all your rights to use the Software to another person or legal entity provided that: (a) you also transfer this Agreement, the Software and all other software or hardware bundled or pre-installed with the Software, including all copies, updates and prior versions, and all copies of font software converted into other formats, to such person or entity; (b) you retain no copies, including backups and copies stored on a computer; (c) the receiving party secures a personalized key code from Altova; and (d) the receiving party accepts the terms and conditions of this Agreement and any other terms and conditions upon which you legally purchased a license to the Software. Notwithstanding the foregoing, you may not transfer education, pre-release, or not-for-resale copies of the Software.

4. PRE-RELEASE AND EVALUATION PRODUCT ADDITIONAL TERMS

If the product you have received with this license is pre-commercial release or beta Software ("Pre-

release Software"), then this Section applies. In addition, this section applies to all evaluation and/ or demonstration copies of Altova software ("Evaluation Software") and continues in effect until you purchase a license. To the extent that any provision in this section is in conflict with any other term or condition in this Agreement, this section shall supersede such other term(s) and condition(s) with respect to the Pre-release and/or Evaluation Software, but only to the extent necessary to resolve the conflict. You acknowledge that the Pre-release Software is a pre-release version, does not represent final product from Altova, and may contain bugs, errors and other problems that could cause system or other failures and data loss. CONSEQUENTLY, THE PRE-RELEASE AND/OR EVALUATION SOFTWARE IS PROVIDED TO YOU **"AS-IS" WITH NO WARRANTIES FOR USE OR PERFORMANCE,** AND ALTOVA DISCLAIMS ANY WARRANTY OR LIABILITY OBLIGATIONS TO YOU OF ANY KIND, WHETHER EXPRESS OR IMPLIED. WHERE LEGALLY LIABILITY CANNOT BE EXCLUDED FOR PRE-RELEASE AND/OR EVALUATION SOFTWARE, BUT IT MAY BE LIMITED, ALTOVA'S LIABILITY AND THAT OF ITS SUPPLIERS SHALL BE LIMITED TO THE SUM OF FIFTY DOLLARS (USD $50) IN TOTAL. If the Evaluation Software has a time-out feature, then the software will cease operation after the conclusion of the designated evaluation period. Upon such expiration date, your license will expire unless otherwise extended. Your license to use any output created with the Evaluation Software that contains generated program code (including Unrestricted Source Code) such as Java, C++, C, VB.NET or XSLT and associated project files and build scripts as well as generated XML, XML Schemas, documentation, UML diagrams, and database structures terminates automatically upon the expiration of the designated evaluation period but the license to use such output is revived upon your purchase of a license for the Software that you evaluated and used to create such output. Access to any files created with the Evaluation Software is entirely at your risk. You acknowledge that Altova has not promised or guaranteed to you that Pre-release Software will be announced or made available to anyone in the future, that Altova has no express or implied obligation to you to announce or introduce the Pre-release Software, and that Altova may not introduce a product similar to or compatible with the Pre-release Software. Accordingly, you acknowledge that any research or development that you perform regarding the Pre-release Software or any product associated with the Pre-release Software is done entirely at your own risk. During the term of this Agreement, if requested by Altova, you will provide feedback to Altova regarding testing and use of the Pre-release Software, including error or bug reports. If you have been provided the Pre-release Software pursuant to a separate written agreement, your use of the Software is governed by such agreement. You may not sublicense, lease, loan, rent, distribute or otherwise transfer the Pre-release Software. Upon receipt of a later unreleased version of the Pre-release Software or release by Altova of a publicly released commercial version of the Software, whether as a stand-alone product or as part of a larger product, you agree to return or destroy all earlier Pre-release Software received from Altova and to abide by the terms of the license agreement for any such later versions of the Pre-release Software.

5. LIMITED WARRANTY AND LIMITATION OF LIABILITY

(a)   Limited Warranty and Customer Remedies. Altova warrants to the person or entity that first purchases a license for use of the Software pursuant to the terms of this Agreement that (i) the Software will perform substantially in accordance with any accompanying Documentation for a period of ninety (90) days from the date of receipt, and (ii) any support services provided by Altova shall be substantially as described in Section 6 of this agreement. Some states and jurisdictions do not allow limitations on duration of an implied warranty, so the above limitation may not apply to you. To the extent allowed by applicable law, implied warranties on the Software, if any, are limited to ninety (90) days. Altova's and its suppliers' entire liability and your exclusive remedy shall be, at Altova's option, either (i) return of the price paid, if any, or (ii) repair or replacement of the Software that does not meet Altova's Limited Warranty and which is returned to Altova with a copy of your receipt. This Limited Warranty is void if failure of the Software has resulted from accident, abuse, misapplication, abnormal use, Trojan horse, virus, or

any other malicious external code. Any replacement Software will be warranted for the remainder of the original warranty period or thirty (30) days, whichever is longer. This limited warranty does not apply to Evaluation and/or Pre-release Software.

(b)   No Other Warranties and Disclaimer. THE FOREGOING LIMITED WARRANTY AND REMEDIES STATE THE SOLE AND EXCLUSIVE REMEDIES FOR ALTOVA OR ITS SUPPLIER'S BREACH OF WARRANTY. ALTOVA AND ITS SUPPLIERS DO NOT AND CANNOT WARRANT THE PERFORMANCE OR RESULTS YOU MAY OBTAIN BY USING THE SOFTWARE. EXCEPT FOR THE FOREGOING LIMITED WARRANTY, AND FOR ANY WARRANTY, CONDITION, REPRESENTATION OR TERM TO THE EXTENT WHICH THE SAME CANNOT OR MAY NOT BE EXCLUDED OR LIMITED BY LAW APPLICABLE TO YOU IN YOUR JURISDICTION, ALTOVA AND ITS SUPPLIERS MAKE NO WARRANTIES, CONDITIONS, REPRESENTATIONS OR TERMS, EXPRESS OR IMPLIED, WHETHER BY STATUTE, COMMON LAW, CUSTOM, USAGE OR OTHERWISE AS TO ANY OTHER MATTERS. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, ALTOVA AND ITS SUPPLIERS DISCLAIM ALL OTHER WARRANTIES AND CONDITIONS, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, SATISFACTORY QUALITY, INFORMATIONAL CONTENT OR ACCURACY, QUIET ENJOYMENT, TITLE AND NON-INFRINGEMENT, WITH REGARD TO THE SOFTWARE, AND THE PROVISION OF OR FAILURE TO PROVIDE SUPPORT SERVICES. THIS LIMITED WARRANTY GIVES YOU SPECIFIC LEGAL RIGHTS. YOU MAY HAVE OTHERS, WHICH VARY FROM STATE/JURISDICTION TO STATE/JURISDICTION.

(c)   Limitation of Liability. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW EVEN IF A REMEDY FAILS ITS ESSENTIAL PURPOSE, IN NO EVENT SHALL ALTOVA OR ITS SUPPLIERS BE LIABLE FOR ANY SPECIAL, INCIDENTAL, DIRECT, INDIRECT OR CONSEQUENTIAL DAMAGES WHATSOEVER (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION, OR ANY OTHER PECUNIARY LOSS) ARISING OUT OF THE USE OF OR INABILITY TO USE THE SOFTWARE OR THE PROVISION OF OR FAILURE TO PROVIDE SUPPORT SERVICES, EVEN IF ALTOVA HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. IN ANY CASE, ALTOVA'S ENTIRE LIABILITY UNDER ANY PROVISION OF THIS AGREEMENT SHALL BE LIMITED TO THE AMOUNT ACTUALLY PAID BY YOU FOR THE SOFTWARE PRODUCT. Because some states and jurisdictions do not allow the exclusion or limitation of liability, the above limitation may not apply to you. In such states and jurisdictions, Altova's liability shall be limited to the greatest extent permitted by law and the limitations or exclusions of warranties and liability contained herein do not prejudice applicable statutory consumer rights of person acquiring goods otherwise than in the course of business. The disclaimer and limited liability above are fundamental to this Agreement between Altova and you.

(d)   Infringement Claims. Altova will indemnify and hold you harmless and will defend or settle any claim, suit or proceeding brought against you by a third party that is based upon a claim that the content contained in the Software infringes a copyright or violates an intellectual or proprietary right protected by United States or European Union law ("Claim"), but only to the extent the Claim arises directly out of the use of the Software and subject to the limitations set forth in Section 5 of this Agreement except as otherwise expressly provided. You must notify Altova in writing of any Claim within ten (10) business days after you first receive notice of the Claim, and you shall provide to Altova at no cost such assistance and cooperation as Altova may reasonably request from time to time in connection with the defense of the Claim. Altova shall have sole control over any Claim (including, without limitation, the selection of counsel and the right to settle on your behalf on any terms Altova deems desirable in the sole exercise of its discretion). You may, at your sole cost, retain separate counsel and participate in the defense or settlement negotiations. Altova shall pay actual damages, costs, and attorney fees awarded

against you (or payable by you pursuant to a settlement agreement) in connection with a Claim to the extent such direct damages and costs are not reimbursed to you by insurance or a third party, to an aggregate maximum equal to the purchase price of the Software. If the Software or its use becomes the subject of a Claim or its use is enjoined, or if in the opinion of Altova's legal counsel the Software is likely to become the subject of a Claim, Altova shall attempt to resolve the Claim by using commercially reasonable efforts to modify the Software or obtain a license to continue using the Software. If in the opinion of Altova's legal counsel the Claim, the injunction or potential Claim cannot be resolved through reasonable modification or licensing, Altova, at its own election, may terminate this Agreement without penalty, and will refund to you on a pro rata basis any fees paid in advance by you to Altova. THE FOREGOING CONSTITUTES ALTOVA'S SOLE AND EXCLUSIVE LIABILITY FOR INTELLECTUAL PROPERTY INFRINGEMENT. This indemnity does not apply to situations where the alleged infringement, whether patent or otherwise, is the result of a combination of the Altova software and additional elements supplied by you.

6. SUPPORT AND MAINTENANCE

Altova offers multiple optional "Support & Maintenance Package(s)" ("SMP") for the version of Software product edition that you have licensed, which you may elect to purchase in addition to your Software license. The Support Period, hereinafter defined, covered by such SMP shall be delineated at such time as you elect to purchase a SMP. Your rights with respect to support and maintenance as well as your upgrade eligibility depend on your decision to purchase SMP and the level of SMP that you have purchased:

(a)   If you have not purchased SMP, you will receive the Software AS IS and will not receive any maintenance releases or updates. However, Altova, at its option and in its sole discretion on a case by case basis, may decide to offer maintenance releases to you as a courtesy, but these maintenance releases will not include any new features in excess of the feature set at the time of your purchase of the Software. In addition, Altova will provide free technical support to you for thirty (30) days after the date of your purchase (the "Support Period" for the purposes of this paragraph 6(a), and Altova, in its sole discretion on a case by case basis, may also provide free courtesy technical support during your thirty (30) day evaluation period. Technical support is provided via a Web-based support form only, and there is no guaranteed response time.

(b)   If you have purchased SMP, then solely for the duration of its delineated Support Period, **you are eligible to receive the version of the Software edition** that you have licensed and all maintenance releases and updates for that edition that are released during your Support Period. For the duration of your SMP's Support Period, you will also be eligible to receive upgrades to the comparable edition of the next version of the Software that succeeds the Software edition that you have licensed for applicable upgrades released during your Support Period. The specific upgrade edition that you are eligible to receive based on your Support Period is further detailed in the SMP that you have purchased. Software that is introduced as separate product is not included in SMP. Maintenance releases, updates and upgrades may or may not include additional features. In addition, Altova will provide Priority Technical Support to you for the duration of the Support Period. Priority Technical Support is provided via a Web-based support form only and Altova will make commercially reasonable efforts to respond via e-mail to all requests within forty-eight (48) hours during Altova's business hours (MO-FR, 8am UTC – 10pm UTC, Austrian and US holidays excluded) and to make reasonable efforts to provide work-arounds to errors reported in the Software.

During the Support Period you may also report any Software problem or error to Altova. If Altova determines that a reported reproducible material error in the Software exists and significantly impairs the usability and utility of the Software, Altova agrees to use reasonable commercial efforts to correct or provide a usable work-around solution in an upcoming maintenance release or

update, which is made available at certain times at Altova's sole discretion.

If Altova, in its discretion, requests written verification of an error or malfunction discovered by you or requests supporting example files that exhibit the Software problem, you shall promptly provide such verification or files, by email, telecopy, or overnight mail, setting forth in reasonable detail the respects in which the Software fails to perform. You shall use reasonable efforts to cooperate in diagnosis or study of errors. Altova may include error corrections in maintenance releases, updates, or new major releases of the Software. Altova is not obligated to fix errors that are immaterial. Immaterial errors are those that do not significantly impact use of the Software as determined by Altova in its sole discretion. Whether or not you have purchased the Support & Maintenance Package, technical support only covers issues or questions resulting directly out of the operation of the Software and Altova will not provide you with generic consultation, assistance, or advice under any circumstances.

Updating Software may require the updating of software not covered by this Agreement before installation. Updates of the operating system and application software not specifically covered by this Agreement are your responsibility and will not be provided by Altova under this Agreement. Altova's obligations under this Section 6 are contingent upon your proper use of the Software and your compliance with the terms and conditions of this Agreement at all times. Altova shall be under no obligation to provide the above technical support if, in Altova's opinion, the Software has failed due to the following conditions: (i) damage caused by the relocation of the Software to another location or CPU; (ii) alterations, modifications or attempts to change the Software without Altova's written approval; (iii) causes external to the Software, such as natural disasters, the failure or fluctuation of electrical power, or computer equipment failure; (iv) your failure to maintain the Software at Altova's specified release level; or (v) use of the Software with other software without Altova's prior written approval. It will be your sole responsibility to: (i) comply with all Altova-specified operating and troubleshooting procedures and then notify Altova immediately of Software malfunction and provide Altova with complete information thereof; (ii) provide for the security of your confidential information; (iii) establish and maintain backup systems and procedures necessary to reconstruct lost or altered files, data or programs.

7. SOFTWARE ACTIVATION, UPDATES AND LICENSE METERING

(a)   License Metering. The Software includes a built-in license metering module that is designed to assist you with monitoring license compliance in small local area networks (LAN). The metering module attempts to communicate with other machines on your local area network (LAN). You permit Altova to use your internal network for license monitoring for this purpose. This license metering module may be used to assist with your license compliance but should not be the sole method. Should your firewall settings block said communications, you must deploy an accurate means of monitoring usage by the end user and preventing users from using the Software more than the Permitted Number.

(b)   License Compliance Monitoring. You are required to utilize a process or tool to ensure that the Permitted Number is not exceeded. Without prejudice or waiver of any potential violations of the Agreement, Altova may provide you with additional compliance tools should you be unable to accurately account for license usage within your organization. If provided with such a tool by Altova, you (a) are required to use it in order to comply with the terms of this Agreement and (b) permit Altova to use your internal network for license monitoring and metering and to generate compliance reports that are communicated to Altova from time to time.

(c)   Software Activation. **The Software may use your internal network and Internet connection for the purpose of transmitting license-related data at the time of installation, registration, use, or update to an Altova Master License Server and validating the**

authenticity of the license-related data in order to protect Altova against unlicensed or illegal use of the Software and to improve customer service. Activation is based on the exchange of license related data between your computer and the Altova Master License Server. You agree that Altova may use these measures and you agree to follow any applicable requirements. You further agree that use of license key codes that are not or were not generated by Altova and lawfully obtained from Altova, or an authorized reseller as part of an effort to activate or use the Software violates Altova's intellectual property rights as well as the terms of this Agreement. You agree that efforts to circumvent or disable Altova's copyright protection mechanisms, the license management mechanism, or the Altova Master License Server violate Altova's intellectual property rights as well as the terms of this Agreement. Altova expressly reserves the rights to seek all available legal and equitable remedies to prevent such actions and to recover lost profits, damages and costs.

(d)   LiveUpdate. Altova provides a new LiveUpdate notification service to you, which is free of charge. Altova may use your internal network and Internet connection for the purpose of transmitting license-related data to an Altova-operated LiveUpdate server to validate your license at appropriate intervals and determine if there is any update available for you.

(e)   Use of Data. The terms and conditions of the Privacy Policy are set out in full at http://www.altova.com/privacy and are incorporated by reference into this Agreement. By your acceptance of the terms of this Agreement and/or use of the Software, you authorize the collection, use and disclosure of information collected by Altova for the purposes provided for in this Agreement and/or the Privacy Policy. Altova has the right in its sole discretion to amend this provision of the Agreement and/or Privacy Policy at any time. You are encouraged to review the terms of the Privacy Policy as posted on the Altova Web site from time to time.

(f)   Audit Rights. You agree that Altova may audit your use of the Software for compliance with the terms of this Agreement at any time, upon reasonable notice. In the event that such audit reveals any use of the Software by you other than in full compliance with the terms of this Agreement, you shall reimburse Altova for all reasonable expenses related to such audit in addition to any other liabilities you may incur as a result of such non-compliance.

(g)   Notice to European Users. Please note that the information as described in paragraph 7(d) above may be transferred outside of the European Economic Area, for purposes of processing, analysis, and review, by Altova, Inc., a company located in Beverly, Massachusetts, U.S.A., or its subsidiaries or Altova's subsidiaries or divisions, or authorized partners, located worldwide. You are advised that the United States uses a sectoral model of privacy protection that relies on a mix of legislation, governmental regulation, and self-regulation. You are further advised that the Council of the European Union has found that this model does not provide "adequate" privacy protections as contemplated by Article 25 of the European Union's Data Directive. (Directive 95/46/EC, 1995 O.J. (L 281) 31). Article 26 of the European Union's Data Directive allows for transfer of personal data from the European Union to a third country if the individual has unambiguously given his consent to the transfer of personal information, regardless of the third country's level of protection. By agreeing to this Agreement, you consent to the transfer of all such information to the United States and the processing of that information as described in this Agreement and the Privacy Policy.

8. TERM AND TERMINATION

This Agreement may be terminated (a) by your giving Altova written notice of termination; (b) by Altova, at its option, giving you written notice of termination if you commit a breach of this Agreement and fail to cure such breach within ten (10) days after notice from Altova; or (c) at the

request of an authorized Altova reseller in the event that you fail to make your license payment or other monies due and payable. In addition the Agreement governing your use of a previous version of the Software that you have upgraded or updated is terminated upon your acceptance of the terms and conditions of the Agreement accompanying such upgrade or update. Upon any termination of the Agreement, you must cease all use of the Software that this Agreement governs, destroy all copies then in your possession or control and take such other actions as Altova may reasonably request to ensure that no copies of the Software remain in your possession or control. The terms and conditions set forth in Sections 1(h), 1(i), 1(j), 1(k), 1(l), 2, 5, 7, 9, 10, 11, and 11 survive termination as applicable.

9. RESTRICTED RIGHTS NOTICE AND EXPORT RESTRICTIONS

The Software was developed entirely at private expense and is commercial computer software provided with **RESTRICTED RIGHTS**. Use, duplication or disclosure by the U.S. Government or a U.S. Government contractor or subcontractor is subject to the restrictions set forth in this Agreement and as provided in FAR 12.211 and 12.212 (48 C.F.R. §12.211 and 12.212) or DFARS 227. 7202 (48 C.F.R. §227-7202) as applicable. Consistent with the above as applicable, Commercial Computer Software and Commercial Computer Documentation licensed to U.S. government end users only as commercial items and only with those rights as are granted to all other end users under the terms and conditions set forth in this Agreement. Manufacturer is Altova GmbH, Rudolfsplatz 13a/9, A-1010 Vienna, Austria/EU. You may not use or otherwise export or re-export the Software or Documentation except as authorized by United States law and the laws of the jurisdiction in which the Software was obtained. In particular, but without limitation, the Software or Documentation may not be exported or re-exported (i) into (or to a national or resident of) any U.S. embargoed country or (ii) to anyone on the U.S. Treasury Department's list of Specially Designated Nationals or the U.S. Department of Commerce's Table of Denial Orders. By using the Software, you represent and warrant that you are not located in, under control of, or a national or resident of any such country or on any such list.

10. U.S. GOVERNMENT ENTITIES

Notwithstanding the foregoing, if you are an agency, instrumentality or department of the federal government of the United States, then this Agreement shall be governed in accordance with the laws of the United States of America, and in the absence of applicable federal law, the laws of the Commonwealth of Massachusetts will apply. Further, and notwithstanding anything to the contrary in this Agreement (including but not limited to Section 5 (Indemnification)), all claims, demands, complaints and disputes will be subject to the Contract Disputes Act (41 U.S.C. §§7101 *et seq.*), the Tucker Act (28 U.S.C. §1346(a) and §1491), or the Federal Tort Claims Act (28 U.S.C. §§1346(b), 2401-2402, 2671-2672, 2674-2680), FAR 1.601(a) and 43.102 (Contract Modifications); FAR 12.302(b), as applicable, or other applicable governing authority. For the avoidance of doubt, if you are an agency, instrumentality, or department of the federal, state or local government of the U.S. or a U.S. public and accredited educational institution, then your indemnification obligations are only applicable to the extent they would not cause you to violate any applicable law (e.g., the Anti-Deficiency Act), and you have any legally required authorization or authorizing statute.

11. THIRD PARTY SOFTWARE

The Software may contain third party software which requires notices and/or additional terms and conditions. Such required third party software notices and/or additional terms and conditions are located at our Website at http://www.altova.com/legal_3rdparty.html and are made a part of and incorporated by reference into this Agreement. By accepting this Agreement, you are also accepting the additional terms and conditions, if any, set forth therein.

12. JURISDICTION, CHOICE OF LAW, AND VENUE

If you are located in the European Union and are using the Software in the European Union and not in the United States, then this Agreement will be governed by and construed in accordance with the laws of the Republic of Austria (excluding its conflict of laws principles and the U.N. Convention on Contracts for the International Sale of Goods) and you expressly agree that exclusive jurisdiction for any claim or dispute with Altova or relating in any way to your use of the Software resides in the Handelsgericht, Wien (Commercial Court, Vienna) and you further agree and expressly consent to the exercise of personal jurisdiction in the Handelsgericht, Wien (Commercial Court, Vienna) in connection with any such dispute or claim.

If you are located in the United States or are using the Software in the United States then this Agreement will be governed by and construed in accordance with the laws of the Commonwealth of Massachusetts, USA (excluding its conflict of laws principles and the U.N. Convention on Contracts for the International Sale of Goods) and you expressly agree that exclusive jurisdiction for any claim or dispute with Altova or relating in any way to your use of the Software resides in the federal or state courts of the Commonwealth of Massachusetts and you further agree and expressly consent to the exercise of personal jurisdiction in the federal or state courts of the Commonwealth of Massachusetts in connection with any such dispute or claim.

If you are located outside of the European Union or the United States and are not using the Software in the United States, then this Agreement will be governed by and construed in accordance with the laws of the Republic of Austria (excluding its conflict of laws principles and the U.N. Convention on Contracts for the International Sale of Goods) and you expressly agree that exclusive jurisdiction for any claim or dispute with Altova or relating in any way to your use of the Software resides in the Handelsgericht, Wien (Commercial Court, Vienna) and you further agree and expressly consent to the exercise of personal jurisdiction in the Handelsgericht Wien (Commercial Court, Vienna) in connection with any such dispute or claim. This Agreement will not be governed by the conflict of law rules of any jurisdiction or the United Nations Convention on Contracts for the International Sale of Goods, the application of which is expressly excluded.

13.   TRANSLATIONS

Where Altova has provided you with a foreign translation of the English language version, you agree that the translation is provided for your convenience only and that the English language version will control. If there is any contradiction between the English language version and a translation, then the English language version shall take precedence.

14.   GENERAL PROVISIONS

This Agreement contains the entire agreement and understanding of the parties with respect to the subject matter hereof, and supersedes all prior written and oral understandings of the parties with respect to the subject matter hereof. Any notice or other communication given under this Agreement shall be in writing and shall have been properly given by either of us to the other if sent by certified or registered mail, return receipt requested, or by overnight courier to the address shown on Altova's Web site for Altova and the address shown in Altova's records for you, or such other address as the parties may designate by notice given in the manner set forth above. This Agreement will bind and inure to the benefit of the parties and our respective heirs, personal and legal representatives, affiliates, successors and permitted assigns. The failure of either of us at any time to require performance of any provision hereof shall in no manner affect such party's right at a later time to enforce the same or any other term of this Agreement. This Agreement may be amended only by a document in writing signed by both of us. In the event of a breach or

threatened breach of this Agreement by either party, the other shall have all applicable equitable as well as legal remedies. Each party is duly authorized and empowered to enter into and perform this Agreement. If, for any reason, any provision of this Agreement is held invalid or otherwise unenforceable, such invalidity or unenforceability shall not affect the remainder of this Agreement, and this Agreement shall continue in full force and effect to the fullest extent allowed by law. The parties knowingly and expressly consent to the foregoing terms and conditions.

Last updated: 2015/09/03

Index

A

B

Properties,

for objects, 240

for schema comparisons, 242

for SQL, 245

for SQL folder, 242

for sub-folders, 242

for tables, 240

for XML schemas, 390, 395

object count, 240

project, 242

resetting, 242

restoring, 242

Property window,

toggling on and off, 488

Q

Queries,

creating from Select statements, 495

Querying,

database, 49, 104

using scripts, 104

views, 52

QuickStart Tutorial,

adding rows, 57

appending rows, 57

browsing the database, 45

connecting to a data source, 45

connecting to database, 43

creating views, 51

deleting rows, 58

displaying tables, 46

displaying tables in Design Editor, 48

editing data, 54, 56

editing tables, 53, 56

executing for editing, 55

executing SQL, 50

exporting data, 60

exporting to XML files, 61

generating import script, 59

importing data, 58

opening project, 44

overview, 43

querying the database, 49

retrieving data, 47, 52

selecting tables for editing, 54

showing tables in Design editor, 48

updating data, 53, 56

updating tables, 56

viewing database structure, 47

viewing tables in Design Editor, 48

XML export, 61

Quitting DatabaseSpy, 485

Quotations,

adding to identifiers, 495

removing from identifiers, 495

R

Redo command, 486

Reducing vertical spacing, 420

Regions,

collapsing, 319

creating, 319

expanding, 319

folding margin, 319

inserting, 319, 492

removing, 319

Registering,

XML schemas, 393

Regular expressions,

searching SQL Editor, 334

Relations,

displaying in Design Editor, 467

hiding labels of unselected, 432

highlight active objects, 432

highlighting object of selected relation, 467

showing in Design Editor, 465, 467

viewing in Design Editor, 465

Relationship,

creating, 79

Reloading files, 422, 482

Removing,

all bookmarks, 493

block comment, 330, 492

bookmarks, 319, 492

comments, 314, 330, 495

equal tables, 137

formatting, 495

from design, 499

line comment, 330, 492

objects from favorites, 235

T