MS SQL Maestro

User's guide

Table of Contents

IX Database Tools 362

1 Welcome to MS SQL Maestro!

MS SQL Maestro is the premier Windows GUI admin tool for Microsoft SQL development and management. It allows you to make all the database operations easy and fast.

Basic MS SQL Maestro features

Support of the latest Microsoft SQL features

Use MS SQL Maestro to work with Microsoft SQL server versions up to SQL Server 2022. Among other features and objects implemented in the latest versions of the server, our software supports schemas, synonyms, xml schema collections, assemblies, credentials, certificates, schedules, and a lot of other useful things.

Easy database management

MS SQL Maestro allows you to create new databases and drop existing ones. Database profiles give you the opportunity to connect to databases in one touch and work with the selected databases only. See the Database Management 22 for details.

Powerful database object management

MS SQL Maestro provides you with an ability to manage database objects in various ways. For example, you can perform operations with a group of objects as well as with a single object in Object Manager 66, sort, group and filter the database objects within Object Browser 65, copy an object from one database to another by a drag-and-drop operation inside the explorer tree, use Windows clipboard to copy a set of objects and so on. For details turn to the Database Object Management 69 section.

Working with tables and table subobjects

MS SQL Maestro wizards and editors allow you to create, edit and drop tables as well as their *fields, indexes,* and *foreign keys* in a couple of simple operations. See the Tables 76 section to learn more.

Building and executing queries

MS SQL Maestro provides two powerful tools which allow you either to edit query text directly with syntax highlighting and code completion or to build a query diagram visually selecting tables and fields, setting links between tables and so on. You can find the detailed description in the Queries 311 section.

Powerful data management tools

MS SQL Maestro puts at your disposal a complete set of data management tools with viewing, editing, grouping, sorting and filtering abilities, lookup editors, master-detail data view, BLOB Viewer/Editor, data export, data import and SQL dump modules and more. See the Data Management 327 to learn the details.

Wide choice of additional tools

MS SQL Maestro provides you with a number of tools for working with database metadata and SQL scripts, including Script Runner, SQL Script Editor with code folding and script explorer. Moreover, it gives such tools as Schema Designer, BLOB Viewer, Diagram Viewer, Data Analysis, Dependency Tracker, SQL Generator, Report Designer, and a lot of others. To learn more, see the Database Tools 362 section.

Security management

MS SQL Maestro gives you a comfortable access to Microsoft SQL security features.

Full customization according to your preferences and needs

In MS SQL Maestro you can customize the behavior of all its tools, select a user interface scheme and set a lot of other preferences. All the options and their meanings are listed within the Options 424 dialog description.

1.1 System Requirements

Client environment

- Pentium PC or higher;
- Windows NT4/2000/XP/Vista/Windows 7/Windows 8/Windows 10/Windows 11;
- 512 MB RAM (1 GB recommended);
- 25 MB of free hard disk space;
- SVGA-compatible video adapter.

Server environment

- Microsoft SQL server versions up to SQL Server 2022.

1.2 Installation

To install **MS SQL Maestro** for the first time on your PC:

- download the MS SQL Maestro distribution package from the download page at our site;
- run setup.exe from the local folder and follow the instructions of the installation wizard;
- find the MS SQL Maestro shortcut in the corresponding program group of the Windows Start menu after the installation is completed.

To upgrade the installed copy of MS SQL Maestro to the latest version:

- download the MS SQL Maestro executable file from the download page at our site;
- unzip downloaded file to any local folder, e.g. *c:\unzipped*;
- exit from MS SQL Maestro if it is running;
- replace previous version of MS SQL Maestro by copying unzipped files to the MS SQL Maestro folder;
- run MS SQL Maestro using its shortcut in the Windows Start menu.

You can also use the full distribution package to upgrade your current version of MS SQL Maestro. In this case you should repeat the steps of the first-time installation. Note that the full distribution package is larger than a single executable file.

1.3 How can I purchase MS SQL Maestro?

Thank you for your interest in purchasing **MS SQL Maestro**!

You can select licensing options and register MS SQL Maestro at its on-line order page. It is possible to purchase on-line, by fax, mail, toll-free phone call, or place a purchase order. We send the software activation key by email within 24 hours after completion of the order process. If you have not received the activation key within this period, please contact our sales department.

All our products and bundles are shipped with 12 months of free upgrades (minor and major ones) or with 36 months of free upgrades for a quite small additional fee. After this period you may renew your license for the next 12(36) months with a 50% discount.

MS SQL Maestro has a free 30-day trial. Upon purchasing the product you confirm that you have tested it and you are completely satisfied with its current version.

To obtain technical support, please visit the appropriate section on our website or contact us by email to support@sqlmaestro.com.

1.4 License Agreement

Notice to users: carefully read the following legal agreement. The use of the software provided with this agreement (the "SOFTWARE") constitutes your acceptance of these terms. If you do not agree to the terms of this agreement, do not install and/or use this software. The use of this software is conditioned upon the user's compliance with the terms of this agreement.

- **License grant**. SQL Maestro Group grants you a license to use one copy of the version of this SOFTWARE on any single hardware product for as many licenses as you purchase. "You" means a company, an entity or an individual. "Use" means storing, loading, installing, executing or displaying the SOFTWARE. You may not modify the SOFTWARE or disable any licensing or control features of the SOFTWARE except as an intended part of the SOFTWARE's programming features. This license is not transferable to any other company, entity or individual. You may not publish any registration information (serial numbers, registration keys, etc.) or pass it to any other company, entity or individual.

- **Ownership**. The SOFTWARE is owned and copyrighted by SQL Maestro Group. Your license confers no title or ownership of the SOFTWARE and should not be construed as a sale of any rights for the SOFTWARE.

- **Copyright**. The SOFTWARE is protected by the United States copyright law and international treaty provisions. You acknowledge that no title to the intellectual property in the SOFTWARE is transferred to you. You further acknowledge that title and full ownership rights to the SOFTWARE will remain the exclusive property of SQL Maestro Group and you will not acquire any rights to the SOFTWARE except as expressly set forth in this license. You agree that any copies of the SOFTWARE will contain the same proprietary notices which appear on and in the SOFTWARE.

- **License and distribution**. An unregistered copy of the SOFTWARE ("UNREGISTERED SOFTWARE") may be used for evaluation purposes. The UNREGISTERED SOFTWARE may be freely copied and distributed to other users for their evaluation. If you offer this UNREGISTERED SOFTWARE installation package for download, then you agree to:

- replace existing version of the UNREGISTERED SOFTWARE installation package with the new package immediately after a new version of the SOFTWARE is released by SQL Maestro Group, or
- delete an obsolete version of the UNREGISTERED SOFTWARE installation package immediately upon written email notice by SQL Maestro Group.

A registered copy of the SOFTWARE ("REGISTERED SOFTWARE") allows you to use the SOFTWARE only on a single computer and only by a single user at a time. If you wish to use the SOFTWARE for more than one user, you will need a separate license for each individual user. You are allowed to make one copy of the REGISTERED SOFTWARE for back-up purposes.

- **Reverse engineering**. You affirm that you will not attempt to reverse compile, modify, translate, or disassemble the SOFTWARE in whole or in part.

- **Unauthorized use**. You may not use, copy, rent, lease, sell, modify, decompile, disassemble, otherwise reverse engineer, or transfer the SOFTWARE except as provided in this agreement. Any such unauthorized use shall result in immediate and automatic termination of this license.

- **No other warranties**. SQL Maestro Group does not warrant that the SOFTWARE is error-free. SQL Maestro Group disclaims all other warranties with respect to the SOFTWARE, either express or implied, including but not limited to implied warranties of merchantability, fitness for a particular purpose and noninfringement of third party rights. Some jurisdictions do not allow the exclusion of implied warranties or limitations on how long an implied warranty may last, or the exclusion or limitation of incidental or consequential damages, so the above given limitations or exclusions may not apply to you. This warranty gives you specific legal rights and you may also have other rights which vary from jurisdiction to jurisdiction.

- **Limited warranty**. This SOFTWARE is provided on an "AS IS" basis. SQL Maestro Group disclaims all warranties relating to this SOFTWARE, whether expressed or implied, including but not limited to any implied warranties of merchantability or fitness for a particular purpose. Neither SQL Maestro Group nor anyone else who has been involved in the creation, production, or delivery of this SOFTWARE shall be liable for any indirect, consequential, or incidental damages arising out of the use or inability to use such SOFTWARE, even if SQL Maestro Group has been advised of the possibility of such damages or claims. The person using the SOFTWARE bears all risk as to the quality and performance of the SOFTWARE.

Some jurisdictions do not allow limitation or exclusion of incidental or consequential damages, so the above given limitations or exclusion may not apply to you to the extent that liability is by law incapable of exclusion or restriction.

- **Severability**. In the event of invalidity of any provision of this license, the parties agree that such invalidity shall not affect the validity of the remaining portions of this license.

- **No liability for consequential damages**. In no event shall SQL Maestro Group or its suppliers be liable to you for any consequential, special, incidental or indirect damages of any kind arising out of the delivery, performance or use of the SOFTWARE, even if SQL Maestro Group has been advised of the possibility of such damages. In no event will SQL Maestro Group's liability for any claim, whether in contract, tort or any other theory of liability, exceed the license fee paid by you, if any.

- **Entire agreement**. This is the entire agreement between you and SQL Maestro Group which supersedes any prior agreement or understanding, whether written or oral, relating to the subject matter of this license.

- **Reserved rights**. All rights not expressly granted here are reserved to SQL Maestro Group.

1.5 About SQL Maestro Group

SQL Maestro Group is a privately-held company producing high-quality software for database administrators and developers. The united team of eminently qualified developers is pleased to create new software products for commercial, academic and government customers worldwide. We do our best to design and develop products that remove complexity, improve productivity, compress time frames, and increase database performance and availability. We are glad to realize that our products take usual chores upon themselves, so that our customers could have more time left for their creative work.

The company was founded in 2002 as an essential partner for every business that is trying to harness the explosive growth in corporate data. SQL Maestro Group employs an international team concentrating their efforts on cutting-edge DBA tools development.

The slogan of our company is **The Shortest Path to SQL**. It is aimed to denote that we set to create easy-to-use products meant for those who appreciate comfort, friendly program interface and support when working with SQL servers.

- We are pleased to facilitate your job.
- We aim at being of considerable assistance to our clients.
- We feel contented doing our beloved work.

At present, our company offers a series of Windows GUI admin tools for SQL management, control and development of the following servers: **MySQL, Microsoft SQL Server, PostgreSQL, Oracle, SQL Anywhere, DB2, SQLite, Firebird,** and **MaxDB**. We also produce universal tools to be used for administering any database engine accessible via ODBC driver or OLE DB provider. Such products may be the clear-cut decision for those who constantly work with several database servers.

SQL Maestro is the premier Windows GUI admin tool for database development, management, and control. It provides you with the ability to perform all the necessary database operations such as creating, editing, copying, extracting and dropping database objects; moreover, you can build queries visually, execute queries and SQL scripts, view and edit data including BLOBs, represent data as diagrams, export and import data to/from most popular file formats, manage users and their privileges (if possible), and use a lot of other tools designed for making your work with your server comfortable and efficient.	

SQL PHP Generator is a powerful tool for creating database-driven web applications visually. It allows you to generate high-quality PHP scripts for working with tables, views and queries through the web. You needn't have any programming background to use it.

SQL Data Wizard is a high-capacity Windows GUI utility for managing your data.

It provides you with a number of easy-to-use wizards for performing the required data manipulation easily and quickly. The tool allows you to export data from Microsoft SQL tables and queries to most popular formats, import data into the tables, generate SQL dump of selected tables, and export/import BLOB fields from/to files.

SQL Code Factory is a premier GUI tool aimed at the SQL queries and scripts development.

It allows you to manage SQL queries and scripts using such useful features as code folding, code completion and syntax highlighting, build query visually, execute several queries at a time, execute scripts from files, view and edit result data with filtering, sorting and grouping abilities, export data to as many as 14 file formats including Excel, RTF and HTML, import data from Excel, CSV, XML and text files, view and edit BLOBs in various way, build diagrams based on Oracle data, and much more.

Database Converter is a user friendly tool to migrate any local or remote ADO-compatible database to Microsoft SQL.

Such tools transfer database schema and data and are equipped with native support for the most popular database servers.

Data Sync is a powerful and easy-to-use tool for database contents comparison and synchronization.

Such tools can be useful for database administrators, developers and testers that need a quick, easy and reliable way to compare and synchronize their data.

The software products are constantly optimized for the latest server versions support.

You can use the following contact information if necessary:

Our web-site	www.sqlmaestro.com
Postal address:	**SQL Maestro Group** 140 Broadway, Suite 706 New York City, New York 10005 United States

Thank you for your interest to our company!

1.6 What's new

Please find out the latest MS SQL Maestro news at http://www.sqlmaestro.com/products/mssql/maestro/news/

2 Getting Started

The topics in this section provide some basic information about MS SQL Maestro, what it is for and what you can do with it.

How to get started:

- Connect to a database with MS SQL Maestro 13
- Explaining user interface 15
- How MS SQL Maestro looks when you start it for the first time 16
- Shortcut keys 21

Learning more:

❑ Study the **Overview of Database Object Management** 43 section for detailed instructions on using MS SQL Maestro.

❑ See **Database Tools** 362 and **Queries** 311 sections for instructions on more advanced procedures!

❑ Find out more about **Working with Data in MS SQL Maestro** 327.

❑ Customize the way MS SQL Maestro works, see **Program Options** 424 for full details.

2.1 Connect to a database

To manage an existing database with MS SQL Maestro, you have to create the according database profile 24 first. A profile stores database connection settings, and some additional options to customize the way the software works with the database. After the creation database profiles appear as nodes in the Explorer tree on the left (profile properties can be later changed with Database Profile Editor 27).

When the profile is created you can connect to the database. To do so, select the database in the Explorer tree 61, or either select the Database | Connect to Database main menu item or use the Connect to Database item of the popup menu. You can also double click the database node in the explorer tree. If connection succeeds, the database node expands displaying the tree of database objects (tables, views, procedures, etc). The database becomes ready for your activities.

How can I disconnect from a database?

In order to disconnect from a database you should first select the database in the explorer tree, then either

- select the Database | Disconnect from Database main menu item

or

- use the Disconnect from Database item of the popup menu.

See also: Connection parameters 14

2.2 Connection parameters

MS SQL Maestro allows you to connect to Microsoft SQL servers using Windows and SQL Server authentication modes.

Server
The full name of SQL Server you want to connect to. Can be specified as *computer_name* (for default SQL Server instances) or as *computer_name\server_name* (for named instances). To connect to an SQL Server running on a non-default port, specify the value of this field as *computer_name,port_number* (or *computer_name \server_name, port_number*). Press the arrow button to scan for SQL Servers accessible in your network.

Provider
The application allows you to connect to SQL Server using any of SQL Server clients installed on your computer. To choose a client you want to use, select the appropriate item in the Provider combobox. The table below shows the correspondence between the value selected in this combobox and SQL Server client to be used.

Value	**SQL Server client**	**Introduced with**
SQLOLEDB	Microsoft OLE DB Provider for SQL Server (default value)	Comes with Windows
SQLNCLI	SQL Server Native Client	SQL Server 2005
SQLNCLI10	SQL Server 2008 [R2] Native Client	SQL Server 2008 [R2]
SQLNCLI11	SQL Server 2012 Native Client (recommended)	SQL Server 2012

We would recommend you to install and use **SQL Server 2012 Native Client** as it (and only it) supports SQL Server Express LocalDB. Also it provides the best support for data types implemented in the recent versions of SQL Server.

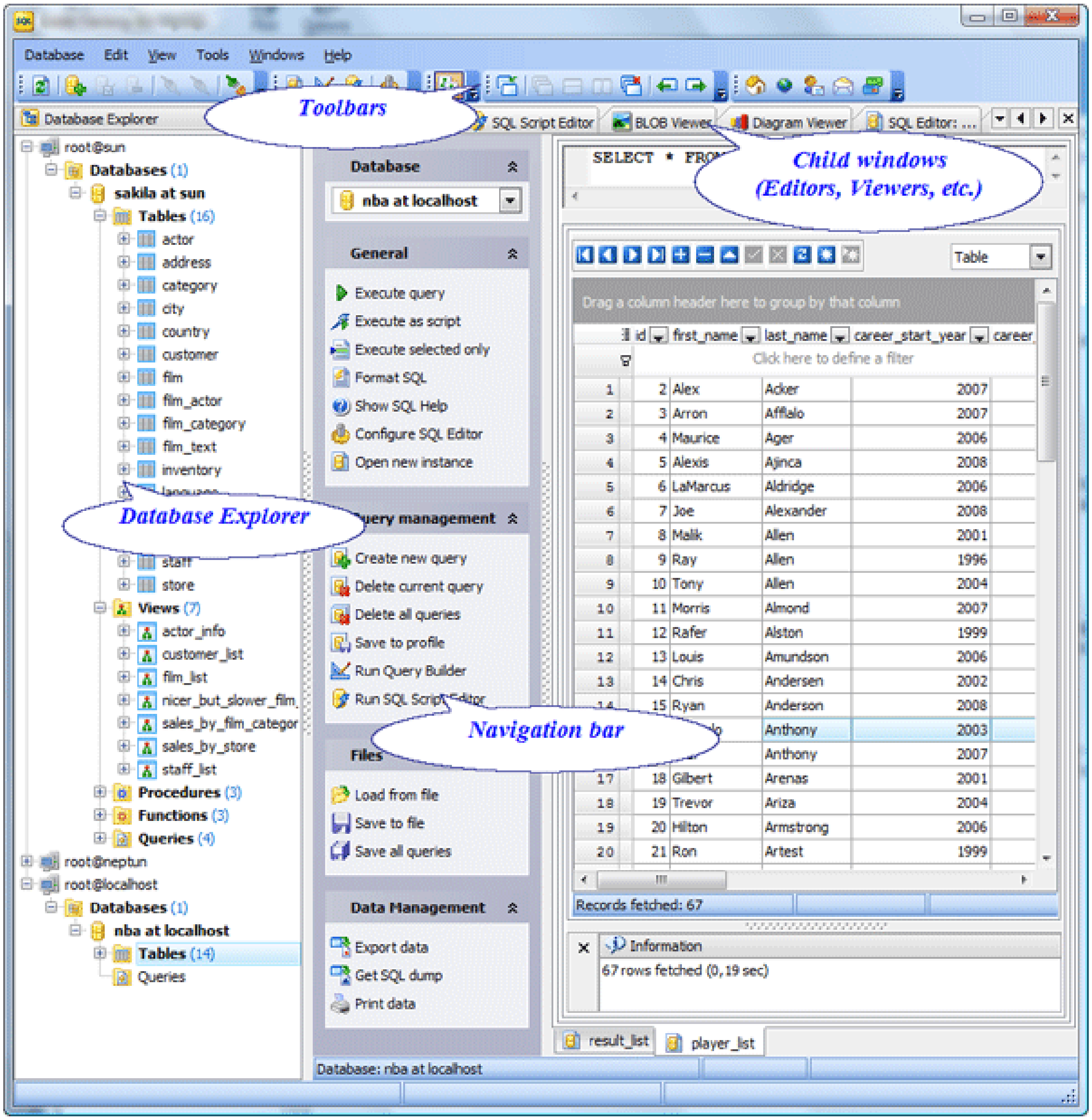
Windows Authentication (more preferable)
Microsoft Windows Authentication mode allows a user to connect through a Windows user account.

SQL Server Authentication
When a user connects with a specified Login name and Password from a non-trusted connection, SQL Server performs the authentication itself by checking to see if a SQL Server login account has been set up and if the specified password matches the one previously recorded.

2.3 Explaining user interface

The SQL Maestro Group products are famous for their clear and intuitive user interface. The programs are built around the three-pane workspace that includes the database explorer and child windows consisting of the navigation bar and work area.

This topic provides a brief guide to the components of MS SQL Maestro's user interface. For detailed descriptions, see below.



```
SELECT * FRO
```

	id	first_name	last_name	career_start_year	career
1	2	Alex	Acker	2007	
2	3	Arron	Afflalo	2007	
3	4	Maurice	Ager	2006	
4	5	Alexis	Ajinca	2008	
5	6	LaMarcus	Aldridge	2006	
6	7	Joe	Alexander	2008	
7	8	Malik	Allen	2001	
8	9	Ray	Allen	1996	
9	10	Tony	Allen	2004	
10	11	Morris	Almond	2007	
11	12	Rafer	Alston	1999	
12	13	Louis	Amundson	2006	
13	14	Chris	Andersen	2002	
14	15	Ryan	Anderson	2008	
[illegible]	[illegible]	[illegible]	Anthony	2003	
[illegible]	[illegible]	[illegible]	Anthony	2007	
17	18	Gilbert	Arenas	2001	
18	19	Trevor	Ariza	2004	
19	20	Hilton	Armstrong	2006	
20	21	Ron	Artest	1999	

Records fetched: 67

67 rows fetched (0,19 sec)

Database Explorer

The Database Explorer 61 occupies the left side of MS SQL Maestro main window. It represents all the connected databases objects including system objects 28.

The explorer provides the fastest way to reach the object properties, to perform the following operations with database profiles using the popup menu:

- create new objects (database profiles, database objects, table objects...);
- edit currently selected objects;
- remove currently selected objects from the explorer tree;
- duplicate objects;
- rename objects if available and edit object comments out of the object editor.

See also: Filtering explorer content 64

Editors and Viewers

According to the MDI style implementation the MS SQL Maestro tools and editors are opened in appropriate windows. Each window consists of a navigation bar and work area. The software supports Classic and Tabbed MDI.

See also: Switching between windows 19, Tabbed MDI 17

Navigation bar

The Navigation Bar contains a set of logically grouped links provided to realize the corresponding actions. Just position the mouse over a link and wait for a second to display the appropriate action shortcut making it possible for experienced users to control the program almost entirely with the keyboard.

See also: Shortcut keys 21

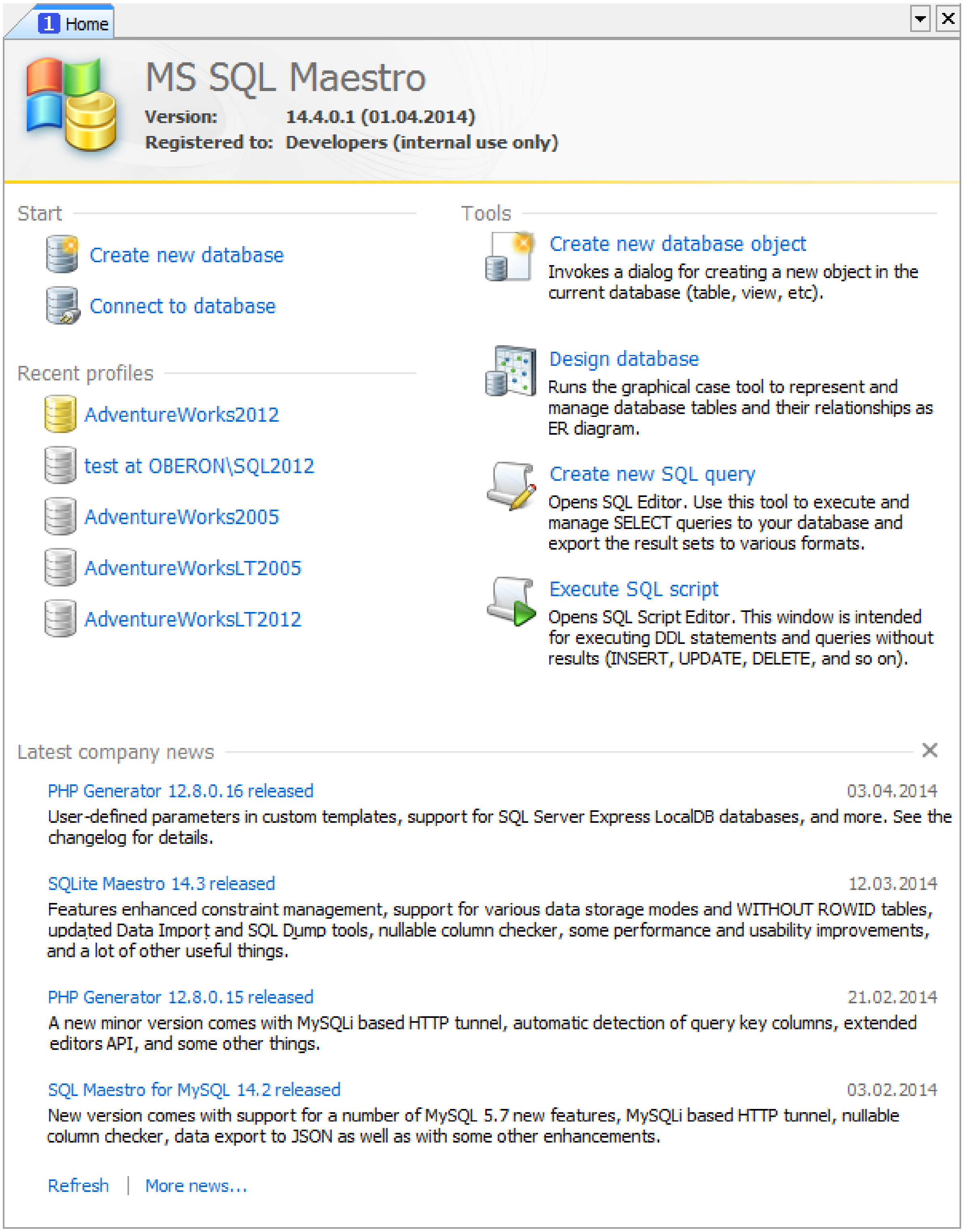
Toolbars

The bars occupy the top of the main window. The Toolbars provide quick access to the most frequently-used functions. Just position the mouse over a tool and wait for a second to display a brief text describing what it is for.

2.3.1 First time started

This is how MS SQL Maestro looks when you run it for the first time. The Create new database 34 and Connect to database 24 links allow you to start working with anew and existing databases.

The window provides you with quick access to the Create Database Object 44 dialog, Schema Designer 401, SQL Editor 313, and SQL Script Editor 364 recently used database profiles. At the bottom of the page the latest company news and current discount programs are represented.



MS SQL Maestro

Version: 14.4.0.1 (01.04.2014)

Registered to: Developers (internal use only)

Create new database object

Invokes a dialog for creating a new object in the current database (table, view, etc).

Design database

Runs the graphical case tool to represent and manage database tables and their relationships as ER diagram.

Create new SQL query

Opens SQL Editor. Use this tool to execute and manage SELECT queries to your database and export the result sets to various formats.

Execute SQL script

Opens SQL Script Editor. This window is intended for executing DDL statements and queries without results (INSERT, UPDATE, DELETE, and so on).

PHP Generator 12.8.0.16 released 03.04.2014

User-defined parameters in custom templates, support for SQL Server Express LocalDB databases, and more. See the changelog for details.

SQLite Maestro 14.3 released 12.03.2014

Features enhanced constraint management, support for various data storage modes and WITHOUT ROWID tables, updated Data Import and SQL Dump tools, nullable column checker, some performance and usability improvements, and a lot of other useful things.

PHP Generator 12.8.0.15 released 21.02.2014

A new minor version comes with MySQLi based HTTP tunnel, automatic detection of query key columns, extended editors API, and some other things.

SQL Maestro for MySQL 14.2 released 03.02.2014

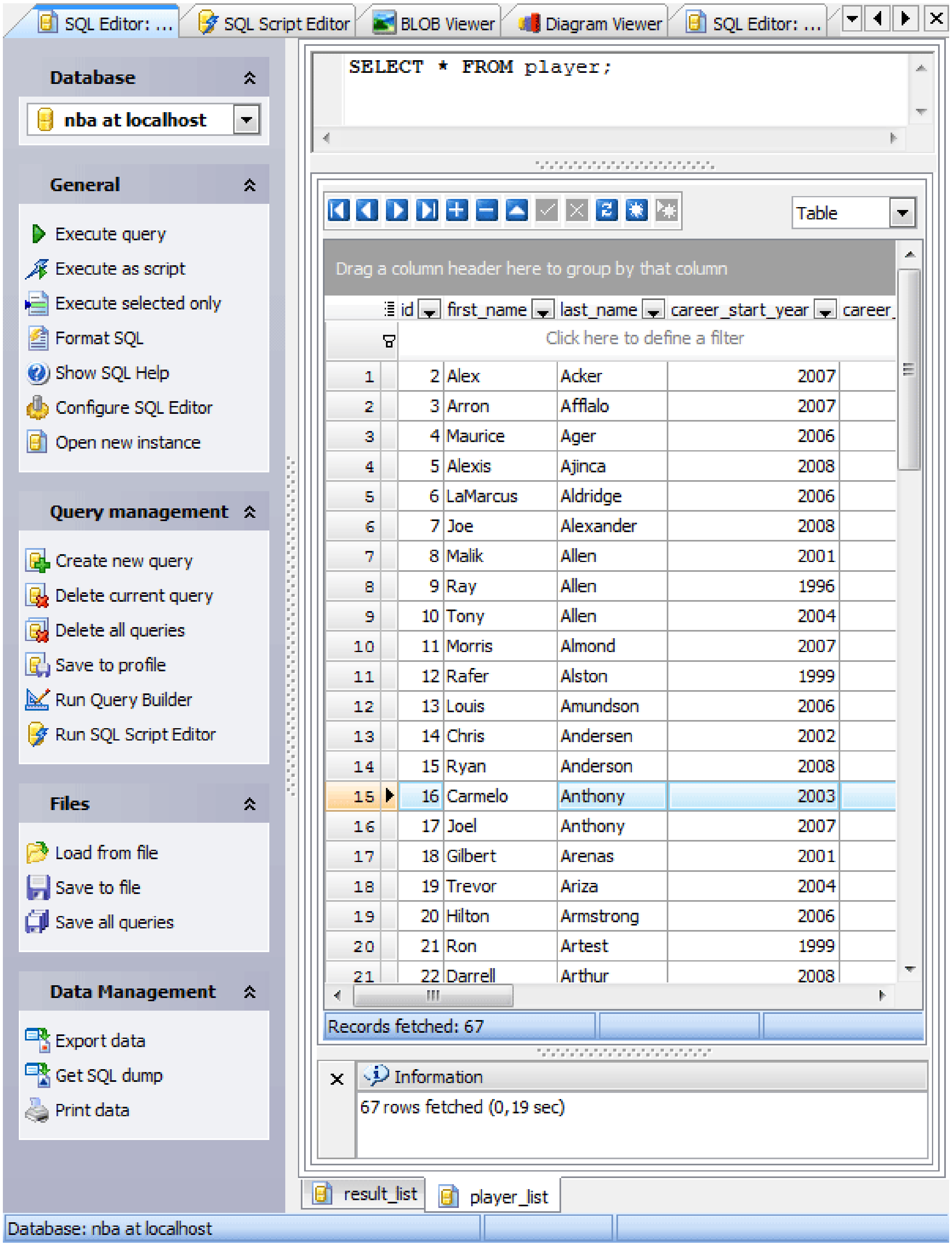
New version comes with support for a number of MySQL 5.7 new features, MySQLi based HTTP tunnel, nullable column checker, data export to JSON as well as with some other enhancements.

2.3.2 Tabbed MDI

MS SQL Maestro provides you with a possibility to choose (Options|Application) your favorite UI. Among the **classic MDI style** the **tabbed MDI style** is also available.

Applying the style you'll get all the objects editors opening on separate sheets. You can move from one sheet to another by clicking the sheet tabs at the bottom of the working area. The tab for the active sheet is underlined in the color you choose; tabs for inactive sheets are fully colored.

You can switch between the sheets with corresponding sheet tabs or using **Ctrl+Tab**. If you don't see the tool you want, click the tab scrolling buttons to display the tab, and then click the tab. You can also move the sheets.



```
SELECT * FROM player;
```

	id	first_name	last_name	career_start_year	career_
1	2	Alex	Acker	2007	
2	3	Arron	Afflalo	2007	
3	4	Maurice	Ager	2006	
4	5	Alexis	Ajinca	2008	
5	6	LaMarcus	Aldridge	2006	
6	7	Joe	Alexander	2008	
7	8	Malik	Allen	2001	
8	9	Ray	Allen	1996	
9	10	Tony	Allen	2004	
10	11	Morris	Almond	2007	
11	12	Rafer	Alston	1999	
12	13	Louis	Amundson	2006	
13	14	Chris	Andersen	2002	
14	15	Ryan	Anderson	2008	
15	16	Carmelo	Anthony	2003	
16	17	Joel	Anthony	2007	
17	18	Gilbert	Arenas	2001	
18	19	Trevor	Ariza	2004	
19	20	Hilton	Armstrong	2006	
20	21	Ron	Artest	1999	
21	22	Darrell	Arthur	2008	

Records fetched: 67

67 rows fetched (0,19 sec)

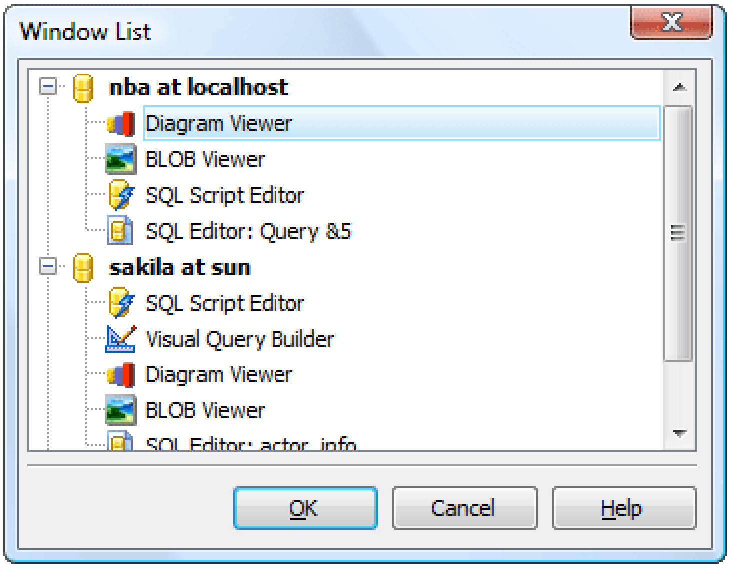
Database: nba at localhost

2.3.3 Switching between windows

The Window List dialog allows you to switch the child application windows quickly. To open the dialog select the Windows | Window List... item of the main menu or use the

Alt+0 hot keys combination.

Most of the windows are linked according to their active databases and displayed in the form of a tree, e.g. Table Editor, SQL Editor, Diagram Viewer, etc. Windows which are common for the entire program are shown as separate nodes of the tree.



To activate the window you need, select one of the window tree items and click the **OK** button.

2.4 Shortcut keys

The following table describes the default shortcut keys in MS SQL Maestro.

Interface 15	
Window list	**Alt+0**
Previous Window	**F6**
Next Window	**Ctrl+F6**
Show Database Explorer	**F11**
Refresh	**F5**
Exit	**Alt+F4**
MS SQL Maestro help	**F1**
Clipboard	
Cut	**Ctrl+X**
Copy	**Ctrl+C**
Paste	**Ctrl+V**
Select all	**Ctrl+A**
Find	**Ctrl+F**
Replace	**Ctrl+H**
Search again	**F3**
Undo	**Ctrl+Z**
Redo	**Shift+Ctrl+Z**
SQL Editors 313	
Open SQL Editor	**Ctrl+E**
Open SQL Script Editor	**Ctrl+R**
Open Visual Query Builder	**Ctrl+Q**
Execute query	**(F9) or (F8)**
Execute query as script	**(Shift+F9) or (Shift+F8)**
Execute selected only	**(Alt+F9) or (Alt +F8)**
Go to line	**Ctrl+G**
Format selected SQL	**Ctrl+Alt+F**
Create new query	**Ctrl+N**
Delete current query	**Ctrl+Alt+D**
Load script from file	**Ctrl+O**
Database management 22	
Create a new database profile	**Shift+Ctrl+P**
Edit an existing database profile	**Shift+Ctrl+E**
Rename a database profile (object)	**F2**
Remove database profile	**Shift+Ctrl+R**
Connect to the database	**Shift+Ctrl+C**
Disconnect from the database	**Shift+Ctrl+D**
Create a database object	**Shift+Ctrl+N**
Object Browser	**Shift+Ctrl+O**
Open BLOB Viewer	**Ctrl+B**

3 Databases and Database Profiles

MS SQL Maestro allows you to manipulate databases by means of database profiles. Profile contains database connection settings and a set of options to automatize common manipulations with databases (a possibility to connect to the database at MS SQL Maestro startup, login prompt before connection, etc.). To start working with databases in MS SQL Maestro, you should create database profile(s) first.

Use the following links for details:

How can I create a new database?

Use for this purpose Create Database Wizard 34. In order to run the wizard you should either

- select the Database | Create New Database... main menu item

or

- use the Create New Database... item of the popup menu.

Using Create Database Wizard set the Create profile after creating the database option to create a new profile and open the Database Profile Properties dialog after the database is created.

How can I change attributes of an existing database?

To edit a database:

- select the database to edit in the explorer tree;
- edit database properties within the appropriate tabs of Database Editor 39.

How can I drop an existing database?

In order to drop a database you should first select the database to drop in the explorer tree and establish connection (if you are not connected to the database yet), then either

- select the Database | Drop Database main menu item

or

- use the Drop Database item of the popup menu

and confirm dropping in the dialog window to complete the operation.

How can I create new database profiles?

In MS SQL Maestro database profiles are created within Create Database Profiles Wizard 24. In order to run the wizard you should either

- select the Database | Create Database Profiles... main menu item

or

- use the Create Database Profiles... item of the popup menu.

Using Create Database Profiles Wizard set the necessary connection and authorization options and click the Ready button to complete the operation.

How can I edit existing database profile options?

Database connection properties and profile options are edited within the Database Profile Properties 27 dialog window. In order to open the dialog for the selected database profile you should either

- select the Database | Edit Database Profile... main menu item

or

- use the Edit Database Profile... item of the popup menu.

How can I remove database profiles?

In order to remove a database profile you should first select the database profile in the explorer tree, then either select the Database | Remove Database Profile main menu item, or use the Remove Database Profile item of the popup menu and confirm removing profile in the dialog window to complete the operation.

How can I connect to a database?

In order to connect to a database you should first select the database in the explorer tree, then either

- select the Database | Connect to Database main menu item

or

- use the Connect to Database item of the popup menu.

How can I disconnect from a database?

In order to disconnect from a database you should first select the database in the explorer tree, then either

- select the Database | Disconnect from Database main menu item

or

- use the Disconnect from Database item of the popup menu.

3.1 Creating Database Profiles

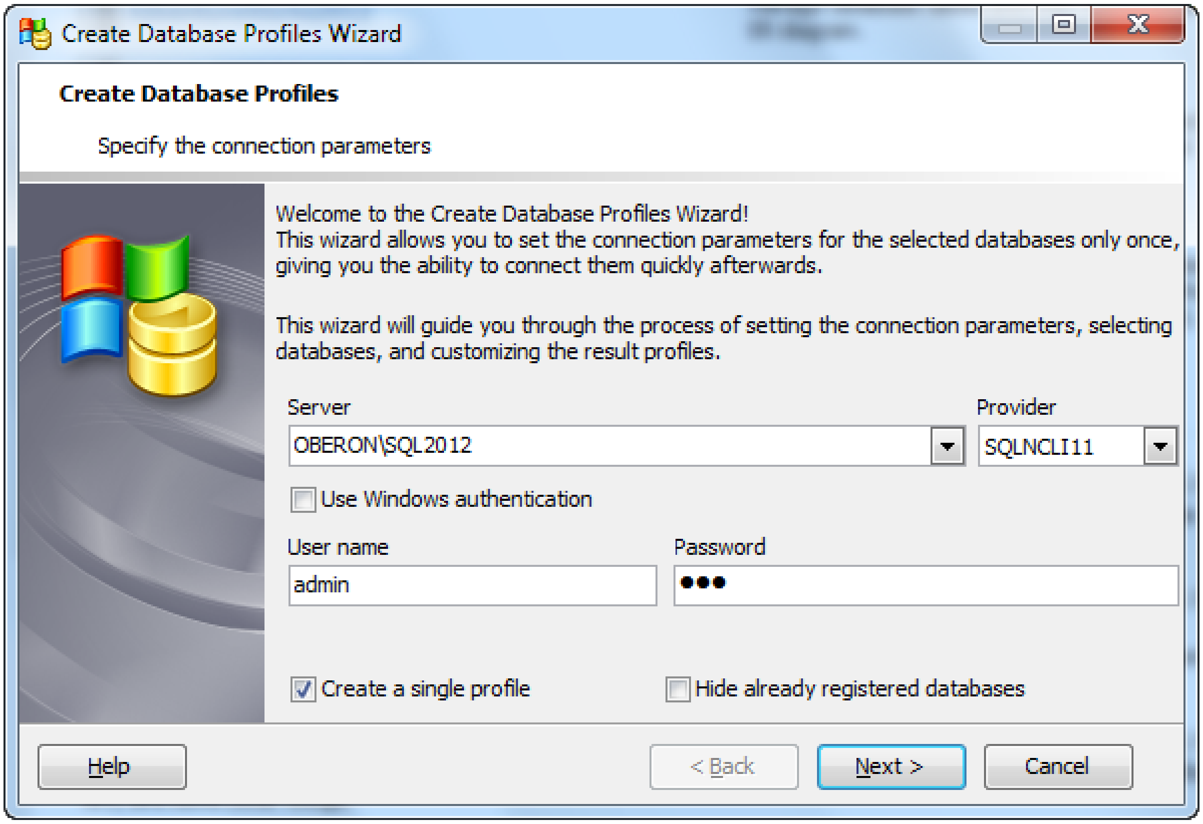
Create Database Profiles Wizard allows you to create a single database profile or several profiles from one host. To run the wizard, select the Database | Create Database Profiles... main menu item, or press the **Shift+Ctrl+P** hot keys combination. You can also use the Create Database Profiles button of the main toolbar.

- Set connection properties 24
- Specify database profile options 24

See also: Edit Database Profile Dialog 27

3.1.1 Setting connection properties

Specify Microsoft SQL connection properties 14 to be used on further connections.



Create Database Profiles Wizard

Create Database Profiles

Specify the connection parameters

Welcome to the Create Database Profiles Wizard!
This wizard allows you to set the connection parameters for the selected databases only once, giving you the ability to connect them quickly afterwards.

This wizard will guide you through the process of setting the connection parameters, selecting databases, and customizing the result profiles.

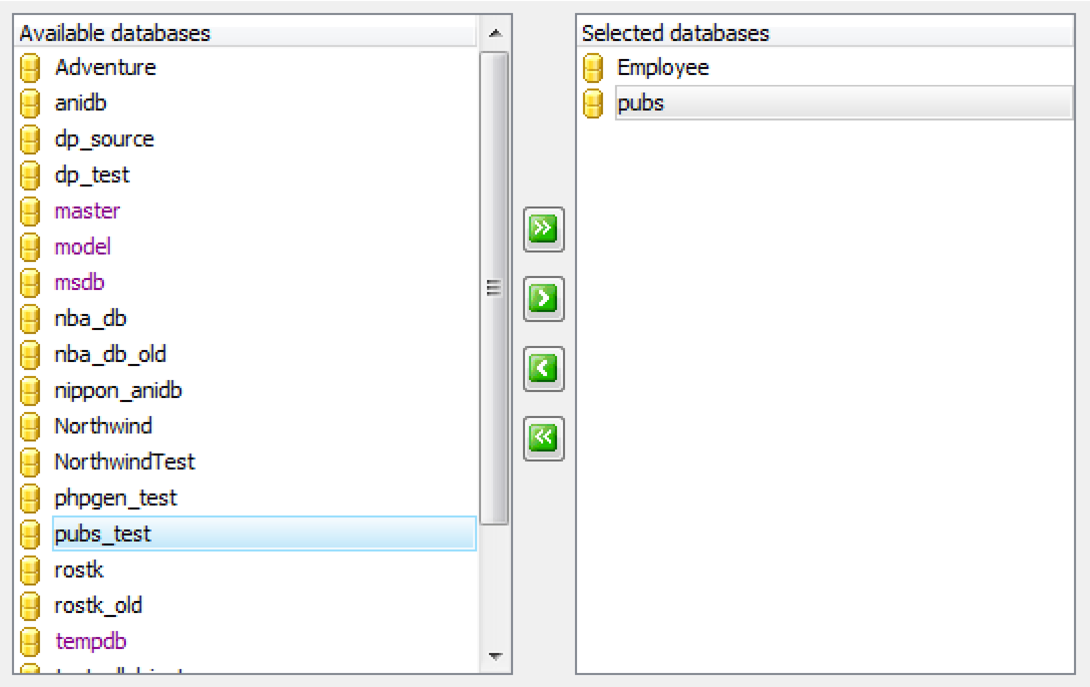
Server: OBERON\SQL2012
Provider: SQLNCLI11
☐ Use Windows authentication
User name: admin
Password: ●●●
☑ Create a single profile
☐ Hide already registered databases

☑ Check the Create a single profile option to set the database name manually and create a single profile for this database.

☑ Hide already registered databases
Check the box to shorten the databases list on the next wizard step.

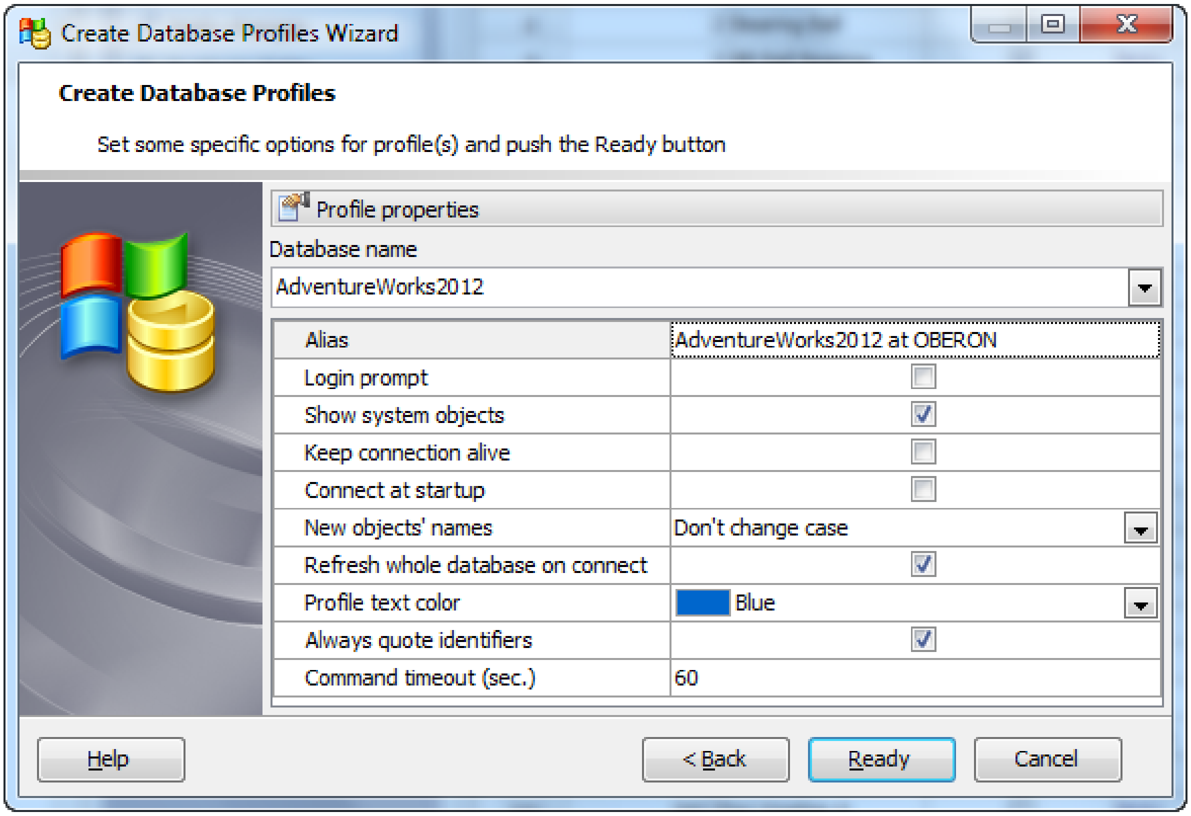
3.1.2 Setting profile options

To create a new profile, select databases to be registered by moving them from the Available databases list to the Selected databases list. This step is available if the Create a single profile option is unchecked on the previous wizard step.



Available databases	Selected databases
Adventure	Employee
anidb	pubs
dp_source	
dp_test	
master	
model	
msdb	
nba_db	
nba_db_old	
nippon_anidb	
Northwind	
NorthwindTest	
phpgen_test	
pubs_test	
rostk	
rostk_old	
tempdb	

You can select several databases to set options for all the selected databases at once (except the alias which should be unique for each individual database).



Create Database Profiles

Set some specific options for profile(s) and push the Ready button

Database name: AdventureWorks2012

Alias	AdventureWorks2012 at OBERON
Login prompt	☐
Show system objects	☑
Keep connection alive	☐
Connect at startup	☐
New objects' names	Don't change case
Refresh whole database on connect	☑
Profile text color	Blue
Always quote identifiers	☑
Command timeout (sec.)	60

☑ Login prompt before connection
Use the option to enable MS SQL Maestro to prompt for user name and password every time you connect to the database.

☑ Always quote identifiers
The checkbox enables quoting all the identifiers by the back quote symbol.
Uncheck to make SQL scripts obtained using the Extract Database Wizard 372 compatible with older versions of Microsoft SQL. Note that this option works only with those servers which support quoting aliases.

☑ Show system objects
Check the box to make system objects visible.

☑ Keep connection alive
Check the box for pinging server before each query execution.

☑ Connect at startup
With this option on connection to the profile database is automatically established at the application startup.

New objects' names (Don't change case, Convert to upper case, Convert to lower case)
The option allows you to specify the newly created objects case.

☑ Refresh whole database on connect
Use the option along with the Show empty schemas explorer options to hide/show empty schemas in the explorer tree.

Profile text color
Select the color to be used to represent the database profile name at the Explorer tree. For example this option may be useful to mark development and production databases in different colors in order to prevent casual metadata or data changes in the production.

Click the Ready button when done to start working with the selected databases in MS SQL Maestro.

3.2 Editing Database Profile

Use the Edit Database Profile dialog to edit the profile properties set on its creation. To open the dialog, select the database in the explorer tree, then select the Database | Edit Database Profile... main menu item or press the **Shift+Ctrl+E** hot key combination. You can also use the Edit Database Profile button of the main toolbar.

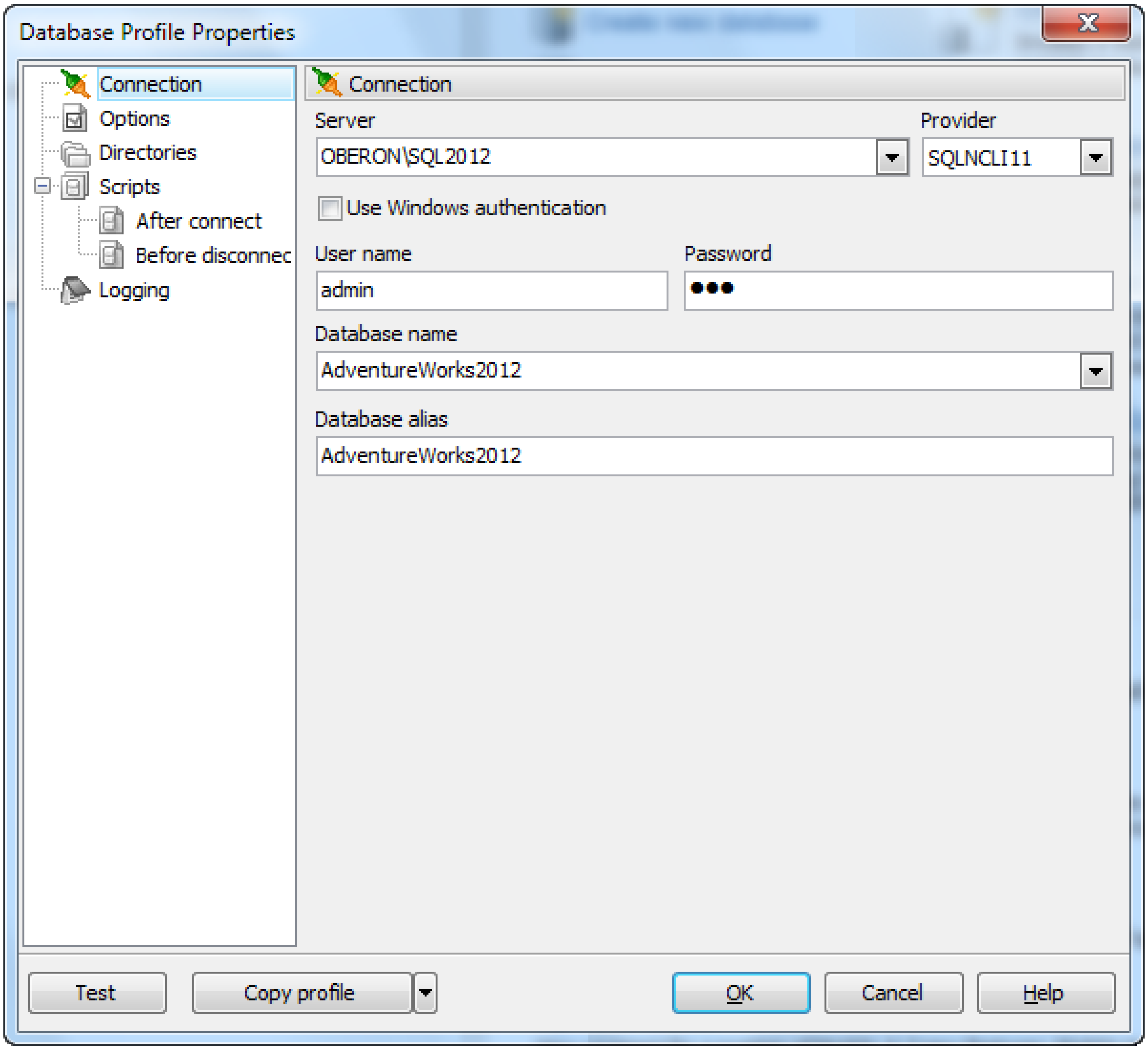
Instead of manual profile options editing you can copy all the options from the another existing profile with the Copy profile button.

- Editing database connection properties 27
- Settings database options 28
- Setting default directories for database tools 30
- Editing obligatory scripts to execute 31
- Setting log options and file names 31

See also: Create Database Profile Wizard 24

3.2.1 Editing connection properties

The tab allows you to change connection properties 14 of an existing database profile. Here you can change the database group, database info and edit the database alias, an optional name to display the database in the Explorer tree and in all the application tools.



Server: OBERON\SQL2012

Provider: SQLNCLI11

☐ Use Windows authentication

User name: admin

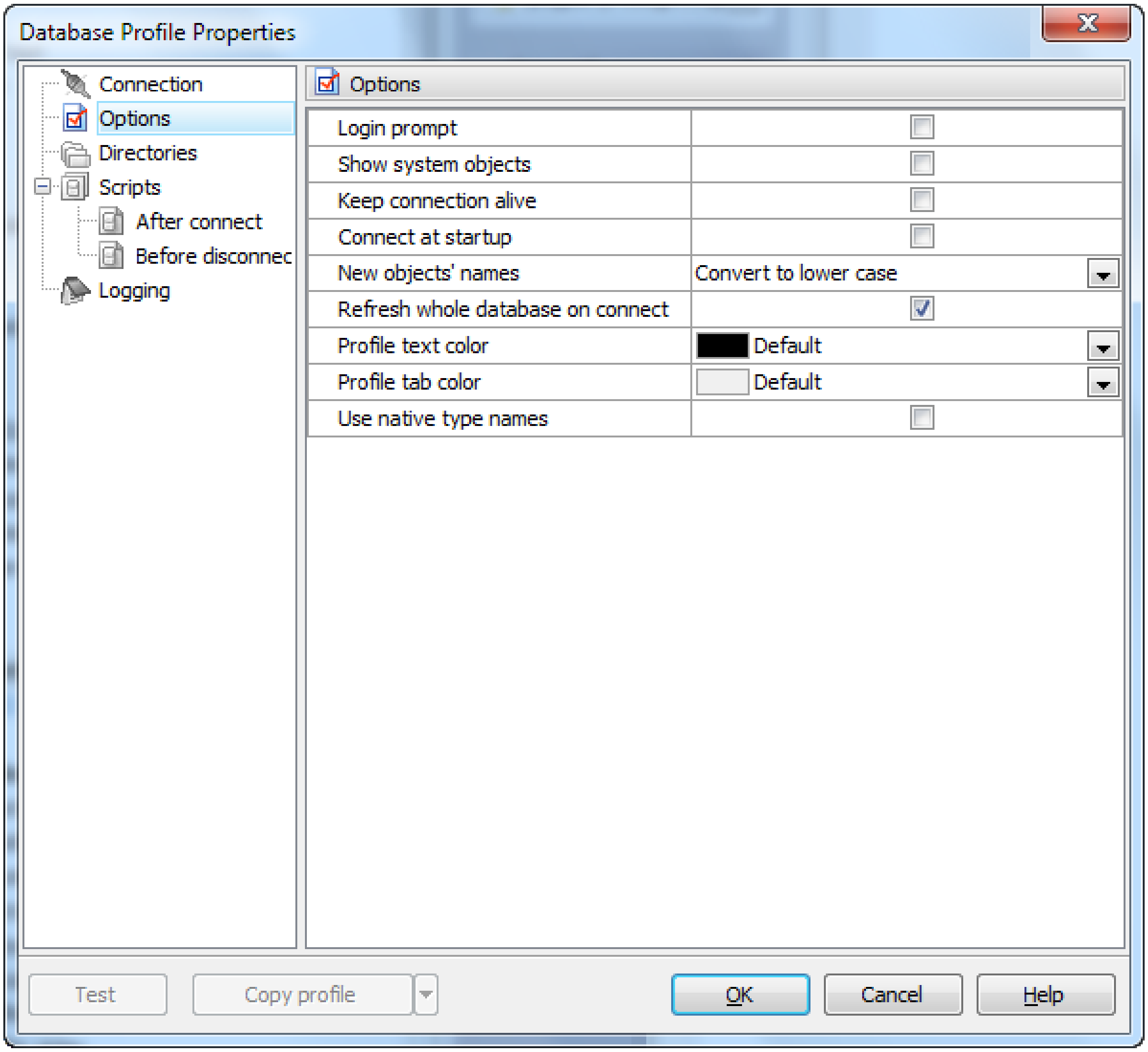
Password: ●●●

Database name: AdventureWorks2012

Database alias: AdventureWorks2012

3.2.2 Setting profile options

Customize database options according to your needs. The detailed description is given below.



Login prompt	☐
Show system objects	☐
Keep connection alive	☐
Connect at startup	☐
New objects' names	Convert to lower case
Refresh whole database on connect	☑
Profile text color	■ Default
Profile tab color	□ Default
Use native type names	☐

☑ **Login prompt**

Use the option to enable MS SQL Maestro to prompt for user name and password every time you connect to the database.

☑ **Show system objects**

Check the option to make system objects visible.

☑ **Keep connection alive**

Check the box for pinging server before each query execution.

☑ **Connect at startup**

With this option on connection to the profile database is automatically established at the application startup.

☑ **Always quote identifiers**

Enables quoting all the identifiers by the backquote symbol ('`').
Uncheck the option to make SQL scripts obtained using Extract Database Wizard 372 compatible with earlier versions of Microsoft SQL. Note that this option works only with

servers that support quoting aliases.

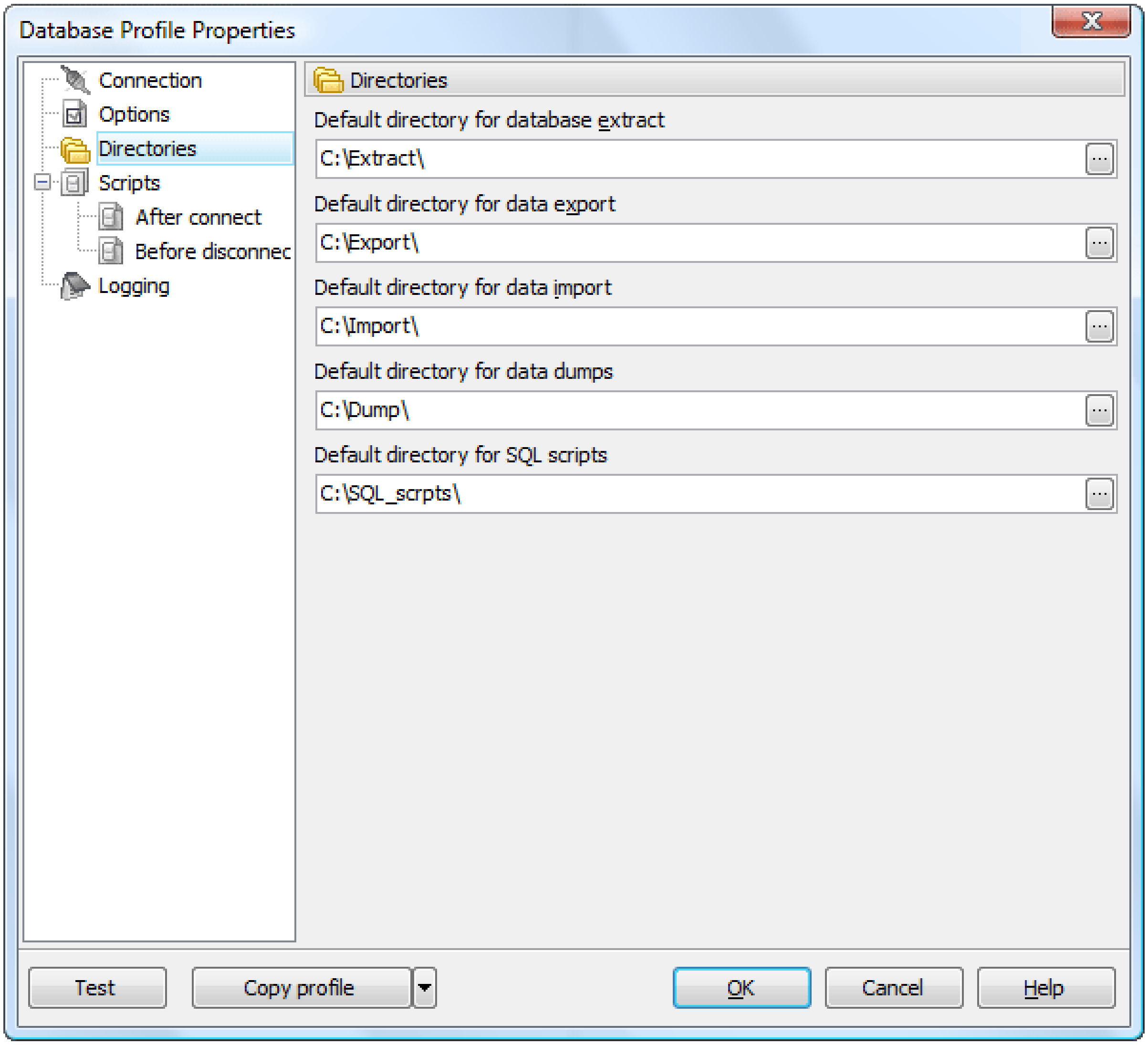
New objects' names (Don't change case, Convert to upper case, Convert to lower case)
Use the option to change the case for newly created objects.

☑ Refresh whole database on connect
Use the option along with the Show empty schemas[430] explorer options to hide/show empty schemas in the explorer tree.

You can also change here the font color the profile name is represented at the Explorer tree.

3.2.3 Setting default directories

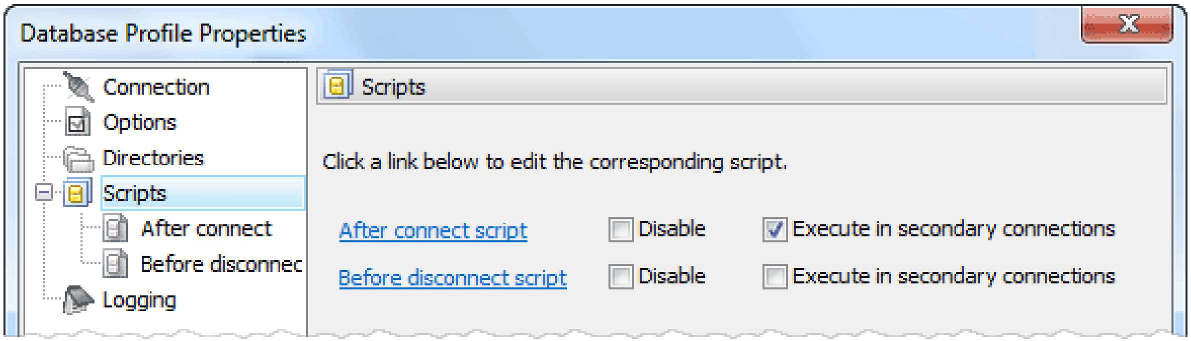
Use the tab to specify the default directories respectively for database extract, data export, data import, and data dump.



Default directory for database extract: C:\Extract\
Default directory for data export: C:\Export\
Default directory for data import: C:\Import\
Default directory for data dumps: C:\Dump\
Default directory for SQL scripts: C:\SQL_scrpts\

3.2.4 Editing obligatory scripts to execute

Use the tab to specify the obligatory scripts to execute in all database connections established by the software (on executing queries, browsing objects data, etc.). There is a possibility to enable/disable a written script.

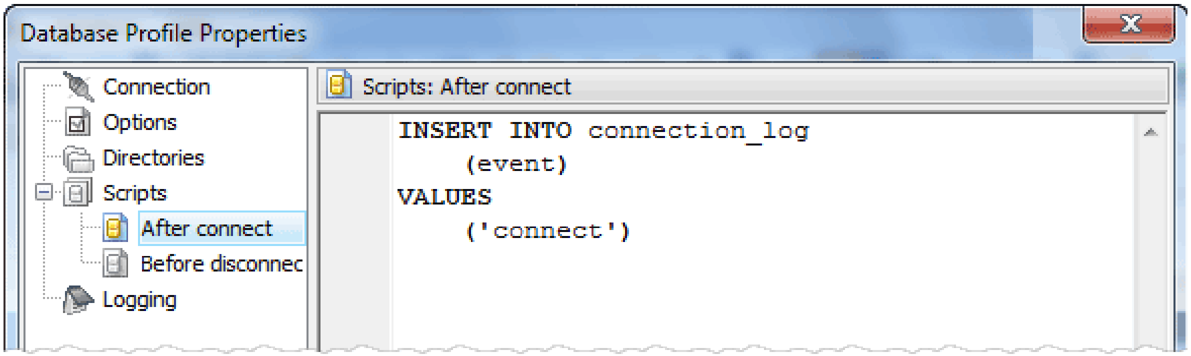


Click a link below to edit the corresponding script.

After connect script ☐ Disable ☑ Execute in secondary connections

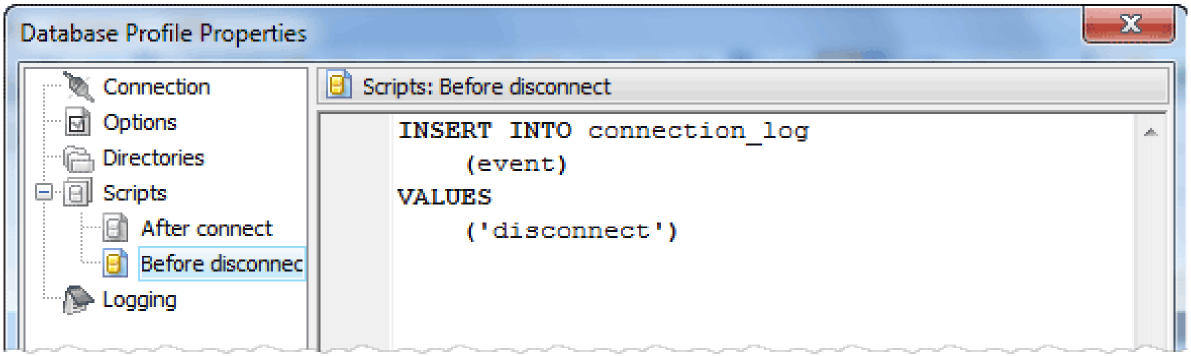
Before disconnect script ☐ Disable ☐ Execute in secondary connections

Below you can find an example of an obligatory script to execute after MS SQL Maestro will connect to the database. The script writes a connect time to the log table.



```
INSERT INTO connection_log
     (event)
VALUES
     ('connect')
```

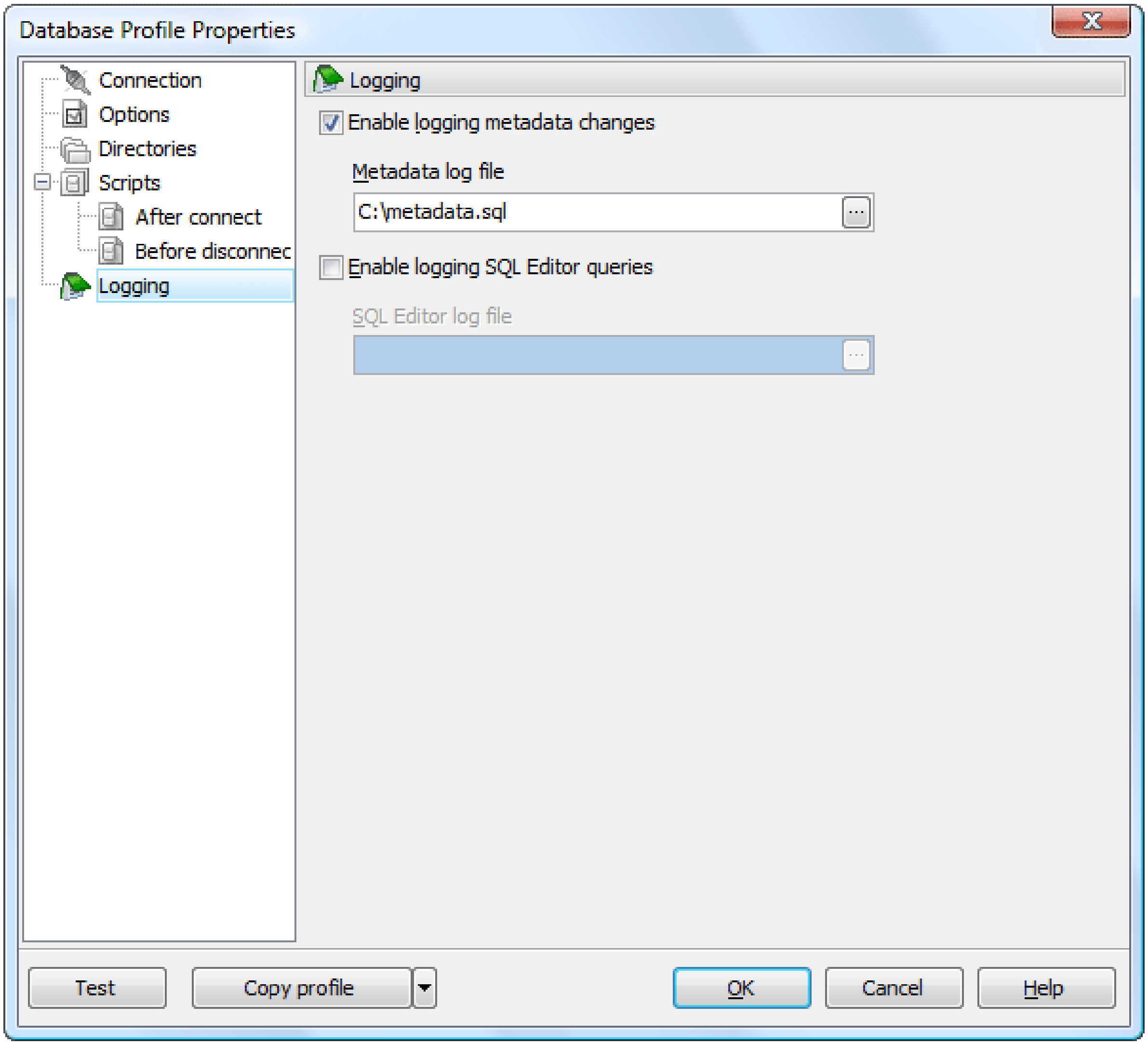
The next screen represents the example of an obligatory script to execute before MS SQL Maestro will disconnect from the database. The script writes a disconnect time to the log table.



```
INSERT INTO connection_log
     (event)
VALUES
     ('disconnect')
```

3.2.5 Setting log options

Enable/disable metadata changes logging and SQL query logging and specify the corresponding log file names if necessary.



☑ Enable logging metadata changes

Metadata log file

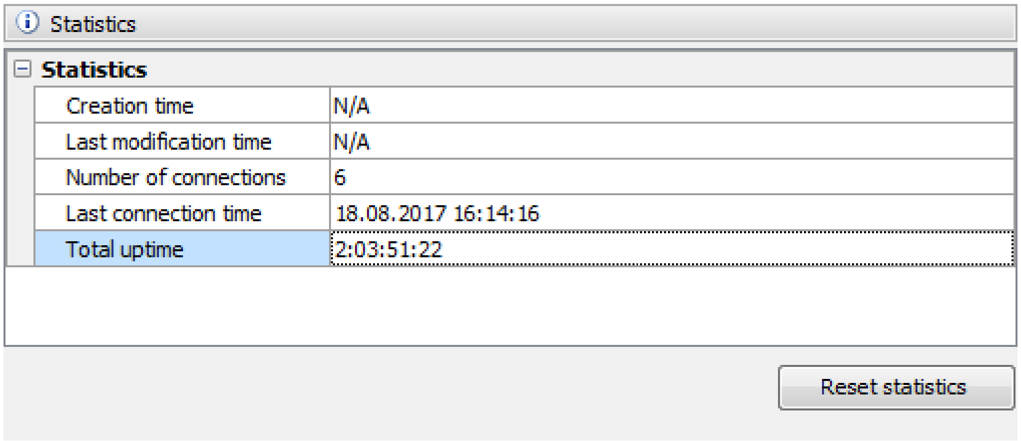
C:\metadata.sql

☐ Enable logging SQL Editor queries

SQL Editor log file

3.2.6 Statisitcs

This tab allows you to view usage statistics for the current profile. Click the **Reset Statistics** button to clear all the displayed values.



Statistics

Creation time	N/A
Last modification time	N/A
Number of connections	6
Last connection time	18.08.2017 16:14:16
Total uptime	2:03:51:22

3.3 Create Database Wizard

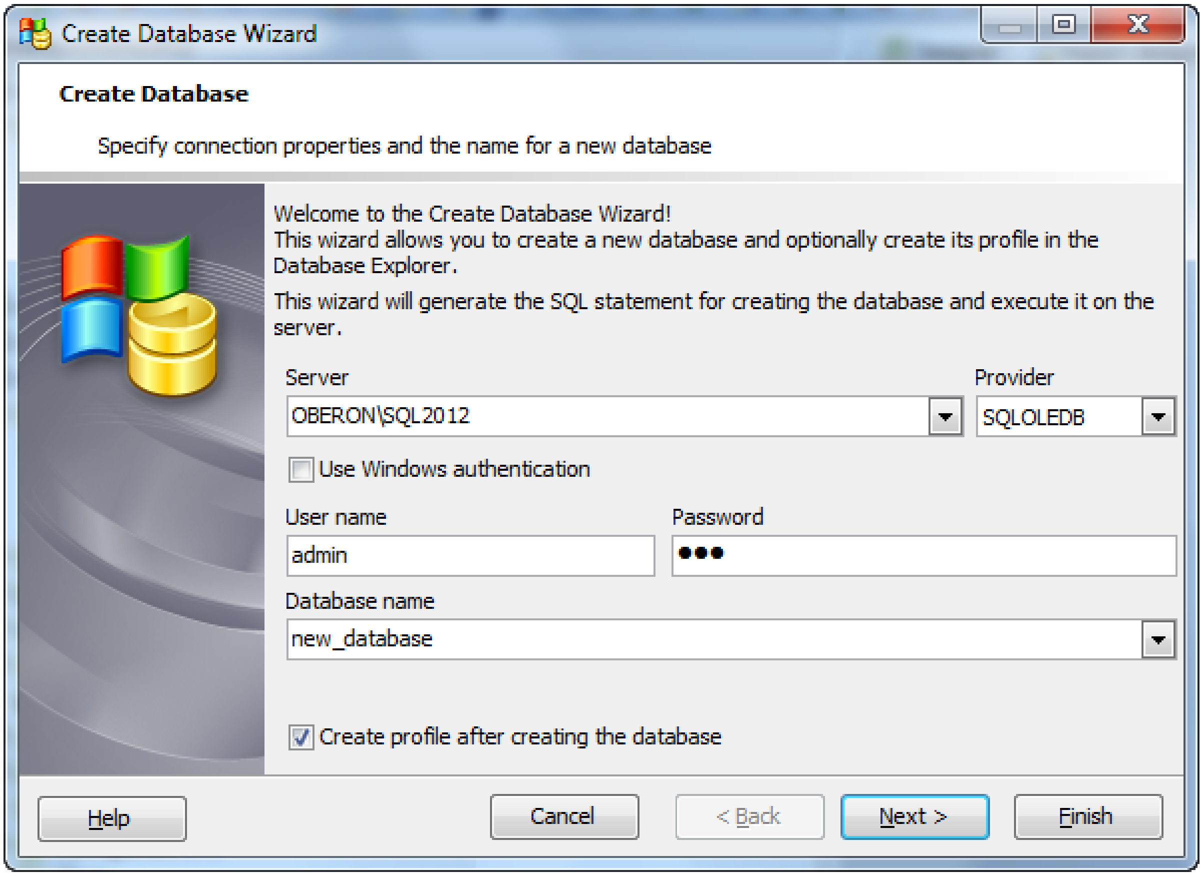
To run the Create Database Wizard, select the Database | Create New Database... main menu item or click the Create New Database button on the main toolbar.

- Setting database connection properties 34
- Managing database files 34
- Specifying database properties 35

See also: Database Editor 39

3.3.1 Setting connection properties

Set the connection properties 14 of the new database. If the Create profile after creating the database option is checked, the Edit Database Profile 27 dialog is opened after the new database is created.



Create Database

Specify connection properties and the name for a new database

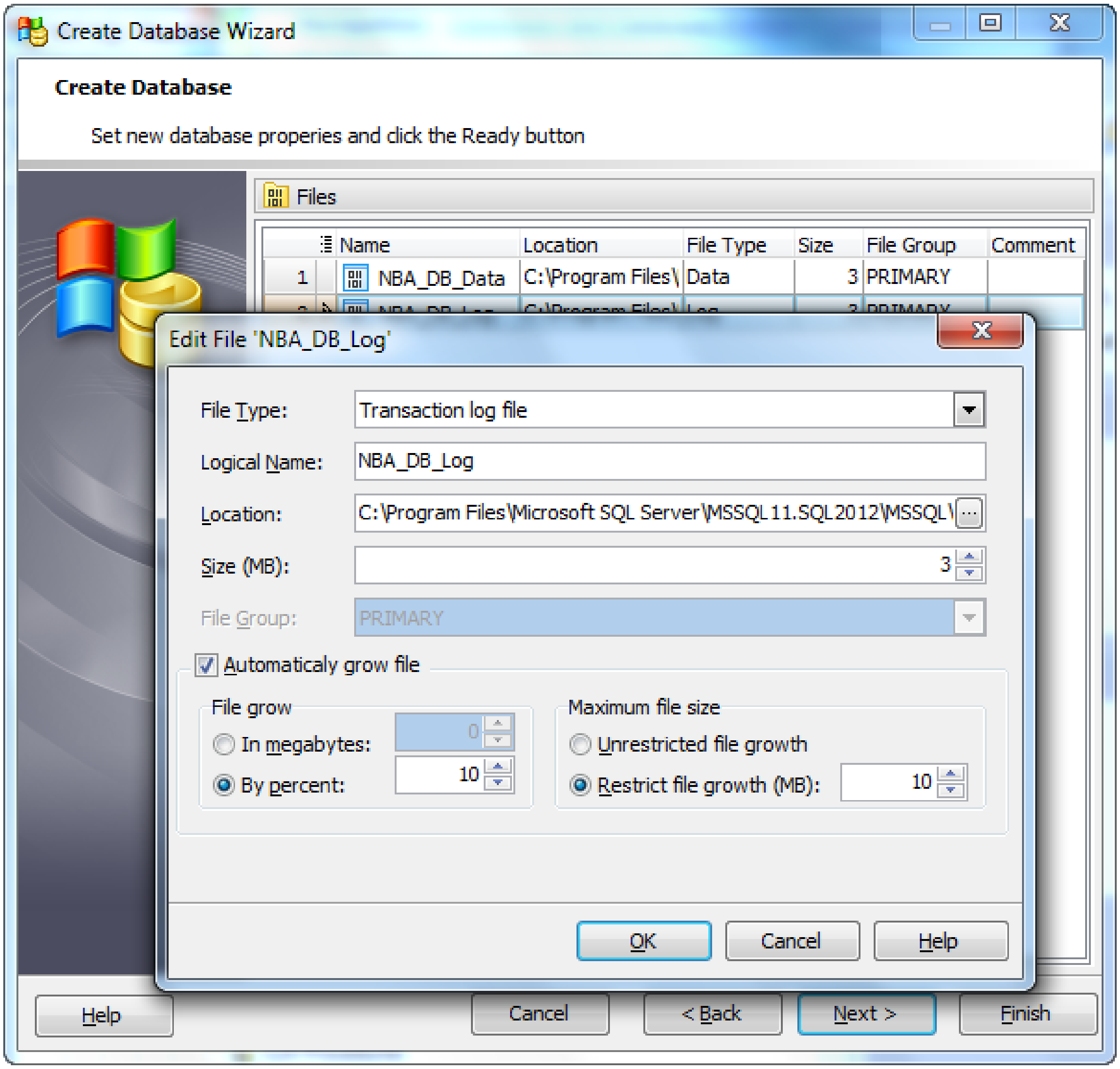
Welcome to the Create Database Wizard!
This wizard allows you to create a new database and optionally create its profile in the Database Explorer.

This wizard will generate the SQL statement for creating the database and execute it on the server.

Server: OBERON\SQL2012
Provider: SQLOLEDB
☐ Use Windows authentication
User name: admin
Password: ●●●
Database name: new_database
☑ Create profile after creating the database

3.3.2 Managing database files

The second wizard step allows you to manage database files.



Create Database

Set new database properies and click the Ready button

	Name	Location	File Type	Size	File Group	Comment
1	NBA_DB_Data	C:\Program Files\	Data	3	PRIMARY	

Edit File 'NBA_DB_Log'

File Type: Transaction log file
Logical Name: NBA_DB_Log
Location: C:\Program Files\Microsoft SQL Server\MSSQL11.SQL2012\MSSQL\
Size (MB): 3
File Group: PRIMARY

☑ Automaticaly grow file

File grow
○ In megabytes: 0
● By percent: 10

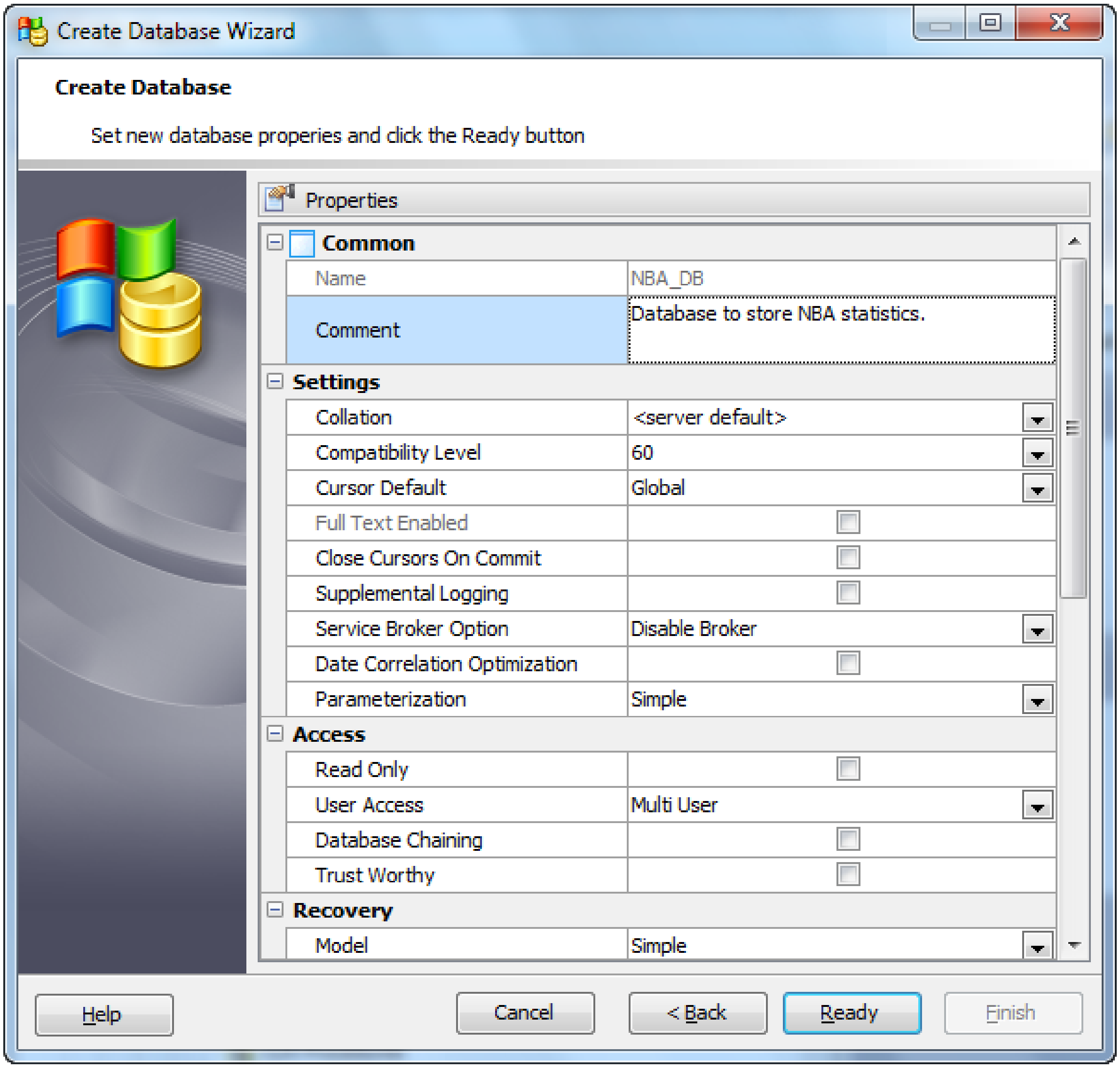
Maximum file size
○ Unrestricted file growth
● Restrict file growth (MB): 10

Click the Add button to add a new file and set its properties in the File Editor 208. Click the Edit button to modify the selected file, or the Delete button to remove one (alternatively, you can use the corresponding popup menu items).

Click Next after you have added all the database files. Click Finish to create the database without setting special properties.

3.3.3 Specifying database properties

The next wizard step allows you to set common database options. All fields below are optional, i.e. it is not obligatory for you to fill them.



Create Database

Set new database properies and click the Ready button

Common	
Name	NBA_DB
Comment	Database to store NBA statistics.
Settings	
Collation	<server default>
Compatibility Level	60
Cursor Default	Global
Full Text Enabled	☐
Close Cursors On Commit	☐
Supplemental Logging	☐
Service Broker Option	Disable Broker
Date Correlation Optimization	☐
Parameterization	Simple
Access	
Read Only	☐
User Access	Multi User
Database Chaining	☐
Trust Worthy	☐
Recovery	
Model	Simple

Compatibility Level

Sets the version of SQL Server for the database to be compatible with (60 = SQL Server 6.0, 65 = SQL Server 6.5, 70 = SQL Server 7.0, 80 = SQL Server 2000, 90 = SQL Server 2005).

Cursor Default *(GLOBAL, LOCAL)*

Controls whether cursor scope uses LOCAL or GLOBAL. When LOCAL is specified and a cursor is not defined as GLOBAL when created, the scope of the cursor is local to the batch, the stored procedure, or the trigger in which the cursor was created. The cursor name is valid within this scope only. When GLOBAL is specified and a cursor is not defined as LOCAL when created, the scope of the cursor is global to the connection. The cursor name can be referenced in any stored procedure or a batch executed by the connection.

Full Text Enabled

Enables full text support.

Close Cursors On Commit
If checked, all currently open cursors are closed when a transaction is committed or rolled back. Otherwise, cursors remain open when a transaction is committed; rolling back a transaction closes any cursors except those defined as INSENSITIVE or STATIC.

Read Only
Users can read data from the database but not modify it.

User Access *(SINGLE USER, RESTRICTED USER, MULTI USER)*
Controls user access to the database. If single user is specified, only one user at a time can access the database. Restricted user allows members of the db_owner fixed database role and dbcreator and sysadmin fixed server roles only to connect to the database, but does not limit their number. If multi user is selected, all users that have the appropriate permissions to connect to the database are allowed.

Torn page detection
The option allows you to enable or disable verifying the database pages to find the corrupted ones. In Microsoft SQL Server 2005 the option is called Page Verify and admits to the following values: Checksum, Torn page detection, None.

Ansi Null Default
Determines the default value, NULL or NOT NULL, of a column or user-defined data type for which the nullability is not explicitly defined in CREATE TABLE or ALTER TABLE statements. If checked, the default value is NULL.

Ansi Nulls
If checked, all comparisons to a null value evaluate to UNKNOWN. Otherwise, comparisons of non-UNICODE values to a null value evaluate to TRUE if both values are NULL.

Ansi Warnings
If checked, errors or warnings are issued when conditions such as divide-by-zero occur or null values appear in aggregate functions.

Recursive Triggers
If checked, recursive firing of AFTER triggers is allowed.

Quoted Identifiers
If checked, double quotation marks can be used to enclose delimited identifiers. Otherwise, identifiers cannot be in quotation marks and must follow all Transact-SQL rules for identifiers. Literals can be delimited by either single or double quotation marks.

Null Concat
If checked, the result of a concatenation operation is NULL when either operand is NULL.

Arithmetic Abort
If checked, a query is ended when an overflow or divide-by-zero error occurs during query execution. Otherwise, a warning message is displayed when one of these errors occurs, but the query, batch, or transaction continues to process as if no error occurred.

Auto Close
If checked, the database is shut down cleanly and its resources are set free after the

last user exits.

Auto Create Statistics
If checked, any missing statistics required by a query for optimization are automatically built during query optimization. Otherwise, statistics must be manually created.

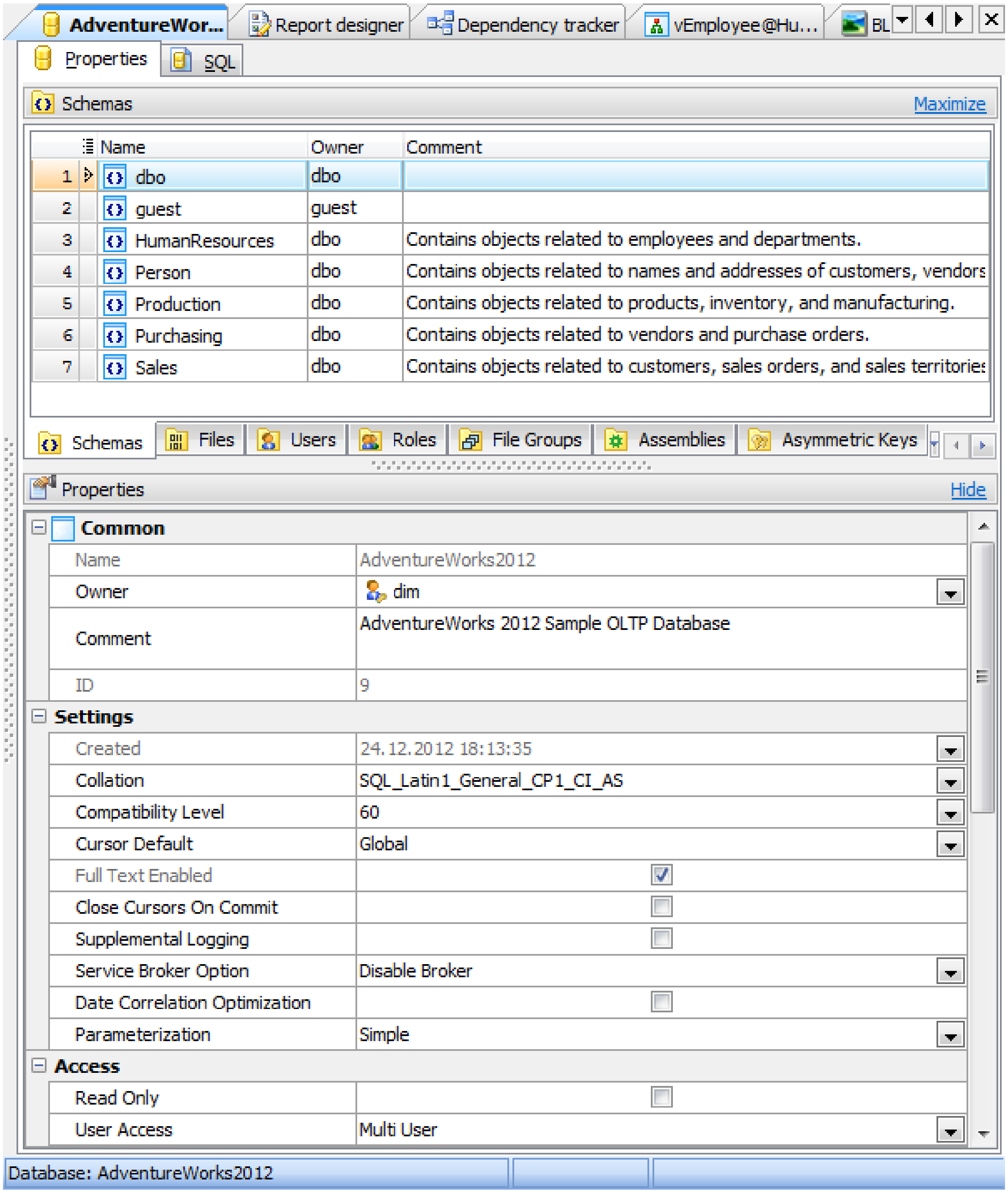
Auto Shrink
If checked, the database files are automatically shrunk during periodic checks for unused space.

Auto Update Statistics
If checked, any out-of-date statistics required by a query for optimization are automatically updated during query optimization. Otherwise, statistics must be manually updated.

3.4 Database Editor

Database Editor allows you to browse, add, edit and delete all objects of the selected database and its main properties.

To open the editor, use popup menu of the database node at the Explorer tree.



	Name	Owner	Comment
1	dbo	dbo	
2	guest	guest	
3	HumanResources	dbo	Contains objects related to employees and departments.
4	Person	dbo	Contains objects related to names and addresses of customers, vendors
5	Production	dbo	Contains objects related to products, inventory, and manufacturing.
6	Purchasing	dbo	Contains objects related to vendors and purchase orders.
7	Sales	dbo	Contains objects related to customers, sales orders, and sales territories

Common	
Name	AdventureWorks2012
Owner	dim
Comment	AdventureWorks 2012 Sample OLTP Database
ID	9
Settings	
Created	24.12.2012 18:13:35
Collation	SQL_Latin1_General_CP1_CI_AS
Compatibility Level	60
Cursor Default	Global
Full Text Enabled	☑
Close Cursors On Commit	☐
Supplemental Logging	☐
Service Broker Option	Disable Broker
Date Correlation Optimization	☐
Parameterization	Simple
Access	
Read Only	☐
User Access	Multi User

Database: AdventureWorks2012

Subitems

Every tab is intended for managing corresponding database objects (e.g. *tables, views, queries*, etc.). Open the object in its editor by double-clicking or pressing the **Enter** key. The popup menu allows you to create new, edit or drop the selected database objects. Using this menu you can also create a copy of the object.

You can operate on several objects at a time. For this you have to select database objects with the **Shift** or the **Ctrl** key pressed. After the group of objects is selected, you can operate on it, e.g. delete several objects at once, as it was a single object.

The Properties tab displays available database parameters. Below you can find some of their descriptions.

Comment
This field stores a comment to the database.

Created
Displays the date when the database was created.

Collation
Specifies the default collation for the database. Collation name can be either a Windows collation name or a SQL collation name.

Compatibility Level
Sets the version of SQL Server for the database to be compatible with (60 = SQL Server 6.0, 65 = SQL Server 6.5, 70 = SQL Server 7.0, 80 = SQL Server 2000, 90 = SQL Server 2005).

Cursor Default *(GLOBAL, LOCAL)*
Controls whether cursor scope uses *LOCAL* or *GLOBAL*. When *LOCAL* is specified and a cursor is not defined as *GLOBAL* when created, the scope of the cursor is local to the batch, the stored procedure, or the trigger in which the cursor was created. The cursor name is valid within this scope only. When *GLOBAL* is specified and a cursor is not defined as *LOCAL* when created, the scope of the cursor is global to the connection. The cursor name can be referenced in any stored procedure or batch executed by the connection.

Full Text Enabled
Enables full text support.

Close Cursors On Commit
If checked, all currently open cursors are closed when a transaction is committed or rolled back. Otherwise, cursors remain open when a transaction is committed; rolling back a transaction closes any cursors except those defined as *INSENSITIVE* or *STATIC*.

Read Only
Users can only read data from the database but not modify it.

User Access *(SINGLE USER, RESTRICTED USER, MULTI USER)*
Controls user access to the database. If *single user* is specified, only one user at a time can access the database. *Restricted user* allows only members of the *db_owner* fixed database role and *dbcreator* and *sysadmin* fixed server roles only to connect to the database, but does not limit their number. If *multi user* is selected, all users that have the appropriate permissions to connect to the database are allowed.

Model *(FULL, BULK LOGGED, SIMPLE)*
Controls database recovery options and disk I/O error checking.
A *full* backup strategy provides full recovery after media failure by using transaction log backups. If a data file is damaged, media recovery can restore all committed transactions.
A *bulk logged* backup strategy provides recovery after media failure by combining the best performance and the least amount of log-space use for certain large-scale or bulk operations.
A *simple* backup strategy uses minimal log space. Log space can be automatically reused when it is no longer required for server failure recovery.

Page Verify *(CHECKSUM, TORN_PAGE_DETECTION, NONE)*
You are provided with three options to discover damaged database pages caused by disk I/O path errors. Disk I/O path errors can be the cause of database corruption problems and are usually the result of power failures or disk hardware failures that occur at the time the page is actively being written to disk.
A *checksum* is calculated over the contents of the entire page and stored in the page header when a page is written to disk. A checksum failure is an indication of an I/O path problem and requires investigation of your hardware, firmware drivers, BIOS, filter drivers (for example, virus software), and other I/O path components to determine the root cause.
A *torn page detection* is a specific bit that is saved for each 512-byte sector in the 8-kilobyte (KB) database page and stored in the database page header when the page is written to disk. When the page is read from disk, the torn bits stored in the page header are compared to the actual page sector information. Should the values not match, it is an indication that only part of the page was written to disk. Torn pages are usually detected by database recovery if it is truly an incomplete write of a page. However, other I/O path failures can express themselves as a torn page at any time.
If the page verify is *none*, database page records will not generate a CHECKSUM or TORN PAGE DETECTION value.

Ansi Null Default
Determines the default value (*NULL* or *NOT NULL*) of a column or user-defined data type for which the nullability is not explicitly defined in CREATE TABLE or ALTER TABLE statements. If checked, the default value is *NULL*.

Ansi Nulls
If checked, all comparisons to a null value evaluate to *UNKNOWN*. Otherwise, comparisons of non-UNICODE values to a null value evaluate to *TRUE* if both values are *NULL*.

Ansi Padding
If checked, strings are padded to the same length before conversion or inserting to a varchar or nvarchar data type.

Ansi Warnings
If checked, errors or warnings are issued when conditions such as divide-by-zero occur or null values appear in aggregate functions.

Recursive Triggers
If checked, recursive firing of AFTER triggers is allowed.

Quoted Identifiers

If checked, double quotation marks can be used to enclose delimited identifiers. Otherwise, identifiers cannot be in quotation marks and must follow all Transact-SQL rules for identifiers. Literals can be delimited by either single or double quotation marks.

Numeric Round Abort

If checked, an error is generated when some loss of precision occurs in an expression. Otherwise, losses of precision do not generate error messages and the result is rounded to the precision of the column or variable storing the result.

Null Concat

If checked, the result of a concatenation operation is *NULL* when either operand is *NULL*.

Arithmetic Abort

If checked, a query is ended when an overflow or divide-by-zero error occurs during query execution. Otherwise, a warning message is displayed when one of these errors occurs, but the query, batch, or transaction continues to process as if no error occurred.

Auto Close

If checked, the database is shut down cleanly and its resources are set free after the last user exits.

Auto Create Statistics

If checked, any missing statistics required by a query for optimization are automatically built during query optimization. Otherwise, statistics must be manually created.

Auto Shrink

If checked, the database files are automatically shrunk during periodic checks for unused space.

Auto Update Statistics

If checked, any out-of-date statistics required by a query for optimization are automatically updated during query optimization. Otherwise, statistics must be manually updated.

4 Database Object Management

MS SQL Maestro provides you with several tools to manage and navigate Microsoft SQL objects. To browse and modify objects, at least one connection to a database should be established.

- Browse Database Objects 61
- Create New Objects 44
- Edit Existing Objects 48
- Duplicate Objects 56

The options to create or edit an object in MS SQL Maestro follow the parameters defined by Microsoft SQL. If you need clarification on what an option means or how it should be used, see Microsoft SQL's documentation for more information. The documentation provides detailed description of objects, including their purpose, properties, and restrictions. The MS SQL Maestro manual provides you with only brief review of Microsoft SQL objects.

4.1 Create Objects

MS SQL Maestro provides a number of Create Object Wizards 45 to accomplish the most facile Microsoft SQL object creation.

There are several ways to invoke the necessary Create Object Wizard:

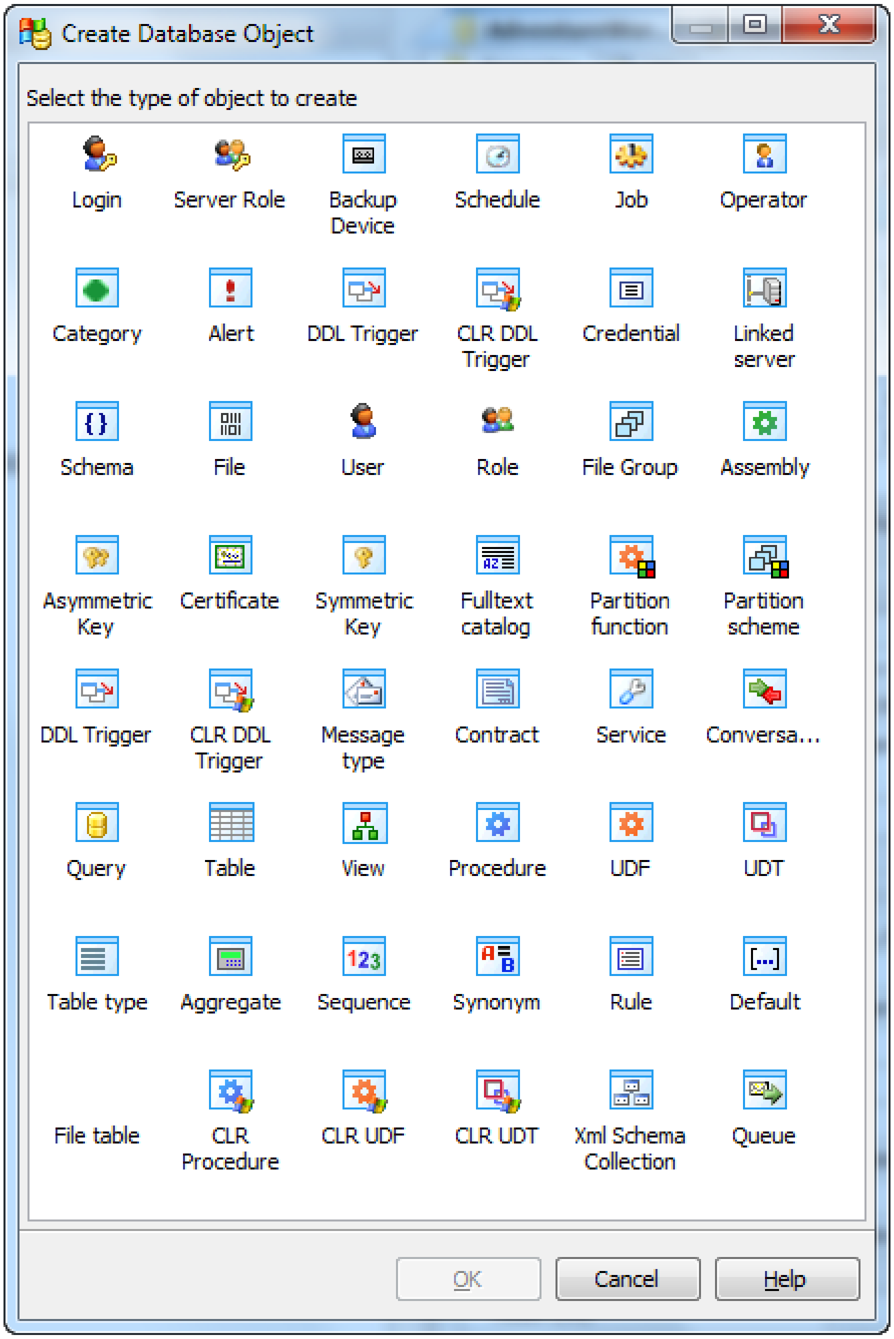
- select the Object | Create Database Object... main menu item;
- select the necessary icon (table, query, view, trigger, etc.) in the Create Database Object 44 dialog

or

- select the object list (Tables, Views, Triggers, etc.) or any object from that list in the Explorer tree (Object Manager and Object Browser);
- select the Create New Table (View, Trigger, etc.)... item from the popup menu or press **Insert**.

4.1.1 Create Database Object Dialog

The Create Database Object dialog allows you to create any type of database object supported by MS SQL Maestro. To open the dialog select the Object | Create Database Object... main menu item or use the **Shift+Ctrl+N** hot keys combination. Select an object type icon and click the OK button to invoke the corresponding wizard or dialog.



4.1.2 Overview of Create Objects Wizards

Several steps of Create Object Wizards are common for all of them. This part purpose is the formulation of the basic principles for the Create Object Wizard organization.

- On the first wizard step 46 you need to specify the new object name.
- On the second one you have to define all the object properties. To clear up the object properties meanings see the appropriate topic of the respective Create Object Wizard section.

- Some objects has subitems (e.g. each table contains fields, indexes, procedures have parameters, etc). In this case the next step allows you to manage such subobjects of the object being created. We recommend you to store the following shortcuts in order to speed your work: the Ins key adds a new subobject, the Enter key displays the subobject's editor, and the Del key drops the subobject.

- The next wizard step 47 is final. It is provided to sum up the Create Object Wizard operation.

Note: There are some objects to have an additional Create Object Wizard steps. The detailed description of the steps you can find at the appropriate topic of the corresponding section.

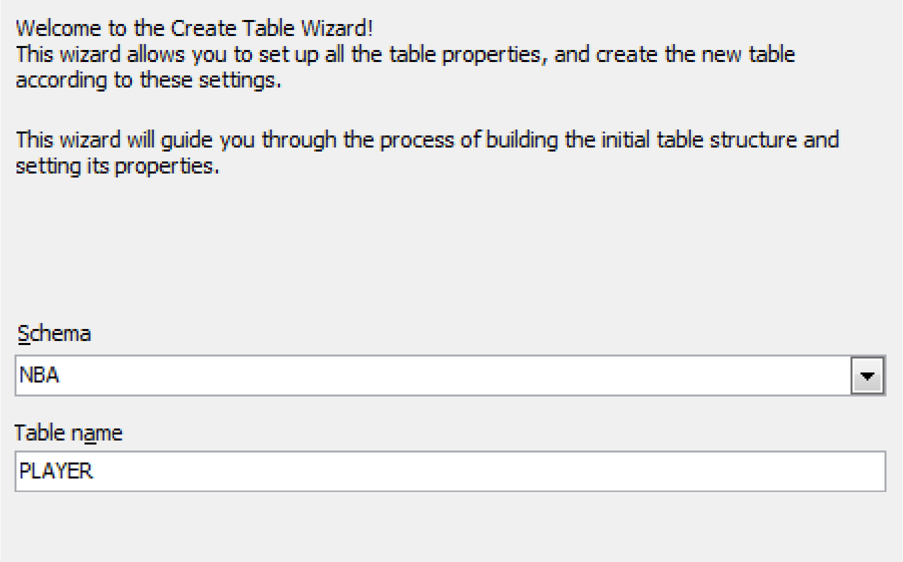
See also:

- Create Schema Wizard 71
- Create Table Wizard 77
- Create View Wizard 109
- Create UDF Wizard 129
- Create UDT Wizard 137
- Create Aggregate Wizard 193
- Creation of a New Query 311
- Create User Wizard 201
- Create Role Wizard 205
- Create File Wizard 208
- Create File Group Wizard 211
- Create Assembly Wizard 215
- Create Symmetric Key Wizard 223
- Create Asymmetric Key Wizard 219
- Create Certificate Wizard 226
- Create Procedure Wizard 119
- Create Synonym Wizard 142
- Create Rule Wizard 146
- Create Default Wizard 151
- Create CLR Procedure Wizard 157
- Create CLR UDF Wizard 162
- Create CLR UDT Wizard 167
- Create Xml Schema Collection Wizard 170
- Create Queue Wizard 174
- Create Message Type Wizard 180
- Create Contract Wizard 183
- Create Service Wizard 186
- Create Conversation Wizard 189

4.1.2.1 Setting object name

Select the container (table, schema, database, etc.) for the new object from the list of available containers and enter the new object name in the respective box.

Note: the name of the object must be unique among all the object names in its container. Moreover, all the objects that are source of data need unique names among themselves. You can use any identifier that is allowed by Microsoft SQL server.


Welcome to the Create Table Wizard!
This wizard allows you to set up all the table properties, and create the new table according to these settings.

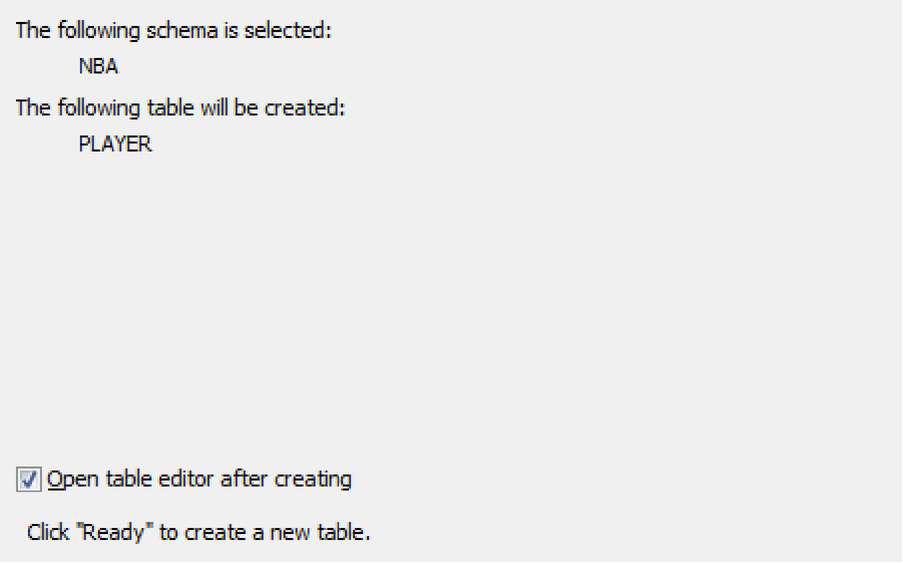
This wizard will guide you through the process of building the initial table structure and setting its properties.

Schema: NBA

Table name: PLAYER

4.1.2.2 Viewing common information

At this step common information about the object to be created is displayed. Select the Open object editor after creating option to open the appropriate Object Editor after the new object is created. Click the Ready button to complete creation of the object.


The following schema is selected:
NBA
The following table will be created:
PLAYER

☑ Open table editor after creating

Click "Ready" to create a new table.

4.2 Edit Objects

MS SQL Maestro allows you to view and modify existing database objects in several ways:

- edit object comment with the Describe Object 55 dialog;
- briefly view and modify object properties 54;
- view and modify the object including subitems within the object editor.

To open an Object Editor 48, just double click its node in the Database Explorer tree. Of course this action is also available through popup menus, navigation bars, and so on.

4.2.1 Overview of Object Editors

Database Object Editors are the basic MS SQL Maestro tools for working with existing objects. The proper editor can be opened automatically after the object is created. You can also open the necessary object editor with the corresponding items of popup menus of the Explorer Tree 61, Object Manager 66 or Object Browser 65.

The editors consist of a several tabs. Some tabs are similar for all editors. This part purpose is to formulate the basic principles of all Object Editors in MS SQL Maestro.

- To edit object options such as name, owner, etc. use the Properties tab. To understand an option, see the appropriate topic of the corresponding Object Editor manual section and Microsoft SQL documentation.

  This tab also allows you to manage objects belonging to the selected one. To reset any tab to default settings, open it when holding the **Ctrl** key.

- Use the Permissions 49 tab to manage access privileges (grants) of the corresponding object.

- In a similar manner, some objects called grantees (e.g. users or roles) can have rights to do something with other objects (e.g. a user can read data from a table). This relationship can be set up at the Grants 50 tab.

- Object correlation with another Microsoft SQL objects is represented on Dependencies 51.

- Most of objects have a possibility to be created from an SQL script (SQL definition). If so, the corresponding script is available at the SQL 52 tab of the editor.

- There is a Result tab in editors of such routines as functions and procedures that can take parameters, perform calculations or other actions, and return a result. You can execute 53 any routine directly from its editor.

Note: Some object editors have additional tabs. The detailed description of them you can find at the appropriate topic of the corresponding section.

To apply the changes, select the Apply Changes item in the Navigation bar or use **Ctrl**

+F9 or **Ctrl+F7** shortcut keys.

It is also possible to modify object properties without opening the object editor: use the Object Properties item of the popup menu of the selected object from the explorer tree.

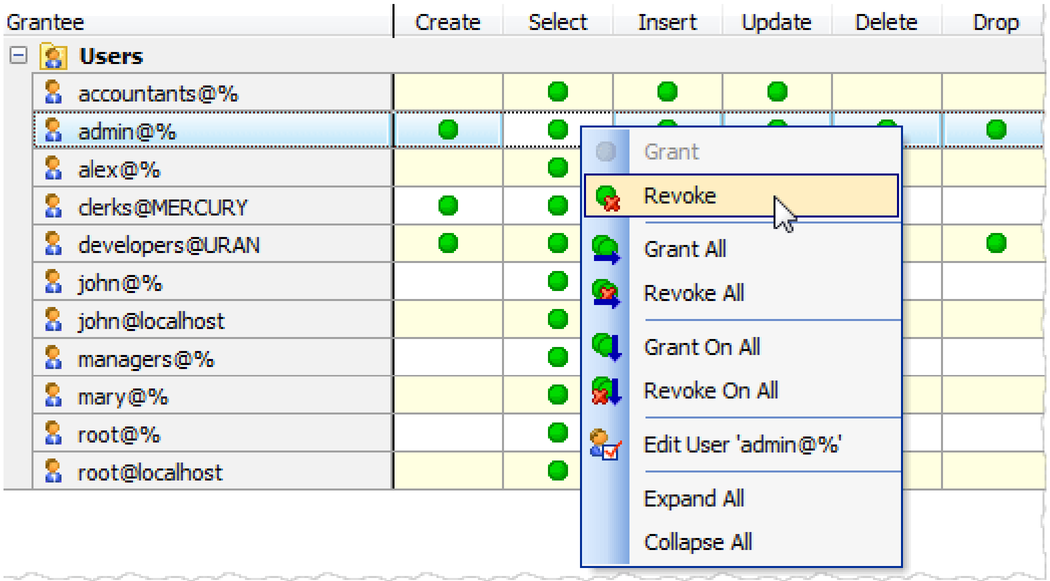
See also:

- Schema Editor 73
- Table Editor 79
- View Editor 114
- UDF Editor 132
- UDT Editor 139
- Aggregate Editor 194
- User Editor 202
- Role Editor 206
- File Editor 208
- File Group Editor 212
- Assembly Editor 216
- Symmetric Key Editor 223
- Asymmetric Key Editor 220
- Certificate Editor 227
- Procedure Editor 123
- Synonym Editor 143
- Rule Editor 147
- Default Editor 152
- CLR Procedure Editor 158
- CLR UDF Editor 163
- CLR UDT Editor 167
- Xml Schema Collection Editor 171
- Queue Editor 175
- Message Type Editor 180
- Contract Editor 183
- Service Editor 186
- Conversation Editor 190

4.2.1.1 Permissions of the Object

The Permissions grid allows you to manage access privileges (grants) of users and database roles.

Grants give specific privileges for an object (*schemas, tables, views, procedures, UDFs, UDTs, XML schema collections, users, roles, assemblies, etc*.) to one or more users.



Grantee	Create	Select	Insert	Update	Delete	Drop
Users						
accountants@%		●	●	●		
admin@%	●	●	●	●	●	●
alex@%		●				
clerks@MERCURY	●	●				
developers@URAN	●	●				●
john@%		●				
john@localhost		●				
managers@%		●				
mary@%		●				
root@%		●				
root@localhost		●				

Using the grid you can grant/revoke privileges as well as sort and filter displayed grantees.

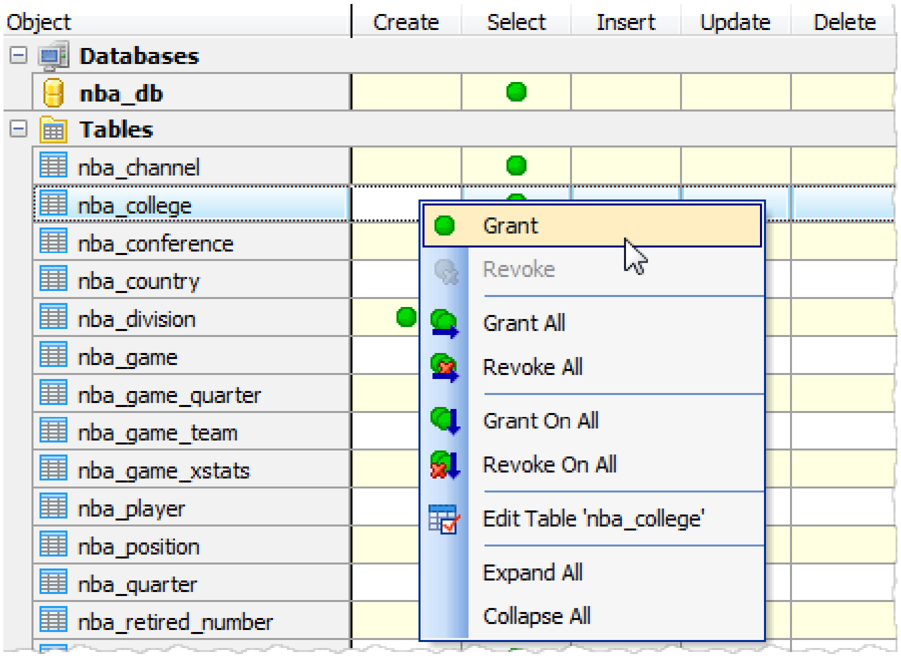
See also: Users 201, and Roles 204

4.2.1.2 Object grants

The Grants grid allows you to manage access privileges (grants) of the current object.

A grant gives specific privileges on an object (*table, view, procedure*) to the current one.

All objects are grouped by kind. Filter the object kinds using the checkboxes at the bottom of the window. Using the grid you may sort and filter data.



To grant the subject privilege on the object double-click an empty field; to revoke the privilege double-click a grant with grant option.

Use grid's popup menu to *grant, grant all, grant with grant option, grant all with grant option, grant on all, grant on all with grant option, revoke, revoke all* and *revoke on all*:

- select the Grant item to grant the subject privilege on the object;
- select the Grant All item to grant all the privileges on the object;
- if the Grant With Grant Option item is selected, the recipient of the privilege may in turn grant it to others (without a grant option specified, the recipient cannot do that; at present, grant options can only be granted to individual subject, not to groups or Public);
- select the Grant All With Grant Option item to grant with grant options the privilege on all the objects of the kind;
- select Grant On All or Grant On All With Grant Option to grant or grant with grant options respectively the subject privilege on all the objects;
- select Deny item to deny the subject privilege on the object;
- select Deny All item to deny all the privileges on the object;
- use Deny on All to deny the subject privilege on all the objects;
- to revoke the privilege, all the privileges on the object or the privileges on all the server objects select the Revoke, Revoke All or Revoke On All items respectively.

Using the popup menu you can also collapse or expand all the object kinds.

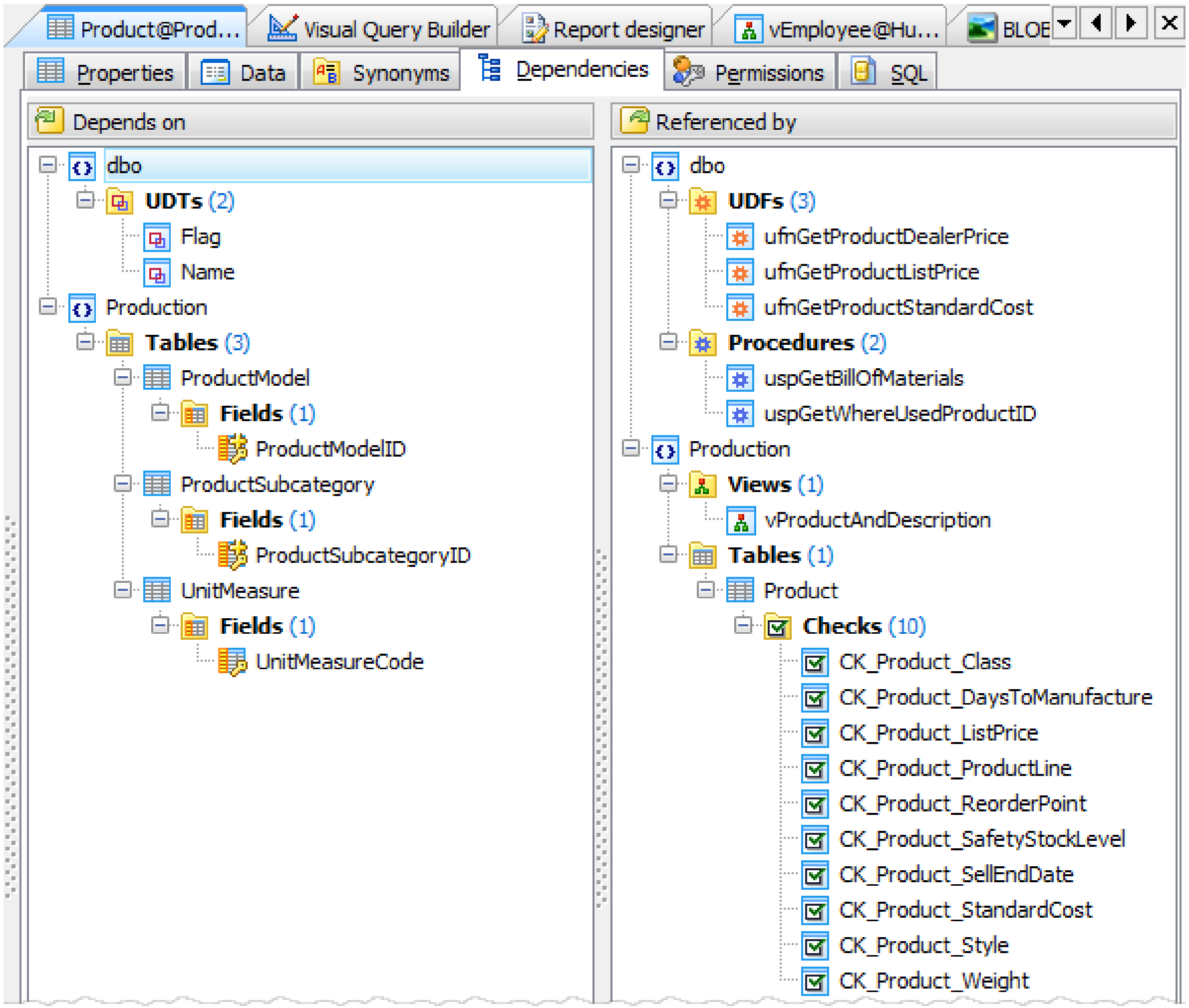
4.2.1.3 Object Dependencies

When you create complex database structures involving many tables with foreign key constraints, views, triggers, UDFs, etc. you will implicitly create a net of dependencies between the objects. For instance, a table with a foreign key constraint depends on the table it references. We tried to create a software to make your work easier, i.e. to give you the tools for efficient database objects management. The Dependencies tab allows you to control the correlation of objects efficiently.

The Depends on window represents all the objects the current object depends on.

The Referenced by window contains the tree of database objects constituting the dependency relationship on the current object.

Use items of the popup menu to edit the selected object in Object Editor or to drop the object.



4.2.1.4 SQL Definition

The SQL tab displays the SQL definition for the object with all its properties. Bear in mind that this text is read-only. If you want to change the object definition, use the appropriate editor tabs instead, or copy the text to the Windows Clipboard to paste it in SQL Editor or SQL Script Editor.

The SQL definition window allows you to browse the text effectively. The popup menu and the extensive system of hot keys give you the opportunity to search expressions within the text, to select the whole text for copying it to the Windows Clipboard, to save the definition to the **.sql* or **.txt* files, to print the document, etc.

You can customize the displayed definition using the Editors & Viewers 447 options.

The Properties item of the popup menu displays the Options dialog in which you can establish optional settings concerning the current database.

The Code Folding item group makes it possible to view either the whole text or its logical parts (regions). Each region can be collapsed and extended.

In extended mode the whole text is displayed (set by default)

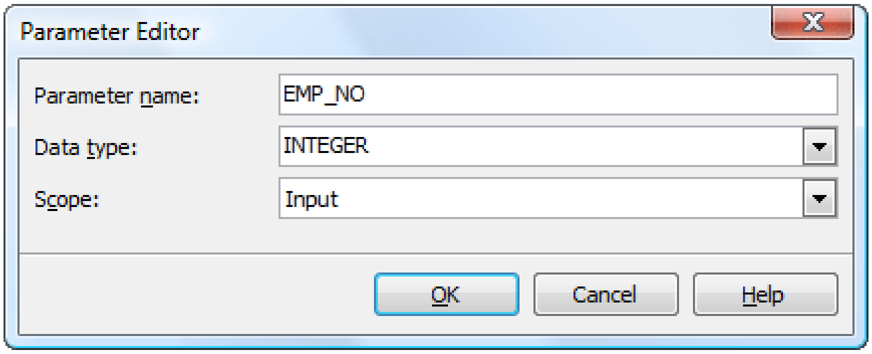
In collapsed mode the text is hidden behind one text line denoting the first line of the collapsed region.

Navigation Bar on the SQL tab allows you to copy the object's SQL definition (DDL) to the SQL Script Editor for future modifications.

4.2.1.5 Parameter Editor

The editor allows you to change the Parameter name, Data type and the Default value that specifies a parameter value to be used when no value is explicitly supplied.

Microsoft SQL supports the scope function parameters (*Input* is the default value).



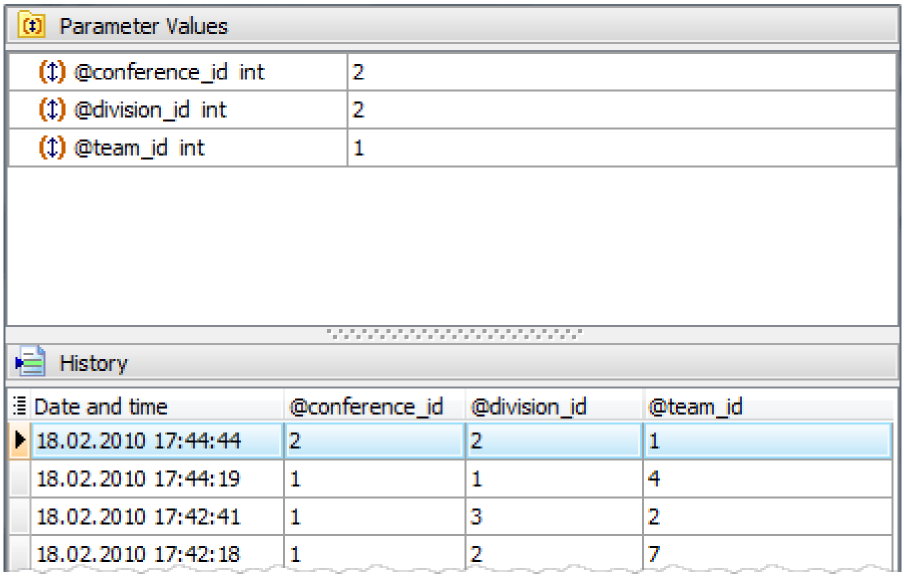
Parameter name: EMP_NO
Data type: INTEGER
Scope: Input

Use the Size edit box to define the length of the parameter value for float, char and other data types and the Precision edit box to define the precision of the parameter value, e.g. for decimal data type.

4.2.1.6 Executing functions and procedures

Procedure/UDF/CLR UDF Editor provides an opportunity to execute current routine by opening the Results tab, by clicking the Execute item of the Navigation Bar, or by pressing the **F9** key.

If the procedure/UDF/CLR UDF has parameters, MS SQL Maestro will ask you to specify the values for these parameters in the Input parameters dialog which appears before the procedure execution. Input parameters dialog allows you to specify the values for all input parameters. After changes are made, click the OK button to execute the UDF, or the Cancel button to abort the execution.



Parameter Values

@conference_id int	2
@division_id int	2
@team_id int	1

History

Date and time	@conference_id	@division_id	@team_id
18.02.2010 17:44:44	2	2	1
18.02.2010 17:44:19	1	1	4
18.02.2010 17:42:41	1	3	2
18.02.2010 17:42:18	1	2	7

MS SQL Maestro supports Parameter History. Values that have been set previously as the routine parameters are represented in the History tab of the Input Parameter dialog with a date and time of their last using. Double click a necessary set of values to set them as the routine parameters. You can manage the Parameter History with Delete history item and Clear history links.

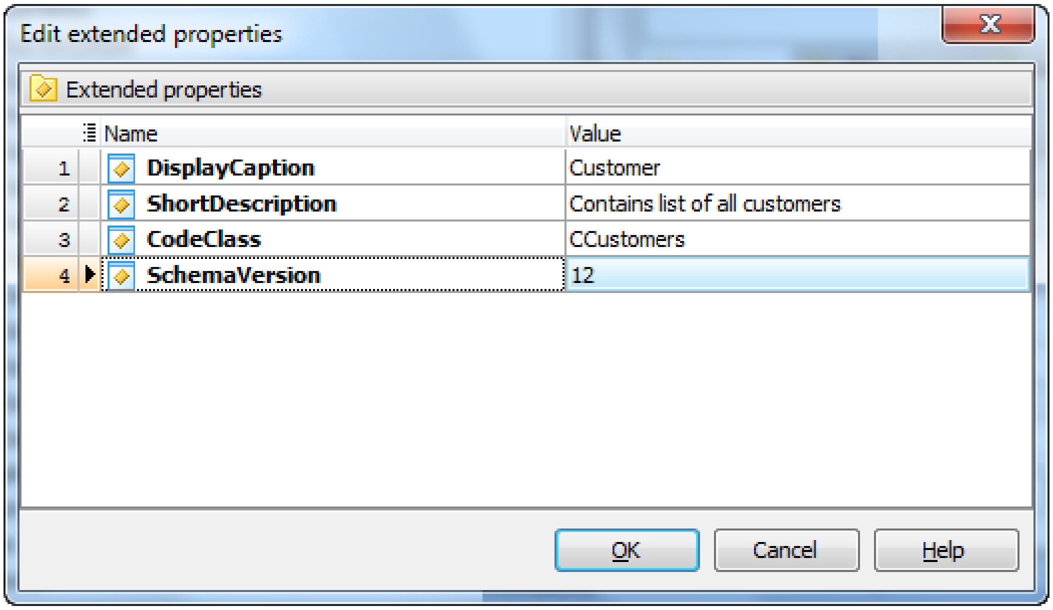
The result of the successfully executed routine can be found within the Results tab of Procedure/UDF/CLR UDF Editor.

Note: If any unsaved changes are applied to the routine being currently edited, the execution of the routine is impossible until changes are saved by the Compile procedure item of the Navigation Bar.

4.2.2 Modify Object Properties

You can rename all objects those can be renamed with the corresponding option of the popup menu of the object at the Explorer tree. To edit other properties of the selected object without opening its editor, use the Object Properties dialog. To open this dialog, select the according item of the same popup menu. To clear up the object properties meanings, see the appropriate topic of the respective Object Editor section.

MS SQL Maestro also supports extended properties. In using extended properties, you can add text, such as descriptive or instructional content, add input masks, and add formatting rules as properties of objects in a database or of the database itself. For example, you can add an extended property to a schema, a schema's view, or to a column in the view. Because extended properties are stored in the database, all applications reading the properties can evaluate the object in the same way. This helps enforce consistency in the way data is treated by all the programs in the system.
MS SQL Maestro allows you to add/edit/drop such properties for tables, views, procedures, columns, indexes, etc. Each extended property has a user-defined name and value. The value of an extended property can contain up to 7,500 bytes of data. Multiple extended properties can be added to a single object.



	Name	Value
1	**DisplayCaption**	Customer
2	**ShortDescription**	Contains list of all customers
3	**CodeClass**	CCustomers
4	**SchemaVersion**	12

4.2.3 Describe Objects

Essentially a comment is the most often altered object property. To simplify it's editing, the MS SQL Maestro provides an ability to Describe the object within the Database Explorer immediately without opening of the object's editor.

Step-by-step:

- Select the necessary object in the explorer tree;
- Choose the Describe Object... item in the popup menu;
- Edit object comments within the Describe Object window;
- To commit the changes, push OK button.

4.3 Duplicate Objects

MS SQL Maestro offers several ways of objects duplicating.

1. **Duplicate Object Wizard.** The wizard is the most flexible tool of the coping. Along with a possibility to adjust the new object definition it allows you to copy data (for tables). But it consists of several steps 56 and takes more time than other manners.

2. **Duplicate Object** window allows you to attune new object's SQL definition. It is preferred for creation a copy of selected object. Here 58 you can find some additional info.

3. By **Drag-n-Drop** 59 **operation.**

4.3.1 Duplicate Object Wizard

The Duplicate Object Wizard allows you to create a new database object with the same properties as the existing one. It is the most flexible tool of copying objects provided by MS SQL Maestro. It also allows you to copy data of the selected table to the new one.

To run the wizard select the Object | Duplicate Database Object... main menu item.

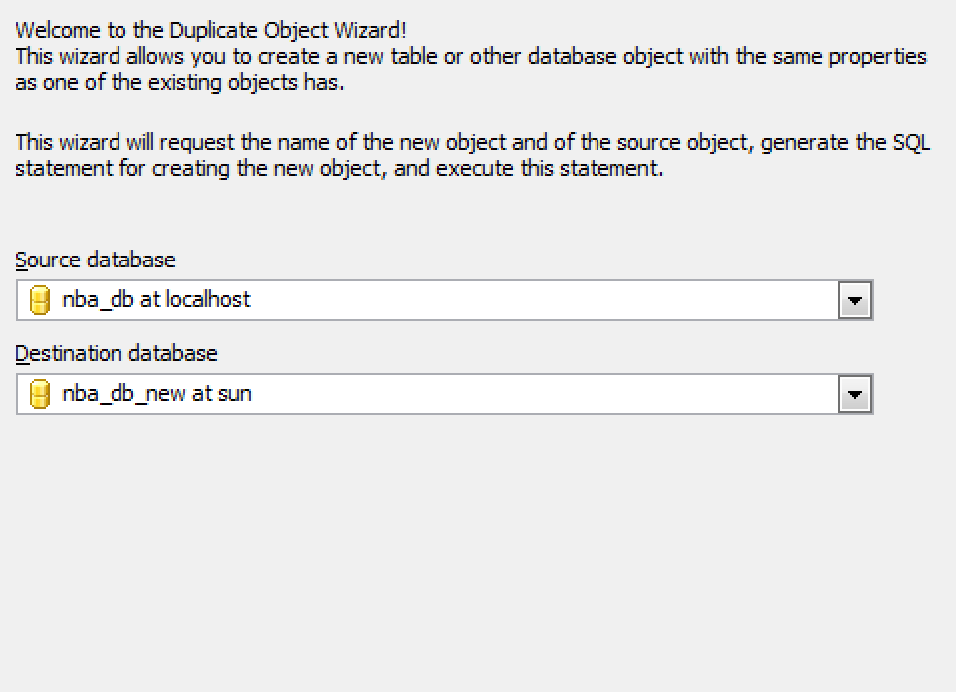
- Selecting source and destination databases 56
- Selecting object to duplicate 57
- Modifying definition of a new object 58

See also: Create Database Object 44

4.3.1.1 Selecting source and destination databases

Select the database containing a source object from the list of connected databases, and then specify the database for the duplicated object.

You should connect to the destination database beforehand (see Database Management 22).



Welcome to the Duplicate Object Wizard!
This wizard allows you to create a new table or other database object with the same properties as one of the existing objects has.

This wizard will request the name of the new object and of the source object, generate the SQL statement for creating the new object, and execute this statement.

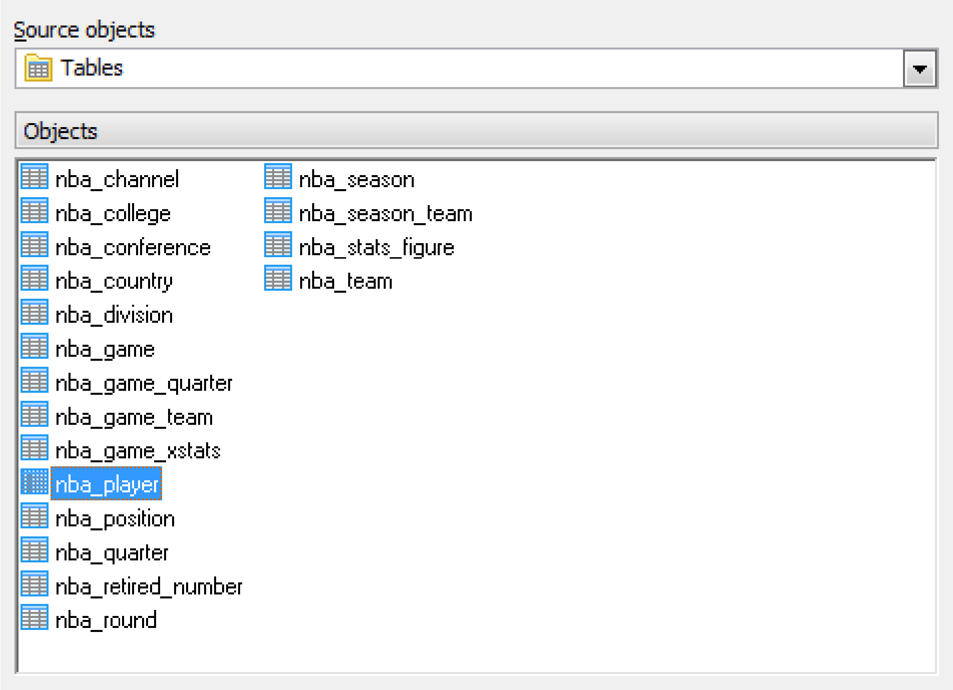
Source database: nba_db at localhost

Destination database: nba_db_new at sun

4.3.1.2 Selecting object to duplicate

Specify a database object to create the new one with the same properties.

1. Select the type of the object to duplicate from the Source objects drop-down list.
2. Pick up the necessary object from the list.



Source objects: Tables

Objects

nba_channel
nba_college
nba_conference
nba_country
nba_division
nba_game
nba_game_quarter
nba_game_team
nba_game_xstats
nba_player
nba_position
nba_quarter
nba_retired_number
nba_round
nba_season
nba_season_team
nba_stats_figure
nba_team

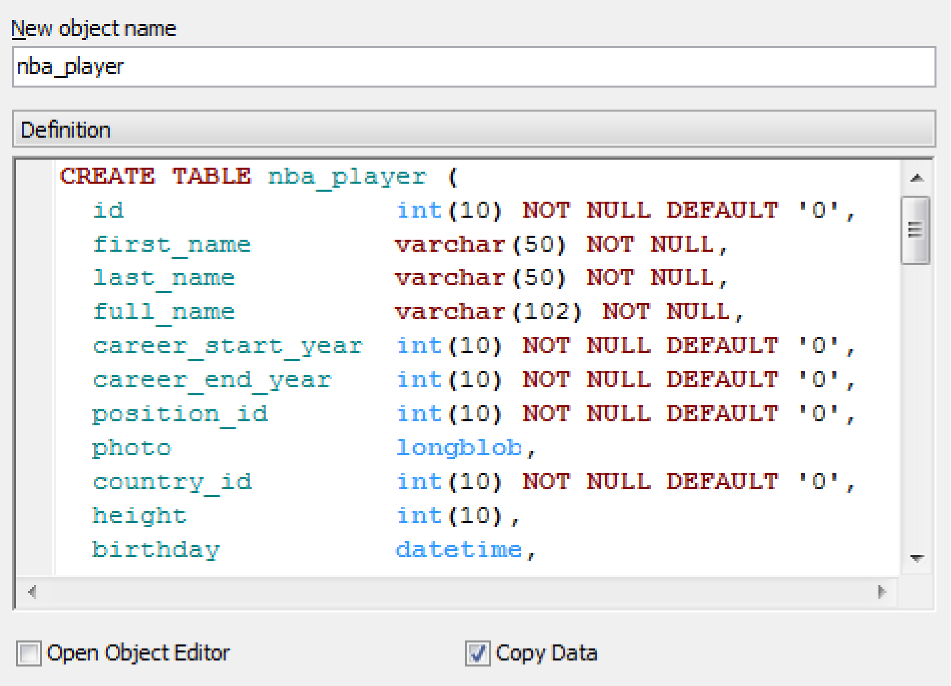
4.3.1.3 Modifying new object definition

The last wizard step allows you to edit the new object definition directly.

Use this step to edit the name of object been creating (New Object Name). By default MS SQL Maestro generates the new object definition with the same name if the duplicating is to the source database, or like "%SOURCE_OBJECT_NAME%01" otherwise.

> **Note:** the name of the object must be unique among all the object names in its container. Moreover, all the objects that are source of data need unique names among themselves. You can use any identifier that is allowed by Microsoft SQL server.

You can edit the result SQL statement manually, add or remove fields, change field types, using the New object definition text area. Click the Ready button to complete the operation.
Check the according boxes to Copy Data (only for tables) and to Open Object Editor after the duplicating.



New object name: nba_player

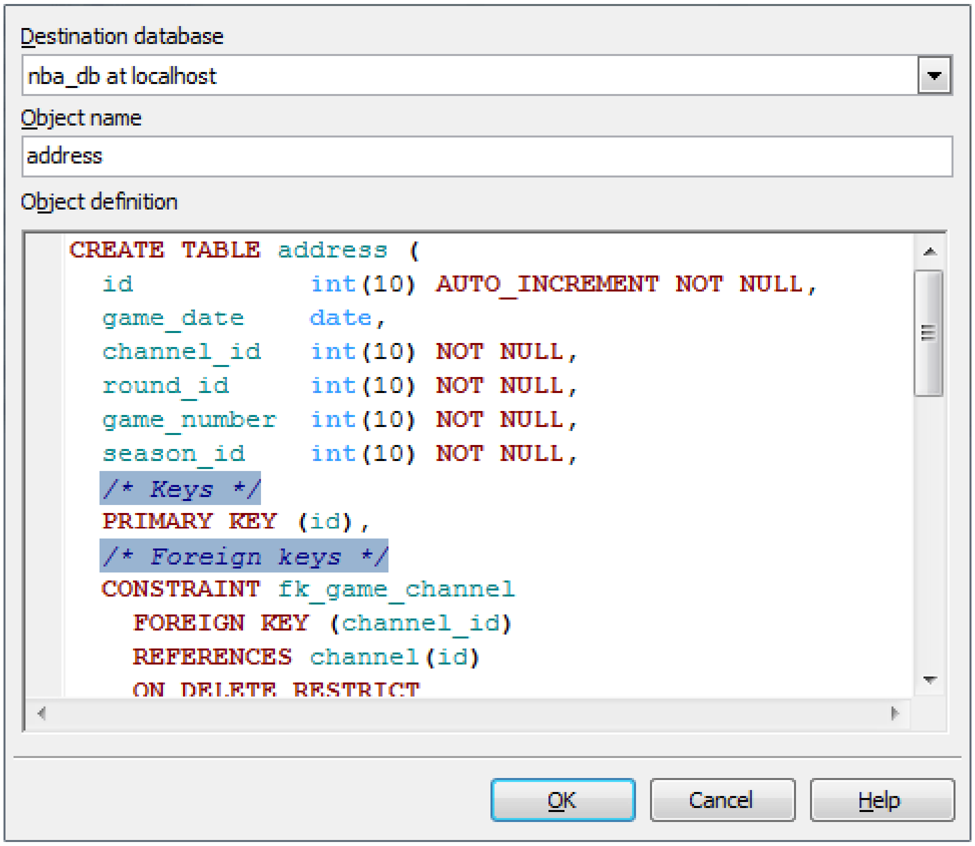
```
CREATE TABLE nba_player (
  id                int(10) NOT NULL DEFAULT '0',
  first_name        varchar(50) NOT NULL,
  last_name         varchar(50) NOT NULL,
  full_name         varchar(102) NOT NULL,
  career_start_year int(10) NOT NULL DEFAULT '0',
  career_end_year   int(10) NOT NULL DEFAULT '0',
  position_id       int(10) NOT NULL DEFAULT '0',
  photo             longblob,
  country_id        int(10) NOT NULL DEFAULT '0',
  height            int(10),
  birthday          datetime,
```

☐ Open Object Editor ☑ Copy Data

4.3.2 Duplicate Selected Object

Within the Duplicate Object window you can duplicate a selected object fast and with some modifications.

It is available from the corresponding link of the object's popup menu at the Database Explorer.



Destination database: nba_db at localhost

Object name: address

```
CREATE TABLE address (
  id            int(10) AUTO_INCREMENT NOT NULL,
  game_date     date,
  channel_id    int(10) NOT NULL,
  round_id      int(10) NOT NULL,
  game_number   int(10) NOT NULL,
  season_id     int(10) NOT NULL,
  /* Keys */
  PRIMARY KEY (id),
  /* Foreign keys */
  CONSTRAINT fk_game_channel
    FOREIGN KEY (channel_id)
    REFERENCES channel(id)
    ON DELETE RESTRICT
```

Select the database for a new object from the list of connected databases first.

Enter the name for the new object.

> **Note:** the name of the object must be unique among all the object names in its container. Moreover, all the objects that are source of data need unique names among themselves. You can use any identifier that is allowed by Microsoft SQL server.

You can also edit the SQL definition of the object if necessary (add or remove fields, change field types, etc.).

4.3.3 Copy, Paste and Drag-n-Drop features

MS SQL Maestro provides you with an ability of copying database objects within the database or even from one database to another (in this case you should connect to both the source and the destination databases first).

To copy an object, just drag the object in a source window (such as Database Explorer, Object Manager, Object Browser) and drop it to the target container in another window. You also can use the Edit | Copy and the Edit | Paste main menu items or the **Ctrl+C**/**Ctrl+V** hot keys combinations respectively. Copying several objects at a time is also available.

It is also possible to drag and drop objects between Database Explorer, Object Manager, Object Browser and SQL Editor or SQL Script Editor. This works as follows:

SQL Editor: after dropping the object you will get a query to retrieve object data (e.g. SELECT * FROM table_name) or the full name of the object if it doesn't contain data (domains, indexes, etc.).

SQL Script Editor: after dropping the object you will get its SQL definition if applicable.

See also: Database Explorer, Object Manager, and Object Browser

4.4 Browse Objects

MS SQL Maestro allows to browse objects stored in a Remote Server database in several ways:

- Database Explorer 61: objects are represented as a hierarchy (grouped by kind and listed under the according Microsoft SQL servers/database node, provided with subobjects if exist)
- Object Browser 65: an extension of explorer with ability to sort, group, filter and multiple select objects.
- Object Manager 66: an extension of the explorer with ability to select several objects at a time (to copy, drop, etc.)

All tool allows you to drag-and-drop between them and to perform all necessary operations upon database objects.

4.4.1 Database Explorer

Database Explorer is the basic feature of MS SQL Maestro which allows you to perform practically all necessary operations upon databases and their objects. The Database Explorer area occupies the left side of the MS SQL Maestro main window. All the objects at the Explorer tree are grouped by kind and listed under the according Microsoft SQL servers/database node.

To start working with a database you need to create its profile first. The conception of database profiles gives you an opportunity to connect to databases in one touch and work with the selected databases only.

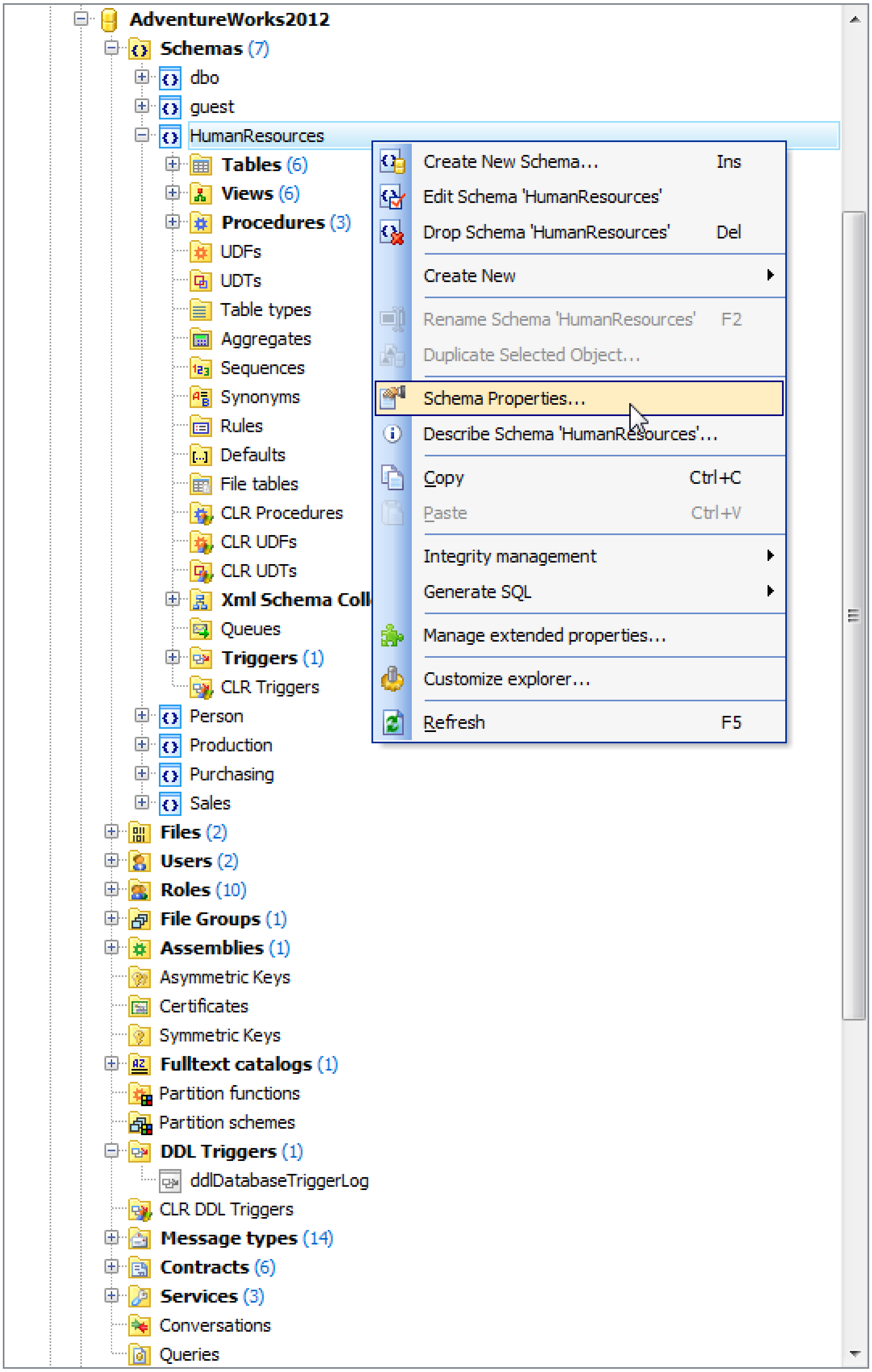
See also: Object Manager 66, Object Browser 65

Note: In case your databases have a large quantity of objects you can speed up the object search by typing first letters of the object name in the explorer area.

Note: Explorer options 430 allow you to hide/display table subobjects, represent system objects in different color, etc.

The sections below describe each of these actions in detail.

- What operation can I accomplish upon database profiles within the Explorer Tree? 63
- How can I connect to a database? 61
- How can I disconnect from a database? 63
- What operations can I accomplish upon database objects within the Explorer Tree? 63
- Can I copy a database object from one database to another? 63
- Can I filter Explorer content? 64
- How can I create new/drop a database? 64


AdventureWorks2012
Schemas (7)
dbo
guest
HumanResources
Tables (6)
Views (6)
Procedures (3)
UDFs
UDTs
Table types
Aggregates
Sequences
Synonyms
Rules
Defaults
File tables
CLR Procedures
CLR UDFs
CLR UDTs
Xml Schema Coll
Queues
Triggers (1)
CLR Triggers
Person
Production
Purchasing
Sales
Files (2)
Users (2)
Roles (10)
File Groups (1)
Assemblies (1)
Asymmetric Keys
Certificates
Symmetric Keys
Fulltext catalogs (1)
Partition functions
Partition schemes
DDL Triggers (1)
ddlDatabaseTriggerLog
CLR DDL Triggers
Message types (14)
Contracts (6)
Services (3)
Conversations
Queries
Create New Schema... Ins
Edit Schema 'HumanResources'
Drop Schema 'HumanResources' Del
Create New
Rename Schema 'HumanResources' F2
Duplicate Selected Object...
Schema Properties...
Describe Schema 'HumanResources'...
Copy Ctrl+C
Paste Ctrl+V
Integrity management
Generate SQL
Manage extended properties...
Customize explorer...
Refresh F5

Operations upon database profiles in the Explorer Tree

Using popup menu of the Explorer area you can realize the following operations:

- create new database profiles 24 (the Create Database Profiles... item);
- rename currently selected database profile (the Rename Database Profile... item);
- edit currently selected database profile 27 (the Edit Database Profile... item);
- reorder existing database profiles (the Reorder Databases...item of Databases node's popup menu or using drag-n-drop);
- reorder servers (the Reorder Servers...item of a server's popup menu);
- remove currently selected database profile from the explorer tree (the Remove Database Profile item);
- remove all profiles of selected server (the Remove all Profiles item of Databases node's popup menu).

In addition to these operations, Database Explorer gives you an ability to reorder existing profiles by performing drag-and-drop operations within the explorer tree.

How can I connect to a database?

You can establish connection to a database in Database Explorer by selecting the database profile and double-clicking it or pressing the Enter key (alternatively, you may use the Shift+Ctrl+C hot key combination). The same operation is also available through the Connect to Database item from the explorer popup menu, or through the Database | Connect to Database main menu item.

How can I disconnect from a database?

You can abort connection from a database in Database Explorer by selecting the database profile and pressing the Shift+Ctrl+D hot key combination. The same operation is also available through the Disconnect from Database item from the explorer popup menu, or through the Database | Disconnect from Database main menu item.

Operations upon database objects

Database Explorer allows you to perform the following operations with database objects using its popup menu (note that the popup menu contains object-specific items only when some database object is currently selected in the explorer tree):

- create a new database object (the Create New Object... item);
- edit currently selected database object (using the Edit Object... item, pressing the Enter key or double-clicking the database object);
- drop the selected object from the database (the Drop Object... item);
- rename the selected database object (the Rename Object... item);
- edit the database object properties (the Object properties ... item);
- duplicate the selected object (the Duplicate Object... item).
- run the Object Browser tool (the Browse ... item).

Can I copy a database object from one database to another?

Database Explorer provides you with an ability of copying database objects from one database to another. To perform this operation, you should connect to both the source and the destination databases first. After the connection is established, simply drag and drop an object to copy from the source database to the corresponding node (Tables, Queries, etc.) of the destination database.

Note: You also can use the Edit | Copy and the Edit | Paste main menu items to copy/paste a database object using Windows clipboard (alternatively, you may use the Ctrl+C/Ctrl+V hot keys combinations respectively).

How can I create new/drop a database?

To create a new Microsoft SQL database (not existing on your Microsoft SQL server) with Database Explorer, select the Create New Database... item from the popup menu and set all the necessary options within the Create Database Wizard 34.

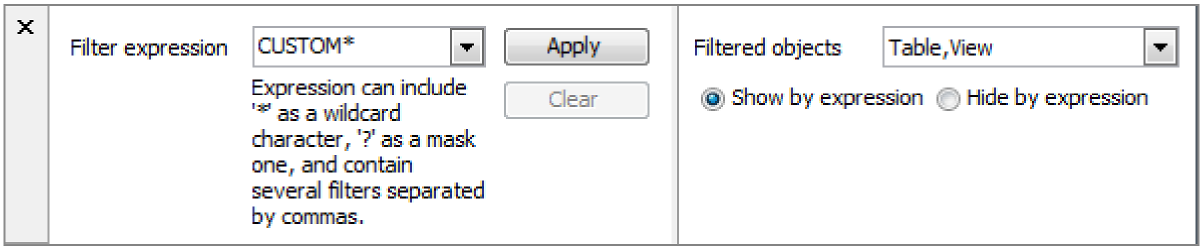
To drop an existing database using Database Explorer, connect to the database you wish to drop, select the Drop Database item from the popup menu of the database and confirm dropping in the dialog window.

Note: Alternatively, you can use the Database | Create New (Drop) Database main menu item to perform these operations.

4.4.1.1 Filtering explorer content

MS SQL Maestro allows you to reduce the number of represented objects in the explorer tree. To hide seldom usable objects, filter your explorer content.

Filter Panel is available through the View | Show Filter Panel main menu item.



Filter expression: CUSTOM*

Expression can include '*' as a wildcard character, '?' as a mask one, and contain several filters separated by commas.

Filtered objects: Table,View

● Show by expression ○ Hide by expression

- Specify the Filter expression. The expression can contain any part of object name combined with an asterisk ('*') as a wildcard character and a question-mark ('?') as a mask character.

- Define the Filtered objects, object types for filtering in the explorer tree.

- Check the according radio button (Show by expression, Hide by expression) to define whether database objects will be shown or hidden in accordance with the filter expression.

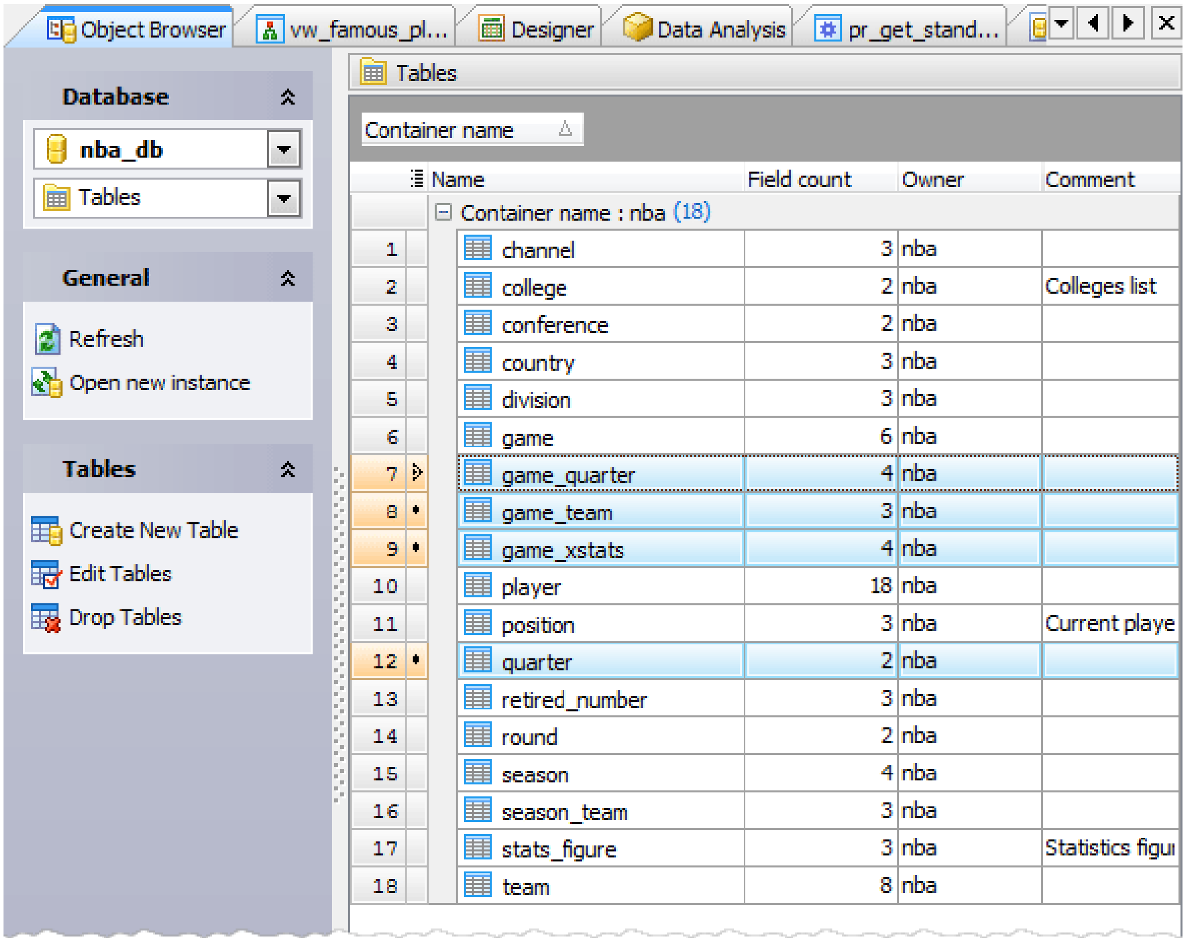
- Click Apply button.

Note: A filter expression, if applied to the content of Database Explorer, is applied to the content of Object Manager 66 and Object Browser 65 as well.

4.4.2 Object Browser

Object Browser is a tool for operating on database objects designed as an extension of Database Explorer 61 with ability to *sort*, *group* and *filter* the database objects. It also provides such operation as multiple selecting of objects (for *copying*, *dropping*, etc.) and the ability of using drag-and-drop operations between Object Browser and Database Explorer. To open Object Browser select the Object | Object Browser main menu item.

Note: At least one connection to a database should be established to make Object Browser available.



Container name : nba (18)

	Name	Field count	Owner	Comment
1	channel	3	nba	
2	college	2	nba	Colleges list
3	conference	2	nba	
4	country	3	nba	
5	division	3	nba	
6	game	6	nba	
7	game_quarter	4	nba	
8	game_team	3	nba	
9	game_xstats	4	nba	
10	player	18	nba	
11	position	3	nba	Current playe
12	quarter	2	nba	
13	retired_number	3	nba	
14	round	2	nba	
15	season	4	nba	
16	season_team	3	nba	
17	stats_figure	3	nba	Statistics figu
18	team	8	nba	

Sorting database objects

Object Browser represents database objects in a grid. The object kind to display is defined on the top of the Navigation bar. The columns correspond to the objects properties and rows correspond to the objects. Click the column caption to sort objects by the values of this column in the ascending or descending mode. The navigation buttons allow you to open current object editor, create new or drop the existing one.

As Object Manager the browser allows you to operate on several objects at a time. You have an opportunity to select a batch of objects and after the object group is selected, you can operate on it (e.g. *drop several objects at once*) as if it were a single object.

The unique feature of the MS SQL Maestro is an opportunity of drag-and-drop operations between Object Browser and SQL Editor, SQL Script Editor. After the action objects are represented in SQL Editor as SQL queries (if they contain data) or as their full name in the database otherwise. SQL Script Editor displays the objects as SQL definition.

Grouping database objects
You can group grid objects by any of the columns by dragging the column header to the destination area. Now all the records are displayed as subnodes to the grouping row value as shown in the picture. To reverse grouping, just drag the column name from the upper area back.

Filtering database objects
You can filter objects in the grid using one of the following methods:

- use the drop-down button in the column caption area to filter objects by the value of the selected column
- click the drop-down button in the column caption area, then select the Custom item and build a simple filter within the dialog in the following way: select a logical operator for checking the column values (like is less than, is greater than, etc) and set the value to be checked by this operator in the neighboring box; then set the second condition if necessary in the following way and set the relation between these two conditions, whether both of them should be matched or just one of them; use the '_' character to represent any single symbol in the condition and the '%' character to represent any series of symbols in the condition

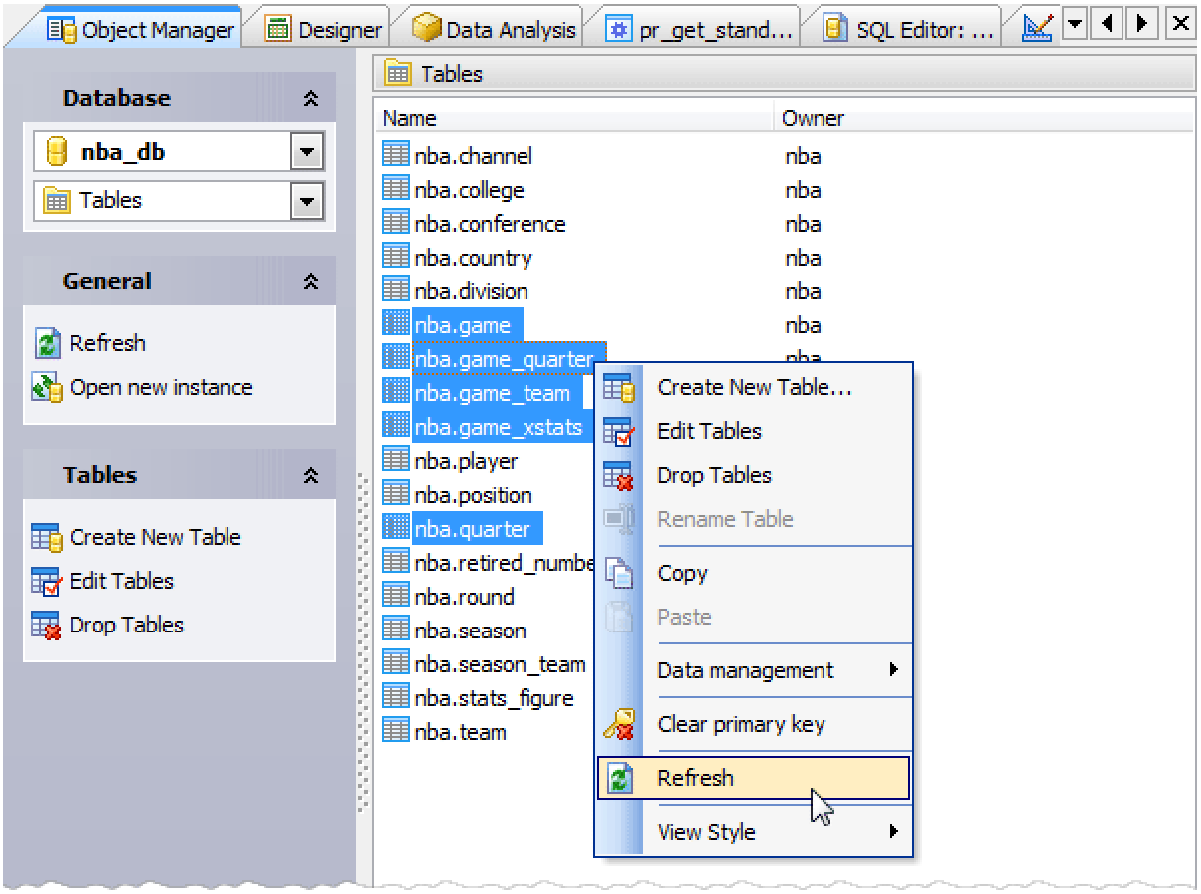
After you set a filter, the filtering panel becomes visible at the bottom of the grid where you can see the active filtering condition and easily enable or disable it by clicking the check box on the left. Using this panel you can also customize your filter in a more complicated way by clicking the Customize button and building your filter within the Filter Builder dialog.

See also: Object Manager 66, Data View 328

4.4.3 Object Manager

Object Manager is a tool for operating on database objects designed as an extension of the Database Explorer 61 with advanced features, such as multiple selecting of objects (for *copying*, *dropping*, etc.) and the ability of using drag-and-drop operations between Object Manager and Database Explorer as well as between two instances of the Object Manager. To open Object Manager select the Object | Object Manager main menu item.

Note: At least one connection to a database should be established to make Object Manager available.



Using popup menu

The popup menu of Object Manager may have different content depending on the current selection. The common menu items allow you to switch the object list view between four standard modes *(large icons, small icons, list* and *report*), refresh the current view, and select all the objects in the view. If none of objects are currently selected, other menu items are unavailable to use, except of the one for creating a new object. If one or more objects are selected, clipboard operations (such as copy and paste) become available as well as the items for editing and dropping selected object(s). If the current object type of the Object Manager is *"Tables"*, the *Empty Table(s)* menu item is also available.

Multiple selecting of database objects

Object Manager allows you to operate on several objects at a time. You have an opportunity to select a batch of objects and after the object group is selected, you can operate on it (e.g. *drop several objects at once*) as if it were a single object.

See also: Object Browser 65

4.4.4 Filter Builder Dialog

Filter Builder Dialog allows to limit represented objects according to specified conditions. It may be useful for filtering records in data grids of Table Editors, SQL Editor or Visual Query Builder as well as to filter database objects in Object Browser, and on setting a condition on anew view creating. All these cases are similar, see how it works on the following example.

5 Database Objects

The following list contains database objects supported by MS SQL Maestro. To work with database objects you should connect to the database 13 first.

- Schemas 70
- Tables 76
- Views 108
- UDFs 128
- UDTs 136
- Aggregates 192
- Users 201
- Roles 204
- Files 208
- File Groups 211
- Assemblies 214
- Symmetric Keys 222
- Asymmetric Keys 218
- Certificates 225
- Procedures 118
- Synonyms 141
- Rules 145
- Defaults 150
- CLR Procedures 156
- CLR UDFs 161
- CLR UDTs 166
- Xml Schema Collections 169
- Queues 173
- Message Types 179
- Contracts 182
- Services 185
- Conversations 188
- DDL Triggers 230

5.1 Schemas

Schemas are a purely logical structure and the privilege system determines whether one can access them. A database is a collection of schemas and the schemas contain tables, views, UDFs, etc. The full hierarchy is: server, database, schema, table (or some other kind of object, such as a UDF). Schemas are being implemented in Microsoft Server 2005.

How can I add a new schema?

New schemas are created within Create Schema Wizard 71. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the Schema icon in the Create Database Object dialog

or

- select the Schemas list or any object from that list in the explorer tree;
- select the Create New Schema... item from the popup menu

or

- open the database in Database Editor and the Schemas tab there;
- press the **Insert** key or select the Create New Schema item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

To create a new schema with the same properties as one of the existing schemas has:

- select the Object | Duplicate Database Object... main menu item;
- follow the instructions of Duplicate Object Wizard.

How can I edit an existing schema?

Schemas are edited within Schema Editor 73. In order to run the editor you should either

- select the schema for editing in the explorer tree (type the first letters of the schema name for quick search);
- select the Edit Schema... item from the popup menu

or

- open the database in Database Editor and the Schemas tab there;
- select the schema to edit;
- press the **Enter** key or select the Edit Schema item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

How can I drop an existing schema?

To drop a schema:

- select the schema to drop in the explorer tree;
- select the Drop Schema item from popup menu

or

- open the database in Database Editor and the Schemas tab there;
- select the schema to drop;
- press the **Delete** key or select the Drop Schema item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

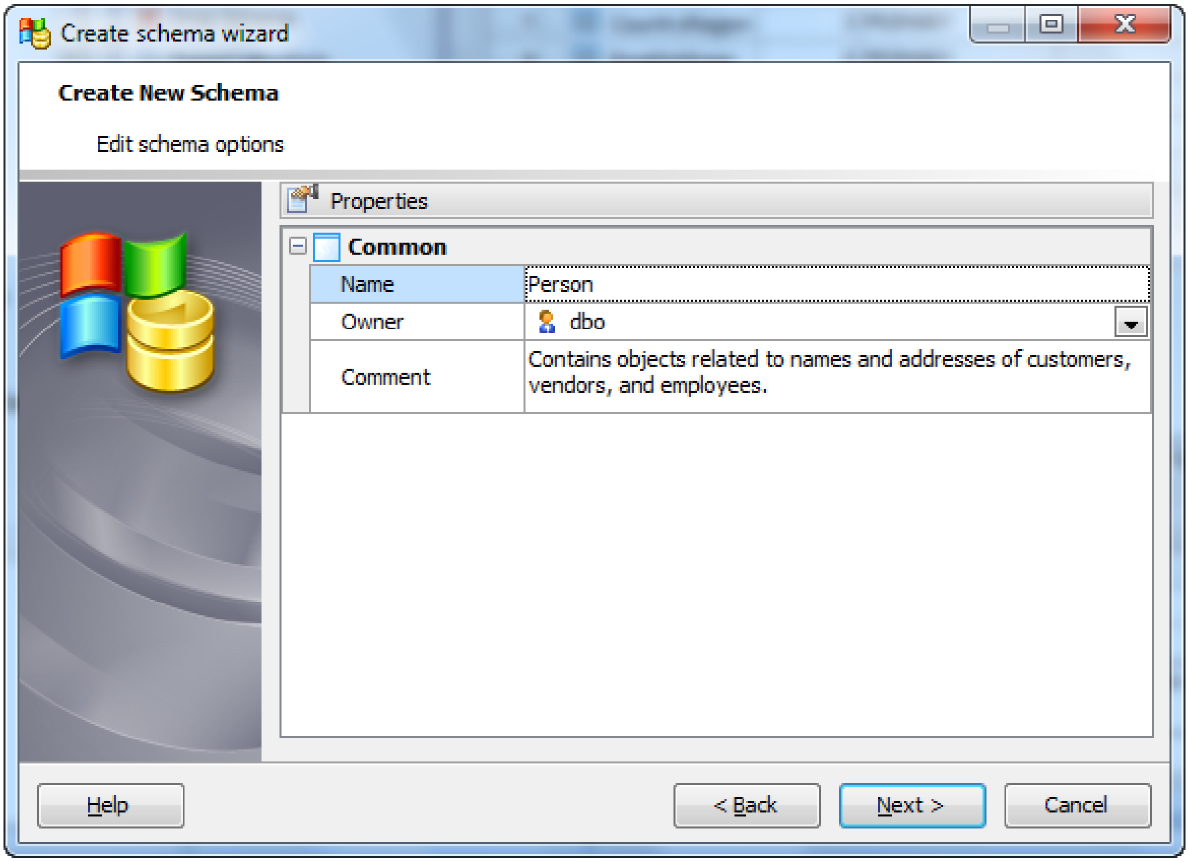
and confirm dropping in the dialog window.

5.1.1 Create Schema Wizard

Create Schema Wizard guides you through the process of creating a new database schema.
The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

Schema options
Owner
Use the field to specify the owner of the new schema. The default owner is the user who have created the schema. By default, only the owner of an object can perform various operations with the object. In order to allow other users to operate it, privileges must be granted. (However, users that have the superuser attribute can always access any object.)

Comment
The box allows you to set optional text describing the new schema.

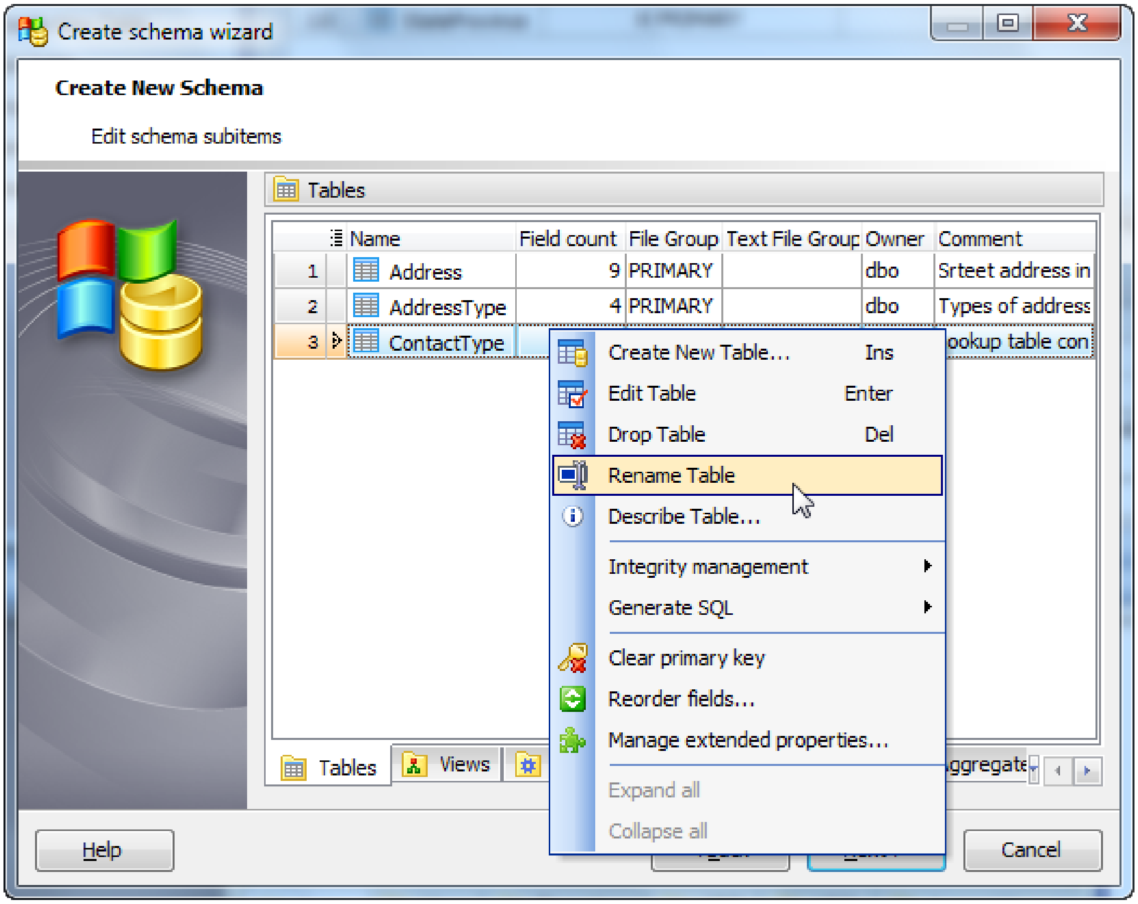


Name: Person
Owner: dbo
Comment: Contains objects related to names and addresses of customers, vendors, and employees.

Adding schema content
This wizard step allows you to create the new schema along with schema content. To add an object to the new schema:

- Open the corresponding tab (Tables - to manage schema tables, Views - to manage schema views, and so on);
- Use the Create Object Wizard link of the tab's pop-up menu or press **Insert**;
- Complete the corresponding create object wizard. To find out the wizards description, read the corresponding topics:

Create Table Wizard 77
Create Procedure Wizard 119
Create UDF Wizard 129
Create UDT Wizard 137
Create Aggregate Wizard 193
Create Synonym Wizard 142
Create Rule Wizard 146
Create Default Wizard 151
Create View Wizard 109

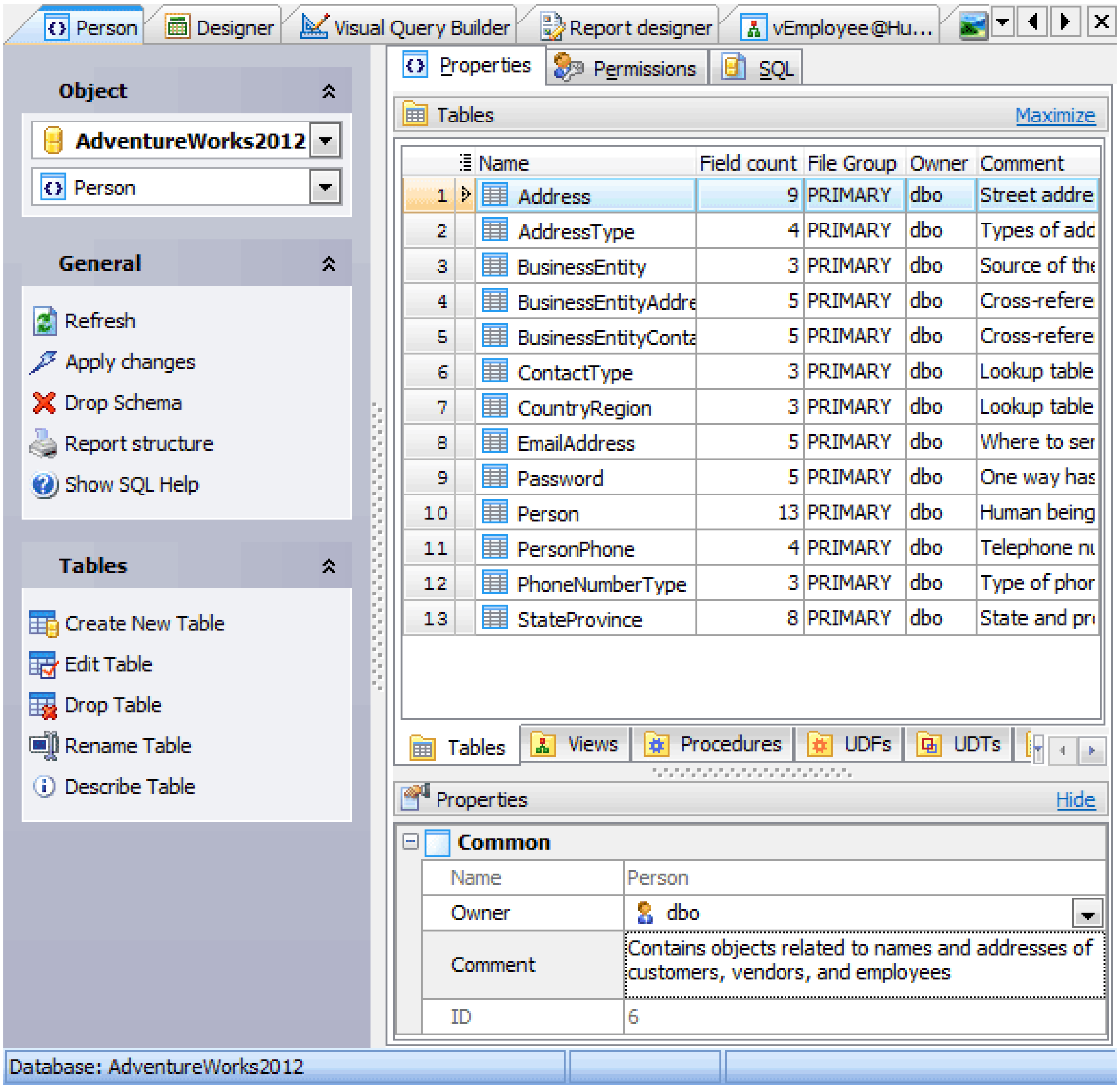


	Name	Field count	File Group	Text File Group	Owner	Comment
1	Address	9	PRIMARY		dbo	Srteet address in
2	AddressType	4	PRIMARY		dbo	Types of address
3	ContactType					ookup table con

5.1.2 Schema Editor

Schema Editor allows you to browse schema content, manage users permissions on the schema objects, and see the SQL definition of this schema. To open the editor, use the corresponding items of popup menus of the Explorer Tree 61, Object Manager 66 or Object Browser 65.

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.



	Name	Field count	File Group	Owner	Comment
1	Address	9	PRIMARY	dbo	Street addre
2	AddressType	4	PRIMARY	dbo	Types of adc
3	BusinessEntity	3	PRIMARY	dbo	Source of the
4	BusinessEntityAddre	5	PRIMARY	dbo	Cross-refere
5	BusinessEntityConta	5	PRIMARY	dbo	Cross-refere
6	ContactType	3	PRIMARY	dbo	Lookup table
7	CountryRegion	3	PRIMARY	dbo	Lookup table
8	EmailAddress	5	PRIMARY	dbo	Where to ser
9	Password	5	PRIMARY	dbo	One way has
10	Person	13	PRIMARY	dbo	Human being
11	PersonPhone	4	PRIMARY	dbo	Telephone nı
12	PhoneNumberType	3	PRIMARY	dbo	Type of phor
13	StateProvince	8	PRIMARY	dbo	State and pr

Common

Name	Person
Owner	dbo
Comment	Contains objects related to names and addresses of customers, vendors, and employees
ID	6

Database: AdventureWorks2012

The Properties tab allows you to view schema options and to browse schema content divided into groups according to their types (*tables, views, functions,* etc.). The popup menu of each tab allows you to create new, edit, copy or drop the appropriate schema object. The grid allows you to operate with several objects at a time. For this purpose select objects with the **Shift** or the **Ctrl** key pressed. After a group of objects is selected you can operate with them, e.g. *delete several objects* at once, as if it is a single object.

Name

Here you can view the schema name.

Owner

This field allows you to modify the schema owner. By default, only the owner of an object can perform various operations with the object. In order to allow other users to operate it, privileges must be granted. (However, users that have the superuser attribute can always access any object.)Microsoft SQL denies to change owners of system schemas.

Use the Comment field to set the schema description.

To apply the changes, select the Apply Changes item in the Navigation bar or use **Ctrl+F9** or **Ctrl+F7** shortcut keys.

It is also possible to modify object properties without opening the object editor: use the Object Properties item of the popup menu of the selected object from the explorer tree.

5.2 Tables

MS SQL Maestro allows you to manipulate tables with easy: add new tables to the database, modify existing ones, browse table options and data. The sections below describe each of these actions in detail.

⊟ How can I add a new table?

New tables are created within Create Table Wizard 77. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the Table icon in the Create Database Object dialog

or

- select the Tables list or any object from that list in the explorer tree;
- select the Create New Table... item from the popup menu.

To create a new table with the same properties as one of the existing tables has:

- select the Object | Duplicate Database Object... main menu item;
- follow the instructions of Duplicate Object Wizard.

⊟ How can I work with an existing table?

Tables can be edited within Table Editor 79. In order to run the editor you should either

- select the table for editing in the explorer tree (type the first letters of the table name for quick search);
- select the Edit Table... item from the popup menu

or

- open Schema (Database) Editor and the Tables tab there;
- select the table to edit;
- press the **Enter** key or select the Edit Table item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

You can also view and edit table properties without launching Table Editor:

- select the table for editing in the explorer tree (type the first letters of the table name for quick search);
- select the Table Properties... item from the popup menu;
- edit table properties within the Table Properties dialog.

You can change the name of the table using the Rename Table dialog. To open the dialog you should either

- select the table to rename in the explorer tree;

- select the Rename Table item from the popup menu

or

- open Schema (Database) Editor and the Tables tab there;
- select the table to rename;
- select the Rename Table item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

How can I drop the existing table?

To drop a table:

- select the table to drop in the explorer tree;
- select the Drop Table item from the popup menu

or

- open Schema (Database) Editor and the Tables tab there;
- select the table to drop;
- press the **Delete** key or select the Drop Table item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.

Table Editor allows you to work with table data including master-detail views 81, generate simple SQL statements 408, CRUD procedures 409 to work with this table, and split the table 412 into two separate tables.

5.2.1 Create Table Wizard

Create Table Wizard guides you through the process of creating a new database table. The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

Name
The name of the table being created as it was specified at the previous step.

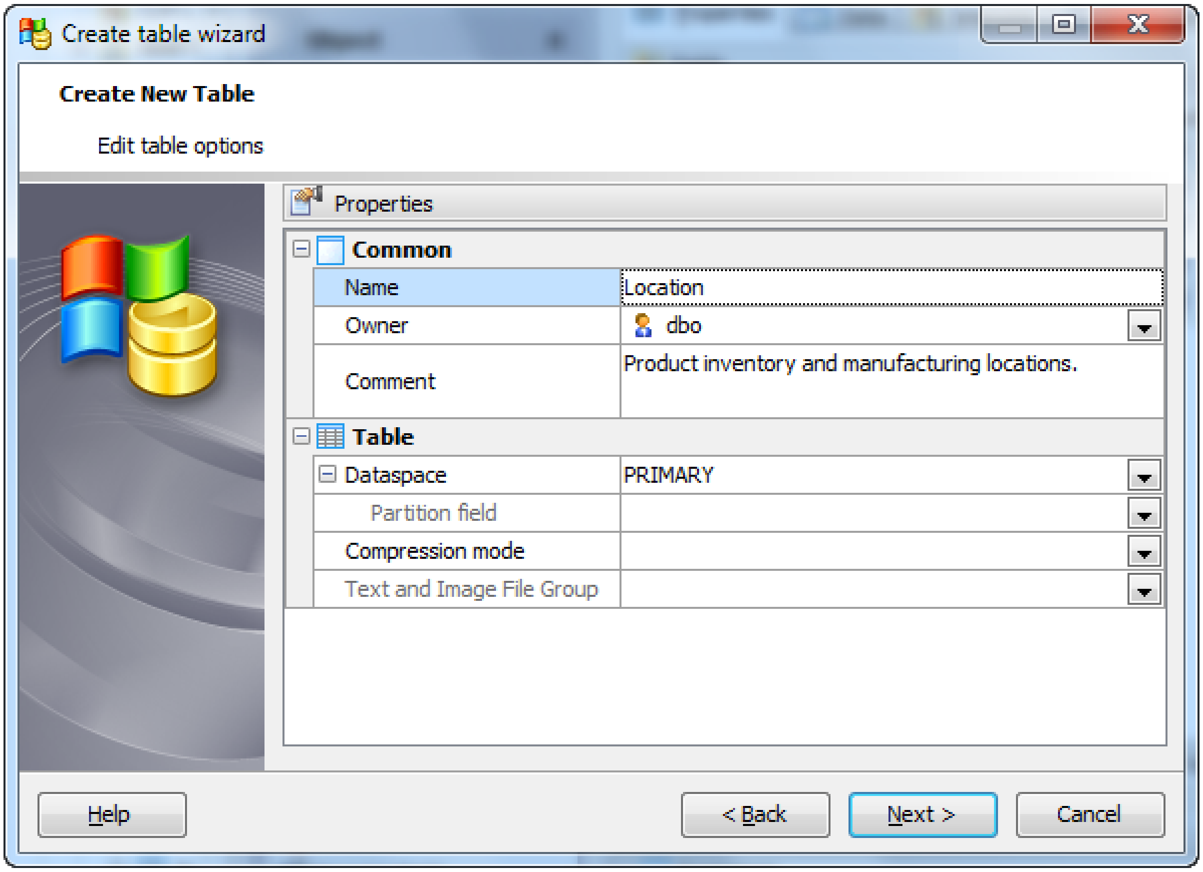
Owner
You can specify here the name of the Microsoft SQL server user that will own the new table, or leave this field blank to use the default user (namely, the user executing the command). By default, only the owner of an object can perform various operations with the object. In order to allow other users to operate it, privileges must be granted. (However, users that have the superuser attribute can always access any object.)

Comment
Set the optional text to describe the new table.

File Group
Put the file group 211 name to associate with the new table, or leave this field blank to use the template table file group. This file group will be the default one used for objects created in this table. The default file group is based on the template table file group.

Define the Text and Image File Group keywords to indicate that the *text, ntext, image, xml, varchar(max), nvarchar(max), varbinary(max),* and *CLR user-defined type* columns are stored on the specified file group or leave this field blank to use the default file group.

Note: The text and image file group is not allowed if there are no large value columns in the table.



Common

Name	Location
Owner	dbo
Comment	Product inventory and manufacturing locations.

Table

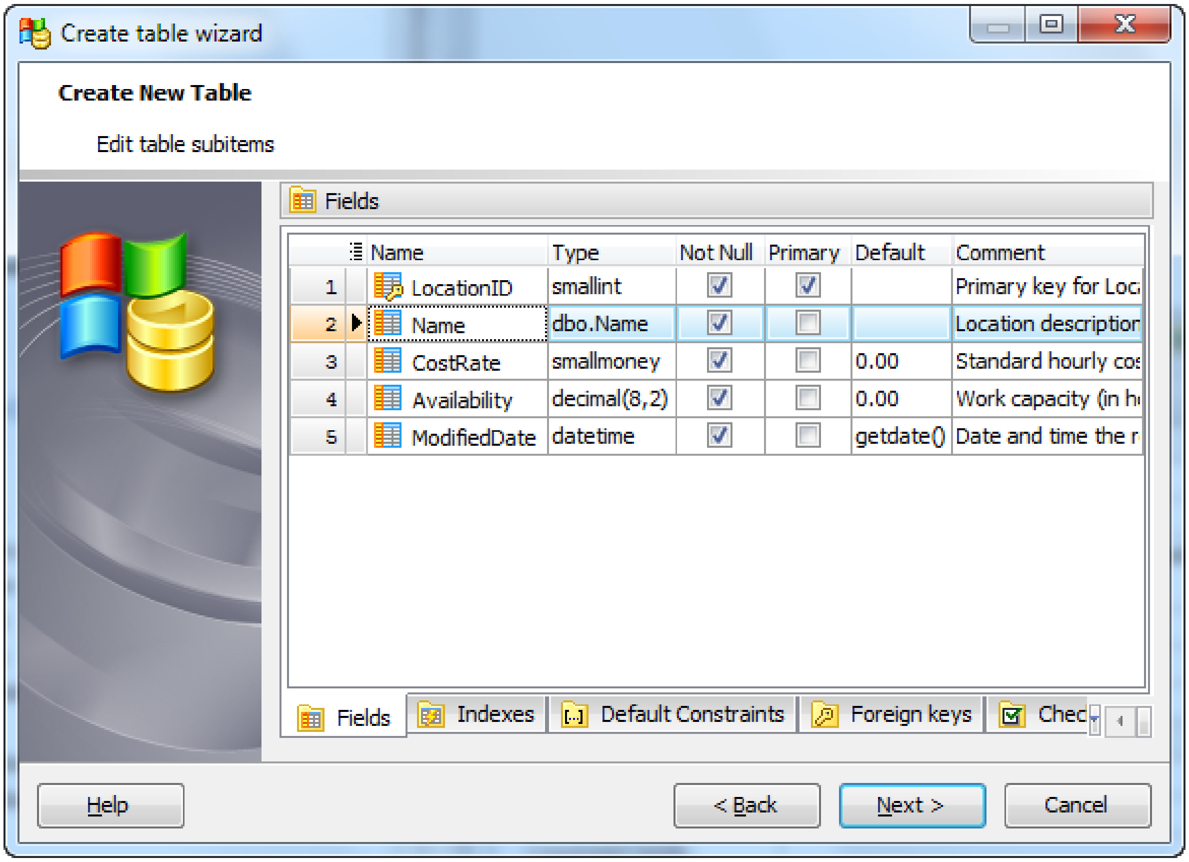
Dataspace	PRIMARY
Partition field	
Compression mode	
Text and Image File Group	

Adding table subitems

On this step of the wizard you can fullfill the new table with fields, indexes, and foreign keys. To add a new object:

- Choose the necessary page (Fields - to add table fields, Indexes - table indexes, and so on);
- Follow the corresponding link of the tab's pop-up menu;
- Specify properties of the new object. To find the description of field 83, foreign key 90, check 92, trigger 99, default constraint 95, and index 87, follow the according link.

The popup menu of each tab allows to edit, drop, reorder, and rename specified objects, etc.



	Name	Type	Not Null	Primary	Default	Comment
1	LocationID	smallint	☑	☑		Primary key for Loc
2	Name	dbo.Name	☑	☐		Location description
3	CostRate	smallmoney	☑	☐	0.00	Standard hourly cos
4	Availability	decimal(8,2)	☑	☐	0.00	Work capacity (in h
5	ModifiedDate	datetime	☑	☐	getdate()	Date and time the r

Click Add All or Add to include table(s) to table definition. Use the Remove or Remove All items to exclude table(s) from the list.

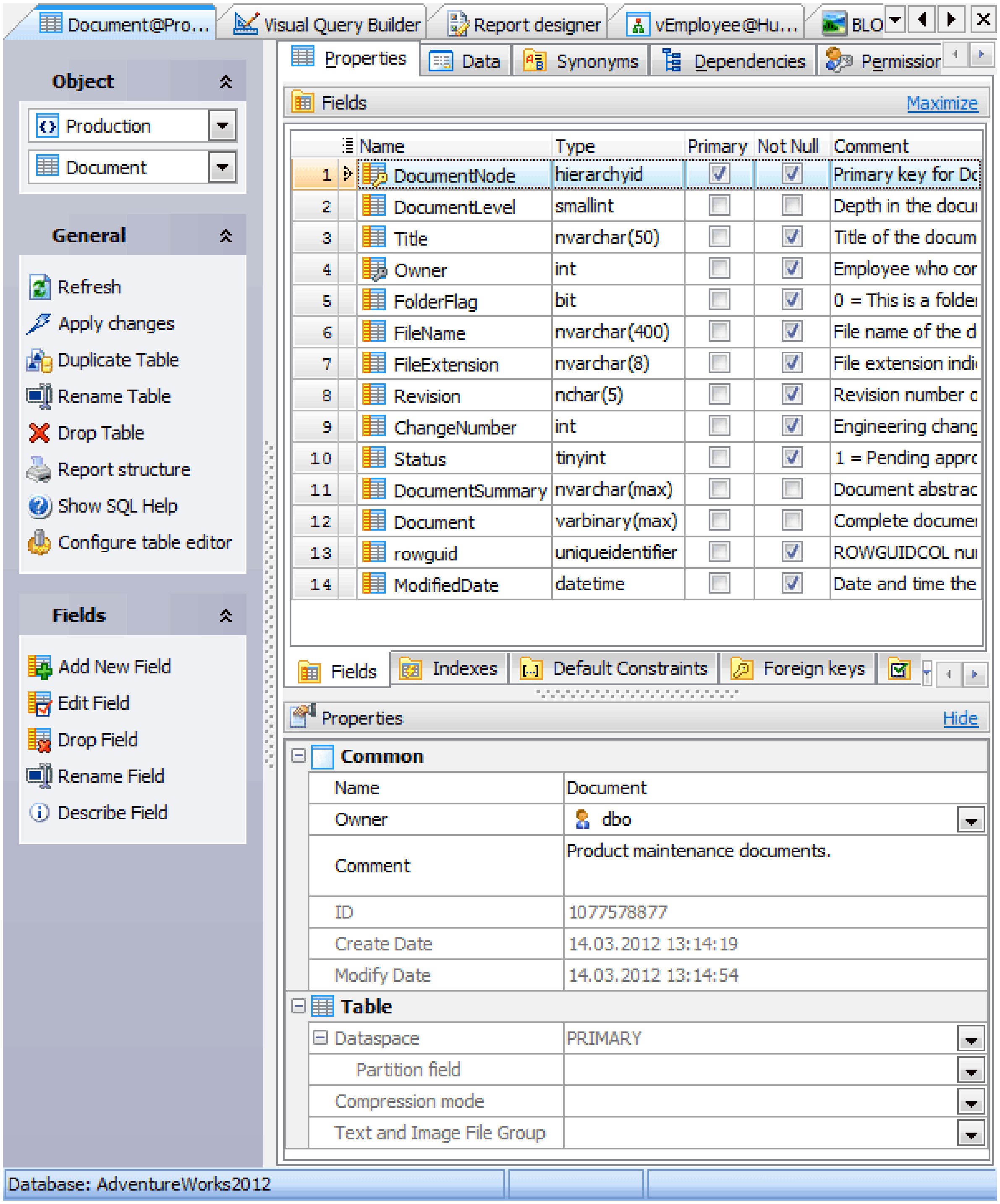
5.2.2 Table Editor

Table Editor allows you to create, edit and drop table fields, indexes, foreign keys, manage table data and other table subobjects. It can be opened automatically after the table is created and is available on editing the table. To open Table Editor, double-click the corresponding node at the Explorer Tree or Object Manager.
The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.

- Editing table properties 79
- Viewing table data 81

5.2.2.1 Editing table properties

The Properties section allows you to view general table properties and also to modify the table name, the table owner, and a comment for the table.



	Name	Type	Primary	Not Null	Comment
1	DocumentNode	hierarchyid	☑	☑	Primary key for Dc
2	DocumentLevel	smallint	☐	☐	Depth in the docu
3	Title	nvarchar(50)	☐	☑	Title of the docum
4	Owner	int	☐	☑	Employee who cor
5	FolderFlag	bit	☐	☑	0 = This is a folde
6	FileName	nvarchar(400)	☐	☑	File name of the d
7	FileExtension	nvarchar(8)	☐	☑	File extension indi
8	Revision	nchar(5)	☐	☑	Revision number c
9	ChangeNumber	int	☐	☑	Engineering chang
10	Status	tinyint	☐	☑	1 = Pending appr
11	DocumentSummary	nvarchar(max)	☐	☐	Document abstrac
12	Document	varbinary(max)	☐	☐	Complete docume
13	rowguid	uniqueidentifier	☐	☑	ROWGUIDCOL nu
14	ModifiedDate	datetime	☐	☑	Date and time the

Common

Name	Document
Owner	dbo
Comment	Product maintenance documents.
ID	1077578877
Create Date	14.03.2012 13:14:19
Modify Date	14.03.2012 13:14:54

Table

Dataspace	PRIMARY
Partition field	
Compression mode	
Text and Image File Group	

Subitems

Every tab is intended for work with defined objects (*fields*, *indexes,* etc.). To modify any object, double click it or use grid's popup menu. The menu also allows you to add new, rename, describe, copy/paste, and drop selected objects. To operate with several objects at a time, select them with the **Shift** or the **Ctrl** key pressed. After a group of objects is selected you can operate with it, e.g. *delete several objects at once*, as if it is a single object.

See also: Fields 83, Foreign Keys 90, Checks 92, Triggers 100, Default Constraints 95, and Indexes 87.

Owner
You can view and modify the name of the table Owner.
By default, only the owner of an object can perform various operations with the object. In order to allow other users to operate it, privileges must be granted. (However, users that have the superuser attribute can always access any object.)

Comment
This field contains a comment to the table.
Create Date
Displays the date when the table was created.

Modify Date
Displays the date when the table was last modified.

File Group
Represents the file group on which the table is stored.

Text and Image File Group
Represents the text and image file group.

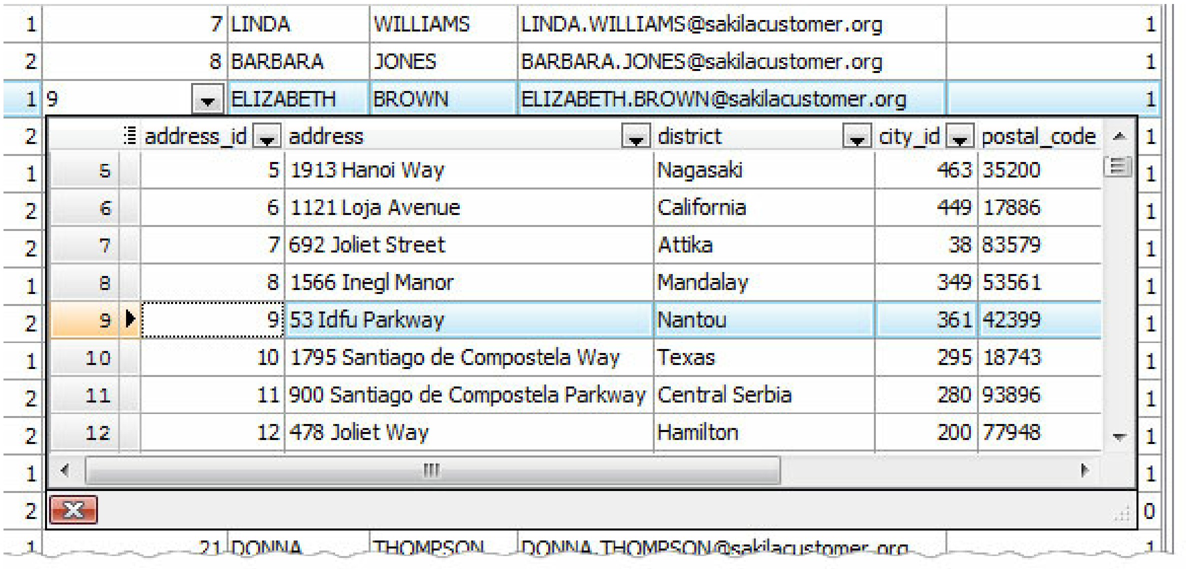
To apply the changes, select the Apply Changes item in the Navigation bar or use **Ctrl+F9** or **Ctrl+F7** shortcut keys.

It is also possible to modify object properties without opening the object editor: use the Object Properties item of the popup menu of the selected object from the explorer tree.

5.2.2.2 Managing table data

The Data tab displays the table data as a grid or as info cards (see Data View 328 for details). To edit/add a table record, use Data Input Form or type the new data directly in the grid (card). To export/import/get SQL dump of the table data, invoke corresponding modules from the grid's popup menu. To view and edit the content of BLOB columns, run BLOB Editor 339.

Lookup editors
Lookup editor displays the content of parent table’s columns within the drop-down window. MS SQL Maestro enables a lookup editor for a column linked by a foreign key with a single column from another table. To get the corresponding data, double click the field or use **F2** shortcut and press **Alt+Down Arrow Key**.



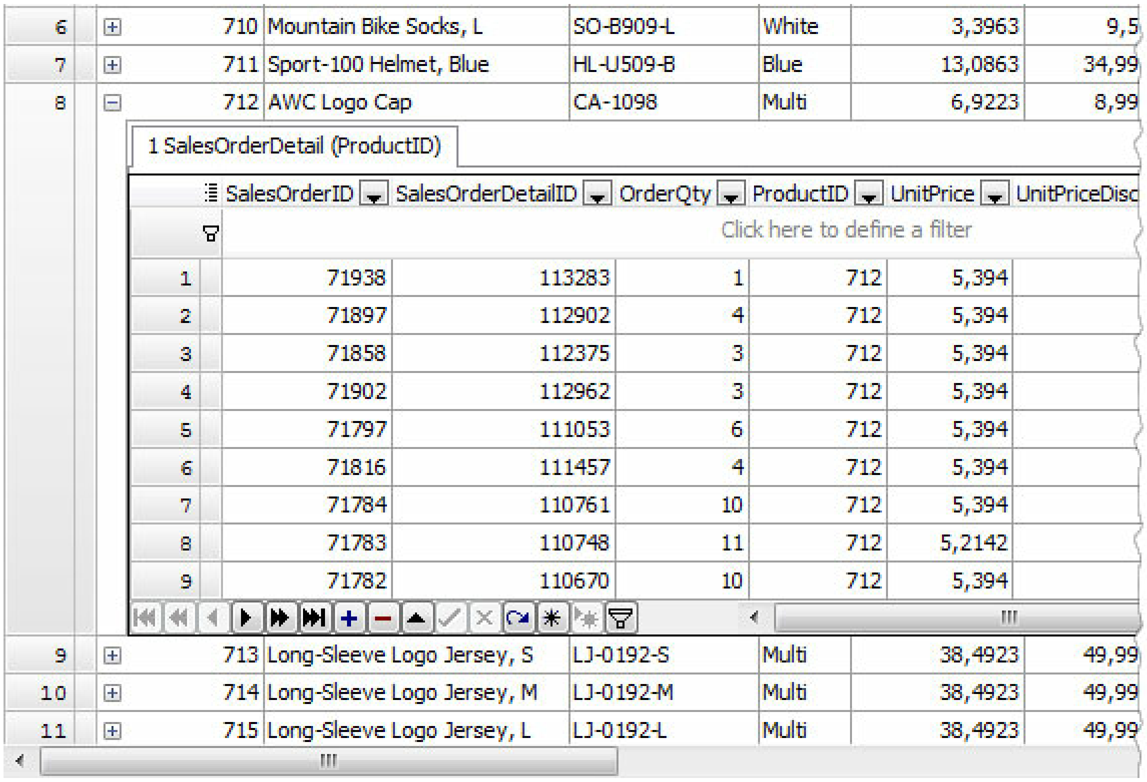
1	7	LINDA	WILLIAMS	LINDA.WILLIAMS@sakilacustomer.org	1
2	8	BARBARA	JONES	BARBARA.JONES@sakilacustomer.org	1
1	9	ELIZABETH	BROWN	ELIZABETH.BROWN@sakilacustomer.org	1

	address_id	address	district	city_id	postal_code
5	5	1913 Hanoi Way	Nagasaki	463	35200
6	6	1121 Loja Avenue	California	449	17886
7	7	692 Joliet Street	Attika	38	83579
8	8	1566 Inegl Manor	Mandalay	349	53561
9	9	53 Idfu Parkway	Nantou	361	42399
10	10	1795 Santiago de Compostela Way	Texas	295	18743
11	11	900 Santiago de Compostela Parkway	Central Serbia	280	93896
12	12	478 Joliet Way	Hamilton	200	77948

1	21	DONNA	THOMPSON	DONNA.THOMPSON@sakilacustomer.org	1

Master-Detail Data View

To get data in the master-detail view mode (multiple detail pages are displayed for a single master row), use the Show/Hide details link at the editor's navigation bar. This mode allows you add/edit/delete data of detail pages. To open/close the appropriate detail page click the +/- icon or use **+/-** shortcuts.



6	710	Mountain Bike Socks, L	SO-B909-L	White	3,3963	9,5
7	711	Sport-100 Helmet, Blue	HL-U509-B	Blue	13,0863	34,99
8	712	AWC Logo Cap	CA-1098	Multi	6,9223	8,99

1 SalesOrderDetail (ProductID)

Click here to define a filter

	SalesOrderID	SalesOrderDetailID	OrderQty	ProductID	UnitPrice	UnitPriceDisc
1	71938	113283	1	712	5,394	
2	71897	112902	4	712	5,394	
3	71858	112375	3	712	5,394	
4	71902	112962	3	712	5,394	
5	71797	111053	6	712	5,394	
6	71816	111457	4	712	5,394	
7	71784	110761	10	712	5,394	
8	71783	110748	11	712	5,2142	
9	71782	110670	10	712	5,394	

9	713	Long-Sleeve Logo Jersey, S	LJ-0192-S	Multi	38,4923	49,99
10	714	Long-Sleeve Logo Jersey, M	LJ-0192-M	Multi	38,4923	49,99
11	715	Long-Sleeve Logo Jersey, L	LJ-0192-L	Multi	38,4923	49,99

Import from Clipboard

MS SQL Maestro supports data import from clipboard. It is supposed that columns within the data block are separated by the tabulation symbol, records are separated by newlines and the first line of the data block contains column headers.

Example:

ColHeader1	ColHeader2
R1C1	R1C2
R2C1	R2C2

The same data format is supported by a lot of other applications, so the ability allows you to copy data from MS Excel, another table or view, or even from a data set from a different DBMS especially if it is opened with an appropriate our product.

Uploading files as BLOBs

MS SQL Maestro allows you to upload files as BLOBs into a table. For this purpose the file names must contain the information on the record they need to be placed to: the files need to be named in the same manner and include content of one or several table columns that can uniquely identify each row. To import files, specify the file name template using filed name tags (i.e. %id%, %user%, where 'id' and 'user' are the fact table columns). You can also set the default file to be uploaded to NULL fields.

Example:
Suppose we have a table 'employee' with Non-Blob data as follows:

Id	User
1	Max
2	July

And we need to import the 1.jpg and 2.jpg files to a BLOB column of the table. The files are stored in the "D:\Images" directory. In this case we need to specify the "D:\Images\%Id%.jpg" file name template.

5.2.3 Fields

Table columns are created and edited within the Field Editor.

How to add a new column to a table?

To add a new table column, you should either:

- open the table in Table Editor and the Fields tab there;
- press the **Insert** key or select the Add New Field... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

or

- select the table in the explorer tree and use the Create New Field popup menu item

or

- select the table Fields node or any field within the table in the explorer tree and use the Add New Field... popup menu item.

How to edit an existing table field?

Table fields are edited within the Field Editor 83 dialog window. In order to open the dialog you should either

- open the table in Table Editor and the Fields tab there;
- press the **Enter** key or select the Edit Field item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

or

- select the field to edit in the explorer tree and use the Edit Field popup menu item.

You can change the name of the field using the Rename Field dialog. To open the dialog you should either

- select the field to rename in the explorer tree;
- select the Rename Field item from the popup menu

or

- open the table in Table Editor and the Fields tab there;
- select the field to rename;
- select the Rename Field item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

⊟ How to drop an existing table field?

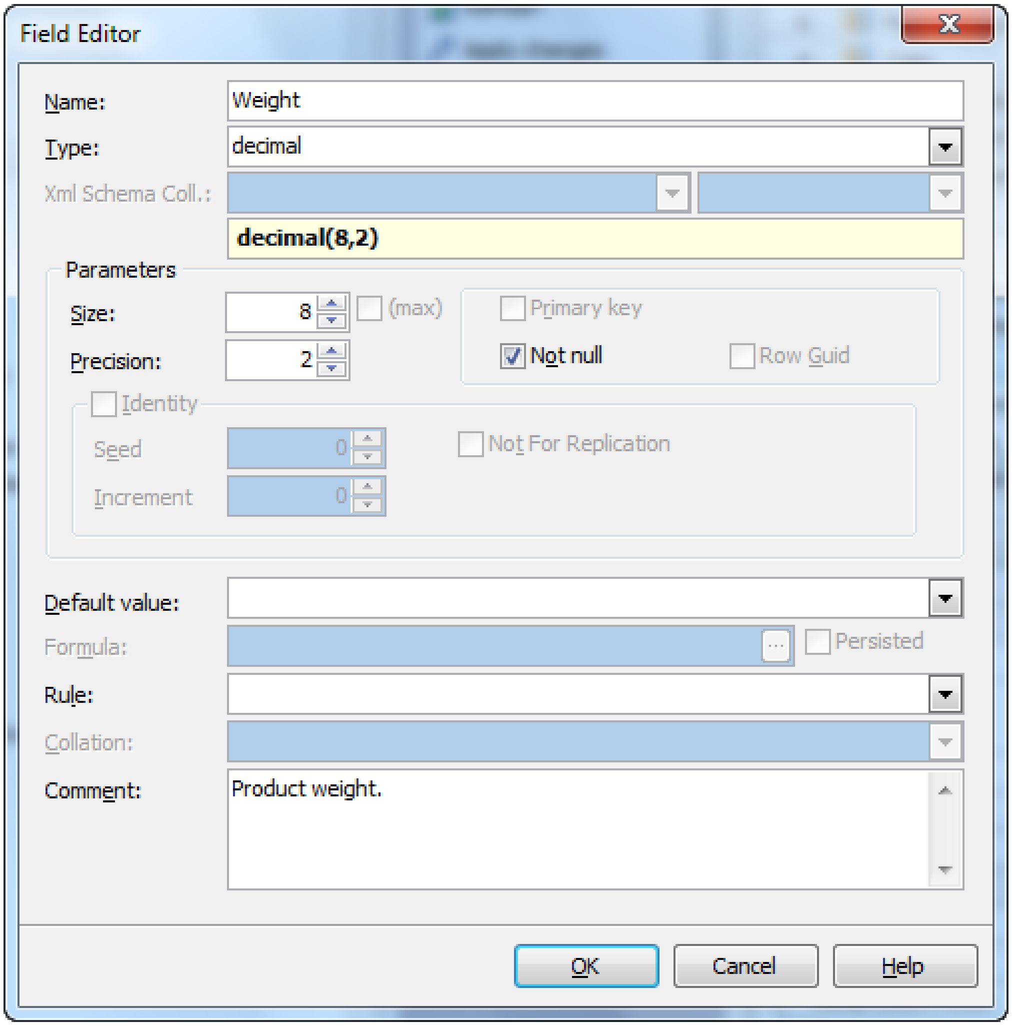
To drop the table field:

- select the field to drop in the explorer tree;
- select the Drop Field item from the popup menu

or

- open the table in Table Editor and the Fields tab there;
- press the **Delete** key or select the Drop Field item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.



Name: Weight
Type: decimal
decimal(8,2)
Size: 8 ☐ (max)
Precision: 2
☐ Primary key
☑ Not null ☐ Row Guid
☐ Identity
Seed 0
Increment 0
☐ Not For Replication
☐ Persisted
Comment: Product weight.

To specify the Data Type, select it from the drop-down list.

> **Note:** the name of the object must be unique among all the object names in the table. You can use any identifier that is allowed by Microsoft SQL server.

XML schema collection

Used only in case ***xml*** is selected as **Type**. Applies only to the XML data type for associating an XML schema collection with the type. Before using you are to create the schema in the database first.

See also: XML schema collection 169

Content|Document

Denotes that each instance of the xml data type can contain multiple top-level elements (for Content) or only one top-level element (for Document).

Parameters

Use the Size edit box to define the length of the field value for integer, float, char and other data types and use Precision to define the precision of the field value, e.g. for *float* data type.

The max value applies only to the varchar, nvarchar, and varbinary data types for storing 2^31 bytes of character and binary data, and 2^30 bytes of Unicode data.

Field flags

☑ Not Null
Forbids the NULL values for the field.

☑ Unique
Includes the field into the unique key (index).

☑ Primary Key
With this option checked the field becomes the only field with a primary key. If you check this field, you will not be able to set this attribute for any other field in the table. Hence if you want to create a compound primary key, do not check this field but create a primary key through the Indexes tab of Table Editor 79 or the appropriate step of Create Table Wizard 77.

☑ Row Guid
Indicates that the new column is a row GUID column.

Formula
Represents an expression that defines the value of a computed column. A computed column is a virtual column that is not physically stored in the table unless the column is checked Persisted.

Rule
Binds the rule to a column data type. The rule is to be created in the database beforehand.

See also: Rules 145

Default value
Within the box you can assign a default value for the field column. The action is optional. If the default value was specified during the new row created and no values is specified for some of the columns, the columns will be filled with their respective default values.

The Comment box allows you to set optional text describing the field.

Check the Identity option to indicate that the new column is an identity column.

Seed
Defines the value used for the very first row loaded into the table.

Increment
Defines the incremental value added to the identity value of the previous row loaded.

Not For Replication
If checked, values are not incremented in identity columns when replication agents perform inserts.

5.2.4 Indexes

Indexes are primarily used to enhance database performance (though inappropriate use may result in slower performance). The key field(s) for the index are specified as column names. Multiple fields can be specified if the index method supports multicolumn indexes.

⊟ How can I create a table index?

Table indexes are created within the Index Properties 87 dialog window. In order to open the dialog you should either

- open the table in Table Editor and the Indexes tab there;
- press the **Insert** key or select the Add New Index... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

or

- select the table in the explorer tree and use the Create New Index popup menu item

or

- select the table Indexes node or any index within the table in the explorer tree and use the Add New Index... popup menu item.

⊟ How can I edit an existing index?

Table indexes are edited within the Index Properties 87 dialog window. In order to open the dialog you should either

- open the table in Table Editor and the Indexes tab there;
- press the **Enter** key or select the Edit Index item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

or

- select the index to edit in the explorer tree and use the Edit Index popup menu item.

You can change the name of the index using the Rename Index dialog. To open the dialog you should either

- select the index to rename in the explorer tree;
- select the Rename Index item from the popup menu

or

- open the table in Table Editor and the Indexes tab there;
- select the index to rename;
- select the Rename Index item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

How can I drop a table index?

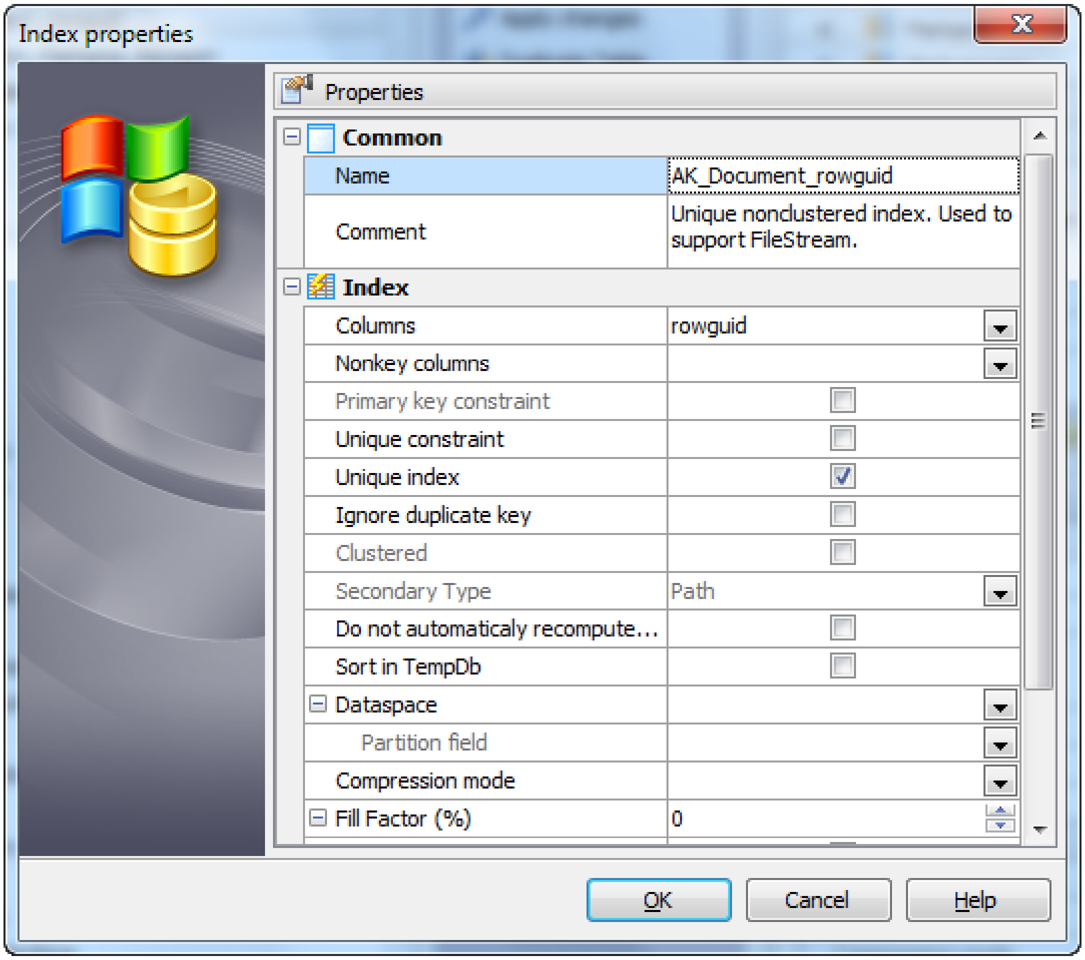
To drop the table index:

- select the index to drop in the explorer tree;
- select the Drop Index item from the popup menu

or

- open the table in Table Editor and the Indexes tab there;
- press the **Delete** key or select the Drop Index item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.



Name	AK_Document_rowguid
Comment	Unique nonclustered index. Used to support FileStream.
Columns	rowguid
Nonkey columns	
Primary key constraint	☐
Unique constraint	☐
Unique index	☑
Ignore duplicate key	☐
Clustered	☐
Secondary Type	Path
Do not automaticaly recompute...	☐
Sort in TempDb	☐
Dataspace	
Partition field	
Compression mode	
Fill Factor (%)	0

Use the Columns drop-down list to select a key field(s) for the index.

☑ Primary key constraint
With this option checked this field becomes a compound primary key. It is useful in case the table has more than one primary key.

☑ Unique Index
If checked, creates a unique index for the table, i.e. the database system ensures that no two rows of the specified table have the same values in the indexed columns. In this way, if two rows both contain the NULL value for all columns of an index, the two index values are not considered to be identical. If at least one column does not contain the NULL value, two rows that have the same value in all non-NULL columns are considered to be identical.

Constraint
Creates an index on a specified table as a table constraint.

Index
Creates an index on a specified table as database object.

☑ Ignore duplicate key
Specifies the error response to duplicate key values in a multiple-row INSERT transaction on a unique clustered or unique nonclustered index. If checked, a warning message is issued and only the rows violating the unique index fail. If not checked, an error message is issued and the entire INSERT transaction is rolled back.

Index Filegroup
If a Filegroup is specified the index is stored in the named filegroup.

Fill Factor
Specifies the percentage that indicates how full the Database Engine should make the leaf level of each index page during index creation or change.

☑ Pad Index
Specifies index padding. If checked, the percentage of free space portioned by Fill Factor is applied to the intermediate-level pages of the index.

Secondary Type
Specifies the type of secondary XML index.

Create as CLUSTERED
If checked, creates an index in which the logical order of the key values determines the physical order of the corresponding rows in a table.

Do not automatically recompute statistics
If checked, out-of-date statistics are not automatically recomputed.

Primary
Defines primary key.

Sort in TempDB
Specifies whether to store sort results in a TempDB. If checked, the intermediate sort results that are used to build the index are stored in a TempDB.Fill factor
Use the parameter to determine how full the index method will try to pack index pages. For B-trees, leaf pages are filled to this percentage during initial index build, and also when extending the index at the right (largest key values). If pages subsequently become completely full, they will be split, leading to gradual degradation in the index's efficiency. B-trees use a default fill factor of 90, but any value from 10 to 100 can be selected. If the table is static then fill factor 100 is best to minimize the index's physical

size, but for heavily updated tables a smaller fill factor is better to minimize the need for page splits. The other index methods use fillfactor in different but roughly analogous ways; the default fillfactor varies between methods.

5.2.5 Foreign Keys

A foreign key is a field (or collection of fields) in one table that uniquely identifies a row of another table. In other words, a foreign key is a column or a combination of columns that is used to establish and enforce a link between the data in two tables.

> **Note:** To create a foreign key constraint, it is necessary to have this privilege for both the referencing and referenced tables.

How can I add a new foreign key?

Foreign keys are created within the Foreign Key Properties 90 dialog window. In order to open the dialog you should either

- open the table in Table Editor and the Foreign Keys tab there;
- press the **Insert** key or select the Add New Foreign Key... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

or

- select the table in the explorer tree and use the Create New Foreign Key popup menu item

or

- select the table Foreign Keys node or any foreign key within the table in the explorer tree and use the Add New Foreign Key... popup menu item.

How can I edit an existing foreign key?

Foreign Keys are edited within the Foreign Key Properties 90 dialog window. In order to open the dialog you should either

- open the table in Table Editor and the Foreign Keys tab there;
- press the **Enter** key or select the Edit Foreign Key item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

or

- select the foreign key to edit in the explorer tree and use the Edit Foreign Key popup menu item.

You can change the name of the foreign key using the Rename Foreign Key dialog. To open the dialog you should either

- select the foreign key to rename in the explorer tree;
- select the Rename Foreign Key item from the popup menu

or

- open the table in Table Editor and the Foreign Keys tab there;
- select the foreign key to rename;
- select the Rename Foreign Key item from the popup menu

(alternatively, you may use the corresponding link of the Navigation Bar).

How can I drop a foreign key?

To drop the foreign key:

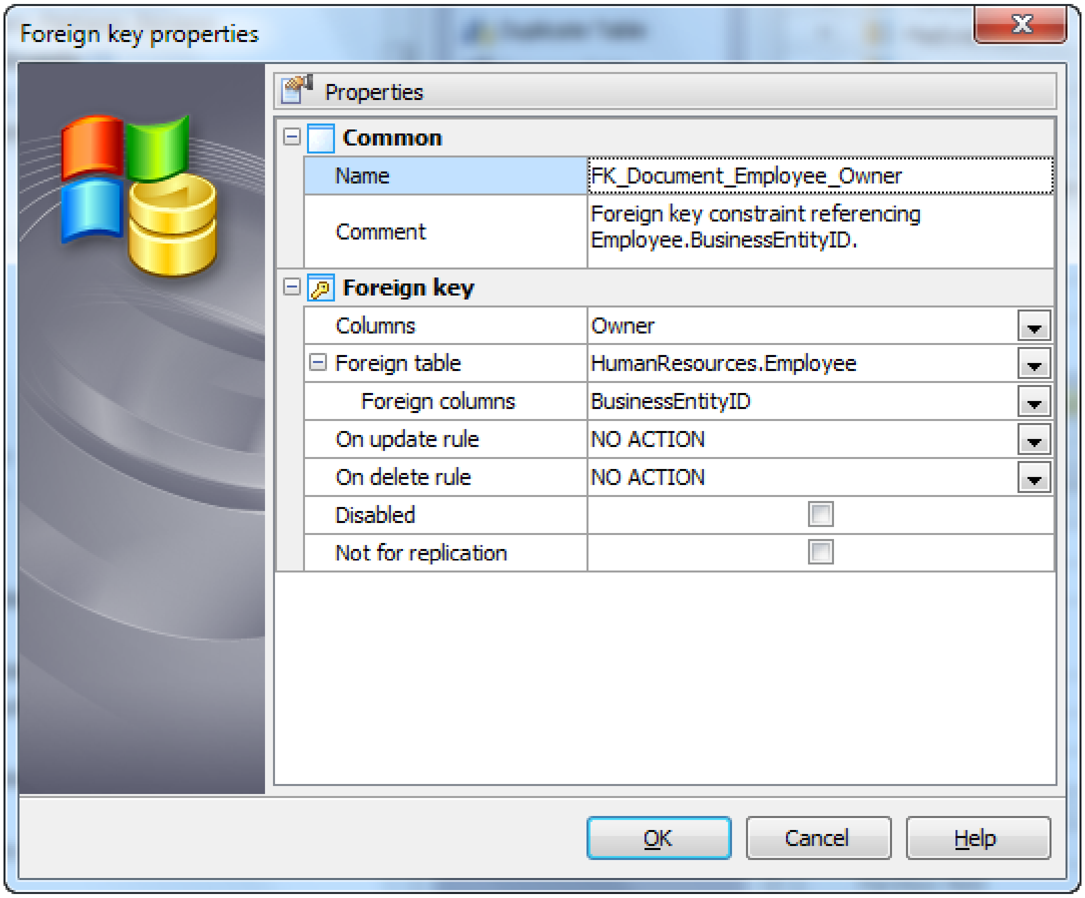
- select the foreign key to drop in the explorer tree;
- select the Drop Foreign Key item from the popup menu

or

- open the table in Table Editor and the Foreign Keys tab there;
- press the **Delete** key or select the Drop Foreign Key item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.

Set the Foreign Key Name, select Columns from the Available Fields list to include into the foreign key, select the Foreign Table Name from the drop-down list and its fields from the list to include, set other foreign key properties and apply the changes by clicking the OK button.



Common	
Name	FK_Document_Employee_Owner
Comment	Foreign key constraint referencing Employee.BusinessEntityID.
Foreign key	
Columns	Owner
Foreign table	HumanResources.Employee
Foreign columns	BusinessEntityID
On update rule	NO ACTION
On delete rule	NO ACTION
Disabled	☐
Not for replication	☐

All the fields which are included into the Foreign Key must be included into indexes as well. See Indexes 87 for details.

Comment
The box allows you to set optional text describing the foreign key.

☑ Check Existing Data
Specifies whether the data in the table is or is not validated against a newly added or re-enabled FOREIGN KEY constraint.

☑ Disabled
When checked, disables foreign key.

Set rules ON DELETE and ON UPDATE from the respective drop-down lists.

- *NO ACTION* Produce an error indicating that the deletion or update will create a foreign key constraint violation. If the constraint is deferred this error will be produced at constraint check time if there still exist any referencing rows. This is the default action.

- *CASCADE* Delete any rows referencing the deleted row, or update the value of the referencing column to the new value of the referenced column, respectively.

- *SET NULL* Set the referencing column(s) to null.

- *SET DEFAULT* Set the referencing column(s) to their default values.

5.2.6 Checks

A check constraint is the most generic constraint type. It allows you to specify that the value in a certain column must satisfy a Boolean (truth-value) expression.

The Check Properties editor allows you to add a new check constraint or edit an existing one. This dialog can be invoked from Table Editor 79, or via the popup menu of the corresponding nodes of the explorer tree.

⊟ How can I add a new check?

Checks are created within Check Properties 92. In order to run the wizard you should either

- open the table in Table Editor and the Checks tab there;
- press the **Insert** key or select the Add New Check... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

or

- select the table in the explorer tree and use the Create New Check... popup menu item

or

- select the table Checks node or any check within the table in the explorer tree and use the Add New Check... popup menu item.

How can I edit an existing check?

Checks are edited within the Check Properties 92 dialog window. In order to open the dialog you should either

- open the table in Table Editor the Checks tab there;
- press the **Enter** key or select the Edit Check item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

or

- select the check to edit in the explorer tree and use the Edit Check popup menu item.

You can change the name of the check using the Rename Check dialog. To open the dialog you should either

- select the check to rename in the explorer tree;
- select the Rename Check item from the popup menu

or

- open the table in Table Editor and the Checks tab there;
- select the check to rename;
- select the Rename Check item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

How can I drop a check?

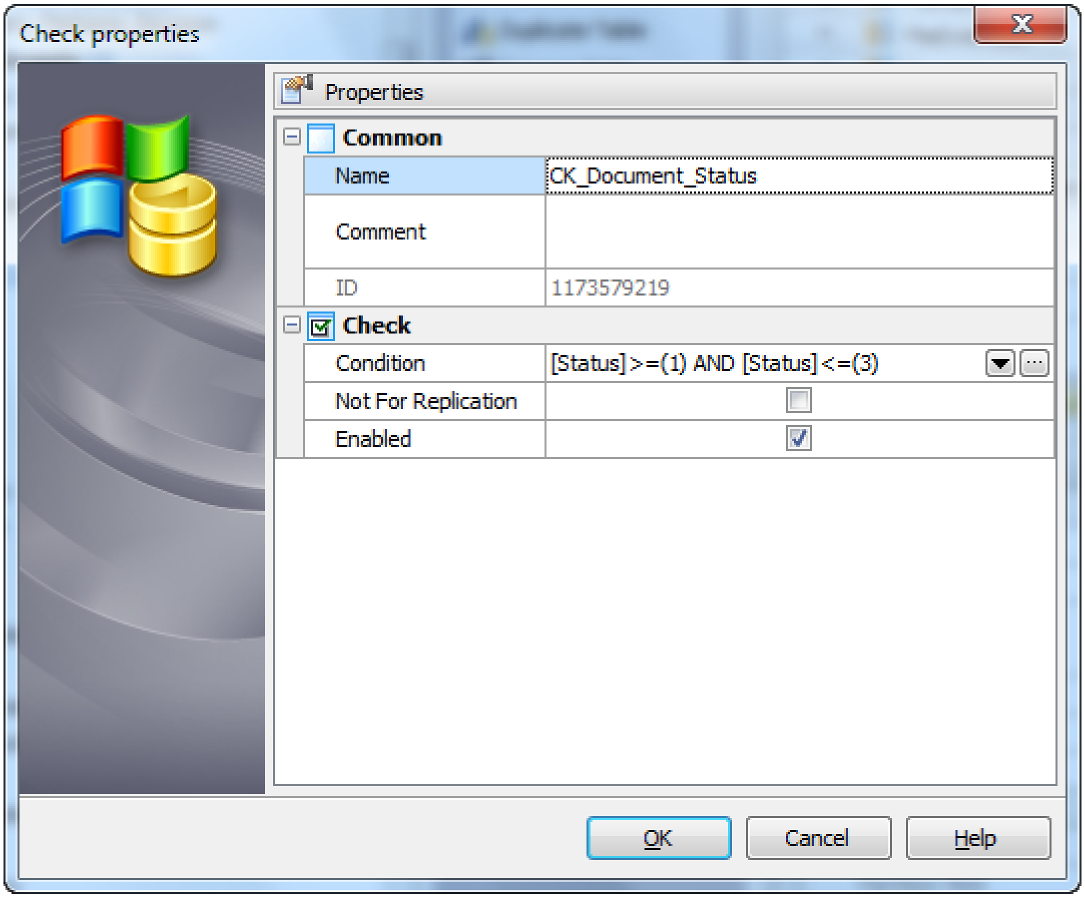
To drop the check:

- select the check to drop in the explorer tree;
- select the Drop Check item from the popup menu

or

- open the table in Table Editor and the Checks tab there;
- press the **Delete** key or select the Drop Check item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.



Common

Name	CK_Document_Status
Comment	
ID	1173579219

Check

Condition	[Status]>=(1) AND [Status]<=(3)
Not For Replication	☐
Enabled	☑

Name
The name of the check constraint.

Comment
This field contains an optional text describing the check.

Condition
Specify an expression producing a Boolean result which new or updated rows must satisfy for an insert or update operation to succeed. Expressions evaluating to True or Unknown succeed. In case any row of an insert or update operation produce a FALSE result an error exception is raised and the insert or update does not alter the database.

☑ Check Existing Data
Specifies whether the data in the table is or is not validated against a newly added or re-enabled CHECK constraint.

☑ Not For Replication
If checked, the constraint is not enforced when replication agents perform insert, update, or delete operations.

☑ Disabled
If the option is selected, the check is disabled.

5.2.7 Default constraints

Default Constraints can be used to provide values for a new column in the existing rows of data. SQL Server defaults provide data to columns and user-defined data types when no other data is available on an INSERT statement execution. Constraints allow you to define the way Microsoft® SQL Server automatically enforces the integrity of a database. Constraints define rules regarding the values allowed in columns and are the standard mechanism for enforcing integrity.

⊟ How can I add a new default constraint?

Table default constraints are created within the Default Constraint Properties 95 dialog window. In order to open the dialog you should either

- open the table in Table Editor and the Default Constraints tab there;
- press the **Insert** key or select the Add Default Constraint... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

or

- select the table in the explorer tree and use the Create New Default Constraint popup menu item

or

- select the table Default Constraints node or any default constraint within the table in the explorer tree and use the Add New Default Constraint... popup menu item.

⊟ How can I edit an existing default constraint?

Table default constraints are edited within the Default Constraint Properties 95 dialog window. In order to open the dialog you should either

- open the table in Table Editor and the Default Constraints tab there;
- press the **Enter** key or select the Edit Default Constraint item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

or

- select the default constraint to edit in the explorer tree and use the Edit Default Constraint popup menu item.

You can change the name of the default constraint using the Rename Default Constraint dialog. To open the dialog you should either

- select the default constraint to rename in the explorer tree;
- select the Rename Default Constraint item from the popup menu

or

- open the table in Table Editor and the Default Constraints tab there;
- select the default constraint to rename;
- select the Rename Default Constraint item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

How can I drop a default constraint?

To drop the table default constraint:

- select the default constraint to drop in the explorer tree;
- select the Drop Default Constraint item from the popup menu

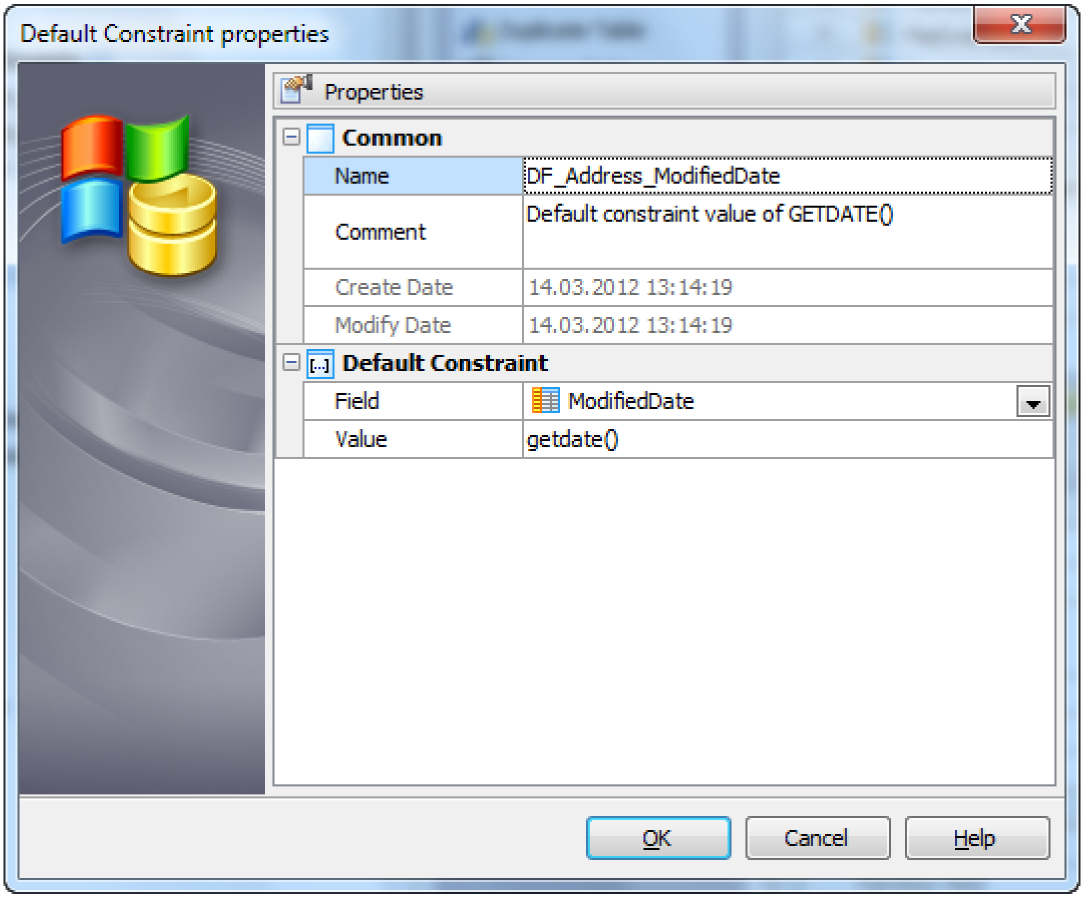
or

- open the table in Table Editor and the Default Constraints tab there;
- press the **Delete** key or select the Drop Default Constraint item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.

allows you to add a new default constraint or edit an existing one. It is available from Table Editor[79], or from the corresponding nodes of the explorer tree.

The Default Constraint Properties window allows you to set the constraint name, enter some optional text describing it, select a field to define the default constraint on, and also to specify the default value.



Common	
Name	DF_Address_ModifiedDate
Comment	Default constraint value of GETDATE()
Create Date	14.03.2012 13:14:19
Modify Date	14.03.2012 13:14:19
Default Constraint	
Field	ModifiedDate
Value	getdate()

Note: the name of the object must be unique among all the object names in its container. Moreover, all the objects that are source of data need unique names among themselves. You can use any identifier that is allowed by Microsoft SQL server.

Note: default constraints cannot be added to columns that have a *timestamp* data type, an *Identity* property, an existing default constraint, or a bound default. If the column has an existing default constraint, the default must be dropped before the new default can be added.

See also: Defaults 150

5.2.8 Triggers

A trigger is a specification that the database should automatically execute a particular function whenever a certain type of operation is performed. A trigger can be defined to execute before or after an INSERT, UPDATE, or DELETE operation, either once per modified row, or once per SQL statement. If a trigger event occurs, the trigger fires.

How can I add a new trigger?

Triggers are created within Create Trigger Wizard 99. In order to run the wizard you should either

- open the table in Table Editor and the Triggers tab there;
- press the **Insert** key or select the Add New Trigger... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

or

- select the table in the explorer tree and use the Create New Trigger... popup menu item

or

- select the table Triggers node or any trigger within the table in the explorer tree and use the Add New Trigger... popup menu item.

⊟ How can I edit an existing trigger?

Triggers are edited within the Trigger Editor 100 dialog window. In order to open the dialog you should either

- open the table in Table Editor and the Triggers tab there;
- press the **Enter** key or select the Edit Trigger item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

or

- select the trigger to edit in the explorer tree and use the Edit Trigger popup menu item.

You can change the name of the trigger using the Rename Trigger dialog. To open the dialog you should either

- select the trigger to rename in the explorer tree;
- select the Rename Trigger item from the popup menu

or

- open the table in Table Editor and the Triggers tab there;
- select the trigger to rename;
- select the Rename Trigger item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

⊟ How can I drop a trigger?

To drop the trigger:

- select the trigger to drop in the explorer tree;
- select the Drop Trigger item from the popup menu

or

- open the table in Table Editor and the Triggers tab there;
- press the **Delete** key or select the Drop Trigger item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.

5.2.8.1 Create Trigger Wizard

Create Trigger Wizard guides you through the process of creating of a new table trigger.

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

Specifying trigger properties

To define a new trigger, you need to set its.

Comment
This field contains a comment to the table trigger.

Execute As
Specifies the security context under which the trigger is executed.

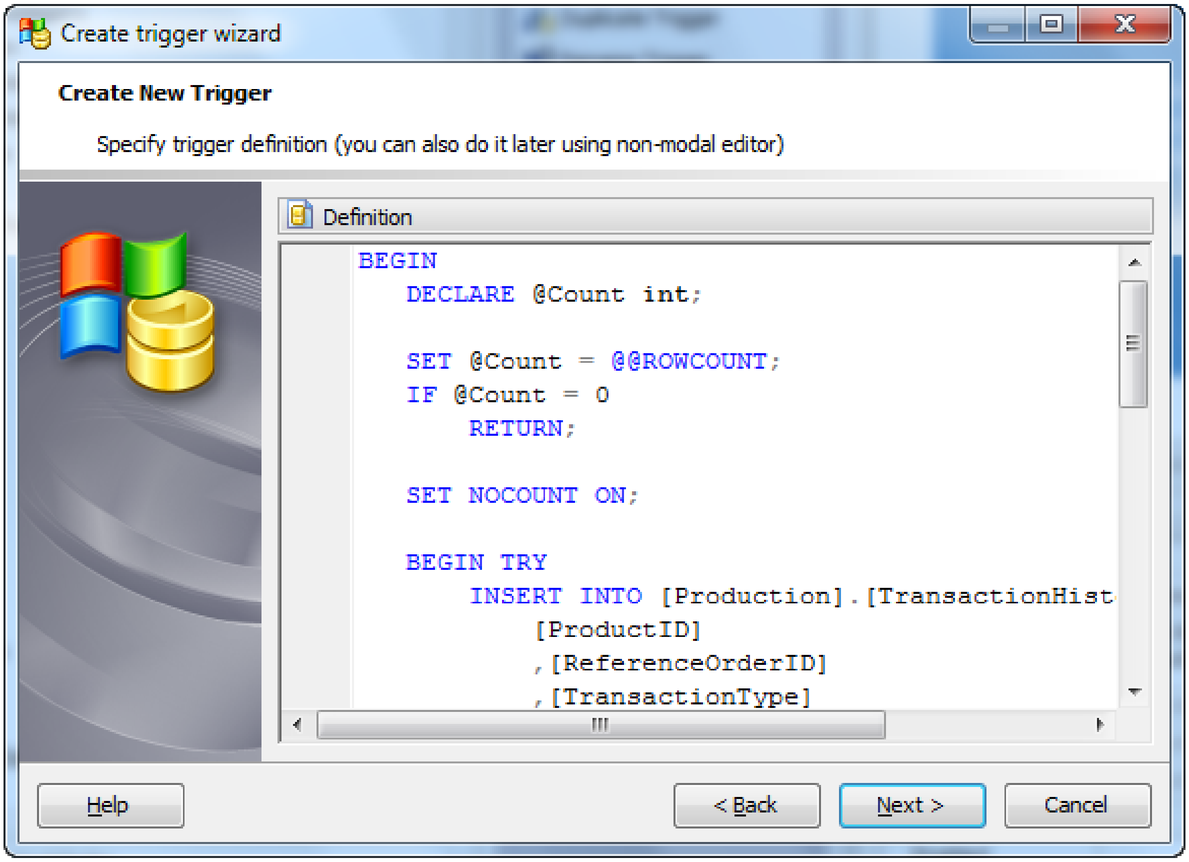
Type *(After, Instead of)*
Specifies the trigger type.
After: the trigger is fired only when all operations specified in the triggering SQL statement have executed successfully.
Instead Of: the trigger is executed instead of the triggering SQL statement, therefore, overriding the actions of the triggering statements.

Events
One of *Insert, Update,* or *Delete*; this specifies the event that will fire the trigger.

☑ Enabled
If checked, the trigger is enabled.

☑ Not For Replication
When checked, it is indicated that the trigger should not be executed when a replication agent modifies the table that is involved in the trigger.

Specifying trigger definition
Here you can specify the trigger definition. Specify the trigger steps to be executed when the trigger fires. The step is optional: you can do it later using a non-modal editor.



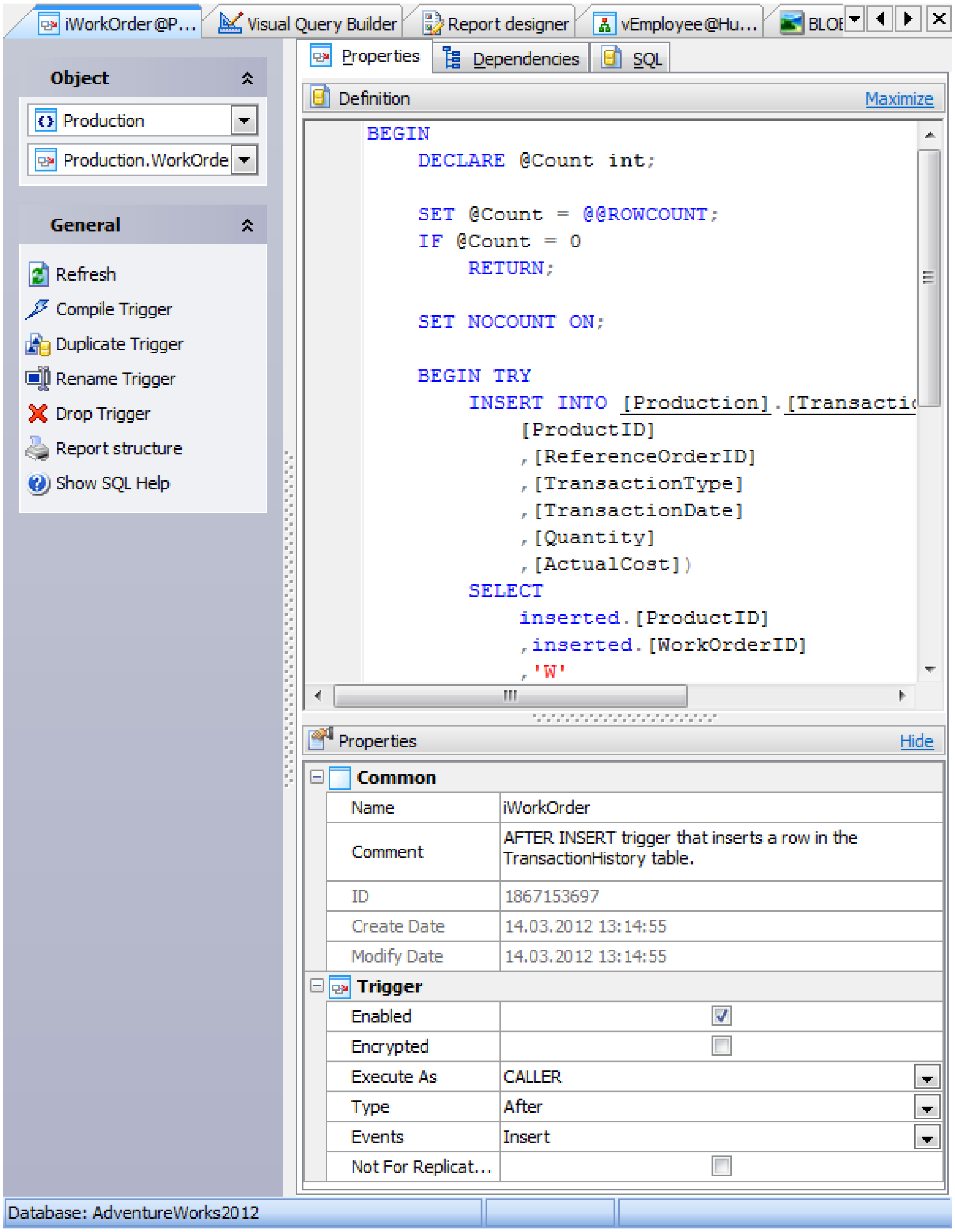
Create New Trigger

Specify trigger definition (you can also do it later using non-modal editor)

```
BEGIN
   DECLARE @Count int;

   SET @Count = @@ROWCOUNT;
   IF @Count = 0
       RETURN;

   SET NOCOUNT ON;

   BEGIN TRY
       INSERT INTO [Production].[TransactionHist
           [ProductID]
           ,[ReferenceOrderID]
           ,[TransactionType]
```

5.2.8.2 Trigger Editor

Trigger Editor can be opened automatically after the trigger is created and is available on editing the trigger.
The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.



```
BEGIN
    DECLARE @Count int;

    SET @Count = @@ROWCOUNT;
    IF @Count = 0
        RETURN;

    SET NOCOUNT ON;

    BEGIN TRY
        INSERT INTO [Production].[Transacti
            [ProductID]
            ,[ReferenceOrderID]
            ,[TransactionType]
            ,[TransactionDate]
            ,[Quantity]
            ,[ActualCost])
        SELECT
            inserted.[ProductID]
            ,inserted.[WorkOrderID]
            ,'W'
```

Common	
Name	iWorkOrder
Comment	AFTER INSERT trigger that inserts a row in the TransactionHistory table.
ID	1867153697
Create Date	14.03.2012 13:14:55
Modify Date	14.03.2012 13:14:55
Trigger	
Enabled	☑
Encrypted	☐
Execute As	CALLER
Type	After
Events	Insert
Not For Replicat...	☐

The main tab of the editor consists of several parts:, trigger definition, and trigger properties.

Definition
Defines the trigger conditions and actions.

Properties
Name
Here you can view and change the trigger name.

Note: the name of the object must be unique among all the object names in its container. Moreover, all the objects that are source of data need unique names among themselves. You can use any identifier that is allowed by Microsoft SQL server.

Comment
This field contains a comment to the trigger.

Create Date
Displays the date when the trigger was created.

Modify Date
Displays the date when the trigger was last modified.

Encrypted
If checked, it prevents the trigger from being published as part of SQL Server replication.

Execute As (With Exec)
Specifies the security context under which the trigger is to be executed.

Type
Specifies the trigger type.
After: the trigger is fired only after all operations specified in the triggering SQL statement have executed successfully.
Instead Of: the trigger is executed instead of the triggering SQL statement, therefore, overriding the actions of the triggering statements.

Events
One of *Insert, Update,* or *Delete;* this specifies the event that will fire the trigger.

☑ Enabled
If checked, the trigger is enabled.

☑ Not For Replication
If checked, it is indicated that the trigger should not be executed when a replication agent modifies the table that is involved in the trigger.
To apply the changes, select the Apply Changes item in the Navigation bar or use **Ctrl+F9** or **Ctrl+F7** shortcut keys.

It is also possible to modify object properties without opening the object editor: use the Object Properties item of the popup menu of the selected object from the explorer tree.

5.2.9 CLR Triggers

You can build CLR triggers using the SQL Server integration with the .NET Framework common language runtime (CLR).

Table CLR Triggers are managed within the CLR triggers tab of Table Editor.

How can I add a new CLR trigger?

CLR triggers are created within Create CLR Trigger Wizard 103. In order to run the wizard you should either

- open the table in Table Editor and the CLR Triggers tab there;
- press the **Insert** key or select the Add CLR New Trigger... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

or

- select the table in the explorer tree and use the Create New CLR Trigger... popup menu item

or

- select the table CLR Triggers node or any trigger within the table in the explorer tree and use the Add New CLR Trigger...

How can I edit an existing CLR trigger?

CLR triggers are edited within the CLR Trigger Editor dialog window. In order to open the dialog you should either

- open the table in Table Editor and the CLR Triggers tab there;
- press the **Enter** key or select the Edit CLR Trigger item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

or

- select the trigger to edit in the explorer tree and use the Edit CLR Trigger popup menu item.

How can I drop a CLR trigger?

To drop the CLR trigger:

- select the CLR trigger to drop in the explorer tree;
- select the Drop CLR Trigger item from the popup menu

or

- open the table in Table Editor and the CLR Triggers tab there;
- press the **Delete** key or select the Drop CLR Trigger item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.

5.2.9.1 Create CLR Trigger Wizard

Create CLR Trigger Wizard guides you through the process of creating a new CLR trigger. See How To Create CLR trigger 102 for instructions on running this wizard.

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

Specify CLR trigger options according to your needs. The detailed description is given below.

Comment

Supply a comment for the new CLR trigger.

Execute As
Specifies the security context under which the CLR trigger is to be executed.

Type
Defines the type of the CLR trigger. Possible values are *After* (specifies that the trigger is fired only when all operations specified in the triggering SQL statement have been executed successfully) and *Instead of* (specifies that the trigger is executed instead of the triggering SQL statement, therefore, overriding the actions of the triggering statements).

Events
Specifies the data modification statements that activate the trigger when it is tried against this table or view. At least one option must be specified. Any combination of these options in any order is allowed in the trigger definition.

☑ Enabled
If checked, it is denoted that a trigger action takes effect when fired.

☑ Not For Replication
When checked, it is indicated that the trigger should not be executed when a replication agent modifies the table that is involved in the trigger.

Assembly, Class, Method
Specifies the method of a .NET Framework assembly for a CLR stored function to reference. Class must be a valid SQL Server identifier and must exist as a class in the Assembly. The assembly is to be created in the database beforehand.

See also: Assemblies 214

5.2.9.2 CLR Trigger Editor

CLR Trigger Editor allows you to edit CLR Trigger definition (CLR trigger name, CLR trigger comment, etc.). It opens when you create a new CLR trigger or edit the existing one (see How to edit CLR trigger 102 for details).

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.

Specify CLR trigger options according to your needs. The detailed description is given below.

Name
You can edit the CLR trigger name here.

Note: the name of the object must be unique among all the object names in its container. Moreover, all the objects that are source of data need unique names among themselves. You can use any identifier that is allowed by Microsoft SQL server.

Comment
You can supply a comment for the CLR trigger.

Create Date
Displays the date when the CLR trigger was created.

Modify Date
Displays the date when the CLR trigger was last modified.

Execute As
Specifies the security context under which the CLR trigger is to be executed.

Type
Defines the type of the CLR trigger. Possible values are After (specifies that the trigger is only fired after all operations specified in the triggering SQL statement have executed successfully) and Instead of (specifies that the trigger is executed instead of the triggering SQL statement, therefore, overriding the actions of the triggering statements).

Events
Specifies the data modification statements that activate the trigger when it is tried against this table or view. At least one option must be specified. Any combination of these options in any order is allowed in the trigger definition.

☑ Enabled
If checked, it is denoted that a trigger action takes effect when fired.

☑ Not For Replication
When checked, it is indicated that the trigger should not be executed when a replication agent modifies the table that is involved in the trigger.

Assembly, Class, Method
Specifies the method of a .NET Framework assembly for a CLR stored function to reference. Class must be a valid SQL Server identifier and must exist as a class in the Assembly. The assembly is to be created in the database beforehand.

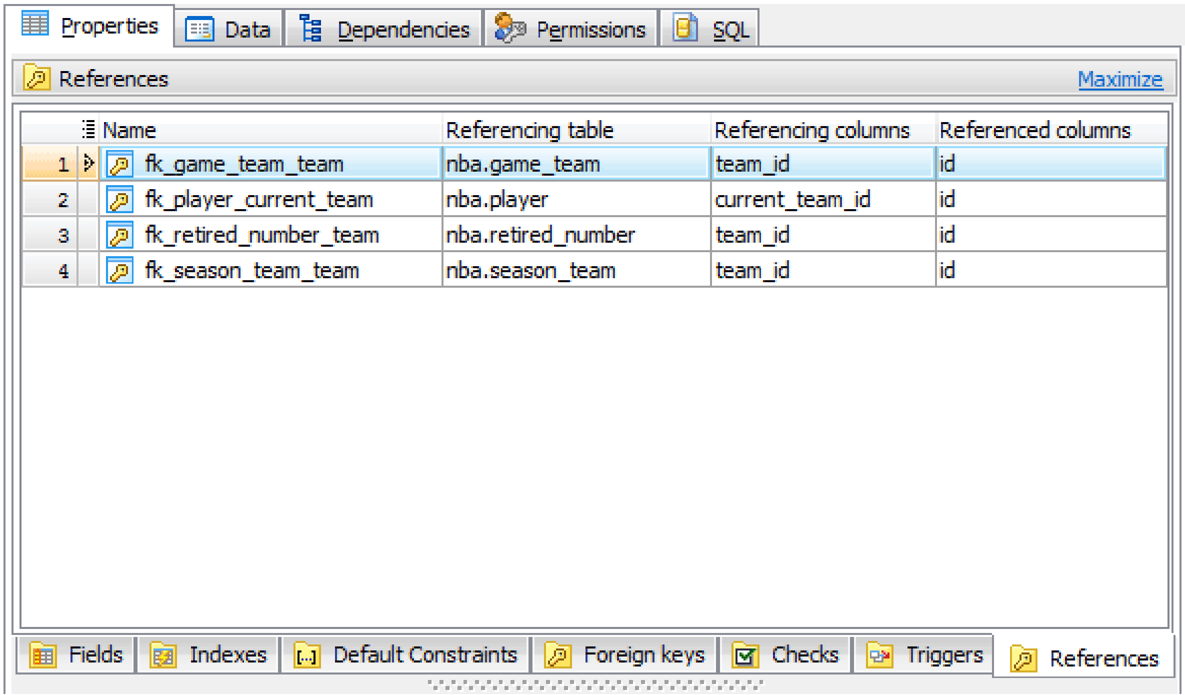
See also: Assemblies 214

To apply the changes, select the Apply Changes item in the Navigation bar or use **Ctrl+F9** or **Ctrl+F7** shortcut keys.

It is also possible to modify object properties without opening the object editor: use the Object Properties item of the popup menu of the selected object from the explorer tree.

5.2.10 Foreign Key References

A foreign key specifies that the values in a column (or a group of columns) must match the values appearing in some row of another table. The Foreign Key References tab allows you to manage foreign keys created in other tables and reference for columns of the current one. Table objects are managed within the corresponding tab of Table Editor 79. Unlike *tables* or *views,* Foreign Key References are actually not database objects. These are only references to foreign keys. They are designed specially for easy foreign keys management.



	Name	Referencing table	Referencing columns	Referenced columns
1	fk_game_team_team	nba.game_team	team_id	id
2	fk_player_current_team	nba.player	current_team_id	id
3	fk_retired_number_team	nba.retired_number	team_id	id
4	fk_season_team_team	nba.season_team	team_id	id

See also: Foreign Keys 90

How can I add a new foreign key reference?

Table foreign key references are edited within the Foreign Key Editor 90 dialog window. In order to open the dialog you should either

- open the table in Table Editor;
- open the Subitems item and the Foreign Keys References tab there;
- press **Enter** key or select the Edit Foreign Key item from the popup menu

or

- select the foreign key to edit in the appropriate table group of the explorer tree and use the Edit Foreign Key popup menu item.

How can I edit an existing foreign key reference?

Table foreign key references are edited within the Foreign Key Editor 90 dialog window. In order to open the dialog you should either

- open the table in Table Editor;
- open the Subitems item and the Foreign Keys References tab there;
- press **Enter** key or select the Edit Foreign Key item from the popup menu

or

- select the foreign key to edit in the appropriate table group of the explorer tree and use the Edit Foreign Key popup menu item.

⊟ How can I drop a foreign key reference?

To drop the foreign key reference:

- open the table in Table Editor;
- open the Subitems item and the Foreign Keys References tab there;
- press **Delete** key or select the Drop Foreign Key item from the popup menu;

or

- select the foreign key to drop in the appropriate table group of the explorer tree and use the Drop Foreign Key popup menu item.

and confirm dropping in the dialog window.

5.3 Views

Views are useful for allowing users to access a set of relations (tables) as if it were a single table, and limiting their access to just that. Views can also be used to restrict access to rows (a subset of a particular table).

How can I create a new view?

New views are created within Create View Wizard 109. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the View icon in the Create Database Object dialog

or

- select the Views list or any object from that list in the explorer tree;
- select the Create New View... item from the popup menu

or

- open Schema (Database) Editor and the Views tab there;
- press the **Insert** key or select the Create New View item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

To create a new view with the same properties as one of the existing views has:

- select the Object | Duplicate Database Object... main menu item.
- follow the instructions of Duplicate Object Wizard.

How can I edit an existing view definition?

Views can be edited within View Editor 114. In order to run the editor you should either

- select the view for editing in the explorer tree (type the first letters of the view name for quick search);
- select the Edit View... item from the popup menu

or

- open Schema (Database) Editor and the Views tab there;
- select the view to edit;
- press the **Enter** key or select the Edit View item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

You can change the name of the view using the Rename View dialog. To open the dialog you should either

- select the view to rename in the explorer tree;
- select the Rename View item from the popup menu

or

- open Schema (Database) Editor and the Views tab there;

- select the view to rename;
- select the Rename View item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

How can I drop a view?

To drop a view:

- select the view to drop in the explorer tree;
- select the Drop View item from the popup menu

or

- open Schema (Database) Editor and the Views tab there;
- select the view to drop;
- press the **Delete** key or select the Drop View item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

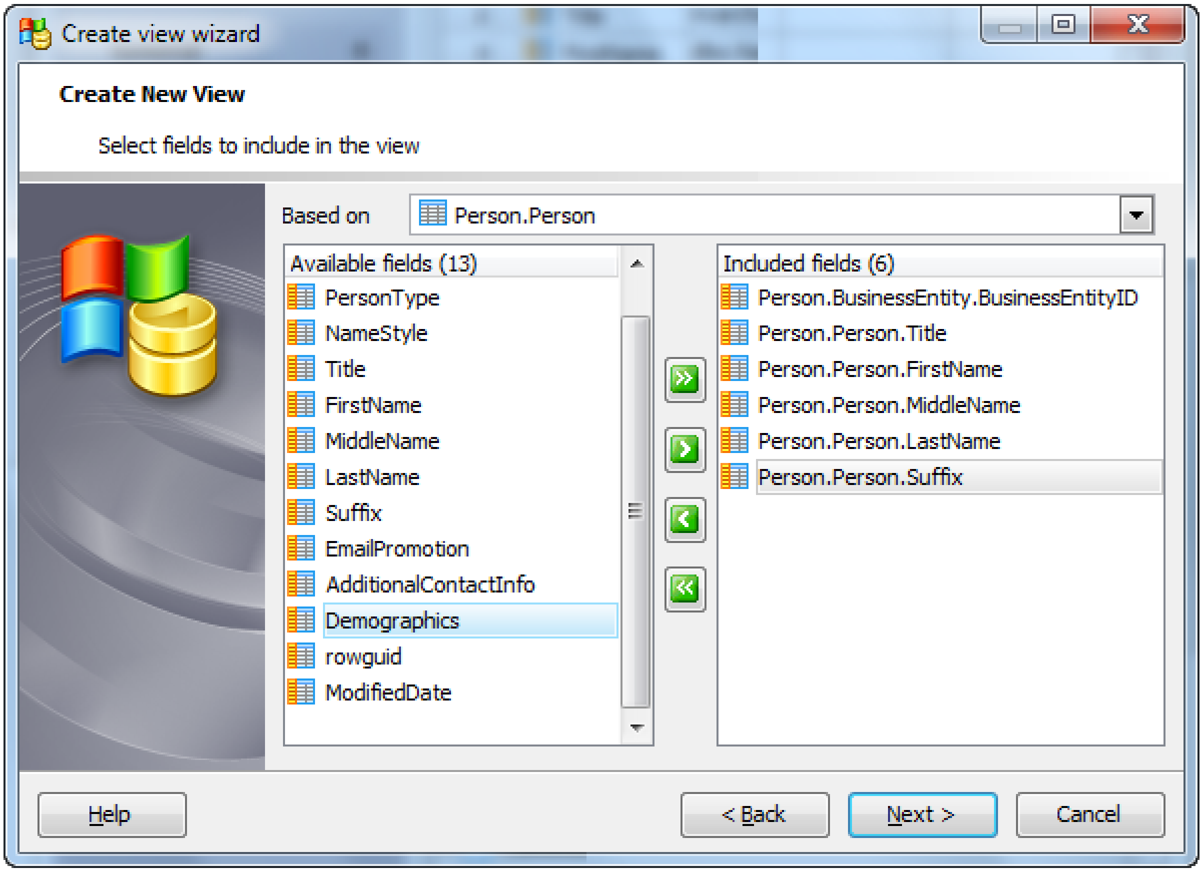
and confirm dropping in the dialog window.

5.3.1 Create View Wizard

Create View Wizard guides you through the process of creating a new view. See How To Create View 108 to learn how to run this wizard.

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

Selecting fields for a new view
Select a table or a view from the Based on drop-down menu. Then specify which fields will be used in the new view. Use Add All or Add Selected buttons to include field(s) into view definition. Use the Remove Selected or Remove All items to exclude field(s) from the view's field list. Click the Next button to proceed.



Specifying view options

Name

You may specify here the name of the view being created.

Owner

Defines the owner of the new view. By default, only the owner of an object can perform various operations with the object. In order to allow other users to operate it, privileges must be granted. (However, users that have the superuser attribute can always access any object.)

Comment

The box allows you to set optional text describing the view.

Schema Binding

Binds the view to the schema of the underlying table (or tables). If checked, the base table (or tables) cannot be modified in a way that would affect the view definition. All the referenced objects must be in the same database.

☑ With Check Option

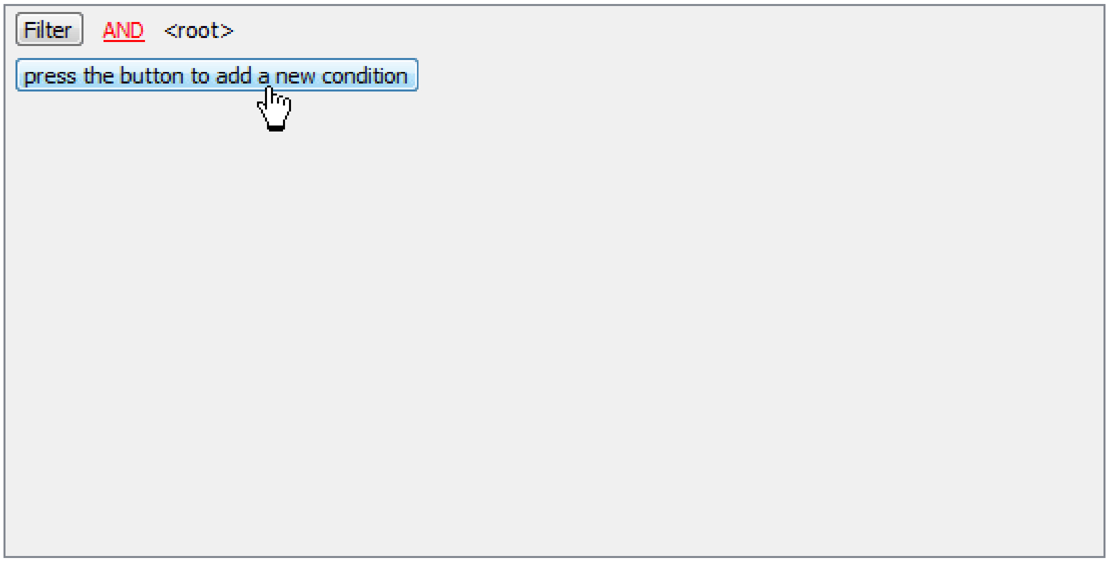
Forces all data modification statements executed against the view to follow the criteria set within the select statement. When a row is modified through a view, the With Check option ensures the data remaining visible through the view after the modification is committed. To check the option the owner of the view table must have been granted the INSERT, UPDATE, or DELETE privilege for the view table.

View Meta Data
Specifies that the instance of SQL Server will return to the DB-Library, ODBC, and OLE DB APIs the metadata information about the view, instead of the base table or tables when browse-mode metadata is being requested for a query that references the view. If checked, the browse-mode metadata returns the view name and not the base table names when it describes columns from the view in the result set.

Specifying the WHERE condition
MS SQL Maestro provides the Filter Builder dialog to facilitate a creating of the WHERE condition.

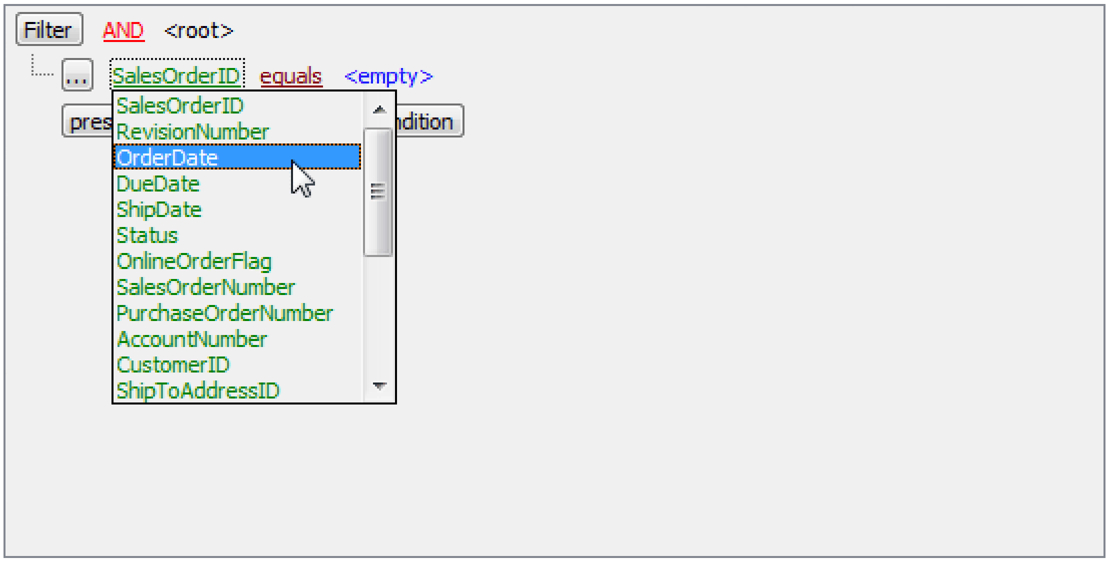
⊟ Adding a new condition to the filter

Suppose we need to select orders from the sample table *Orders* made between 01.02.2010 and 10.02.2010. These criteria are applied to the *OrderDate* column. Press the button to add this condition. Alternatively, you can use the Filter button and select the Add Condition option from the drop-down menu.



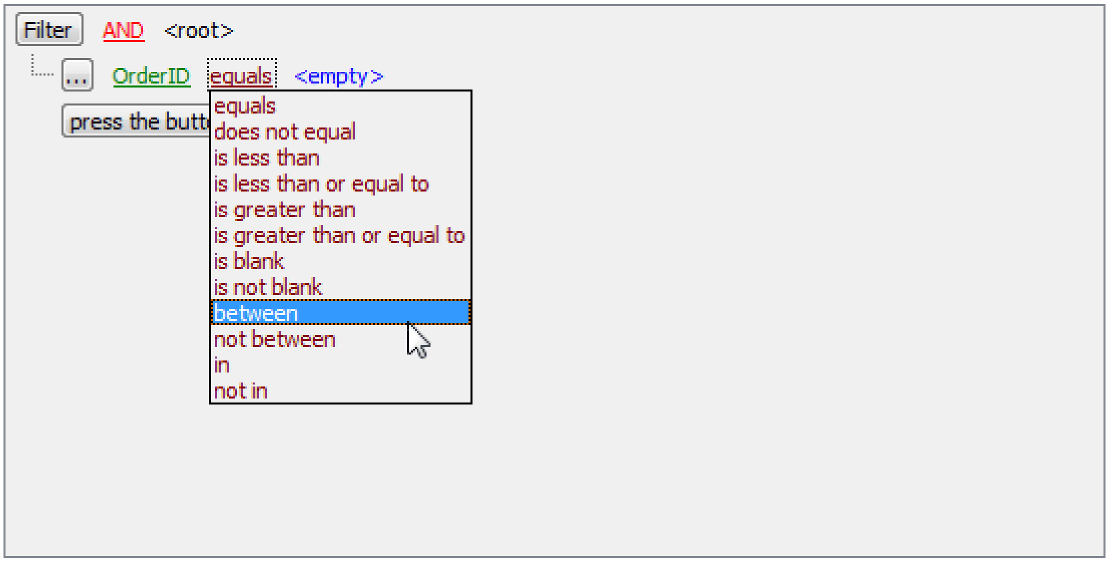
⊟ Setting a filter criteria in the condition

Select the *OrderDate* column in the drop-down list of the available columns.



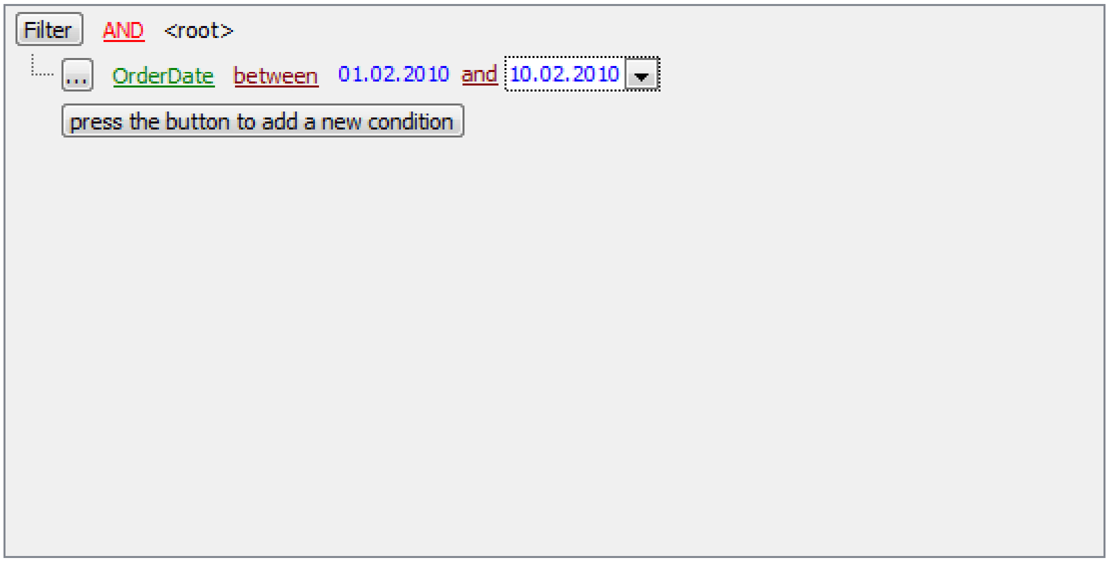
Setting an operator in the condition

Set the proper operator. In our example it is BETWEEN.



Setting criteria values in the condition

Next, you need to specify the range values for the selected operator. The editor used in value boxes is determined by the editor type assigned to the corresponding column.

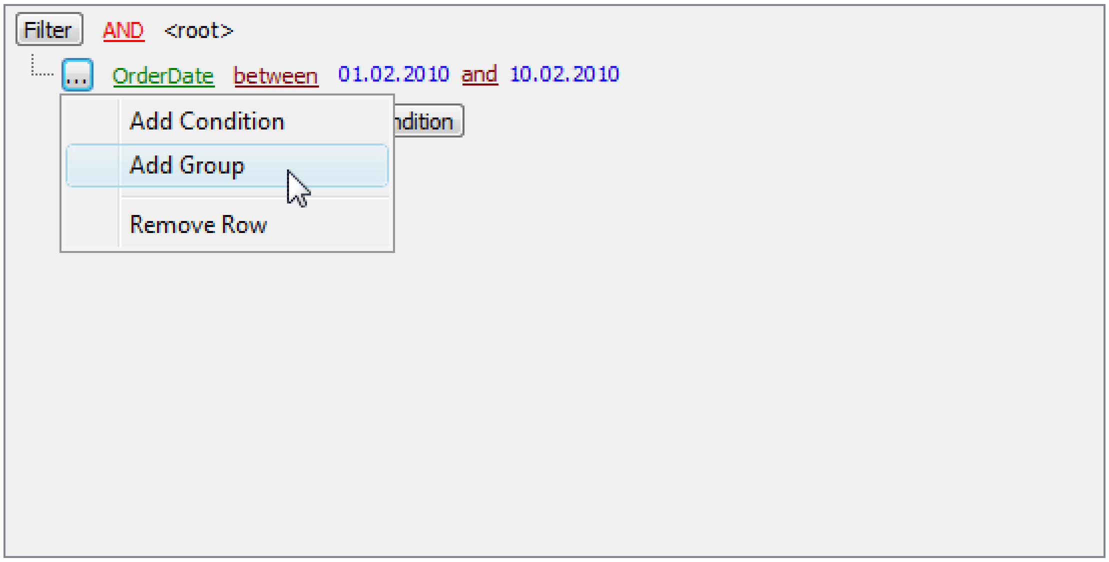


Now use the Apply button to see the filter result.

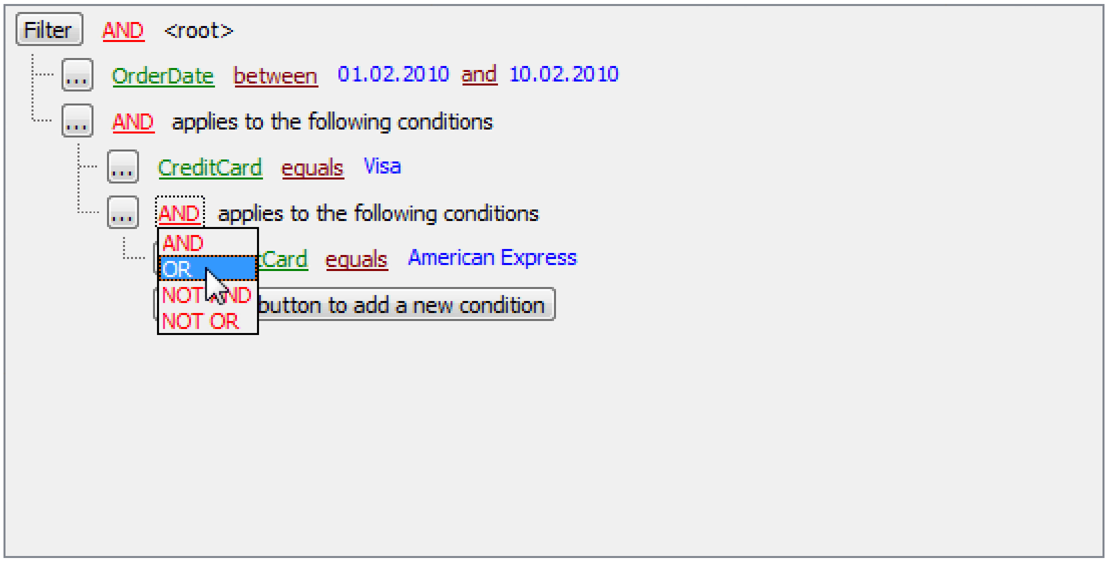
You can add additional conditions to the same root level to be combined by the AND operator.

Adding a new group

Suppose we need to select orders made between 01.02.2010 and 10.02.2010 and payed via 'Visa' or 'American Express'. This is a complex filter condition combining two simple conditions with the OR operator. Conditions from the same root level are combined by the AND operator. To add a condition combined with the previous one with the OR (NOT AND, NOT OR) operator, use a new group of conditions.



The next screen represents the finished filter conditions for this example.


Filter AND <root>
OrderDate between 01.02.2010 and 10.02.2010
AND applies to the following conditions
CreditCard equals Visa
AND applies to the following conditions
Card equals American Express
button to add a new condition
AND
OR
NOT AND
NOT OR

Adding view subitems

On this step of the wizard you can specify subobjects of the new view.
To add a new object:

- Choose a necessary page (Triggers - to manage view triggers, CLR Triggers - to manage view CLR triggers);
- Press Insert or use pop-up menu to open the appropriate Create Object Wizard (triggers 99, or CLR triggers 102);
- Specify new object properties.

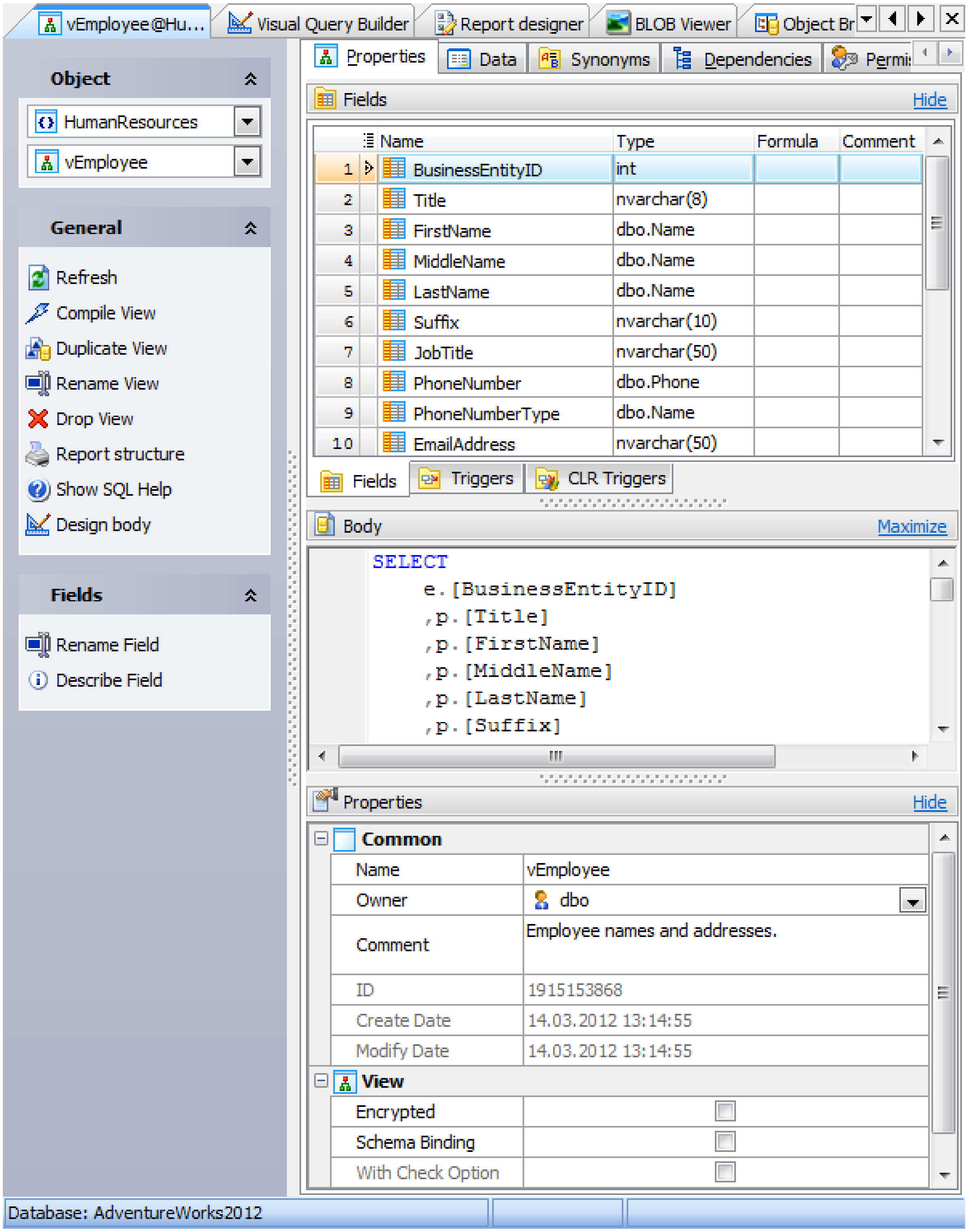
5.3.2 View Editor

View Editor allows you to edit the existing view definition (view name and the SELECT statement it implements).

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.

- Editing view properties 115
- Viewing data 117

See also: Create View Wizard 109



	Name	Type	Formula	Comment
1	BusinessEntityID	int		
2	Title	nvarchar(8)		
3	FirstName	dbo.Name		
4	MiddleName	dbo.Name		
5	LastName	dbo.Name		
6	Suffix	nvarchar(10)		
7	JobTitle	nvarchar(50)		
8	PhoneNumber	dbo.Phone		
9	PhoneNumberType	dbo.Name		
10	EmailAddress	nvarchar(50)		

```
SELECT
    e.[BusinessEntityID]
    ,p.[Title]
    ,p.[FirstName]
    ,p.[MiddleName]
    ,p.[LastName]
    ,p.[Suffix]
```

Common	
Name	vEmployee
Owner	dbo
Comment	Employee names and addresses.
ID	1915153868
Create Date	14.03.2012 13:14:55
Modify Date	14.03.2012 13:14:55
View	
Encrypted	☐
Schema Binding	☐
With Check Option	☐

5.3.2.1 Editing view properties

View Editor provides you with an ability to edit view properties. The Properties tab allows you to change the view name, view definition, the view owner and the comment for the view.

Subitems

Every tab is intended for managing some view subitems (e.g. *fields*, indexes, *triggers*, and *CLR triggers*). Each object can be opened in its editor. Use grid's popup menu to create new, edit or drop the selected view subitems. Using the popup menu you can also copy the selected objects to clipboard or paste previously copied objects.
You can operate on several objects at a time. For this you have to select view objects with the **Shift** or the **Ctrl** key pressed. After a group of objects is selected you can operate on it, e.g. *delete several objects at once*, as if it were a single object.

See also: Fields 83, Indexes 87 , Triggers 97, and CLR Triggers 102

Body

You can edit the view definition in this box.

Use the Name field to specify the view name.

Note: the name of the object must be unique among all the object names in its container. Moreover, all the objects that are source of data need unique names among themselves. You can use any identifier that is allowed by Microsoft SQL server.

Owner

Represents the view owner. By default, only the owner of an object can perform various operations with the object. In order to allow other users to operate it, privileges must be granted. (However, users that have the superuser attribute can always access any object.)

Comment

This field stores a comment to the view.

Create Date

Displays the date when the view was created.

Modify Date

Displays the date when the view was last modified.

Encrypted

If checked, the view is prevented from being published as part of SQL Server replication.

Schema Binding

Binds the view to the schema of the underlying table or tables. If checked, the base table or tables cannot be modified in a way that would affect the view definition. All referenced objects must be in the same database.

With Check Option

Forces all data modification statements executed against the view to follow the criteria set within select statement. When a row is modified through a view, the With Check option ensures that the data remains visible through the view after the modification is committed.

View Meta Data

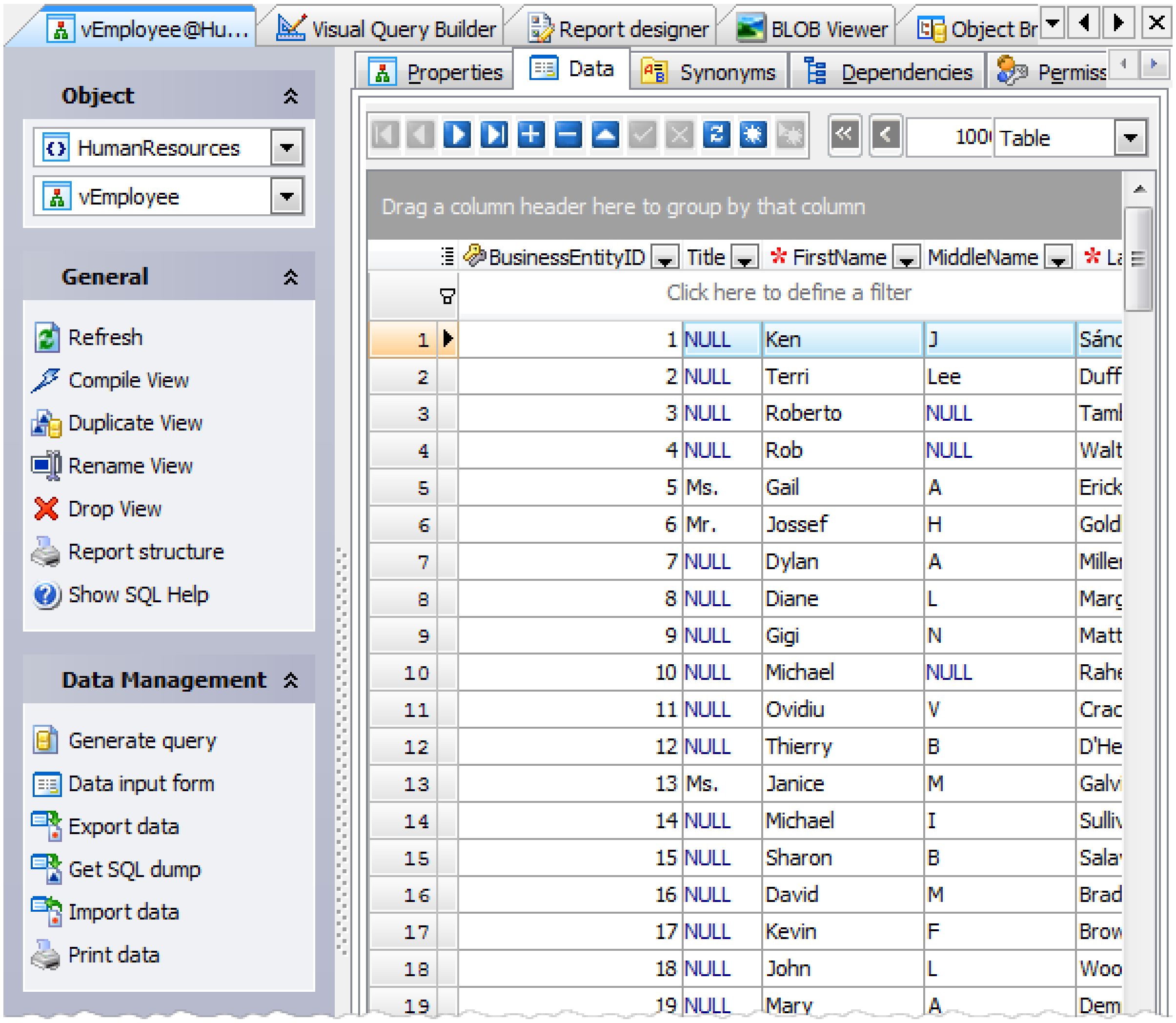
Specifies that the instance of SQL Server will return to the DB-Library, ODBC, and OLE DB APIs the metadata information about the view, instead of the base table or tables when browse-mode metadata is being requested for a query that references the view. If

checked, the browse-mode metadata returns the view name and not the base table names when it describes columns from the view in the result set.
To apply the changes, select the Apply Changes item in the Navigation bar or use **Ctrl+F9** or **Ctrl+F7** shortcut keys.

It is also possible to modify object properties without opening the object editor: use the Object Properties item of the popup menu of the selected object from the explorer tree.

5.3.2.2 Viewing data

The Data tab displays the data represented in the view as a grid (see Data View for details). The popup menu of this tab and the Data Management navigation bar allow you to export data, get SQL dump, set the value of the selected record to *Null* or to *Now* (for Date values). In tables with BLOB fields you can also call BLOB Editor to view and edit the BLOB fields.



	BusinessEntityID	Title	* FirstName	MiddleName	* La
1	1	NULL	Ken	J	Sánc
2	2	NULL	Terri	Lee	Duff
3	3	NULL	Roberto	NULL	Taml
4	4	NULL	Rob	NULL	Walt
5	5	Ms.	Gail	A	Erick
6	6	Mr.	Jossef	H	Gold
7	7	NULL	Dylan	A	Mille
8	8	NULL	Diane	L	Marg
9	9	NULL	Gigi	N	Matt
10	10	NULL	Michael	NULL	Rahe
11	11	NULL	Ovidiu	V	Crac
12	12	NULL	Thierry	B	D'He
13	13	Ms.	Janice	M	Galv
14	14	NULL	Michael	I	Sulli
15	15	NULL	Sharon	B	Sala
16	16	NULL	David	M	Brad
17	17	NULL	Kevin	F	Brow
18	18	NULL	John	L	Woo
19	19	NULL	Mary	A	Dem

5.4 Procedures

A stored procedure is a saved collection of Transact-SQL statements or a reference to a Microsoft .NET Framework common language runtime (CLR) method that can take and return user-supplied parameters. Procedures can be created for permanent use or for temporary use within a session, local temporary procedure, or for temporary use within all sessions, global temporary procedure.

How can I create a new procedure?

New procedures are created within Create Procedure Wizard 119. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the Procedure icon in the Create Database Object dialog

or

- select the Procedures list or any object from that list in the explorer tree;
- select the Create New Procedure... item from the popup menu

or

- open Schema Editor and the Procedures tab there;
- press the **Insert** key or select the Create New Procedure item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

To create a new procedure with the same properties as one of the existing procedures has:

- select the Object | Duplicate Database Object... main menu item;
- follow the instructions of Duplicate Object Wizard.

How can I edit an existing procedure definition?

Procedures can be edited within Procedure Editor 123. In order to run the editor you should either

- select the procedure for editing in the explorer tree (type the first letters of the procedure name for quick search);
- select the Edit Procedure... item from the popup menu

or

- open Schema Editor and the Procedures tab there;
- select the procedure to edit;
- press the **Enter** key or select the Edit Procedure item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

You can change the name of the procedure using the Rename Procedure dialog. To open the dialog you should either

- select the procedure to rename in the explorer tree;
- select the Rename Procedure item from the popup menu

or

- open Schema Editor and the Procedures tab there;
- select the procedure to rename;
- select the Rename Procedure item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

⊟ How can I execute a procedure?

To execute a procedure:

- select the procedure in the explorer tree (type the first letters of the procedure name for quick search);
- select the Edit Procedure... item from the popup menu;
- execute the procedure using the Execute link of the Navigation Bar

or

open Schema Editor and the Procedures tab there;

- select the procedure to execute;
- press the **Enter** key or select the Edit Procedure item from the popup menu, or use the corresponding link of the Navigation Bar;
- execute the procedure using the Execute link of the Navigation bar.

⊟ How can I drop a procedure?

To drop a procedure:

- select the procedure to drop in the explorer tree;
- select the Drop Procedure item from the popup menu

or

- open Schema Editor and the Procedures tab there;
- select the procedure to drop;
- press the **Delete** key or select the Drop Procedure item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.

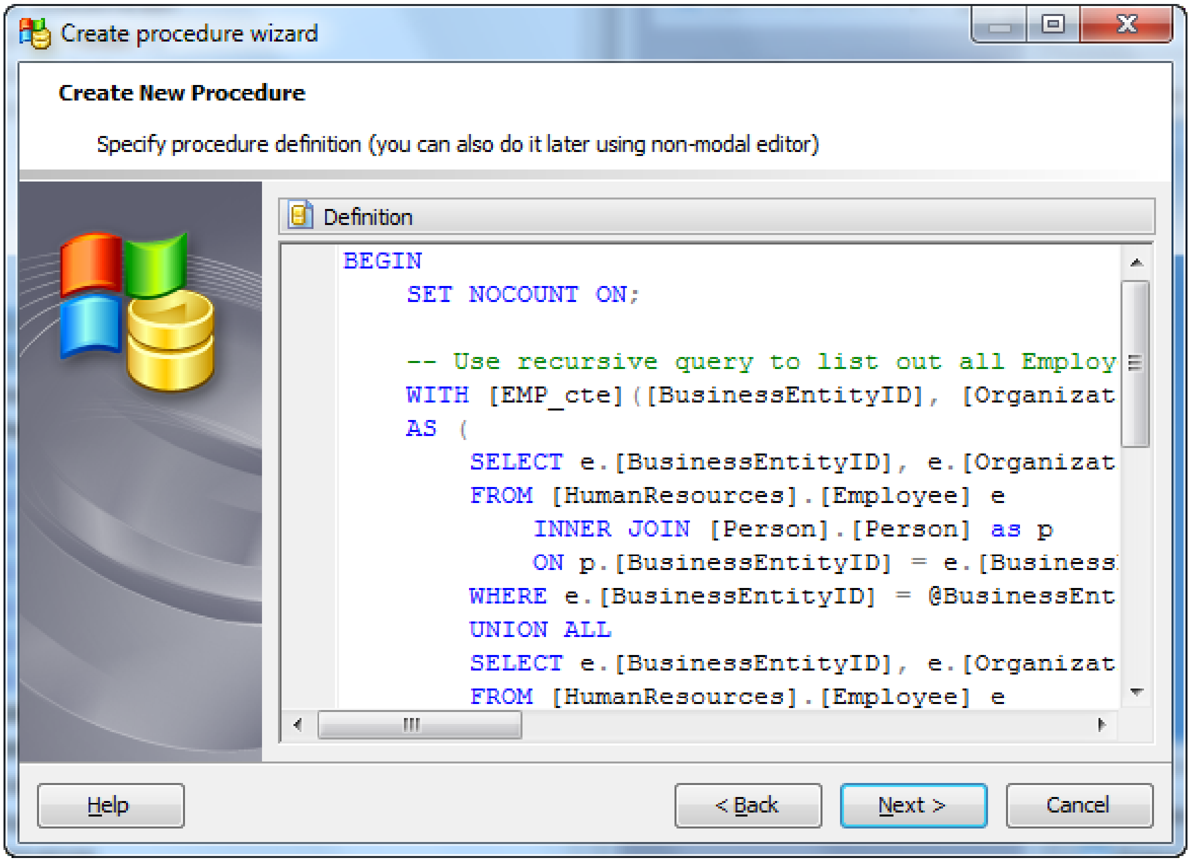
5.4.1 Create Procedure Wizard

Create Procedure Wizard guides you through the process of creating a new procedure. See How To Create Procedure 118 for instructions on running this wizard.

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

- Specifying procedure options 120
- Managing parameters of a new procedure 122
- Specifying procedure definition 122

See also: Procedure Editor 123

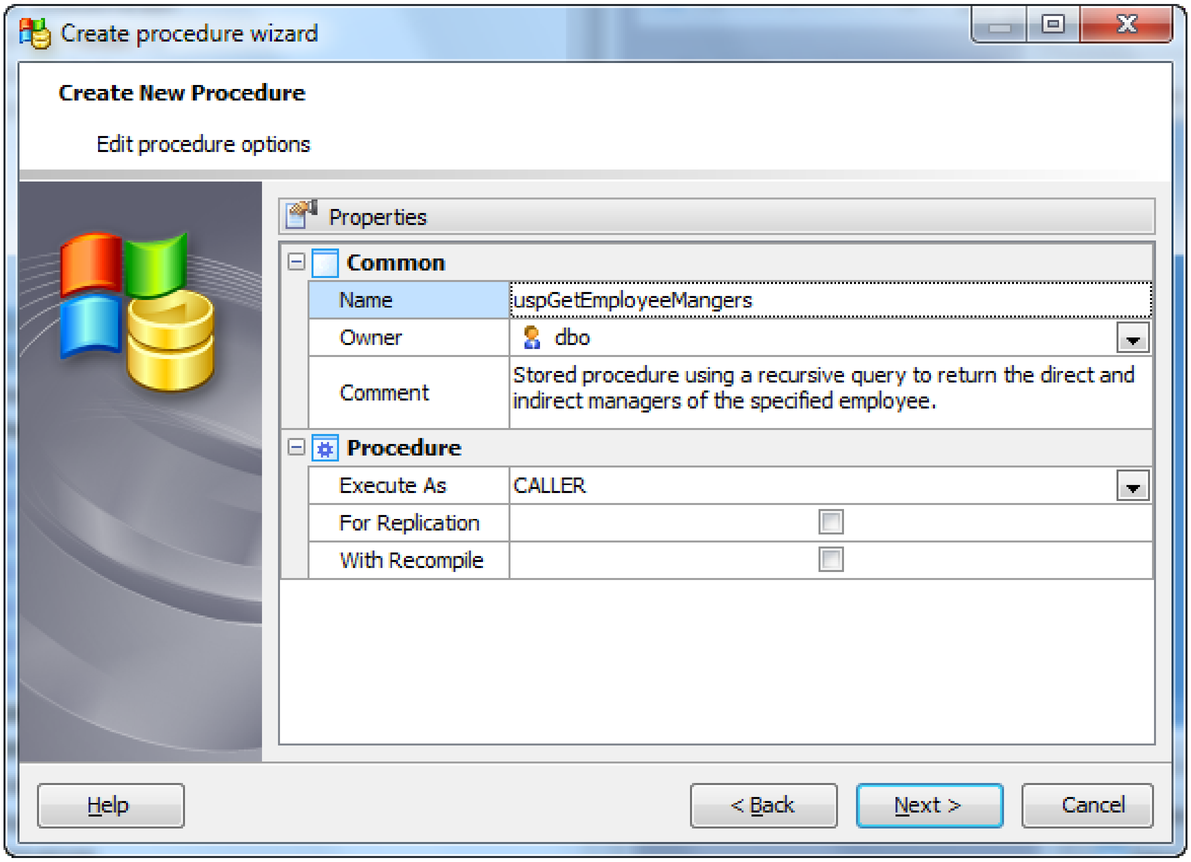


Create procedure wizard
Create New Procedure
Specify procedure definition (you can also do it later using non-modal editor)
Definition
Help
< Back
Next >
Cancel

```
BEGIN
    SET NOCOUNT ON;

    -- Use recursive query to list out all Employ
    WITH [EMP_cte]([BusinessEntityID], [Organizat
    AS (
        SELECT e.[BusinessEntityID], e.[Organizat
        FROM [HumanResources].[Employee] e
            INNER JOIN [Person].[Person] as p
            ON p.[BusinessEntityID] = e.[Business
        WHERE e.[BusinessEntityID] = @BusinessEnt
        UNION ALL
        SELECT e.[BusinessEntityID], e.[Organizat
        FROM [HumanResources].[Employee] e
```

5.4.1.1 Specifying procedure options

Specify procedure options according to your needs. The detailed description is given below.



Common

Name	uspGetEmployeeMangers
Owner	dbo
Comment	Stored procedure using a recursive query to return the direct and indirect managers of the specified employee.

Procedure

Execute As	CALLER
For Replication	☐
With Recompile	☐

Name
The new procedure name as it was set on the previous step.

Owner
Defines the owner for the procedure. By default, only the owner of an object can perform various operations with the object. In order to allow other users to operate it, privileges must be granted. (However, users that have the superuser attribute can always access any object.)

Comment
Specify a comment for the procedure.

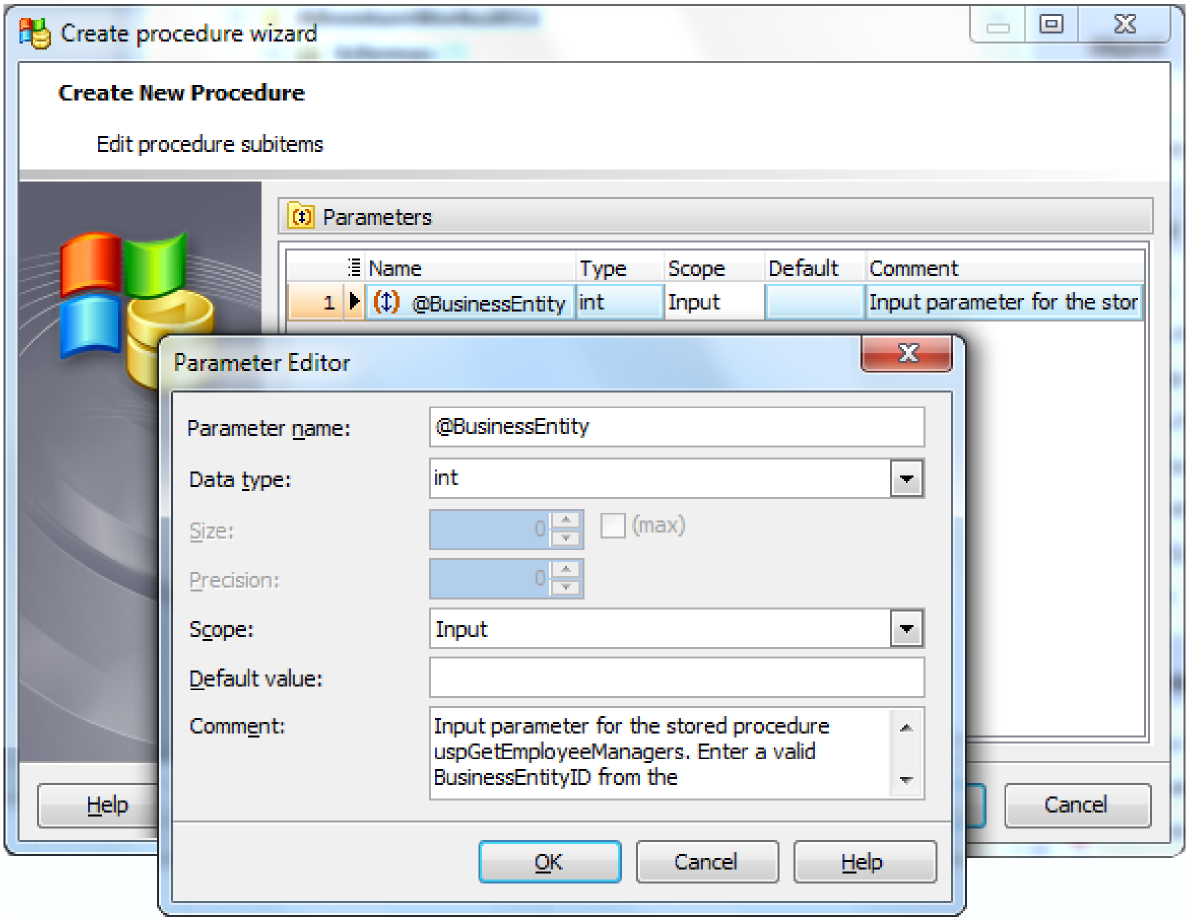
Execute As
Specifies the security context under which to execute the stored procedure (For details see Execute As Clause (Microsoft SQL 2005 References)).

☑ Check the For Replication option to indicate that the procedure is executed only during replication.

☑ **With Recompile**
Indicates that the Database Engine does not cache a plan for this procedure and the procedure is compiled at run time.

5.4.1.2 Managing parameters

Use the pop-up menu or press **Insert** to add a new parameter and set its properties in Parameter Editor 53. Press **Enter** or use the appropriate pop-up menu item to edit the selected parameter, or the Delete to delete one.

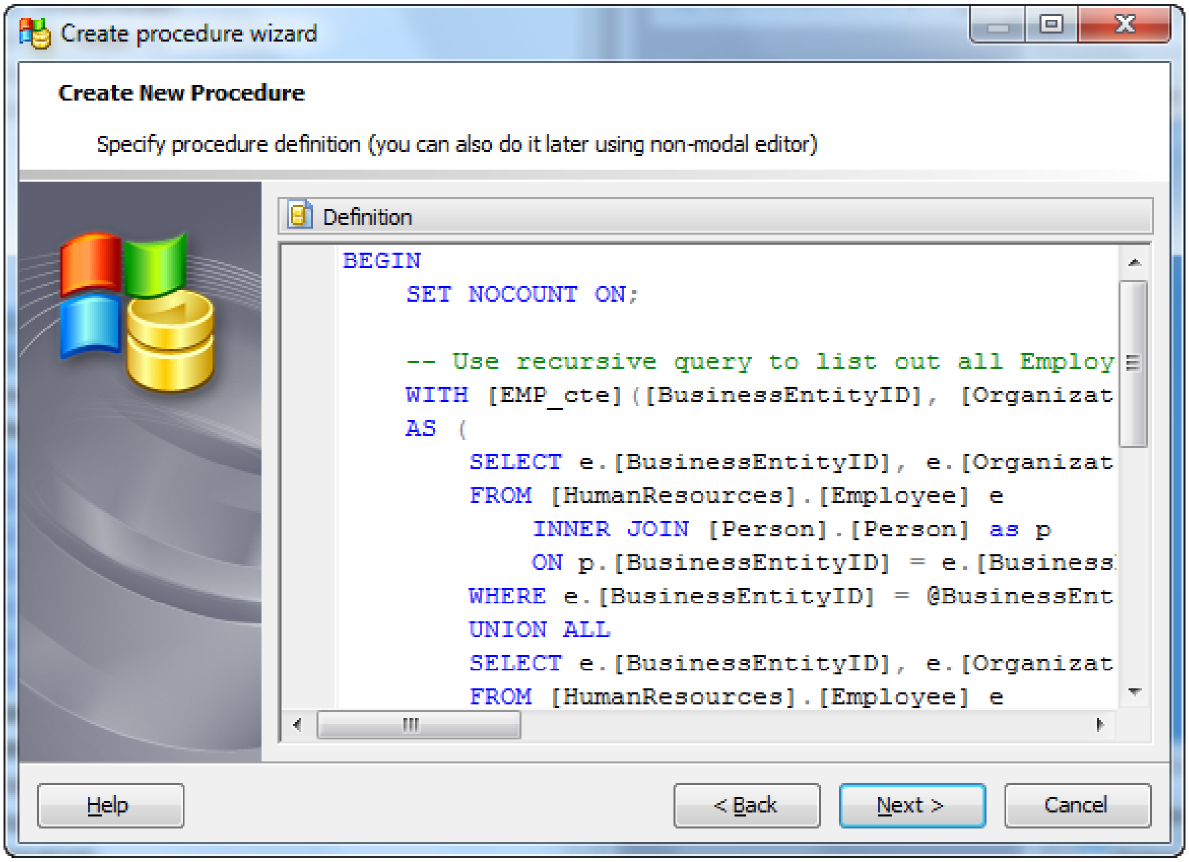


	Name	Type	Scope	Default	Comment
1	@BusinessEntity	int	Input		Input parameter for the stor

Parameter name: @BusinessEntity
Data type: int
Size: 0 ☐ (max)
Precision: 0
Scope: Input
Default value:
Comment: Input parameter for the stored procedure uspGetEmployeeManagers. Enter a valid BusinessEntityID from the

5.4.1.3 Specifying procedure definition

At this step you can specify the SQL definition for the new procedure. The step is optional: you can do it later using a non-modal editor.



Specify procedure definition (you can also do it later using non-modal editor)

```
BEGIN
    SET NOCOUNT ON;

    -- Use recursive query to list out all Employ
    WITH [EMP_cte]([BusinessEntityID], [Organizat
    AS (
        SELECT e.[BusinessEntityID], e.[Organizat
        FROM [HumanResources].[Employee] e
            INNER JOIN [Person].[Person] as p
            ON p.[BusinessEntityID] = e.[Business
        WHERE e.[BusinessEntityID] = @BusinessEnt
        UNION ALL
        SELECT e.[BusinessEntityID], e.[Organizat
        FROM [HumanResources].[Employee] e
```

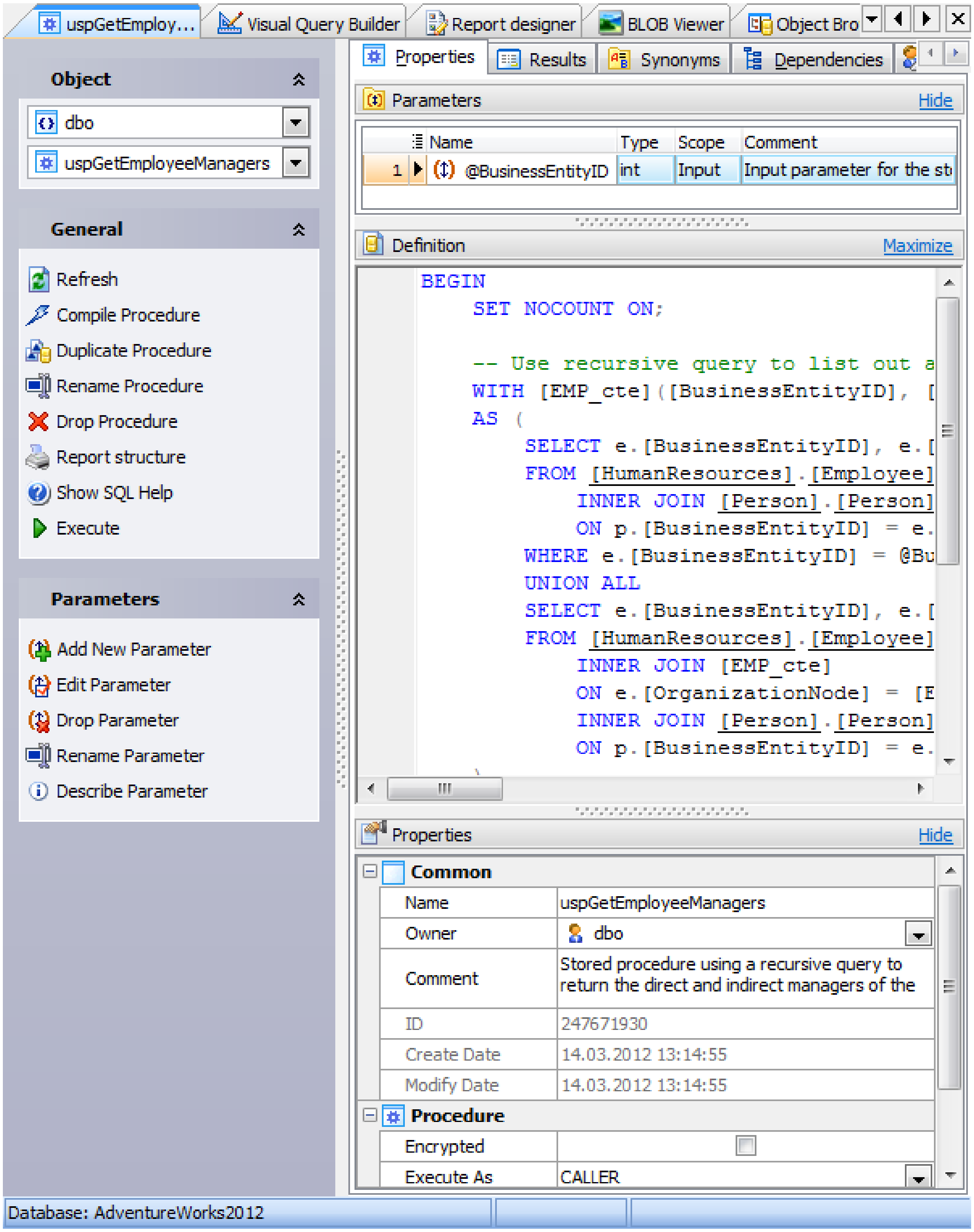
5.4.2 Procedure Editor

Procedure Editor allows you to execute the existing procedures or edit their definition (*procedure name, parameter list, procedure body,* etc.). It opens when you create a new procedure or edit the existing one (see How to edit procedure 118 for details).

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.

- Editing procedure properties 124
- Viewing procedure results 126

See also: Create Procedure Wizard 119



dbo

uspGetEmployeeManagers

	Name	Type	Scope	Comment
1	@BusinessEntityID	int	Input	Input parameter for the st

```
BEGIN
    SET NOCOUNT ON;

    -- Use recursive query to list out a
    WITH [EMP_cte]([BusinessEntityID], [
    AS (
        SELECT e.[BusinessEntityID], e.[
        FROM [HumanResources].[Employee]
            INNER JOIN [Person].[Person]
            ON p.[BusinessEntityID] = e.
        WHERE e.[BusinessEntityID] = @Bu
        UNION ALL
        SELECT e.[BusinessEntityID], e.[
        FROM [HumanResources].[Employee]
            INNER JOIN [EMP_cte]
            ON e.[OrganizationNode] = [E
            INNER JOIN [Person].[Person]
            ON p.[BusinessEntityID] = e.
```

Common	
Name	uspGetEmployeeManagers
Owner	dbo
Comment	Stored procedure using a recursive query to return the direct and indirect managers of the
ID	247671930
Create Date	14.03.2012 13:14:55
Modify Date	14.03.2012 13:14:55
Procedure	
Encrypted	☐
Execute As	CALLER

Database: AdventureWorks2012

5.4.2.1 Editing properties

The Body contains the definition of the procedure. The Parameters tab contains the list of the current procedure parameters with its options. Here you can view the Name and the Type of each parameter of the procedure and supply a Comment for the parameter. Use grid's popup menu to add, edit parameters in Parameter Editor. It also allows you to drop and rename procedure parameters.

Parameters can be edited within the Parameter Editor dialog window. In order to open the dialog you should

- open the object in its editor and the Parameters tab there;
- select the parameter to edit;
- press the **Enter** key or select the Edit Parameter item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

Name

Defines the procedure name.

Note: the name of the object must be unique among all the object names in its container. Moreover, all the objects that are source of data need unique names among themselves. You can use any identifier that is allowed by Microsoft SQL server.

Owner

There is the owner for the procedure. By default, only the owner of an object can perform various operations with the object. In order to allow other users to operate it, privileges must be granted. (However, users that have the superuser attribute can always access any object.)

Comment

This field contains a comment to the procedure.

Create Date

Indicates the date when the procedure was created.

Modify Date

Indicates the date when the procedure was last modified.

Check the Encrypted option to indicate that SQL Server will convert the original text of the CREATE PROCEDURE statement to an obfuscated format. The output of the obfuscation is not directly visible.

Execute As

Specifies the security context under which to execute the stored procedure (For details see Execute As Clause (Microsoft SQL 2005 References)).

Check the For Replication to indicate that the procedure is executed only during replication.

Parameters

Here you can find the list of the procedure parameters.

With Recompile

Indicates that the Database Engine does not cache a plan for this procedure and the procedure is compiled at run time.

Number

Is an optional integer that is used to group procedures of the same name.

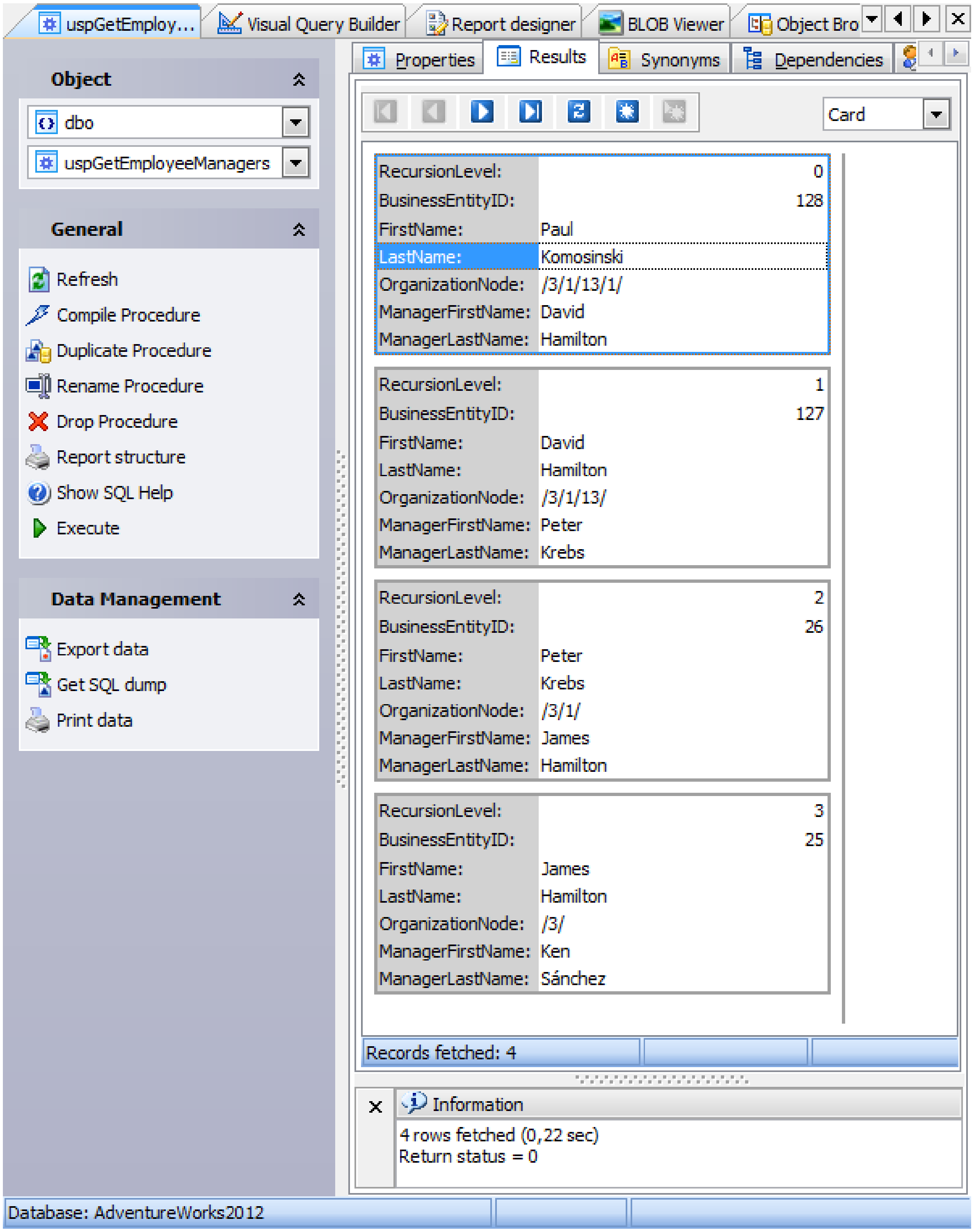
To apply the changes, select the Apply Changes item in the Navigation bar or use **Ctrl+F9** or **Ctrl+F7** shortcut keys.

It is also possible to modify object properties without opening the object editor: use the Object Properties item of the popup menu of the selected object from the explorer tree.

5.4.2.2 Viewing procedure results

The Results tab represents the result of the successfully executed procedure, if it returns a data that is represented as a grid (see Data View for details). Use grid's popup menu to export data, get SQL dump. In result with BLOB fields you can also call the BLOB Editor to view the BLOB fields. The data in result set is always read-only.

See also: Executing procedure 53



RecursionLevel: 0
BusinessEntityID: 128
FirstName: Paul
LastName: Komosinski
OrganizationNode: /3/1/13/1/
ManagerFirstName: David
ManagerLastName: Hamilton

RecursionLevel: 1
BusinessEntityID: 127
FirstName: David
LastName: Hamilton
OrganizationNode: /3/1/13/
ManagerFirstName: Peter
ManagerLastName: Krebs

RecursionLevel: 2
BusinessEntityID: 26
FirstName: Peter
LastName: Krebs
OrganizationNode: /3/1/
ManagerFirstName: James
ManagerLastName: Hamilton

RecursionLevel: 3
BusinessEntityID: 25
FirstName: James
LastName: Hamilton
OrganizationNode: /3/
ManagerFirstName: Ken
ManagerLastName: Sánchez

Records fetched: 4

```
4 rows fetched (0,22 sec)
Return status = 0
```

Database: AdventureWorks2012

5.5 UDFs

A user-defined function (UDF) is stored as a database object providing reusable code that can be used in Transact-SQL (T-SQL) statements such as SELECT, in applications calling the function, in the definition of another user-defined function, to parameterize a view or improve the functionality of an indexed view, to define a column in a table, to define a CHECK constraint for a column or to replace a stored procedure.

User-defined functions have been expanded in Microsoft SQL Server 2005 to include functions written in any .NET programming language. SQL Anywhere allows to define user-specific database functions. In an SQL statement, you can then use these user-defined database functions in the same way as any other predefined functions.

⊟ How can I create a new UDF?

New UDFs are created within Create UDF Wizard 129. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the UDF icon in the Create Database Object dialog

or

- select the UDFs list or any object from that list in the explorer tree;
- select the Create New UDF... item from the popup menu

or

- open Database Editor, or (for Microsoft server 2005) Schema Editor and the UDFs tab there;
- press the **Insert** key or select the Create New UDF item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

To create a new UDF with the same properties as one of the existing UDFs has:

- select the Object | Duplicate Database Object... main menu item;
- follow the instructions of Duplicate Object Wizard.

⊟ How can I edit an existing UDF definition?

UDFs can be edited within UDFEditor 132. In order to open the editor you should either

- select the UDF for editing in the explorer tree (type the first letters of the UDF name for quick search);
- select the Edit UDF item from the popup menu

or

- open Schema Editor and the UDFs tab there;
- select the UDF to edit;
- press the **Enter** key or select the Edit UDF item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

You can change the name of the UDF using the Rename UDF dialog:

- select the UDF to rename in the explorer tree;
- select the Rename UDF item from the popup menu.

How can I execute a UDF?

To execute the UDF:

- select the UDF in the explorer tree (type the first letters of the UDF name for quick search);
- select the Edit UDF... item from the popup menu;
- execute the UDF using the Execute link of the Navigation Bar

or

- open Schema Editor and the UDFs tab there;
- select the UDF to execute;
- press the **Enter** key or select the Edit UDF item from the popup menu, or use the corresponding link of the Navigation Bar;
- execute the UDF using the Execute link of the Navigation bar.

How can I drop a UDF?

To drop a UDF:

- select the UDF to drop in the explorer tree;
- select the Drop UDF item from the popup menu

or

- open Schema Editor and the UDFs tab there;
- select the UDF to drop;
- press the **Delete** key or select the Drop UDF item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

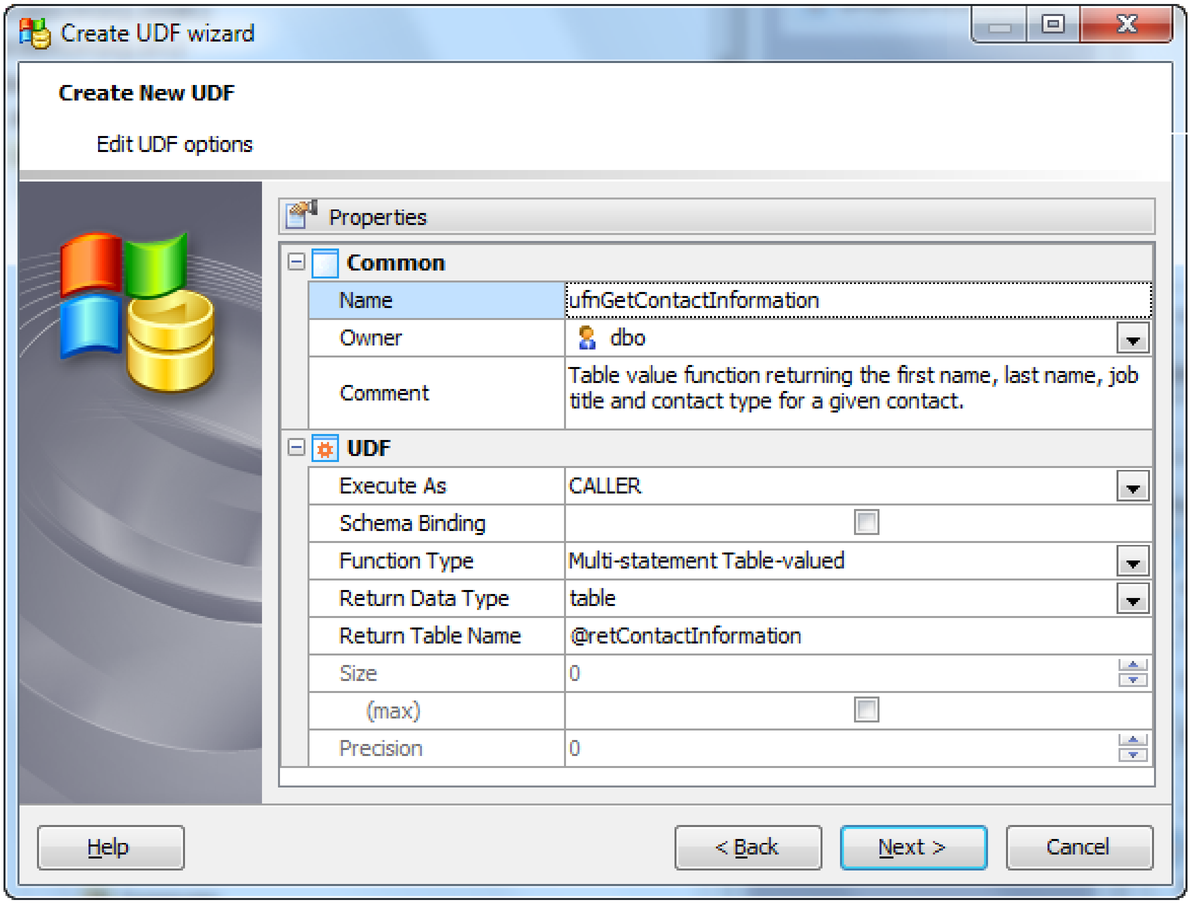
and confirm dropping in the dialog window.

5.5.1 Create UDF Wizard

Create UDF Wizard guides you through the process of creating a new UDF. See How To create UDF 128 for instructions on running this wizard.

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

Specify the properties for the new UDF according to your needs. The detailed description is given below.



Common	
Name	ufnGetContactInformation
Owner	dbo
Comment	Table value function returning the first name, last name, job title and contact type for a given contact.
UDF	
Execute As	CALLER
Schema Binding	☐
Function Type	Multi-statement Table-valued
Return Data Type	table
Return Table Name	@retContactInformation
Size	0
(max)	☐
Precision	0

Specifying UDF properties

Name

Specify a name for the function.

Owner

Select the owner of the UDF from the drop-down list. By default, only the owner of an object can perform various operations with the object. In order to allow other users to operate it, privileges must be granted. (However, users that have the superuser attribute can always access any object.)

Comment

Supply a comment to the UDF if necessary.
Defines the data type of the function result.

Language

The field stores the name of the language the function is implemented in. Select either of the available languages: *SQL*, *C*, *internal*, or the name of a u ser-defined procedural language. For backward compatibility, the name may be enclosed by single quotes.

Check the Strict option to indicate that the function always returns NULL whenever any of its arguments are null. If this option is specified, the function is not executed when there are null arguments; a null result is assumed automatically instead. Uncheck the Strict option to indicate that the function will be called normally when some of its arguments are null. It is then the function author's responsibility to check for null values if necessary and to respond in the appropriate way.

Execution Privileges

Select *Invoker* to indicate that the function is to be executed with the privileges of the user that calls it (the default value).
Select *Definer* to specify that the function is to be executed with the privileges of the user that created it.

Stability

Set the attribute to inform the system whether it is safe to replace multiple evaluations of the function with a single evaluation, for run-time optimization. If none of these appear, *Volatile* is the default assumption.
Immutable indicates that the function always returns the same result when given the same argument values; that is, it does not do database lookups or otherwise use information not directly present in its argument list. If this option is given, any call of the function with all-constant arguments can be immediately replaced with the function value.
Stable indicates that within a single table scan the function will consistently return the same result for the same argument values, but that its result could change across SQL statements. This is the appropriate selection for the functions in which the results depend on database lookups, parameter variables (such as the current time zone), etc. Also note that the *current_timestamp* family of functions qualify as stable, since their values do not change within a transaction.
Volatile indicates that the function value can change even within a single table scan, so no optimizations can be made. Relatively few database functions are volatile in this sense; some examples are *random()*, *currval()*, *timeofday()*. Note that any function that has side-effects must be classified volatile (even if its result is quite predictable) to prevent calls from being optimized away; an example is *setval()*.

Note: You can also add the function definition within the Properties 132 tab of Function Editor 132.

Execute As

Specify the security context under which the UDF is to be executed (For details see Execute As Clause (Microsoft SQL 2005 References)).

Schema Binding

Check the option to indicate that the UDF is bound to the database objects it references.

Function Type

Define *Scalar*, *Inline Table-valued* or *Multi-statement Table-valued* function type for the UDF.

Return Data Type

The return value of a scalar user-defined function is to be specified here. For Transact-SQL functions, all data types (including CLR user-defined types) are allowed except for the *timestamp* data type. For CLR UDFs, all data types (including CLR user-defined types) are allowed except for the *text*, *ntext*, *image* and *timestamp* data types. The nonscalar *cursor* and *table* types cannot be specified as a return data type in Transact-SQL or CLR UDFs.

Managing parameters of a new UDF

Use popup menu Add New Parameter... item to add a new parameter and set its properties in Parameter Editor 53. Use the Edit and Delete items to manage UDF parameters.

Specifying Function Definition

At this step you can specify the SQL definition for the new function. The step is optional: you can do it later using a non-modal editor.

5.5.2 UDF Editor

UDF Editor allows you to execute the existing UDF, and edit its definition (UDF name, parameter list, etc.). In order to open the editor you should either

- select the UDF for editing in the explorer tree (type the first letters of the UDF name for quick search);
- select the Edit UDF item from the popup menu

or

- open Schema Editor and the UDFs tab there;
- select the UDF to edit;
- press the **Enter** key or select the Edit UDF item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.

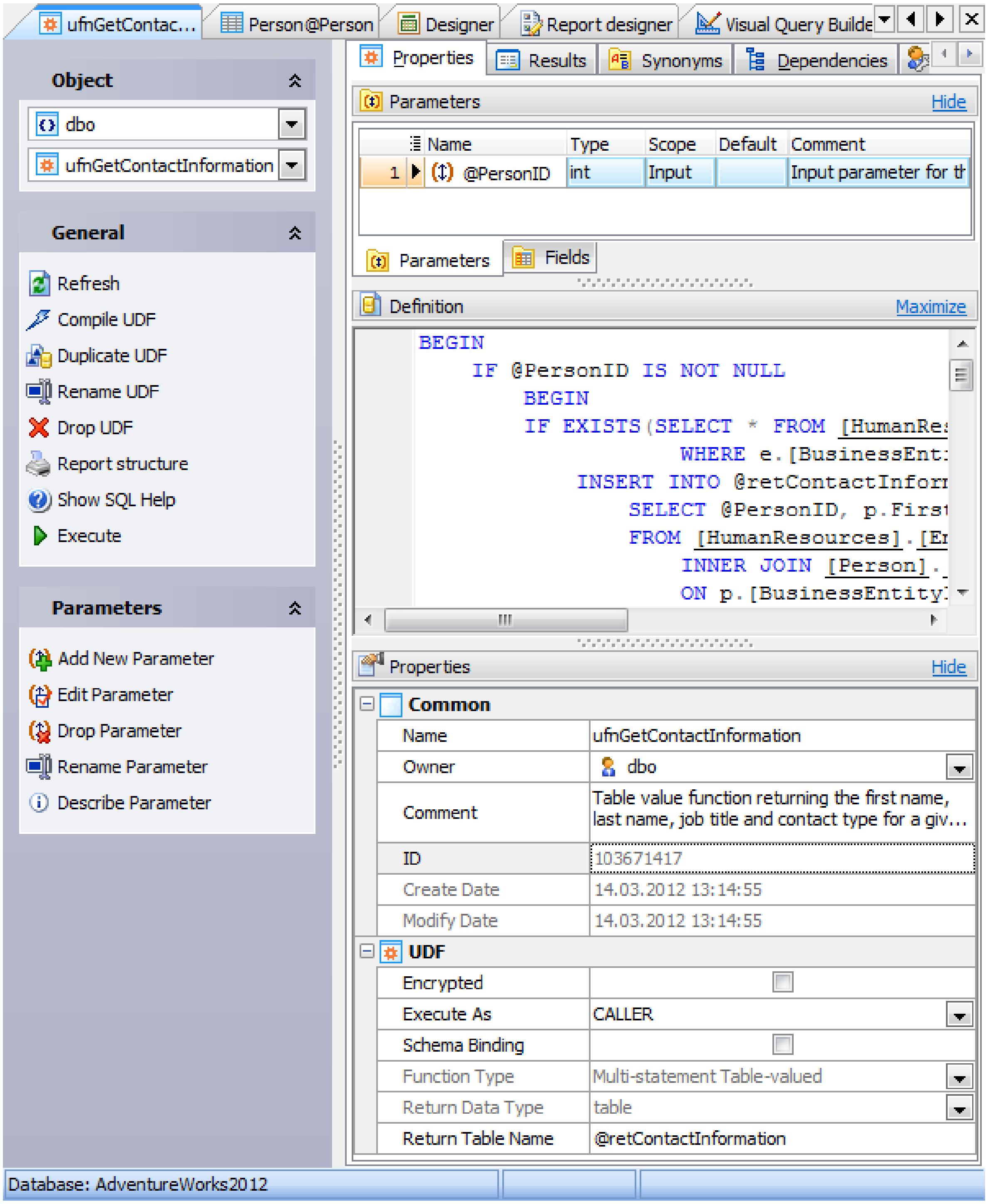
- Editing UDF properties 132
- Viewing UDF results 134

5.5.2.1 Editing properties

The Parameters tab contains the list of the current UDF parameters with its options. Here you can view the Name and the Type of each UDF parameter and also supply a Comment for the parameter.

Parameters can be edited within the Parameter Editor dialog window. In order to open the dialog you should

- open the object in its editor and the Parameters tab there;
- select the parameter to edit;
- press the **Enter** key or select the Edit Parameter item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).
  The Definition field contains the definition of the UDF. Specify a string constant defining the UDF here; the meaning depends on the language. It may be an internal UDF name, the path to an object file, an SQL command or text in a procedural language.



	Name	Type	Scope	Default	Comment
1	@PersonID	int	Input		Input parameter for th

```
BEGIN
    IF @PersonID IS NOT NULL
        BEGIN
        IF EXISTS(SELECT * FROM [HumanRes
                    WHERE e.[BusinessEnt
            INSERT INTO @retContactInforn
                SELECT @PersonID, p.First
                FROM [HumanResources].[En
                    INNER JOIN [Person].
                    ON p.[BusinessEntity]
```

Common	
Name	ufnGetContactInformation
Owner	dbo
Comment	Table value function returning the first name, last name, job title and contact type for a giv...
ID	103671417
Create Date	14.03.2012 13:14:55
Modify Date	14.03.2012 13:14:55
UDF	
Encrypted	☐
Execute As	CALLER
Schema Binding	☐
Function Type	Multi-statement Table-valued
Return Data Type	table
Return Table Name	@retContactInformation

Name

You can edit the UDF name here. The name of the UDF must be unique among all the UDF names in the database.

Owner

The field contains the owner of the UDF. By default, only the owner of an object can perform various operations with the object. In order to allow other users to operate it, privileges must be granted. (However, users that have the superuser attribute can always access any object.)

Comment
Specify a comment to the UDF if necessary.

Create Date
Displays the date when the UDF was created.

Modify Date
Displays the date when the UDF was last modified.

The Encrypted option is checked to indicate that Microsoft SQL server will convert the original text of the CREATE FUNCTION statement to an obfuscated format. The output of the obfuscation is not directly visible.

Execute As
Specify the security context under which the UDF is to be executed (For details see Execute As Clause (Microsoft SQL 2005 References)).

Schema Binding
Check the option to indicate that the UDF is bound to the database objects it references.

Function Type
Define *Scalar*, *Inline Table-valued* or *Multi-statement Table-valued* function type for the UDF.

Return Data Type
The return value of a scalar user-defined function is to be specified here. For Transact-SQL functions, all data types (including CLR user-defined types) are allowed except for the *timestamp* data type. For CLR UDFs, all data types (including CLR user-defined types) are allowed except for the *text*, *ntext*, *image* and *timestamp* data types. The nonscalar *cursor* and *table* types cannot be specified as a return data type in Transact-SQL or CLR UDFs.

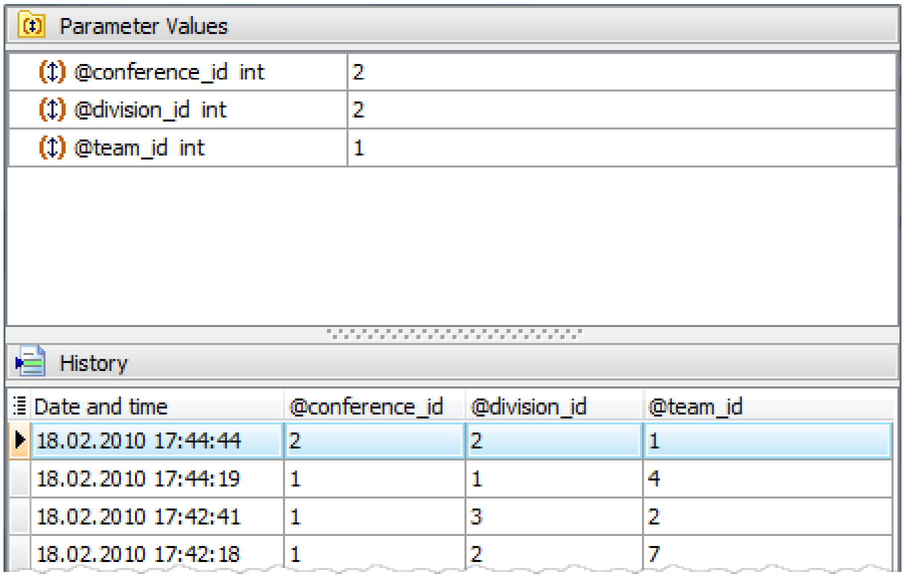
To apply the changes, select the Apply Changes item in the Navigation bar or use **Ctrl+F9** or **Ctrl+F7** shortcut keys.

It is also possible to modify object properties without opening the object editor: use the Object Properties item of the popup menu of the selected object from the explorer tree.

5.5.2.2 Viewing UDF results

Procedure/UDF/CLR UDF Editor provides an opportunity to execute current routine by opening the Results tab, by clicking the Execute item of the Navigation Bar, or by pressing the **F9** key.

If the procedure/UDF/CLR UDF has parameters, MS SQL Maestro will ask you to specify the values for these parameters in the Input parameters dialog which appears before the procedure execution. Input parameters dialog allows you to specify the values for all input parameters. After changes are made, click the OK button to execute the UDF, or the Cancel button to abort the execution.



@conference_id int	2
@division_id int	2
@team_id int	1

Date and time	@conference_id	@division_id	@team_id
18.02.2010 17:44:44	2	2	1
18.02.2010 17:44:19	1	1	4
18.02.2010 17:42:41	1	3	2
18.02.2010 17:42:18	1	2	7

MS SQL Maestro supports Parameter History. Values that have been set previously as the routine parameters are represented in the History tab of the Input Parameter dialog with a date and time of their last using. Double click a necessary set of values to set them as the routine parameters. You can manage the Parameter History with Delete history item and Clear history links.

The result of the successfully executed routine can be found within the Results tab of Procedure/UDF/CLR UDF Editor.

Note*:* If any unsaved changes are applied to the routine being currently edited, the execution of the routine is impossible until changes are saved by the Compile procedure item of the Navigation Bar.

5.6 UDTs

SQL Server 2005 adds support for **user-defined types (UDTs)** implemented with the Microsoft .NET Framework common language runtime (CLR). The CLR is integrated into SQL Server, and this new mechanism enables you to extend the type system of the database. UDTs provide user extensibility of the SQL Server data type system, and also the ability to define complex structured types.

⊟ How can I create a new UDT?

New UDTs are created within Create UDT Wizard. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the UDT icon in the Create Database Object dialog

or

- select the UDTs list or any object from that list in the explorer tree;
- select the Create New UDT... item from the popup menu

or

- open Schema Editor and the UDTs tab there;
- press the **Insert** key or select the Create New UDT item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

To create a new UDT with the same properties as one of the existing UDTs has:

- select the Object | Duplicate Database Object... main menu item;
- follow the instructions of Duplicate Object Wizard.

⊟ How can I edit an existing UDT?

UDTs can be edited within UDT Editor 139. In order to run the editor you should either

- select the UDT for editing in the explorer tree (type the first letters of the UDT name for quick search);
- select the Edit UDT... item from the popup menu

or

- open Schema Editor and the UDTs tab there;
- select the UDT to edit;
- press the **Enter** key or select the Edit UDT item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

You can change the name of the UDT using the Rename UDT dialog. To open the dialog you should either

- select the UDT to rename in the explorer tree;
- select the Rename UDT item from the popup menu

or

- open Schema Editor and the UDTs tab there;
- select the UDT to rename;
- select the Rename UDT item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

How can I drop a UDT?

To drop a UDT:

- select the UDT to drop in the explorer tree;
- select the Drop UDT item from the popup menu

or

- open Schema Editor and the UDTs tab there;
- select the UDT to drop;
- press the **Delete** key or select the Drop UDT item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.

5.6.1 Create UDT Wizard

New UDTs are created within Create UDT Wizard. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the UDT icon in the Create Database Object dialog

or

- select the UDTs list or any object from that list in the explorer tree;
- select the Create New UDT... item from the popup menu

or

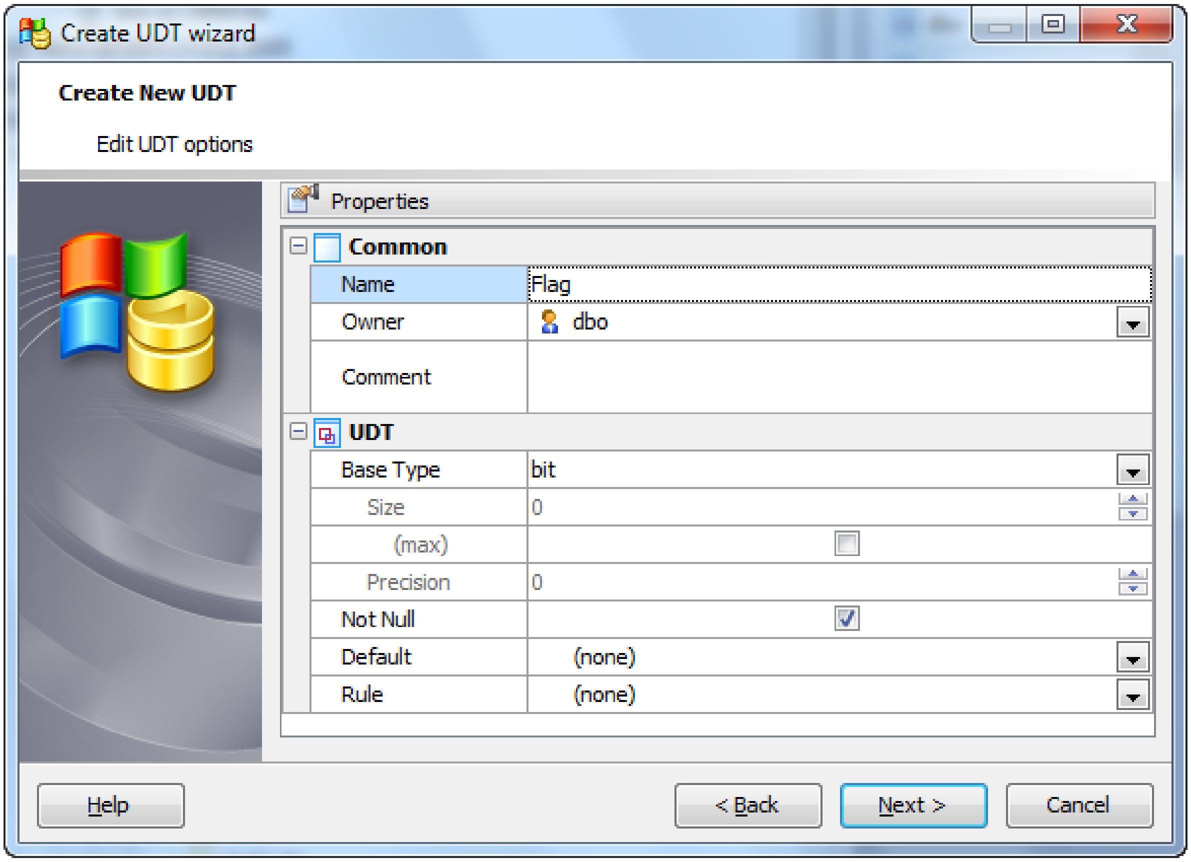
- open Schema Editor and the UDTs tab there;
- press the **Insert** key or select the Create New UDT item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

To create a new UDT with the same properties as one of the existing UDTs has:

- select the Object | Duplicate Database Object... main menu item;
- follow the instructions of Duplicate Object Wizard.

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

Specify the properties for the new UDT according to your needs. The detailed description is given below.



Name	Flag
Owner	dbo
Comment	
Base Type	bit
Size	0
(max)	☐
Precision	0
Not Null	☑
Default	(none)
Rule	(none)

Owner

Select the owner of the UDT from the drop-down list. By default, only the owner of an object can perform various operations with the object. In order to allow other users to operate it, privileges must be granted. (However, users that have the superuser attribute can always access any object.)

Comment

Supply a comment to the UDT if necessary.

Data Type

Select the underlying data type for the UDT. This may include array specifiers.

Use the Size edit box to define the length of the parameter value for *float*, *char* and other data types, and the Precision edit box to define the precision of the parameter value, e.g. for *decimal* data type.

☑ Not Null

The checkbox indicates that the values of the UDT are not allowed to be null.

☑ Default

The field displays the UDT value accepted by default when no value is explicitly supplied.

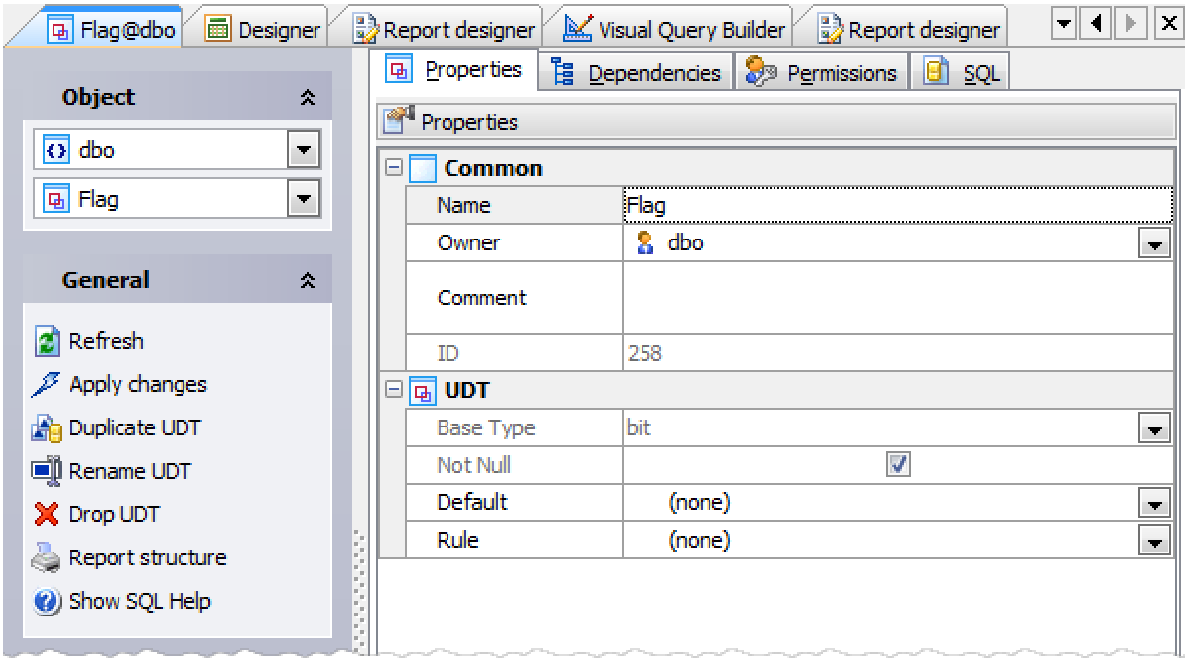
Rule
Specify a rule for the UDT.

5.6.2 UDT Editor

UDT Editor is opened automatically after a new UDT is created and is available on editing the existing one (see Edit UDT 136 for details).

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.

UDT Editor provides you with an ability to edit UDT properties. The Properties tab allows you to change the UDT name, the UDT owner, etc.



Name	Flag
Owner	dbo
Comment	
ID	258
Base Type	bit
Not Null	☑
Default	(none)
Rule	(none)

Name
Here you can view and edit the UDT name. The name of the UDT must be unique among all the UDT names in the database.

Owner
By default, only the owner of an object can perform various operations with the object. In order to allow other users to operate it, privileges must be granted. (However, users that have the superuser attribute can always access any object.)

Comment
This field stores a comment to the UDT.

☑ Not Null
The checkbox indicates that the values of the UDT are not allowed to be null.

Default

The field displays the UDT value accepted by default when no value is explicitly supplied.

Rule
The field specifies a rule for the UDT.

To apply the changes, select the Apply Changes item in the Navigation bar or use **Ctrl+F9** or **Ctrl+F7** shortcut keys.

It is also possible to modify object properties without opening the object editor: use the Object Properties item of the popup menu of the selected object from the explorer tree.

5.7 Synonyms

A synonym is an alternative name for a schema-scoped object. You can use a single-part name to reference a base object by using a synonym instead of using a two-part, three-part, or four-part name to reference the base object. Synonyms were implemented in SQL Server 2005.

How can I create a new synonym?

New synonyms are created within Create Synonym Wizard. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the Synonym icon in the Create Database Object dialog

or

- select the Synonyms list or any object from that list in the explorer tree;
- select the Create New Synonym... item from the popup menu

or

- open the schema in Schema Editor and the Synonyms tab there;
- press the **Insert** key or select the Create New Synonym... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

To create a new synonym with the same properties as one of the existing synonyms has:

- select the Object | Duplicate Database Object... main menu item;
- follow the instructions of Duplicate Object Wizard.

How can I edit an existing synonym?

Synonyms can be edited within Synonym Editor. In order to run the editor you should either

- select the synonym for editing in the explorer tree (type the first letters of the synonym name for quick search);
- select the Edit Synonym ... item from the popup menu

or

- open the schema in Schema Editor and the Synonyms tab there;
- select the synonym to edit;
- press the **Enter** key or select the Edit Synonym item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

You can change the name of the synonym using the Rename Synonym dialog. To open the dialog you should either

- select the synonym to rename in the explorer tree;
- select the Rename Synonym item from the popup menu

or

- open the schema in Schema Editor and the Synonyms tab there;
- select the synonym to rename;
- select the Rename Synonym item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

How can I drop a synonym?

To drop a synonym:

- select the synonym to drop in the explorer tree;
- select the Drop Synonym item from the popup menu

or

- open the schema in Schema Editor and the Synonyms tab there;
- select the synonym to drop;
- press the **Delete** key or select the Drop Synonym item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.

5.7.1 Create Synonym Wizard

New synonyms are created within Create Synonym Wizard. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the Synonym icon in the Create Database Object dialog

or

- select the Synonyms list or any object from that list in the explorer tree;
- select the Create New Synonym... item from the popup menu

or

- open the schema in Schema Editor and the Synonyms tab there;
- press the **Insert** key or select the Create New Synonym... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

To create a new synonym with the same properties as one of the existing synonyms has:

- select the Object | Duplicate Database Object... main menu item;
- follow the instructions of Duplicate Object Wizard.

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

Owner
Select the owner for the synonym. By default, only the owner of an object can perform various operations with the object. In order to allow other users to operate it, privileges must be granted. (However, users that have the superuser attribute can always access any object.)

Comment
Specify a comment to the synonym.

Object
Specify the base object that the synonym references.

See also: Synonym Editor 143

5.7.2 Synonym Editor

Synonyms can be edited within Synonym Editor. In order to run the editor you should either

- select the synonym for editing in the explorer tree (type the first letters of the synonym name for quick search);
- select the Edit Synonym ... item from the popup menu

or

- open the schema in Schema Editor and the Synonyms tab there;
- select the synonym to edit;
- press the **Enter** key or select the Edit Synonym item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

You can change the name of the synonym using the Rename Synonym dialog. To open the dialog you should either

- select the synonym to rename in the explorer tree;
- select the Rename Synonym item from the popup menu

or

- open the schema in Schema Editor and the Synonyms tab there;
- select the synonym to rename;
- select the Rename Synonym item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.

See also: Synonym properties 143, Create Synonym Wizard 142

5.7.2.1 Editing synonym properties

Synonym Properties

Name
You can edit the synonym name here.

Owner
Here you can view the owner for the synonym. By default, only the owner of an object can perform various operations with the object. In order to allow other users to operate it, privileges must be granted. (However, users that have the superuser attribute can

always access any object.)

Comment
Specify a comment to the synonym.

Create Date
Stores the date when the synonym was created.

Modify Date
Stores the date when the synonym was last modified.

Object
The field represents the base object that the synonym references.

Synonym Data

The Data tab displays the Alias data as a grid (see Data View 328 for details). Use grid's popup menu to open Data Input Form, to invoke the Export Data, and Get SQL Dump modules, to set the value of the selected record to *NULL* or to *Now* (for *Date* values). For your convenience it was implemented two modes of viewing data: as table and as info cards.

5.8 Rules

A SQL Server rule is a Transact-SQL syntax element that defines a data-integrity constraint. A rule can be bound to a column or user-defined data type. The condition is executed to validate data for a single column when a value is inserted into the column bound by the rule. A rule condition can be any expression valid in a WHERE clause and can include elements such as arithmetic operators, relational operators, and predicates (for example, IN, LIKE, BETWEEN). Rules will be removed in the future versions of Microsoft SQL Server. Avoid using rules in new development work, and be ready to modify the applications that currently use it. We recommend that you use check 92 constraints instead.

⊟ How can I create a new rule?

New rules are created within Create Rule Wizard. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the Rule icon in the Create Database Object dialog

or

- select the Rules list or any object from that list in the explorer tree;
- select the Create New Rule... item from the popup menu

or

- open the schema in Schema Editor and the Rules tab there;
- press the Insert key or select the Create New Rule... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

To create a new rule with the same properties as one of the existing rules has:

- select the Object | Duplicate Database Object... main menu item;
- follow the instructions of Duplicate Object Wizard 56.

⊟ How can I edit an existing rule?

Rules can be edited within Rule Editor. In order to run the editor you should either

- select the rule for editing in the explorer tree (type the first letters of the rule name for quick search);
- select the Edit Rule... item from the popup menu

or

- open the schema in Schema Editor and the Rules tab there;
- select the rule to edit;
- press the **Enter** key or select the Edit Rule item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

You can change the name of the rule using the Rename Rule dialog.

To open the dialog you should either

- select the rule to rename in the explorer tree;
- select the Rename Rule item from the popup menu

or

- open the schema in Schema Editor and the Rules tab there;
- select the rule to rename;
- select the Rename Rule item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

⊟ How can I drop a rule?

To drop a rule:

- select the rule to drop in the explorer tree;
- select the Drop Rule item from the popup menu

or

- open the schema in Schema Editor 73 and the Rules tab there;
- select the rule to drop;
- press the **Delete** key or select the Drop Rule item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.

5.8.1 Create Rule Wizard

New rules are created within Create Rule Wizard. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the Rule icon in the Create Database Object dialog

or

- select the Rules list or any object from that list in the explorer tree;
- select the Create New Rule... item from the popup menu

or

- open the schema in Schema Editor and the Rules tab there;
- press the Insert key or select the Create New Rule... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

To create a new rule with the same properties as one of the existing rules has:

- select the Object | Duplicate Database Object... main menu item;
- follow the instructions of Duplicate Object Wizard 56.

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

To add a new rule, specify a Comment to the rule and the rule Condition. As a condition you can use any expression valid in a WHERE clause and which can include elements such as arithmetic operators, relational operators, and predicates (for example, IN, LIKE, BETWEEN).

A rule cannot reference columns or other database objects. Built-in functions that do not reference database objects can be included. User-defined functions cannot be used.

The condition must include one variable. The at sign (@) precedes each local variable. The expression refers to the value entered with the UPDATE or INSERT statement. Any name or symbol can be used to represent the value when creating the rule, but the first character must be the at sign (@).

See also: Rule Editor 147

5.8.2 Rule Editor

Rules can be edited within Rule Editor. In order to run the editor you should either

- select the rule for editing in the explorer tree (type the first letters of the rule name for quick search);
- select the Edit Rule... item from the popup menu

or

- open the schema in Schema Editor and the Rules tab there;
- select the rule to edit;
- press the **Enter** key or select the Edit Rule item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

You can change the name of the rule using the Rename Rule dialog. To open the dialog you should either

- select the rule to rename in the explorer tree;
- select the Rename Rule item from the popup menu

or

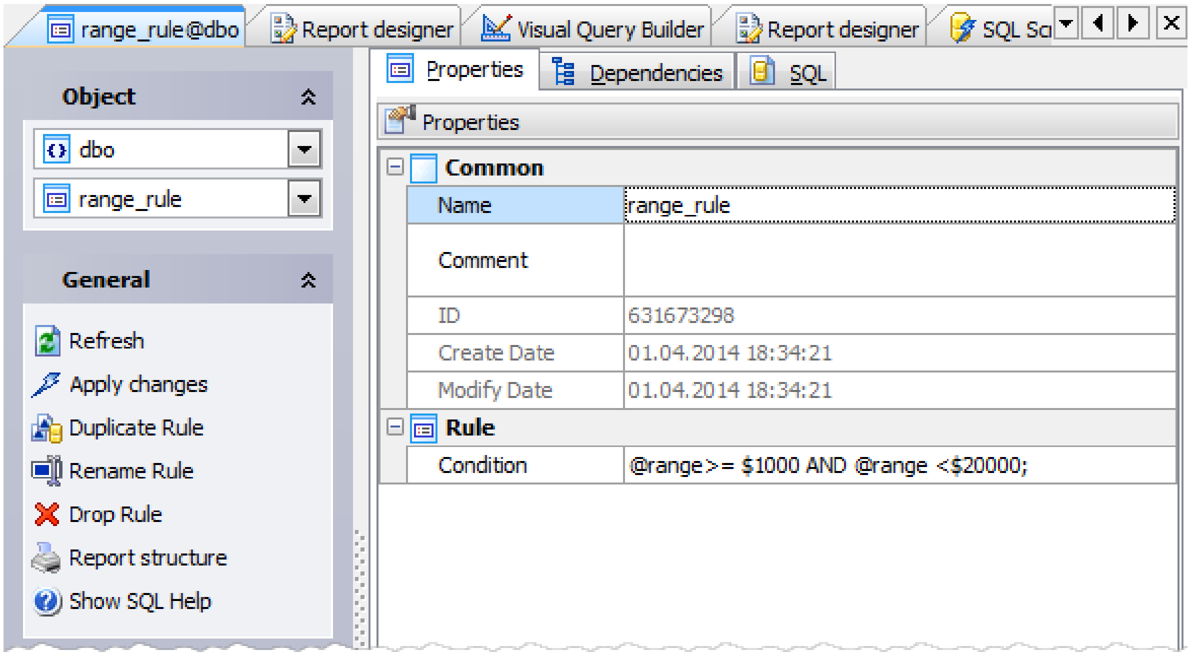
- open the schema in Schema Editor and the Rules tab there;
- select the rule to rename;
- select the Rename Rule item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.

See also: Rule properties 147, Create Rule Wizard 146

5.8.2.1 Editing rule properties

Specify rule options according to your needs. The detailed description is given below.



Name	range_rule
Comment	
ID	631673298
Create Date	01.04.2014 18:34:21
Modify Date	01.04.2014 18:34:21
Condition	@range>= $1000 AND @range <$20000;

Name
Defines a name for the rule.

Comment
Specify a comment to the rule.

Create Date
Displays the date when the rule was created.

Modify Date
Displays the date when rule was last modified.

Condition
The field represents the condition or conditions that define the rule. As a condition you can use any expression valid in a WHERE clause and which can include elements such as arithmetic operators, relational operators, and predicates (for example, IN, LIKE, BETWEEN).

A rule cannot reference columns or other database objects. Built-in functions that do not reference database objects can be included. User-defined functions cannot be used.

The condition must include one variable. The at sign (@) precedes each local variable. The expression refers to the value entered with the UPDATE or INSERT statement. Any name or symbol can be used to represent the value when creating the rule, but the first character must be the at sign (@).

To apply the changes, select the Apply Changes item in the Navigation bar or use **Ctrl+F9** or **Ctrl+F7** shortcut keys.

It is also possible to modify object properties without opening the object editor: use the

Object Properties item of the popup menu of the selected object from the explorer tree.

5.9 Defaults

A default specifies a value to be inserted into the column to which the object is bound (or into all columns, in the case of a user-defined data type) when no value is explicitly supplied during an insert action.

How can I create a new default?

New defaults are created within Create Default Wizard. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the Default icon in the Create Database Object dialog

or

- select the Defaults list or any object from that list in the explorer tree;
- select the Create New Default... item from the popup menu

or

- open the schema in Schema Editor and the Defaults tab there;
- press the **Insert** key or select the Create New Default item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

To create a new default with the same properties as one of the existing defaults has:

- select the Object | Duplicate Database Object... main menu item;
- follow the instructions of Duplicate Object Wizard.

How can I edit an existing default?

Defaults can be edited within Default Editor. In order to run the editor you should either

- select the default for editing in the explorer tree (type the first letters of the default name for quick search);
- select the Edit Default ... item from the popup menu

or

- open the schema in Schema Editor and the Defaults tab there;
- select the default to edit;
- press the **Enter** key or select the Edit Default item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

You can change the name of the default using the Rename Default dialog. To open the dialog you should either

- select the default to rename in the explorer tree;
- select the Rename Default item from the popup menu

or

- open the schema in Schema Editor and the Defaults tab there;

- select the default to rename;
- select the Rename Default item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

How can I drop a default?

To drop a default:

- select the default to drop in the explorer tree;
- select the Drop Default item from the popup menu

or

- open the schema in Schema Editor and the Defaults tab there;
- select the default to drop;
- press the **Delete** key or select the Drop Default item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.

5.9.1 Create Default Wizard

New defaults are created within Create Default Wizard. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the Default icon in the Create Database Object dialog

or

- select the Defaults list or any object from that list in the explorer tree;
- select the Create New Default... item from the popup menu

or

- open the schema in Schema Editor and the Defaults tab there;
- press the **Insert** key or select the Create New Default item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

To create a new default with the same properties as one of the existing defaults has:

- select the Object | Duplicate Database Object... main menu item;
- follow the instructions of Duplicate Object Wizard.

To add a new default, specify a comment to the default and set the default value. The value must be an expression that contains only constant values (it cannot include the names of any columns or other database objects). Any constant, built-in function, or mathematical expression can be used, except those that contain alias data types. User-defined functions cannot be used. Enclose character and date constants in single quotation marks ('); monetary, integer, and floating-point constants do not require quotation marks. Binary data must be preceded by 0x, and monetary data must be preceded by a dollar sign ($). The default value must be compatible with the column data type.

See also: Default Editor 152

5.9.2 Default Editor

Defaults can be edited within Default Editor. In order to run the editor you should either

- select the default for editing in the explorer tree (type the first letters of the default name for quick search);
- select the Edit Default ... item from the popup menu

or

- open the schema in Schema Editor and the Defaults tab there;
- select the default to edit;
- press the **Enter** key or select the Edit Default item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

You can change the name of the default using the Rename Default dialog. To open the dialog you should either

- select the default to rename in the explorer tree;
- select the Rename Default item from the popup menu

or

- open the schema in Schema Editor and the Defaults tab there;
- select the default to rename;
- select the Rename Default item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic[48]. Below you will find a description of editor tabs that are unique for the current object.

See also: Default properties[152], Create Default Wizard[151]

5.9.2.1 Editing default properties

Specify default options according to your needs. The detailed description is given below.

Name
You can edit the default name here.

Comment
Specify a comment to the default.

Create Date
Indicates the date when the default was created.

Modify Date
Indicates the date when the default was last modified.

Value
Specify the value for the default. It can be an expression that contains only constant values (it cannot include the names of any columns or other database objects). Any constant, built-in function, or mathematical expression can be used, except those that contain alias data types. User-defined functions cannot be used. Enclose character and date constants in single quotation marks ('); monetary, integer, and floating-point constants do not require quotation marks. Binary data must be preceded by 0x, and

monetary data must be preceded by a dollar sign ($). The default value must be compatible with the column data type.

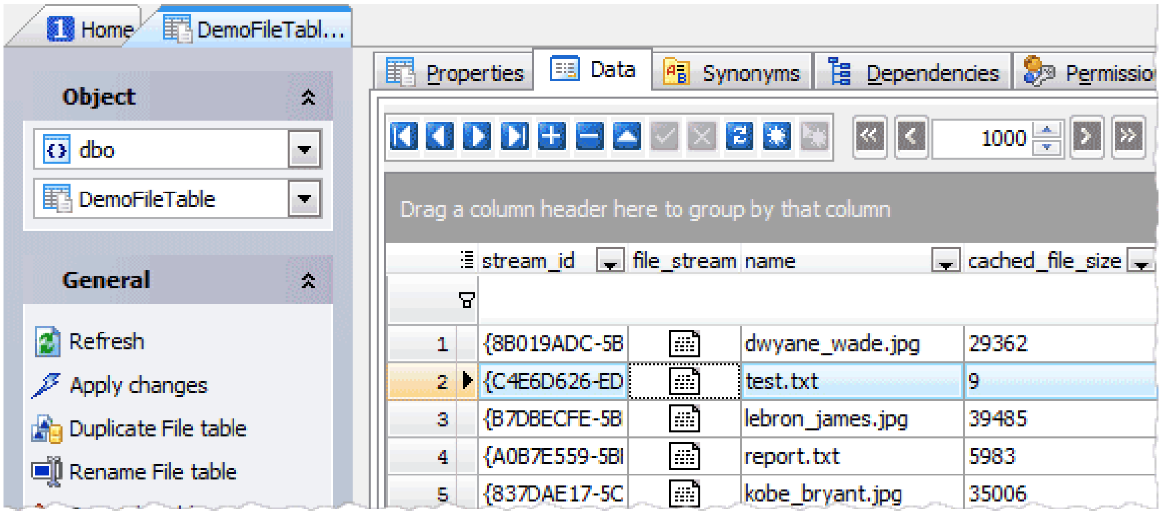
5.10 File Tables

File tables brings support for the Windows file namespace and compatibility with Windows applications to the file data stored in SQL Server. You can store files and documents in special tables in SQL Server, but access them from Windows applications as if they were stored in the file system, without making any changes to your client applications. Find out more on file tables in Microsoft SQL Server community.

A file table is a specialized user table that has a pre-defined and fixed schema. This schema stores *FILESTREAM* data, file and directory information, and file attributes. To add a new file table, you need to specify the directory that serves as the root directory for all the files and directories stored in the file table as File Name Directory Name, and the name of the collation to be applied to the Name column in the file table as File Name Column Collation. If you do not provide a directory name when you create the file table, then the name of the file table itself is used as the directory name.

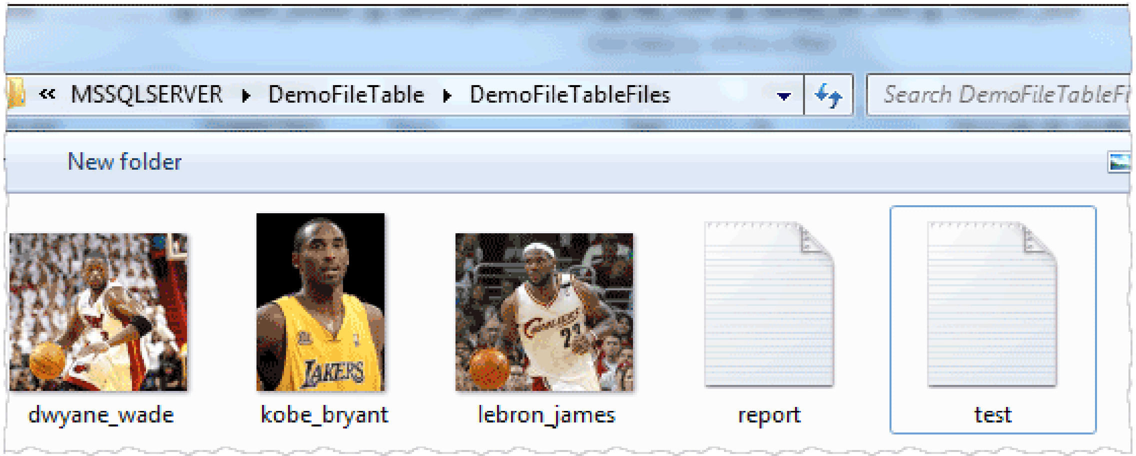
After the creation, file tables are available to use as ordinary database tables. You can work with their data within the corresponding editor as usual: load files to tables 328, query, export, and import tables data and as well as through the Windows file system.

File Table Editor



	stream_id	file_stream	name	cached_file_size
1	{8B019ADC-5B		dwyane_wade.jpg	29362
2	{C4E6D626-ED		test.txt	9
3	{B7DBECFE-5B		lebron_james.jpg	39485
4	{A0B7E559-5BI		report.txt	5983
5	{837DAE17-5C		kobe_bryant.jpg	35006

Access to table files through the Windows file system



5.11 CLR Procedures

A CLR Procedure is a reference to a Microsoft .NET Framework common language runtime (CLR) method that can take and return user-supplied parameters.

⊟ How can I create a new CLR Procedure?

New CLR procedures are created within Create CLR Procedure Wizard.
In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the CLR Procedure icon in the Create Database Object dialog

or

- select the CLR Procedures list or any object from that list in the explorer tree;
- select the Create New CLR Procedure... item from the popup menu

or

- open the schema in Schema Editor and the CLR Procedures tab there;
- press the **Insert** key or select the Create New CLR Procedure... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

To create a new CLR procedure with the same properties as one of the existing CLR procedures has:

- select the Object | Duplicate Database Object... main menu item;
- follow the instructions of Duplicate Object Wizard.

⊟ How can I edit an existing CLR Procedure?

CLR procedures can be edited within CLR Procedure Editor. In order to run the editor you should either

- select the CLR procedure for editing in the explorer tree (type the first letters of the CLR procedure name for quick search);
- select the Edit CLR Procedure ... item from the popup menu

or

- open the schema in Schema Editor and the CLR Procedures tab there;
- select the CLR procedure to edit;
- press the **Enter** key or select the Edit CLR Procedure item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

You can change the name of the CLR procedure using the Rename CLR Procedure dialog. To open the dialog you should either

- select the CLR procedure to rename in the explorer tree;

- select the Rename CLR Procedure item from the popup menu

or

- open the schema in Schema Editor and the CLR Procedures tab there;
- select the CLR procedure to rename;
- select the Rename CLR Procedure item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

⊟ How can I execute a CLR Procedure?

To execute a CLR procedure:

- select the CLR procedure in the explorer tree (type the first letters of the CLR procedure name for quick search);
- select the Edit CLR Procedure ... item from the popup menu;
- execute the CLR procedure using the Execute link of the Navigation Bar

or

- open the schema in Schema Editor and the CLR Procedures tab there;
- select the CLR procedure to execute;
- press the **Enter** key or select the Edit CLR Procedure item from the popup menu, or use the corresponding link of the Navigation Bar;
- execute the CLR procedure using the Execute link of the Navigation bar.

⊟ How can I drop a CLR Procedure?

To drop a CLR procedure:

- select the CLR procedure to drop in the explorer tree;
- select the Drop CLR Procedure item from the popup menu

or

- open the schema in Schema Editor and the CLR Procedures tab there;
- select the CLR procedure to drop;
- press the **Delete** key or select the Drop CLR Procedure item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.

5.11.1 Create CLR Procedure Wizard

New CLR procedures are created within Create CLR Procedure Wizard. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the CLR Procedure icon in the Create Database Object dialog

or

- select the CLR Procedures list or any object from that list in the explorer tree;
- select the Create New CLR Procedure... item from the popup menu

or

- open the schema in Schema Editor and the CLR Procedures tab there;
- press the **Insert** key or select the Create New CLR Procedure... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

To create a new CLR procedure with the same properties as one of the existing CLR procedures has:

- select the Object | Duplicate Database Object... main menu item;
- follow the instructions of Duplicate Object Wizard.

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

Specifying CLR Procedure options

Use this step to specify Comment and Owner for the CLR procedure. By default, only the owner of an object can perform various operations with the object. In order to allow other users to operate it, privileges must be granted. (However, users that have the superuser attribute can always access any object.)

The Execute As option defines the security context under which the CLR procedure is to be executed (for details see Execute As Clause (Microsoft SQL 2005 References)).

Assembly, Class, Method

Specifies the method of a .NET Framework assembly for a CLR stored procedure to reference. **Class** must be a valid SQL Server identifier and it must exist as a class in the assembly 214. The assembly is to have been already created in the database.

Managing CLR Procedure parameters

Click the Add button to set new parameter options within Parameter Editor.

Use the Parameter name field to set the name to a parameter. Note that the name of the parameter must be unique among parameter names in the procedure.

The Data type drop-down list defines the type of the parameter.

The Default value box specifies a parameter value when no value is explicitly supplied.

Use the Scope box to set whether the parameter is Input or Input-Output.

The Comment field specifies a comment to the parameter.

See also: CLR Procedure Editor 158

5.11.2 CLR Procedure Editor

CLR procedures can be edited within CLR Procedure Editor. In order to run the editor you should either

- select the CLR procedure for editing in the explorer tree (type the first letters of the CLR procedure name for quick search);
- select the Edit CLR Procedure ... item from the popup menu

or

- open the schema in Schema Editor and the CLR Procedures tab there;
- select the CLR procedure to edit;
- press the **Enter** key or select the Edit CLR Procedure item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

You can change the name of the CLR procedure using the Rename CLR Procedure dialog. To open the dialog you should either

- select the CLR procedure to rename in the explorer tree;
- select the Rename CLR Procedure item from the popup menu

or

- open the schema in Schema Editor and the CLR Procedures tab there;
- select the CLR procedure to rename;
- select the Rename CLR Procedure item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.

- CLR procedure properties 160
- Editing CLR procedure properties 159

See also: Create CLR Procedure Wizard 157

5.11.2.1 Editing properties

The Properties tab consists of two parts: Parameters and Properties.

The Parameters tab contains the list of the current CLR procedure parameters with its options. Here you can find the Name and the Type of each parameter of the CLR procedure and also the Comment for the parameter.

The Properties tab allows you to specify CLR procedure options according to your needs. The detailed description is given below.

Name
You can edit the CLR procedure name here.

Owner
The field represents the owner of the CLR procedure. By default, only the owner of an object can perform various operations with the object. In order to allow other users to operate it, privileges must be granted. (However, users that have the superuser attribute can always access any object.)

Comment
Specify a comment to the CLR procedure.

Create Date
Indicates the date when the CLR procedure was created.

Modify Date
Indicates the date when the CLR procedure was last modified.

Execute As
Specifies the security context under which to execute the CLR procedure (For details see Execute As Clause (Microsoft SQL 2005 References)).

Parameters
Here you can find the list of the CLR procedure parameters.

Assembly, Class, Method
Specifies the method of a .NET Framework assembly for a CLR stored procedure to reference. Class must be a valid SQL Server identifier and must exist as a class in the Assembly 214.

5.11.2.2 Executing CLR procedure

CLR Procedure Editor provides an ability to execute current procedure. For this purpose use the corresponding item at the Navigation Bar. If the procedure has parameters, the Input Parameters dialog appears before the execution. It allows you to specify the values for all input parameters. After changes are made, click the OK button to execute the CLR procedure, or the Cancel button to abort the execution.

Note: If any unsaved changes are applied to the procedure currently edited, the execution of the procedure is impossible until changes are saved by the Compile CLR procedure item of the Navigation Bar.

5.12 CLR UDFs

User-defined functions are routines that can take parameters, perform calculations or other actions, and return a result. In Microsoft SQL Server 2005, you can write user-defined functions in any Microsoft .NET Framework programming language, such as Microsoft Visual Basic .NET or Microsoft C#.

⊟ How can I create a new CLR UDF?

New CLR UDFs are created within Create CLR UDF Wizard. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the CLR UDF icon in the Create Database Object dialog

or

- select the CLR UDFs list or any object from that list in the explorer tree;
- select the Create New CLR UDF... item from the popup menu

or

- open the schema in Schema Editor and the CLR UDFs tab there;
- press the **Insert** key or select the Create New CLR UDF... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

To create a new CLR UDF with the same properties as one of the existing CLR UDFs has:

- select the Object | Duplicate Database Object... main menu item;
- follow the instructions of Duplicate Object Wizard.

⊟ How can I edit an existing CLR UDF?

CLR UDFs can be edited within CLR UDF Editor. In order to run the editor you should either

- select the CLR UDF for editing in the explorer tree (type the first letters of the CLR UDF name for quick search);
- select the Edit CLR UDF ... item from the popup menu

or

- open the schema in Schema Editor and the CLR UDFs tab there;
- select the CLR UDF to edit;
- press the **Enter** key or select the Edit CLR UDF item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

You can change the name of the CLR UDF using the Rename CLR UDF dialog. To open the dialog you should either

- select the CLR UDF to rename in the explorer tree;
- select the Rename CLR UDF item from the popup menu

or

- open the schema in Schema Editor and the CLR UDFs tab there;
- select the CLR UDF to rename;
- select the Rename CLR UDF item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

How can I execute a CLR UDF?

To execute a CLR UDF:

- select the CLR UDF in the explorer tree (type the first letters of the CLR UDF name for quick search);
- select the Edit CLR UDF ... item from the popup menu;
- execute the CLR UDF using the Execute link of the Navigation Bar

or

- open the schema in Schema Editor and the CLR UDFs tab there;
- select the CLR UDF to execute;
- press the **Enter** key or select the Edit CLR UDF item from the popup menu, or use the corresponding link of the Navigation Bar;
- execute the CLR UDF using the Execute link of the Navigation bar .

How can I drop a CLR UDF?

To drop a CLR UDF:

- select the CLR UDF to drop in the explorer tree;
- select the Drop CLR UDF item from the popup menu

or

- open the schema in Schema Editor and the CLR UDFs tab there;
- select the CLR UDF to drop;
- press the **Delete** key or select the Drop CLR UDF item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.

5.12.1 Create CLR UDF Wizard

New CLR UDFs are created within Create CLR UDF Wizard. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the CLR UDF icon in the Create Database Object dialog

or

- select the CLR UDFs list or any object from that list in the explorer tree;
- select the Create New CLR UDF... item from the popup menu

or

- open the schema in Schema Editor and the CLR UDFs tab there;
- press the **Insert** key or select the Create New CLR UDF... item from the popup menu

(alternatively, you may use the corresponding link of the Navigation Bar).

To create a new CLR UDF with the same properties as one of the existing CLR UDFs has:

- select the Object | Duplicate Database Object... main menu item;
- follow the instructions of Duplicate Object Wizard.

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

CLR UDF options

Owner
Select the owner of the CLR UDF. By default, only the owner of an object can perform various operations with the object. In order to allow other users to operate it, privileges must be granted. (However, users that have the superuser attribute can always access any object.)

Comment
Specify a comment to the CLR UDF.

Execute As
Specifies the security context under which the CLR UDF is to be executed.

Function Type
Defines Scalar, Inline Table-valued or Multi-statement Table-valued UDF function type.

Return Data Type
The field represents the return data type of UDF. All data types, including CLR user-defined types, are allowed except text, ntext, image, and timestamp data types.

Assembly, Class, Method
Specifies the method of a .NET Framework assembly for a CLR stored function to reference. Class must be a valid SQL Server identifier and it must exist as a class in the Assembly 214. The assembly is to have been already created in the database.

Managing parameters

Click the Add button to add a new parameter and set its properties in Parameter Editor 53. Click the Edit button to edit the selected parameter, or the Delete button to delete one.

See also: CLR UDF Editor 163

5.12.2 CLR UDF Editor

CLR UDF Editor allows you to execute the existing CLR UDFs or edit its definition (CLR UDF name, CLR UDF value, etc.). It can be opened when you create a new CLR UDF or edit the existing one (see How to edit CLR UDF 161 for details).

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.

- CLR UDF Properties 164

The Results tab represents the result of the successfully executed CLR UDF.

See also: Create CLR UDF Wizard 162

5.12.2.1 Editing CLR UDF properties

The Properties tab consists of two parts: Parameters and Properties.

The Parameters tab contains the list of the current CLR UDF parameters with its options. Here you can find the Name and the Type of each parameter of the CLR UDF and also the Comment for the parameter.

The Properties tab allows you to specify CLR UDF options according to your needs. The detailed description is given below.

Name
You can edit the CLR UDF name here.

Owner
Select the owner of the CLR UDF. By default, only the owner of an object can perform various operations with the object. In order to allow other users to operate it, privileges must be granted. (However, users that have the superuser attribute can always access any object.)

Comment
Specify a comment to the CLR UDF.

Create Date
Indicates the date when the CLR UDF was created.

Modify Date
Indicates the date when the CLR UDF was last modified.

Execute As
Specifies the security context under which the CLR UDF is to be executed.

Function Type
Defines Scalar, Inline Table-valued or Multi-statement Table-valued UDF function type.

Return Data Type
The field stands for the return data type of UDF. All data types, including CLR user-defined types, are allowed except text, ntext, image, and timestamp data types.

Parameters
Here you can find the list of the CLR UDF parameters.

Assembly, Class, Method
Specifies the method of a .NET Framework assembly for a CLR stored function to reference. Class must be a valid SQL Server identifier and must exist as a class in the Assembly.

See also: Assemblies 214

To apply the changes, select the Apply Changes item in the Navigation bar or use **Ctrl+F9** or **Ctrl+F7** shortcut keys.

It is also possible to modify object properties without opening the object editor: use the Object Properties item of the popup menu of the selected object from the explorer tree.

5.13 CLR UDTs

Microsoft SQL Server 2005 adds support for user-defined types (UDTs) implemented with the Microsoft .NET Framework common language runtime (CLR). The CLR is integrated into the SQL Server, and this new mechanism enables you to extend the type system of the database. UDTs provide user extensibility of the SQL Server data type system, and also the ability to define complex structured types.

⊟ How can I create a new CLR UDT?

New CLR UDTs are created within Create CLR UDT Wizard. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the CLR UDT icon in the Create Database Object dialog

or

- select the CLR UDTs list or any object from that list in the explorer tree;
- select the Create CLR UDT... item from the popup menu

or

- open the schema in Schema Editor and the CLR UDTs tab there;
- press the **Insert** key or select the Create New CLR UDT... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

To create a new CLR UDT with the same properties as one of the existing CLR UDTs has:

- select the Object | Duplicate Database Object... main menu item;
- follow the instructions of Duplicate Object Wizard.

⊟ How can I edit an existing CLR UDT?

CLR UDTs can be edited within CLR UDT Editor. In order to run the editor you should either

- select the CLR UDT for editing in the explorer tree (type the first letters of the CLR UDT name for quick search);
- select the Edit CLR UDT ... item from the popup menu

or

- open the schema in Schema Editor and the CLR UDTs tab there;
- select the CLR UDT to edit;
- press the **Enter** key or select the Edit CLR UDT item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

You can change the name of the CLR UDT using the Rename CLR UDT dialog. To open the dialog you should either

- select the CLR UDT to rename in the explorer tree;
- select the Rename CLR UDT item from the popup menu

or

- open the schema in Schema Editor and the CLR UDTs tab there;
- select the CLR UDT to rename;
- select the Rename CLR UDT item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

⊟ How can I drop a CLR UDT?

To drop a CLR UDT:

- select the CLR UDT to drop in the explorer tree;
- select the Drop CLR UDT item from the popup menu

or

- open the schema in the Schema Editor and the CLR UDTs tab there;
- select the CLR UDT to drop;
- press the **Delete** key or select the Drop CLR UDT item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.

5.13.1 Create CLR UDT Wizard

Create CLR UDT Wizard guides you through the process of creating a new CLR UDT. See How To Create CLR UDT 166 for instructions on running this wizard.

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

To add a new CLR UDT, specify its Comment and Owner. By default, only the owner of an object can perform various operations with the object. In order to allow other users to operate it, privileges must be granted. (However, users that have the superuser attribute can always access any object.)

Assembly, Class
Specifies the method of a .NET Framework assembly for a CLR stored function to reference. Class must be a valid SQL Server identifier and must exist as a class in the Assembly 214.

See also: CLR UDT Editor 167

5.13.2 CLR UDT Editor

CLR UDTs can be edited within CLR UDT Editor. In order to run the editor you should either

- select the CLR UDT for editing in the explorer tree (type the first letters of the CLR UDT name for quick search);

- select the Edit CLR UDT ... item from the popup menu

or

- open the schema in Schema Editor and the CLR UDTs tab there;
- select the CLR UDT to edit;
- press the **Enter** key or select the Edit CLR UDT item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

You can change the name of the CLR UDT using the Rename CLR UDT dialog. To open the dialog you should either

- select the CLR UDT to rename in the explorer tree;
- select the Rename CLR UDT item from the popup menu

or

- open the schema in Schema Editor and the CLR UDTs tab there;
- select the CLR UDT to rename;
- select the Rename CLR UDT item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.

CLR UDF Properties

Name
Specify the CLR UDT name.

Owner
Select the owner of the CLR UDT.

Comment
Specify a comment to the CLR UDT.

Assembly, Class
Defines the method of a .NET Framework assembly for a CLR stored function to reference. Class must be a valid SQL Server identifier and must exist as a class in the Assembly 214.

See also: Create UDT Wizard 151

5.14 XML Schema Collection

SQL Server provides native storage of XML data through the xml data type. You can optionally associate XSD schemas with a variable or a column of xml type through an XML schema collection. The XML schema collection stores the imported XML schemas and is then used to do the following:

- Validate XML instances
- Type the XML data as it is stored in the database

Before you can create typed XML variables, parameters or columns, you must first register the XML schema collection. You can then associate the XML schema collection with variables, parameters, or columns of the xml data type.

Note: the XML schema collection is a metadata entity like a table in the database. You can create, modify, and drop them.

How can I create a new XML schema collection?

New XML schema collections are created within Create XML Schema Collection Wizard. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the XML Schema Collection icon in the Create Database Object dialog

or

- select the XML Schema Collections list or any object from that list in the explorer tree;
- select the Create New XML Schema Collection... item from the popup menu

or

- open the schema in Schema Editor and the XML Schema Collections tab there;
- press the **Insert** key or select the Create New XML Schema Collection... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

To create a new XML schema collection with the same properties as one of the existing XML schema collections has:

- select the Object | Duplicate Database Object... main menu item;
- follow the instructions of Duplicate Object Wizard.

How can I edit an existing XML schema collection?

XML Schema Collections can be edited within XML Schema Collection Editor. In order to run the editor you should either

- select the XML Schema Collection for editing in the explorer tree (type the first letters of the XML Schema Collection name for quick search);

- select the Edit XML Schema Collection ... item from the popup menu

or

- open the schema in Schema Editor and the XML Schema Collections tab there;
- select the XML Schema Collection to edit;
- press the **Enter** key or select the Edit XML Schema Collection item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

You can change the name of the XML Schema Collection using the Rename XML Schema Collection dialog. To open the dialog you should either

- select the XML Schema Collection to rename in the explorer tree;
- select the Rename XML Schema Collection item from the popup menu

or

- open the schema in Schema Editor and the XML Schema Collections tab there;
- select the XML Schema Collection to rename;
- select the Rename XML Schema Collection item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

How can I drop a XML schema collection?

To drop an XML Schema Collection:

- select the XML Schema Collection to drop in the explorer tree;
- select the Drop XML Schema Collection item from the popup menu

or

- open the schema in Schema Editor and the XML Schema Collections tab there;
- select the XML Schema Collection to drop;
- press the **Delete** key or select the Drop XML Schema Collection item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.

5.14.1 Create XML Schema Collection Wizard

New XML schema collections are created within Create XML Schema Collection Wizard. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the XML Schema Collection icon in the Create Database Object dialog

or

- select the XML Schema Collections list or any object from that list in the explorer

  tree;
- select the Create New XML Schema Collection... item from the popup menu

or

- open the schema in Schema Editor and the XML Schema Collections tab there;
- press the **Insert** key or select the Create New XML Schema Collection... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

To create a new XML schema collection with the same properties as one of the existing XML schema collections has:

- select the Object | Duplicate Database Object... main menu item;
- follow the instructions of Duplicate Object Wizard.

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

To add a new XML Schema Collection, specify the following properties.

Owner
Defines the owner of the XML Schema Collection.

Comment
Specify a comment to the XML Schema Collection.

Definition
Is a string constant or scalar variable. Is varchar, varbinary, nvarchar, or xml type. You need not to set the definition on XML Schema Collection on creation. It is an optional wizard step. You can do it later using a non-modal editor.

See also: XML Schema Collection Editor 171

5.14.2 XML Schema Collection Editor

XML Schema Collections can be edited within XML Schema Collection Editor. In order to run the editor you should either

- select the XML Schema Collection for editing in the explorer tree (type the first letters of the XML Schema Collection name for quick search);
- select the Edit XML Schema Collection ... item from the popup menu

or

- open the schema in Schema Editor and the XML Schema Collections tab there;
- select the XML Schema Collection to edit;
- press the **Enter** key or select the Edit XML Schema Collection item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

You can change the name of the XML Schema Collection using the Rename XML Schema Collection dialog. To open the dialog you should either

- select the XML Schema Collection to rename in the explorer tree;
- select the Rename XML Schema Collection item from the popup menu

or

- open the schema in Schema Editor and the XML Schema Collections tab there;
- select the XML Schema Collection to rename;
- select the Rename XML Schema Collection item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.

Name
Defines the name of the XML Schema Collection.

Owner
Defines the owner of the XML Schema Collection. By default, only the owner of an object can perform various operations with the object. In order to allow other users to operate it, privileges must be granted. (However, users that have the superuser attribute can always access any object.)

Comment
Specify a comment to the XML Schema Collection.

Definition
Is a string constant or scalar variable. Is varchar, varbinary, nvarchar, or xml type.

See also: Create XML Schema Collection Wizard 170

5.15 Queues

MS SQL Maestro supports *Service Broker* communications designed around reliable, asynchronous message delivery. **Queues** store messages. When Service Broker receives a message for a service, Service Broker inserts the message into the queue for that service. To get messages sent to the service, an application receives messages from the queue. Service Broker manages queues and presents a view of a queue that is similar to a table.

Each service is associated with one queue. When a message arrives for a service, Service Broker places the message in the queue associated with that service.

Each message is a row in the queue. The row contains the content of the message as well as information about the message type, the service targeted by the message, the contract that the message follows, the validation performed on the message, the conversation that the message is a part of, and information internal to the queue. An application uses the information in the message row to identify each message uniquely and process the message appropriately.

⊟ How can I create a new queue?

Create Queue Wizard 174 guides you through the process of creating a new queue. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the Queues icon in the Create Database Object dialog

or

- select the Queues list or any object from that list in the explorer tree;
- select the Create Queue ... item from the popup menu

or

- open the schema in Schema Editor and the Queues tab there;
- press the **Insert** key or select the Create New Queue ... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

To create a new queue with the same properties as one of the existing one has:

- select the Object | Duplicate Database Object... main menu item;
- follow the instructions of Duplicate Object Wizard.

⊟ How can I edit an existing queue?

Queue Editor 175 allows you to change queue status, retention, and also activation options. In order to run the editor you should either

- select the queue for editing in the explorer tree (type the first letters of the queue name for quick search);
- select the Edit Queue ... item from the popup menu

or

- open the schema in Schema Editor and the Queues tab there;
- select the queue to edit;
- press the **Enter** key or select the Edit Queue item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

⊟ How can I drop a queue?

To drop a queue:

- select the queue to drop in the explorer tree;
- select the Drop Queue item from the popup menu

and confirm dropping in the dialog window.

5.15.1 Create Queue Wizard

Create Queue Wizard 174 guides you through the process of creating a new queue. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the Queues icon in the Create Database Object dialog

or

- select the Queues list or any object from that list in the explorer tree;
- select the Create Queue ... item from the popup menu

or

- open the schema in Schema Editor and the Queues tab there;
- press the **Insert** key or select the Create New Queue ... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

To create a new queue with the same properties as one of the existing one has:

- select the Object | Duplicate Database Object... main menu item;
- follow the instructions of Duplicate Object Wizard.

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

Queue Properties

Name
Enter the name of the queue to create. This name must meet the guidelines for SQL Server identifiers.

Owner
Defines the owner of the queue.

Status
Specifies whether the queue is available (checked) or unavailable (unchecked). When the queue is unavailable, no messages can be added to the queue or removed from the queue. You can create the queue in an unavailable state in order to keep messages from

arriving on the queue until the queue is made available within Queue Editor 175.

Retention
Specifies the retention setting for the queue. If checked, all messages sent or received on conversations using this queue are retained in the queue until the conversations have ended. This allows you to retain messages for auditing purposes, or to perform compensating transactions if an error occurs.

Activation options (Procedure name, Status, Max readers, Execute As)
Specifies information about the stored procedure to activate to process messages in this queue:

Procedure name
Specifies the name of the stored procedure to activate to process messages in this queue. This value must be a SQL Server identifier.

Status
Specifies whether or not Service Broker activates the stored procedure. When checked, the queue starts the stored procedure specified with Procedure name when the number of procedures currently running is less than Max readers and when messages arrive on the queue faster than the stored procedures receive messages. When unchecked, the queue does not activate the stored procedure.

Max readers
Specifies the maximum number of instances of the activation stored procedure that the queue starts at the same time. The value of readers must be a number between 0 and 32767.

Execute As
Specifies the SQL Server database user account under which the activation stored procedure runs. SQL Server must be able to check the permissions for this user at the time that the queue activates the stored procedure. For an NT domain user, the server must be connected to the domain when the procedure is activated or activation fails. For a SQL Server user, the server can always check permissions.

See also: Queue Editor 175

5.15.2 Queue Editor

Queue Editor allows you to edit queue options. It can be opened when you create a new queue or edit the existing one (see How to edit queue 173 for details).

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.

- Editing queue properties 175
- Managing queue data 176

See also: Create Queue Wizard 174

5.15.2.1 Editing queue properties

Use the Receive messages link on the Navigation bar to open SQL Editor with a corresponding query. Execute the query to receive messages (with their deleting from the queue data).

You can also change queue properties here. The detailed description is given below.

Common options tab represents the name, owner and ID of the queue. By default, only the owner of an object can perform various operations with the object. In order to allow other users to operate it, privileges must be granted. (However, users that have the superuser attribute can always access any object.)

Status
Use the checkbox to specify whether the queue is available (ON) or unavailable (OFF). When the queue is unavailable, no messages can be added to the queue or removed from the queue.

Retention
Specifies the retention setting for the queue. If checked, all messages sent or received on conversations using this queue are retained in the queue until the conversations have ended. This allows you to retain messages for auditing purposes, or to perform compensating transactions if an error occurs

Note: Setting the retention ON can reduce performance. This setting should only be used if required to meet the service level agreement for the application.

Activation options (Procedure name, Status, Max readers, Execute As)
Specifies information about the stored procedure to activate to process messages in this queue:

Procedure name
Specifies the name of the stored procedure to activate to process messages in this queue. This value must be a SQL Server identifier.

Status
Specifies whether or not Service Broker activates the stored procedure. When checked, the queue starts the stored procedure specified with Procedure name when the number of procedures currently running is less than Max readers and when messages arrive on the queue faster than the stored procedures receive messages. When unchecked, the queue does not activate the stored procedure.

Max readers
Use the field to change the maximum number of instances of the activation stored procedure that the queue starts at the same time. The value of readers must be a number between 0 and 32767.

Execute As
Specifies the SQL Server database user account under which the activation stored procedure runs. SQL Server must be able to check the permissions for this user at the time that the queue activates the stored procedure. For an NT domain user, the server must be connected to the domain when the procedure is activated or activation fails. For a SQL Server user, the server can always check permissions.

5.15.2.2 Managing queue data

The Data tab displays the queue messages as a grid (see Data View 328 for details). Use grid’s popup menu to open Data Input Form, to invoke the Export Data, and Get SQL

Dump modules, to set the value of the selected record to *NULL* or to *Now* (for *Date* values). To delete the queue from the grid, use the Receive messages link on the Navigation bar and execute the corresponding query in the SQL Editor. You can find description of the queue columns below.

For your convenience it was implemented two modes of viewing data: as table and as info cards.

The following table lists the columns in a queue.

Column name	**Data type**	**Description**
status	tinyint	Status of the message. For messages returned by the RECEIVE command, the status is always 1. Messages in the queue may contain one of the following values: 0=Received message1=Ready2=Not yet complete3=Retained sent message
queuing_order	bigint	Message order number within the queue.
conversation_group_id	uniqueidentifier	Identifier for the conversation group that this message belongs to.
conversation_handle	uniqueidentifier	Handle for the conversation that this message is part of.
message_sequence_number	bigint	Sequence number of the message within the conversation.
service_name	nvarchar(512)	Name of the service that the conversation is to.
service_id	int	SQL Server object identifier of the service that the conversation is to.
service_contract_name	nvarchar(256)	Name of the contract that the conversation follows.
service_contract_id	int	SQL Server object identifier of the contract that the conversation follows.
message_type_name	nvarchar(256)	Name of the message type that describes the message.
message_type_id	int	SQL Server object identifier of the message type that describes the message.
validation	nchar(2)	Validation used for the message. E=EmptyN=NoneX=XML
message_body	varbinary(MAX)	Content of the message.
message_id	uniqueidentifier	Unique identifier for the message.

5.16 Message types

MS SQL Maestro supports *Service Broker* communications designed around reliable, asynchronous message delivery. Applications that use Service Broker communicate by sending messages to one another as part of a conversation. The participants in a conversation must agree on the name and content of each message. A **message type** object defines a name for a message type and defines the type of data that the message contains. Message types persist in the database where the message type is created. You create an identical message type in each database that participates in a conversation.

Each message type specifies the validation that SQL Server performs for messages of that type. SQL Server can validate that the message contains valid XML, that the message contains XML conforming to a particular schema, or that the message contains no data at all. For arbitrary or binary data, the message type can specify that SQL Server does not validate the content of the message.

Validation is performed when the destination service receives the message. If the content of the message does not match the validation specified, Service Broker returns an error message to the service that sent the message.

How can I create a new message type?

Create Message Type Wizard 180 guides you through the process of creating a new type of Service Broker messages. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the Message Types icon in the Create Database Object dialog

or

- select the Message Types list or any object from that list in the explorer tree;
- select the Create Message Type ... item from the popup menu

or

- open the database in Database Editor and the Message Types tab there;
- press the **Insert** key or select the Create New Message Type ... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

How can I edit an existing message type?

Message Type Editor 180 allows you to change the type options. In order to run the editor you should either

- select the type for editing in the explorer tree (type the first letters of the type name for quick search);
- select the Edit Message Type ... item from the popup menu

or

- open the database in Database Editor and the Message Types

tab there;
- select the type to edit;
- press the **Enter** key or select the Edit Message Type item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

How can I drop a message type?

To drop a message type:

- select the type to drop in the explorer tree;
- select the Drop Message Type item from the popup menu

and confirm dropping in the dialog window.

5.16.1 Create Message Type Wizard

The wizard guides you through the process of creating a new type of Service Broker messages. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the Message Types icon in the Create Database Object dialog

or

- select the Message Types list or any object from that list in the explorer tree;
- select the Create Message Type ... item from the popup menu

or

- open the database in Database Editor and the Message Types tab there;
- press the **Insert** key or select the Create New Message Type ... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

To add a new message type, specify the following options.

See also: Message Type Editor 180

5.16.2 Message Type Editor

Message Type Editor allows you to change the type options. In order to run the editor you should either

- select the type for editing in the explorer tree (type the first letters of the type name for quick search);
- select the Edit Message Type ... item from the popup menu

or

- open the database in Database Editor and the Message Types tab there;
- select the type to edit;
- press the **Enter** key or select the Edit Message Type item from the popup menu

(alternatively, you may use the corresponding link of the Navigation Bar).

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.

- Editing message type properties 181

See also: Create Message Type Wizard 180

5.16.2.1 Editing message type properties

The Name field displays the message type name.

Owner
Sets the owner of the message type to the specified database user or role. When the current user is dbo or sa, the Owner may be the name of any valid user or role. Otherwise, Owner must be the name of the current user, the name of a user that the current user has impersonate permissions for, or the name of a role to which the current user belongs. In case the field is blanck the message type belongs to the current user.

Validation type
Specifies how *Service Broker* validates the message body for messages of this type.

None	No validation is performed. The message body may contain any data, or may be NULL.
Empty	The message body must be NULL.
XML	The message body must contain well-formed XML.

XML Schema Collection
The message body must contain XML that conforms to a schema in the specified schema collection The XML Schema Collection must be the name of an existing XML schema collection.

5.17 Contracts

MS SQL Maestro supports *Service Broker* communications designed around reliable, asynchronous message delivery. A contract is an agreement between two services about which messages each service sends to accomplish a particular task. Contract definitions persist in the database where the type is created. A contract defines which message types an application uses to accomplish a particular task.

You create an identical contract in each database that participates in a conversation. For example, if a human resources application wants to verify an employee ID, the service that requests the verification must know which types of messages the other service expects. The requesting service also must know which types of messages it can expect to receive so that it is prepared to process them.

The contract specifies which message types can be used to accomplish the desired work. The contract also specifies which participant in the conversation can use each message type. Some message types can be sent by either participant; other message types are restricted to be sent only by the initiator or only by the target. A contract must contain a message type sent by the initiator or a message type sent by either participant; otherwise, there is no way for the initiator to begin a conversation that uses the contract.

How can I create a new contract?

Create Contract Wizard 183 guides you through the process of creating a new Service Broker contract. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the Contracts icon in the Create Database Object dialog

or

- select the Contracts list or any object from that list in the explorer tree;
- select the Create Contract ... item from the popup menu

or

- open the database in Database Editor and the Contracts tab there;
- press the **Insert** key or select the Create New Contract ... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

How can I edit an existing contract?

Contract Editor 183 allows you to change contract options. In order to run the editor you should either

- select the contract for editing in the explorer tree (type the first letters of the queue name for quick search);
- select the Edit Contract ... item from the popup menu

or

- open the database in Database Editor and the Contracts tab there;

- select the contract to edit;
- press the **Enter** key or select the Edit Contract item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

How can I drop a contract?

To drop a contract:

- select the contract to drop in the explorer tree;
- select the Drop Contract item from the popup menu

and confirm dropping in the dialog window.

5.17.1 Create Contract Wizard

The wizard guides you through the process of creating a new Service Broker contract. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the Contracts icon in the Create Database Object dialog

or

- select the Contracts list or any object from that list in the explorer tree;
- select the Create Contract ... item from the popup menu

or

- open the database in Database Editor and the Contracts tab there;
- press the **Insert** key or select the Create New Contract ... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

To add a new contract, set Contract Properties 184.

See also: Contract Editor 183

5.17.2 Contract Editor

This editor allows you to manage existing contracts. In order to run the editor you should either

- select the contract for editing in the explorer tree (type the first letters of the queue name for quick search);
- select the Edit Contract ... item from the popup menu

or

- open the database in Database Editor and the Contracts tab there;
- select the contract to edit;
- press the **Enter** key or select the Edit Contract item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.

See also: Contract Properties 184, Create Contract Wizard 183

5.17.3 Contract Properties

Common options tab represents the name, owner and ID of the contract. By default, only the owner of an object can perform various operations with the object. In order to allow other users to operate it, privileges must be granted. (However, users that have the superuser attribute can always access any object.)

Message types (Initiator, Target)
Check names of a message types to be included as part of the contract.

Initiator: Indicates that only the initiator of the conversation can send messages of the specified message type. A service that begins a conversation is referred to as the initiator of the conversation.

Target: Indicates that only the target of the conversation can send messages of the specified message type. A service that accepts a conversation that was started by another service is referred to as the target of the conversation.

5.18 Services

MS SQL Maestro supports *Service Broker* communications designed around reliable, asynchronous message delivery. A **service** exposes the functionality provided by contracts with which it is associated, so that they can be used by other services.

Conversations initiated from this service may use any contract. You create a service without specifying contracts when the service only initiates conversations.

When *Service Broker* accepts a new conversation from a remote service, the name of the target service determines the queue where the broker places messages in the conversation.

How can I create a new service?

Create Service Wizard 186 guides you through the process of creating a new *Service Broker* service. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the Services icon in the Create Database Object dialog

or

- select the Services list or any object from that list in the explorer tree;
- select the Create Service ... item from the popup menu

or

- open the database in Database Editor and the Services tab there;
- press the **Insert** key or select the Create New Service ... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

How can I edit an existing service?

Service Editor 186 allows you to change common service options. In order to run the editor you should either

- select the service for editing in the explorer tree (type the first letters of the service name for quick search);
- select the Edit Service ... item from the popup menu

or

- open the database in Database Editor and the Services tab there;
- select the service to edit;
- press the **Enter** key or select the Edit Service item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

How can I drop a service?

To drop a service:

- select the service to drop in the explorer tree;
- select the Drop Service item from the popup menu

and confirm dropping in the dialog window.

5.18.1 Create Service Wizard

The wizard guides you through the process of creating a new *Service Broker* service. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the Services icon in the Create Database Object dialog

or

- select the Services list or any object from that list in the explorer tree;
- select the Create Service ... item from the popup menu

or

- open the database in Database Editor and the Services tab there;
- press the **Insert** key or select the Create New Service ... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

Name
Enter the name of the service to create. The name must be a valid *sysname*.

Owner
Sets the owner of the service to the specified database user or role. When the current user is dbo or sa, owner may be the name of any valid user or role. Otherwise, owner must be the name of the current user, the name of a user that the current user has impersonate permissions for, or the name of a role to which the current user belongs.

Queue
Specifies the queue that receives messages for the service. The queue must exist in the same database as the service.

Contracts
Specify contracts for which this service may be a target. Service programs initiate conversations to this service using the contracts specified. If no contracts are specified, the service may only initiate conversations.

See also: Service Editor 186

5.18.2 Service Editor

This editor allows you to change common service options. In order to run the editor you should either

- select the service for editing in the explorer tree (type the first letters of the service name for quick search);
- select the Edit Service ... item from the popup menu

or

- open the database in Database Editor and the Services tab there;
- select the service to edit;
- press the **Enter** key or select the Edit Service item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.

See also: Service Properties 187, Create Service Wizard 186

5.18.2.1 Editing service properties

Common options tab represents the name, owner and ID of the queue. By default, only the owner of an object can perform various operations with the object. In order to allow other users to operate it, privileges must be granted. (However, users that have the superuser attribute can always access any object.)

Queue
Specifies the queue that receives messages for the service. The queue must exist in the same database as the service.

Contracts
Specify contracts for which this service may be a target. Service programs initiate conversations to this service using the contracts specified. If no contracts are specified, the service may only initiate conversations.

5.19 Conversations

MS SQL Maestro supports *Service Broker* communications designed around reliable, asynchronous message delivery. All messages sent by *Service Broker* are part of a conversation. A **dialog conversation**, or dialog, is a conversation between two services. In effect, a dialog is a reliable, persistent bidirectional stream of messages between two services.

Dialogs provide exactly-once-in-order (EOIO) message delivery. Dialogs use the conversation identifier and sequence numbers that are contained in each message to identify related messages and deliver messages in the correct order. A dialog is a reliable, persistent stream of messages between two services.

A dialog conversation has two participants. The initiator begins the conversation. The target accepts a conversation begun by the initiator. Whether a participant begins the conversation determines the messages that the participant can send, as specified in the contract for the conversation.

How can I create a new dialog conversation?

Create Conversation Wizard 189 guides you through the process of creating a new *Service Broker* conversation. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the Conversations icon in the Create Database Object dialog

or

- select the Conversations list or any object from that list in the explorer tree;
- select the Create Conversation ... item from the popup menu

or

- open the database in Database Editor and the Conversations tab there;
- press the **Insert** key or select the Create New Conversation ... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

How can I edit an existing conversation?

Conversation Editor 190 allows you to change common conversation options. In order to run the editor you should either

- select the conversation for editing in the explorer tree (type the first letters of the conversation name for quick search);
- select the Edit Conversation ... item from the popup menu

or

- open the database in Database Editor and the Conversations tab there;
- select the conversation to edit;
- press the **Enter** key or select the Edit Conversation item from the popup menu (alternatively, you may use the corresponding

link of the Navigation Bar).

How can I drop a conversation?

To drop a conversation:

- select the conversation to drop in the explorer tree;
- select the Drop Conversation item from the popup menu

and confirm dropping in the dialog window.

5.19.1 Create Conversation Wizard

The wizard guides you through the process of creating a new *Service Broker* conversation. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the Conversations icon in the Create Database Object dialog

or

- select the Conversations list or any object from that list in the explorer tree;
- select the Create Conversation ... item from the popup menu

or

- open the database in Database Editor and the Conversations tab there;
- press the **Insert** key or select the Create New Conversation ... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

Conversation options

Initiator service
Specifies the service that initiates the dialog. The name specified must be the name of a service in the current database. The queue specified for the initiator service receives messages returned by the target service and messages created by *Service Broker* for this conversation.

Target service
Specifies the target service with which to initiate the dialog. The target_service_name is of type nvarchar(256). *Service Broker* uses a byte-by-byte comparison to match the target_service_name string. In other words, the comparison is case-sensitive and does not take into account the current collation.

Contract
Specifies the contract that this conversation follows. The contract must exist in the current database. If the target service does not accept new conversations on the contract specified, Service Broker returns an error message on the conversation.

Related conversation
Specifies the existing conversation group that the new dialog is added to. When this clause is present, the new dialog belongs to the same conversation group as the dialog specified by related conversation. The conversation must be of a type implicitly

convertible to type uniqueidentifier.

Related conversation group
Specifies the existing conversation group that the new dialog is added to. When this clause is present, the new dialog will be added to the conversation group specified by related conversation group.

Life time
Specifies the maximum amount of time the dialog will remain open. For the dialog to complete successfully, both endpoints must explicitly end the dialog before the lifetime expires. The lifetime value must be expressed in seconds. Lifetime is of type int.

Encryption
Specifies whether or not messages sent and received on this dialog must be encrypted when they are sent outside of a Microsoft SQL Server instance. A dialog that must be encrypted is a secured dialog. When unchecked and the certificates required to support encryption are not configured, *Service Broker* returns an error message on the conversation. If checked, encryption is used if a remote service binding is configured for the target service; otherwise messages are sent unencrypted.

Messages are the information exchanged between applications that use *Service Broker*.

Each message is part of a conversation. A message has a specific type, which is determined by the application that sends the message. Each message has a unique conversation identity, as well as a sequence number within the conversation. When receiving messages, *Service Broker* uses the conversation identity and the sequence number of the message to enforce message ordering.

The content of the message is determined by the application. When a message is received, *Service Broker* validates the content of the message to ensure that the content is valid for the message type. Regardless of the message type, SQL Server stores the content of the message as type varbinary(max). Therefore, a message can contain any data that can be converted to varbinary(max).

See also: Conversation Editor 190

5.19.2 Conversation Editor

This editor allows you to change common conversation options. In order to run the editor you should either

- select the conversation for editing in the explorer tree (type the first letters of the conversation name for quick search);
- select the Edit Conversation ... item from the popup menu

or

- open the database in Database Editor and the Conversations tab there;
- select the conversation to edit;
- press the **Enter** key or select the Edit Conversation item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.

See also: Conversation properties [191], Create Conversation Wizard [189]

5.19.2.1 Editing conversation properties

Initiator service
Specifies the service that initiates the dialog. The name specified must be the name of a service in the current database. The queue specified for the initiator service receives messages returned by the target service and messages created by *Service Broker* for this conversation.

Target service
Specifies the target service with which to initiate the dialog. The target_service_name is of type nvarchar(256). *Service Broker* uses a byte-by-byte comparison to match the target_service_name string. In other words, the comparison is case-sensitive and does not take into account the current collation.

Contract
Specifies the contract that this conversation follows. The contract must exist in the current database. If the target service does not accept new conversations on the contract specified, Service Broker returns an error message on the conversation.

Related conversation
Specifies the existing conversation group that the new dialog is added to. When this clause is present, the new dialog belongs to the same conversation group as the dialog specified by related conversation. The conversation must be of a type implicitly convertible to type uniqueidentifier.

Related conversation group
Specifies the existing conversation group that the new dialog is added to. When this clause is present, the new dialog will be added to the conversation group specified by related conversation group.

Life time
Specifies the maximum amount of time the dialog will remain open. For the dialog to complete successfully, both endpoints must explicitly end the dialog before the lifetime expires. The lifetime value must be expressed in seconds. Lifetime is of type int.

Encryption
Specifies whether or not messages sent and received on this dialog must be encrypted when they are sent outside of a Microsoft SQL Server instance. A dialog that must be encrypted is a secured dialog. When unchecked and the certificates required to support encryption are not configured, *Service Broker* returns an error message on the conversation. If checked, encryption is used if a remote service binding is configured for the target service; otherwise messages are sent unencrypted.

5.20 Aggregates

Like most other relational database products, Microsoft SQL supports aggregate functions. An aggregate function computes a single result from multiple input rows. Some basic and commonly-used aggregate functions are included with the distribution. If one defines new types or needs an aggregate function not already provided, then an Aggregate can be used to provide the stated features.

⊟ How can I create a new aggregate?

New aggregates are created within Create Aggregate Wizard 193. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the Aggregate icon in the Create Database Object dialog

or

- select the Aggregates list or any object from that list in the explorer tree;
- select the Create New Aggregate... item from the popup menu

or

- open the schema in Schema Editor and the Aggregates tab there;
- press the **Insert** key or select the Create New Aggregate item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

To create a new aggregate with the same properties as one of the existing aggregates has:

- select the Object | Duplicate Database Object... main menu item;
- follow the instructions of Duplicate Object Wizard.

⊟ How can I edit an existing aggregate?

Aggregates can be edited within Aggregate Editor 194. In order to run the editor you should either

- select the aggregate for editing in the explorer tree (type the first letters of the aggregate name for quick search);
- select the Edit Aggregate ... item from the popup menu

or

- open the schema in Schema Editor and the Aggregates tab there;
- select the aggregate to edit;
- press the **Enter** key or select the Edit Aggregate item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

You can change the name of the aggregate using the Rename Aggregate dialog. To open the dialog you should either

- select the aggregate to rename in the explorer tree;
- select the Rename Aggregate item from the popup menu

or

- open the schema in Schema Editor and the Aggregates tab there;
- select the aggregate to rename;
- select the Rename Aggregate item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

⊟ How can I drop an aggregate?

To drop an aggregate:

- select the aggregate to drop in the explorer tree;
- select the Drop Aggregate item from the popup menu

or

- open the schema in Schema Editor and the Aggregates tab there;
- select the aggregate to drop;
- press the **Delete** key or select the Drop Aggregate item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.

5.20.1 Create Aggregate Wizard

New aggregates are created within Create Aggregate Wizard 193. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the Aggregate icon in the Create Database Object dialog

or

- select the Aggregates list or any object from that list in the explorer tree;
- select the Create New Aggregate... item from the popup menu

or

- open the schema in Schema Editor and the Aggregates tab there;
- press the **Insert** key or select the Create New Aggregate item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

To create a new aggregate with the same properties as one of the existing aggregates has:

- select the Object | Duplicate Database Object... main menu item;
- follow the instructions of Duplicate Object Wizard.

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

Several options can be specified for the aggregate function that govern the expected behaviour of the function. The detailed description is given below.

Owner
Defines the owner of the new aggregate. By default, only the owner of an object can perform various operations with the object. In order to allow other users to operate it, privileges must be granted. (However, users that have the superuser attribute can always access any object.)

Comment
The box allows you to set optional text to describe the new aggregate.

Parameter Name
The field represents a parameter in the user-defined aggregate. The value of the parameter must be supplied by the user when the aggregate function is executed.

Input Type
Defines the type of parameter.

Return Type
Defines the return type of the aggregate function.

CLR
Specifies the assembly 214 to bind with the user-defined aggregate function and the name of the class in the assembly that implements the user-defined aggregate. The assembly is to have been already created in the database.

See also: Aggregate Editor 194

5.20.2 Aggregate Editor

Aggregates can be edited within Aggregate Editor 194. In order to run the editor you should either

- select the aggregate for editing in the explorer tree (type the first letters of the aggregate name for quick search);
- select the Edit Aggregate ... item from the popup menu

or

- open the schema in Schema Editor and the Aggregates tab there;
- select the aggregate to edit;
- press the **Enter** key or select the Edit Aggregate item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

You can change the name of the aggregate using the Rename Aggregate dialog. To open the dialog you should either

- select the aggregate to rename in the explorer tree;
- select the Rename Aggregate item from the popup menu

or

- open the schema in Schema Editor and the Aggregates tab there;
- select the aggregate to rename;
- select the Rename Aggregate item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.

- Editing aggregate properties 195

See also: Create Aggregate Wizard 193

5.20.2.1 Editing aggregate properties

Aggregate Editor provides you with an ability to edit aggregate properties. The Properties tab allows you to change the aggregate name, the aggregate owner and the comment for the aggregate.

Name
Here you can change the aggregate name.

Owner
Set the owner for the aggregate. By default, only the owner of an object can perform various operations with the object. In order to allow other users to operate it, privileges must be granted. (However, users that have the superuser attribute can always access any object.)

Comment
This field stores a comment to the aggregate.

Parameter Name
The field stores a parameter in the user-defined aggregate. The value of the parameter must be supplied by the user when the aggregate function is executed.

Input Type
Defines the type of parameter.

Return Type
Defines the return type of the aggregate function.

CLR
Specifies the assembly to bind with the user-defined aggregate function and the name of the class in the assembly that implements the user-defined aggregate. The assembly is to have been already created in the database.

See also: Assemblies 214

To apply the changes, select the Apply Changes item in the Navigation bar or use **Ctrl+F9** or **Ctrl+F7** shortcut keys.

It is also possible to modify object properties without opening the object editor: use the Object Properties item of the popup menu of the selected object from the explorer tree.

5.21 Sequences

A sequence is a user-defined schema-bound object that generates a sequence of numeric values according to the specification with which the sequence was created. The sequence of numeric values is generated in an ascending or descending order at a defined interval and may cycle (repeat) as requested. Sequences, unlike identity columns, are not associated with tables. An application refers to a sequence object to receive its next value. The relationship between sequences and tables is controlled by the application. User applications can reference a sequence object and coordinate the values keys across multiple rows and tables.

How can I create a new sequence?

New sequences are created within Create Sequence Wizard 197. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the Sequence icon in the Create Database Object dialog

or

- select the Sequences list or any object from that list in the explorer tree;
- select the Create New Sequence... item from the popup menu

or

- open the schema in Schema Editor and the Sequences tab there;
- press the **Insert** key or select the Create New Sequence item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

To create a new sequence with the same properties as one of the existing sequences has:

- select the Object | Duplicate Database Object... main menu item;
- follow the instructions of Duplicate Object Wizard.

How can I edit an existing sequence?

Sequences can be edited within Sequence Editor 198. In order to run the editor you should either

- select the sequence for editing in the explorer tree (type the first letters of the sequence name for quick search);
- select the Edit Sequence ... item from the popup menu

or

- open the schema in Schema Editor and the Sequences tab there;
- select the sequence to edit;
- press the **Enter** key or select the Edit Sequence item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

How can I drop a sequence

To drop a sequence:

- select the sequence to drop in the explorer tree;
- select the Drop Sequence item from the popup menu

or

- open the schema in Schema Editor and the Sequences tab there;
- select the sequence to drop;
- press the **Delete** key or select the Drop Sequence item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.

5.21.1 Create Sequence Wizard

Sequences may be created with Create Sequence Wizard. Just specify the wizard options according to your needs.

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

Name
The field allows you to specify the new sequence name being set on the previous wizard step.

Owner
Defines the owner of the new sequence.

Comment
The box allows you to set optional text to describe the new sequence.

Value
Specify the starting sequence value here. The default starting value is *minvalue* for ascending sequences and *maxvalue* for descending ones.

Increment by
Specify which value is added to the current sequence value to create a new value. A positive value will make an ascending sequence, a negative one a descending sequence. The default value is 1.

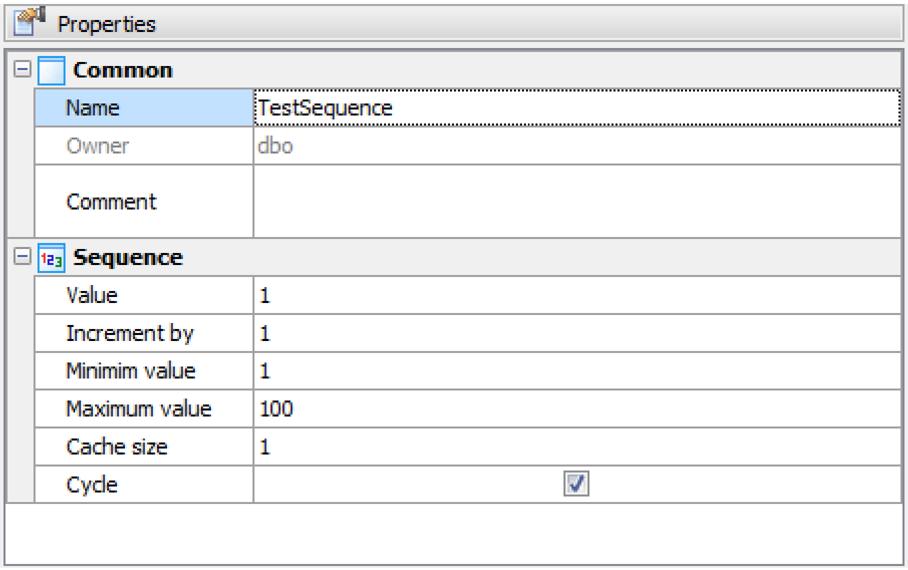
Maximum Value
Determine the maximum value for the sequence. If this clause is not supplied or NO MAXVALUE is specified, then default values will be used. The defaults are 2^63-1 and -1 for ascending and descending sequences, respectively.

Minimum Value
Determine the minimum value a sequence can generate. If this clause is not supplied or NO MINVALUE is specified, then defaults will be used. The defaults are 1 and -263-1 for ascending and descending sequences, respectively.

Cashe

Specify how many sequence numbers are to be preallocated and stored in memory for faster access. The minimum value is 1 (only one value can be generated at a time, i.e., no cache), and this is also the default.

☑ Cycle
The CYCLE option allows the sequence to wrap around when the *maxvalue* or *minvalue* has been reached by an ascending or descending sequence respectively. If the limit is reached, the next number generated will be the minvalue or maxvalue, respectively.

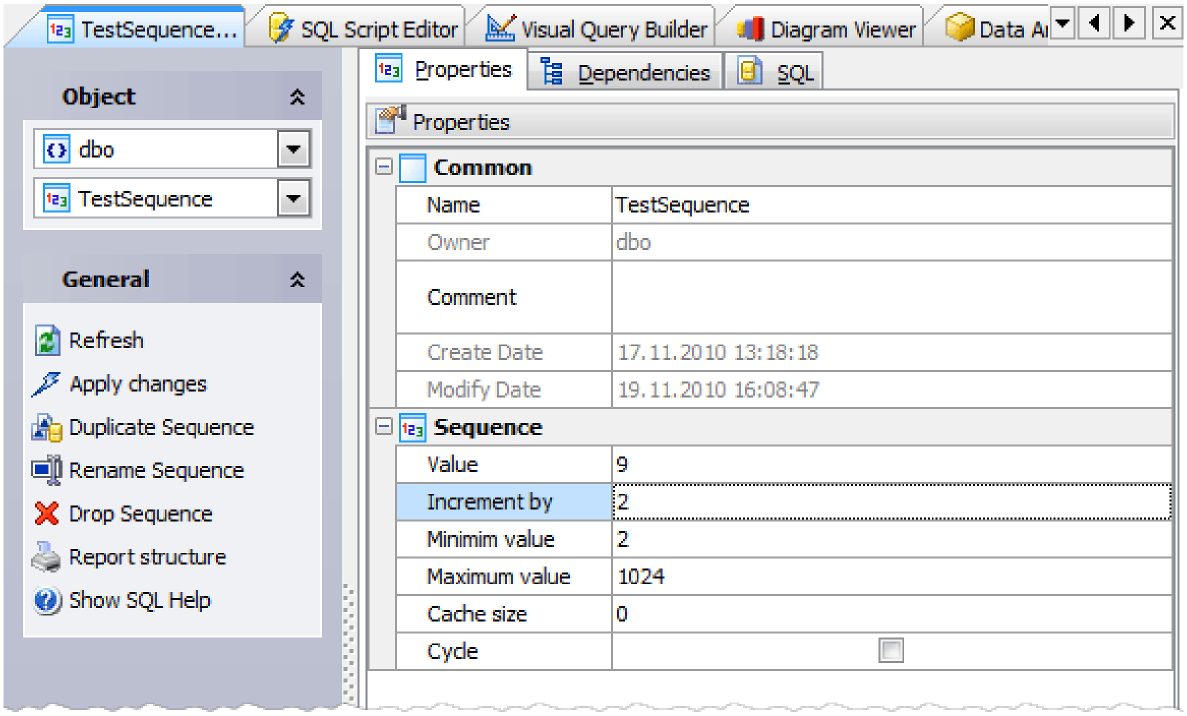


Common	
Name	TestSequence
Owner	dbo
Comment	
Sequence	
Value	1
Increment by	1
Minimim value	1
Maximum value	100
Cache size	1
Cycle	☑

5.21.2 Sequence Editor

Use Sequence Editor to change properties of existing sequences. The editor can be opened automatically after the sequence is created 197 or from the Explorer Tree and Object Manager.

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.



Object: dbo, TestSequence

Common	
Name	TestSequence
Owner	dbo
Comment	
Create Date	17.11.2010 13:18:18
Modify Date	19.11.2010 16:08:47
Sequence	
Value	9
Increment by	2
Minimim value	2
Maximum value	1024
Cache size	0
Cycle	☐

Name
Here you can rename the sequence.

Owner
Shows the owner of the sequence. By default, only the owner of an object can perform various operations with the object. In order to allow other users to operate it, privileges must be granted. (However, users that have the superuser attribute can always access any object.)

Comment
This field stores a comment to the sequence.

Created
The field represents the date the sequence was created.

OID
This field contains the sequence OID (object identifier). This is a serial number that is automatically added by PostgreSQL to all sequences.

The Value contains the starting sequence value here.

Increment by
Here you can edit the value which is added to the current sequence value to create a new value.

The Maximum Value and the Minimum Value contain the maximum and the minimum values for the sequence.

Cashe
Specify how many sequence numbers are to be preallocated and stored in memory for

faster access.

☑ Cycle
The checkbox represents whether the sequence is cycle.

To apply the changes, select the Apply Changes item in the Navigation bar or use **Ctrl+F9** or **Ctrl+F7** shortcut keys.

It is also possible to modify object properties without opening the object editor: use the Object Properties item of the popup menu of the selected object from the explorer tree.

5.22 Users

A user passes through two stages of security when working in Microsoft® SQL Server™: authentication and authorization (permissions validation). The authentication stage identifies the user using a login account 245 and verifies only the ability to connect to an instance of SQL Server. If authentication is successful, the user connects to an instance of SQL Server. The user then needs permissions to access databases on the server performed by *granting access* to an account in each database, mapped to the user login.

Every database cluster contains a set of database users. Those users are separate from the users managed by the operating system under which the server runs. Users own database objects (for example, *tables*) and can assign privileges on those objects to other users to control who has access to which object.

Note: Use of roles 204 simplifies security administration in databases with a large number of users or with a complex security system.

5.22.1 Create User Wizard

Create User Wizard guides you through the process of creating a new database user. See How To Create User 201 to learn how to run the wizard.

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

Specify the Name by which the user is identified inside this database.

Note: the name of the object must be unique among all the object names in its container. Moreover, all the objects that are source of data need unique names among themselves. You can use any identifier that is allowed by Microsoft SQL server.

Comment
This field stores a comment to the user.

Type
Indicates the user type. Possible variants are: *for login*, *certificate* and *asymmetric key*. Indicates the respective database object type for which the database user is being created.

Login
For the *login* user type only. Specifies the SQL Server login for which the database user is being created.

Default Schema
For the *login* user type only. Specifies the first schema that will be searched by the server when it resolves the names of objects for this database user.

Certificate
For the *certificate* user type only. Specifies the certificate for which the database user

is being created.

Asymmetric key
For the *asymmetric key* user type only. Specifies the asymmetric key for which the database user is being created.

See also: User Editor 202

5.22.2 User Editor

User Editor allows you to edit user properties, permissions, grants, and manage user's objects (see How to edit database user 201 for details).

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.

This editor consists of several tabs: the Properties 202 tab allows to browse user options and the Objects tab allows to browse database objects owned by the user.

See also: Create User Wizard 201

5.22.2.1 Editing user properties

The Properties tab allows you to change the user name, the user password, the user's ability to create databases and users, the group list the user belongs to.

The roles list represents all the database roles. To grant/revoke role privileges use the checkboxes in the Granted column. The Grantor and Time columns displays the database administrators that granted the role to the user and the time of the operation respectively.

Name
Displays the user name.

Comment
This field stores a comment to the user.

Create Date
Indicates the date when the user was created.

Modify Date
Indicates the date when the user was last modified.

Type
Defines the user type. Possible variants are: *for login*, *certificate* and *asymmetric key*. Indicates the respective database object type for which the database user has been created.

Login
For the *login* user type only. Specifies the SQL Server login for which the database user is created.

Default Schema
For the *login* user type only. Specifies the first schema that will be searched by the server when it resolves the names of objects for this database user.

Certificate
For the *certificate* user type only. Specifies the certificate for which the database user is created.

Asymmetric key
For the *asymmetric key* user type only. Specifies the asymmetric key for which the database user is created.

Roles
Add or remove a database role for the current user with the help of the corresponding checkboxes.

Database permissions
Adds or removes a database permission for the current user with the help of the the respective checkboxes.

To apply the changes, select the Apply Changes item in the Navigation bar or use **Ctrl+F9** or **Ctrl+F7** shortcut keys.

It is also possible to modify object properties without opening the object editor: use the Object Properties item of the popup menu of the selected object from the explorer tree.

5.23 Roles

Principals are individuals, groups and processes that can request SQL Server resources. Like other components of the SQL Server authorization model, principals can be arranged in a hierarchy. The scope of influence of a principal depends on the scope of the definition of the principal: Windows, server, database; and whether the principal is indivisible or a collection. A Windows Login is an example of an indivisible principal, and a Windows Group is an example of a principal that is a collection. Every principal has a unique security identifier (SID).
Windows-level principals

- Windows Domain Login
- Windows Local Login

SQL Server-level principal

- SQL Server Login

Database-level principals

- Database User
- Database Role
- Application Role Using roles can simplify security administration in databases with a large number of users or with a complex security system.

A database role is created as a separate object, and applies only to the database in which that role is created. Microsoft® SQL Server™ allows Microsoft Windows NT® 4.0 or Windows® 2000 users and groups, SQL Server users, and SQL Server database roles to be members of other roles.

⊟ How can I create a new role?

New roles are created within Create Role Wizard 205. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the Role icon in the Create Database Object dialog

or

- select the Roles list or any object from that list in the explorer tree;
- select the Create New Role... item from the popup menu

or

- open the database in Database Editor and the Roles tab there;
- press the **Insert** key or select the Create New Role... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

To create a new role with the same properties as one of the existing roles has:

- select the Object | Duplicate Database Object... main menu item;

- follow the instructions of Duplicate Object Wizard.

⊟ How can I edit an existing role?

Roles can be edited within Role Editor 206. In order to run the editor you should either

- select the role for editing in the explorer tree (type the first letters of the role name for quick search);
- select the Edit Role ... item from the popup menu

or

- open the database in Database Editor and the Roles tab there;
- select the role to edit;
- press the **Enter** key or select the Edit Role item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

You can change the name of the role using the Rename Role dialog. To open the dialog you should either

- select the role to rename in the explorer tree;
- select the Rename Role item from the popup menu

or

- open the database in Database Editor and the Roles tab there;
- select the role to rename;
- select the Rename Role item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

⊟ How can I drop a role?

To drop a role (note that you can drop database roles only):

- select the role to drop in the explorer tree;
- select the Drop Role item from the popup menu

or

- open the database in Database Editor and the Roles tab there;
- select the role to drop;
- press the **Delete** key or select the Drop Role item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.

5.23.1 Create Role Wizard

Create Role Wizard guides you through the process of creating a new database role. See How To Create role 204 to learn how to run this wizard.

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

Specify role options according to your needs. The detailed description is given below.

Name
The field contains new role name as it was set on the previous step.

Owner
The database user or role that is to own the new role. By default, only the owner of an object can perform various operations with the object. In order to allow other users to operate it, privileges must be granted. (However, users that have the superuser attribute can always access any object.)

Role Type
Defines the role type. Role can be *Standard* or *Application*. Unlike standard roles, A *pplication Roles* allow the application, rather than SQL Server, to take over the responsibility of user authentication. However, because SQL Server still must authenticate the application when it accesses databases, the application must provide a password because there is no other way to authenticate an application.

Password
For Application roles only. Specifies the password that database users will use to activate the application role.

Default Schema
For Application roles only. Specifies the first schema that will be searched by the server when it resolves the names of objects for this role.

Managing role members
The wizard step allows to define users and roles to be the new role members, to add the role being created to another database roles, and to grant new role permissions. Just open the corresponding tab and check the appropriate boxes.

5.23.2 Role Editor

Role Editor allows you to edit role properties and permissions. It can be open automatically after the role is created and is available on editing (see How to edit role 204 for details).

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.

The Properties tab allows you to change the role name, the role password and set statement permissions.

Name
The field allows you to view and modify the role name.

Create Date
Displays the date when the role was created.

Modify Date
Displays the date when the role was last modified.

Role Type

Defines the role type. Role can be *standard* or *application*. Unlike standard roles, *application roles* allow the application, rather than SQL Server, to take over the responsibility of user authentication. However, because SQL Server still must authenticate the application when it accesses databases, the application must provide a password because there is no other way to authenticate an application.

Password

For Application roles only. Specifies the password that database users will use to activate the application role.

Default Schema

For Application roles only. Specifies the first schema that will be searched by the server when it resolves the names of objects for this role.

Statement permissions

You can set statement permissions to the role by selecting essential checkboxes.

To apply the changes, select the Apply Changes item in the Navigation bar or use **Ctrl+F9** or **Ctrl+F7** shortcut keys.

It is also possible to modify object properties without opening the object editor: use the Object Properties item of the popup menu of the selected object from the explorer tree.

5.24 Files

Microsoft SQL Server maps a database over a set of operating system files. Data and log information are never mixed in the same file, and individual files are used only by one database.

How can I create a new file?

New files are created within the File Editor dialog window. In order to open the dialog you should either

- select the Object | Create Database Object... main menu item;
- select the File icon in the Create Database Object dialog

or

- select the Files list or any object from that list in the explorer tree;
- select the Create New File... item from the popup menu

or

- open the database in Database Editor and the Files tab there;
- press the **Insert** key or select the Create New File... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

To create a new file with the same properties as one of the existing files has:

- select the Object | Duplicate Database Object... main menu item;
- follow the instructions of Duplicate Object Wizard.

How can I edit an existing file?

Files can be edited within File Editor. In order to run the editor you should either

- select the file for editing in the explorer tree (type the first letters of the file name for quick search);
- select the Edit File ... item from the popup menu

or

- open the database in Database Editor and the Files tab there;
- select the file to edit;
- press the **Enter** key or select the Edit File item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

You can change the name of the file using the Rename File dialog. To open the dialog you should either

- select the file to rename in the explorer tree;
- select the Rename File item from the popup menu

or

- open the database in Database Editor and the Files tab there;

- select the file to rename;
- select the Rename File item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

How can I drop a file?

To drop a file:

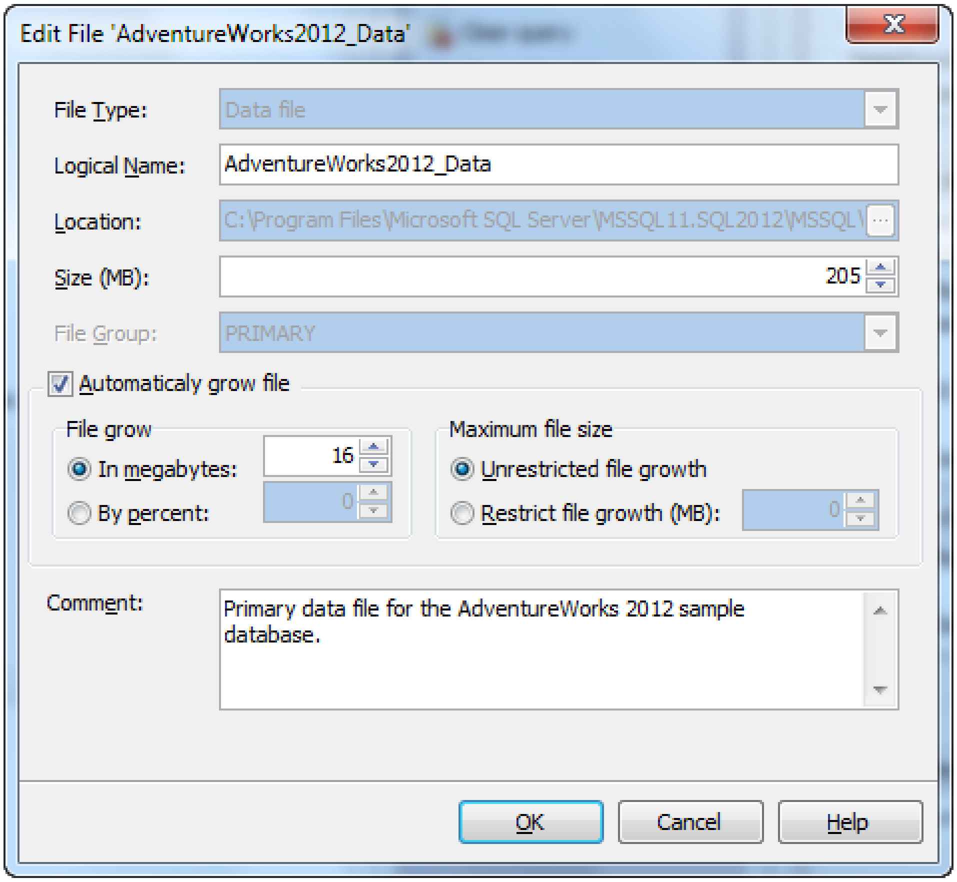
- select the file to drop in the explorer tree;
- select the Drop File item from the popup menu

or

- open the database in Database Editor and the Files tab there;
- select the file to drop;
- press the **Delete** key or select the Drop File item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.

Files are created and edited within the File Editor window.



Edit File 'AdventureWorks2012_Data'

File Type: Data file

Logical Name: AdventureWorks2012_Data

Location: C:\Program Files\Microsoft SQL Server\MSSQL11.SQL2012\MSSQL\

Size (MB): 205

File Group: PRIMARY

☑ Automaticaly grow file

File grow

● In megabytes: 16

○ By percent: 0

Maximum file size

● Unrestricted file growth

○ Restrict file growth (MB): 0

Comment: Primary data file for the AdventureWorks 2012 sample database.

File Type
Specifies the file type.

File Name
The field contains the logical name used in SQL Server when referencing the file.

Location
Displays the operating system (physical) file name.

Size
Indicates the file size.

File Group
Specifies the filegroup to which the specified file is added.

File grow
Stands for the automatic growth increment of the file. The FILEGROWTH setting for a file cannot exceed the MAXSIZE setting.

Maximum file size
Specifies the maximum file size to which the file can grow.

Comment
The box allows you to set optional text to describe the file.

5.25 File Groups

File groups are named collections of files and are used to simplify data placement and administrative tasks such as backup and restore operations.

⊟ How can I create a new file group?

File groups are created within Create File Group Wizard 212. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the File Group icon in the Create Database Object dialog

or

- select the File Groups list or any object from that list in the explorer tree;
- select the Create New File Group... item from the popup menu

or

- open the database in Database Editor and the File Groups tab there;
- press the **Insert** key or select the Create New File Group... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

To create a new file group with the same properties as one of the existing file groups has:

- select the Object | Duplicate Database Object... main menu item;
- follow the instructions of Duplicate Object Wizard.

⊟ How can I edit an existing file group?

File groups can be edited within File Group Properties. In order to run the editor you should either

- select the file group for editing in the explorer tree (type the first letters of the file group name for quick search);
- select the Edit File Group ... item from the popup menu

or

- open the database in Database Editor and the File Groups tab there;
- select the file group to edit;
- press the **Enter** key or select the Edit File Group item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

You can change the name of the file group using the Rename File Group dialog. To open the dialog you should either

- select the file group to rename in the explorer tree;
- select the Rename File Group item from the popup menu

or

- open the database in Database Editor and the File Groups tab there;
- select the file group to rename;
- select the Rename File Group item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

How can I drop a file group?

To drop a file group:

- select the file group to drop in the explorer tree;
- select the Drop File Group item from the popup menu

or

- open the database in Database Editor and the File Groups tab there;
- select the file group to drop;
- press the **Delete** key or select the Drop File Group item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.

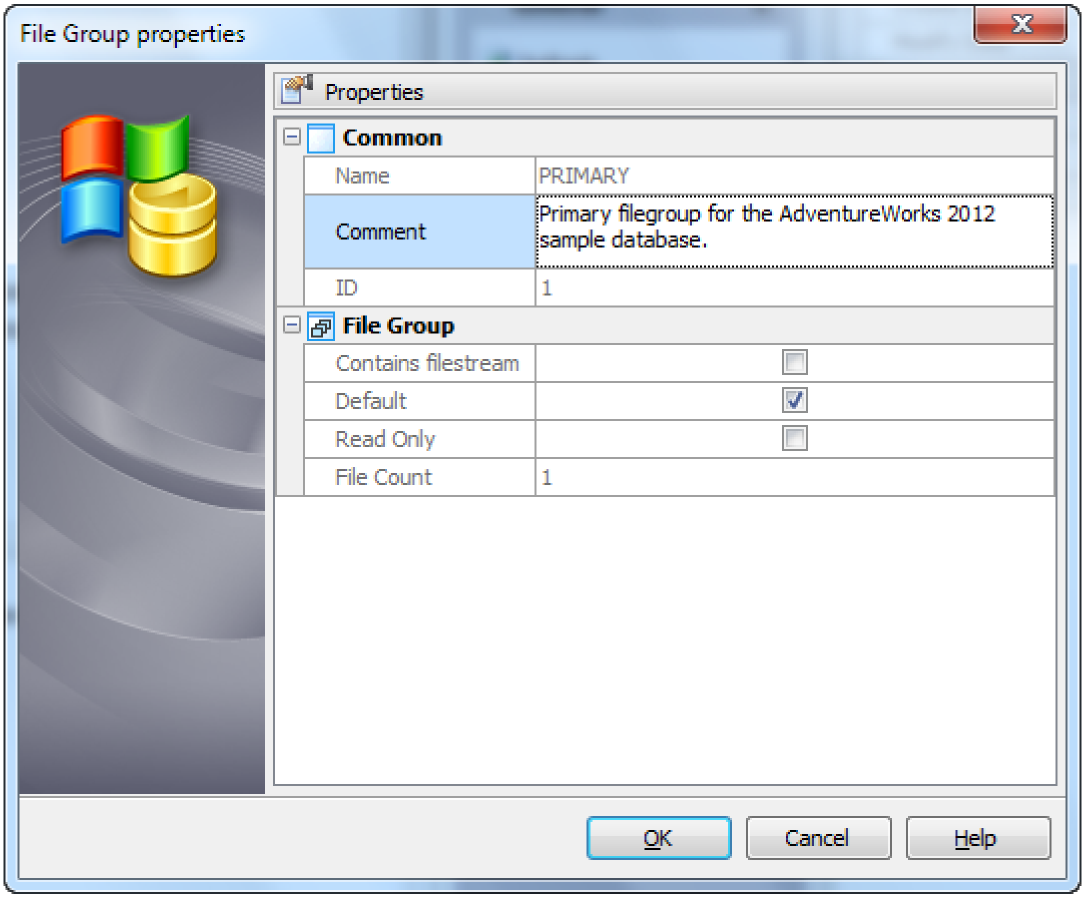
5.25.1 File Group Properties

File Group Properties window allows you to create a new file group or to edit an existing one.

To open this dialog, select the File Group Properties item from the popup menu of the selected object in the explorer tree or double click the object.

On creation you need to set only the new file group Name. The file group name must be unique among the file group names in the database.

Comment
The field allows you to set optional text to describe the new file group.



Common

Name	PRIMARY
Comment	Primary filegroup for the AdventureWorks 2012 sample database.
ID	1

File Group

Contains filestream	☐
Default	☑
Read Only	☐
File Count	1

5.26 Assemblies

Assemblies are managed application modules that contain classes metadata and managed code as an object in an instance of SQL Server. By referencing these modules, common language runtime (CLR) functions, stored procedures, triggers, user-defined aggregates, and user-defined types can be created in the database.

Note: The object is available only for Microsoft SQL server 2005.

⊟ How can I create a new assembly?

New assemblies are created within Create Assembly Wizard 215. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the Assembly icon in the Create Database Object dialog

or

- select the Assemblies list or any object from that list in the explorer tree;
- select the Create New Assembly... item from the popup menu

or

- open the database in Database Editor and the Assemblies tab there;
- press the **Insert** key or select the Create New Assembly... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

To create a new assembly with the same properties as one of the existing assemblies has:

- select the Object | Duplicate Database Object... main menu item;
- follow the instructions of Duplicate Object Wizard.

⊟ How can I edit an existing assembly?

Assemblies can be edited within Assembly Editor 216. In order to open the editor you should either

- select the assembly for editing in the explorer tree (type the first letters of the assembly name for quick search);
- select the Edit Assembly ... item from the popup menu

or

- open the database in Database Editor and the Assemblies tab there;
- select the assembly to edit;
- press the **Enter** key or select the Edit Assembly item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

⊟ How can I drop an assembly?

To drop an assembly:

- select the assembly to drop in the explorer tree;
- select the Drop Assembly item from the popup menu

or

- open the database in Database Editor and the Assemblies tab there;
- select the assembly to drop;
- press the **Delete** key or select the Drop Assembly item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.

5.26.1 Create Assembly Wizard

Create Assembly Wizard guides you through the process of creating a new database assembly. See How To Create assembly 214 to learn how to run this wizard.

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

Assembly options

Name

The field contains the assembly name.

Owner

Specifies the name of a user or role as the owner of the assembly. By default, only the owner of an object can perform various operations with the object. In order to allow other users to operate it, privileges must be granted. (However, users that have the superuser attribute can always access any object.)

Comment

This field stores a comment to the assembly.

Location

Specifies the local path or network location where the assembly that is being uploaded is located, and also the manifest file name that corresponds to the assembly.

Permission Set

Specifies a set of code access permissions that are granted to the assembly when it is accessed by the SQL Server.

Assembly files

On this step of the wizard you can specify the files of the assembly being created. Use **Insert** hot key or pop-up menu to open the Create Assembly File Wizard. The wizard guides you through the process of new assembly file creating.

See also: Assembly Editor 216

5.26.2 Assembly Editor

The editor allows you to work with existing

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.

Description of Assembly Properties are covered in the corresponding topic 206.

See also: Create Assembly Wizard 215

5.26.2.1 Assembly properties

The editor allows to manage assembly properties, files, and classes.

The Files tab displays files belonging to the assembly. Use the corresponding items of popup menu to add a new assembly file within Create Assembly File Wizard and drop assembly files.

Location
Uploads a file to be associated with the assembly, such as source code, debug files or other related information, into the server.

The Classes tab represents the classes metadata as it is written in *.dll files.

The Properties tab allows you to view the assembly name, change the assembly owner, comment, location, etc.

Name
Here you can view the assembly name.

Owner
Specify the owner of the assembly. By default, only the owner of an object can perform various operations with the object. In order to allow other users to operate it, privileges must be granted. (However, users that have the superuser attribute can always access any object.)

Comment
This field stores a comment to the assembly.

Create Date
Indicates the date when the assembly was created.

Location
Specifies the manifest file name that corresponds to the assembly.

Visibility
Indicates whether the assembly is visible for creating common language runtime (CLR) functions, stored procedures, triggers, user-defined data types, and user-defined aggregate functions against it.

Permission Set

Specifies a set of code access permissions that are granted to the assembly when it is accessed by SQL Server.

To apply the changes, select the Apply Changes item in the Navigation bar or use **Ctrl+F9** or **Ctrl+F7** shortcut keys.

It is also possible to modify object properties without opening the object editor: use the Object Properties item of the popup menu of the selected object from the explorer tree.

5.27 Asymmetric Keys

An asymmetric key is a securable entity on the database level. Asymmetric encryption requires a pair of keys: one public, the other private. You can create an asymmetric key entity from a key or pair of keys retrieved from a file or assembly.

Note: The object is available only for Microsoft SQL server 2005.

How can I create a new asymmetric key?

New asymmetric keys are created within Create Asymmetric Key Wizard 219. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the Asymmetric Key icon in the Create Database Object dialog

or

- select the Asymmetric Keys list or any object from that list in the explorer tree;
- select the Create New Asymmetric Key... item from the popup menu

or

- open the database in Database Editor and the Asymmetric Keys tab there;
- press the **Insert** key or select the Create New Asymmetric Key... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

To create a new asymmetric key with the same properties as one of the existing asymmetric keys has:

- select the Object | Duplicate Database Object... main menu item;
- follow the instructions of Duplicate Object Wizard.

How can I edit an existing asymmetric key?

Asymmetric keys can be edited within Asymmetric Key Editor 220. In order to run the editor you should either

- select the asymmetric key for editing in the explorer tree (type the first letters of the asymmetric key name for quick search);
- select the Edit Asymmetric Key ... item from the popup menu

or

- open the database in Database Editor and the Asymmetric Keys tab there;
- select the asymmetric key to edit;
- press the **Enter** key or select the Edit Asymmetric Key item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

⊟ How can I drop an asymmetric key?

To drop an asymmetric key:

- select the asymmetric key to drop in the explorer tree;
- select the Drop Asymmetric Key item from the popup menu

or

- open the database in Database Editor and the Asymmetric Keys tab there;
- select the asymmetric key to drop;
- press the **Delete** key or select the Drop Asymmetric Key item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.

5.27.1 Create Asymmetric Key Wizard

Create Asymmetric Key Wizard guides you through the process of creating a new asymmetric key. See How To Create asymmetric key 218 for instructions on running this wizard.

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

To add a new asymmetric key, specify the following options.

Owner
Specify the owner for the new asymmetric key. By default, only the owner of an object can perform various operations with the object. In order to allow other users to operate it, privileges must be granted. (However, users that have the superuser attribute can always access any object.)

Source Type
Defines the asymmetric key source type.

Source
If *File* or *Executable File* source type is selected, use this field to specify the path to the strong-name file from which the key pair is to be loaded or the assembly file from which the public key is to be loaded.

Assembly
If *Assembly* source type is selected, use this field to specify the name of the assembly 214 from which the public key is to be loaded. The assembly must have been already created in the database.

Algorithm
If *Generate New* source type is selected, you should specify an encryption algorithm in this field.

Use Master Key

If checked, the private key will be encrypted with the database master key. Otherwise you have to specify **password** with which the private key will be encrypted.

Password
Specifies the password the private key is to be encrypted with.

See also: Asymmetric Key Editor 220

5.27.2 Asymmetric Key Editor

Asymmetric Key Editor allows you to edit asymmetric key properties, view the SQL statement for creating the asymmetric key, etc. It can be opened when you create a new asymmetric key or edit the existing one (see How to edit asymmetric key 218 for details).

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.

Description of Assembly Properties are covered in the corresponding topic 220.

See also: Create Asymmetric Key Wizard 219

5.27.2.1 Editing asymmetric key properties

Specify asymmetric key options according to your needs. The detailed description is given below.

Name
Define a name for the asymmetric key.

Owner
Specify the owner for the asymmetric key. By default, only the owner of an object can perform various operations with the object. In order to allow other users to operate it, privileges must be granted. (However, users that have the superuser attribute can always access any object.)

Algorithm
If *Generate New* source type is selected, you should specify an encryption algorithm in this field.

Length
Defines the length of the asymmetric key in bits.

Private Key Encryption
Specifies private key encryption.

Decryption password
Specifies the old password the private key is currently encrypted with.

Encryption password
Specifies a new password for encrypting the private key.

To apply the changes, select the Apply Changes item in the Navigation bar or use **Ctrl**

+F9 or **Ctrl+F7** shortcut keys.

It is also possible to modify object properties without opening the object editor: use the Object Properties item of the popup menu of the selected object from the explorer tree.

5.28 Symmetric Keys

Symmetric encryption is an encryption process that uses a single key to encrypt and decrypt data. This is considered to be a very fast encryption and decryption process.

Note: The object is available only for Microsoft SQL server 2005.

How can I create a new symmetric key?

New symmetric keys are created within Create Symmetric Key Wizard 223. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the Symmetric Key icon in the Create Database Object dialog

or

- select the Symmetric Keys list or any object from that list in the explorer tree;
- select the Create New Symmetric Key... item from the popup menu

or

- open the database in Database Editor and the Symmetric Keys tab there;
- press the **Insert** key or select the Create New Symmetric Key... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

To create a new symmetric key with the same properties as one of the existing symmetric keys has:

- select the Object | Duplicate Database Object... main menu item;
- follow the instructions of Duplicate Object Wizard.

How can I edit an existing symmetric key?

Symmetric keys can be edited within Symmetric Key Editor 223. In order to run the editor you should either

- select the symmetric key for editing in the explorer tree (type the first letters of the symmetric key name for quick search);
- select the Edit Symmetric Key ... item from the popup menu

or

- open the database in Database Editor and the Symmetric Keys tab there;
- select the symmetric key to edit;
- press the **Enter** key or select the Edit Symmetric Key item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

How can I drop a symmetric key?

To drop a symmetric key:

- select the symmetric key to drop in the explorer tree;
- select the Drop Symmetric Key item from the popup menu

or

- open the database in Database Editor and the Symmetric Keys tab there;
- select the symmetric key to drop;
- press the **Delete** key or select the Drop Symmetric Key item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.

5.28.1 Create Symmetric Key Wizard

New symmetric keys are created within Create Symmetric Key Wizard 223. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the Symmetric Key icon in the Create Database Object dialog

or

- select the Symmetric Keys list or any object from that list in the explorer tree;
- select the Create New Symmetric Key... item from the popup menu

or

- open the database in Database Editor and the Symmetric Keys tab there;
- press the **Insert** key or select the Create New Symmetric Key... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

To create a new symmetric key with the same properties as one of the existing symmetric keys has:

- select the Object | Duplicate Database Object... main menu item;
- follow the instructions of Duplicate Object Wizard.

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

To create a new symmetric key, define the encryption algorithm (possible values: DES, TRIPLE DES, RC2, RC4, DESX, AES 128, AES 192, AES 256) and specify the symmetric key encryptions.

See also: Symmetric Key Editor 223, Asymmetric Keys 218.

5.28.2 Symmetric Key Editor

Symmetric Key Editor allows you to edit symmetric key properties, view the SQL statement for creating the symmetric key, etc. It can be opened automatically when you create a new symmetric key or edit the existing one (see How to edit symmetric key

222 for details).

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.

Description of Assembly Properties are covered in the corresponding topic 224.

See also: Create Symmetric Key Wizard 223

5.28.2.1 Symmetric Key Properties

Specify symmetric key options according to your needs. The detailed description is given below.

The Encryptions tab allows you specify the symmetric key encryptions: create new and drop existing one. Select the mechanism you wish to be used for the current encryption. Possible values are *Certificate*, *Password*, *Symmetric Key*, *Asymmetric Key*.

Name
Define a name for the symmetric key.

Owner
Specify the owner of the symmetric key. By default, only the owner of an object can perform various operations with the object. In order to allow other users to operate it, privileges must be granted. (However, users that have the superuser attribute can always access any object.)

Create Date
Displays the date when symmetric key was created.

Modify Date
Displays the date when symmetric key was last modified.

Algorithm
Defines the encryption algorithm. Possible values are DES, TRIPLE DES, RC2, RC4, DESX, AES 128, AES 192, AES 256.

Length
Indicates the length of the asymmetric key in bits.

5.29 Certificates

A certificate is a database-level securable that follows the X.509 standard and supports X.509 V1 fields. You can load a certificate from a file or an assembly 214, or generate a key pair and create a self-signed certificate.

Private keys generated by SQL Server are 1024 bits long. Private keys imported from an external source have a minimum length of 384 bits and a maximum length of 3,456 bits. The length of an imported private key must be an integer multiple of 64 bits.

The private key must correspond to the public key specified by certificate. When you create a certificate from a container, loading the private key is optional. But when SQL Server generates a self-signed certificate, the private key is always created. By default, the private key is encrypted with the help of the database master key. If the database master key does not exist and no password is specified, the action will fail.

Note: The object is available only for Microsoft SQL server 2005.

⊟ How can I create a new certificate?

New certificates are created within Create Certificate Wizard 226. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the Certificate icon in the Create Database Object dialog

or

- select the Certificates list or any object from that list in the explorer tree;
- select the Create New Certificate... item from the popup menu

or

- open the database in Database Editor and the Certificates tab there;
- press the **Insert** key or select the Create New Certificate... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

To create a new certificate with the same properties as one of the existing certificates has:

- select the Object | Duplicate Database Object... main menu item;
- follow the instructions of Duplicate Object Wizard.

⊟ How can I edit an existing certificate?

Certificates can be edited within Certificate Editor 227. In order to run the editor you should either

- select the certificate for editing in the explorer tree (type the first letters of the certificate name for quick search);
- select the Edit Certificate ... item from the popup menu

or

- open the database in Database Editor and the Certificates tab there;
- select the certificate to edit;
- press the **Enter** key or select the Edit Certificate item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

How can I drop an certificate?

To drop a certificate:

- select the certificate to drop in the explorer tree;
- select the Drop Certificate item from the popup menu

or

- open the database in Database Editor and the Certificates tab there;
- select the certificate to drop;
- press the **Delete** key or select the Drop Certificate item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.

5.29.1 Create Certificate Wizard

Create Certificate Wizard guides you through the process of creating a new certificate. See How To Create certificate 225 for instructions on running this wizard.

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

To add a new certificate, specify the following options.

Owner
Specify the owner of the certificate. By default, only the owner of an object can perform various operations with the object. In order to allow other users to operate it, privileges must be granted. (However, users that have the superuser attribute can always access any object.)

Source Type
Define the certificate source type.

File
If *File* or *Executable File* source type is selected, use this field to specify the complete path (including the file name) to the certificate or to a file that has been signed by a certificate. This can be a local path or a UNC path to a network location.

Assembly
If *Assembly* source type is selected, use this field to specify a signed assembly from which the certificate will be loaded. The assembly 214 must have been already created in the database.

Subject

The term "subject" refers to a field in the certificate metadata as defined in the X.509 standard. The subject can be up to 4096 bytes long. Subjects that exceed 4096 bytes will be truncated when stored in the catalog, but the BLOB containing the certificate will retain the full subject name.

Private Key File

Specifies the complete path (including the file name) to the private key. This can be a local path or a UNC path to a network location.

Decryption Password

Specifies the password for decrypting a private key that is retrieved from a file. You do not have to specify a decryption password when the private key is encrypted with the database master key.

Encryption password

Specifies the password that will be used to encrypt the private key that is retrieved from a file. The option is not required when the private key will be encrypted with the database master key. Use this option only in case the private key is encrypted with a password. If no password is specified, the private key of the certificate will be encrypted using the database master key. Omitting this clause will cause an error if the master key of the database cannot be opened.

☑ Use Master Key

If checked, the private key will be encrypted with the database master key. Otherwise, the private key will be encrypted with a password.

Password

Set the password to encrypt the certificate private key with.

Start Date

The field represents the date when the certificate becomes valid.

Expiry Date

The field represents the date when the certificate expires.

☑ Active For Begin Dialog

If checked, makes the certificate available to the initiator of a Service Broker dialog conversation.

See also: Certificate Editor 227

5.29.2 Certificate Editor

Certificate Editor allows you to edit certificate properties and permissions, view the SQL statement for creating the certificate, etc. It can be opened when you create a new certificate or edit the existing one (see How to edit certificate 225 for details).

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.

Description of certificate properties are covered in the corresponding topic 228.

See also: Create Certificate Wizard 226

5.29.2.1 Editing certificate properties

Specify certificate options according to your needs. The detailed description is given below.

Name
Define a name for the certificate.

Owner
Specify the owner of the certificate. By default, only the owner of an object can perform various operations with the object. In order to allow other users to operate it, privileges must be granted. (However, users that have the superuser attribute can always access any object.)

Subject
Represents the certificate subject. The term "subject" refers to a field in the certificate metadata as defined in the X.509 standard. The subject can be up to 4096 bytes long. Subjects that exceed 4096 bytes will be truncated when stored in the catalog, but the BLOB containing the certificate will retain the full subject name.

Issuer
Displays the certificate issuer (known as *Certificate Authority*).

Private Key Encryption
Specifies the private key encryption.

Private Key File
Specifies the complete path (including the file name) to the private key. This can be a local path or a UNC path to a network location.

Decryption Password
Specifies the password for decrypting a private key that is retrieved from a file.

Encryption password
Specifies the password to be used for encrypting the private key that is retrieved from a file.

Start Date
The field indicates the date when the certificate becomes valid.

Expiry Date
The field indicates the date when the certificate expires.

☑ Active For Begin Dialog
If checked, makes the certificate available to the initiator of a Service Broker dialog conversation.

To apply the changes, select the Apply Changes item in the Navigation bar or use **Ctrl+F9** or **Ctrl+F7** shortcut keys.

It is also possible to modify object properties without opening the object editor: use the Object Properties item of the popup menu of the selected object from the explorer tree.

5.30 DDL Triggers

DDL triggers, like standard triggers, execute stored procedures in response to an event. But unlike standard triggers, they do not execute procedures in response to UPDATE, INSERT, or DELETE statements on a table or view. Instead, they primarily execute in response to data definition language (DDL) statements. These include CREATE, ALTER, DROP, GRANT, DENY, REVOKE, and UPDATE STATISTICS statements. DDL triggers are available in Microsoft SQL 2005 and higher.

⊟ How can I create a new DDL trigger?

New DDL Triggers are created within Create DDL Trigger Wizard 231. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the DDL Trigger icon in the Create Database Object dialog

or

- select the DDL Triggers list or any object from that list in the explorer tree;
- select the Create New DDL Trigger... item from the popup menu

or

- open the database in Database Editor and the DDL Triggers tab there;
- press the **Insert** key or select the Create New DDL Trigger... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

To create a new DDL Trigger with the same properties as one of the existing DDL Triggers has:

- select the Object | Duplicate Database Object... main menu item;
- follow the instructions of Duplicate Object Wizard.

⊟ How can I edit an existing DDL trigger?

DDL Triggers can be edited within DDL Trigger Editor 232. In order to run the editor you should either

- select the DDL Trigger for editing in the explorer tree (type the first letters of the DDL Trigger name for quick search);
- select the Edit DDL Trigger ... item from the popup menu

or

- open the database in Database Editor and the DDL Triggers tab there;
- select the DDL Trigger to edit;
- press the **Enter** key or select the Edit DDL Trigger item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

You can change the name of the DDL Trigger using the Rename DDL Trigger dialog. To open the dialog you should either

- select the DDL Trigger to rename in the explorer tree;
- select the Rename DDL Trigger item from the popup menu

or

- open the database in Database Editor and the DDL Triggers tab there;
- select the DDL Trigger to rename;
- select the Rename DDL Trigger item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

⊟ How can I drop a DDL trigger?

To drop a DDL Trigger:

- select the DDL Trigger to drop in the explorer tree;
- select the Drop DDL Trigger item from the popup menu

or

- open the database in Database Editor and the DDL Triggers tab there;
- select the DDL Trigger to drop;
- press the **Delete** key or select the Drop DDL Trigger item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.

5.30.1 Create DDL Trigger Wizard

Create DDL Trigger Wizard guides you through the process of creating a new DDL trigger. See How To Create DDL trigger 230 for instructions on running this wizard.

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

To create a new DDL Trigger, specify the following options: a comment (optional), the Enabled option, the security context under which the DDL trigger is to be executed (see Execute As Clause (Microsoft SQL 2005 References) for details) and the events which execution cause the DDL trigger to fire.

To add events, use the ellipse button (**...**) and specify event types and groups in the Add Event dialog. If an event group is specified, the DDL trigger fires after execution of any event from this group.

Specify the trigger steps to be executed when the DDL trigger fires as trigger Definition.

See also: DDL Trigger Editor 232

5.30.2 DDL Trigger Editor

DDL Trigger Editor allows you to edit DDL trigger definition (DDL trigger name, DDL trigger comment, etc.). It can be opened when you create a new DDL trigger or edit the existing one (see DDL Triggers 230 for details).

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.

- Editing DDL trigger properties 232

See also: Create DDL Trigger Wizard 230

5.30.2.1 Editing DDL trigger properties

Specify DDL Trigger options according to your needs. The detailed description is given below.

Definition
The field contains the definition of the DDL trigger. It describes the trigger conditions and actions. The trigger actions specified in the Transact-SQL statements go into effect when the DDL operation is tried. Triggers can include any number and kind of Transact-SQL statements, with exceptions.

Name
Displays the DDL trigger name.

Comment
Specify a comment to the DDL trigger.

Create Date
Indicates the date when the DDL trigger was created.

Modify Date
Indicates the date when the DDL trigger was last modified.

☑ Encrypted
Encrypts the text of the CREATE TRIGGER statement. Using Encrypted prevents the trigger from being published as part of SQL Server replication.

Execute As
Specifies the security context under which the DDL Trigger is to be executed (For details see Execute As Clause (Microsoft SQL 2005 References)).

Events
This field allows you to specify the events execution of which causes a DDL trigger to fire. Click ellipse button (...) on the right of the events text area to select this events or event groups using the Add Event Dialog.

☑ Enabled
Disabling a trigger does not drop it. The DDL trigger still exists as an object in the

current database/server. However, the DDL trigger does not fire when any Transact-SQL statements on which it was programmed are executed.

To apply the changes, select the Apply Changes item in the Navigation bar or use **Ctrl+F9** or **Ctrl+F7** shortcut keys.

It is also possible to modify object properties without opening the object editor: use the Object Properties item of the popup menu of the selected object from the explorer tree.

5.31 CLR DDL Triggers

You can build triggers using the SQL Server integration with the .NET Framework common language runtime (CLR).
CLR DDL triggers, like standard triggers, execute stored procedures in response to an event. But unlike standard triggers, they do not execute procedures in response to UPDATE, INSERT, or DELETE statements on a table or view. Instead, they primarily execute in response to data definition language (CLR DDL) statements. These include CREATE, ALTER, DROP, GRANT, DENY, REVOKE, and UPDATE STATISTICS statements. CLR DDL triggers are available since Microsoft SQL 2005.

How can I create a new CLR DDL trigger?

New CLR DDL Triggers are created within Create CLR DDL Trigger Wizard. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the CLR DDL Trigger icon in the Create Database Object dialog

or

- select the CLR DDL Triggers list or any object from that list in the explorer tree;
- select the Create New CLR DDL Trigger... item from the popup menu

or

- open the database in Database Editor and the CLR DDL Triggers tab there;
- press the **Insert** key or select the Create New CLR DDL Trigger... item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

To create a new CLR DDL Trigger with the same properties as one of the existing CLR DDL Triggers has:

- select the Object | Duplicate Database Object... main menu item;
- follow the instructions of Duplicate Object Wizard.

How can I edit an existing CLR DDL trigger?

CLR DDL Triggers can be edited within CLR DDL Trigger Editor. In order to run the editor you should either

- select the CLR DDL Trigger for editing in the explorer tree (type the first letters of the CLR DDL Trigger name for quick search);
- select the Edit CLR DDL Trigger ... item from the popup menu

or

- open the database in Database Editor and the CLR DDL Triggers tab there;
- select the CLR DDL Trigger to edit;
- press the **Enter** key or select the Edit CLR DDL Trigger item from the popup menu (alternatively, you may use the corresponding

link of the Navigation Bar).

You can change the name of the CLR DDL Trigger using the Rename CLR DDL Trigger dialog. To open the dialog you should either

- select the CLR DDL Trigger to rename in the explorer tree;
- select the Rename CLR DDL Trigger item from the popup menu

or

- open the database in Database Editor and the CLR DDL Triggers tab there;
- select the CLR DDL Trigger to rename;
- select the Rename CLR DDL Trigger item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar).

⊟ How can I drop a CLR DDL trigger?

To drop a CLR DDL Trigger:

- select the CLR DDL Trigger to drop in the explorer tree;
- select the Drop CLR DDL Trigger item from the popup menu

or

- open the database in Database Editor and the CLR DDL Triggers tab there;
- select the CLR DDL Trigger to drop;
- press the **Delete** key or select the Drop CLR DDL Trigger item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.

5.31.1 Create CLR DDL Trigger Wizard

Create CLR DDL Trigger Wizard guides you through the process of creating a new CLR DDL trigger. See How To Create CLR DDL trigger 234 for instructions on running this wizard.

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

- Specifying CLR DDL trigger options 235

See also: CLR DDL Trigger Editor 236

5.31.1.1 Specifying CLR DDL trigger options

This step of the wizard allows you to define CLR DDL trigger options according to your needs. The detailed description is given below.

Comment
Specify a comment to the CLR DDL trigger.

Execute As
Specifies the security context under which the CLR DDL trigger is to be executed (for details see Execute As Clause (Microsoft SQL 2005 References)).

Events
This field allows you to specify the events execution of which causes a CLR DDL trigger to fire. Click the ellipse button (...) to the right from the events text area to select this event or event group using the Add Event Dialog.

☑ **Enabled**
Disabling a trigger does not drop it. The CLR DDL trigger still exists as an object in the current database/server. However, the CLR DDL trigger does not fire when any Transact-SQL statements on which it was programmed are executed.

Assembly, Class, Method
Specifies the method of a .NET Framework assembly for a CLR stored function to reference. Class must be a valid SQL Server identifier and must exist as a class in the Assembly. The assembly must have been already created in the database.

See also: Assemblies 214

5.31.2 CLR DDL Trigger Editor

CLR DDL Trigger Editor allows you to edit CLR DDL trigger definition (CLR DDL trigger name, CLR DDL trigger comment, etc.). It can be opened automatically when you create a new CLR DDL trigger or edit the existing one (see How to edit CLR DDL trigger 234 for details).

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.

- Editing CLR DDL trigger properties 236

See also: Create DDL Trigger Wizard 235

5.31.2.1 Editing CLR DDL trigger properties

Specify CLR DDL trigger options according to your needs. The detailed description is given below.

Name
Displays CLR DDL trigger name.

Note: the name of the object must be unique among all the object names in the server. You can use any identifier that is allowed by Microsoft SQL server.

Comment
Specify a comment to the CLR DDL trigger.

Create Date
Indicates the date when the CLR DDL trigger was created.

Modify Date
Indicates the date when the CLR DDL trigger was last modified.

Execute As
Specifies the security context under which the CLR DDL trigger is to be executed (For details see Execute As Clause (Microsoft SQL 2005 References)).

Events
This field allows you to specify events the execution of which causes a CLR DDL trigger to fire. Click ellipse button (...) on the right of the events text area to select this events or event groups using the Add Event Dialog.

☑ Enabled
Disabling a trigger does not drop it. The CLR DDL trigger still exists as an object in the current database/server. However, the CLR DDL trigger does not fire when any Transact-SQL statements on which it was programmed are executed.

Assembly, Class, Method
Specifies the method of a .NET Framework assembly for a CLR stored function to reference. Class must be a valid SQL Server identifier and must exist as a class in the Assembly. The assembly must already have been created in the database.

See also: Assemblies 214

To apply the changes, select the Apply Changes item in the Navigation bar or use **Ctrl+F9** or **Ctrl+F7** shortcut keys.

It is also possible to modify object properties without opening the object editor: use the Object Properties item of the popup menu of the selected object from the explorer tree.

5.32 Table Types

Table type is a special data type that can be used to store a result set for processing at a later time. Table data type is primarily used is for temporary storage of a set of rows returned as the result set of a table-valued function.

⊟ How can I create a new table type?

New table types are created within Create Table Type Wizard 238. In order to run the wizard you should either

- select the Object | Create Database Object... main menu item;
- select the Table Type icon in the Create Database Object dialog

or

- select the Table Types list or any object from that list in the explorer tree;
- select the Create New Table Type... item from the popup menu

To create a new table type with the same properties as one of the existing one has:

- select the Object | Duplicate Database Object... main menu item;
- follow the instructions of Duplicate Object Wizard.

⊟ How can I edit an existing table type?

Table data types can be edited within Table Type Editor 239. In order to run the editor you should either

- select the table type for editing in the explorer tree (type the first letters of the type name for quick search);
- select the Edit Table Type... item from the popup menu

⊟ How can I drop a table type?

To drop a table data type:

- select the type to drop in the explorer tree;
- select the Drop Table Type item from the popup menu

and confirm dropping in the dialog window.

5.32.1 Create Table Type Wizard

Create Table Type Wizard guides you through the process of creating a new data type. All the information how to run the wizard you can find here 238.

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

First of all set the new table type name. You can also describe the type using the Comment field.

On the next step specify the table type fields and their properties. Use the pop-up menu or press **Insert** to add a new field and set its properties in Field Editor.

5.32.2 Table Type Editor

Table Type Editor allows you to edit table data type properties. To open the editor, use the popup menu of the corresponding node on the Explorer tree.

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.

Use the editor to rename and describe selected table type.

To apply the changes, select the Apply Changes item in the Navigation bar or use **Ctrl+F9** or **Ctrl+F7** shortcut keys.

It is also possible to modify object properties without opening the object editor: use the Object Properties item of the popup menu of the selected object from the explorer tree.

6 Server Objects

With MS SQL Maestro you can accomplish all the popular actions of server administration. Note that before working with server objects you should connect to any server database first (see Database Management 22).

The following list contains the most common server objects supported by MS SQL Maestro.

- Databases 243;
- Logins 245;
- Credentials 301;
- Jobs 271;
- Alerts 294;
- Operators 283;
- Backup Devices 263;
- Linked servers 307;
- Processes 406;
- Server Variables 261.

Below you can find some common ways of server object management.

Creating of a new server object

New server objects are created within the appropriate Create Object Wizard. In order to run the wizard you should either

- select the corresponding object list (such as Logins) or any object from that list and then use the Create New... item from the popup menu

or

- open the server in Server Editor 242 and the necessary objects' tab there and press **Insert** or select the Create New... item from the popup menu (Alternatively, use the corresponding link of the Navigation Bar).

Editing of an existing server object

Server objects are edited within the corresponding Object Editor. In order to open the editor you should either

- select the server object for editing in the explorer tree (type the first letters of the object name for quick search);
- select the Edit Object item from the popup menu

or

- open the server in Server Editor and the corresponding objects' tab there;
- select the server object to edit;
- press the **Enter** key or select the Edit Object item from the popup menu (alternatively, you can use the corresponding link of the Navigation Bar).

⊟ Dropping of a server object

To drop the existing server object:

- select the server object to drop in the explorer tree;
- select the Drop Object item from the popup menu

or

- open the server in Server Editor and the appropriate objects' tab there;
- select the server object to drop;
- press the **Delete** key or select the Drop Object item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.

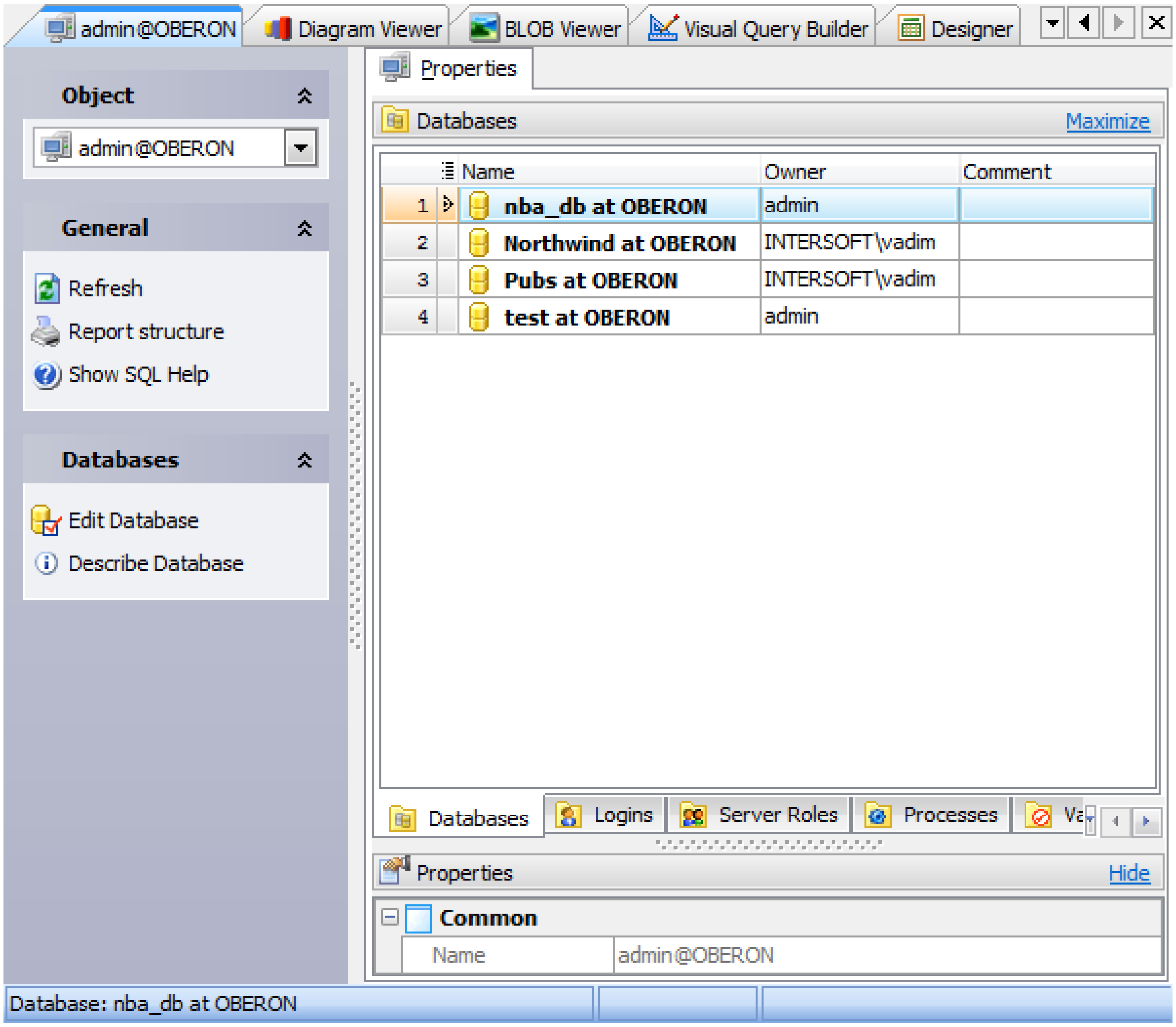
6.1 Server Editor

Server Editor allows you to look through all the server objects. Only connected databases are available for examination.

To run Server Editor, select the essential server from the Explorer Tree and press **Enter**, or use popup menu.

Server Editor consists of several tabs according different kinds of server-scope objects. Every tab is intended for managing server objects (e.g. *databases, logins, server roles, operators, categories, alerts, jobs, schedules, backup devices*, etc.). Any object can be opened in its editor. The popup menu allows you to create new, edit or drop the selected server object. Using the popup menu you can also create a copy of the object.

You can operate on several objects at a time. For this you have to select database objects with the **Shift** or the **Ctrl** key pressed. After the group of objects is selected, you can operate on it, e.g. *delete several objects* at once, as it were a single object.



	Name	Owner	Comment
1	nba_db at OBERON	admin	
2	Northwind at OBERON	INTERSOFT\vadim	
3	Pubs at OBERON	INTERSOFT\vadim	
4	test at OBERON	admin	

Common

Name: admin@OBERON

Database: nba_db at OBERON

6.2 Databases

MS SQL Maestro allows you to manipulate databases by means of database profiles. Profile contains database connection settings and a set of options to automatize common manipulations with databases (a possibility to connect to the database at MS SQL Maestro startup, login prompt before connection, etc.). To start working with databases in MS SQL Maestro, you should create database profile(s) first.

Use the following links for details:

⊟ How can I create a new database?

Use for this purpose Create Database Wizard 34. In order to run the wizard you should either

- select the Database | Create New Database... main menu item

or

- use the Create New Database... item of the popup menu.

Using Create Database Wizard set the Create profile after creating the database option to create a new profile and open the Database Profile Properties dialog after the database is created.

⊟ How can I change attributes of an existing database?

To edit a database:

- select the database to edit in the explorer tree;
- edit database properties within the appropriate tabs of Database Editor 39.

⊟ How can I drop an existing database?

In order to drop a database you should first select the database to drop in the explorer tree and establish connection (if you are not connected to the database yet), then either

- select the Database | Drop Database main menu item

or

- use the Drop Database item of the popup menu

and confirm dropping in the dialog window to complete the operation.

⊟ How can I create new database profiles?

In MS SQL Maestro database profiles are created within Create Database Profiles Wizard 24. In order to run the wizard you should either

- select the Database | Create Database Profiles... main menu item

or

- use the Create Database Profiles... item of the popup menu.

Using Create Database Profiles Wizard set the necessary connection and authorization options and click the Ready button to complete the operation.

⊟ How can I edit existing database profile options?

Database connection properties and profile options are edited within the Database Profile Properties 27 dialog window. In order to open the dialog for the selected database profile you should either

- select the Database | Edit Database Profile... main menu item

or

- use the Edit Database Profile... item of the popup menu.

⊟ How can I remove database profiles?

In order to remove a database profile you should first select the database profile in the explorer tree, then either select the Database | Remove Database Profile main menu item, or use the Remove Database Profile item of the popup menu and confirm removing profile in the dialog window to complete the operation.

⊟ How can I connect to a database?

In order to connect to a database you should first select the database in the explorer tree, then either

- select the Database | Connect to Database main menu item

or

- use the Connect to Database item of the popup menu.

⊟ How can I disconnect from a database?

In order to disconnect from a database you should first select the database in the explorer tree, then either

- select the Database | Disconnect from Database main menu item

or

- use the Disconnect from Database item of the popup menu.

6.3 Logins

A user passes through two stages of security when working in Microsoft® SQL Server™: *authentication* and *authorization (permissions validation)*. The *authentication* stage identifies the user by a login account and verifies only the ability to connect to an instance of SQL Server. If *authentication* is successful, the user connects to an instance of SQL Server. The user 201 then needs permissions to access databases on the server done by *granting access* to an account in each database mapped to the user login.

To give permissions for changing server objects to a login, you should include the login into the appropriate server role 258.

There are four types of logins: *Windows Authentication*, *SQL Server Authentication*, and also *Certificate*, and *Asymmetric Key* for Microsoft SQL Server 2005.

- It is recommended to you use *Windows Authentication* for security reasons because the users who connect through a Microsoft Windows NT® 4.0 or Windows® 2000 user account can make use of trusted connections. Trusted connections are those validated by Windows NT 4.0 or Windows 2000.

- It is also possible to connect using *SQL Server Authentication*. When a user connects with a specified login name and password from a nontrusted connection, SQL Server performs the authentication itself by checking to see if a SQL Server login account has been set up and whether the specified password matches the one previously recorded. If SQL Server does not have a login account set, authentication fails and the user receives an error.

- To enable encrypted connections, create login from a *Certificate* or from an *Asymmetric Key.*

How can I create a new login?

New logins are created within Create Login Wizard 246. In order to run the wizard you should either

- select the Logins list or any object from that list and then use the Create New Login item from the popup menu

or

- open the server in Server Editor and the Logins tab there and press **Insert** or select the Create New Login item from the popup menu (Alternatively, use the corresponding link of the Navigation Bar).

How can I edit an existing login?

Logins are edited within Login Editor 253. In order to open the editor you should either

- select the login for editing in the explorer tree (type the first letters of the login name for quick search);
- select the Edit Login item from the popup menu

or

- open the server in Server Editor and the Logins tab there;
- select the login to edit;
- press the **Enter** key or select the Edit Login item from the popup menu (alternatively, you can use the corresponding link of the Navigation Bar).

How can I drop a login?

To drop the existing login:

- select the login to drop in the explorer tree;
- select the Drop Login item from the popup menu

or

- open the server in Server Editor and the Logins tab there;
- select the login to drop;
- press the **Delete** key or select the Drop Login item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.

6.3.1 Create Login Wizard

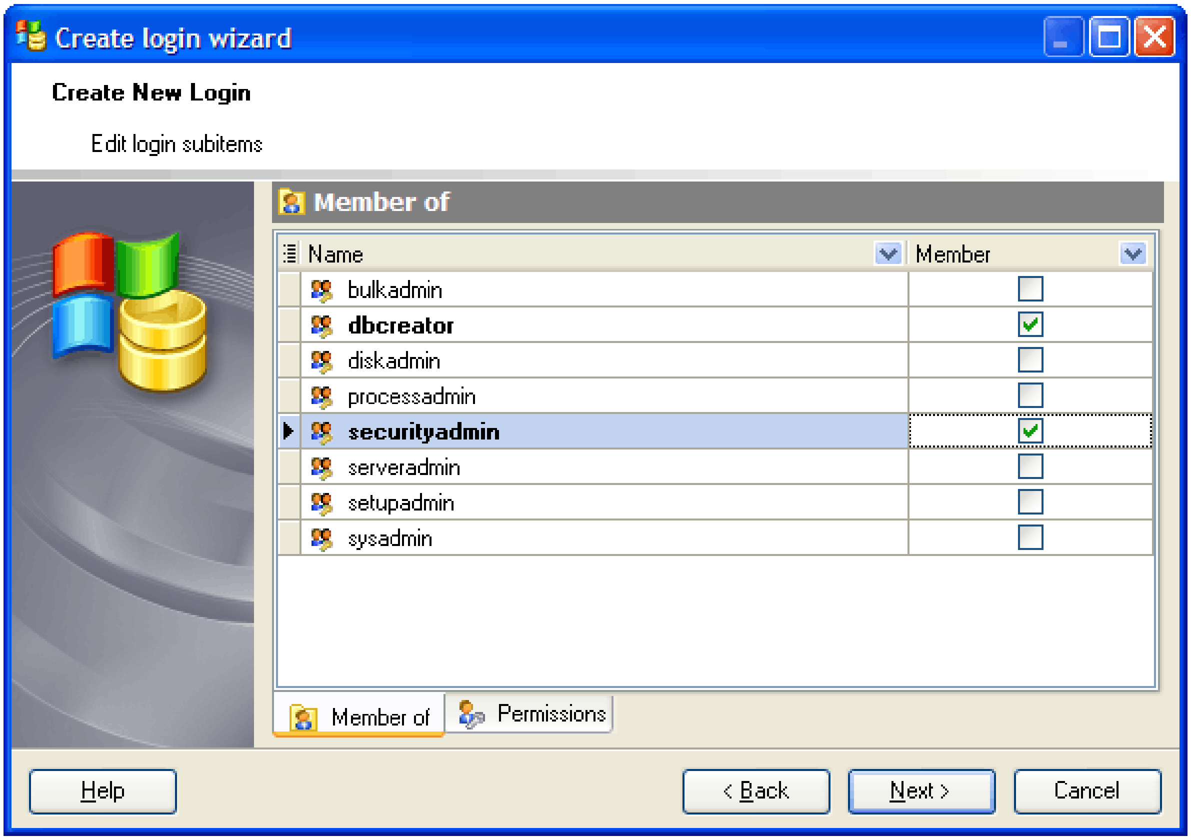
Create Login Wizard guides you through the process of creating a new login. See How To Create Login 245 for instructions on running this wizard.

Note: To view, edit, create or remove SQL Server logins, the connected user must be a member of the SQL Server *securityadmin* and *sysadmin* fixed server roles. Each member of a fixed server role can add other logins within the role he belongs to.

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

- Windows Authentication 247
- SQL Server Authentication 249
- Certificate 250
- Asymmetric Key 252
- Setting login permissions 253

See also: Login Editor 253



Create New Login

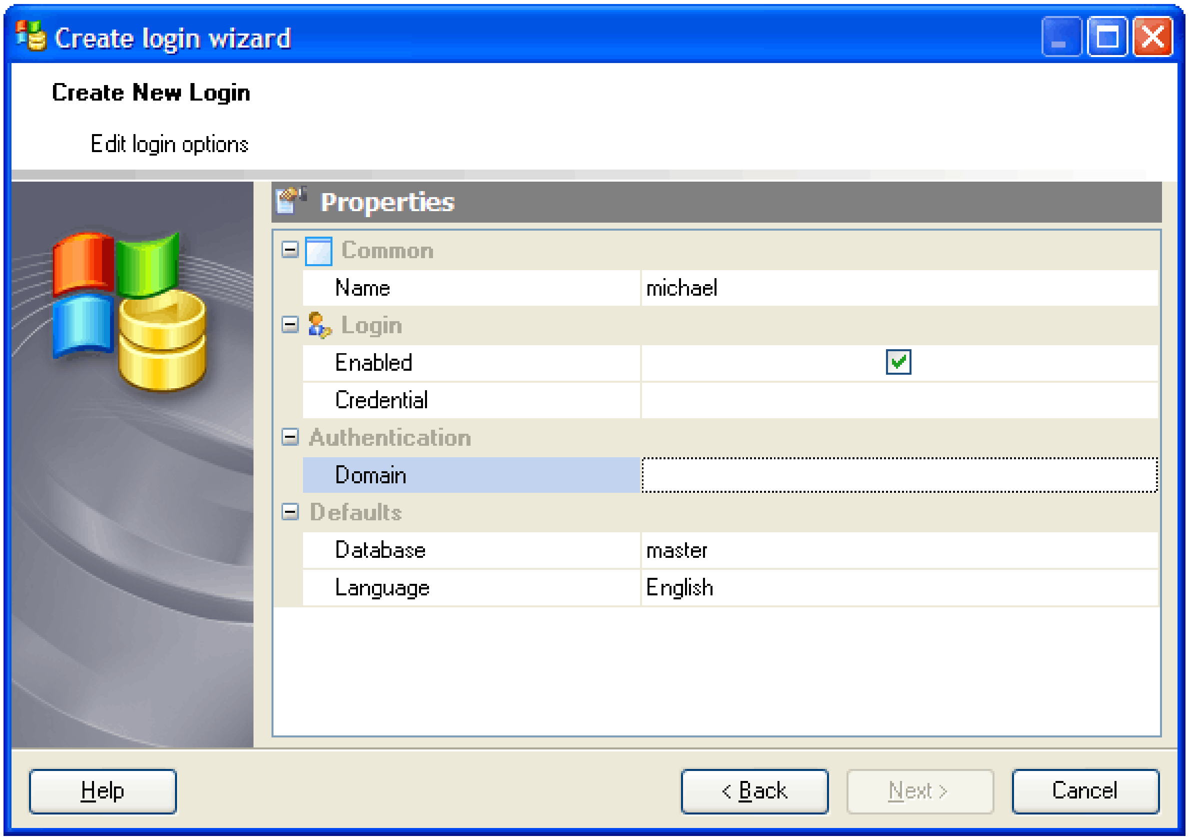
Edit login subitems

Member of

Name	Member
bulkadmin	☐
dbcreator	☑
diskadmin	☐
processadmin	☐
securityadmin	☑
serveradmin	☐
setupadmin	☐
sysadmin	☐

6.3.1.1 Windows Authentication login options

Fill in the following fields to create a Windows Authentication login:



Create New Login

Edit login options

Properties

Common	
Name	michael
Login	
Enabled	☑
Credential	
Authentication	
Domain	
Defaults	
Database	master
Language	English

Name

Specify the name of the *Microsoft SQL Server* login that is being created.

Note: the name of the object must be unique among all the object names in the server. You can use any identifier that is allowed by Microsoft SQL server.

In Microsoft SQL server 2005 you can also use Credentials. The name of a credential 301 is to be mapped to the new SQL Server login. A credential is a record containing the authentication information required for connecting to a resource *outside of SQL Server*. The credential must already exist in the server.

Database

Select the default database to be assigned to the login from the drop-down list of connected databases including the *master* database.

Language

Specify the default language to be assigned to the login. The default language is set to the current default language of the server. If the default language of the server is changed in the future, the default language of the login will remain unchanged.

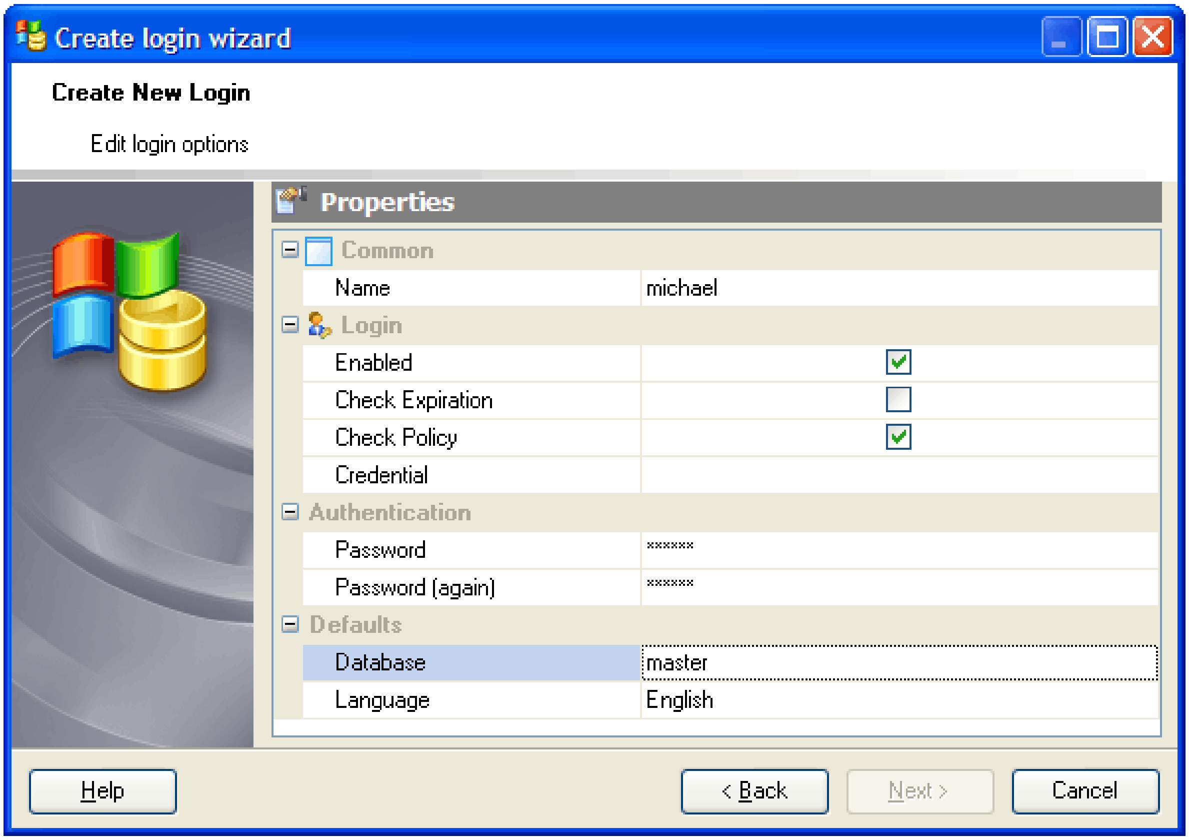
Domain

Specify the domain to be used to validate the login network security attributes. SQL Server achieves login security integration with Windows NT 4.0 by using the security attributes of a network user to control login access. A user's network security attributes are established at network login time and are validated by a Windows domain controller.

Security Access
If *Deny access* value is selected, the login is disabled. (up to Microsoft SQL 2005)

6.3.1.2 SQL Server Authentication login options

Fill in the following fields to create a SQL Server Authentication login:



Common	
Name	michael
Login	
Enabled	☑
Check Expiration	☐
Check Policy	☑
Credential	
Authentication	
Password	******
Password (again)	******
Defaults	
Database	master
Language	English

Name
Specify the name of the *SQL Server* login that is being created.

Note: the name of the object must be unique among all the object names in the server. You can use any identifier that is allowed by Microsoft SQL server.

☑ Enabled (in Microsoft SQL 2005)
If checked, enables the login.

☑ Check Expiration (in Microsoft SQL 2005)
Specify whether password expiration policy should be enforced on this login.

☑ Check Policy (in Microsoft SQL 2005)
Specify whether the Windows password policies of the computer with SQL Server running should be enforced on this login.

In Microsoft SQL server 2005 you can also use Credentials. The name of a credential 301 is to be mapped to the new SQL Server login. A credential is a record containing the authentication information required for connecting to a resource *outside of SQL Server*. The credential must already exist in the server.

Database
Specify the default database to be assigned to the login. The default database is set to MASTER.

Language
Specify the default language to be assigned to the login. The default language is set to the current default language of the server. If the default language of the server is changed in the future, the default language of the login will remain unchanged.

Password
Enter the password according to the server requirements. In Microsoft SQL 2005 this value may be supplied already hashed when creating SQL Server logins.

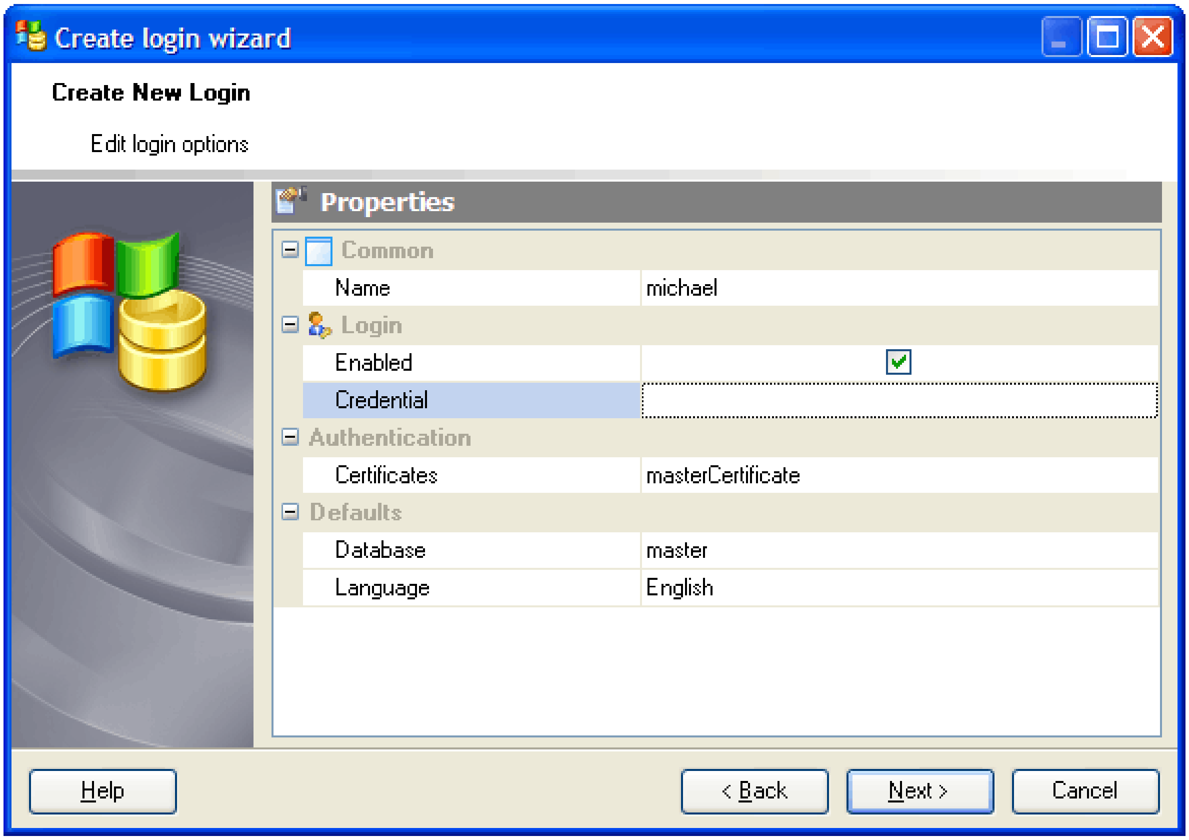
The option Hashed is available only in Microsoft SQL 2005. Check it if the password is already hashed. If this option is not selected, the string entered as password will be hashed prior to being stored in the database.

☑ Must Change (in Microsoft SQL 2005)
If this option is checked, SQL Server will prompt the user for a new password the first time the new login is used.

6.3.1.3 Certificate login options

Certificate logins were implemented in Microsoft SQL server 2005.

This step of the wizard allows you to specify Certificate login options according to your needs. The detailed description is given below.



Common	
Name	michael
Login	
Enabled	☑
Credential	
Authentication	
Certificates	masterCertificate
Defaults	
Database	master
Language	English

Name
Specifies the name of the *Certificate* login that is being created.

> **Note:** the name of the object must be unique among all the object names in the server. You can use any identifier that is allowed by Microsoft SQL server.

☑ Enabled
If checked, the login is enabled.

Credential
Define here the name of a credential 301 to be mapped to the new SQL Server login. A credential is a record containing the authentication information required for connecting to a resource *outside of SQL Server*. The credential must already exist in the server.

Database
Specify the default database to be assigned to the login. The default database is set to MASTER.

Language
Specify the default language to be assigned to the login. The default language is set to the current default language of the server. If the default language of the server is changed in the future, the default language of the login will remain unchanged.

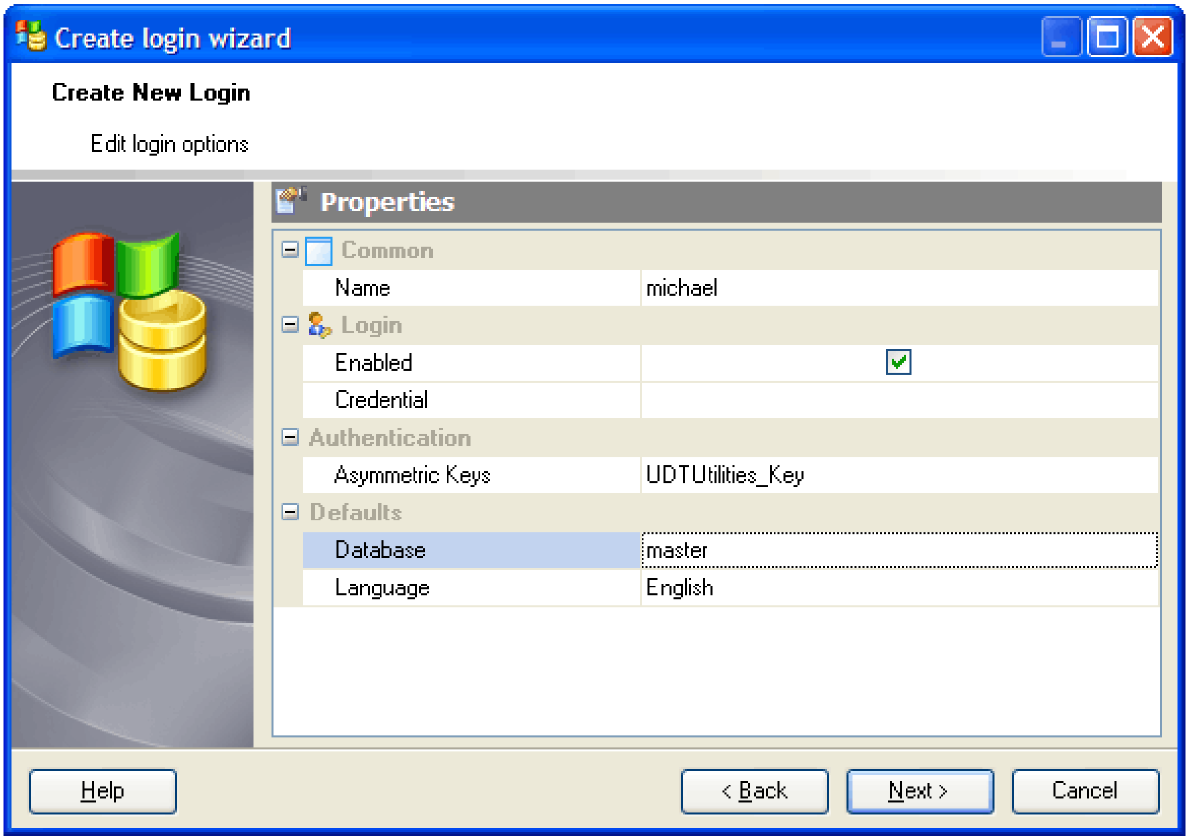
Certificate
Specifies the name of a certificate to be associated with this login. This certificate must

already exist in the master database.

6.3.1.4 Asymmetric Key login options

Asymmetric Key logins were implemented in Microsoft SQL server 2005.

This step of the wizard allows you to specify Asymmetric Key login options according to your needs. The detailed description is given below.



Create login wizard

Create New Login

Edit login options

Properties

Common

Name: michael

Login

Enabled: ☑

Credential:

Authentication

Asymmetric Keys: UDTUtilities_Key

Defaults

Database: master

Language: English

Name
Specify the name of the *Asymmetric Keys* login that is created.

Note: the name of the object must be unique among all the object names in the server. You can use any identifier that is allowed by Microsoft SQL server.

☑ Enabled
If checked, the login is enabled.

Credential
Define here the name of a credential 301 to be mapped to the new SQL Server login. A credential is a record containing the authentication information required for connecting to a resource *outside of SQL Server*. The credential must already exist in the server.

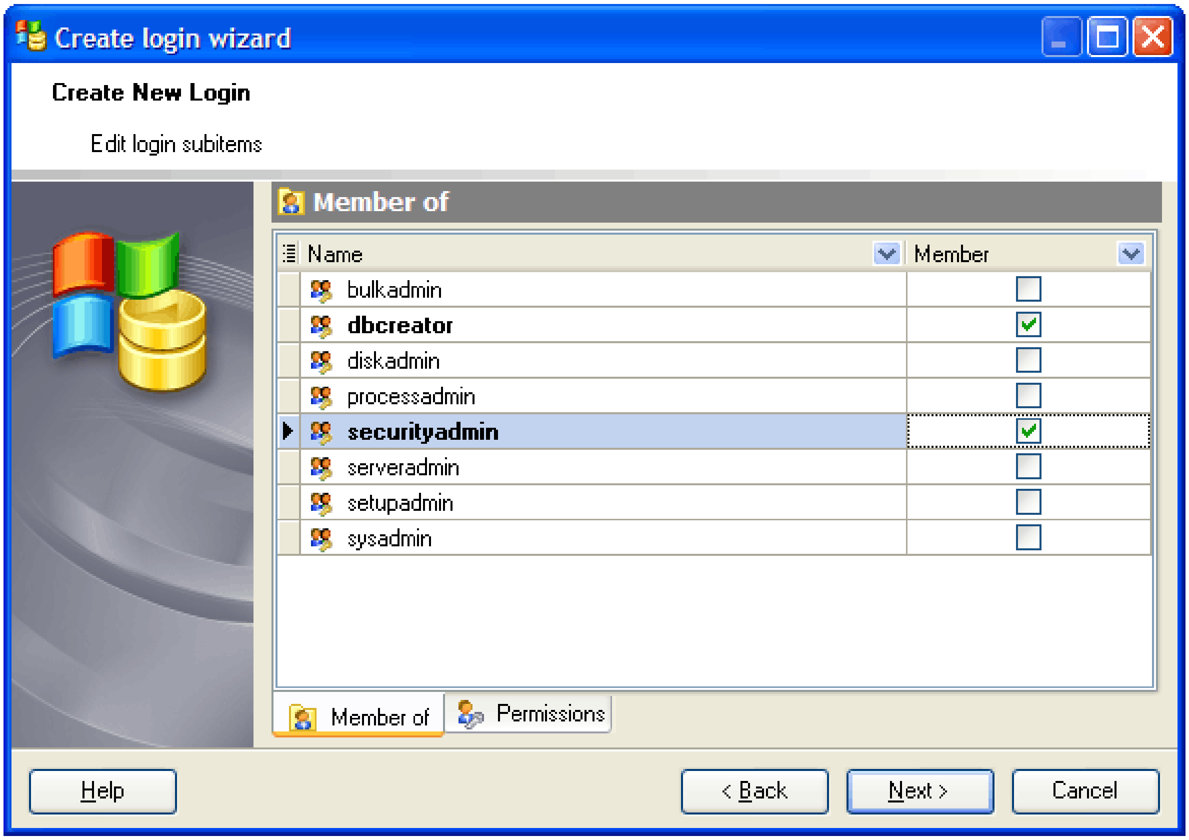
Database
Specify the default database to be assigned to the login. The default database is set to MASTER.

Language
Specify the default language to be assigned to the login. The default language is set to the current default language of the server. If the default language of the server is changed in the future, the default language of the login will remain unchanged.

Asymmetric Key
Specify the name of an asymmetric key to be associated with this login. This key must already exist in the master database.

6.3.1.5 Setting login permissions

The wizard step allows to define fixed server roles membership for the login being created and to grant the new login permissions. Just open the corresponding tab and check the appropriate boxes.



Name	Member
bulkadmin	☐
dbcreator	☑
diskadmin	☐
processadmin	☐
securityadmin	☑
serveradmin	☐
setupadmin	☐
sysadmin	☐

6.3.2 Login Editor

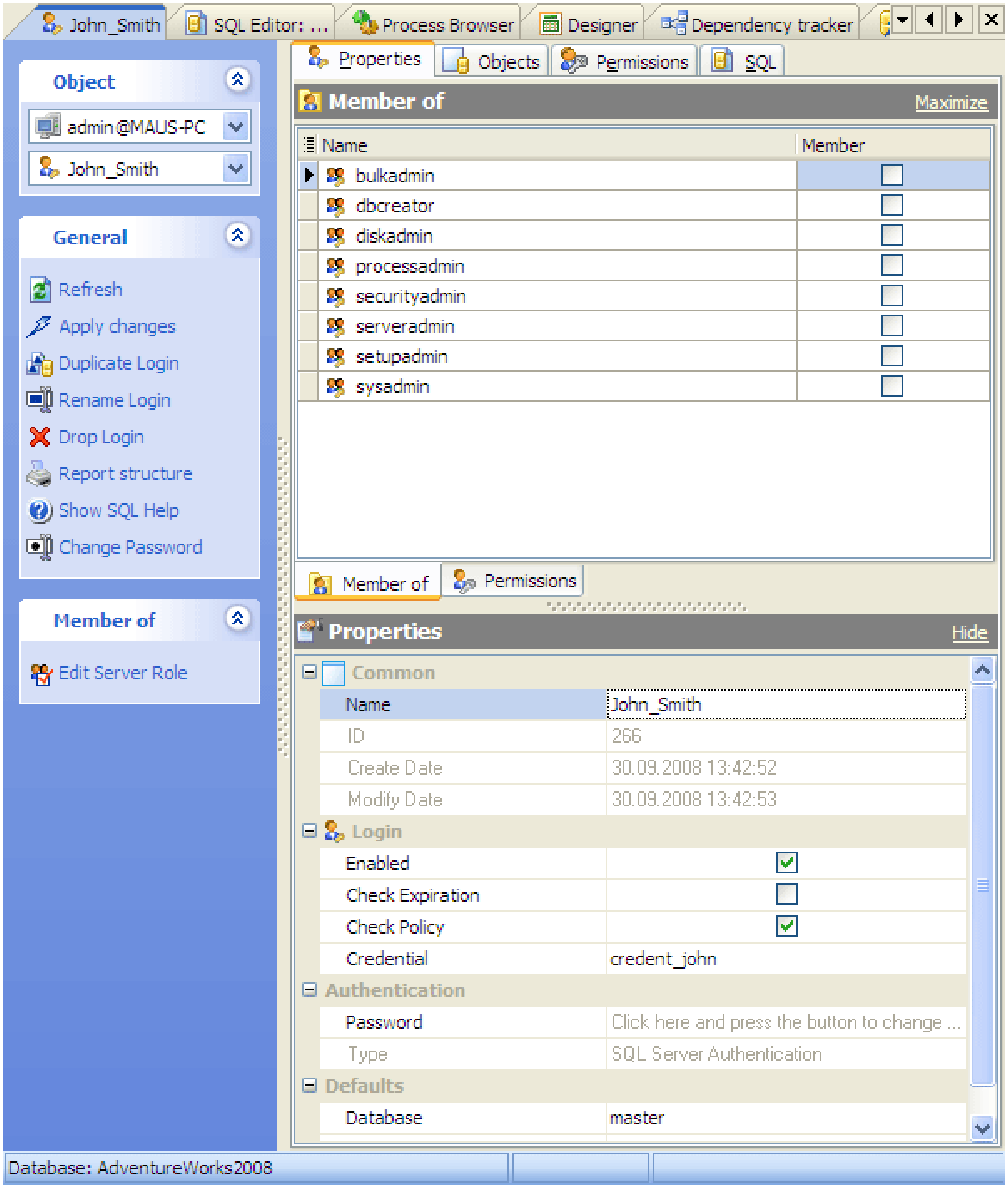
Login Editor allows you to edit login properties, view the SQL statement for creating the login, etc.

> **Note:** To view, edit, or remove SQL Server logins, the connected user must be a member of the SQL Server *securityadmin* and *sysadmin* fixed server roles.

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.

- Editing login properties 254
- Editing login objects 257

See also: Create Login Wizard 246

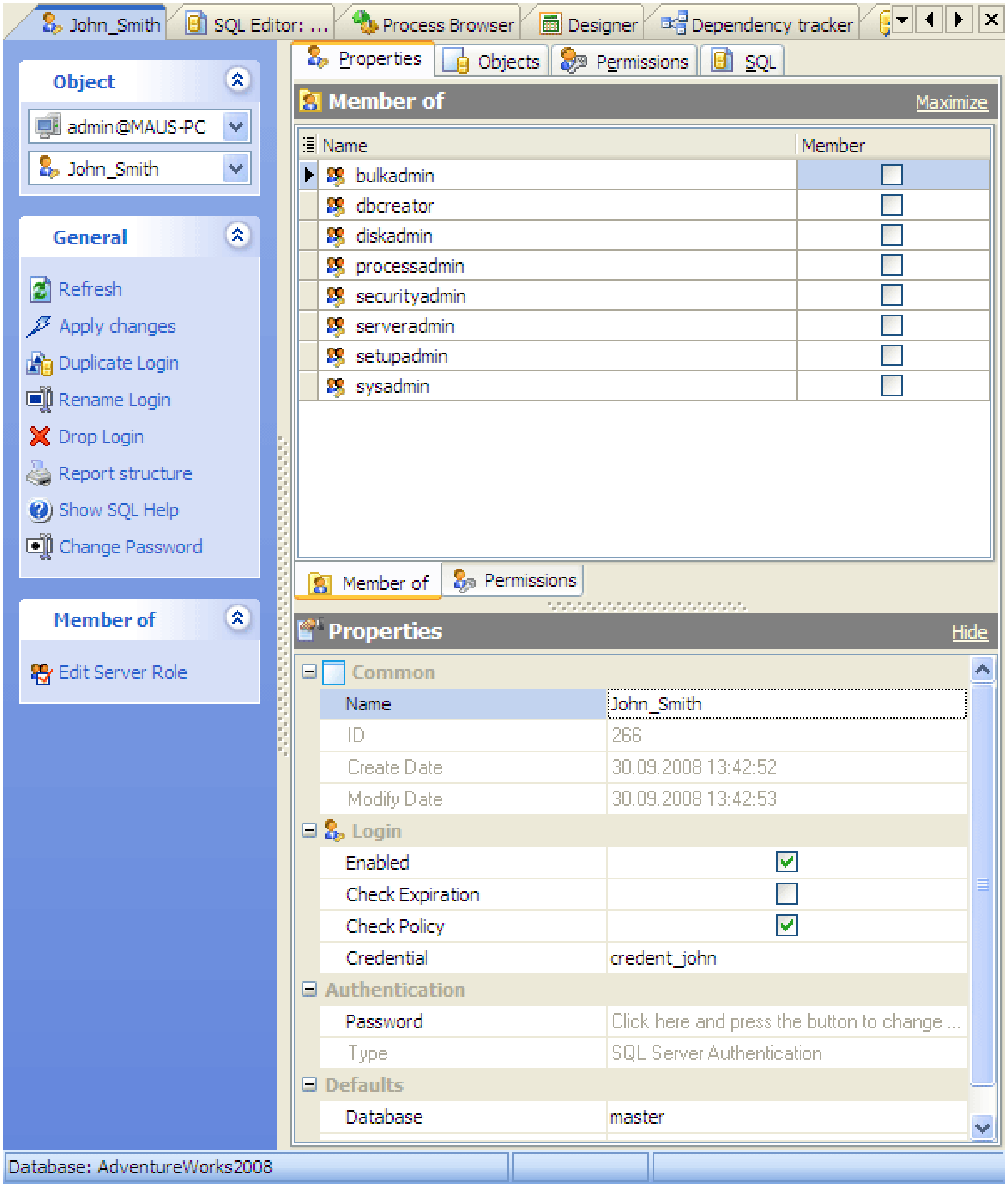


Name	Member
bulkadmin	☐
dbcreator	☐
diskadmin	☐
processadmin	☐
securityadmin	☐
serveradmin	☐
setupadmin	☐
sysadmin	☐

Common	
Name	John_Smith
ID	266
Create Date	30.09.2008 13:42:52
Modify Date	30.09.2008 13:42:53
Login	
Enabled	☑
Check Expiration	☐
Check Policy	☑
Credential	credent_john
Authentication	
Password	Click here and press the button to change ...
Type	SQL Server Authentication
Defaults	
Database	master

6.3.2.1 Editing login properties

Specify login options according to your needs. The detailed description is given below.



Name	Member
bulkadmin	☐
dbcreator	☐
diskadmin	☐
processadmin	☐
securityadmin	☐
serveradmin	☐
setupadmin	☐
sysadmin	☐

Common	
Name	John_Smith
ID	266
Create Date	30.09.2008 13:42:52
Modify Date	30.09.2008 13:42:53
Login	
Enabled	☑
Check Expiration	☐
Check Policy	☑
Credential	credent_john
Authentication	
Password	Click here and press the button to change ...
Type	SQL Server Authentication
Defaults	
Database	master

Name

Specify the name of the SQL Server login that is created.

Note: the name of the object must be unique among all the object names in the server. You can use any identifier that is allowed by Microsoft SQL server.

Create Date

Displays the date when the login was created.

Modify Date
Displays the date when the login was last modified.

☑ Enabled (in Microsoft SQL 2005)
If checked, this login is enabled.

☑ Check Expiration (in Microsoft SQL 2005)
This field appears when SQL Server Authentication type is selected. Specifies whether password expiration policy should be enforced on this login.

☑ Check Policy (in Microsoft SQL 2005)
This field appears when SQL Server Authentication type is selected. Applies to SQL Server logins only. Specify that the Windows password policies of the computer on which SQL Server is running should be enforced on this login.

Credential (since Microsoft SQL 2005)
The name of a credential to be mapped to the new SQL Server login. The credential must already exist in the server.

Database
Specify the default database to be assigned to the login. The default database is set to MASTER.

Language
Specify the default language to be assigned to the login. The default language is set to the current default language of the server. If the default language of the server is changed in the future, the default language of the login will remain unchanged.

Type
Specify the type of login authentication. Possible types of authentication are *Windows*, *SQL Server*, *Certificate* and *Asymmetric Key*.

Domain
This field is available when *the Windows Authentication* login type is selected. Specify the Windows domain.

Security Access (up to Microsoft SQL 2005)
You can change the value of this field only if the *Windows Authentication* login type is selected. If *Deny access* value is selected, the login is disabled.

Password
Applies to *SQL Server logins* only. Specify the password for the login that is being created. This value may be supplied already hashed when creating SQL Server logins. Pre-hashed passwords cannot be used when creating Windows logins.

☑ Hashed (in Microsoft SQL 2005)
Applies to *SQL Server logins* only. Specify whether the password entered after the PASSWORD argument is already hashed. If this option is not checked, the string entered as password will be hashed prior to being stored in the database.

☑ Must Change (in Microsoft SQL 2005)

Applies to *SQL Server logins* only. If this option is included, SQL Server will prompt the user for a new password the first time the new login is used.

Certificate (in Microsoft SQL 2005)
Available only when the *Certificate* authentication type selected. Specify the name of a certificate to be associated with this login. This certificate must already exist in the master database.

Asymmetric Key (since Microsoft SQL 2005)
This field is visible only when the *Asymmetric key* authentication type is selected. Specify the name of an asymmetric key to be associated with this login. This key must already exist in the master database.

Server Roles
Define server roles for the login. See Server Roles 258 for details.

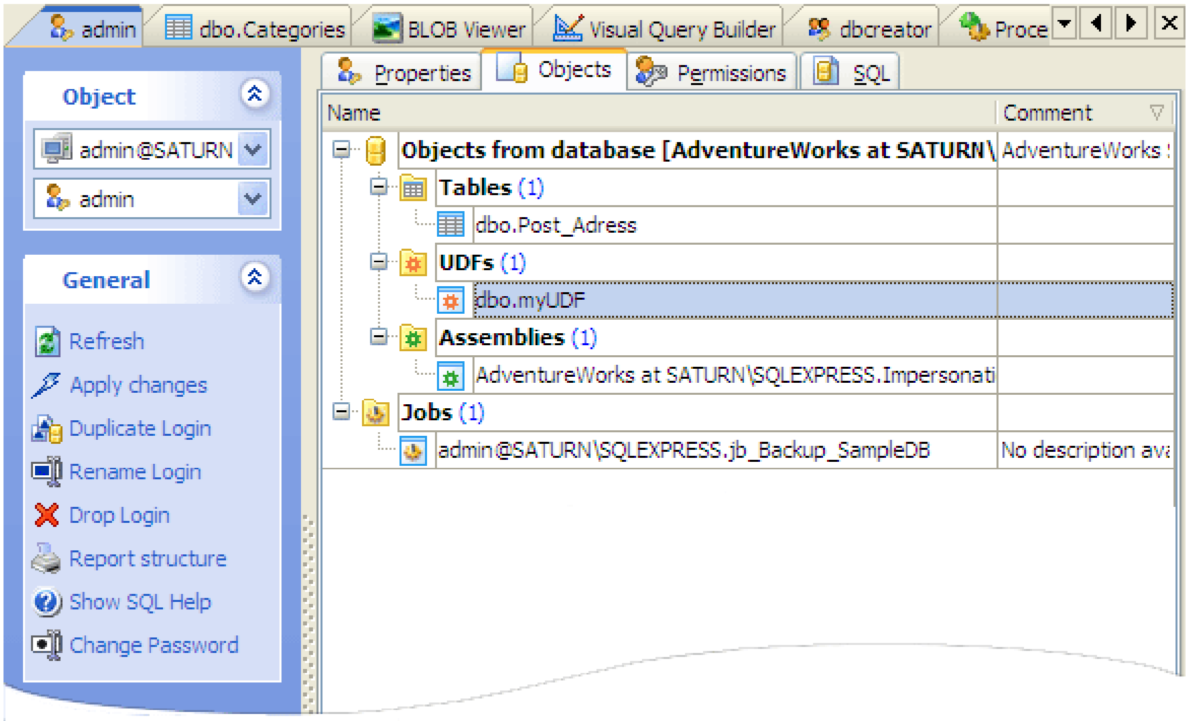
Server permissions
Define actions allowed for the login.

To apply the changes, select the Apply Changes item in the Navigation bar or use **Ctrl+F9** or **Ctrl+F7** shortcut keys.

It is also possible to modify object properties without opening the object editor: use the Object Properties item of the popup menu of the selected object from the explorer tree.

6.3.2.2 Editing login objects

Objects having this login as the owner are managed within the Objects tab of Login Editor. Open an object in the necessary object editor by double-clicking or pressing the **Enter** key. Use grid's popup menu to edit, drop or rename objects.



admin@SATURN
admin

Name	Comment
Objects from database [AdventureWorks at SATURN	AdventureWorks :
Tables (1)	
dbo.Post_Adress	
UDFs (1)	
dbo.myUDF	
Assemblies (1)	
AdventureWorks at SATURN\SQLEXPRESS.Impersonati	
Jobs (1)	
admin@SATURN\SQLEXPRESS.jb_Backup_SampleDB	No description ava

6.4 Server Roles

SQL Server provides server-level roles to help you manage the permissions on a server. Find out more info on server-level roles at SQL Server manual.

SQL Server provides nine fixed server roles. The permissions that are granted to the fixed server roles cannot be changed. Beginning with SQL Server 2012, you can create user-defined server roles and add server-level permissions to the user-defined server roles.
You can add server-level principals (SQL Server logins, Windows accounts, and Windows groups) into server-level roles. Each member of a fixed server role can add other logins to that same role. Members of user-defined server roles cannot add other server principals to the role.

1. *bulkadmin*. Granted: ADMINISTER BULK OPERATIONS.
2. *dbcreator*. Granted: CREATE DATABASE.
3. *diskadmin*. Granted: ALTER RESOURCES.
4. *processadmin*. Granted: ALTER SERVER STATE, ALTER ANY CONNECTION.
5. *securityadmin*. Granted: ALTER ANY LOGIN.
6. *serveradmin*. Granted: ALTER SETTINGS, SHUTDOWN, CREATE ENDPOINT, ALTER SERVER STATE, ALTER ANY ENDPOINT, ALTER RESOURCES.
7. *setupadmin*. Granted: ALTER ANY LINKED SERVER.
8. *sysadmin*. Granted with GRANT option: CONTROL SERVER.

Note: Fixed server roles are server-wide in their scope. Each member of a fixed server role can add other logins within the role.

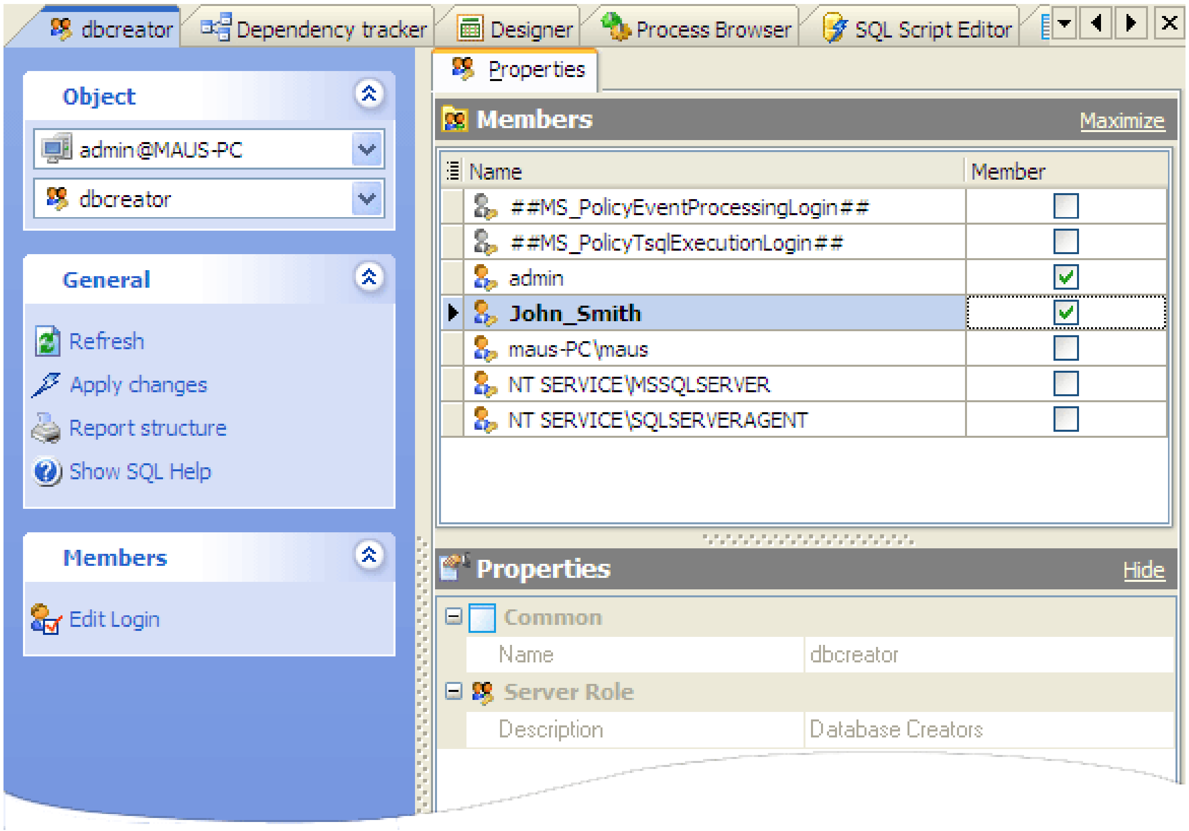
Server roles are edited within Server Roles Editor 259. In order to run the editor you should either

- select the server role for editing in the explorer tree (type the first letters of the server role name for quick search);
- select the Edit Server Role item from the popup menu

or

- open the server in Server Editor and the Server Roles tab there;
- select the server role to edit;
- press the **Enter** key or select the Edit Server Role item from the popup menu (alternatively, you can use the corresponding link of the Navigation Bar).

Server Roles Editor allows you to view the logins implemented to the role, include or exclude existing logins from the definite role.



Name	Member
##MS_PolicyEventProcessingLogin##	☐
##MS_PolicyTsqlExecutionLogin##	☐
admin	☑
John_Smith	☑
maus-PC\maus	☐
NT SERVICE\MSSQLSERVER	☐
NT SERVICE\SQLSERVERAGENT	☐

Common

Name	dbcreator

Server Role

Description	Database Creators

Note: Members of the *securityadmin* fixed server role can grant both server-level and database-level permissions.

The Name and Comment fields represents the server role name and comment respectively.

Logins
Displays the list of all existing logins. If a login name is checked, the login is added to the fixed server role.

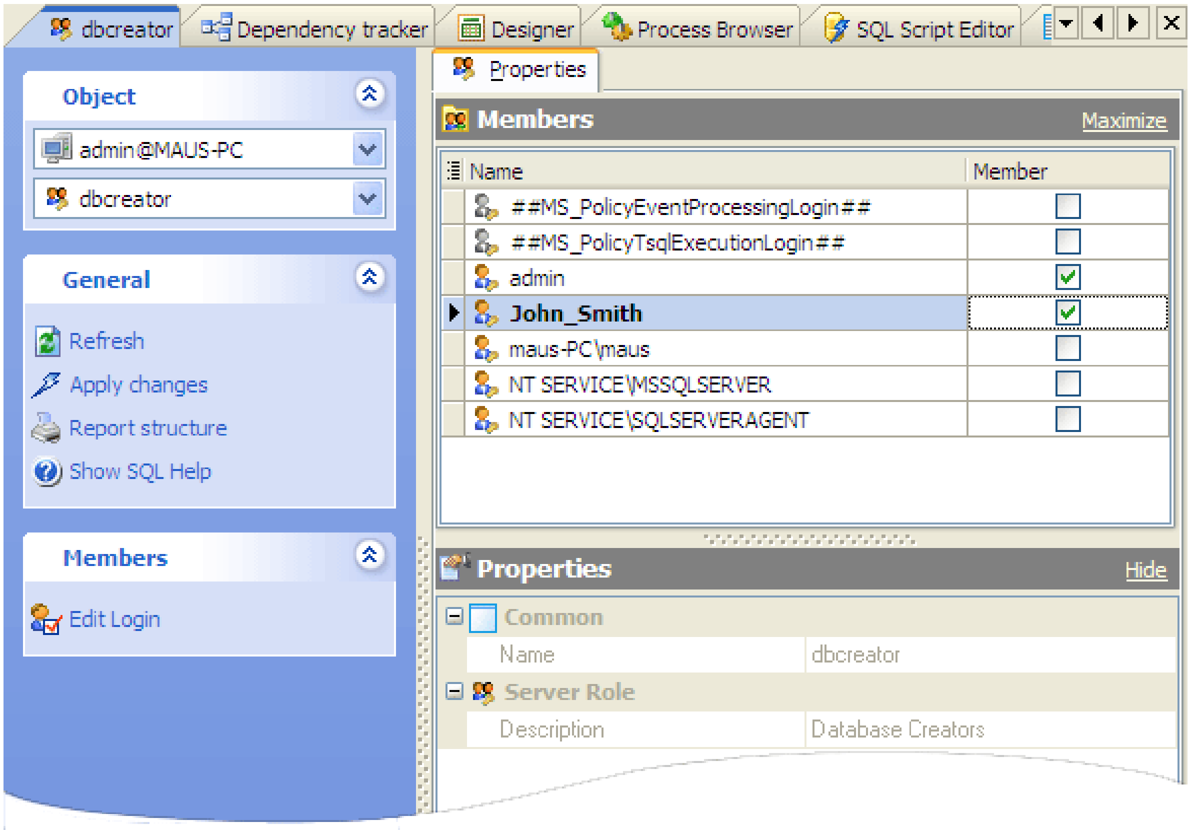
To apply the changes, select the Apply Changes item in the Navigation bar or use **Ctrl+F9** or **Ctrl+F7** shortcut keys.

It is also possible to modify object properties without opening the object editor: use the Object Properties item of the popup menu of the selected object from the explorer tree.

See also: Logins 245

6.4.1 Server Roles Editor

Server Roles Editor allows you to view the logins implemented to the role, include or exclude existing logins from the definite role.



Name	Member
##MS_PolicyEventProcessingLogin##	☐
##MS_PolicyTsqlExecutionLogin##	☐
admin	☑
John_Smith	☑
maus-PC\maus	☐
NT SERVICE\MSSQLSERVER	☐
NT SERVICE\SQLSERVERAGENT	☐

Common	
Name	dbcreator
Server Role	
Description	Database Creators

Note: Members of the *securityadmin* fixed server role can grant both server-level and database-level permissions.

The Name and Comment fields represents the server role name and comment respectively.

Logins
Displays the list of all existing logins. If a login name is checked, the login is added to the fixed server role.

To apply the changes, select the Apply Changes item in the Navigation bar or use **Ctrl+F9** or **Ctrl+F7** shortcut keys.

It is also possible to modify object properties without opening the object editor: use the Object Properties item of the popup menu of the selected object from the explorer tree.

6.5 Server Variables

There are a lot of configuration parameters that affect the behavior of the database system. All parameter (server variables) names are case-insensitive.

Server variables are edited within Server Variable Editor 261. In order to open the editor you should either

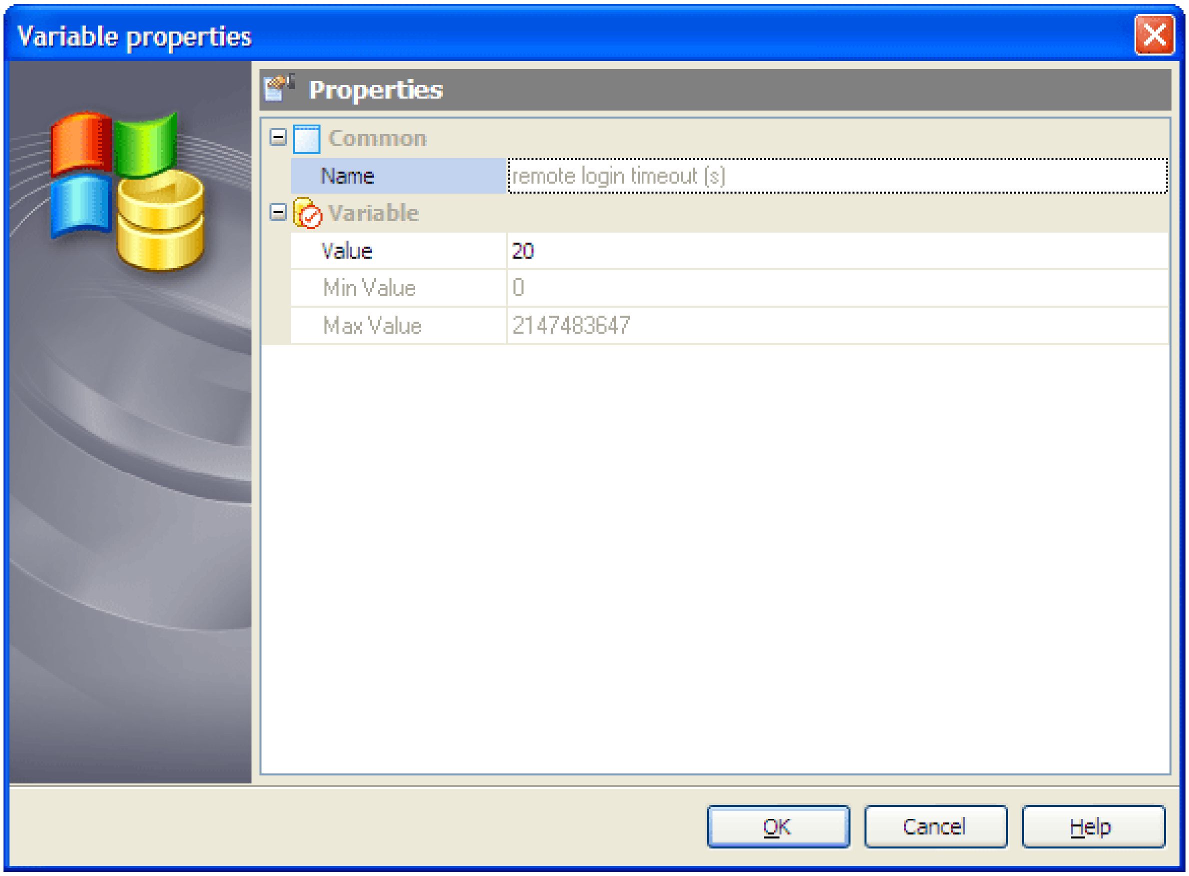
- select the server variables for editing in the explorer tree (type the first letters of the server variable name for quick search);
- select the Edit Server Variable item from the popup menu

or

- open the server in Server Editor and the Server Variables tab there;
- select the server variables to edit;
- press the **Enter** key or select the Edit Server Variable item from the popup menu (alternatively, you can use the corresponding link of the Navigation Bar).

6.5.1 Variable Editor

You can change the Value of server variable.



Name	remote login timeout (s)
Value	20
Min Value	0
Max Value	2147483647

Note: The value (Integer) of variable must be between Min Value and Max Value.

Here you can also edit the variable Name, the current Value of the configuration option, Min Value and Max Value

To apply the changes, select the Apply Changes item in the Navigation bar or use **Ctrl+F9** or **Ctrl+F7** shortcut keys.

It is also possible to modify object properties without opening the object editor: use the Object Properties item of the popup menu of the selected object from the explorer tree.

6.6 Backup Devices

The Backup Device object provides counters to monitor Microsoft SQL Server backup devices used for backup and restore operations. Creating and editing of such devices does not perform any access to the physical device. Access to the specified device only occurs during the backup and restore process.

Creating a logical backup device can simplify backup and restore operations where specifying the *device name* is an alternative using a 'tape' or 'disk' clause to specify the *device path*.

Problems connected with ownership and permissions can interfere with the use of disks or file backup devices. Make sure that the appropriate file permissions are given to the Windows account under which the Database Engine was started.

Note: Creating and editing of Backup Device are required for members of the *diskadmin* fixed server role.

How can I create a new backup device?

New backup devices are created within Create Backup Device Wizard 264. In order to run the wizard you should either

- select the Backup Devices list or any object from that list and then use the Create New Backup Device item from the popup menu

or

- open the server in Server Editor and the Backup Devices tab there and press **Insert** or select the Create New Backup Device item from the popup menu (Alternatively, use the corresponding link of the Navigation Bar).

How can I edit an existing backup device?

Backup devices are edited within Backup Device Editor 265. In order to open the editor you should either

- select the backup device for editing in the explorer tree (type the first letters of the backup device name for quick search);
- select the Edit Backup Device item from the popup menu

or

- open the server in Server Editor and the Backup Devices tab there;
- select the backup device to edit;
- press the **Enter** key or select the Edit Backup Device item from the popup menu (alternatively, you can use the corresponding link of the Navigation Bar).

How can I drop a backup device?

To drop the existing backup device:

- select the backup device to drop in the explorer tree;
- select the Drop Backup Device item from the popup menu

or

- open the server in Server Editor and the Backup Devices tab there;
- select the backup device to drop;
- press the **Delete** key or select the Drop Backup Device item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.

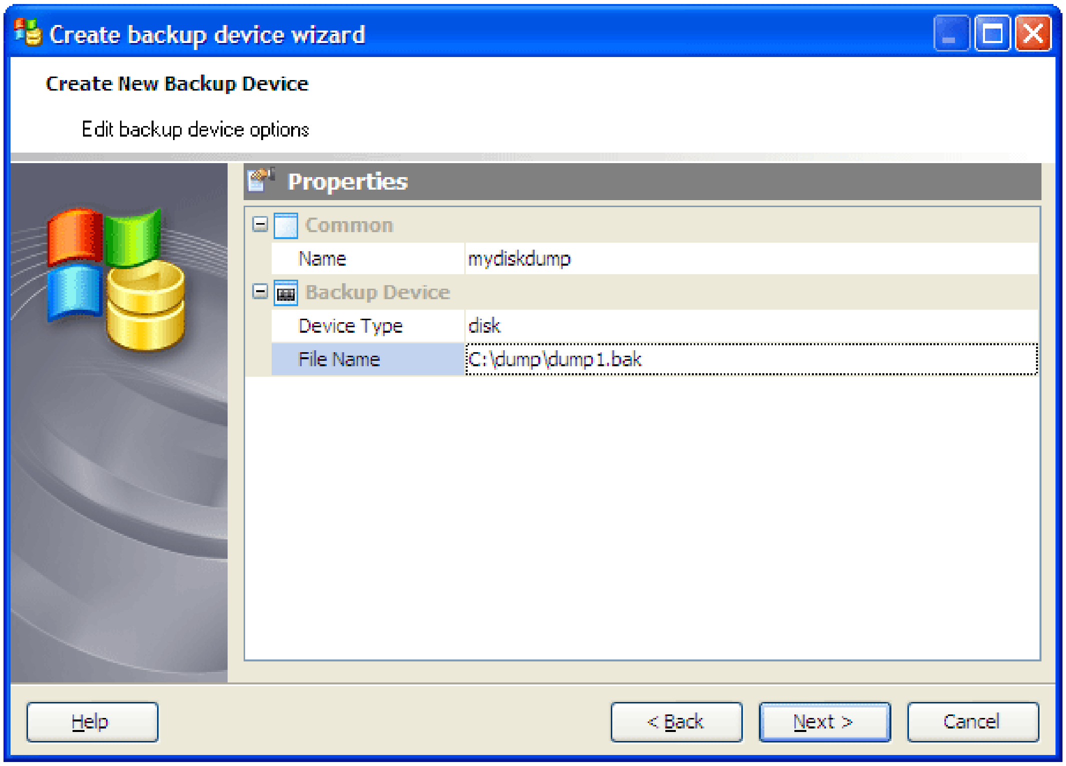
6.6.1 Create Backup Device Wizard

Create Backup Device Wizard guides you through the process of creating a new backup device. See How To Create Backup Device 263 for instructions on running this wizard.

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

- Specifying backup device options 290

See also: Backup Device Editor 291



Common

Name: mydiskdump

Backup Device

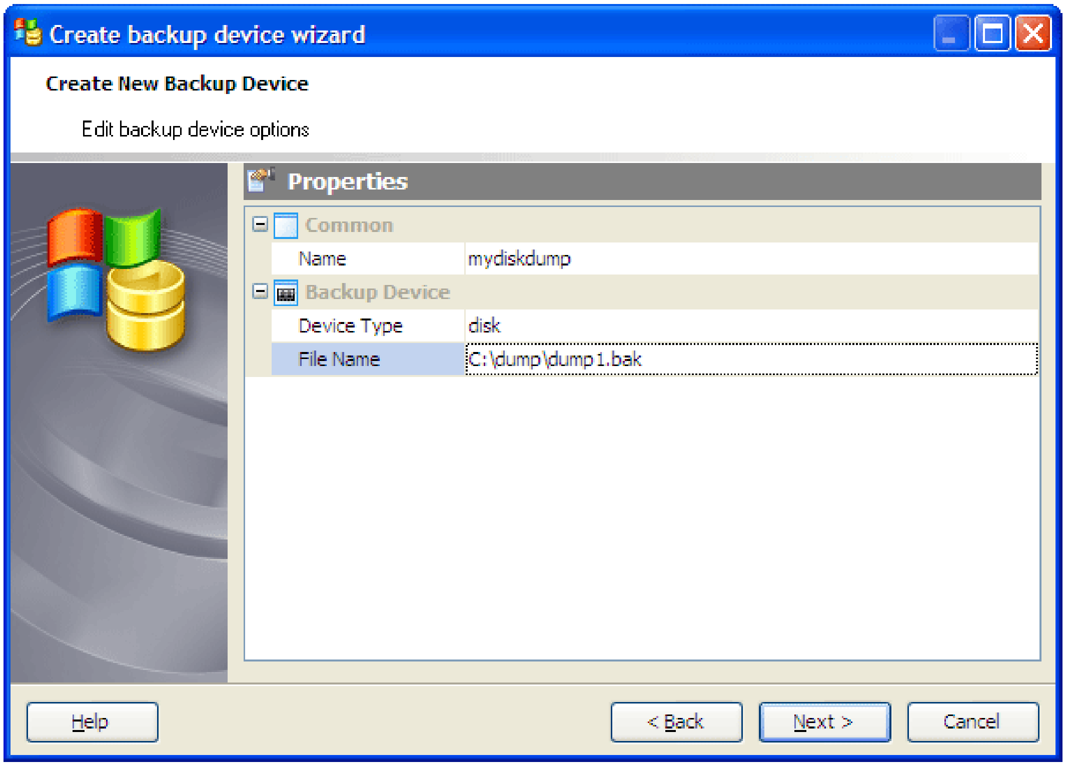
Device Type: disk

File Name: C:\dump\dump1.bak

6.6.1.1 Specifying backup device options

This step of the wizard allows you to select Device Type from the drop-down list. The available values: *disk*, *tape* or *pipe*. Enter the physical name of the backup device in the

File Name field. Physical names must follow the rules for operating-system file names or universal naming conventions for network devices and must include the full path. If you add a *tape* device, this parameter must be the physical name assigned to the local tape device by Windows; for example, \\.\TAPE0 for the first tape device on the computer. The tape device must be attached to the server computer; it cannot be used remotely. The names containing nonalphanumeric characters are to be enclosed in quotation marks.



Name: mydiskdump
Device Type: disk
File Name: C:\dump\dump1.bak

Note: When creating a backup device on a remote network location, be sure that the name under which the Database Engine was started has the appropriate write capabilities on the remote computer.

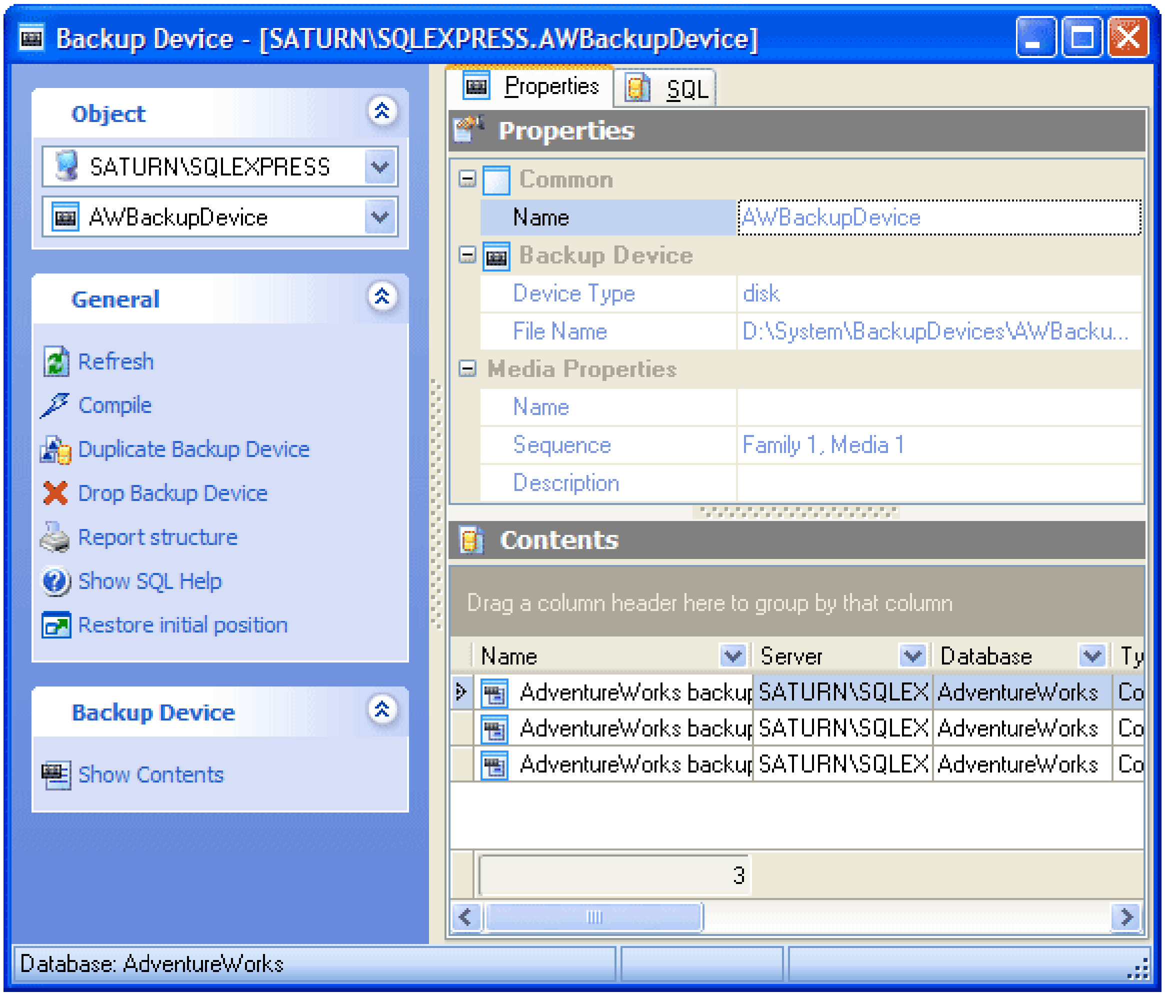
6.6.2 Backup Device Editor

Backup Device Editor allows you to browse its content and properties, view the SQL statement for creating the backup device, etc. It can be opened automatically when you create a new backup device or browse the existing one (see How to edit category 263 for details).

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.

- Viewing backup device properties 266

See also: Create Backup Device Wizard 264



SATURN\SQLEXPRESS

AWBackupDevice

Common

Name: AWBackupDevice

Backup Device

Device Type: disk

File Name: D:\System\BackupDevices\AWBacku...

Media Properties

Name:

Sequence: Family 1, Media 1

Description:

Name	Server	Database	Ty
AdventureWorks backup	SATURN\SQLEX	AdventureWorks	Co
AdventureWorks backup	SATURN\SQLEX	AdventureWorks	Co
AdventureWorks backup	SATURN\SQLEX	AdventureWorks	Co

3

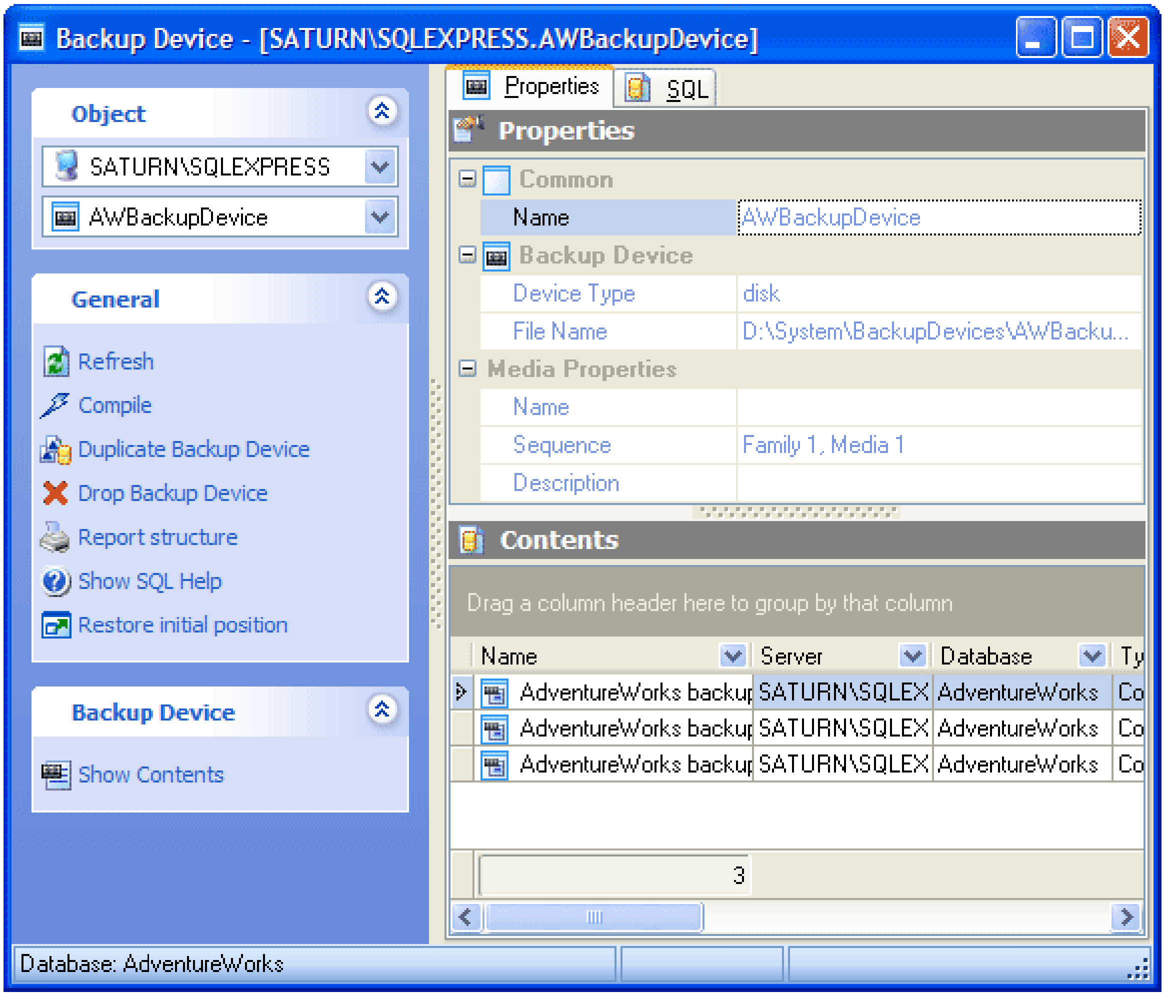
Database: AdventureWorks

6.6.2.1 Viewing backup device properties

Within the Properties tab you can find logical Name and the physical File Name of the backup device, Device Type.

To view backup media properties such as *Name, Sequence* and *Description*, use the Show Contents link of the Navigation bar.

The tab is also designed for convenient backup review on this device. The backups are displayed in the Contents grid. Use the Show Contents link of the Navigation bar. Using the grid you can filter and sort the backups by *Name, Server, Database, Type, Date, Comment, Expiration* and *Size*.



Name	AWBackupDevice
Device Type	disk
File Name	D:\System\BackupDevices\AWBacku...
Name	
Sequence	Family 1, Media 1
Description	

Name	Server	Database	Ty
AdventureWorks backu	SATURN\SQLEX	AdventureWorks	Co
AdventureWorks backu	SATURN\SQLEX	AdventureWorks	Co
AdventureWorks backu	SATURN\SQLEX	AdventureWorks	Co
3			

Database: AdventureWorks

To apply the changes, select the Apply Changes item in the Navigation bar or use **Ctrl+F9** or **Ctrl+F7** shortcut keys.

It is also possible to modify object properties without opening the object editor: use the Object Properties item of the popup menu of the selected object from the explorer tree.

6.7 Schedules

Scheduling administrative jobs[271] means defining the condition(s) that cause the job to begin running. Using these server objects you can schedule any type of job. The same job schedule can be used by more than one job. A user can attach and detach schedules from jobs. The object is available in Microsoft SQL server 2005 or higher.

Only one instance of the job can be run at a certain moment. If you try to run a job manually while it is running as scheduled, SQL Server Agent refuses the request.

To prevent a scheduled job from running, you should either disable the schedule using the corresponding option of the Schedule Editor dialog window, or disable the job using Job Editor, drop the schedule using the Drop Schedule item of the popup menu in the Available Schedules field of Job Editor, or stop the *SQLSERVERAGENT* service on the server. When a job schedule is not enabled, the schedule is not enabled for any job that uses the schedule.
You must explicitly re-enable a schedule that has been disabled.

Note: Jobs can only run when the *SQLSERVERAGENT* service is currently running on the server.

How can I create a new schedule?

New schedules are created within Schedule Editor[269]. In order to open this editor you should either

- select the Schedules list or any object from that list and then use the Create New Schedule item from the popup menu

or

- open the server in Server Editor and the Schedules tab there and press **Insert** or select the Create New Schedule item from the popup menu (Alternatively, use the corresponding link of the Navigation Bar).

How can I edit an existing schedule?

Schedules are edited within Schedule Editor[269]. In order to open the editor you should either

- select the schedule for editing in the explorer tree (type the first letters of the schedule name for quick search);
- select the Edit Schedule item from the popup menu

or

- open the server in Server Editor and the Schedules tab there;
- select the schedule to edit;
- press the **Enter** key or select the Edit Schedule item from the popup menu (alternatively, you can use the corresponding link of the Navigation Bar).

How can I drop a schedule?

To drop the existing schedule:

- select the schedule to drop in the explorer tree;
- select the Drop Schedule item from the popup menu

or

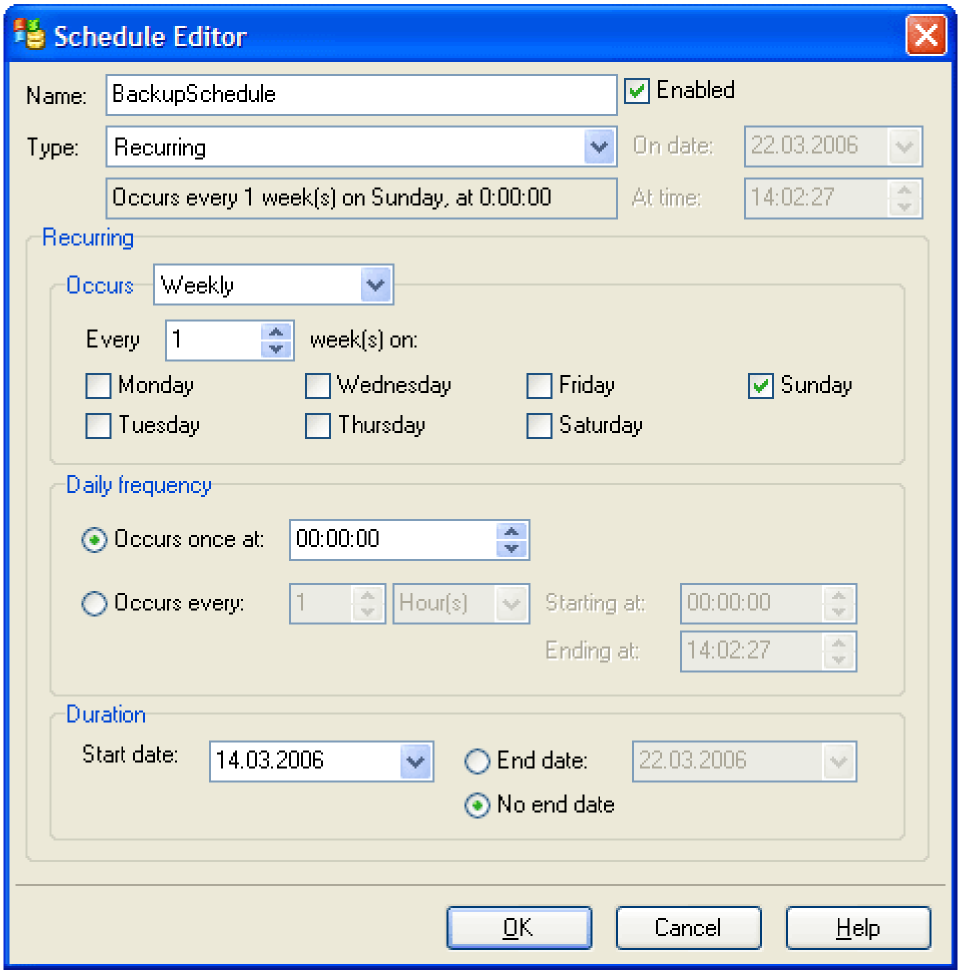
- open the server in Server Editor and the Schedules tab there;
- select the schedule to drop;
- press the **Delete** key or select the Drop Schedule item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.

6.7.1 Schedule Editor

Schedule Editor allows you to set schedule properties. The detailed description is given below.

Note: If the schedule is disabled editing the schedule does not automatically re-enable the schedule.



Name: BackupSchedule
☑ Enabled
Type: Recurring
On date: 22.03.2006
Occurs every 1 week(s) on Sunday, at 0:00:00
At time: 14:02:27
Occurs: Weekly
Every 1 week(s) on:
☐ Monday ☐ Wednesday ☐ Friday ☑ Sunday
☐ Tuesday ☐ Thursday ☐ Saturday
● Occurs once at: 00:00:00
○ Occurs every: 1 Hour(s) Starting at: 00:00:00
Ending at: 14:02:27
Start date: 14.03.2006
○ End date: 22.03.2006
● No end date

Name
The field stores the name of the schedule.

☑ Enabled
The option indicates the current status of the schedule. If not checked, the schedule is not enabled. When the schedule is not enabled, no jobs will run according to this schedule.

Type
Specifies when a job is to be executed.

For Schedule Type, select one of the following:
- Click *Start automatically when SQL Server Agent starts* to start the job when the SQL Server Agent service is started.

- Click *Start whenever the CPUs become idle* to start the job when the CPUs reach an idle condition.

- Click *Recurring* if you want a schedule to run repeatedly. To set the recurring schedule, complete the Frequency, Daily Frequency, and Duration groups on the dialog.

- Click *One time* if you want the schedule to run only once. To set the One time schedule, complete the One-time occurrence group on the dialog.

On date
For the One time schedule type, defines the date when the job is to be executed.

At time
For the One time schedule type, defines the time when the job is to be executed.

Recurring
Defines time tunings for schedule when the Recurring schedule type is selected.

To apply the changes, select Ok button, to reject - Cancel.

6.8 Jobs

The job object exposes the attributes of a single *SQL Server Agent* job. Using *SQL Server Agent* jobs, you can automate administrative tasks and run them on a recurring basis. You can run a job manually or schedule it to run in response to schedules 268 and alerts 294. Jobs can be written to run on the local instance of Microsoft SQL or on multiple servers. To run jobs on multiple servers, you must set up at least one *master* server and one or more target servers.

Anyone can create a job, but a job can be edited only by its owner or members of the *sysadmin* role.

Only one instance of the job can be run at a time. If you try to run a job manually while it is running as scheduled, *SQL Server Agent* refuses the request.

To prevent a scheduled job from running, you should either disable the schedule using the corresponding option of the Schedule Editor dialog window, or disable the job using Job Editor, drop the schedule using the Drop Schedule item of the popup menu in the Available Schedules field of Job Editor, or stop the *SQLSERVERAGENT* service on the server.

> **Note:** Before starting a job make sure that the *SQLSERVERAGENT* service is currently running on the server.

How can I create a new job?

New jobs are created within Create Job Wizard 272. In order to run this wizard you should either

- select the Jobs list or any object from that list and then use the Create New Job item from the popup menu

or

- open the server in Server Editor and the Jobs tab there and press **Insert** or select the Create New Job item from the popup menu (Alternatively, use the corresponding link of the Navigation Bar).

How can I edit an existing job?

Jobs are edited within Job Editor 278. In order to open the editor you should either

- select the job for editing in the explorer tree (type the first letters of the job name for quick search);
- select the Edit Job item from the popup menu

or

- open the server in Server Editor and the Jobs tab there;
- select the job to edit;
- press the **Enter** key or select the Edit Job item from the popup menu (alternatively, you can use the corresponding link of the Navigation Bar).

⊟ How can I drop a job?

To drop the existing job:

- select the job to drop in the explorer tree;
- select the Drop Job item from the popup menu

or

- open the server in Server Editor and the Jobs tab there;
- select the job to drop;
- press the **Delete** key or select the Drop Job item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.

A job step is an action that the job takes on a database or a server. Every job must have at least one job step. Job steps can be operating system commands, Transact-SQL statements, Microsoft® ActiveX® scripts, or replication tasks.

- Create Step Wizard 275
- Job Step Editor 281

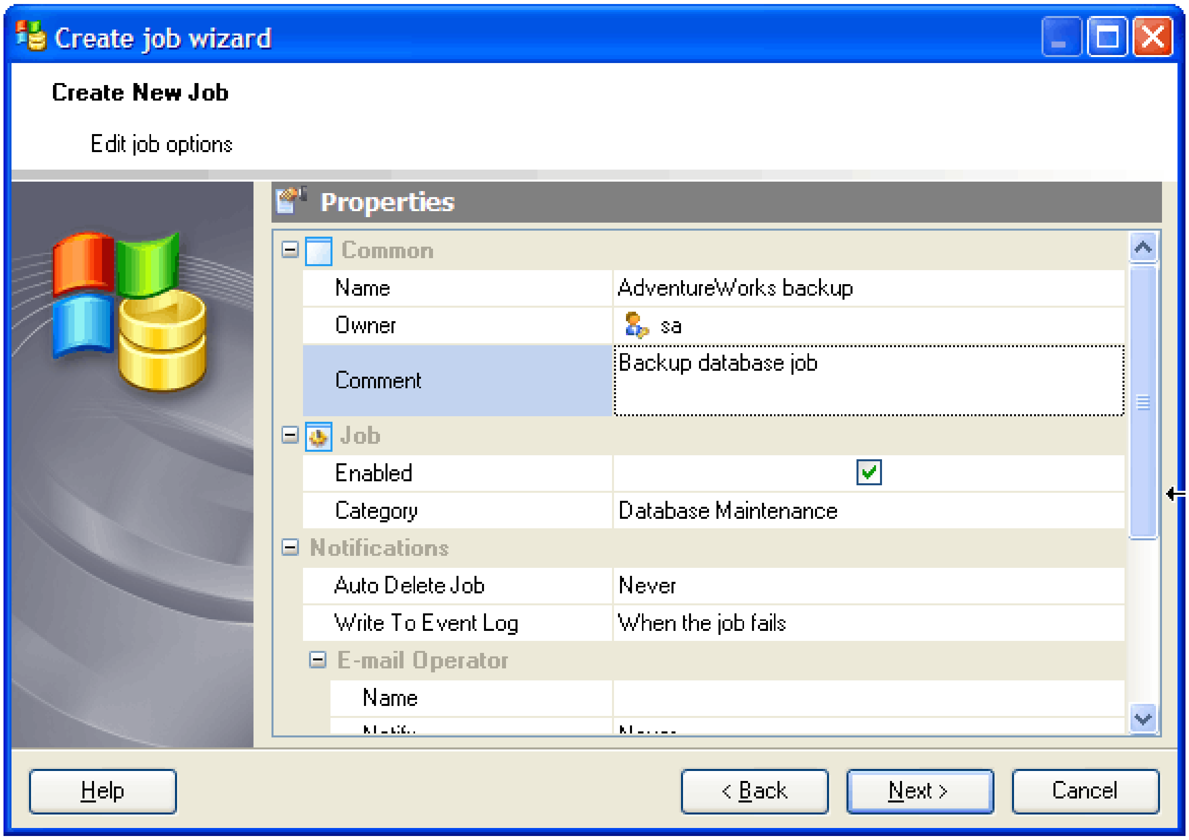
6.8.1 Create Job Wizard

Create Job Wizard guides you through the process of creating a new job. See How To Create Job 271 for instructions on running the wizard.

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

- Specifying job options 273
- Managing job steps 274

See also: Job Editor 278



Create New Job

Edit job options

Properties

Common	
Name	AdventureWorks backup
Owner	sa
Comment	Backup database job
Job	
Enabled	☑
Category	Database Maintenance
Notifications	
Auto Delete Job	Never
Write To Event Log	When the job fails
E-mail Operator	
Name	

6.8.1.1 Specifying job options

Specify job options according to your needs. The detailed description is given below.

Owner
Define the owner of the job.

Comment
Specify a comment to the job (optional).

Enabled
Indicates the status of the added job. If not checked, the job is not enabled and does not run according to its schedule; however, it can be run manually.

Category
The category for the job. See Categories 289 for details.

Auto Delete Job
Select the condition under which the job is to be deleted: *Never* (set by default), *when the job succeeds*, *when the job fails*, *whenever the job completes*.

Write To Event Log
Select the condition under which the entry is to be placed in the Microsoft Windows NT application log for this job: *Never* (set by default), *when the job succeeds*, *when the job fails*, *whenever the job completes*.

E-mail Operator

Name
Set the e-mail name to send e-mail notifications upon job execution (optional).

Notify
Select the condition under which the e-mail notification is to be sent: *Never* (set by default), *when the job succeeds*, *when the job fails*, *whenever the job completes*.

Page Operator
Name
Set the page operator name to send notifications upon job execution (optional).

Notify
Select the condition under which the notification is to be performed: *Never* (set by default), *when the job succeeds*, *when the job fails*, *whenever the job completes*.

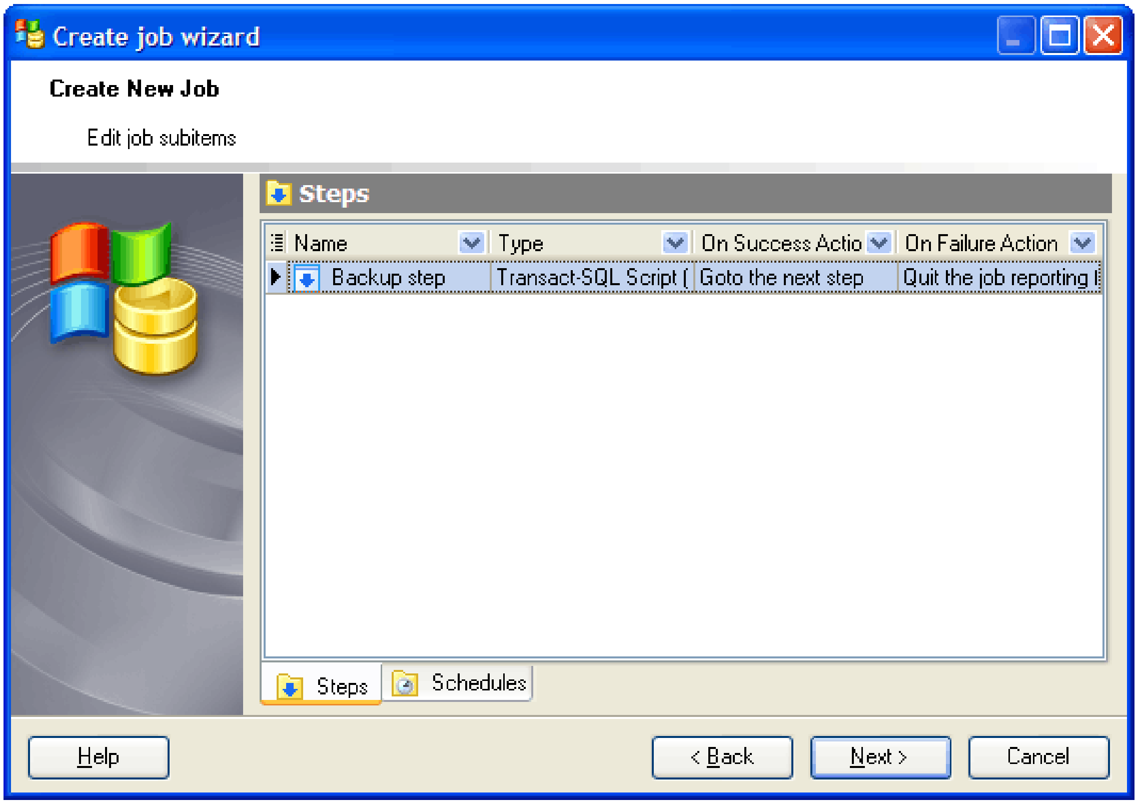
Net send Operator
Name
The name of the person to send a network message to upon completion of this job (optional).

Notify
Select the condition under which the notification is to be performed: *Never* (set by default), *when the job succeeds*, *when the job fails*, *whenever the job completes*.

6.8.1.2 Managing job subitems

On this step of the wizard you can specify the subobjects of the job being created.
To add a new object:

- Choose a necessary page (Job Steps - to specify job steps, Schedulers - to set the condition(s) that cause the job to begin running);
- Using pop-up menu open the appropriate Create Object Wizard;
- Specify new object properties.Click the Add button to add a new step for a job and set its properties in Create Step Wizard 275. Click the Edit and Delete buttons to edit or delete the selected step.



Name	Type	On Success Actio	On Failure Action
Backup step	Transact-SQL Script (	Goto the next step	Quit the job reporting f

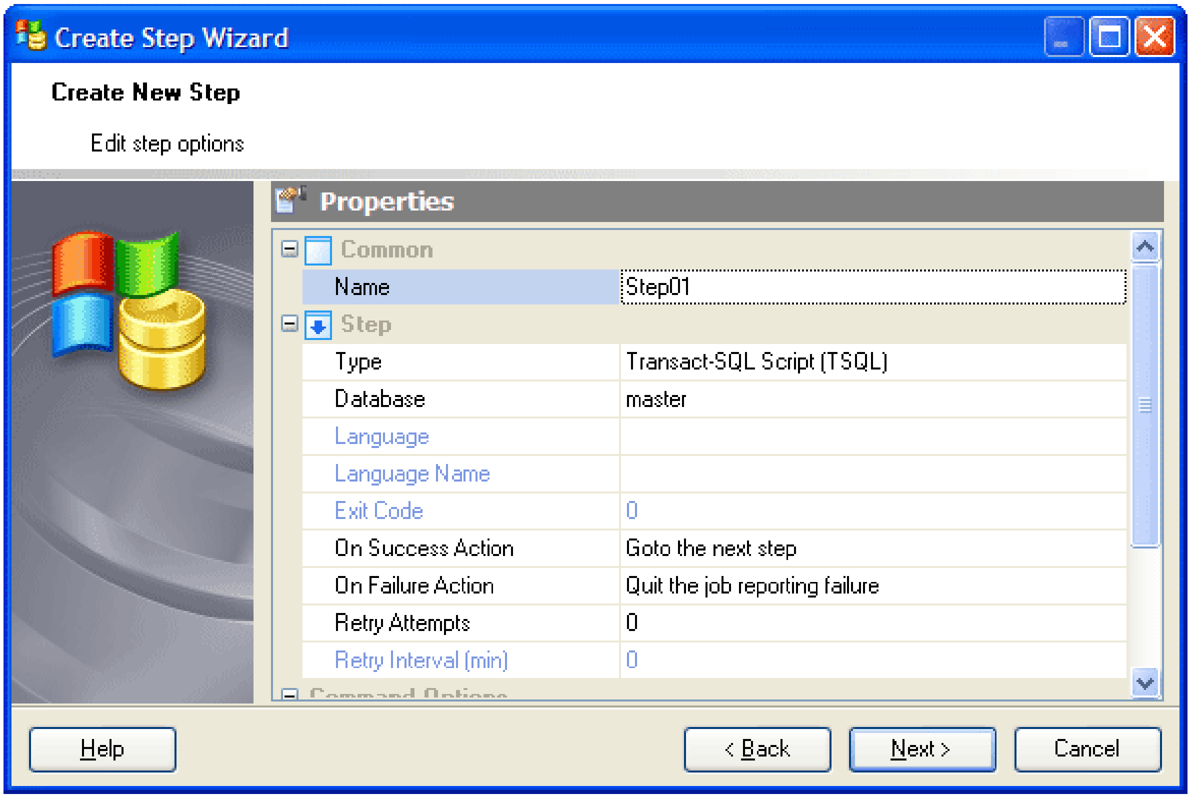
6.8.1.3 Create Job Step Wizard

Create Step Wizard guides you through the process of building the initial step and setting its properties.

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

- Specifying step options 276
- Specifying job step definition 277

See also: Job Step Editor 281



Common	
Name	Step01
Step	
Type	Transact-SQL Script (TSQL)
Database	master
Language	
Language Name	
Exit Code	0
On Success Action	Goto the next step
On Failure Action	Quit the job reporting failure
Retry Attempts	0
Retry Interval (min)	0
Command Options	

6.8.1.3.1 Specifying step options

Specify job step options according to your needs. The detailed description is given below.

Name
The name of the job step.

Type
The field refers to the subsystem used by Microsoft SQL Server Agent service to execute commands.

Database
The name of the database in which a Transact-SQL step is to be executed.

Language
For an ActiveX job step, the field represents the name of the scripting language that the step uses.

Language Name
For an ActiveX job step, the field represents the name of the scripting language that the step uses if Language is selected as Other.

Exit Code
The value returned by a CmdExec subsystem command to indicate that command executed successfully.

On Success Action

Specify the action to be performed if the step succeeds.

On Failure Action
Specify the action to be performed if the step fails.

Retry Attempts
The number of retry attempts to use if the step fails.

Retry Interval (min)
The amount of time between retry attempts (in minutes).

Output File
The name of the file in which the output of this step is saved. This parameter is valid only with commands running on the Transact-SQL or CmdExec subsystems.

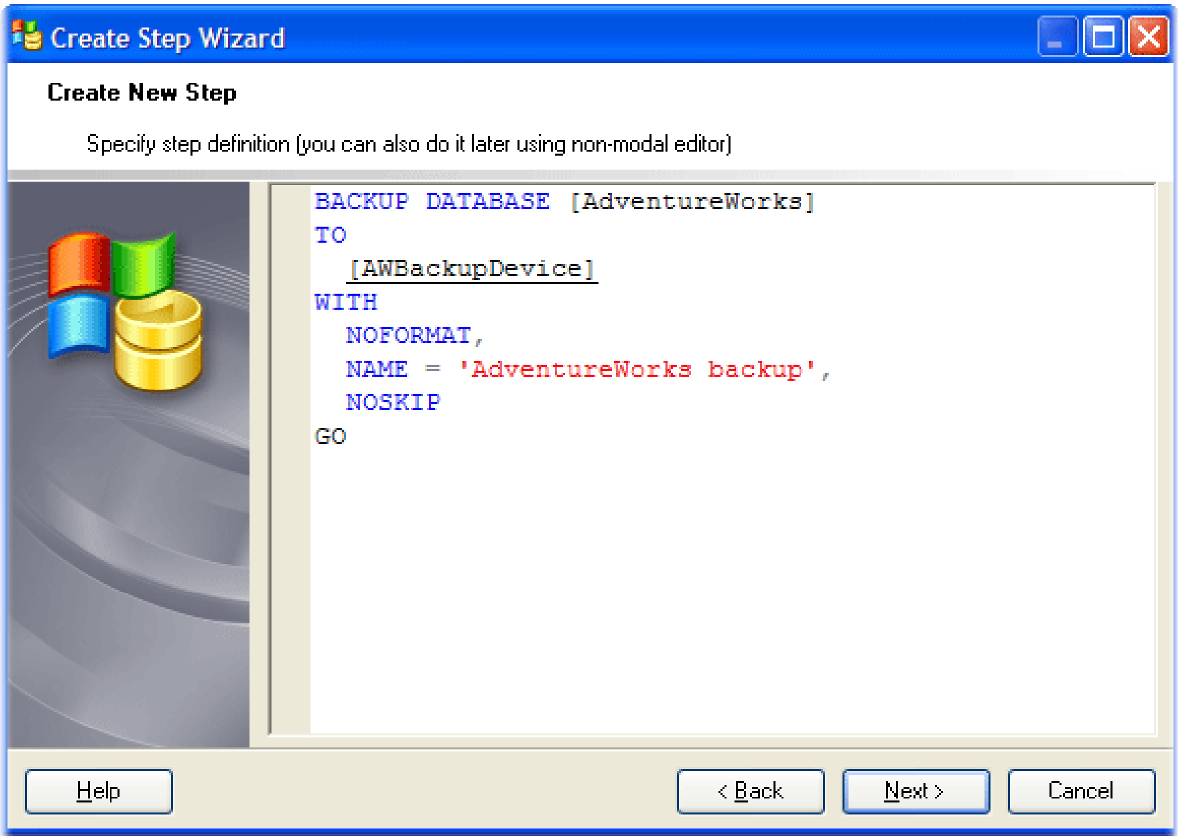
Output File Behaviour
Use the option to control output file behavior.

Write To History
Write Transact-SQL job step output to step history (Transact-SQL job steps only).

User Name
The name of the user account to use when executing a TSQL step.

6.8.1.3.2 Specifying job step definition

At this step you can specify the SQL definition for the new job step. The step is optional: you can do it later using a non-modal editor.


Create Step Wizard
Create New Step
Specify step definition (you can also do it later using non-modal editor)
Help
< Back
Next >
Cancel

```
BACKUP DATABASE [AdventureWorks]
TO
  [AWBackupDevice]
WITH
  NOFORMAT,
  NAME = 'AdventureWorks backup',
  NOSKIP
GO
```

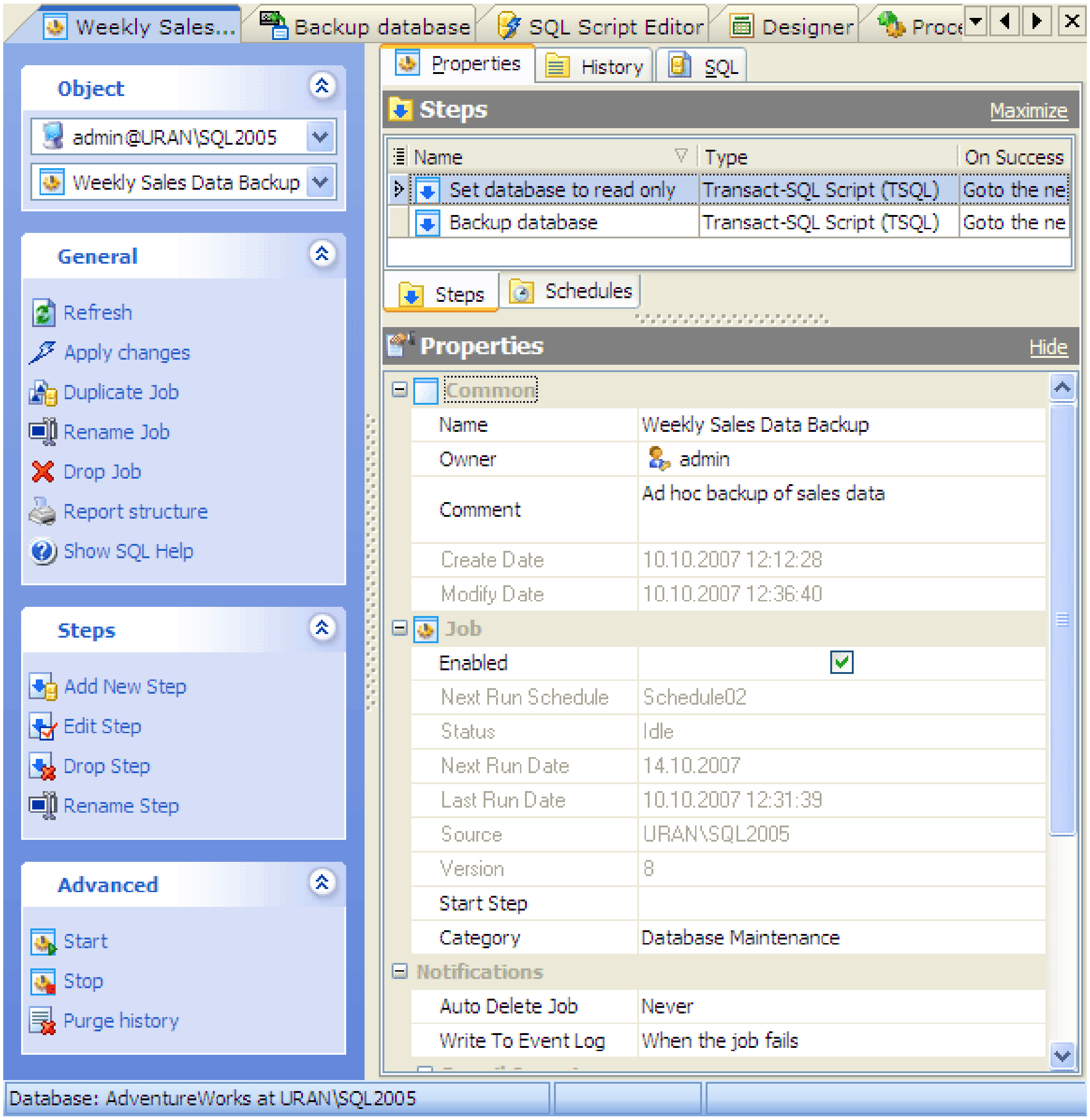
6.8.2 Job Editor

Job Editor allows you to edit job properties, view the SQL statement for creating a job, manage steps, schedules, etc.

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.

- Editing job properties 279

See also: Create Job Wizard 272



admin@URAN\SQL2005

Weekly Sales Data Backup

Name	Type	On Success
Set database to read only	Transact-SQL Script (TSQL)	Goto the ne
Backup database	Transact-SQL Script (TSQL)	Goto the ne

Common	
Name	Weekly Sales Data Backup
Owner	admin
Comment	Ad hoc backup of sales data
Create Date	10.10.2007 12:12:28
Modify Date	10.10.2007 12:36:40
Job	
Enabled	☑
Next Run Schedule	Schedule02
Status	Idle
Next Run Date	14.10.2007
Last Run Date	10.10.2007 12:31:39
Source	URAN\SQL2005
Version	8
Start Step	
Category	Database Maintenance
Notifications	
Auto Delete Job	Never
Write To Event Log	When the job fails

Database: AdventureWorks at URAN\SQL2005

6.8.2.1 Editing job properties

The Properties tab is provided for managing general job options. There are Steps, Schedules and Properties parts there.

The Steps tab displays the job steps. Use Step Editor 281 for editing the step. The popup menu of this tab allows you to create a new step, edit, drop or rename the existing steps.

The Schedules grid allows you to define the condition(s) that cause the job to begin running. Just select a job schedule to edit and use Schedule Editor for editing the schedule. The popup menu of the selected schedule allows you to attach available schedules to the current job, detach schedule, create, edit, drop or rename schedules. For more information see Schedules 268.

All the job common properties are available for editing on the bottom of the Properties window. Here you can modify the job's name, owner, and the comment to the job.

The Create Date field indicates the date when job was created.

The Modify Date field indicates the date when job was last modified.

Next Run Schedule indicates the specific date of the next job running.

You can also find here the job status, the last job run date, and the next job run date

The Version box contains the version of the job (automatically updated each time the job is modified).

Start Step defines the step name to start with.

Enabled indicates the status of the added job. If not checked, the job is not enabled and does not run according to its schedule; however, it can be run manually.

Category represents the job category. See Categories 289 for details.

Auto Delete Job
A value that indicates when the job is to be deleted.

Write To Event Log
A value indicating whether an entry in the Microsoft Windows NT application log for this job is to be placed.

E-mail Operator
Name
The e-mail of the person to send e-mail to when *email_level* is reached.

Notify
A value that indicates when an e-mail upon the completion of this job is to be sent.

Page Operator
Name
The name of the operator to whom the network message is sent upon completion of this job.

Notify
A value that indicates when a network message upon the completion of this job is to be sent.

Net send Operator
Name
The name of the person to page upon completion of this job.

Notify
A value that indicates when a page upon the completion of this job is to be sent.

To apply the changes, select the Apply Changes item in the Navigation bar or use **Ctrl+F9** or **Ctrl+F7** shortcut keys.

It is also possible to modify object properties without opening the object editor: use the Object Properties item of the popup menu of the selected object from the explorer tree.

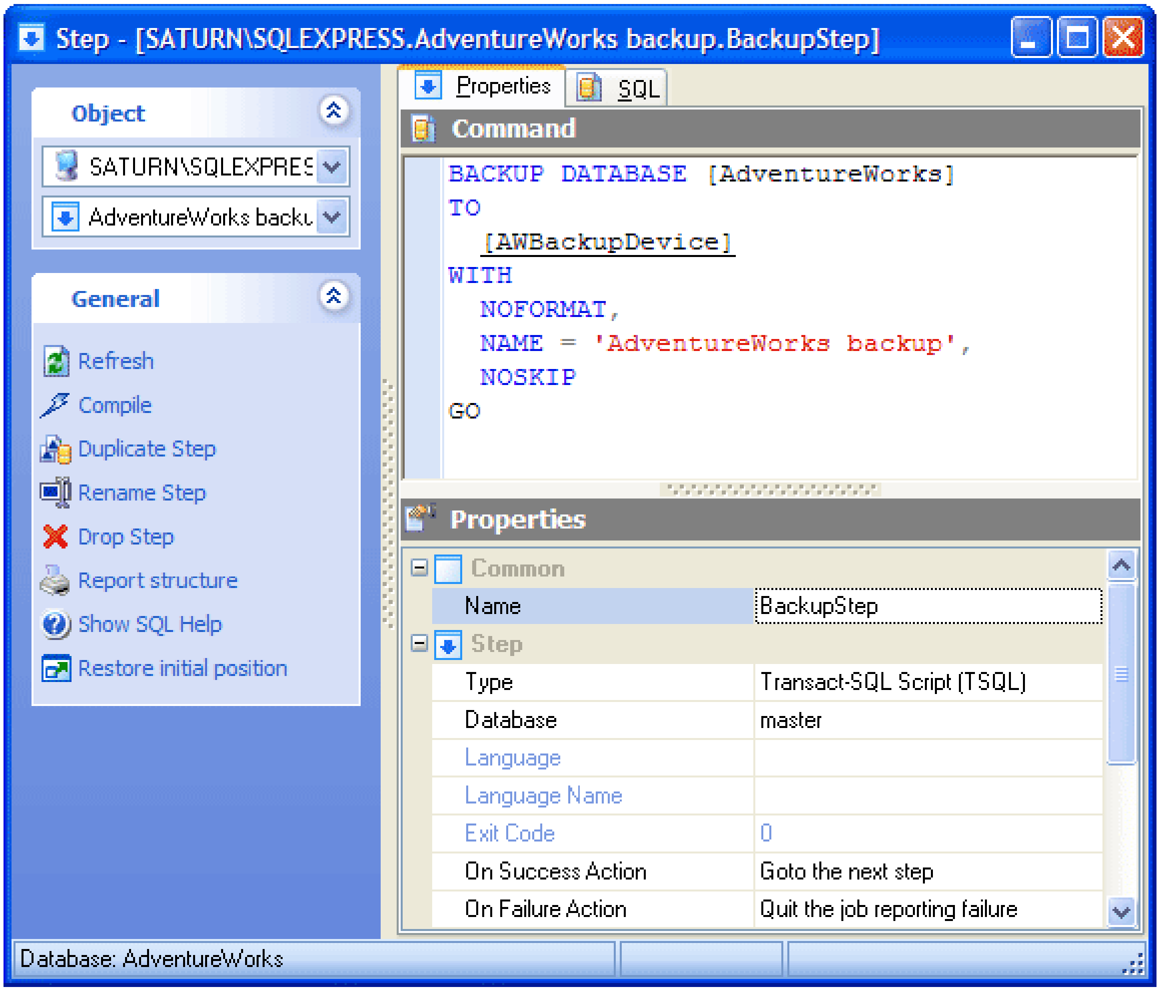
6.8.2.2 Job Step Editor

Job Step Editor allows you to edit step properties, its dependences, view the Transact-SQL statement for creating the step, etc. It can be opened automatically when you create a new job or edit the existing one.

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.

- Editing step properties 281

See also: Create Step Wizard 275



```
BACKUP DATABASE [AdventureWorks]
TO
  [AWBackupDevice]
WITH
  NOFORMAT,
  NAME = 'AdventureWorks backup',
  NOSKIP
GO
```

Name	BackupStep
Type	Transact-SQL Script (TSQL)
Database	master
Language	
Language Name	
Exit Code	0
On Success Action	Goto the next step
On Failure Action	Quit the job reporting failure

Database: AdventureWorks

6.8.2.2.1 Editing step properties

Specify job options according to your needs. The detailed description is given below.

Command
Specify the commands to be executed by SQLServerAgent service through subsystem.

Name
The field contains the name of the step.

Type
The field refers to the subsystem used by Microsoft SQL Server Agent service to execute commands.

Database
The name of the database in which a Transact-SQL step is to be executed.

Language
For an ActiveX job step, the field represents the name of the scripting language that the step uses.

Language Name
For an ActiveX job step, it is the name of the scripting language that the step uses if Language is selected as Other.

Exit Code
The value returned by a CmdExec subsystem command to indicate that the command executed successfully.

On Success Action
Specify the action to be performed if the step succeeds.

On Failure Action
Specify the action to be performed if the step fails.

Retry Attempts
Specify the number of retry attempts to use if this step fails.

Retry Interval (min)
Specify the amount of time in minutes between retry attempts.

Output File
Set the name of the file in which the output of this step is saved. This parameter is valid only with commands running on the Transact-SQL or CmdExec subsystems.

Output File Behaviour
Use the option to control the output file behavior.

Write To History
Write Transact-SQL job step output to step history (Transact-SQL job steps only).

User Name
The name of the user account to use when executing a TSQL step.

To apply the changes, select the Apply Changes item in the Navigation bar or use **Ctrl+F9** or **Ctrl+F7** shortcut keys.

It is also possible to modify object properties without opening the object editor: use the Object Properties item of the popup menu of the selected object from the explorer tree.

6.9 Operators

An operator is notification recipient that is used with alerts 294 and jobs 271.

Note: Before starting a job make sure that the *SQLSERVERAGENT* service is currently running on the server.

How can I create a new operator?

New operators are created within Create Operator Wizard 284. In order to run this wizard you should either

- select the Operators list or any object from that list and then use the Create New Operator item from the popup menu

or

- open the server in Server Editor and the Operators tab there and press **Insert** or select the Create New Operator item from the popup menu (Alternatively, use the corresponding link of the Navigation Bar).

How can I edit an existing operator?

Operators are edited within Operator Editor 286. In order to open the editor you should either

- select the operator for editing in the explorer tree (type the first letters of the operator name for quick search);
- select the Edit Operator item from the popup menu

or

- open the server in Server Editor and the Operators tab there;
- select the operator to edit;
- press the **Enter** key or select the Edit Operator item from the popup menu (alternatively, you can use the corresponding link of the Navigation Bar).

How can I drop an operator?

To drop the existing operator:

- select the operator to drop in the explorer tree;
- select the Drop Operator item from the popup menu

or

- open the server in Server Editor and the Operators tab there;
- select the operator to drop;
- press the **Delete** key or select the Drop Operator item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.

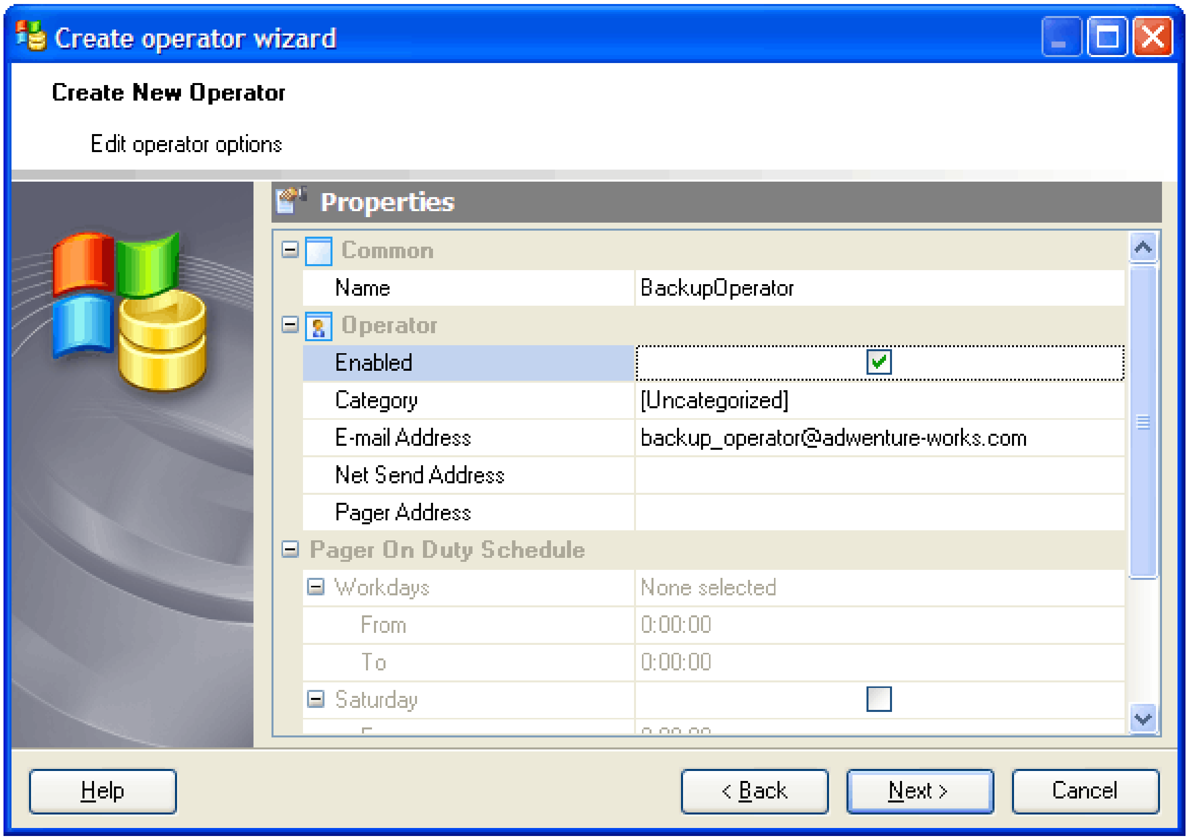
6.9.1 Create Operator Wizard

Create Operator Wizard guides you through the process of creating a new server operator. See How To Create operator 283 to learn how to run this wizard.

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

- Specifying operator options 284
- Managing operator notifications 286

See also: Operator Editor 286



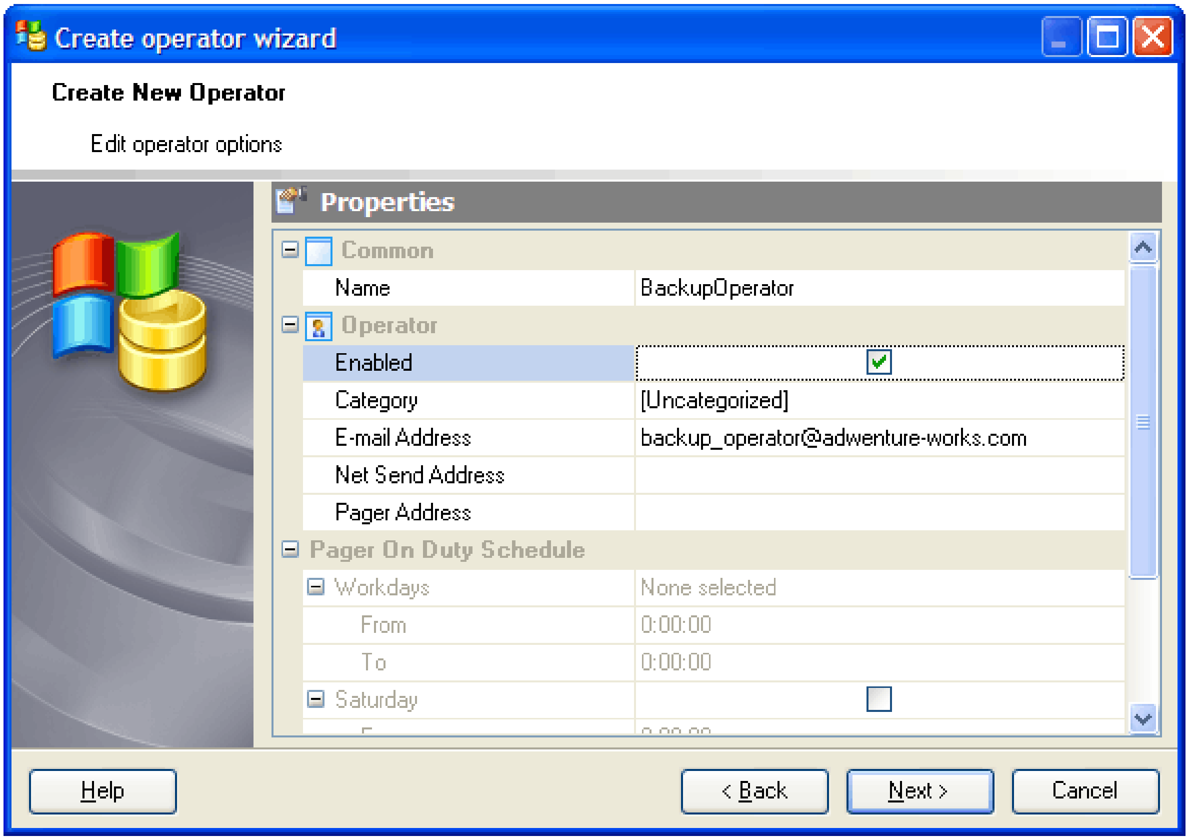
Create New Operator

Edit operator options

Common	
Name	BackupOperator
Operator	
Enabled	☑
Category	[Uncategorized]
E-mail Address	backup_operator@adventure-works.com
Net Send Address	
Pager Address	
Pager On Duty Schedule	
Workdays	None selected
From	0:00:00
To	0:00:00
Saturday	☐

6.9.1.1 Specifying operator options

Specify operator options according to your needs. The detailed description is given below.



Create New Operator

Edit operator options

Common	
Name	BackupOperator
Operator	
Enabled	☑
Category	[Uncategorized]
E-mail Address	backup_operator@adventure-works.com
Net Send Address	
Pager Address	
Pager On Duty Schedule	
Workdays	None selected
From	0:00:00
To	0:00:00
Saturday	☐

☑ Enabled

Indicates the current status of the operator. If not checked, the operator is not enabled and does not receive notifications.

Category

Defines the name of the category for this operator. See Categories 289 for details.

E-mail Address

Defines the e-mail address of the operator.

Net Send Address

Defines the network address of the operator the network message is sent to.

Pager Address

Defines the pager address of the operator.

Pager On Duty Schedule

Here you can select the days of week when that the operator is available for pages.

From

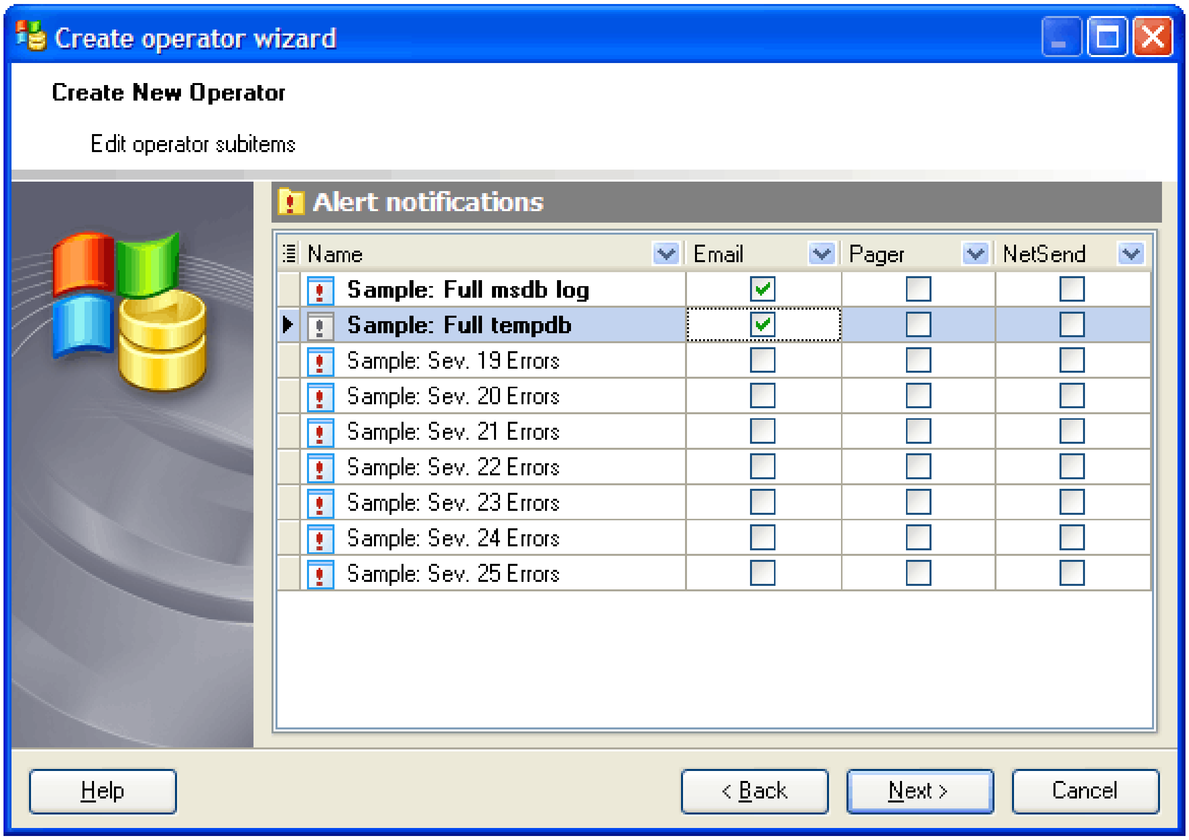
Specify the time after which *SQLServerAgent* service sends pager notification to the operator on the specified days.

To

Specify the time after which *SQLServerAgent* service no longer sends pager notification to the operator on the specified days.

6.9.1.2 Managing operator notifications

The wizard step is provided for your convenient setting of alert notifications for the operator being created. The grid represents all the available server alerts. Just check necessary boxes to assign alert and type of notification.



Name	Email	Pager	NetSend
Sample: Full msdb log	☑	☐	☐
Sample: Full tempdb	☑	☐	☐
Sample: Sev. 19 Errors	☐	☐	☐
Sample: Sev. 20 Errors	☐	☐	☐
Sample: Sev. 21 Errors	☐	☐	☐
Sample: Sev. 22 Errors	☐	☐	☐
Sample: Sev. 23 Errors	☐	☐	☐
Sample: Sev. 24 Errors	☐	☐	☐
Sample: Sev. 25 Errors	☐	☐	☐

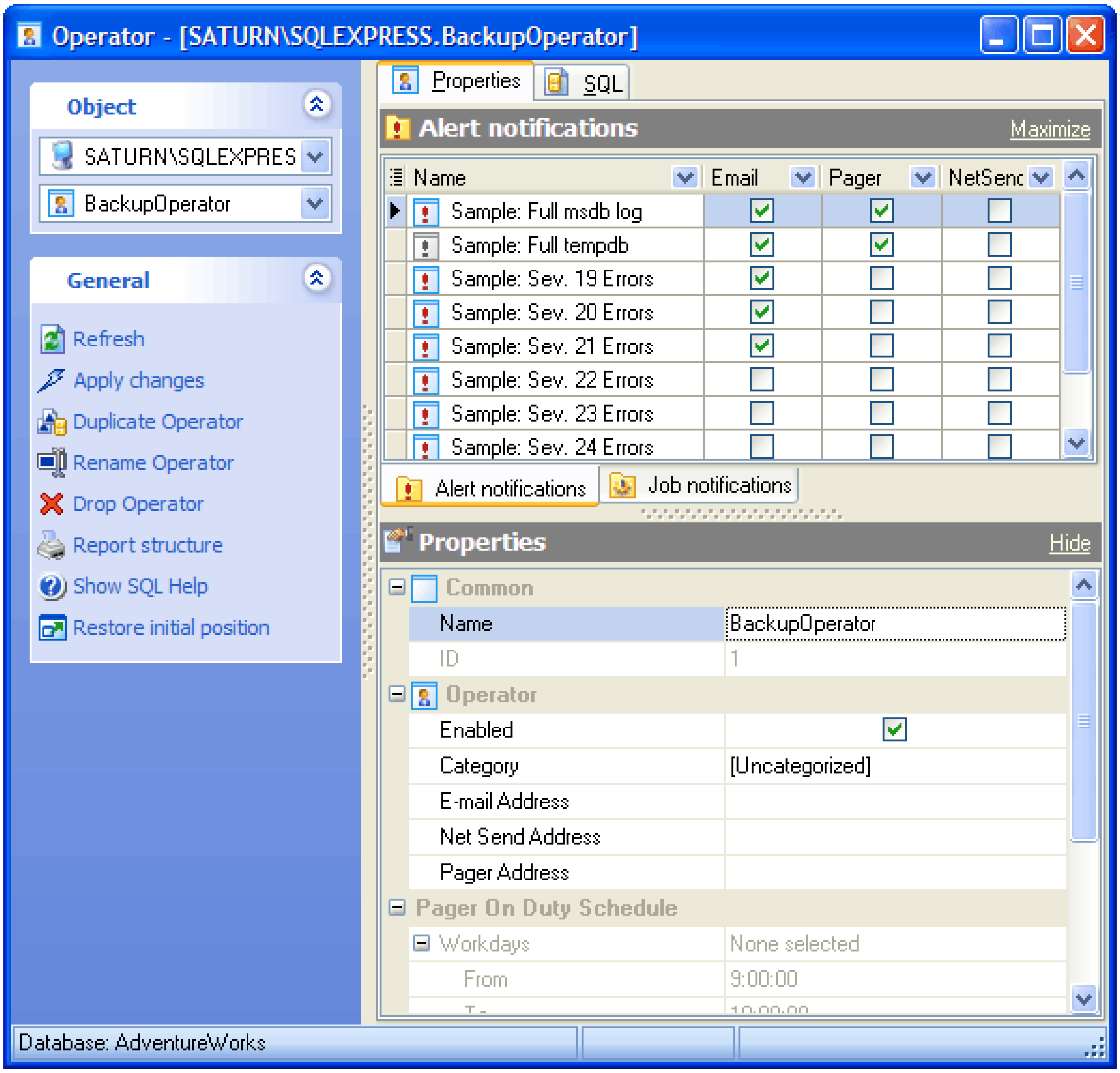
6.9.2 Operator Editor

Operator Editor allows you to edit operator properties, view the Transact-SQL statement for creating the operator, etc.

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.

- Editing operator properties 287

See also: Create Operator Wizard 284



Name	Email	Pager	NetSend
Sample: Full msdb log	☑	☑	☐
Sample: Full tempdb	☑	☑	☐
Sample: Sev. 19 Errors	☑	☐	☐
Sample: Sev. 20 Errors	☑	☐	☐
Sample: Sev. 21 Errors	☑	☐	☐
Sample: Sev. 22 Errors	☐	☐	☐
Sample: Sev. 23 Errors	☐	☐	☐
Sample: Sev. 24 Errors	☐	☐	☐

Common	
Name	BackupOperator
ID	1
Operator	
Enabled	☑
Category	[Uncategorized]
E-mail Address	
Net Send Address	
Pager Address	
Pager On Duty Schedule	
Workdays	None selected
From	9:00:00

6.9.2.1 Editing operator properties

Operator Editor provides you with an ability to edit operator properties.

Name
Edit the operator name in the field.

☑ Enabled
Indicates the current status of the operator. If not checked, the operator is not enabled and does not receive notifications.

Category
You can change the name of the category for this operator. See Categories 289 for details.

Use the E-mail Address, Net Send Address, and Pager Address to re-define the e-mail address, the network address, and the pager address of the operator.

Pager On Duty Schedule

Here you can view the days of week and time when the operator is available for pages.

From

Indicates the time after which *SQLServerAgent* service sends pager notification to the operator on the specified days.

To

Indicates the time after which *SQLServerAgent* service no longer sends pager notification to the operator on the specified days.

Notifications

Alerts

The Alerts tab displays all server alerts as a grid. Select the necessary alert, then press **Enter** or select the Edit... item from the popup menu or open Alert Editor 297. Use grid's popup menu to edit, copy the selected alerts. You can operate on several alerts at a time. For this you have to select alerts with the **Shift** or the **Ctrl** key pressed. After the objects group is selected, you can operate on it as it was a single object. You can specify E-mail, Pager or/and Net notification methods by which the operator is notified when the alert occurs.

Jobs

The Jobs tab displays all server jobs that have this operator as recipient of e-mail, page or net send notification messages. Select the necessary job, then press **Enter** or select the **Edit...** item from the popup menu or open Job Editor 278. Use grid's popup menu to edit, copy the selected alerts. You can operate on several alerts at a time. For this you have to select alerts with the **Shift** or the **Ctrl** key pressed. After the objects group is selected, you can operate on it as it was a single object.

To apply the changes, select the Apply Changes item in the Navigation bar or use **Ctrl+F9** or **Ctrl+F7** shortcut keys.

It is also possible to modify object properties without opening the object editor: use the Object Properties item of the popup menu of the selected object from the explorer tree.

6.10 Categories

The Category object represents the attributes of a *SQL Server Agent* alert 294 category, job 271 category or operator 283 category.

Microsoft SQL Server provides categories to help you organize your jobs, alerts and operators for easy filtering and grouping. For example, you can organize all your database backup jobs in the Database Maintenance category. You can also create your own categories. *Multiserver* categories exist only on a *master* server. There is only one default job category available only on a *master* server: *[Uncategorized (Multi-Server)].*

> **Note:** Alerts are recategorized as necessary when an alert category is removed. Any alerts previously exhibiting the removed category exhibit the category *[Uncategorized]* after the removing.

How can I create a new category?

New categories are created within Create Category Wizard 290. In order to run this wizard you should either

- select the Categories list or any object from that list and then use the Create New Category item from the popup menu

or

- open the server in Server Editor and the Categories tab there and press **Insert** or select the Create New Category item from the popup menu (Alternatively, use the corresponding link of the Navigation Bar).

How can I edit an existing category?

Categories are edited within Category Editor 291. In order to open the editor you should either

- select the category for editing in the explorer tree (type the first letters of the category name for quick search);
- select the Edit Category item from the popup menu

or

- open the server in Server Editor and the Categories tab there;
- select the category to edit;
- press the **Enter** key or select the Edit Category item from the popup menu (alternatively, you can use the corresponding link of the Navigation Bar).

How can I drop a category?

To drop the existing category:

- select the category to drop in the explorer tree;
- select the Drop Category item from the popup menu

or

- open the server in Server Editor and the Categories tab there;

- select the category to drop;
- press **Delete** or select the Drop Category item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.

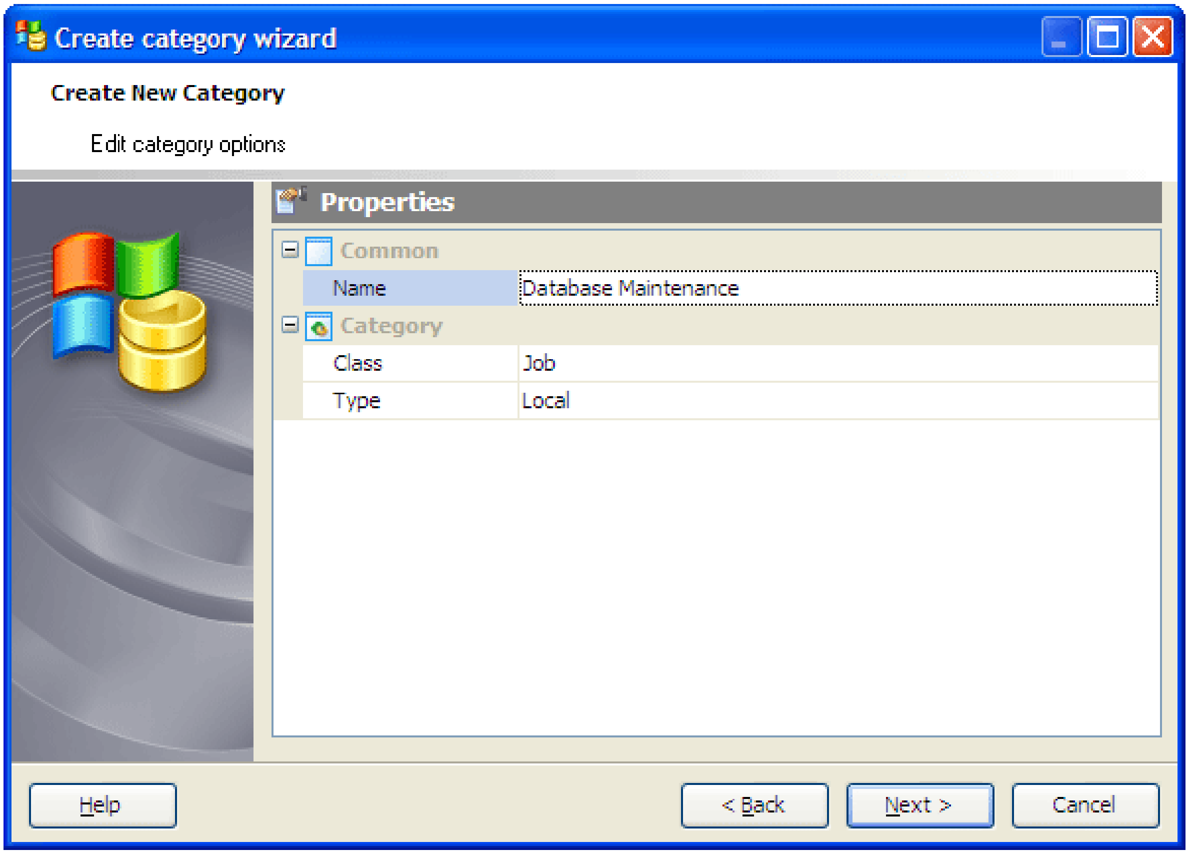
6.10.1 Create Category Wizard

Create Category Wizard guides you through the process of creating a new category. See How To Create Category 289 for instructions on running this wizard.

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

- Specifying category options 290

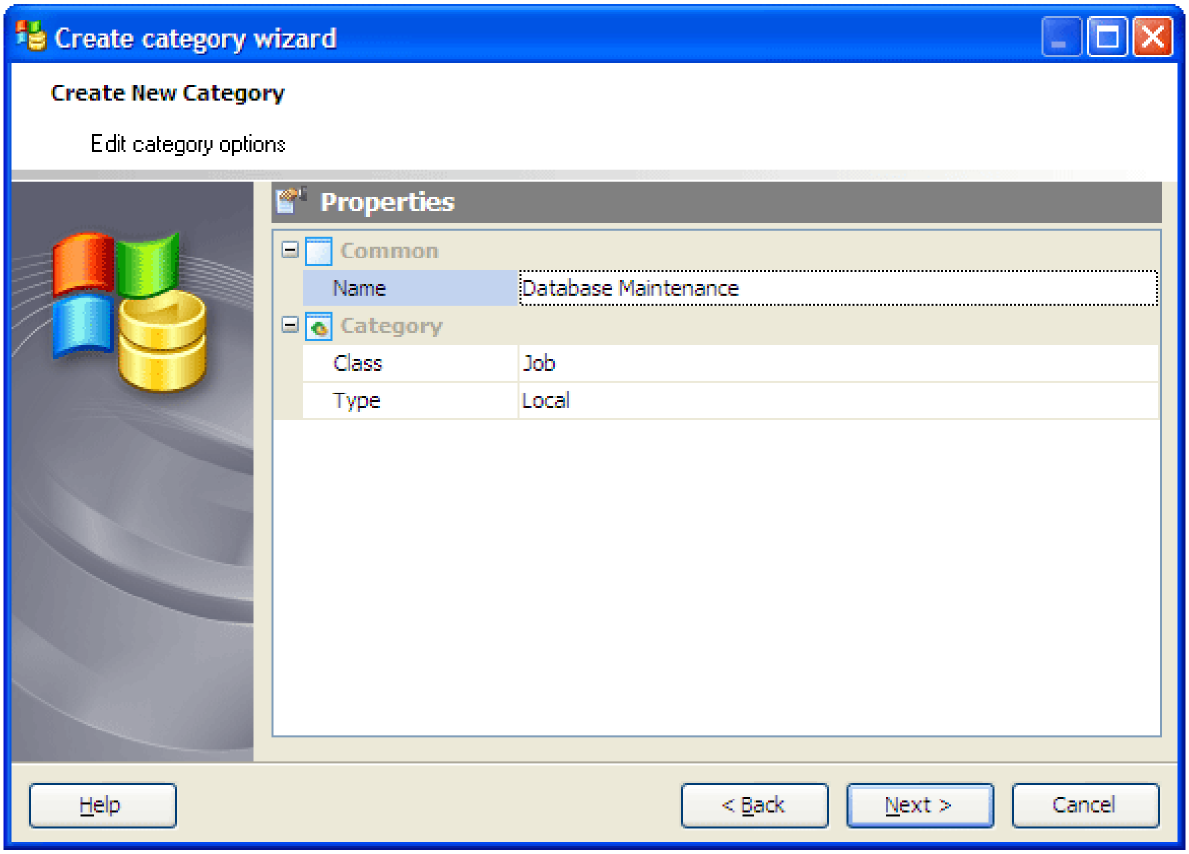
See also: Category Editor 291



Name: Database Maintenance
Class: Job
Type: Local

6.10.1.1 Specifying category options

Specify category options according to your needs. The detailed description is given below.



Name: Database Maintenance

Class: Job

Type: Local

Name
Set the name for the new category. Bear in mind that the name must be unique within the specified class.

Class
Select the class of the category out of the following values: *Job* (adds a job category, set by default), *Alert* (adds an alert category), *Operator* (adds an operator category).

Type
Select the type of the category out of the following values: *Local* (a local job category, set by default), *Multi-Server* (a multiserver job category).

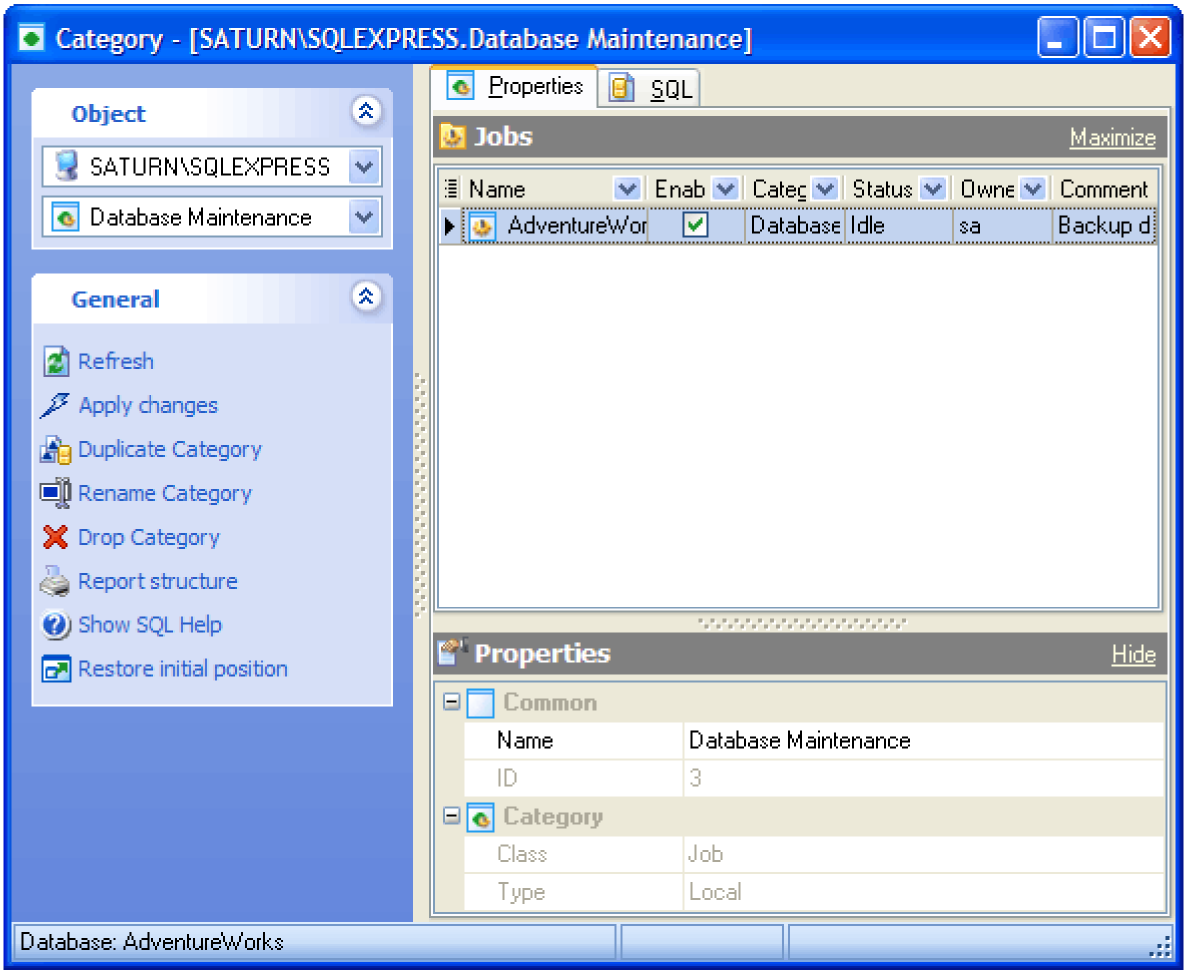
6.10.2 Category Editor

Category Editor allows you to edit category properties, view the Transact-SQL statement for creating the category, etc. It can be opened when you create a new category or edit the existing one (see How to edit category 289 for details).

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.

- Editing category properties 292

See also: Create Category Wizard 290

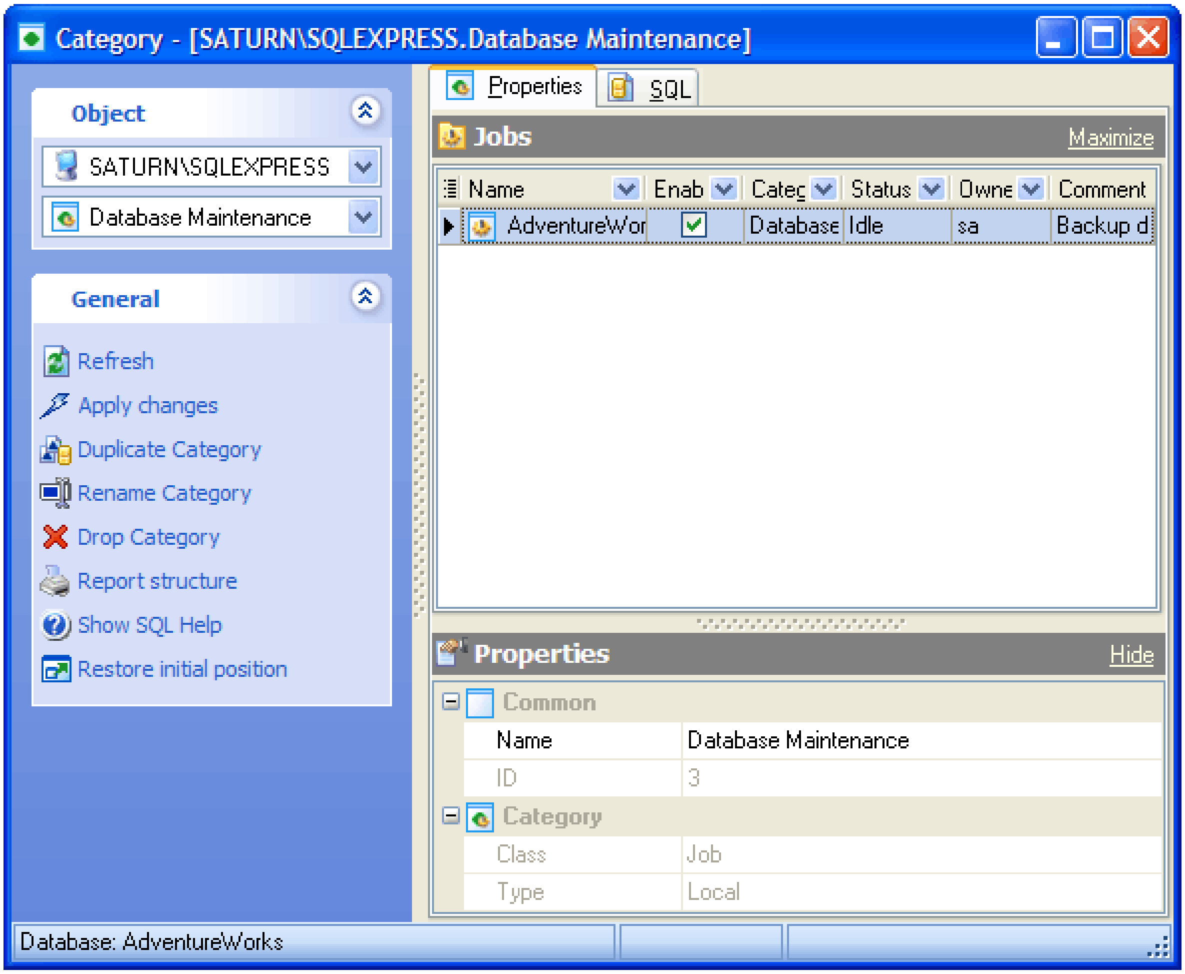


Name	Enab	Categ	Status	Owne	Comment
AdventureWor	☑	Database	Idle	sa	Backup d

Name	Database Maintenance
ID	3
Class	Job
Type	Local

6.10.2.1 Editing category properties

Specify the category options according to your needs. The detailed description is given below.



Name	Enab	Categ	Status	Owne	Comment
AdventureWor	☑	Database	Idle	sa	Backup d

Common

Name	Database Maintenance
ID	3

Category

Class	Job
Type	Local

Database: AdventureWorks

Name
Set the name of the category. Bear in mind that the name must be unique within the specified class.

Class
Select the class of the category out of the following values: *Job* (adds a job category, set by default), *Alert* (adds an alert category), *Operator* (adds an operator category).

Type
Select the type of the category out of the following values: *Local* (a local job category, set by default), *Multi-Server* (a multiserver job category), *None* (a category for a class other than JOB).

Alerts 294, jobs 271 or operators 283 that are connected to the current category are managed in the Operators/Jobs/Alerts in this category list.

To apply the changes, select the Apply Changes item in the Navigation bar or use **Ctrl+F9** or **Ctrl+F7** shortcut keys.

It is also possible to modify object properties without opening the object editor: use the Object Properties item of the popup menu of the selected object from the explorer tree.

6.11 Alerts

The Alert object represents a single *SQL Server Agent* alert. Alerts respond to either specific Microsoft SQL Server error messages or SQL Server errors of a specified severity. By default, only members of the *sysadmin* fixed server role can create alerts.

Note: Before starting a job make sure that the *SQL Server Agent* service currently running on the server.

How can I create a new alert?

New alerts are created within Create Alert Wizard 295. In order to run this wizard you should either

- select the Alerts list or any object from that list and then use the Create New Alert item from the popup menu

or

- open the server in Server Editor and the Alerts tab there and press **Insert** or select the Create New Alert item from the popup menu (Alternatively, use the corresponding link of the Navigation Bar).

How can I edit an existing alert?

Alerts are edited within Alert Editor 297. In order to open the editor you should either

- select the alert for editing in the explorer tree (type the first letters of the alert name for quick search);
- select the Edit Alert item from the popup menu

or

- open the server in Server Editor and the Alerts tab there;
- select the alert to edit;
- press the **Enter** key or select the Edit Alert item from the popup menu (alternatively, you can use the corresponding link of the Navigation Bar).

How can I drop an alert?

To drop the existing alert:

- select the alert to drop in the explorer tree;
- select the Drop Alert item from the popup menu

or

- open the server in Server Editor and the Alerts tab there;
- select the alert to drop;
- press the **Delete** key or select the Drop Alert item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.

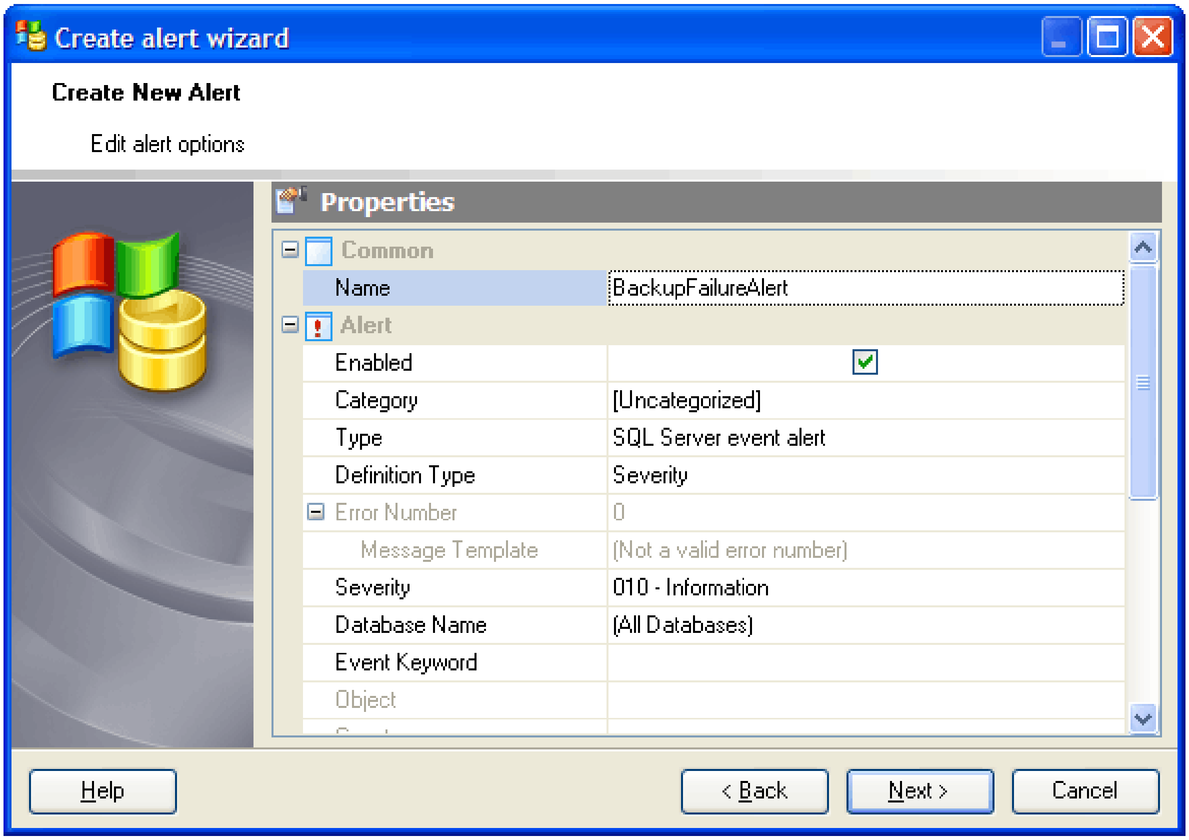
6.11.1 Create Alert Wizard

Create Alert Wizard guides you through the process of creating a new alert. See How To Create Alert 294 for instructions on running this wizard.

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

- Specifying alert options 295
- Setting operator the alert to give notify of 297

See also: Alert Editor 297



Create New Alert

Edit alert options

Properties

Common	
Name	BackupFailureAlert
Alert	
Enabled	☑
Category	[Uncategorized]
Type	SQL Server event alert
Definition Type	Severity
Error Number	0
Message Template	(Not a valid error number)
Severity	010 - Information
Database Name	(All Databases)
Event Keyword	
Object	

6.11.1.1 Specifying alert options

Specify alert options according to your needs.

Category
Define here the name of the alert category. See Categories 289 for details.

Type
Describe the type of the alert in this field. Possible variants are: *SQL Server event* alert and *SQL server performance condition* alert.

Definition Type

This field is available when the *SQL Server event alert* type is selected. Indicates the definition type of *SQL Server event* alert.

Error Number

This field is available when the *Error Number* definition type is selected. Shows the message error number that defines the alert.

Message Template

The field displays the explanation of the error with placeholders for parameters.

Severity

This field is available when the *Severity* definition type is selected. Defines the severity level (from 1 through 25) that defines the alert.

Database Name

The database in which the error must occur for the alert to fire. If no database is supplied, the alert fires regardless of where the error occurred.

Event Keyword

The sequence of characters that the description of the SQL Server error must be like. Transact-SQL *LIKE* expression pattern-matching characters can be used. the maximum size of this field is 100. This parameter is useful for filtering object names (for example, %customer_table%).

Object

This field is available when the *SQL Server performance condition alert* type is selected. A performance object used in performance condition.

Counter

This field is available when the *SQL Server performance condition alert* type is selected. A performance counter used in performance condition.

Instance

This field is available when the *SQL Server performance condition alert* type is selected. A named instance of the counter used in performance condition.

Condition

This field is available when the *SQL Server performance condition alert* type is selected. Defines comparator in performance condition.

Value

This field is available when the *SQL Server performance condition alert* type is selected. Numeric value of the counter.

Enabled

Indicates the current status of the alert. If unchecked, the alert is not enabled and does not fire.

Job

The name of the job to be executed in response to this alert.

Include Alert Error In

The description of the SQL Server error should be included as part of the notification message. Can have one or more of values (None, E-mail, Pager, Net send) combined with an OR logical operator.

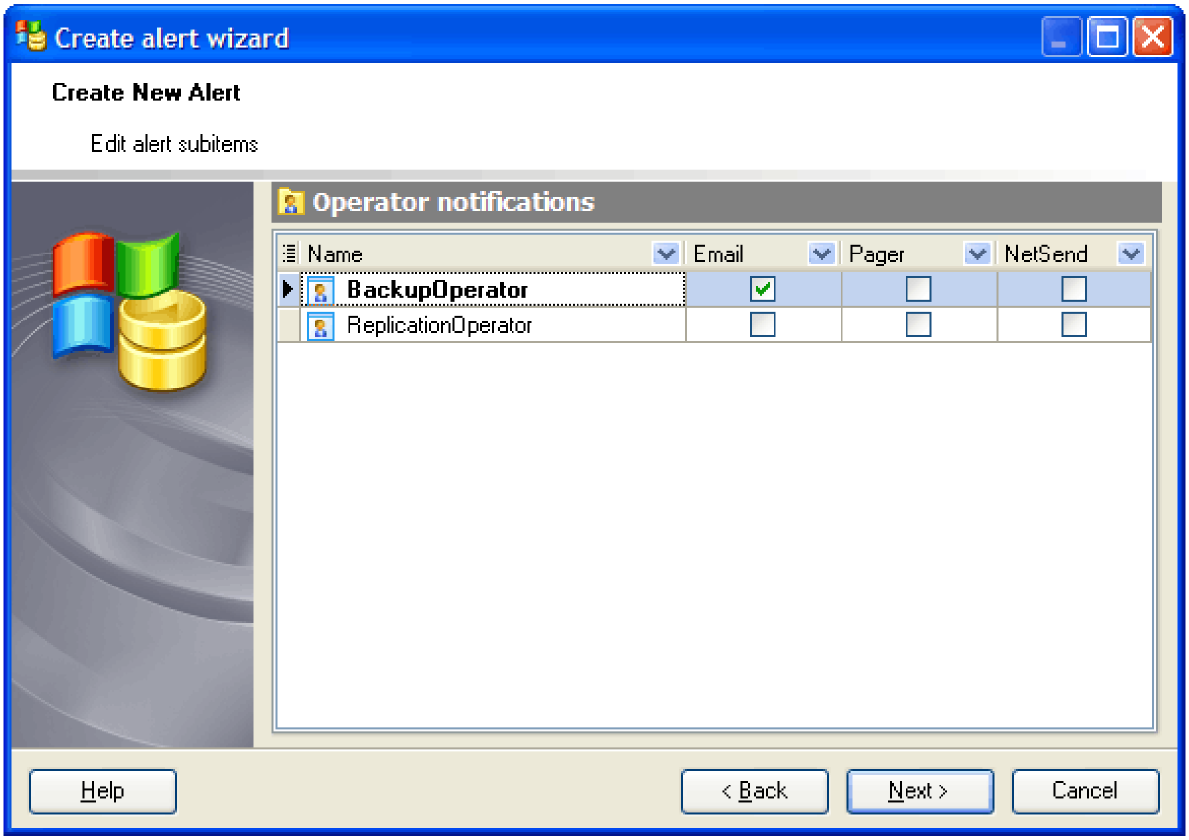
Notification Message
Is an optional additional message sent to the operator as part of the e-mail, net send, or pager notification.

Minutes
The delay period, in minutes, between responses to the alert.

Seconds
The delay period, in seconds, between responses to the alert.

6.11.1.2 Setting operator the alert to give notify of

The wizard step is provided for your convenient setting of operator notifications for the alert being created. The grid represents all the available server operators. Just check necessary boxes to assign operator and type of notification.



Name	Email	Pager	NetSend
BackupOperator	☑	☐	☐
ReplicationOperator	☐	☐	☐

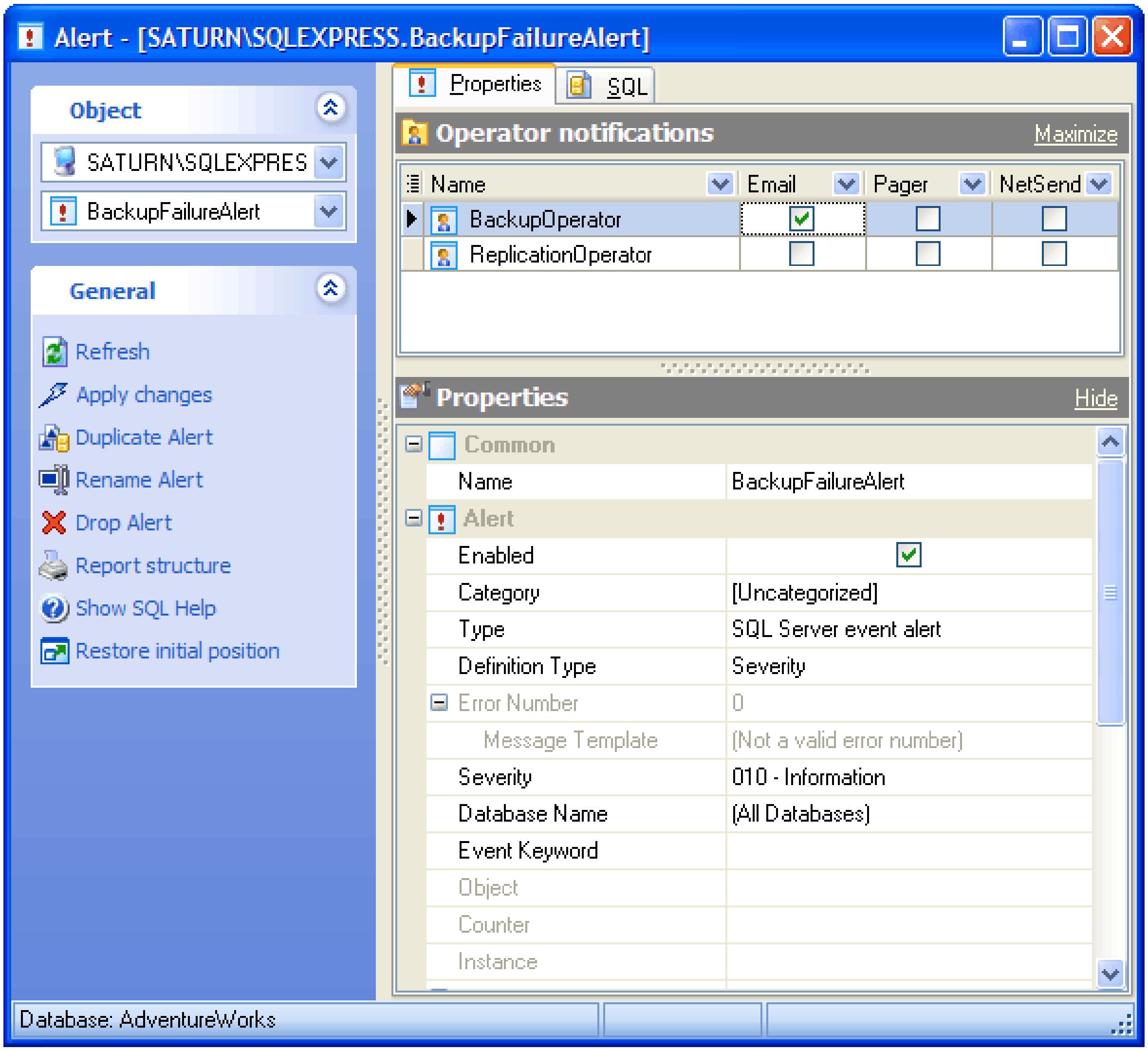
6.11.2 Alert Editor

Alert Editor allows you to edit alert properties, view the Transact-SQL statement for creating the alert, etc. It opens when you create a new alert or edit the existing one (see How to edit alert 294 for details).

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.

- Editing alert properties 298

See also: Create Alert Wizard 295



Operator notifications

Name	Email	Pager	NetSend
BackupOperator	☑	☐	☐
ReplicationOperator	☐	☐	☐

Properties

Common	
Name	BackupFailureAlert
Alert	
Enabled	☑
Category	[Uncategorized]
Type	SQL Server event alert
Definition Type	Severity
Error Number	0
Message Template	(Not a valid error number)
Severity	010 - Information
Database Name	(All Databases)
Event Keyword	
Object	
Counter	
Instance	

6.11.2.1 Editing alert properties

Use the following information to clear up all the alert properties.

Operator notifications
The grid represents all the available server operators. Check necessary box to assign operator the alert to give notice of and the type of notification.

Properties
Name
Here you can edit the name of the alert.

Category
Defines the name of the alert category. See Categories 289 for details.

Type

Describes the type of the alert. Possible variants are: *SQL Server event alert* and *SQL server performance condition alert*.

Definition Type

This field is available when the *SQL Server event alert* type is selected. Indicates the definition type of SQL Server event alert.

Error Number

This field is available when the *Error Number* definition type is selected. Shows the message error number that defines the alert.

Message Template

Shows explanation of the error with placeholders for parameters.

Severity

This field is available when the *Severity* definition type is selected. Defines the severity level (from 1 through 25) that defines the alert.

Database Name

The database in which the error must occur for the alert to fire. If database is not supplied, the alert fires regardless of where the error occurred.

Event Keyword

The sequence of characters that the description of the SQL Server error must be like. Transact-SQL *LIKE* expression pattern-matching characters can be used. the maximum size of this field is 100. This parameter is useful for filtering object names (for example, %customer_table%).

Object

This field is available when the *SQL Server performance condition alert* type is selected. A performance object used in performance condition.

Counter

This field is available when the *SQL Server performance condition alert* type is selected. A performance counter used in performance condition.

Instance

This field is available when the *SQL Server performance condition alert* type is selected. A named instance of the counter used in performance condition.

Condition

This field is available when the *SQL Server performance condition alert* type is selected. Defines comparator in performance condition.

Value

This field is available when the *SQL Server performance condition alert* type is selected. Numeric value of the counter.

Enabled

Indicates the current status of the alert. If unchecked, the alert is not enabled and does not fire.

Job
The name of the job to be executed in response to this alert.

Include Alert Error In
The description of the SQL Server error should be included as part of the notification message. Can have one or more of values (None, E-mail, Pager, Net send) combined with an OR logical operator.

Notification Message
Is an optional additional message sent to the operator as part of the e-mail, net send, or pager notification.

Minutes
The delay period, in minutes, between responses to the alert.

Seconds
The delay period, in seconds, between responses to the alert.

Last Occurred
Shows the last date when the alert has been occurred.

Last Responded
Displays the date when the alert has last raised response.

Occurrence count
Displays the amount of alert occurrences.

Notifications
This tab displays all server operators as a grid. Select the necessary notification, then press **Enter** or select the Edit... item from the popup menu or open Operator Editor 286. Use grid's popup menu to edit, copy the selected operators.

You can manage several operators at a time. For this you have to select operators with the **Shift** or the **Ctrl** key pressed. After the objects group is selected, you can operate on it as if it were a single object. Using the checkboxes you can specify the method by which the operator is notified (E-mail, Pager or/and Net send) when the alert occurs.

To apply the changes, select the Apply Changes item in the Navigation bar or use **Ctrl+F9** or **Ctrl+F7** shortcut keys.

It is also possible to modify object properties without opening the object editor: use the Object Properties item of the popup menu of the selected object from the explorer tree.

6.12 Credentials

Credentials allow users that connect to Microsoft SQL Server using SQL Authentication to connect to Windows or other resources outside of SQL Server. A credential is a record containing the authentication information needed to connect to a resource outside of SQL Server. Most credentials consist of a Windows user login name and password. In Windows 2003 Server and above, the password may not be required. This server object is available since Microsoft SQL 2005.

After creating a credential, you can map it to a *SQL Server login* with Create Login Wizard 246 or Login Editor 253.

Note: A single credential can be mapped to multiple SQL Server logins. But a SQL Server login can be mapped to only one credential.

How can I create a new credential?

New credentials are created within Create Credential Wizard 302. In order to run this wizard you should either

- select the Credentials list or any object from that list and then use the Create New Credential item from the popup menu

or

- open the server in Server Editor and the Credentials tab there and press **Insert** or select the Create New Credential item from the popup menu (Alternatively, use the corresponding link of the Navigation Bar).

How can I edit an existing credential?

Credentials are edited within Credential Editor 303. In order to open the editor you should either

- select the credential for editing in the explorer tree (type the first letters of the credential name for quick search);
- select the Edit Credential item from the popup menu

or

- open the server in Server Editor and the Credentials tab there;
- select the credential to edit;
- press the **Enter** key or select the Edit Credential item from the popup menu (alternatively, you can use the corresponding link of the Navigation Bar).

How can I drop a credential?

To drop the existing credential:

- select the credential to drop in the explorer tree;
- select the Drop Credential item from the popup menu

or

- open the server in Server Editor and the Credentials tab there;

- select the credential to drop;
- press **Delete** or select the Drop Credential item from the popup menu (alternatively, you may use the corresponding link of the Navigation Bar)

and confirm dropping in the dialog window.

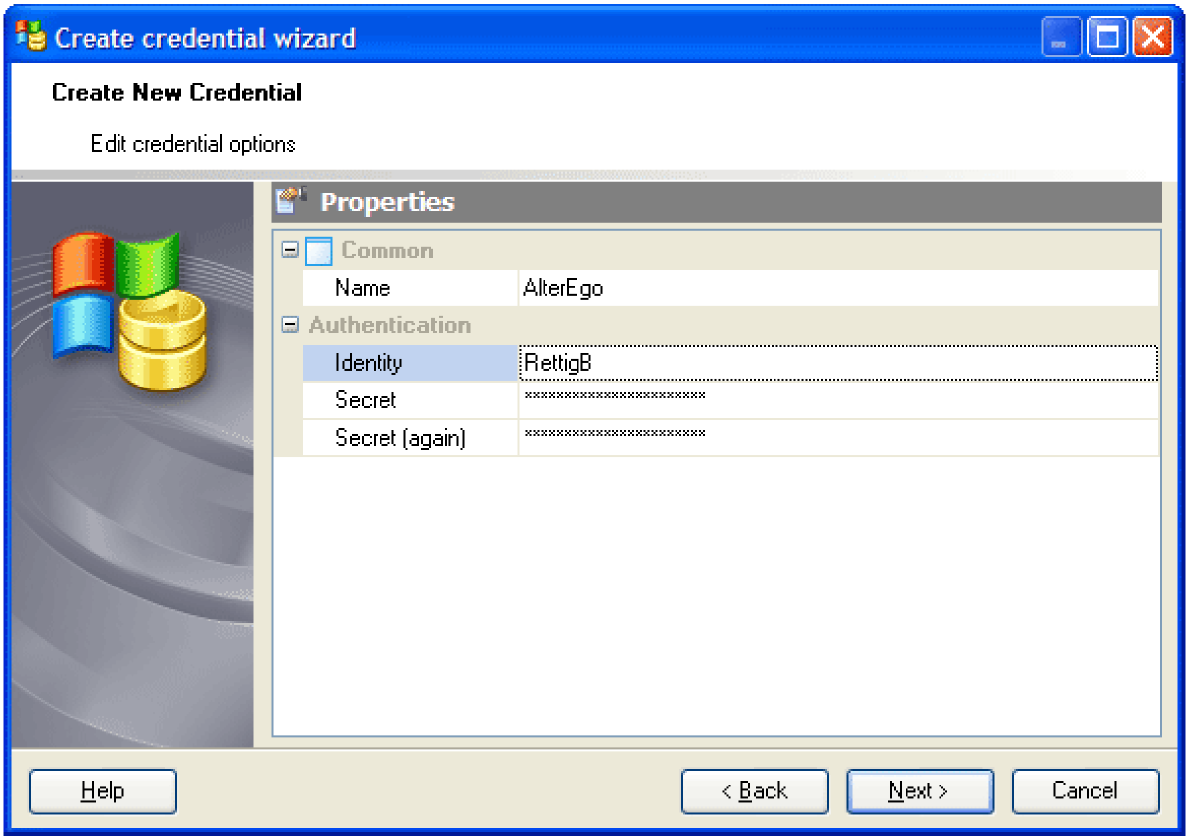
6.12.1 Create Credential Wizard

Create Credential Wizard guides you through the process of creating a new server credential. See How To Create credential 301 to learn how to run this wizard.

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

- Specifying credential options 302

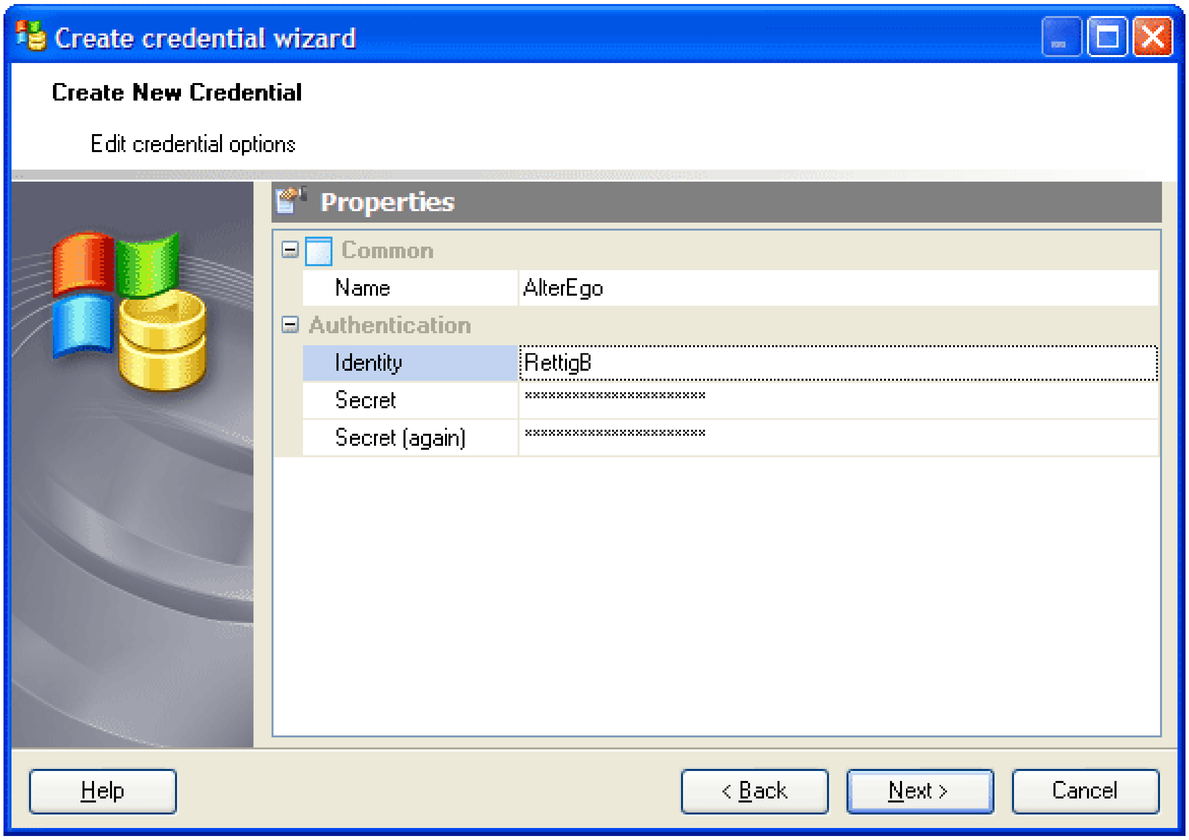
See also: Credential Editor 303



Name: AlterEgo
Identity: RettigB
Secret: ************************
Secret (again): ************************

6.12.1.1 Specifying credential options

Specify credential options according to your needs. The detailed description is given below.



Name: AlterEgo
Identity: RettigB
Secret: **********************
Secret (again): **********************

Identity
Specifies the name of the account to be used when connecting outside the server.

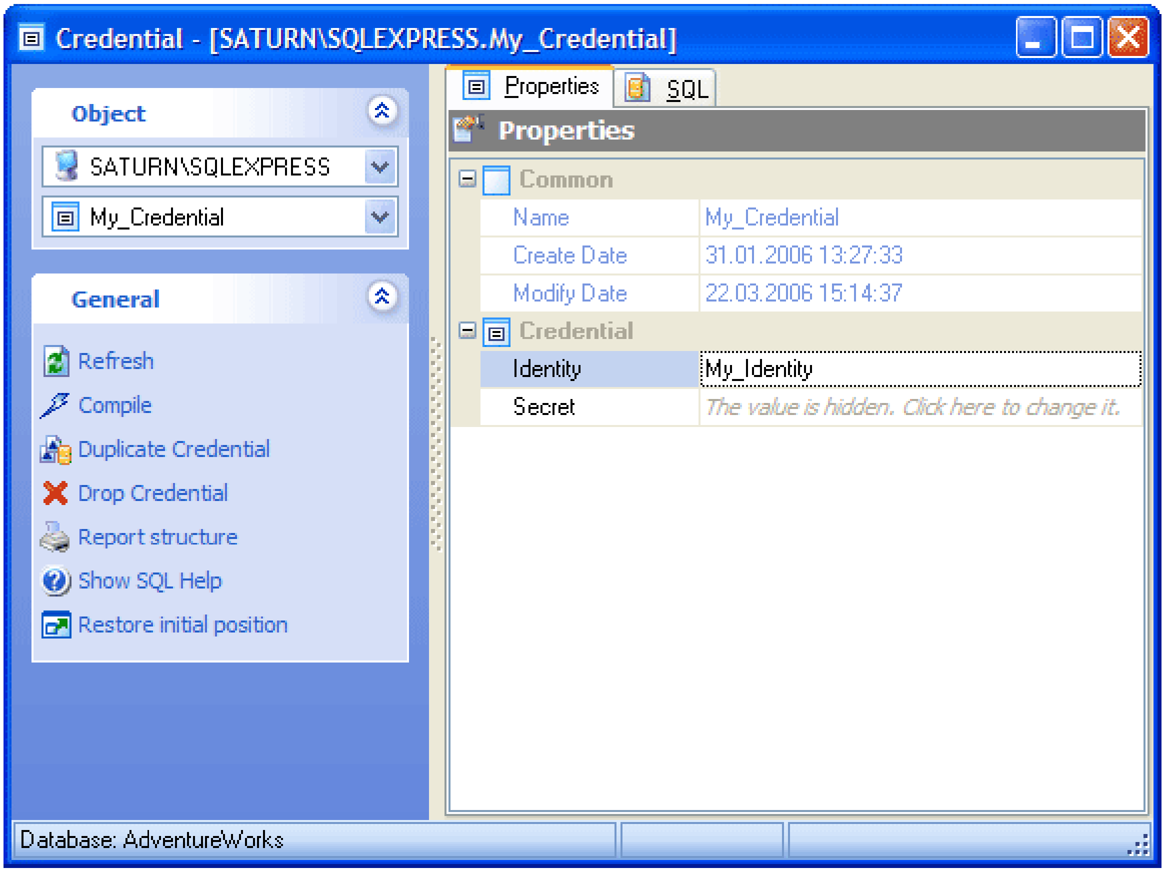
Secret
Supplies the secret needed for outgoing authentication. The secret field is optional. When **Identity** field is a Windows account, the secret may be the password. In Windows Server 2003 and above the field is optional.

6.12.2 Credential Editor

Credential Editor allows you to edit login properties, view the SQL statement for creating the credential.

The basic principles of Object Editors in MS SQL Maestro are explained in a separate topic 48. Below you will find a description of editor tabs that are unique for the current object.

- Editing credential properties 304

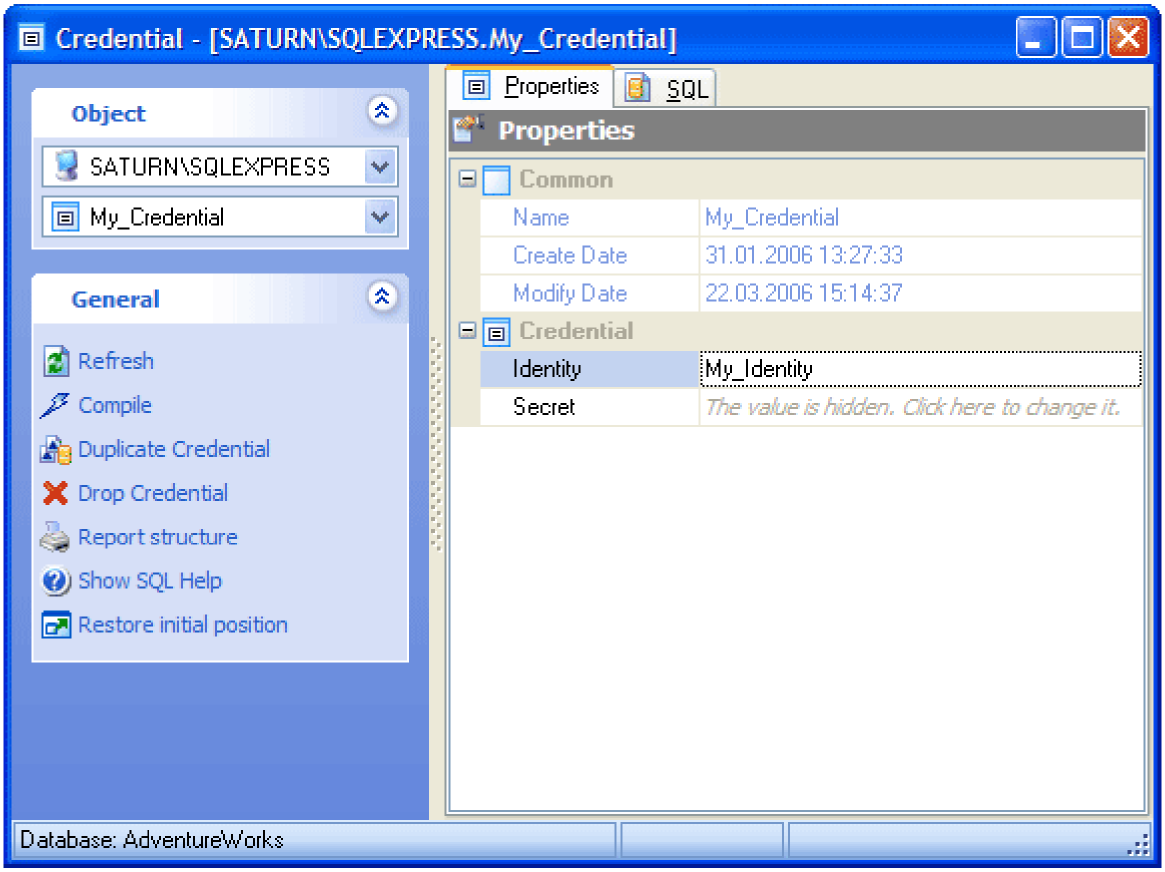


Name	My_Credential
Create Date	31.01.2006 13:27:33
Modify Date	22.03.2006 15:14:37
Identity	My_Identity
Secret	The value is hidden. Click here to change it.

Database: AdventureWorks

6.12.2.1 Editing credential properties

Credential Editor provides you with an ability to edit credential properties.



Name: My_Credential
Create Date: 31.01.2006 13:27:33
Modify Date: 22.03.2006 15:14:37
Identity: My_Identity
Secret: The value is hidden. Click here to change it.

Database: AdventureWorks

Name
Displays credential name.

Create Date
Displays the date when the credential was created.

Modify Date
Displays the date when the credential was last modified.

Identity
Specifies the name of the account to be used when connecting outside the server.

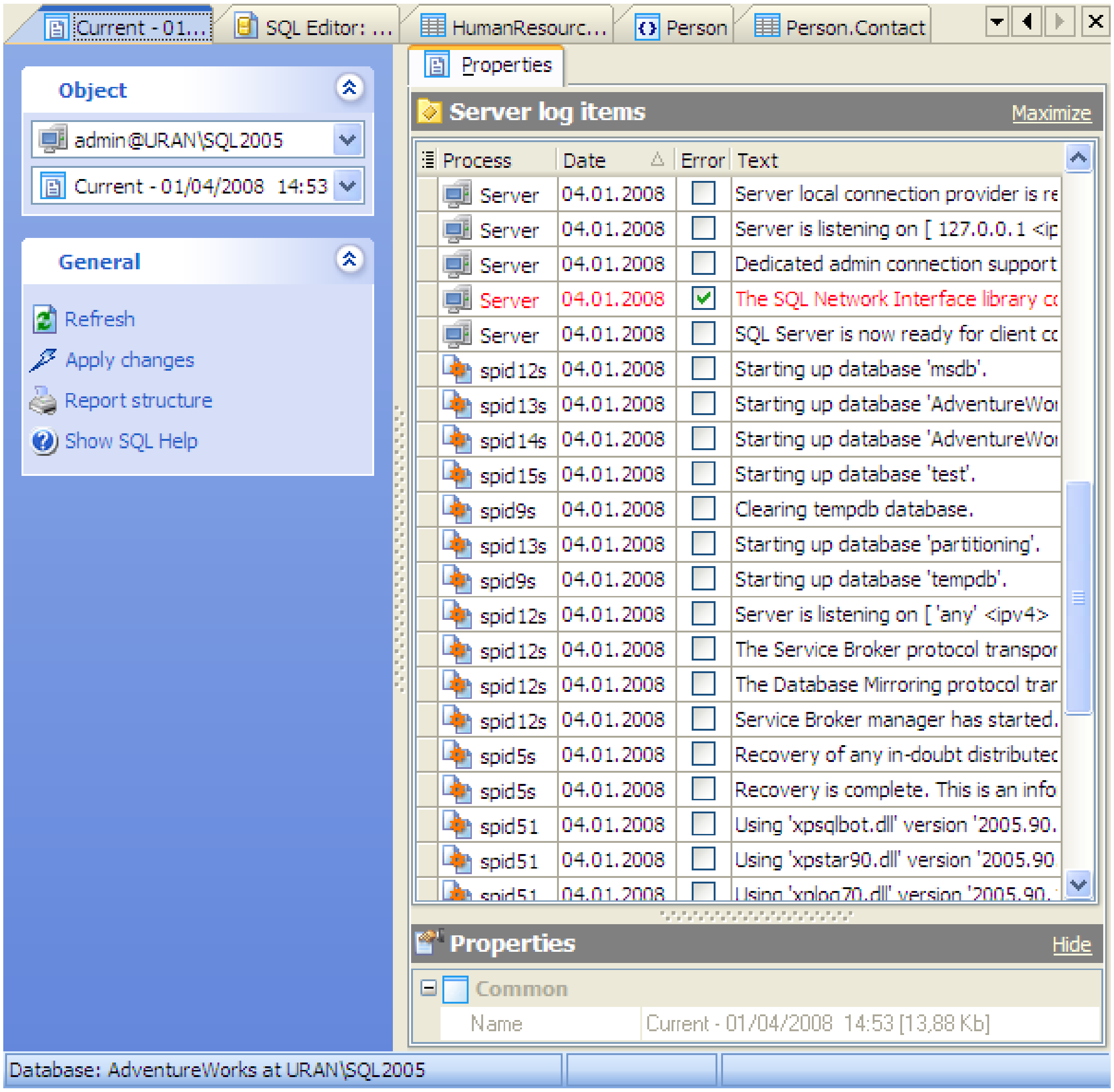
Secret
Supplies the secret needed for outgoing authentication. The secret field is optional. When **Identity** field is a Windows account, the secret may be the password. On Windows Server 2003 and above the secret is not necessary.

To apply the changes, select the Apply Changes item in the Navigation bar or use **Ctrl+F9** or **Ctrl+F7** shortcut keys.

It is also possible to modify object properties without opening the object editor: use the Object Properties item of the popup menu of the selected object from the explorer tree.

6.13 Server logs

MS SQL Maestro allows you to view and analyze SQL server logs. All logs are automatically separated into several parts by event time.



admin@URAN\SQL2005

Current - 01/04/2008 14:53

Process	Date	Error	Text
Server	04.01.2008	☐	Server local connection provider is re
Server	04.01.2008	☐	Server is listening on [127.0.0.1 <ip
Server	04.01.2008	☐	Dedicated admin connection support
Server	04.01.2008	☑	The SQL Network Interface library co
Server	04.01.2008	☐	SQL Server is now ready for client co
spid12s	04.01.2008	☐	Starting up database 'msdb'.
spid13s	04.01.2008	☐	Starting up database 'AdventureWor
spid14s	04.01.2008	☐	Starting up database 'AdventureWor
spid15s	04.01.2008	☐	Starting up database 'test'.
spid9s	04.01.2008	☐	Clearing tempdb database.
spid13s	04.01.2008	☐	Starting up database 'partitioning'.
spid9s	04.01.2008	☐	Starting up database 'tempdb'.
spid12s	04.01.2008	☐	Server is listening on ['any' <ipv4>
spid12s	04.01.2008	☐	The Service Broker protocol transpor
spid12s	04.01.2008	☐	The Database Mirroring protocol tran
spid12s	04.01.2008	☐	Service Broker manager has started.
spid5s	04.01.2008	☐	Recovery of any in-doubt distributed
spid5s	04.01.2008	☐	Recovery is complete. This is an info
spid51	04.01.2008	☐	Using 'xpsqlbot.dll' version '2005.90.
spid51	04.01.2008	☐	Using 'xpstar90.dll' version '2005.90
spid51	04.01.2008	☐	Using 'xplog70.dll' version '2005.90.

Name: Current - 01/04/2008 14:53 [13,88 Kb]

Database: AdventureWorks at URAN\SQL2005

6.14 Linked servers

A linked server configuration allows Microsoft SQL Server to execute commands against OLE DB data sources on different servers. Linked servers offer these advantages:

- Remote server access.
- The ability to issue distributed queries, updates, commands, and transactions on heterogeneous data sources across the enterprise.
- The ability to address diverse data sources similarly.

How can I add a new linked server?

New linked servers are added within Create Linked Server Wizard 308.
In order to run the wizard you should either

- select the Linked Servers list or any object from that list in the explorer tree;
- select the Create New Linked Server... item from the popup menu

To create a new linked server with the same properties as one of the existing one has:

- select the Object | Duplicate Database Object... main menu item;
- follow the instructions of Duplicate Object Wizard.

How can I work with data on a linked server?

Along with using distributed queries at SQL Editor and SQL Script Editor, Linked Server Editor 309 allows you to browse data of the linked server. In order to run the editor you should either

- select the linked server for editing in the explorer tree (type the first letters of the server name for quick search);
- select the Edit Linked Server... item from the popup menu

How can I drop a linked server?

To drop a linked server:

- select the server to drop in the explorer tree;
- select the Drop Linked Server item from the popup menu

and confirm dropping in the dialog window.

6.14.1 Create Linked Server Wizard

Create Linked Wizard allows you to set up a linked server definition. All the information how to run the wizard you can find here 307.

The basic principles of Create Object Wizards in MS SQL Maestro are explained in a separate topic 45. Below you will find a description of wizard steps that are unique for the current object.

The wizard consists of two steps. Register the connection information and data source information with SQL Server on the first step.

Set the new remote server name. It is the name by which the linked server is known within this database.

Provider
Select the provider that corresponds to the data source from the drop-down list of available providers. Note, to work with a database, first of all you need to have the corresponding ODBC driver/OLE DB provider installed on the same PC as your SQL Server.

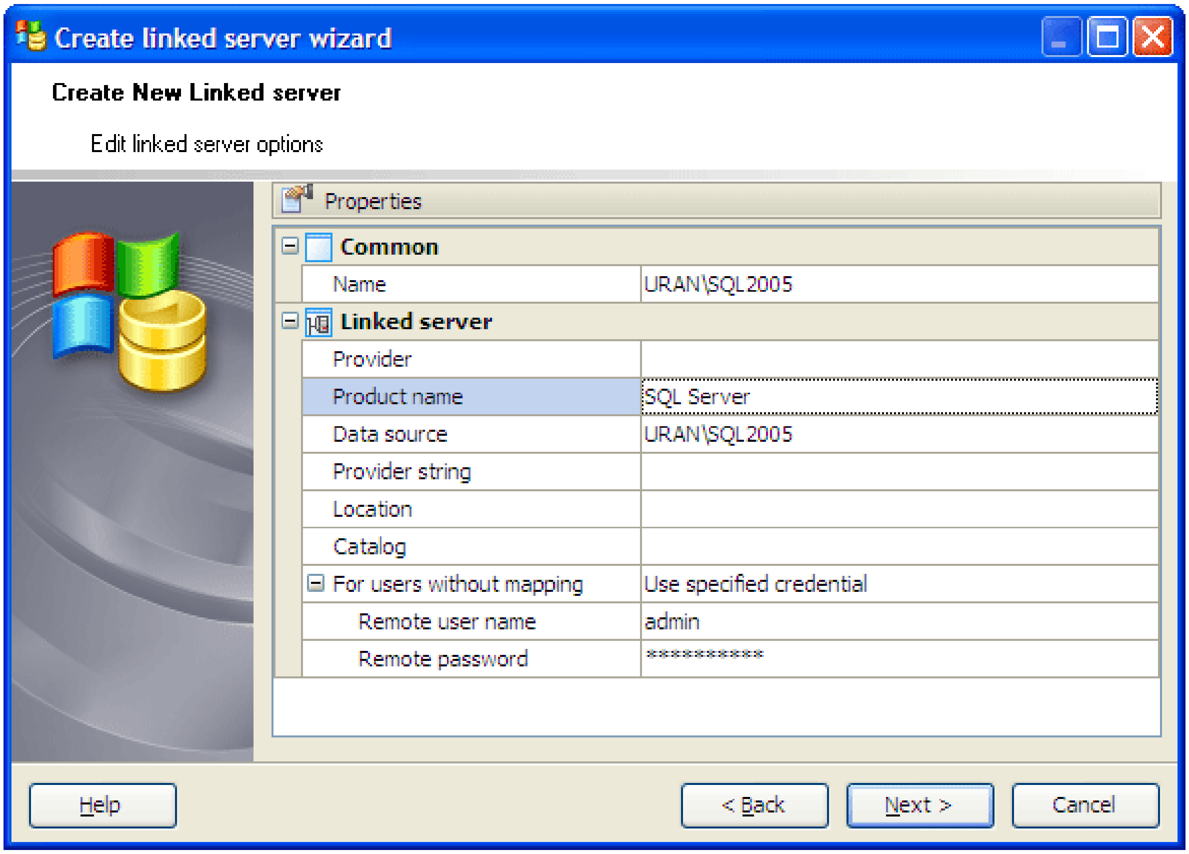
Product name
Set here the name of the OLE DB data source to add as a linked server. If SQL Server is specified, Provider, Data source, Location, Provider string, and Catalog do not have to be specified.

Data source is the name of the data source as it is interpreted by the OLE DB provider.

Provider string is the connection string that identifies a unique data source. To learn more about ODBC drivers, OLE DB Providers and other such stuff, read our brief guide to connection strings.

Location is the location of the database as interpreted by the OLE DB provider.

Catalog is the catalog to be used when a connection is made to the OLE DB provider.
When the linked server is defined against an instance of SQL Server, catalog refers to the default database to which the linked server is mapped.



Create New Linked server

Edit linked server options

Common	
Name	URAN\SQL2005
Linked server	
Provider	
Product name	SQL Server
Data source	URAN\SQL2005
Provider string	
Location	
Catalog	
For users without mapping	Use specified credential
Remote user name	admin
Remote password	***********

The second wizard step allows you to set up remote logins 309 to connect to the linked server.

6.14.2 Linked Server Editor

Linked Server Editor allows you to browse the linked server data and definition, and manage the logins to be provided by the sending server to connect to the receiving server on its behalf. To open the editor, use the popup menu of the corresponding node on the Explorer tree.

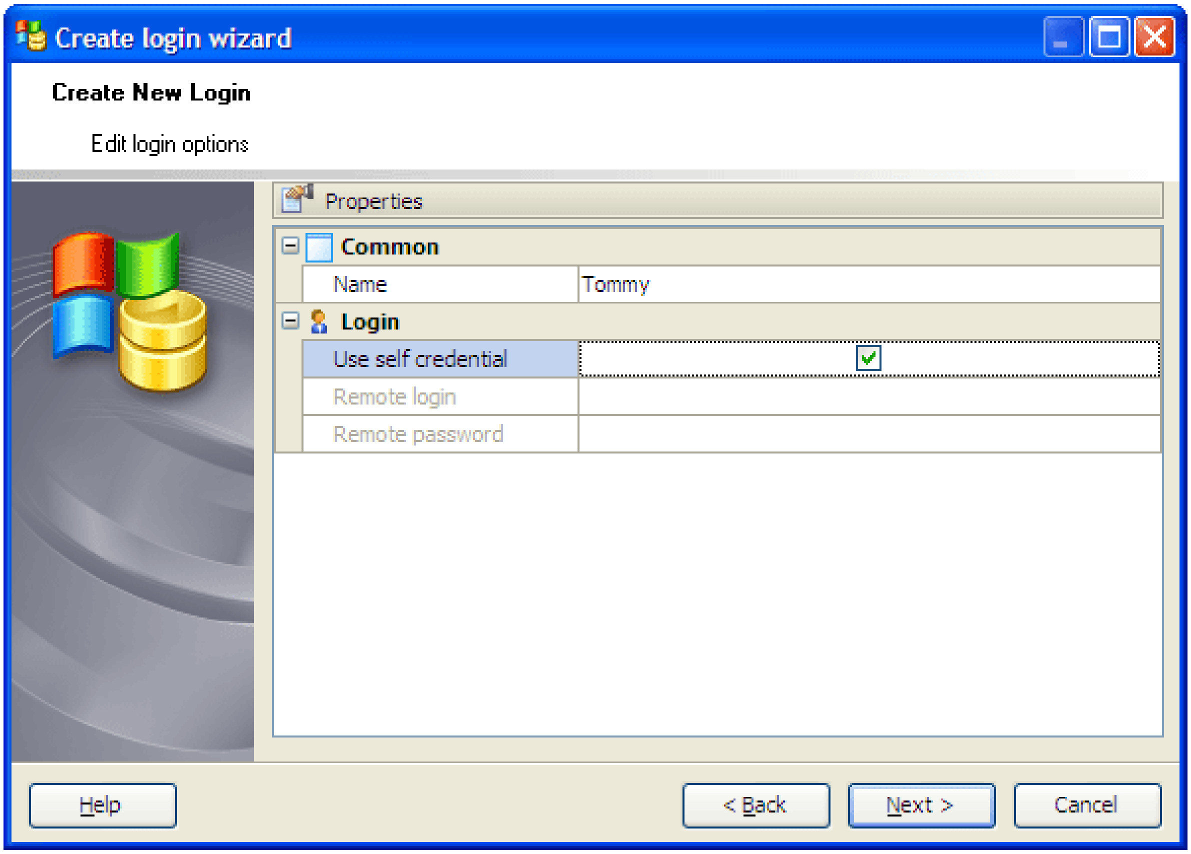
6.14.3 Remote Logins

During a linked server connection (for example, when processing a distributed query), the sending server provides a login name and password to connect to the receiving server on its behalf. To add/drop a remote login use the popup menu of the Logins area.

To emulate the current security credentials of the login, check the Use self credential checkbox.

Remote login specifies the username used to connect to the linked server.

Remote password is the password associated with the remote login.



Name: Tommy

Use self credential: ☑

Remote login:

Remote password:

7 Queries

MS SQL Maestro provides several tools for working with SQL queries:

- SQL Editor 313 for editing the query text directly and executing SELECT queries;
- Visual Query Builder 318 for building SELECT, INSERT, UPDATE and DELETE queries visually;
- SQL Script Editor 364 for executing SQL scripts.

Both SQL Editor and Visual Query Builder supports parameters in queries 317

Save frequently used queries to profiles and manage them in the same way as if they were database objects. This means that you can view queries in the explorer tree, in Object Manager and Object Browser, use them in BLOB Viewer and Diagram Viewer, perform drag-and-drop operation upon them, and copy them to clipboard like you copy an object.

How can I create a new SQL query?

New queries can be created either in SQL Editor or in Visual Query Builder.

To create a new query in SQL Editor:

- select the Tools | SQL Editor main menu item;
- select the Create New Query item from the navigation bar;
- edit the query text on the Editor tab of SQL Editor.

To create a new query in Query Builder:

- select the Tools | Visual Query Builder main menu item;
- build the query on the Diagram tab of Visual Query Builder.

MS SQL Maestro also provides you with SQL Generator, a tool to create simple SQL statements.

How can I save a query to a file/profile?

To save an existing query from the editor:

- to save the query to profile, use the Save to profile link from the Navigation bar.
- to save the current query to an **.sql* file, select the Save to file item from the Navigation bar;
- to save all the opened queries to one file, select the Save all queries item from the Navigation bar;
- to save the designed diagram, select the Save diagram item from the Navigation bar of the Diagram tab of Visual Query Builder.

⊟ How can I edit an existing SQL query?

Queries can be opened either in SQL Editor or in Visual Query Builder.

You can open the query directly from the Explorer tree with a double click or using popup menu. By default it will be opened in SQL Editor.

To edit a query from file, open SQL Editor (the Tools | SQL Editor main menu item) and use Load From File from the Navigation Bar of SQL Editor to load a query from an **.sql* file.

To edit a query in Query Builder, open the builder (the Tools | Visual Query Builder main menu item) and then perform one of the following operations:

- to edit a query from a profile, drag it from the Explorer and drop on the Editor tab;
- to load a previously saved diagram, use the Load Diagram item from the Navigation Bar;
- to load a query from an **.sql* file, open the Editor tab and select the Load query item from the Navigation Bar .

On the Query Builder opening the Diagram tab contains the last edited query.

⊟ How can I execute an SQL query?

To execute a query:

- create a new query or open the existing one;
- select the Execute Query item from the navigation bar of SQL Editor or Visual Query Builder respectively;
- view/edit the returned data on the Result tab.

7.1 SQL Editor

SQL Editor is a tool for creating and executing SELECT queries. It allows you to create and edit SQL text for the query, prepare and execute queries, and view the results of execution. To open SQL Editor, select the Tools | SQL Editor main menu item. The most popular query management actions (creating, editing, deleting) are covered by the corresponding topic 311.

To use the editor for working with several queries, open new query tab with the Create new query link on the Navigation bar. With the tabs' popup menu you can create a new query, close existing one, save the query to profile, etc even if editor's navigation bar is closed. Queries' tabs can been 431 displayed at the all sides of the editor (bottom, top, left or right).

For more information about query executing and working with query result see the corresponding topic 315.

Working with query text

The popup menu of the editing area provides you with standard operations for working with text such as *Cut* (**Ctrl+X**), *Copy* (**Ctrl+C**), *Paste* (**Ctrl +V**), *Undo* (**Ctrl +Z**), *Redo* (**Shift+Ctrl+Z**) along with a possibility to convert selected text to different cases (*lower, UPPER,* and *NameCase*).

You can also comment/uncomment selected text (**Shift+Ctrl+.** and **Shift+Ctrl+,** shortcuts respectively). If no text is selected, the whole line will be commented. By the way, it is not necessary to select commented text to uncomment it, just press **Shift+Ctrl+,** having the cursor inside the commented text. Both kinds of comments (single-line and multi-line) are supported. SQL Formatter 314 is also at your disposal.

SQL Editor allows you to use Visual Query Builder 318 modal instance to design query visually and load the result query text directly in the editor area. For this purpose use the Design query link of the editor area's popup menu.

Code completion

MS SQL Maestro provides you with code completion (as on the screen below) to select from a list of tables, columns, views, or other objects without having to manually enter the object's name in the editor. You can activate the completion list by pressing the **Ctrl+Space** key combination.

Syntax highlighting

Database objects are highlighted in the text. You can open the proper object editor by clicking the object name in the text with the **Ctrl** key pressed or with the Find Object link on the Navigation bar. To adjust the highlighting settings, use SQL highlight options 451.

Line modification markers

Lines of code that have been edited during the current session are indicated with a yellow line in the left margin of the editor. When you save the file, the yellow markers turn green. Thus at any time, yellow markers show changed but unsaved lines of code, and green markers show changes in this session that have been saved.

⊟ Find and replace text

Use find and replace to search for, and optionally, replace text in the SQL Editor. To open Find text/Replace text window, use Edit | Find/Replace main menu item, corresponding link of popup menu, or **Ctrl+F/Ctrl+H** shortcut. You can also use the Search again link to apply recent Find text dialog.

⊟ Managing the query text

To load query from .sql file, use the corresponding link on the Navigation bar. You can also find there links allowing you to save query text to file, export the contents of the editor to RTF and HTML formats (to file or to clipboard), copy the selected text from to clipboard as a ready-to-use string written in one of the following programming languages: C#, C++, Delphi (Object Pascal), and Java, and also print/preview the contents of the editor.

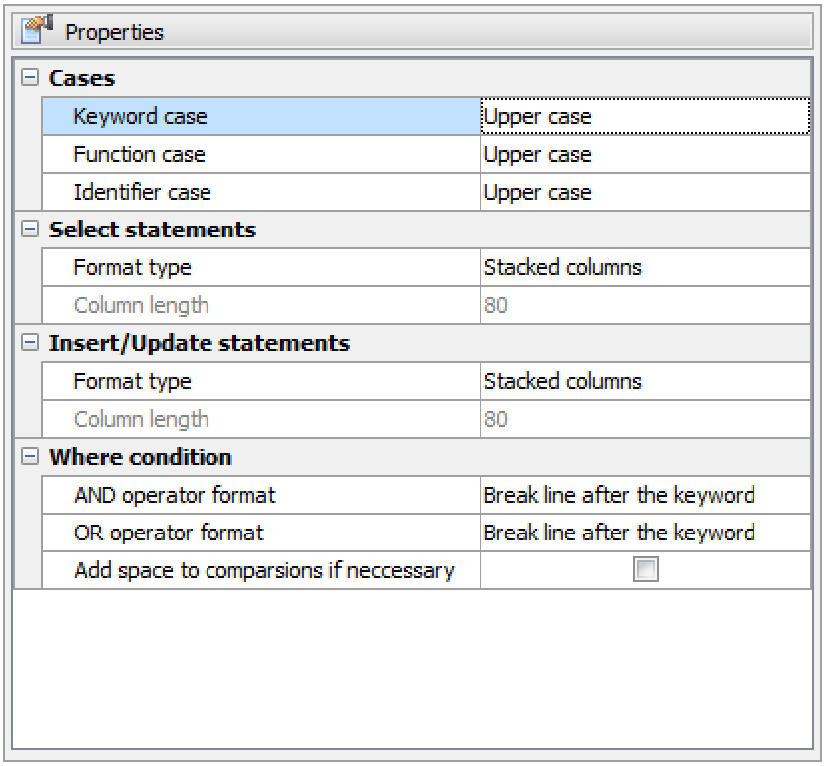
See also: Visual Query Builder 318, SQL Script Editor, SQL Editor Options 431

7.1.1 SQL Formatter

MS SQL Maestro provides you with SQL Formatter for DML statements (*SELECT*, *INSERT*, *UPDATE* and *DELETE*). It can be invoked through the Format SQL link on the SQL Editor's navigation bar (**Ctrl+Alt+D** shortcut).

The following options allows you to tune up SQL scripts according to your preferences.

- Cases (for keywords, functions, and identifiers);
- Format type and column length for *INSERT/UPDATE*, and *SELECT* statements;
- *AND* and *OR* operators format.



Cases	
Keyword case	Upper case
Function case	Upper case
Identifier case	Upper case
Select statements	
Format type	Stacked columns
Column length	80
Insert/Update statements	
Format type	Stacked columns
Column length	80
Where condition	
AND operator format	Break line after the keyword
OR operator format	Break line after the keyword
Add space to comparsions if neccessary	☐

7.1.2 Executing query

SQL Editor provides you with several variants of the query executing.

- To execute all statements of the text area with result data, click the Execute query item of the Navigation bar or use **F5**, **F8**, or **F9** shortcuts. Statements of each tab of SQL Editor are executed together in a separate thread in order to continue your work with the software while the query is executing.

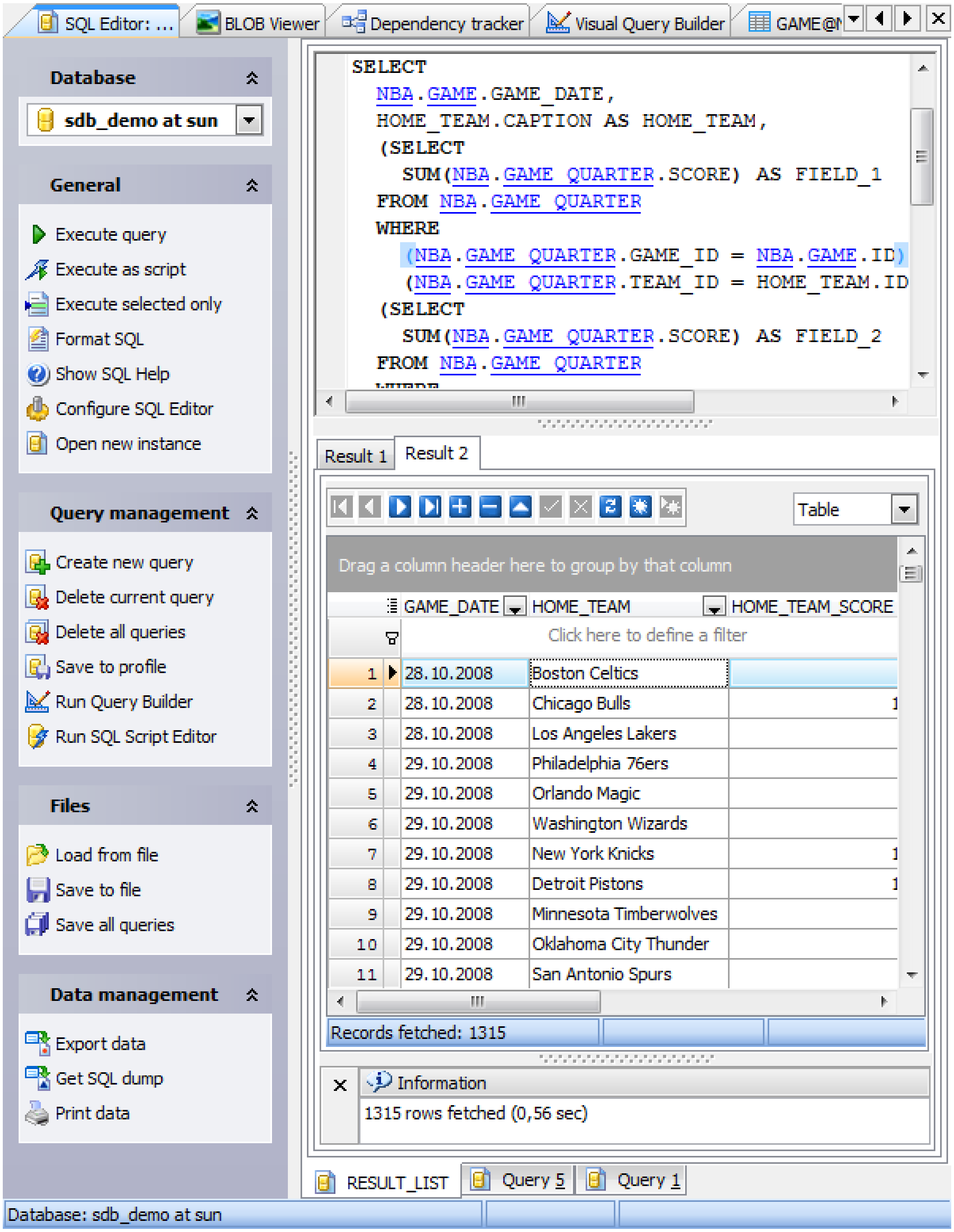
- You can also execute query as script (**Shift+F5**, **Shift+F8**, **Shift+F9**). In this case the query does not return data.

- To execute only a selected part of the query text, use Execute selected only or the **Alt+F5**, **Alt+F8**, **Alt+F9** shortcuts.

- There is also a possibility to execute a statement at the cursor position. For this purpose, use the Execute at cursor link at the Navigation bar or use the **Ctrl+F5**, **Ctrl+F8**, or **Ctrl+F9** shortcuts.

If the query text is correct, the query is executed, and if the query statement is supposed to return data (e.g. SELECT statement), the Result tab opens with the data returned by the query. If an error occurs while executing the query, execution stop is stopped and the appropriate error message is displayed in the Information tab.

The Result area displays the result data in grid. All principles of working with data you can find in Data Management 327 section.



```
SELECT
  NBA.GAME.GAME_DATE,
  HOME_TEAM.CAPTION AS HOME_TEAM,
  (SELECT
    SUM(NBA.GAME_QUARTER.SCORE) AS FIELD_1
  FROM NBA.GAME_QUARTER
  WHERE
    (NBA.GAME_QUARTER.GAME_ID = NBA.GAME.ID)
    (NBA.GAME_QUARTER.TEAM_ID = HOME_TEAM.ID
  (SELECT
    SUM(NBA.GAME_QUARTER.SCORE) AS FIELD_2
  FROM NBA.GAME_QUARTER
```

	GAME_DATE	HOME_TEAM	HOME_TEAM_SCORE
1	28.10.2008	Boston Celtics	
2	28.10.2008	Chicago Bulls	1
3	28.10.2008	Los Angeles Lakers	
4	29.10.2008	Philadelphia 76ers	
5	29.10.2008	Orlando Magic	
6	29.10.2008	Washington Wizards	
7	29.10.2008	New York Knicks	1
8	29.10.2008	Detroit Pistons	1
9	29.10.2008	Minnesota Timberwolves	
10	29.10.2008	Oklahoma City Thunder	
11	29.10.2008	San Antonio Spurs	

Records fetched: 1315

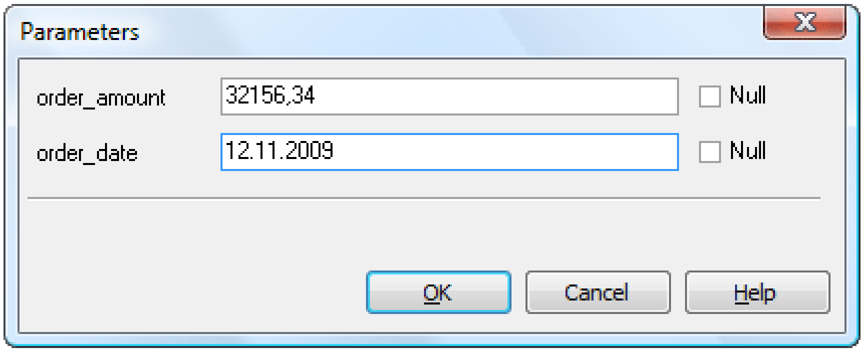
1315 rows fetched (0,56 sec)

Database: sdb_demo at sun

7.1.3 Query Parameters

Both SQL Editor 313 and Visual Query Builder 318 admit to using parameters inside the query text. A parameter is a kind of variable. Its value can be specified just before the query execution in the Parameters window. In the query text the parameter should appear as an identifier with a colon (':') at its beginning, e.g. :*param1*.

The Parameters dialog is used to specify the query parameters as well as values of the input parameters of procedures or functions before the execution. Enter parameter values and click the OK button to apply the values and execute the query or use the Cancel button to abort the execution.



order_amount 32156,34

order_date 12.11.2009

Note: To allow use parameters in query text, check the corresponding option at the Tools 428 tab of MS SQL Maestro Options.

7.2 Visual Query Builder

Visual Query Builder is provided for building data manipulation statements visually. It allows you to create and edit queries without knowledge of SQL, prepare and execute queries, and view the results of the execution. Builder can produce *INSERT*, *UPDATE* and *DELETE* statements as well as the *SELECT* statements containing subqueries and/or *UNIONs*. One instance of the builder can be used only for one query at a time. To open Visual Query Builder, select the Tools | Query Builder main menu item.

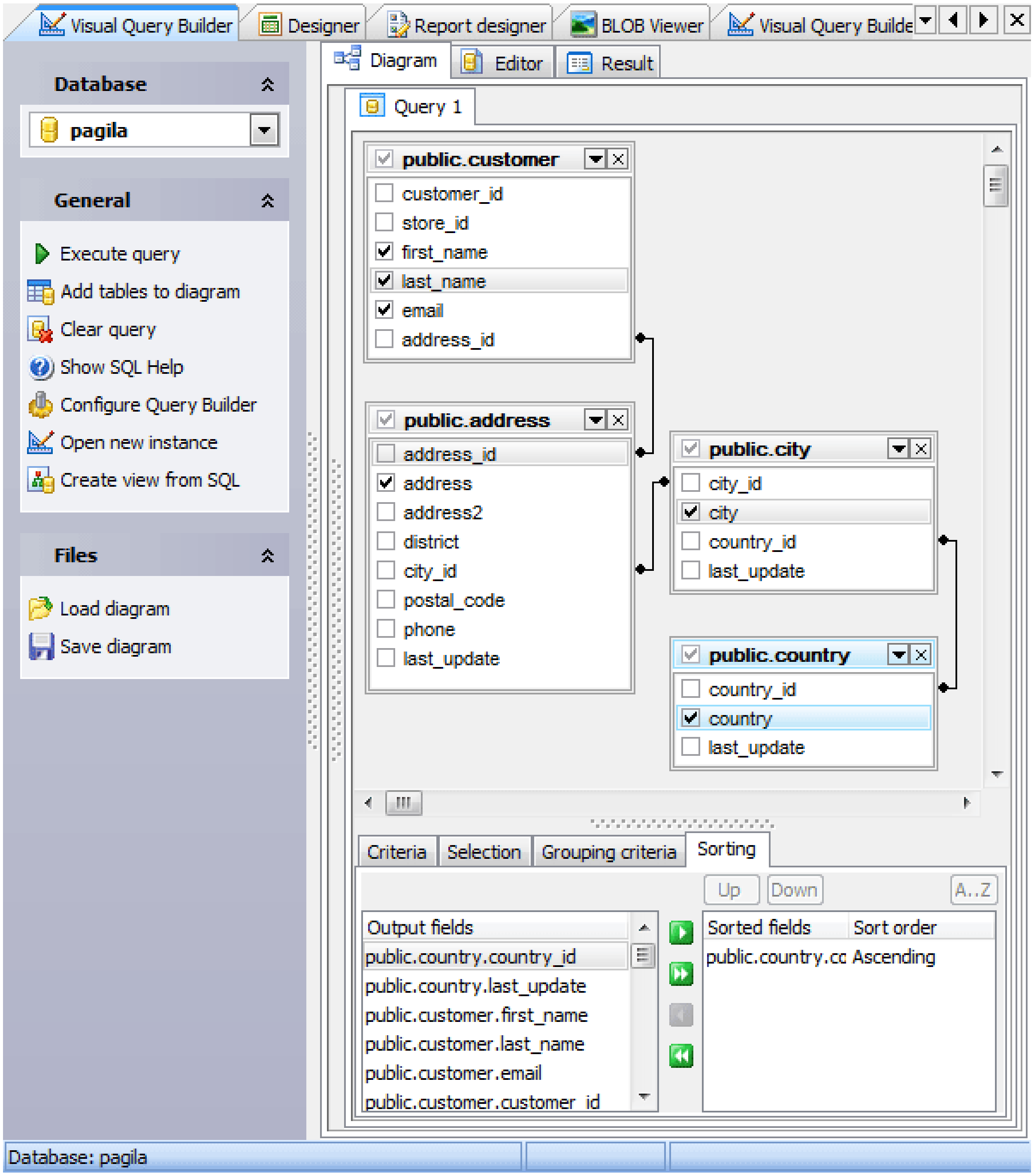
The most popular query management actions (creating, editing, deleting) are covered by the corresponding topic 311.

Builder consists of 3 tabs:

- Diagram 319 - to create a query from a graphical interface,
- Editor 324 – to modify the query text before its executing,
- Result 325 (appears after the query executing) – for working with data the query returns.

The builder also allows you to create a view based on the prepared query. For this purpose after the query creating and possibly testing use the Create view from SQL link at the Navigation bar to invoke the corresponding window, and specify view properties 109.

See also: SQL Editor 313, Visual Query Builder Options 433, Query Parameters 317



7.2.1 Creating query diagram

The Diagram tab is the main area of Visual Query Builder. Using its graphical interface you can select tables and views, join or select columns, and add conditions to the statement.

The Query Explorer field occupies the left side of Visual Query Builder main window. All the queries included in the result query (unions, subqueries) are represented at the Query Explorer for prompt access. They are grouped by kind and listed under the according node.

Below step-by-step description of query diagram creating.

- **Select the statement type** from the drop-down list at the top of the Diagram tab (*SELECT, INSERT, UPDATE, DELETE*).

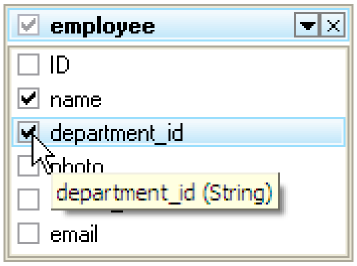
Add required tables to the Diagram area.

Use the Add Table(s) link of the area popup menu and select tables from the opened window (Use **Ctrl** or **Shift** pressed to select several tables).
To add only one table, simply drag it from the Database Explorer or from Object Manager/Browser to the Diagram area.

To remove the object, close its window or select the object and press the **Delete** key.

Pick up columns with data to output

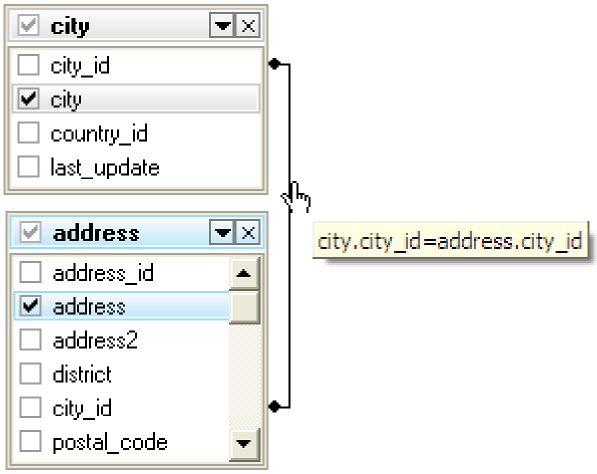
To include a table field to the query, tick off the option box to the left of the field name in the list or double-click it to see the blue icon next to the field name.



To include all the table fields, tick off the option box to the left of the table caption.
In case none fields is included, the SQL statement is generated as SELECT * FROM <Table_Name>, i.e. all the fields are selected.
To remove the fields from the query, uncheck the corresponding boxes.

Join tables if necessary

Visual Query Builder supports *INNER JOIN*, *LEFT OUTER JOIN*, and *RIGHT OUTER JOIN*. To associate database objects by two fields, drag a field from the first object's field list to a field from another object's field list. This will set a link between these objects by the selected fields. After you finish dragging, a line will appear between the linked fields. By default *INNER JOIN* syntax will be used.



city
☐ city_id
☑ city
☐ country_id
☐ last_update

address
☐ address_id
☑ address
☐ address2
☐ district
☐ city_id
☐ postal_code

city.city_id=address.city_id

You can view the properties of the object association from the query tab directly. Just set the cursor to the link line. A hint containing the association condition will appear.
To edit the properties, select the Properties item from the popup menu. A dialog window will appear, there you can change the association condition by selecting it from the list (=, >, <, >=, <=, <>). To create *LEFT OUTER JOIN / RIGHT OUTER JOIN* statements, check All rows from first_table/All rows from second_table from the window.

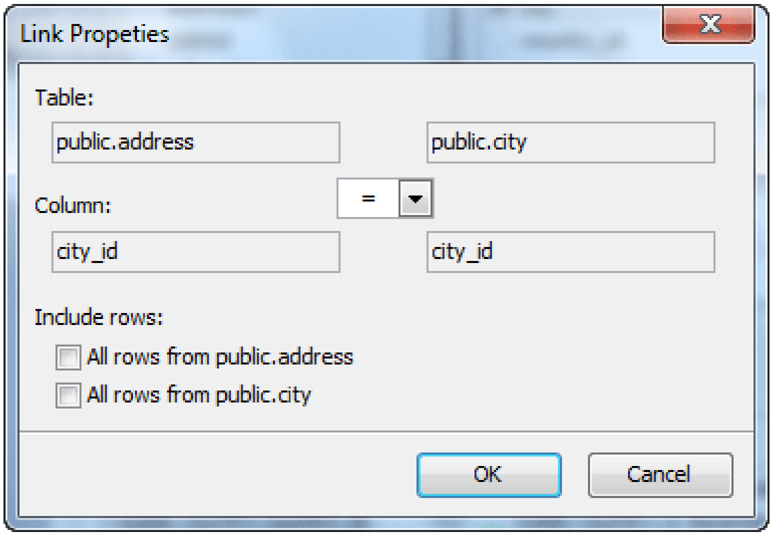


Table:
public.address
public.city

Column:
city_id = city_id

Include rows:
☐ All rows from public.address
☐ All rows from public.city

To remove a link between objects, select the Delete Link item from the popup menu.

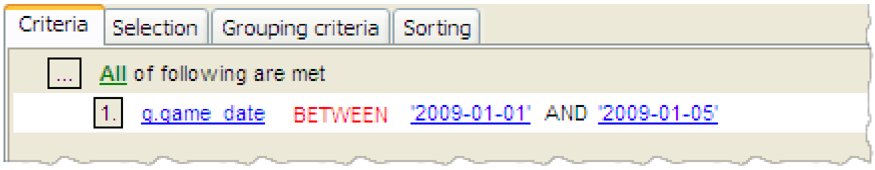
To delete all the links of an object, click the '-' button next to the object alias. To insert a point to the link line, select the Insert Point item from the popup menu, and the new point will appear. Using the point you can move the link line. It does not cause any changes in the query but makes the diagram performance vivid and the visual building more obvious.

Specify WHERE condition

Criteria tab allows you to set the selection conditions. To add a condition, click the button on the left and select the Add condition item in the popup menu. Edit the condition by clicking its parts and setting their values. Clicking the button to the left of the condition string activates the popup menu which allows you to add a new condition of the same enclosure level, add a new enclosure level, delete the current condition, open or close the condition if it is composite.

A simple condition string contains three fields: an argument, a condition and a second argument (if required for the condition). Clicking each field allows you to set its value. Clicking the argument field make it possible to edit the argument as a text field. You can set a table name or a definite value in this field. The popup menu of the field in the editing mode which contains the Insert Field function (also called by the **Shift+Enter** hot keys combination).

This function allows you to choose a field from the list of all the table fields available in the query. The popup menu of the condition field allows you to specify the condition you need. The way of proceeding the condition is set in the upper string of the area (*All, Any, None,* or *Not all* of the following are possible variants). Click the underlined word to modify it.



All of following are met

1. g.game_date BETWEEN '2009-01-01' AND '2009-01-05'

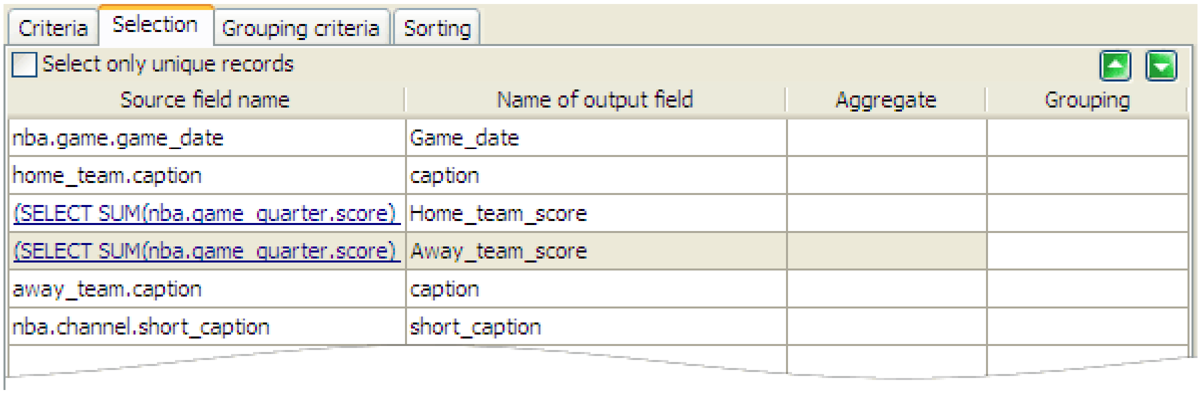
Create subquery if necessary

You can add one or more subqueries to further limit the tables and records returned from a *SELECT* statement when setting a *WHERE* condition in the query builder. To add subquery:
- open Criteria tab;
- click the button on the left and select the Add condition item in the popup menu;
- right click on an argument field and use the Insert query link of the popup menu;
- build the subquery in the new query tab that have appeared in the Diagram area,

or
- open Selection tab;
- use the Insert query link of the popup menu;
- build the subquery in the new query tab that have appeared in the Diagram area.

Use column aliases

You can set/edit the object alias directly from the query tab by double-clicking the object caption.



☐ Select only unique records

Source field name	Name of output field	Aggregate	Grouping
nba.game.game_date	Game_date		
home_team.caption	caption		
(SELECT SUM(nba.game_quarter.score)	Home_team_score		
(SELECT SUM(nba.game_quarter.score)	Away_team_score		
away_team.caption	caption		
nba.channel.short_caption	short_caption		

In case the alias is used as the expression's column name use the Selection tab displays the output fields of the query. It allows you to edit the names of the query or CASE output fields, set their displaying order and set the aggregate functions (*SUM, MIN, MAX, AVG*, etc.) for each field.

AVG	Returns the average of the values in a group
BIT_AND	Returns the bitwise AND of all bits in the expression.
BIT_OR	Returns the bitwise OR of all bits in the expression.
COUNT	Returns the total number of items in a column. This function does not ignore NULL values when calculating results.
GROUP_CONCAT	Returns a string result with the concatenated non-NULL values from a group.
MAX	Returns the maximum value for the column.
MIN	Returns the minimum value for the column.
STD	Returns the population standard deviation of the expression.
STDDEV	Returns the sample standard deviation of a numeric expression evaluated over a set.
SUM	Returns the sum of all the values in the expression.
VARIANCE	Returns the population standard variance of the expression.

To remove the field from the list, select the Delete current row item from the popup menu of the field row.
To modify the input query field, double-click it and then type the field name or select one from the drop-down list.
To modify the output query field name, double-click it and enter the field name.

DISTINCT option

To specify removal of duplicate rows from the result set, open the Selection tab and check the Select only unique records box.

Add HAVING statement

Set the conditions to be included into the HAVING statement within the Grouping Criteria tab. They are set in the same way as the *WHERE* conditions. To set the aggregate function for the field, double-click the field row in the Aggregate column and then type the function name or select one from the drop-down list.

ORDER BY clause

Set the way of sorting the query records within the Sorting tab. The field list on the left represents all the output query fields; the list on the right contains fields by which the query records will be sorted. To move the field from one list to another, drag the selected field or use the Add and Remove buttons. To change the sorting order, select a field in the right list and move it using the Up and Down buttons.

To change the sorting direction, select a filed in the right list and switch the direction (*Ascending, Descending*) using the A..Z/Z..A button.

⊟ Create UNIONs

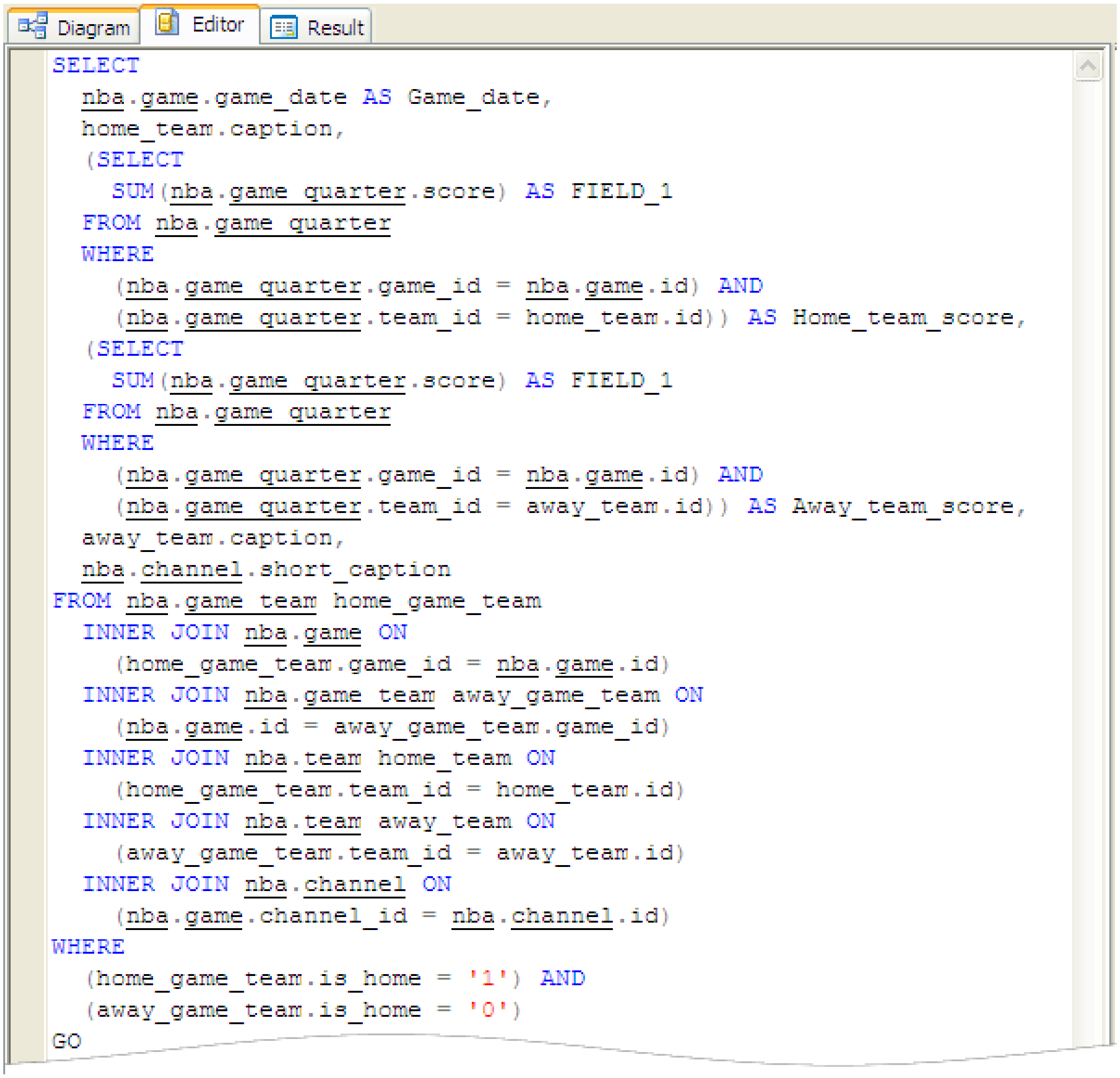
To combine the result from multiple SELECT statements into a single result set, use the Add union link of the Query Explorer popup menu.

Note: The column names from the first SELECT statement are used as the column names for the results returned.
Selected columns listed in corresponding positions of each SELECT statement should have the same data type.

7.2.2 Working with editor area

In the Editor area the query text is automatically generated while you are building the query.

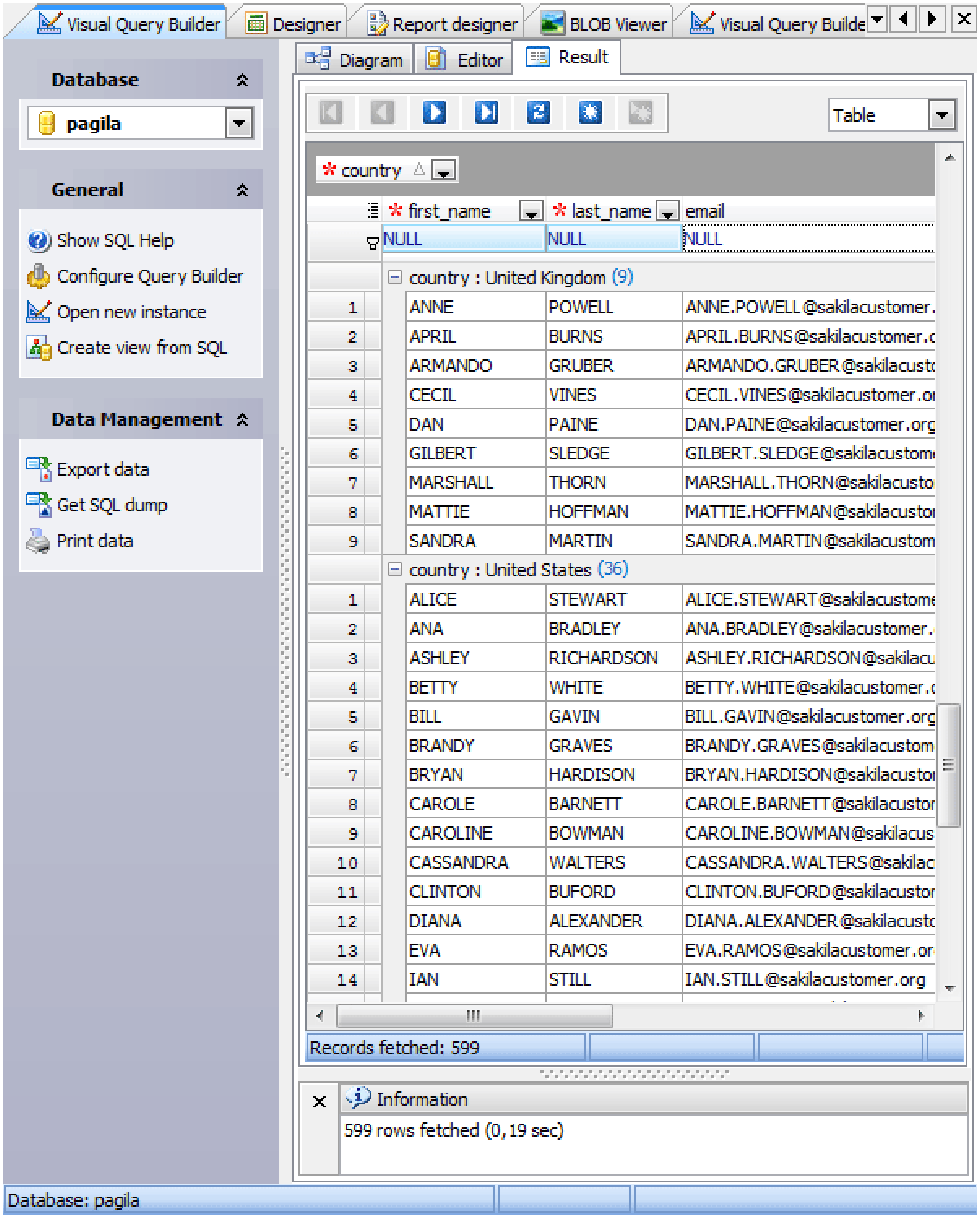
You can edit this text according to the rules of SQL, and all the changes will be displayed on the Diagram page of Visual Query Builder.



```
SELECT
  nba.game.game_date AS Game_date,
  home_team.caption,
  (SELECT
    SUM(nba.game_quarter.score) AS FIELD_1
  FROM nba.game_quarter
  WHERE
    (nba.game_quarter.game_id = nba.game.id) AND
    (nba.game_quarter.team_id = home_team.id)) AS Home_team_score,
  (SELECT
    SUM(nba.game_quarter.score) AS FIELD_1
  FROM nba.game_quarter
  WHERE
    (nba.game_quarter.game_id = nba.game.id) AND
    (nba.game_quarter.team_id = away_team.id)) AS Away_team_score,
  away_team.caption,
  nba.channel.short_caption
FROM nba.game_team home_game_team
  INNER JOIN nba.game ON
    (home_game_team.game_id = nba.game.id)
  INNER JOIN nba.game_team away_game_team ON
    (nba.game.id = away_game_team.game_id)
  INNER JOIN nba.team home_team ON
    (home_game_team.team_id = home_team.id)
  INNER JOIN nba.team away_team ON
    (away_game_team.team_id = away_team.id)
  INNER JOIN nba.channel ON
    (nba.game.channel_id = nba.channel.id)
WHERE
  (home_game_team.is_home = '1') AND
  (away_game_team.is_home = '0')
GO
```

7.2.3 Executing query

To execute the query select the Execute item in the navigation bar. After that the Result tab is displayed. This page contains the result data returned by the query, as a grid (see Data View for details). The popup menu of this tab and the items of the navigation bar allow you to export data and get SQL dump.



	first_name	last_name	email
	NULL	NULL	NULL
	country : United Kingdom (9)		
1	ANNE	POWELL	ANNE.POWELL@sakilacustomer.
2	APRIL	BURNS	APRIL.BURNS@sakilacustomer.c
3	ARMANDO	GRUBER	ARMANDO.GRUBER@sakilacust
4	CECIL	VINES	CECIL.VINES@sakilacustomer.o
5	DAN	PAINE	DAN.PAINE@sakilacustomer.org
6	GILBERT	SLEDGE	GILBERT.SLEDGE@sakilacustom
7	MARSHALL	THORN	MARSHALL.THORN@sakilacusto
8	MATTIE	HOFFMAN	MATTIE.HOFFMAN@sakilacusto
9	SANDRA	MARTIN	SANDRA.MARTIN@sakilacustom
	country : United States (36)		
1	ALICE	STEWART	ALICE.STEWART@sakilacustome
2	ANA	BRADLEY	ANA.BRADLEY@sakilacustomer.
3	ASHLEY	RICHARDSON	ASHLEY.RICHARDSON@sakilacu
4	BETTY	WHITE	BETTY.WHITE@sakilacustomer.c
5	BILL	GAVIN	BILL.GAVIN@sakilacustomer.org
6	BRANDY	GRAVES	BRANDY.GRAVES@sakilacustom
7	BRYAN	HARDISON	BRYAN.HARDISON@sakilacusto
8	CAROLE	BARNETT	CAROLE.BARNETT@sakilacusto
9	CAROLINE	BOWMAN	CAROLINE.BOWMAN@sakilacus
10	CASSANDRA	WALTERS	CASSANDRA.WALTERS@sakilac
11	CLINTON	BUFORD	CLINTON.BUFORD@sakilacusto
12	DIANA	ALEXANDER	DIANA.ALEXANDER@sakilacust
13	EVA	RAMOS	EVA.RAMOS@sakilacustomer.or
14	IAN	STILL	IAN.STILL@sakilacustomer.org

Records fetched: 599

599 rows fetched (0,19 sec)

Database: pagila

8 Data Management

Query results and table data are displayed on the Data 81 or Result 315 tabs of Table Editor 79, SQL Editor 313 or Visual Query Builder 318.

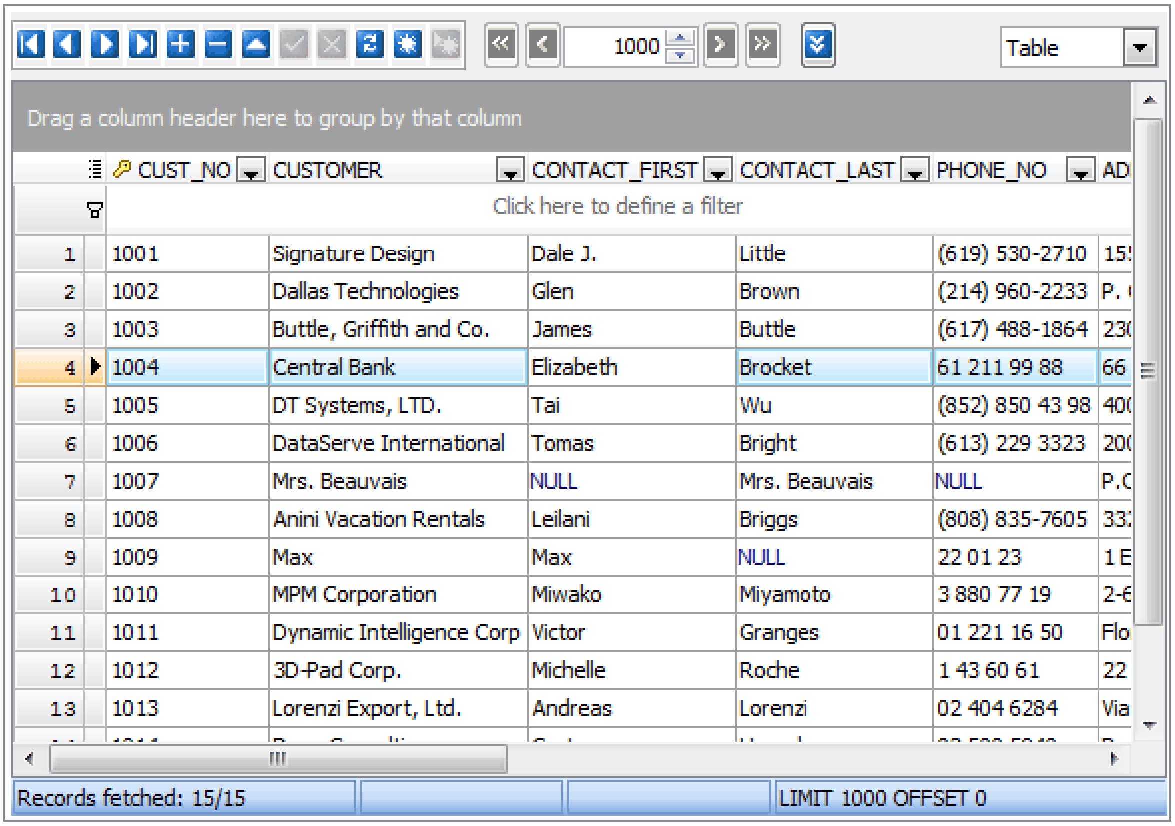
Data are displayed as a grid (or as info cards) which provide a lot of useful features such as editing, grouping, sorting, filtering, etc. See Data View 328 for details.

Navigation bars of these tabs as well as popup menus of their working areas places at your disposal the following functions for managing data:

- Export Data 344 allows you to export data to various formats, including MS Excel, MS Access, RTF, HTML, PDF and more.
- Get SQL Dump 351 exports data to the SQL script as a number of INSERT statements.
- Import Data 354 provides you with possibility to import data from MS Excel, MS Access, DBF, XML, TXT, and CSV.
- Edit BLOB 339 allows you to view and edit the content of BLOB and TEXT fields.

8.1 Data View

MS SQL Maestro represents all data (stored in tables and views, results of queries and procedures) in grid 329 or in info cards 334. By default, data is displayed in a grid - tabular view of data. To change the type of the data representation, use the drop-down list at the top of the tab. Both of the data representations support UNICODE/UTF-8 data. The status bar displays the number of records in the current data set. To reset grid to default settings, open the Data tab when holding the **Ctrl** key.


1000
Table
Drag a column header here to group by that column
Click here to define a filter

	CUST_NO	CUSTOMER	CONTACT_FIRST	CONTACT_LAST	PHONE_NO	AD
1	1001	Signature Design	Dale J.	Little	(619) 530-2710	15
2	1002	Dallas Technologies	Glen	Brown	(214) 960-2233	P.
3	1003	Buttle, Griffith and Co.	James	Buttle	(617) 488-1864	23
4	1004	Central Bank	Elizabeth	Brocket	61 211 99 88	66
5	1005	DT Systems, LTD.	Tai	Wu	(852) 850 43 98	40
6	1006	DataServe International	Tomas	Bright	(613) 229 3323	20
7	1007	Mrs. Beauvais	NULL	Mrs. Beauvais	NULL	P.C
8	1008	Anini Vacation Rentals	Leilani	Briggs	(808) 835-7605	33
9	1009	Max	Max	NULL	22 01 23	1 E
10	1010	MPM Corporation	Miwako	Miyamoto	3 880 77 19	2-6
11	1011	Dynamic Intelligence Corp	Victor	Granges	01 221 16 50	Flo
12	1012	3D-Pad Corp.	Michelle	Roche	1 43 60 61	22
13	1013	Lorenzi Export, Ltd.	Andreas	Lorenzi	02 404 6284	Via

Records fetched: 15/15

LIMIT 1000 OFFSET 0

Navigation buttons

Both data representations are equipped with navigation buttons. They are represented at the top of the data tab and allow you to navigate between records and to accomplish common operations:

- To add a new record, use the *Plus* button or the **Insert** shortcut.
- To delete a new record, use the *Minus* button or the **Delete** shortcut.
- To edit an existing record, push the corresponding button or invoke the Data Input Form 335 using popup menu of the necessary record, with **Ctrl+Alt+D** shortcut, or with the corresponding link at the Navigation bar. To edit a field value, click it and enter the new one inline.

The pagination option allows you to limit the number of browsed records. By default, the number of records represented in grid at once is 1000. To change the number of records represented in the current grid, enter the necessary value in the pagination bar. To specify the default one or to disable pagination, use the data grid option 440.

Navigation bar

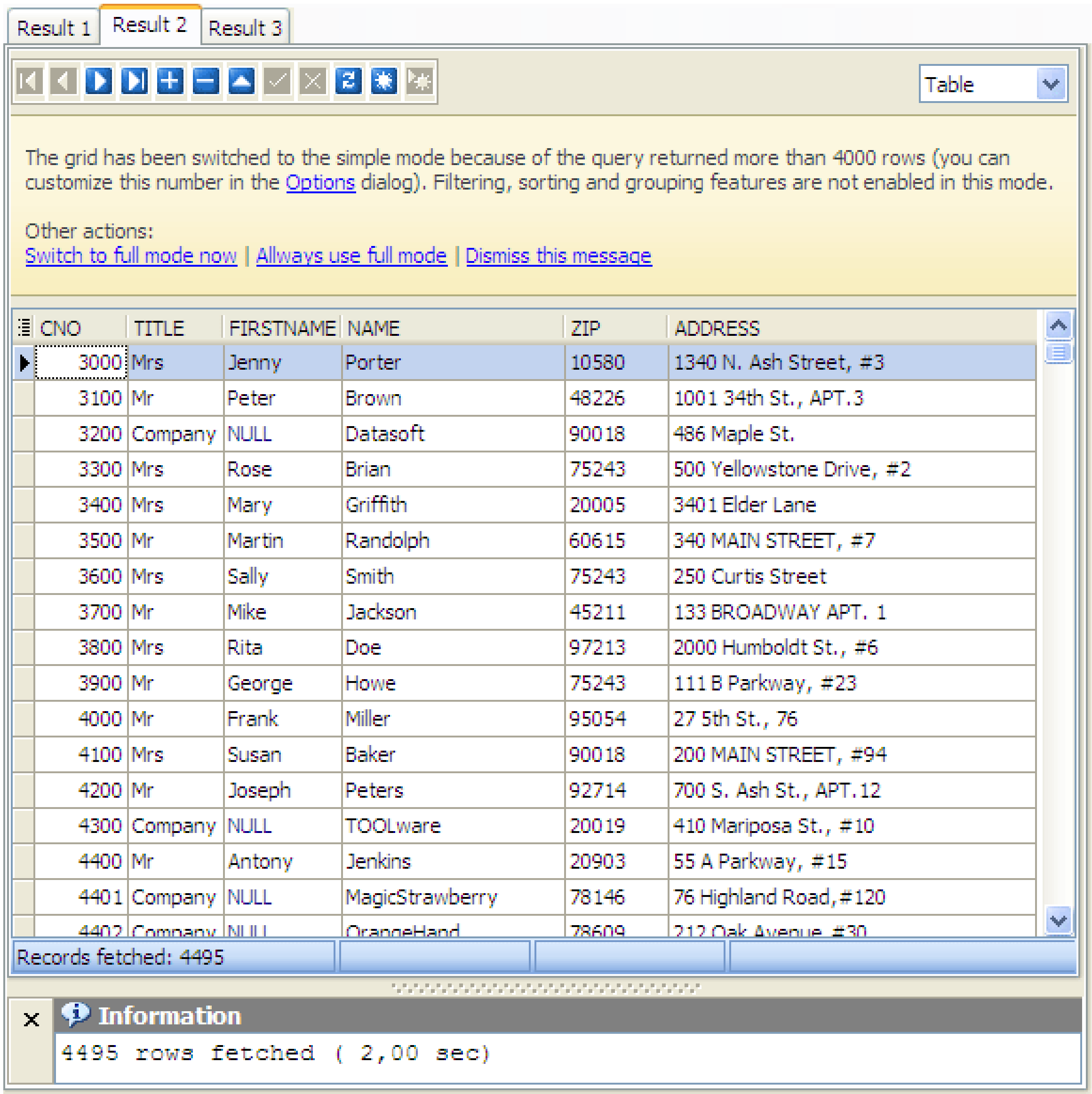
The Data management group of the Navigation bar allows to invoke Data Input Form 335, SQL Editor 313 with SELECT query, Data Export 344, and Data Import 354 modules using corresponding links, also get SQL dump 351 of the current data set and print current data with enabled preview in WYSIWYG mode.

See also: Table Editor 79, SQL Editor 313, Visual Query Builder 318

8.1.1 Working with data grid

Our software offers two grid modes:

- the full grid mode is a fully functional data representation equipped with abilities to filter and to sort data;
- the simple grid mode is provided for working with large number of records. For speed-up data fetching, filtering and sorting abilities are not enabled in this mode. The notification bar at the top of the grid (see the picture below) announces that the grid has been switched to the simple mode.



The grid has been switched to the simple mode because of the query returned more than 4000 rows (you can customize this number in the Options dialog). Filtering, sorting and grouping features are not enabled in this mode.

Other actions:
Switch to full mode now | Allways use full mode | Dismiss this message

CNO	TITLE	FIRSTNAME	NAME	ZIP	ADDRESS
3000	Mrs	Jenny	Porter	10580	1340 N. Ash Street, #3
3100	Mr	Peter	Brown	48226	1001 34th St., APT.3
3200	Company	NULL	Datasoft	90018	486 Maple St.
3300	Mrs	Rose	Brian	75243	500 Yellowstone Drive, #2
3400	Mrs	Mary	Griffith	20005	3401 Elder Lane
3500	Mr	Martin	Randolph	60615	340 MAIN STREET, #7
3600	Mrs	Sally	Smith	75243	250 Curtis Street
3700	Mr	Mike	Jackson	45211	133 BROADWAY APT. 1
3800	Mrs	Rita	Doe	97213	2000 Humboldt St., #6
3900	Mr	George	Howe	75243	111 B Parkway, #23
4000	Mr	Frank	Miller	95054	27 5th St., 76
4100	Mrs	Susan	Baker	90018	200 MAIN STREET, #94
4200	Mr	Joseph	Peters	92714	700 S. Ash St., APT.12
4300	Company	NULL	TOOLware	20019	410 Mariposa St., #10
4400	Mr	Antony	Jenkins	20903	55 A Parkway, #15
4401	Company	NULL	MagicStrawberry	78146	76 Highland Road, #120
4402	Company	NULL	OrangeHand	78609	212 Oak Avenue, #30

Records fetched: 4495

```
4495 rows fetched ( 2,00 sec)
```

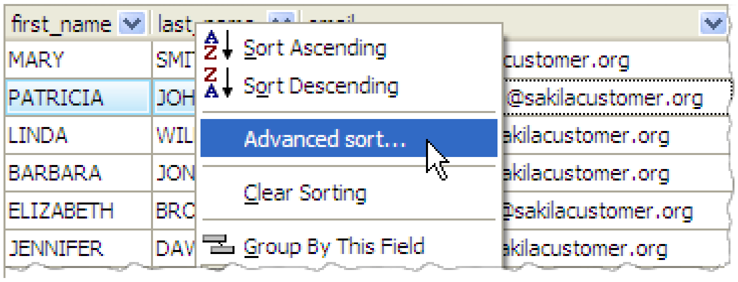
By default, the grid automatically switches to the simple mode for queries returning more than 5000 records (the number can be customized in the Options 440 dialog).

The following abilities are not available in the simple grid mode:

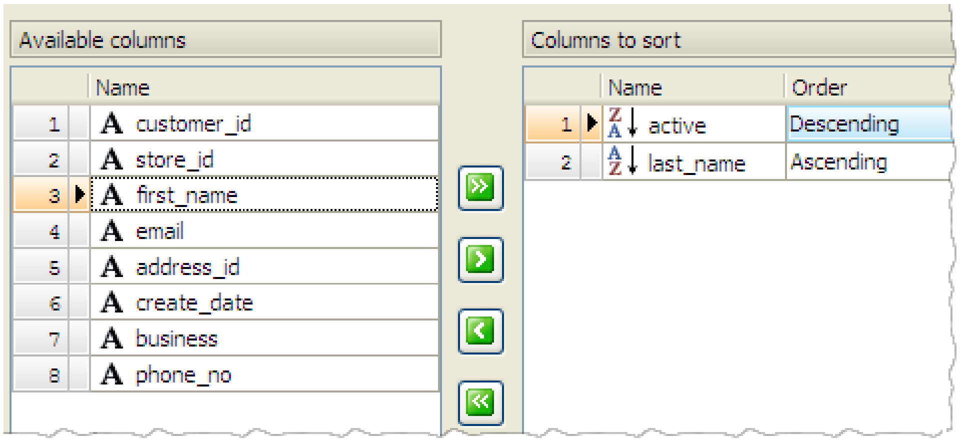
Sorting data (only in the full grid mode)

Click the column caption to sort data by the values of this column. To select sort order (ascending or descending), use popup menu of the column caption.

To sort data on a combination of grid columns, use the Advanced sort... link of the popup menu of the grid's header. The Advanced sorting window will be shown.



Select there the columns you want to sort from the Available columns list in the order of priority. Specify the sort order if necessary and click OK.



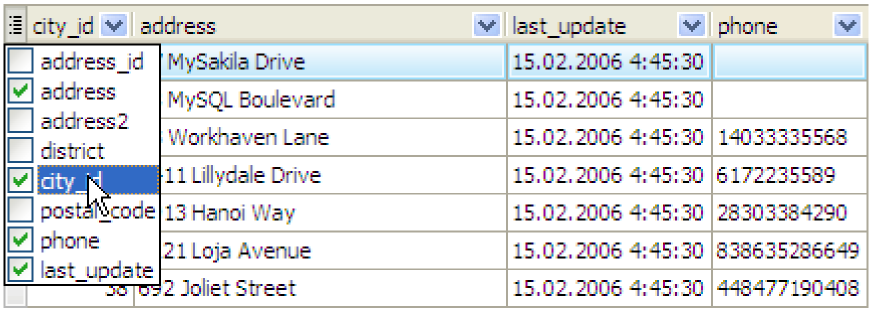
To cancel the sorting order, press **Ctrl** and click on the sorted column caption.

Filtering represented records (only in the full grid mode)

There are several ways to filter data represented in grid. See the corresponding topic 336 to find out their descriptions.

Hiding selected columns

You can show/hide columns using a button in the left top corner of the grid. Just check/uncheck the column in the drop-down list.

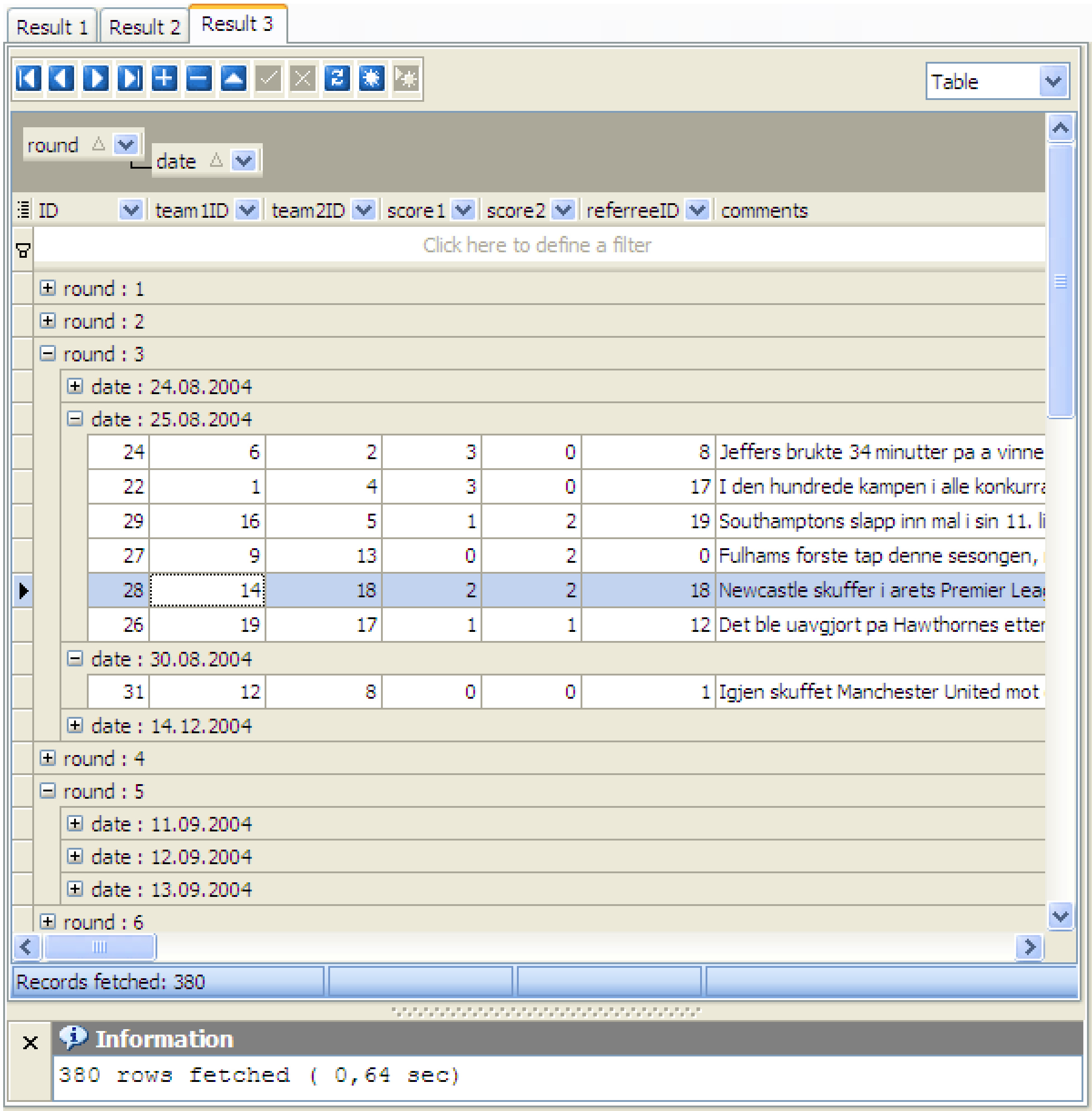


Columns reordering

To reorder columns, use drag-n-drop.

Grouping records

You can group grid data by any of the columns by dragging the column header to the destination area. Now all the records are displayed as subnodes to the grouping row value as shown in the picture. To reverse grouping, just drag the column name from the upper area back.



ID	team1ID	team2ID	score1	score2	referreeID	comments
round : 1						
round : 2						
round : 3						
date : 24.08.2004						
date : 25.08.2004						
24	6	2	3	0	8	Jeffers brukte 34 minutter pa a vinne
22	1	4	3	0	17	I den hundrede kampen i alle konkurra
29	16	5	1	2	19	Southamptons slapp inn mal i sin 11. li
27	9	13	0	2	0	Fulhams forste tap denne sesongen,
28	14	18	2	2	18	Newcastle skuffer i arets Premier Lea
26	19	17	1	1	12	Det ble uavgjort pa Hawthornes etter
date : 30.08.2004						
31	12	8	0	0	1	Igjen skuffet Manchester United mot
date : 14.12.2004						
round : 4						
round : 5						
date : 11.09.2004						
date : 12.09.2004						
date : 13.09.2004						
round : 6						

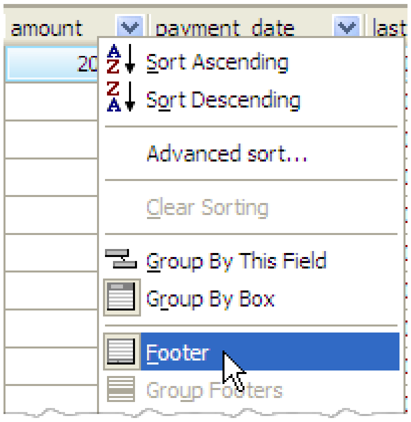
Records fetched: 380

Information

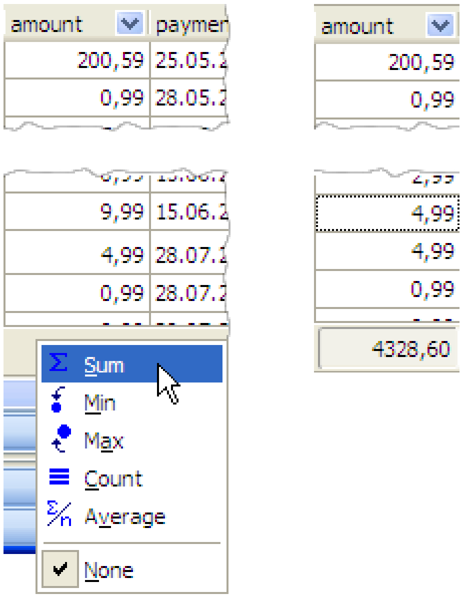
```
380 rows fetched ( 0,64 sec)
```

Using aggregate functions

To get a sum of column values, a min or a max value, an average column value or an amount of records, use Data Grid Footer. Select the Footer item at the grid caption's popup menu.



It will be shown at the bottom of the grid. The popup menu of the footer allows you to get an aggregate function result calculated with the corresponding column values.



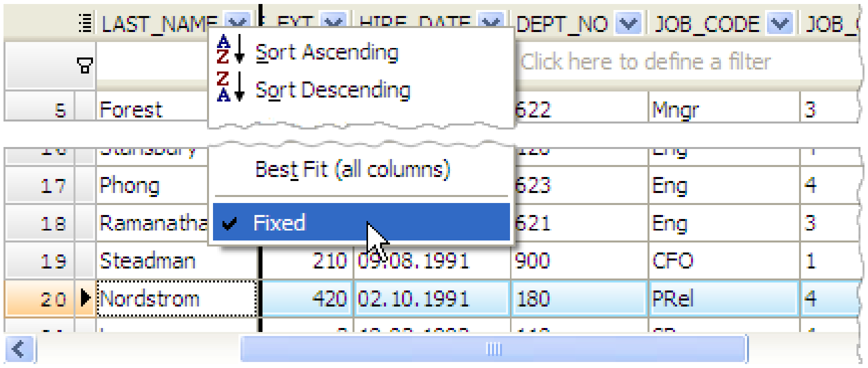
For grouped data use Group Footers.

Data alignment

The grid's header popup menu allows to align column data. Use the Alignment link and select the alignment type.

Fixing columns

You can fix grid columns to view them permanently when working with other grid data. To fix a column, choose the corresponding item from the grid's header popup menu.



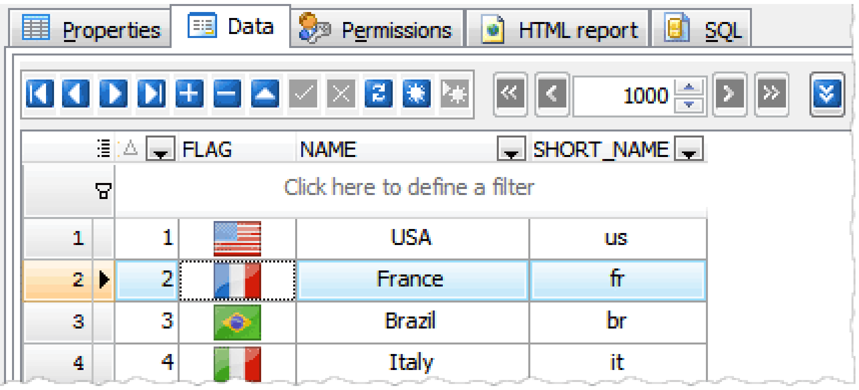
	LAST_NAME	EXT	HIRE_DATE	DEPT_NO	JOB_CODE	JOB_
5	Forest			622	Mngr	3
17	Phong			623	Eng	4
18	Ramanatha			621	Eng	3
19	Steadman	210	09.08.1991	900	CFO	1
20	Nordstrom	420	02.10.1991	180	PRel	4

Row numbering

There is also a possibility to display row numbers in grids. You can adjust 442 the corresponding column to yours liking.

Inline images

It is possible to display images directly in the grid as on the picture below.



		FLAG	NAME	SHORT_NAME
1	1		USA	us
2	2		France	fr
3	3		Brazil	br
4	4		Italy	it

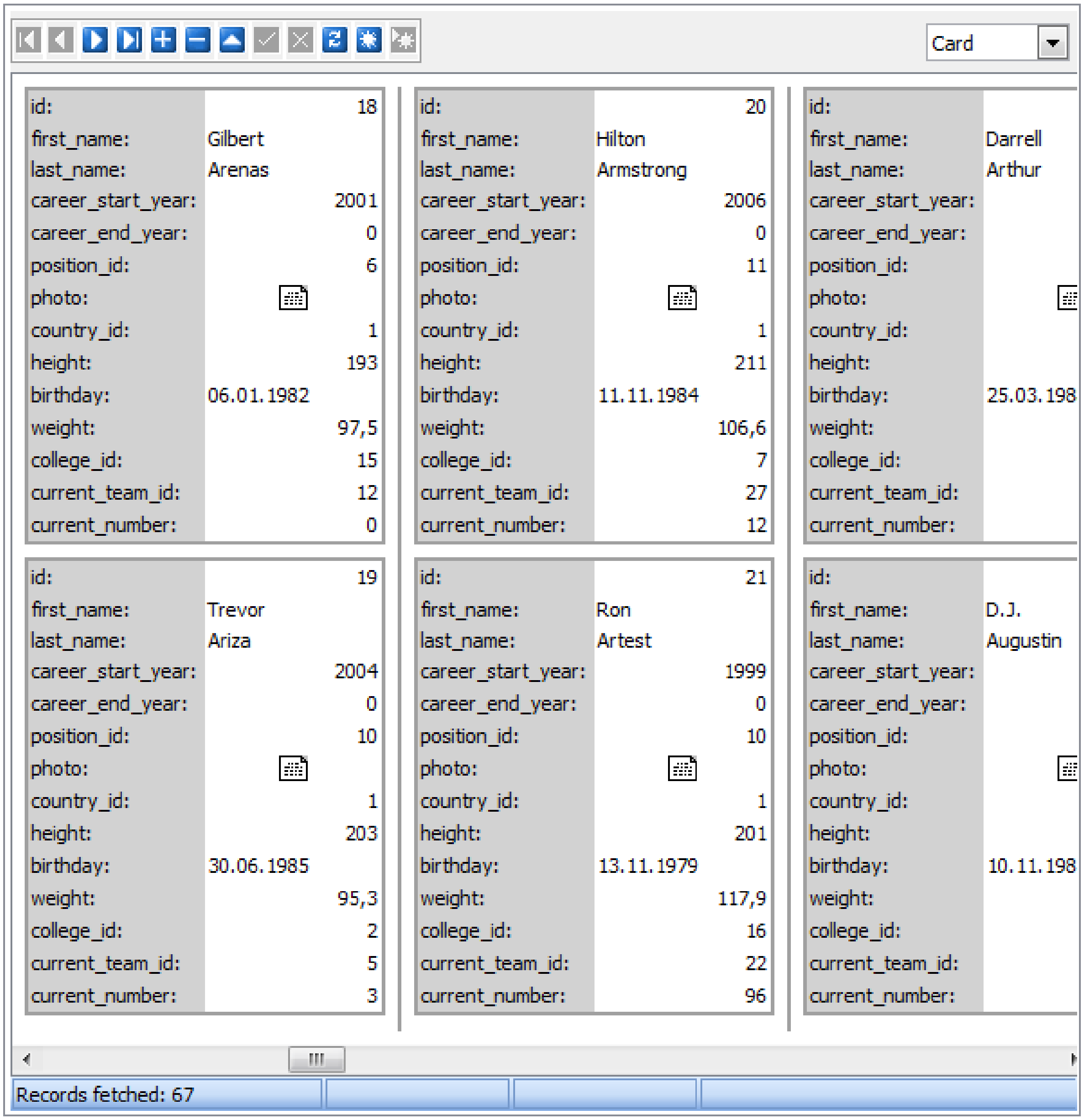
To enable/disable this view mode, open the *Enable inline images* window using the *Manage inline images* item of the column popup menu. The window options allow to set or change the image fitting and specify the row height. To add new images or change existing ones, use BLOB Editor 339 (see below).

Working with BLOBs

To edit a BLOB field 339, double click the field, or use the corresponding popup menu item. There are also possibilities to export all BLOBs stored in the table column to files and import BLOBs from a directory to the table columns. In this case you need to set the Target directory, specify the template to be used for file names and the column BLOBs to be exported from (imported to).

8.1.2 Working with info cards

Info cards correspond to the records. You can filter records by custom conditions 336 and edit data directly in info cards or with Data Input Form 335.



Card

id: 18
first_name: Gilbert
last_name: Arenas
career_start_year: 2001
career_end_year: 0
position_id: 6
photo:
country_id: 1
height: 193
birthday: 06.01.1982
weight: 97,5
college_id: 15
current_team_id: 12
current_number: 0

id: 20
first_name: Hilton
last_name: Armstrong
career_start_year: 2006
career_end_year: 0
position_id: 11
photo:
country_id: 1
height: 211
birthday: 11.11.1984
weight: 106,6
college_id: 7
current_team_id: 27
current_number: 12

id:
first_name: Darrell
last_name: Arthur
career_start_year:
career_end_year:
position_id:
photo:
country_id:
height:
birthday: 25.03.198
weight:
college_id:
current_team_id:
current_number:

id: 19
first_name: Trevor
last_name: Ariza
career_start_year: 2004
career_end_year: 0
position_id: 10
photo:
country_id: 1
height: 203
birthday: 30.06.1985
weight: 95,3
college_id: 2
current_team_id: 5
current_number: 3

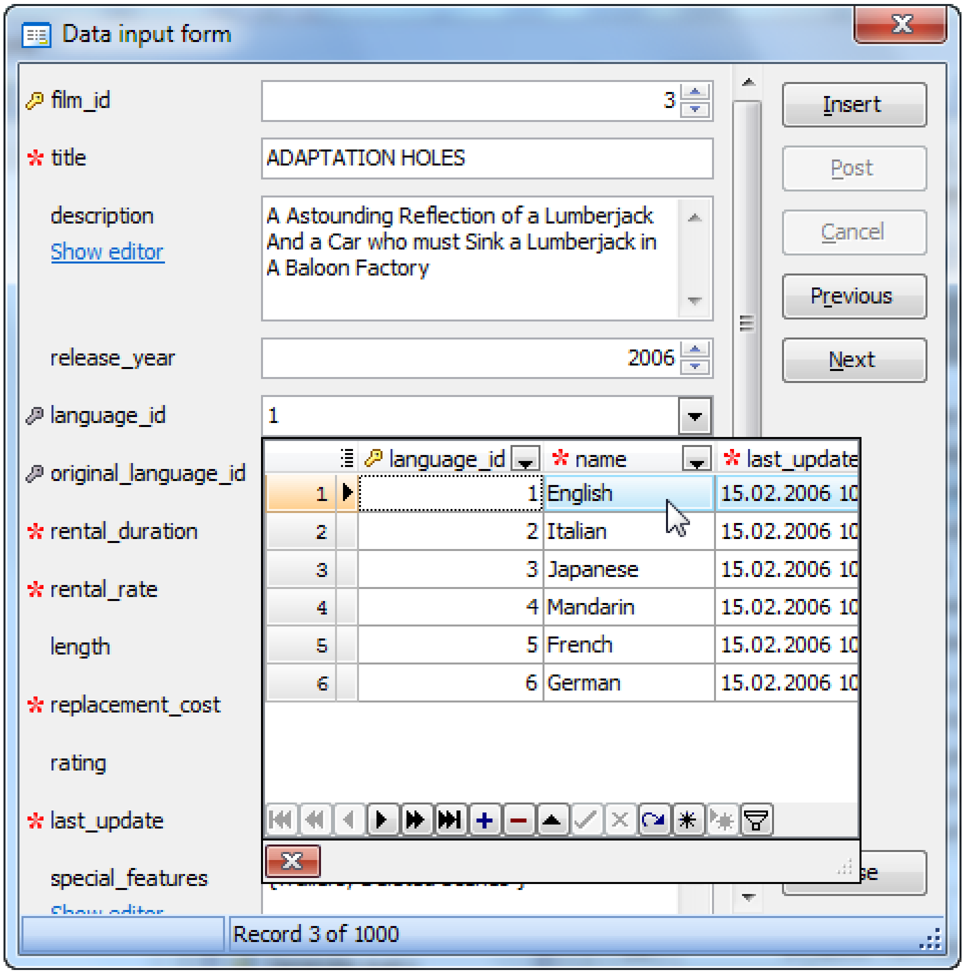
id: 21
first_name: Ron
last_name: Artest
career_start_year: 1999
career_end_year: 0
position_id: 10
photo:
country_id: 1
height: 201
birthday: 13.11.1979
weight: 117,9
college_id: 16
current_team_id: 22
current_number: 96

id:
first_name: D.J.
last_name: Augustin
career_start_year:
career_end_year:
position_id:
photo:
country_id:
height:
birthday: 10.11.198
weight:
college_id:
current_team_id:
current_number:

Records fetched: 67

8.1.3 Data input form

Use Data Input Form to add new records or edit existing ones. To invoke the dialog, use the corresponding link from the popup menu or **Ctrl+Alt+D** shortcut.



film_id	3
* title	ADAPTATION HOLES
description	A Astounding Reflection of a Lumberjack And a Car who must Sink a Lumberjack in A Baloon Factory
release_year	2006
language_id	1
original_language_id	
* rental_duration	
* rental_rate	
length	
* replacement_cost	
rating	
* last_update	
special_features	

	language_id	* name	* last_update
1	1	English	15.02.2006 1
2	2	Italian	15.02.2006 1
3	3	Japanese	15.02.2006 1
4	4	Mandarin	15.02.2006 1
5	5	French	15.02.2006 1
6	6	German	15.02.2006 1

Record 3 of 1000

The dialog's fields contain the values of the current grid row. Use the Insert button to enter values of a new record and the Post button to update the current row. The Cancel button reverts all the field values within a form to their initial values (or to the last posted values). The Previous and Next buttons allow you to switch between grid records without closing the dialog.

Controls containing values of primary and foreign key columns are marked with the 'gold key' and 'silver key' images accordingly. Controls containing values of required (NOT NULL) columns are marked with a red asterisk.

There are possibilities to use lookup editors on working with columns linked with foreign keys, a calendar for *timestamp* columns and a calculator for *decimal* ones.

8.1.4 Data filtering

MS SQL Maestro support filtering records by the following methods:

⊟ **Filter by a column value**

Select the Use as Filter item from the field popup menu to filter records by the current column value.

⊟ **Filter by several column values**

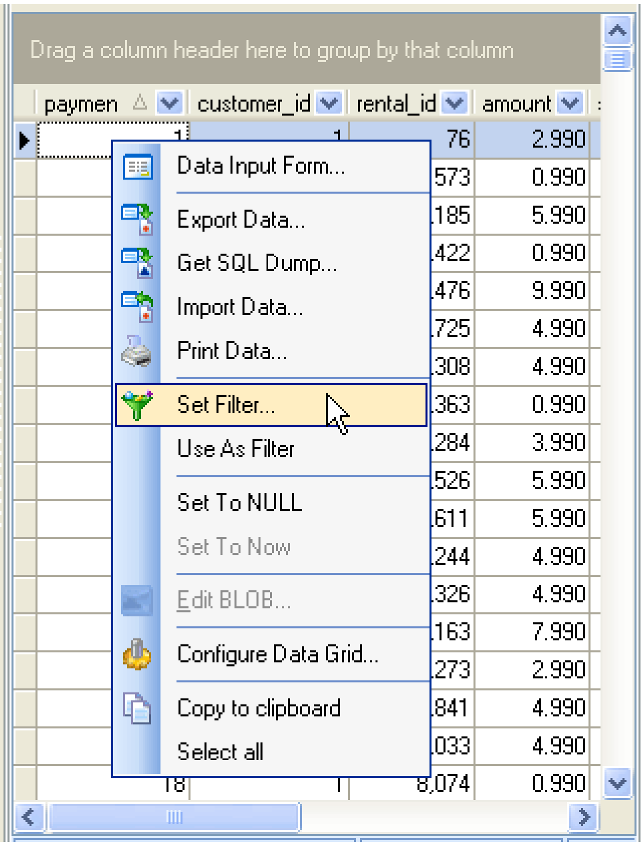
Use the drop-down button in the column caption area to filter records by the selected column value(s) or enter the filter condition directly in the filter row 445.

Filter by two operators

Invoke simple filter dialog using the Custom item of the column caption area drop-down list. Select a logical operator for checking the column values (like "is less than", "is greater than", etc.) and set the value to be checked by this operator in the next box; then set the second condition if necessary in the following way and set the relation between these two conditions, whether both of them should be matched or just one of them; use the '_' character to represent any single symbol in the condition and the '%' character to represent any series of symbols in the condition.

Filter by any custom criteria

To filter data according to more difficult custom conditions, use the Filter Builder dialog. To invoke the dialog, use the Set Filter link of popup menu or click the Customize button on the Filter panel. This panel is visible if any filtering is already applied to the grid (you can use column header menu or grid menu for quick filtering).



The dialog also allows to save filter criteria to an external file for future use.

After you set a filter, the filtering panel becomes visible at the top/bottom of the grid where you can see the active filtering condition and easily enable or disable it by clicking the check box on the left. To customize the filtering process, use filter options 445.

The Copy current filter as SQL condition to clipboard feature is useful in case the same compound filter is applied several times. Just once apply the filter, copy to clipboard as SQL condition, paste to SQL Editor 313 and save as a query. You can also use Generate query link on the Navigation bar.

See also: Data View 328, Table Editor 79, SQL Editor 313, Visual Query Builder 318

8.2 BLOB Editor

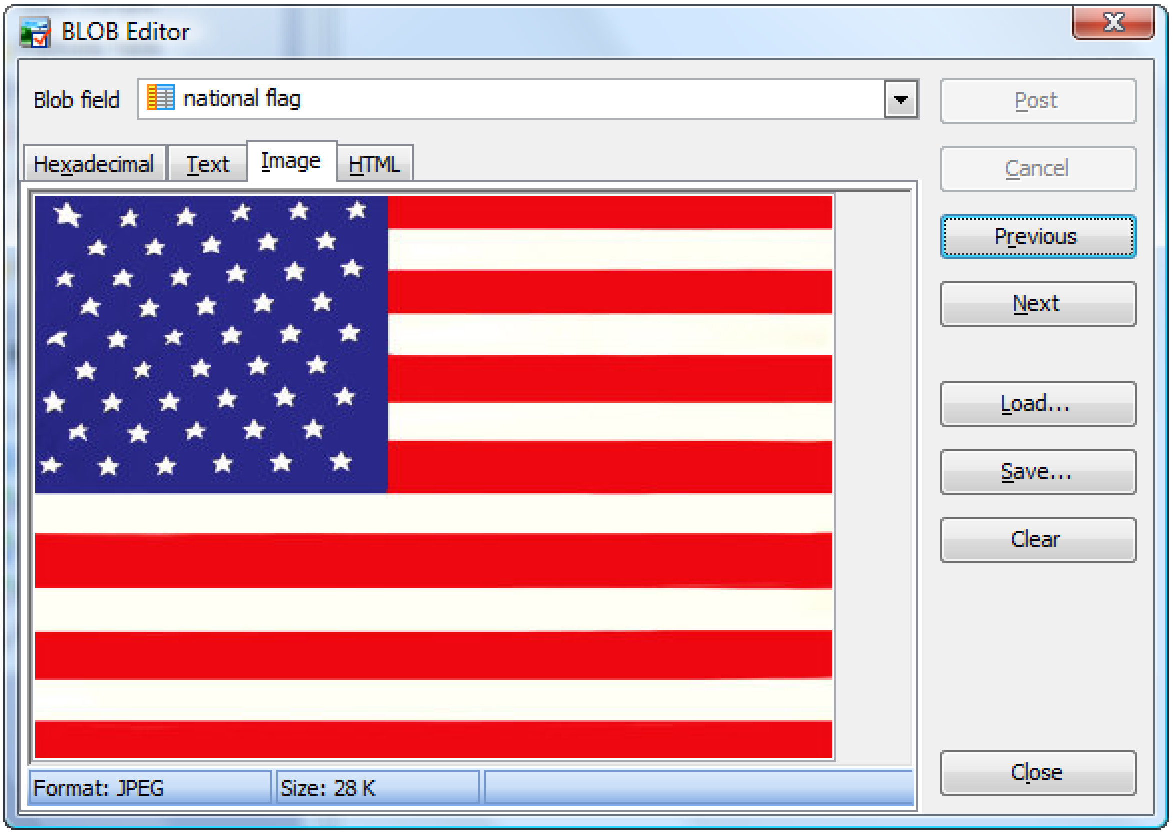
BLOB Editor is a tool to view and edit BLOB data in the following ways: hexadecimal dump 340, plain text 340, graphical image 339, HTML page 341, or PDF document 342. BLOB Editor is invoked from data grid 328 of any table editor 79 or the result tab of SQL Editor 313 and Visual Query Builder 318 by double clicking of the BLOB field to be edited or with the Edit BLOB link of the field's popup menu. The editor also can be called from BLOB Viewer 379 with the Edit current BLOB button.

With BLOB Editor you can work with all BLOB columns of the grid. To switch between columns, select the necessary one from the BLOB field list.

BLOB Editor allows you to navigate between the grid records using the Previous and Next buttons. You can load the new BLOB content and save or clear it using corresponding buttons. After changes are made, click the Post button to apply the changes or the Cancel button to discard them.

8.2.1 Editing as image

The Image panel of BLOB Editor displays field data as graphical image. Use the Save and Load buttons to save the image to a file or load an image from a file. A graphical representation of BLOB data supports five image formats: BMP, Windows metafile, JPEG, GIF and PNG.



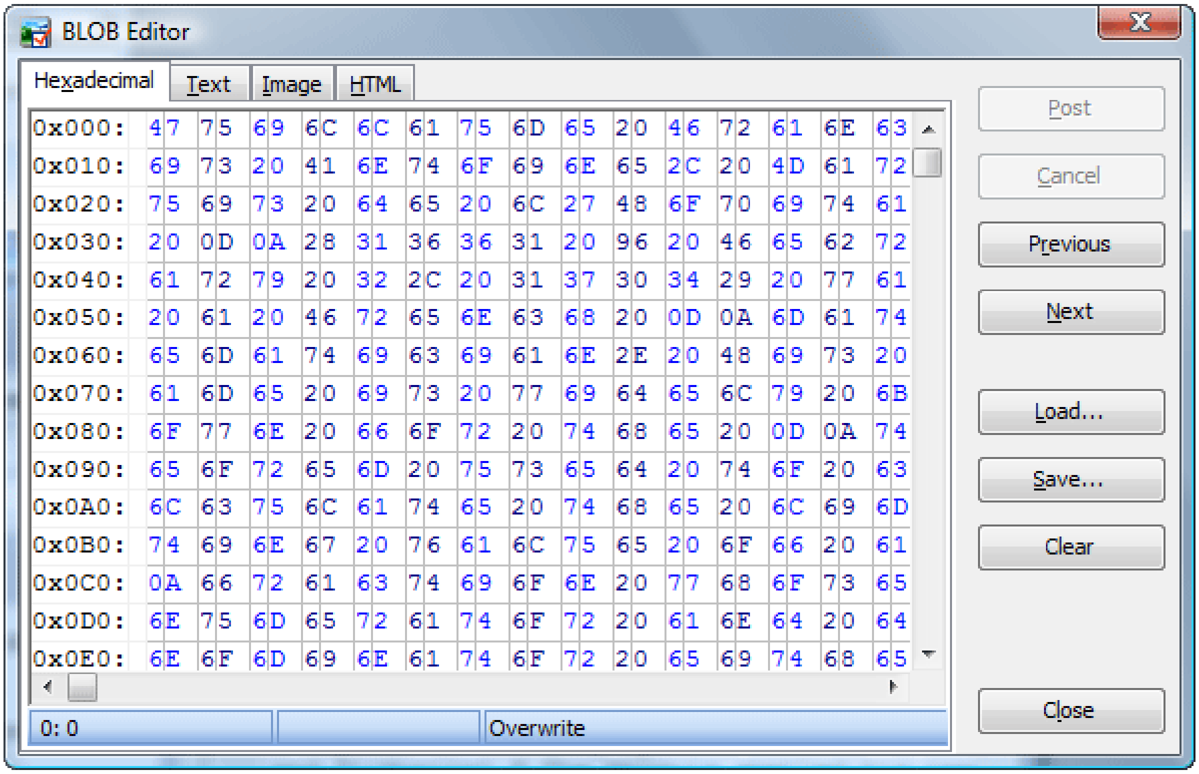
Blob field: national flag

Format: JPEG

Size: 28 K

8.2.2 Editing as hexadecimal dump

The Hexadecimal panel allows you to edit data in hexadecimal mode. To load/save a hexadecimal dump from/to a file, use the corresponding buttons. Use the Insert key to switch between Insert and Overwrite modes.

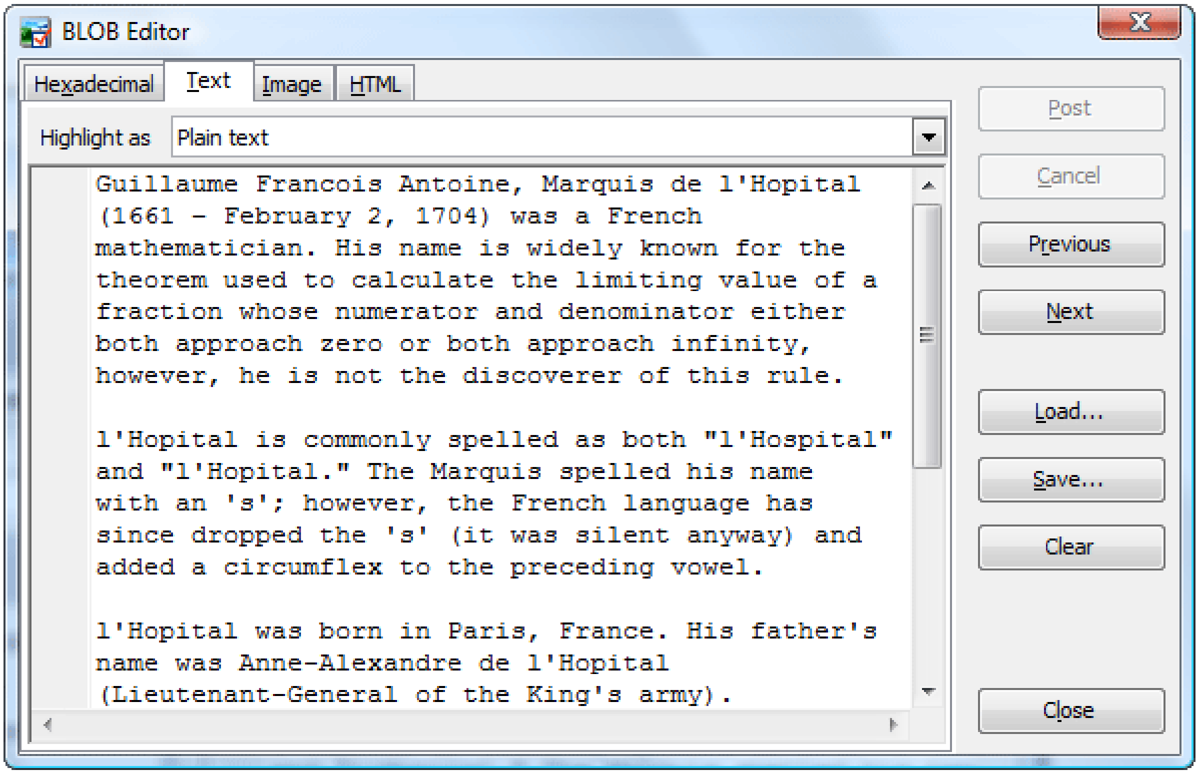


```
0x000:  47 75 69 6C 6C 61 75 6D 65 20 46 72 61 6E 63
0x010:  69 73 20 41 6E 74 6F 69 6E 65 2C 20 4D 61 72
0x020:  75 69 73 20 64 65 20 6C 27 48 6F 70 69 74 61
0x030:  20 0D 0A 28 31 36 36 31 20 96 20 46 65 62 72
0x040:  61 72 79 20 32 2C 20 31 37 30 34 29 20 77 61
0x050:  20 61 20 46 72 65 6E 63 68 20 0D 0A 6D 61 74
0x060:  65 6D 61 74 69 63 69 61 6E 2E 20 48 69 73 20
0x070:  61 6D 65 20 69 73 20 77 69 64 65 6C 79 20 6B
0x080:  6F 77 6E 20 66 6F 72 20 74 68 65 20 0D 0A 74
0x090:  65 6F 72 65 6D 20 75 73 65 64 20 74 6F 20 63
0x0A0:  6C 63 75 6C 61 74 65 20 74 68 65 20 6C 69 6D
0x0B0:  74 69 6E 67 20 76 61 6C 75 65 20 6F 66 20 61
0x0C0:  0A 66 72 61 63 74 69 6F 6E 20 77 68 6F 73 65
0x0D0:  6E 75 6D 65 72 61 74 6F 72 20 61 6E 64 20 64
0x0E0:  6E 6F 6D 69 6E 61 74 6F 72 20 65 69 74 68 65
```

0: 0 | | Overwrite

8.2.3 Editing as plain text

The Text panel allows you to edit data as a simple text. Several types of text highlighting are available (*Plain text, SQL, XML, Java, VBScript, JScript, Cmd batch, PHP, CSS, UnixShell Script, INI,* and *HTML*). The popup menu of the panel allows you to invoke Find Text, Replace Text and Go to line dialogs.

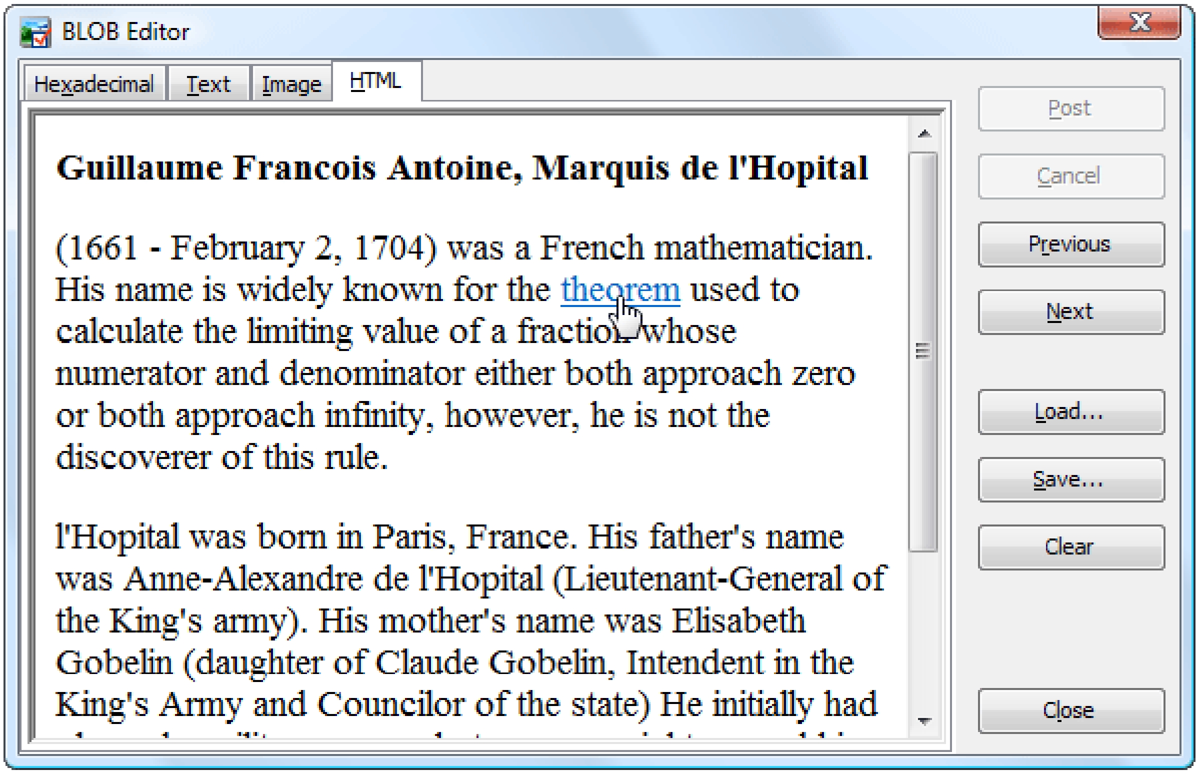


Highlight as: Plain text

```
Guillaume Francois Antoine, Marquis de l'Hopital
(1661 - February 2, 1704) was a French
mathematician. His name is widely known for the
theorem used to calculate the limiting value of a
fraction whose numerator and denominator either
both approach zero or both approach infinity,
however, he is not the discoverer of this rule.

l'Hopital is commonly spelled as both "l'Hospital"
and "l'Hopital." The Marquis spelled his name
with an 's'; however, the French language has
since dropped the 's' (it was silent anyway) and
added a circumflex to the preceding vowel.

l'Hopital was born in Paris, France. His father's
name was Anne-Alexandre de l'Hopital
(Lieutenant-General of the King's army).
```

8.2.4 Editing as HTML

The HTML panel presents field data as HTML. You can load a new content of the current field from a .html file or type it manually within the Text tab of the editor.



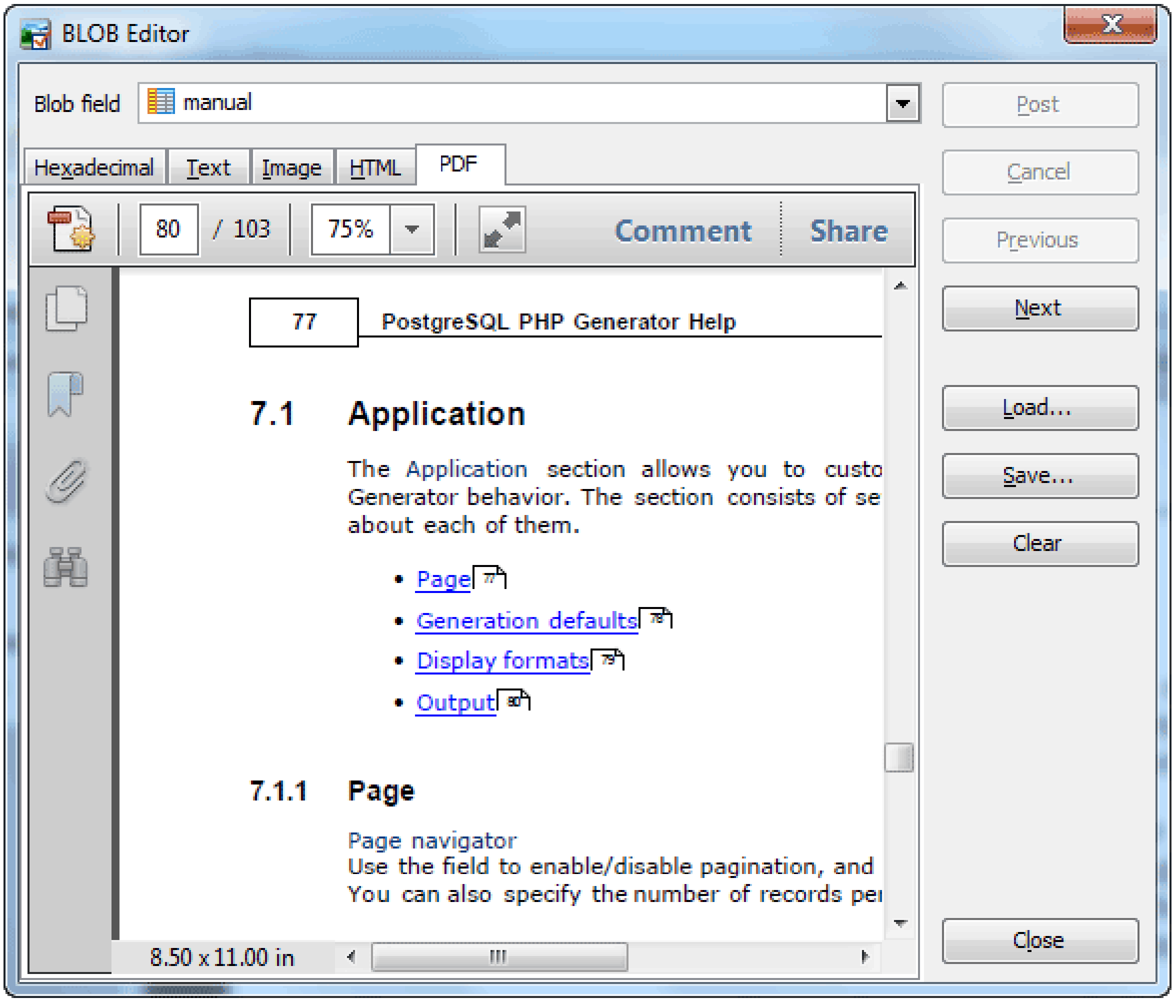
Guillaume Francois Antoine, Marquis de l'Hopital

(1661 - February 2, 1704) was a French mathematician. His name is widely known for the theorem used to calculate the limiting value of a fraction whose numerator and denominator either both approach zero or both approach infinity, however, he is not the discoverer of this rule.

l'Hopital was born in Paris, France. His father's name was Anne-Alexandre de l'Hopital (Lieutenant-General of the King's army). His mother's name was Elisabeth Gobelin (daughter of Claude Gobelin, Intendent in the King's Army and Councilor of the state) He initially had

8.2.5 Editing as PDF document

The PDF panel presents field data as PDF document. To accomplish common operations with data, use the Adobe Reader toolbar.



Blob field: manual

77 PostgreSQL PHP Generator Help

7.1 Application

The Application section allows you to custo
Generator behavior. The section consists of se
about each of them.

- Page 77
- Generation defaults 78
- Display formats 79
- Output 80

7.1.1 Page

Page navigator
Use the field to enable/disable pagination, and
You can also specify the number of records pe

8.3 Export Data Wizard

Data Export wizard is a tool to save data from Microsoft SQL tables, views, and queries to the most popular formats. It allows you to fully customize output files including header and footer, fonts, colors, and data formats.

Export Data tool supports:

- Microsoft Office Excel 97-2003, 2007
- CSV
- HTML
- XML
- Text
- Microsoft Office Word 97-2003, 2007
- Microsoft Office Access
- OpenDocument Spreadsheet
- OpenDocument Text
- DBF
- PDF
- RTF
- DIF
- SYLK
- LaTex.

In order to run the wizard you should either

- open the table in Table Editor;
- go on to the Data tab

or

- open and execute the query in SQL Editor or Query Builder;
- proceed to the Result tab

and select the Export Data item from the Navigation Bar.

To export your data,

- Set the format and the name 344 of the destination file;
- Specify such additional options of the result file as header and footer 345, formats applied to exported data 346 and some format-specific options 347;
- Select columns 346 you want to include into result files;
- Specify other export options 350.

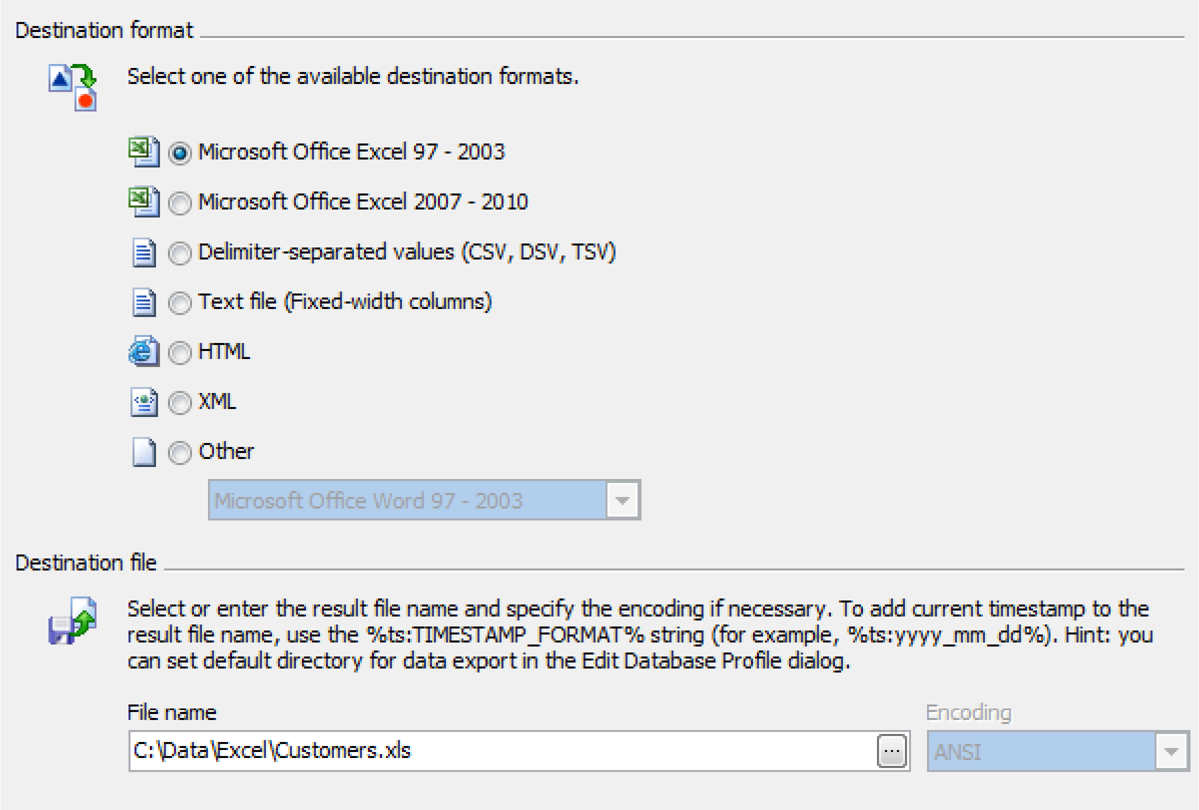
See also: Get SQL Dump, Import Data Wizard 354

8.3.1 Setting destination file name and format

Select one of the available destination formats and set the name for the result file. The file name extension in the Destination file name box varies according to the selected export type.

The file name may contain current timestamp with the %ts:TIMESTAMP_FORMAT% string. Examples of valid log file names:

dbname_export_%ts:yyyy_mm_dd%.log
export_%ts:yyyy_mm_dd_hh_mm%.log
%ts:yyyy_mm_dd_hh_mm_ss%.log



Destination format

Select one of the available destination formats.

● Microsoft Office Excel 97 - 2003
○ Microsoft Office Excel 2007 - 2010
○ Delimiter-separated values (CSV, DSV, TSV)
○ Text file (Fixed-width columns)
○ HTML
○ XML
○ Other

Microsoft Office Word 97 - 2003

Destination file

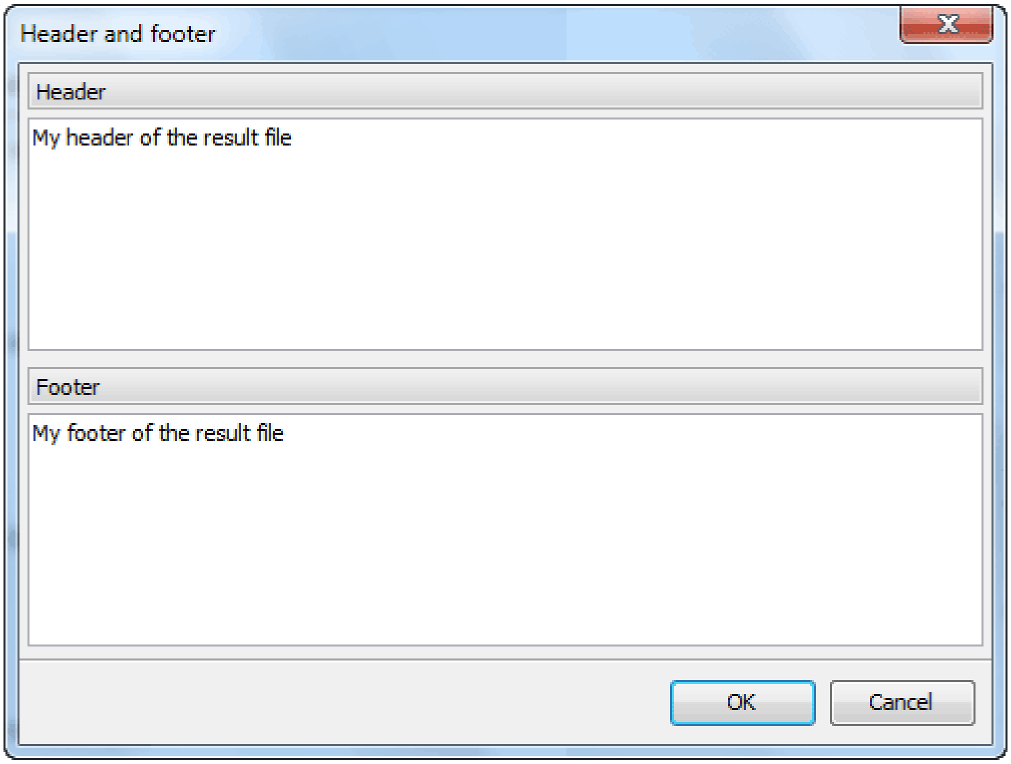
Select or enter the result file name and specify the encoding if necessary. To add current timestamp to the result file name, use the %ts:TIMESTAMP_FORMAT% string (for example, %ts:yyyy_mm_dd%). Hint: you can set default directory for data export in the Edit Database Profile dialog.

File name: C:\Data\Excel\Customers.xls

Encoding: ANSI

8.3.2 Setting header and footer

To specify the result file's header and footer, double click the corresponding button and complete fields of the Header and Footer window.

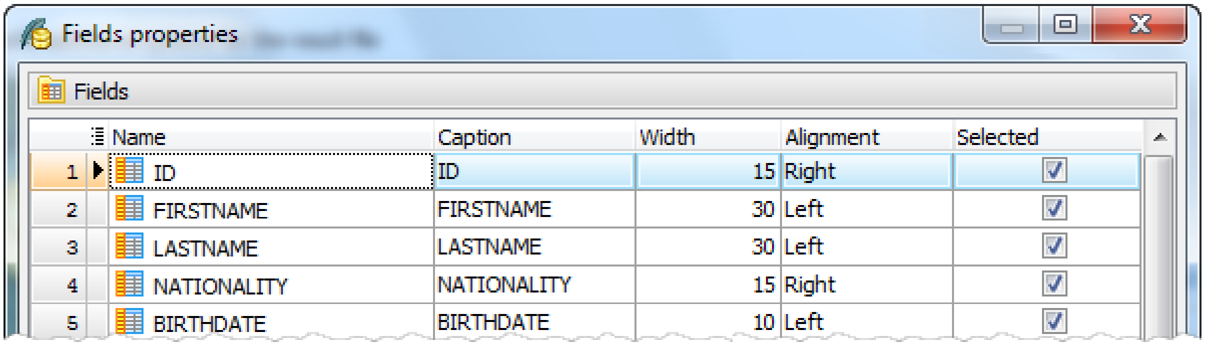


My header of the result file

My footer of the result file

8.3.3 Selecting fields for export

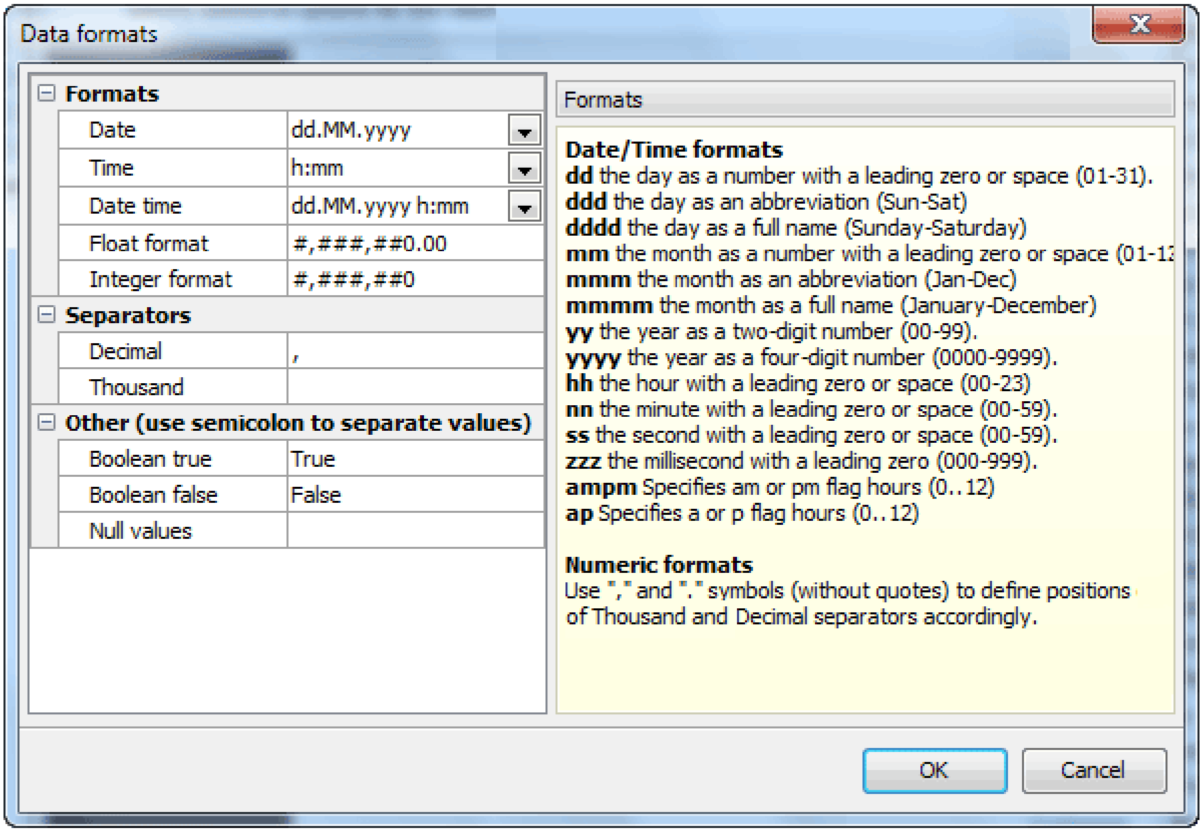
Uncheck the Selected box to exclude the corresponding field from the export, specify a Caption to be used for the result column, and also width, and alignment for output columns (when applicable).



	Name	Caption	Width	Alignment	Selected
1	ID	ID	15	Right	☑
2	FIRSTNAME	FIRSTNAME	30	Left	☑
3	LASTNAME	LASTNAME	30	Left	☑
4	NATIONALITY	NATIONALITY	15	Right	☑
5	BIRTHDATE	BIRTHDATE	10	Left	☑

8.3.4 Adjusting data formats

This step allows you to customize formats applied to exported data. Edit the format masks to adjust the result format in the way you need.



Formats

Date	dd.MM.yyyy
Time	h:mm
Date time	dd.MM.yyyy h:mm
Float format	#,###,##0.00
Integer format	#,###,##0

Separators

Decimal	,
Thousand	

Other (use semicolon to separate values)

Boolean true	True
Boolean false	False
Null values	

Formats

Date/Time formats
dd the day as a number with a leading zero or space (01-31).
ddd the day as an abbreviation (Sun-Sat)
dddd the day as a full name (Sunday-Saturday)
mm the month as a number with a leading zero or space (01-1
mmm the month as an abbreviation (Jan-Dec)
mmmm the month as a full name (January-December)
yy the year as a two-digit number (00-99).
yyyy the year as a four-digit number (0000-9999).
hh the hour with a leading zero or space (00-23)
nn the minute with a leading zero or space (00-59).
ss the second with a leading zero or space (00-59).
zzz the millisecond with a leading zero (000-999).
ampm Specifies am or pm flag hours (0..12)
ap Specifies a or p flag hours (0..12)

Numeric formats
Use "," and "." symbols (without quotes) to define positions of Thousand and Decimal separators accordingly.

8.3.5 Setting format-specific options

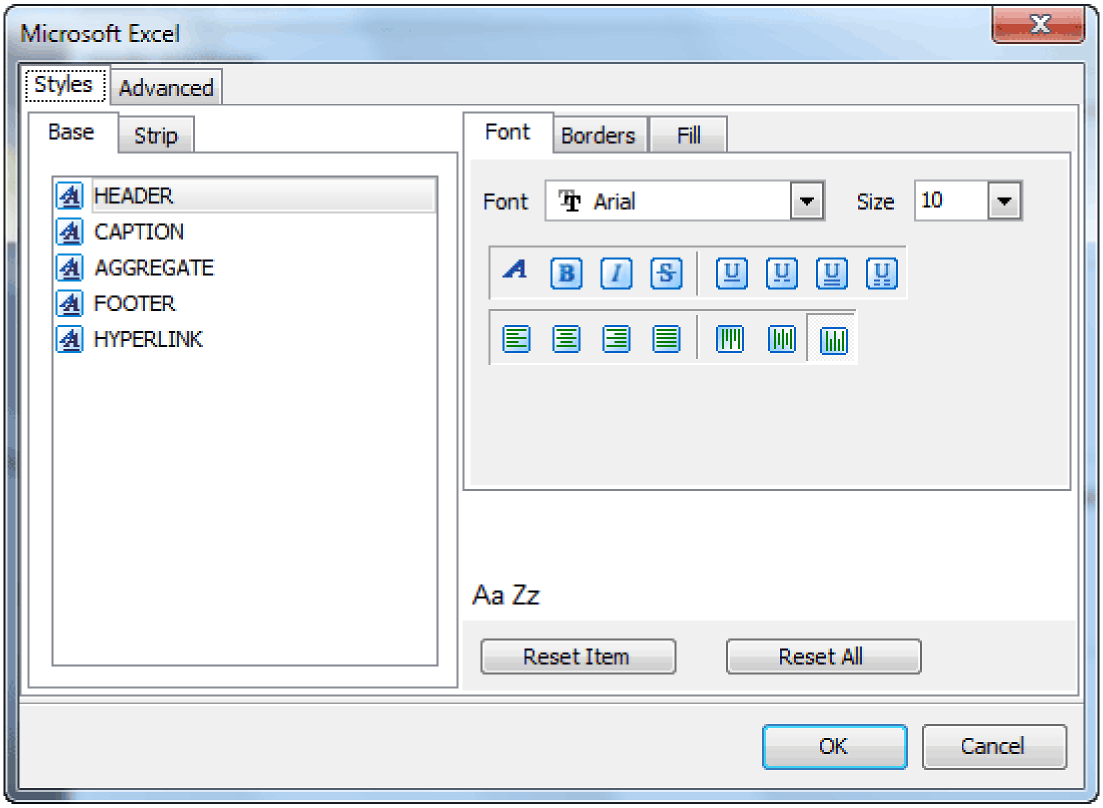
Each format supposes corresponding additional export options. Use the wizard option to adjust export properties depending on the target file format you have selected earlier. The following formats are at your disposal: Microsoft Excel 347, Microsoft Excel 2007, CSV 349, Text 349, HTML 348, XML 349, Microsoft Word, Microsoft Word 2007, Microsoft Access, OpenDocument Spreadsheet, OpenDocument Text, DBF, PDF, RTF, DIF, SYLK, and LaTeX.

Microsoft Excel

The Data Format tab contains general options, which allow you to adjust the format for each kind of Excel cells. This means that you can specify such parameters as font, borders, filling color and method, etc. for each entity (such as data field, header, footer, caption, data, hyperlink and so on) separately. Also it is possible to create styles to make target Excel file be striped by columns or rows (the Styles tab).

The Extensions tab provides a possibility to add hyperlinks and notes to any cell of target file. Click the Plus button to add a new hyperlink or note to target Excel sheet and adjust its parameters. Click the Minus button to delete added hyperlink or note.

The Advanced tab allows you to define page header, page footer and title for target Excel sheet.



Font: Arial
Size: 10

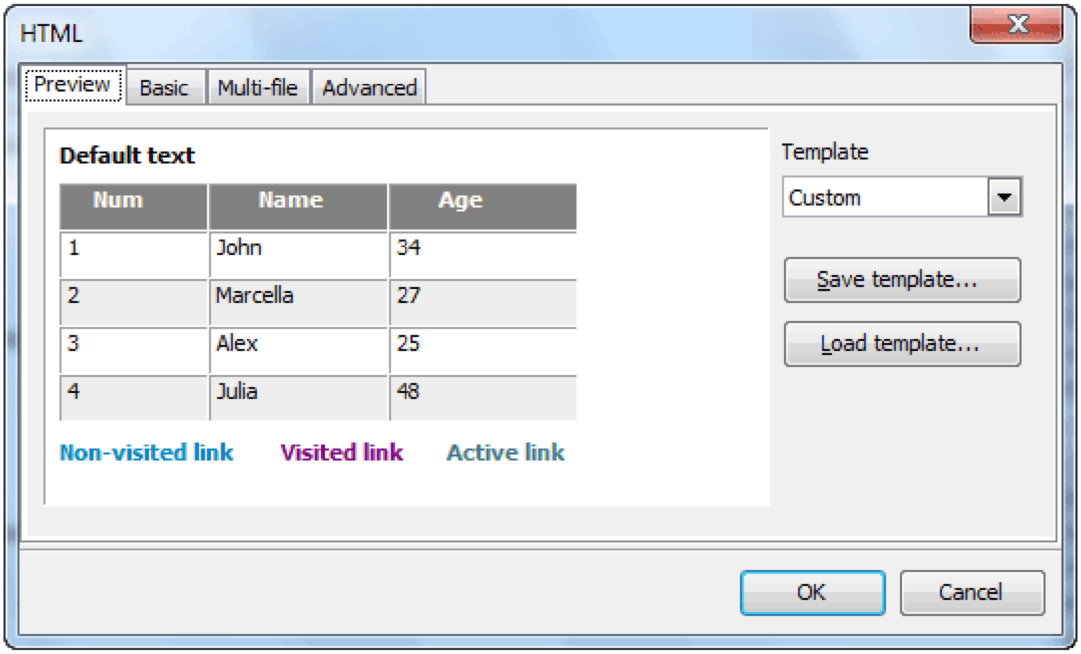
HTML

The Preview tab allows you to select the style of HTML file from a number of built-in templates provided by the Templates combo box. You can choose any of these templates, customize it by clicking on objects in the preview panel, and save it as a custom template using the Save template button. Use the Load template button to load previously saved custom templates from hard disk.

The Basic tab allows you to specify basic parameters of target HTML file, such as its title, cascade style sheet options, etc.

The Multi-file tab provides you with a possibility to split target HTML file into several separated files. This tab allows you to specify the record count for a single file, set an option to generate an index HTML file, and add an ability of navigation between each other to each of exported files.

The Advanced tab contains such HTML options as default font, background, cell padding and spacing, etc.



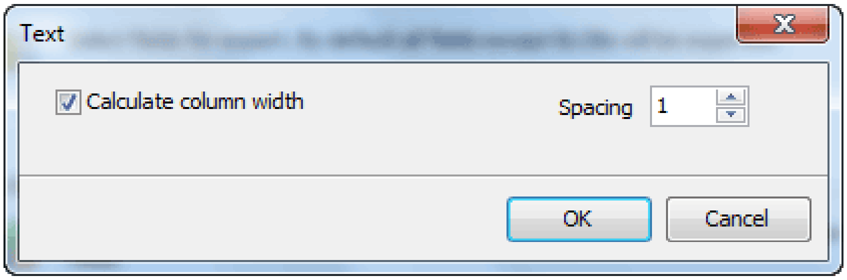
Template: Custom

Default text

Num	Name	Age
1	John	34
2	Marcella	27
3	Alex	25
4	Julia	48

Non-visited link Visited link Active link

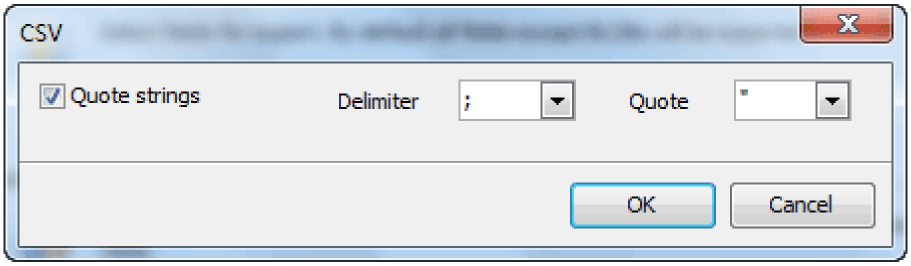
Text files
Set the Calculate column width options on if you want each column of target file to be adjusted to the maximum number of characters in it. The Spacing option specifies the number of spaces between columns in the target file.



☑ Calculate column width

Spacing: 1

CSV files
You can specify column separator and optional values quote character for the target file on this step.

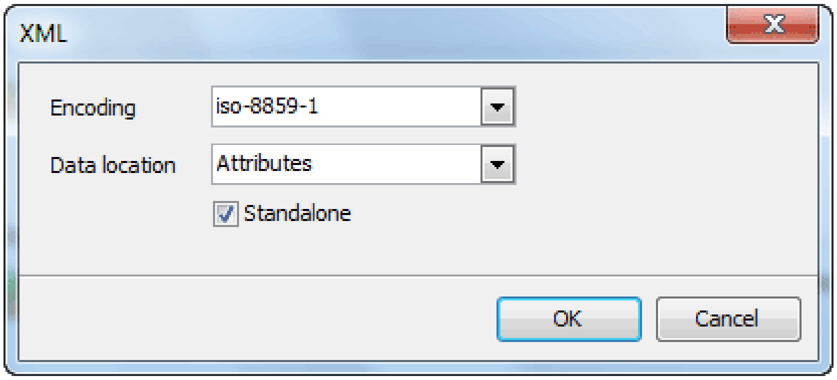


☑ Quote strings

Delimiter: ;

Quote: "

XML documents
Specify XML document encoding in the Encoding edit box and set the Standalone option on if you wish the target document to be standalone.



Encoding: iso-8859-1

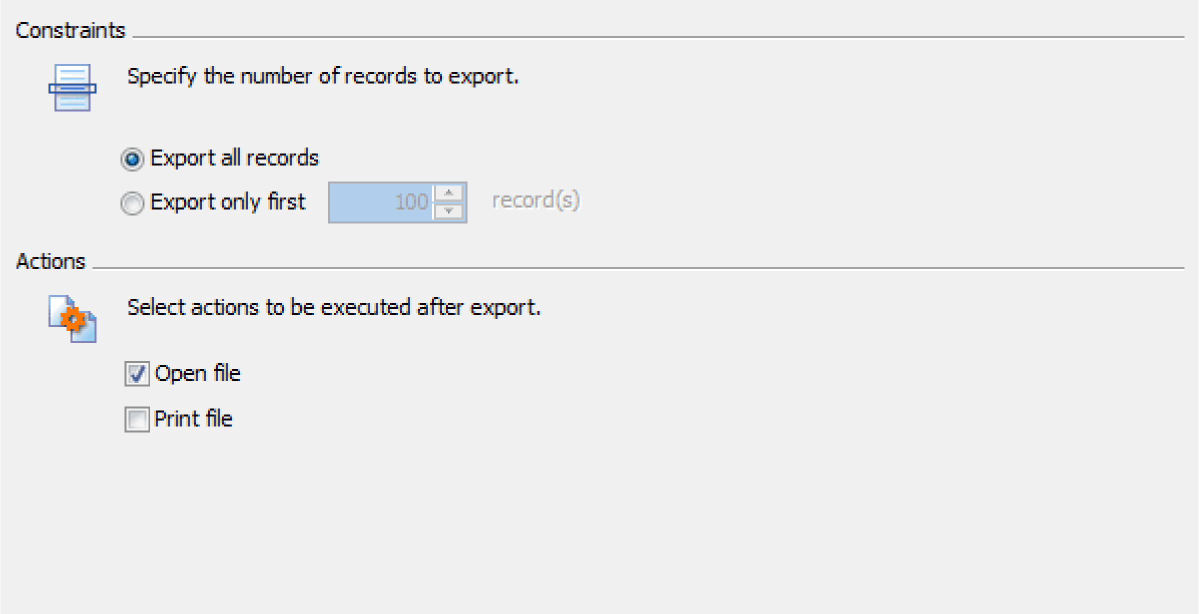
Data location: Attributes

☑ Standalone

8.3.6 Setting common export options

Use this step to specify options to be applied to all exported data:

- Select the number of records to be exported from each table: a fixed number or all records.
- Specify actions to be executed after the export. To open the result files in the associated program (MS Excel, Notepad, default browser, etc), check the Open file box. To send the result files to the default printer, use the Print file checkbox.


Constraints

Specify the number of records to export.

● Export all records

○ Export only first 100 record(s)

Actions

Select actions to be executed after export.

☑ Open file

☐ Print file

8.4 Get SQL Dump

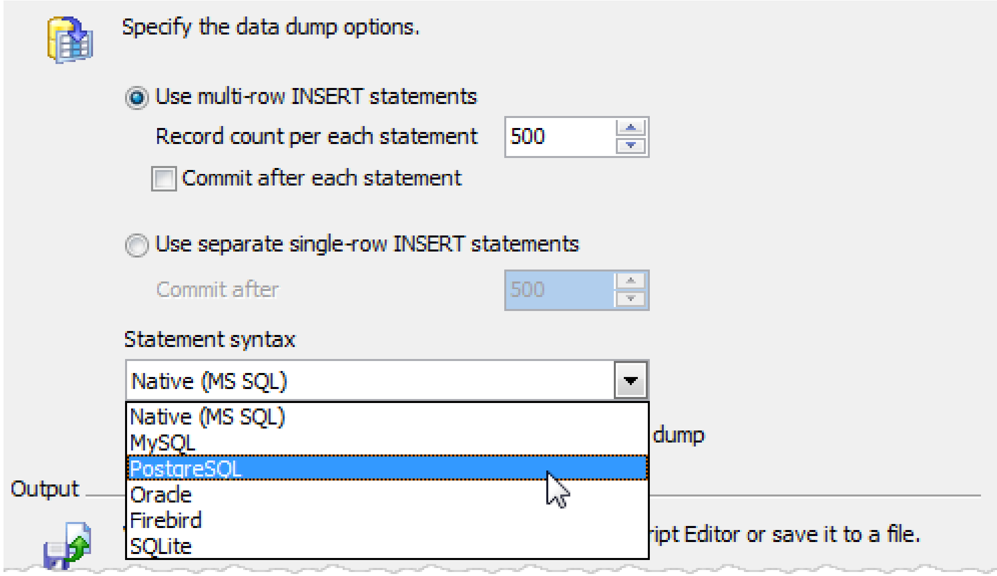
Get SQL Dump Wizard allows you to export data from a table or a query result to the SQL script as a number of INSERT statements.

In order to get a SQL dump from a table or a query:

- open the table in Table Editor or open and execute query in SQL Editor or Query Builder;
- open the Data tab or the Result tab respectively;
- use the Get SQL Dump item of the Navigation Bar.

- Selecting fields to include in the result INSERT statement 351
- Specifying dump options 352

See also: Export Data Wizard 344, SQL Script Editor 364



Specify the data dump options.

● Use multi-row INSERT statements

Record count per each statement 500

☐ Commit after each statement

○ Use separate single-row INSERT statements

Commit after 500

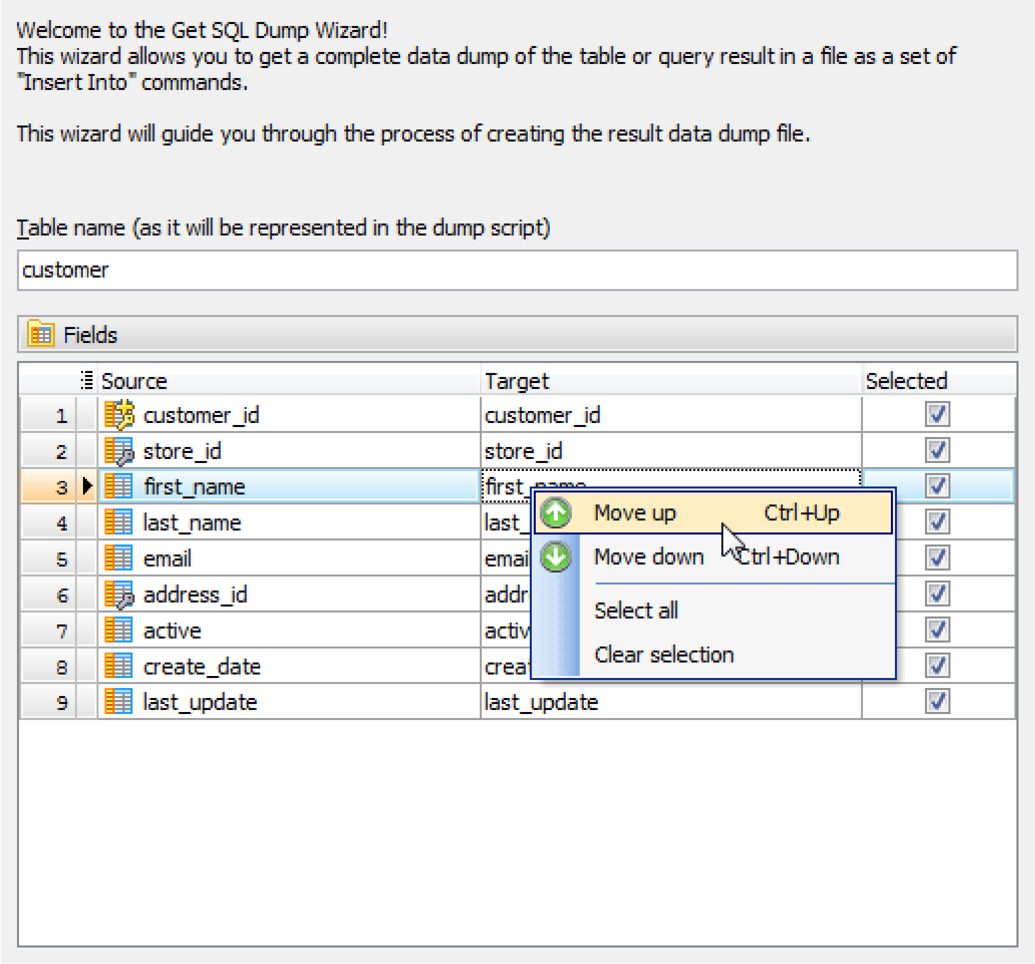
Statement syntax

Native (MS SQL)

8.4.1 Selecting fields

The first wizard step allows you to specify the table name as it will be included in the result script.

You can also select the fields to be included in the result *INSERT* statement. All the table fields are included into the Selected fields list by default. If you do not want some fields to be exported, move them back to the Available fields list. *Text*, *GUID*, *Date*, *Time*, and *DateTime* columns are included in the result INSERT statements according to the Storage Options of the Database Profile 28.



Welcome to the Get SQL Dump Wizard!
This wizard allows you to get a complete data dump of the table or query result in a file as a set of "Insert Into" commands.

This wizard will guide you through the process of creating the result data dump file.

Table name (as it will be represented in the dump script)

customer

Fields

	Source	Target	Selected
1	customer_id	customer_id	☑
2	store_id	store_id	☑
3	first_name	first_	☑
4	last_name	last_	☑
5	email	emai	☑
6	address_id	addr	☑
7	active	activ	☑
8	create_date	creat	☑
9	last_update	last_update	☑

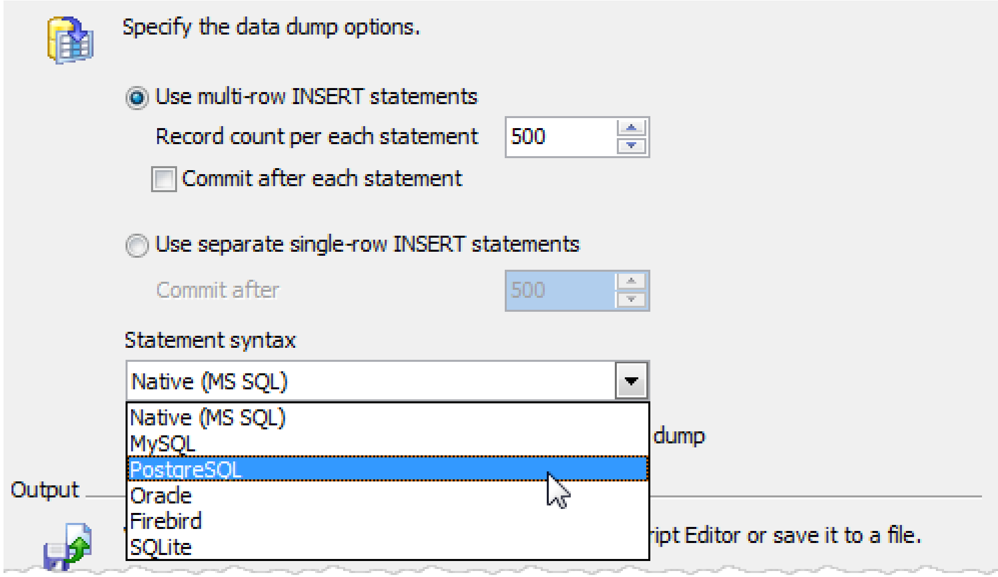
8.4.2 Specifying dump options

Select the data dump mode to be used (Multi-row INSERT statements or separate single-row INSERT statements) and specify commits' frequency.

To add the "CREATE TABLE" to the top of the dump, check the corresponding box.

Get SQL Dump Wizard allows you to send the result script to SQL Script Editor 364 or to save it to a specified file. Select the Send to script editor option to load the result to the editor. To save the result to the file, enter the script file name (*.*sql*).

Click the Ready button to start the process.



Specify the data dump options.

● Use multi-row INSERT statements

Record count per each statement: 500

☐ Commit after each statement

○ Use separate single-row INSERT statements

Commit after: 500

Statement syntax: Native (MS SQL)

8.5 Import Data Wizard

Import Data Wizard provides you with a graphical user interface to import data from the most popular files formats into existing Microsoft SQL tables. It allows you to adjust data formats, empty target tables, execute custom SQL scripts, etc.

Import Data tool supports:

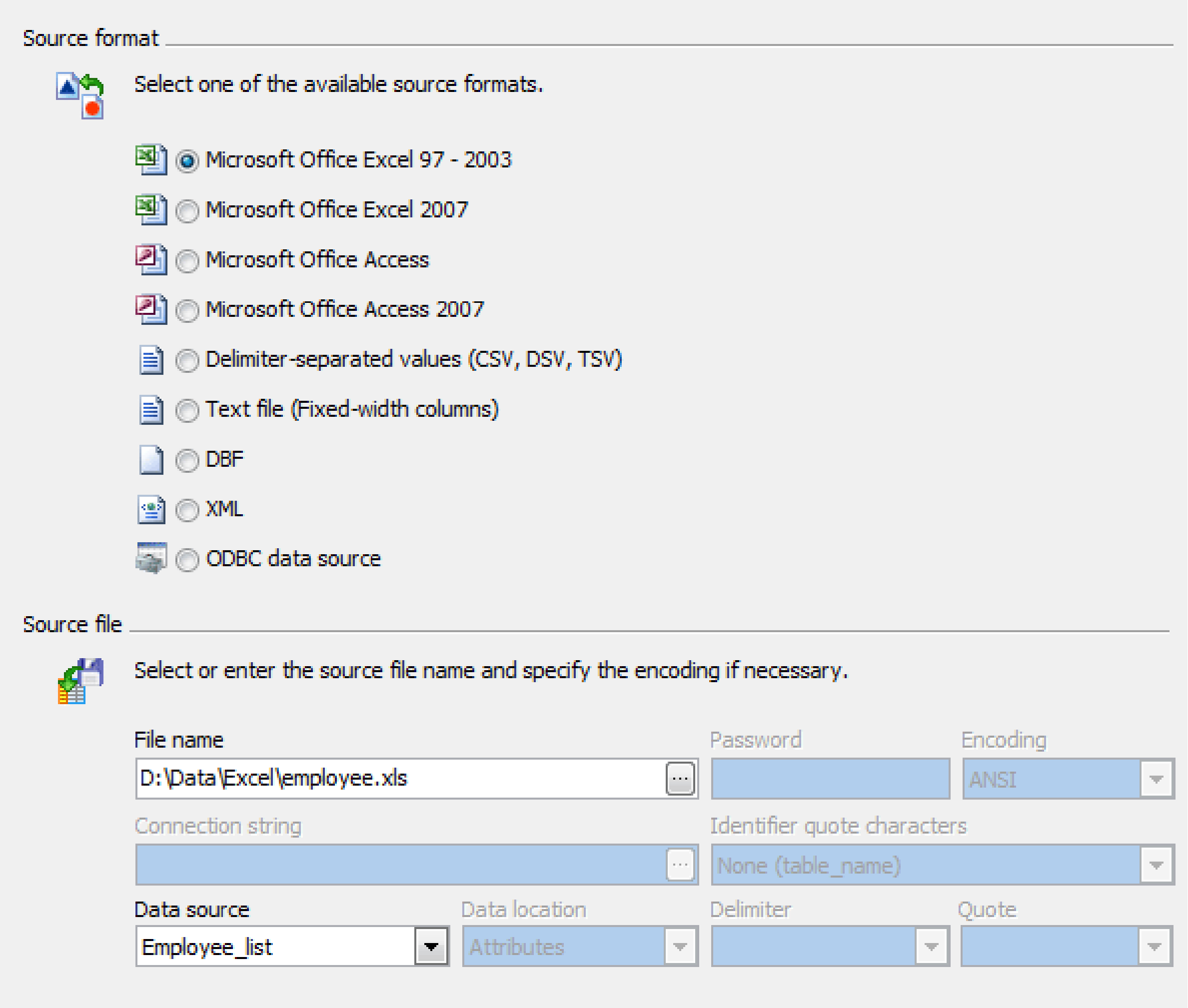
- Microsoft Office Excel 95-2003
- Microsoft Office Excel 2007
- Microsoft Office Access
- Microsoft Office Access 2007
- Delimiter-separated values (CSV, DSV, TSV)
- DBF
- Text files
- XML
- ODBC data sources (any database accessible via an ODBC driver or OLE DB provider, such as SQL Server, MySQL, Oracle, MS Access, Sybase, DB2, PostgreSQL, etc.)

In order to run the wizard you should

- open the table in Table Editor;
- go on to the Data tab;
- select the Import Data item from the Navigation Bar.

To import data,

- Set the format 355 of the input data and the source file name;
- Map source file columns and target table fields 357;
- Specify other import options 360.



Source format

Select one of the available source formats.

- ● Microsoft Office Excel 97 - 2003
- ○ Microsoft Office Excel 2007
- ○ Microsoft Office Access
- ○ Microsoft Office Access 2007
- ○ Delimiter-separated values (CSV, DSV, TSV)
- ○ Text file (Fixed-width columns)
- ○ DBF
- ○ XML
- ○ ODBC data source

Source file

Select or enter the source file name and specify the encoding if necessary.

File name: D:\Data\Excel\employee.xls

Password:

Encoding: ANSI

Connection string:

Identifier quote characters: None (table_name)

Data source: Employee_list

Data location: Attributes

Delimiter:

Quote:

See also: Export Data Wizard 344

8.5.1 Setting source file name and format

1. Select the format of the source file.

2. Specify the file you want to import. The file name extension in the File name box varies according to the selected import type. The wizard allows you to import data from several files at a time.

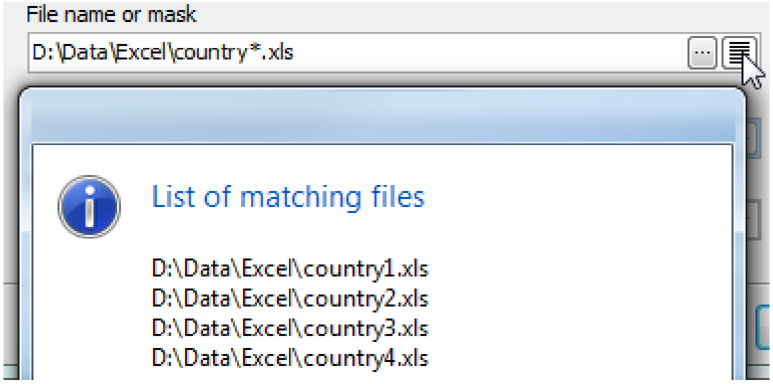
To import data from multiple files with the same structure, set the mask of the file names to the corresponding field. To see the list of matching files, use with the button on the right.

Example 1:

Suppose, you need to import data from the following tables:

D:\Data\Excel\country1.xls
D:\Data\Excel\country2.xls
D:\Data\Excel\country3.xls
D:\Data\Excel\country4.xls

The mask for these file names is *D:\Data\Excel\country*.xls*.



File name or mask

D:\Data\Excel\country*.xls

List of matching files

D:\Data\Excel\country1.xls
D:\Data\Excel\country2.xls
D:\Data\Excel\country3.xls
D:\Data\Excel\country4.xls

3. For ODBC data sources specify the connection string to be used to connect to the data source.

4. Select the data source to import: a table of MS Access database or a spreadsheet of MS Excel.

5. Enter the password to the database (MS Access).

6. For CSV file set the delimiter and quote characters.

7. Select source file Encoding.

8. For .XML files, define the XPath to the data to be imported to the selected table and select whether data is stored in Attributes or in Subnodes.

Example 2:
To import data from the following .xml file, use XPath=*/Employees/Employee* and Data location=*Subnodes*

```
<?xml version="1.0" encoding="utf-8"?>
<Employees>
  <Employee>
    <ID>1</ID>
    <FirstName>Klaus</FirstName>
    <LastName>Salchner</LastName>
    <PhoneNumber>410-727-5112</PhoneNumber>
  </Employee>
  <Employee>
     <ID>2</ID>
    <FirstName>Peter</FirstName>
    <LastName>Pan</LastName>
    <PhoneNumber>604-111-1111</PhoneNumber>
  </Employee>
</Employees>
```

Example 3:
To import data from the .xml file below, use XPath=*DATAPACKET/Data/Item* and Data location=*Attributes*

```
<?xml version="1.0"?>
```

```
<DATAPACKET Version="2.0">
<Data>
    <Item ID="1" FirstName="Klaus" LastName="Salchner" PhoneNumber="410-727-
5112" />
   <Item ID="2" FirstName="Peter" LastName="Pan" PhoneNumber="604-111-1111" />

</Data>
</DATAPACKET>
```

8.5.2 Setting the accordance between source and target columns

The wizard provides you with several ways to map input data to the target table columns.

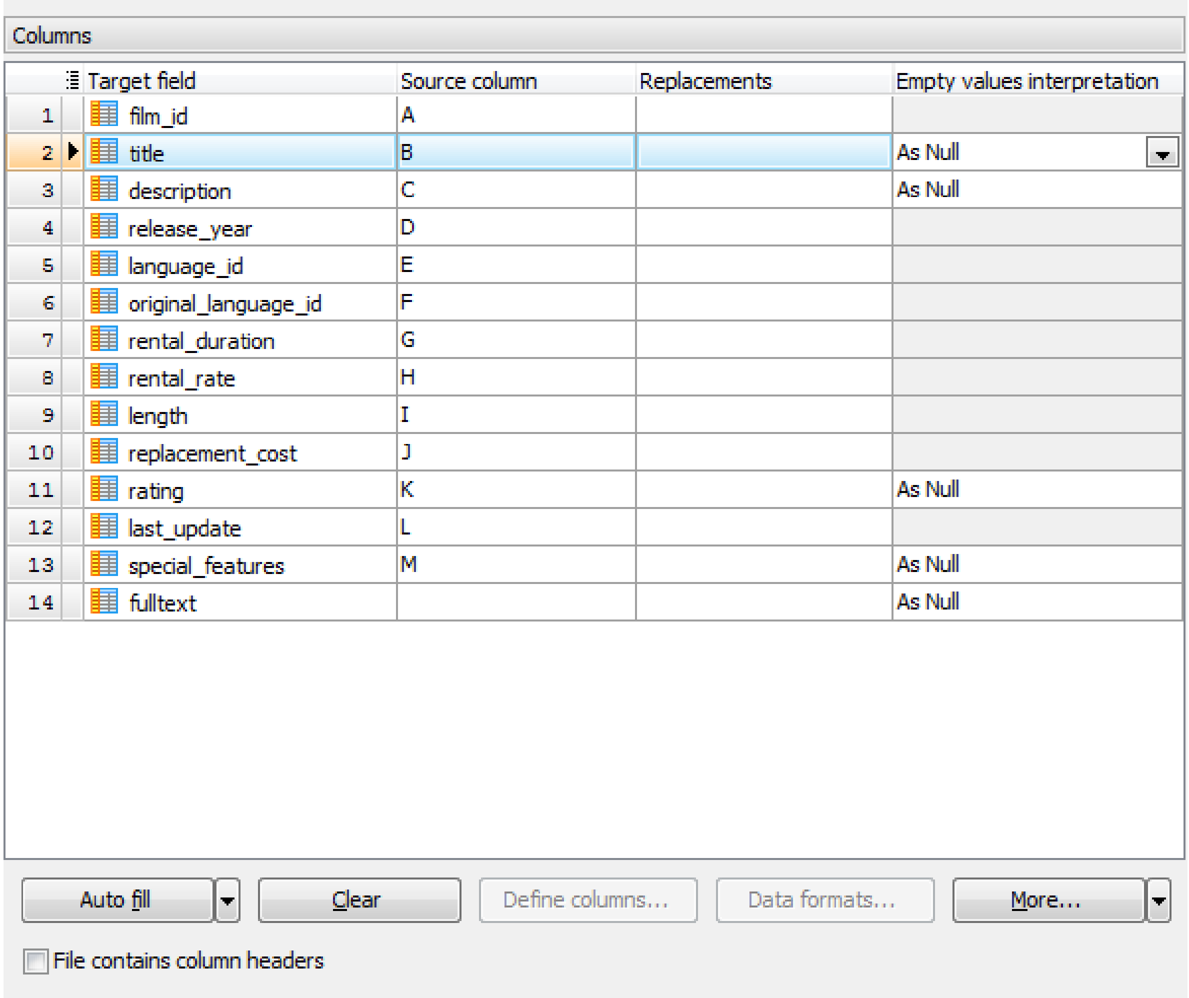
- You can map columns automatically by order with the Auto Fill and Auto fill all maps buttons.
- You can do it manually using the drop-down list of Source column fields.
- To map columns visually, open Map builder 358 with the Build map link.

It's useful to save a specified map to a file for further using it in the next wizard sessions. To save a map, use the More... button and follow the Save map link.

To see the 100 first rows of input file or output table, use the More... button and follow the View source data or Preview results links respectively.

You can also specify Replacements to be applied to the selected column before the import and data format masks 359 used for the input file.

To exclude the first file row, use the File contains column header checkbox.

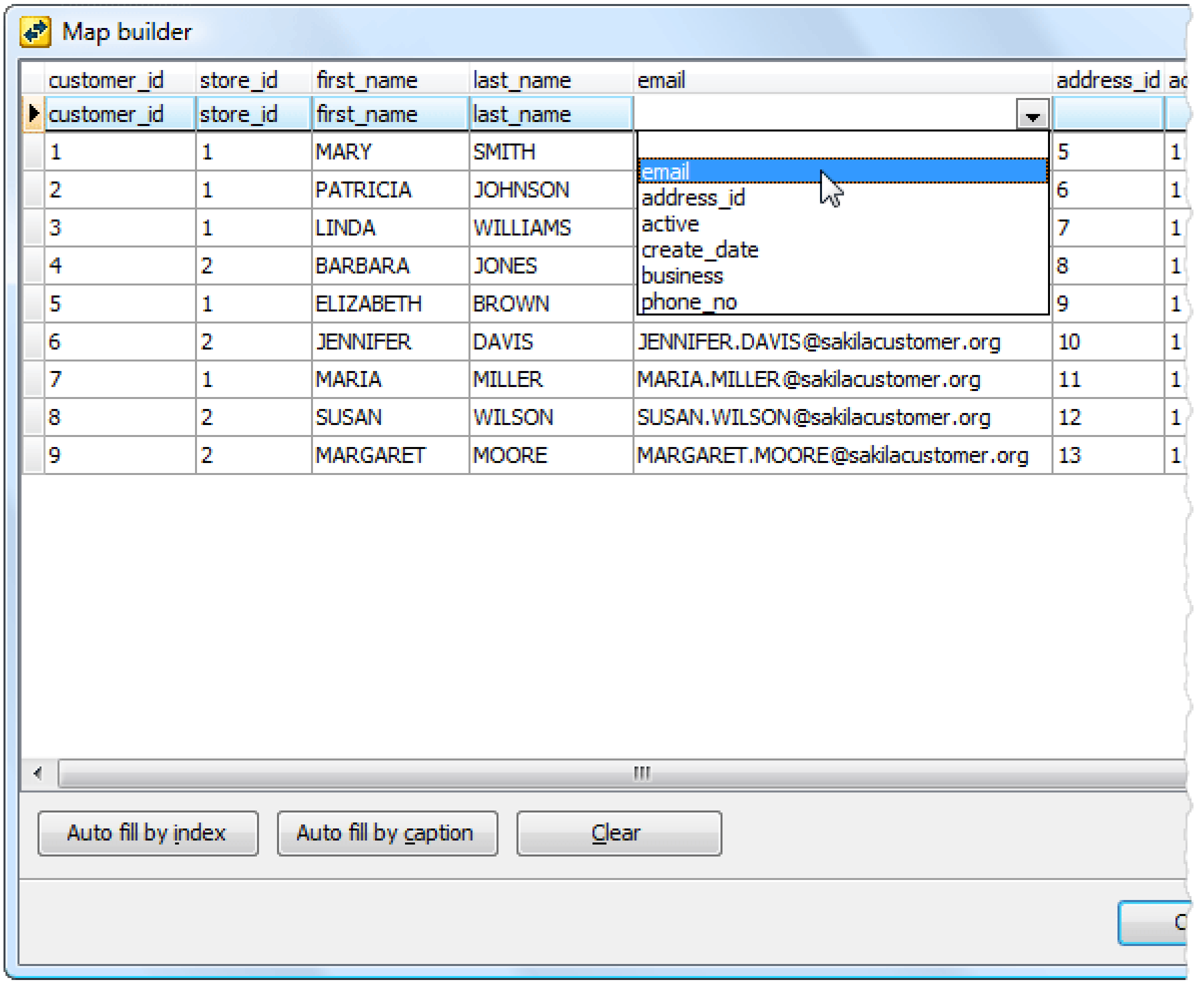


	Target field	Source column	Replacements	Empty values interpretation
1	film_id	A		
2	title	B		As Null
3	description	C		As Null
4	release_year	D		
5	language_id	E		
6	original_language_id	F		
7	rental_duration	G		
8	rental_rate	H		
9	length	I		
10	replacement_cost	J		
11	rating	K		As Null
12	last_update	L		
13	special_features	M		As Null
14	fulltext			As Null

☐ File contains column headers

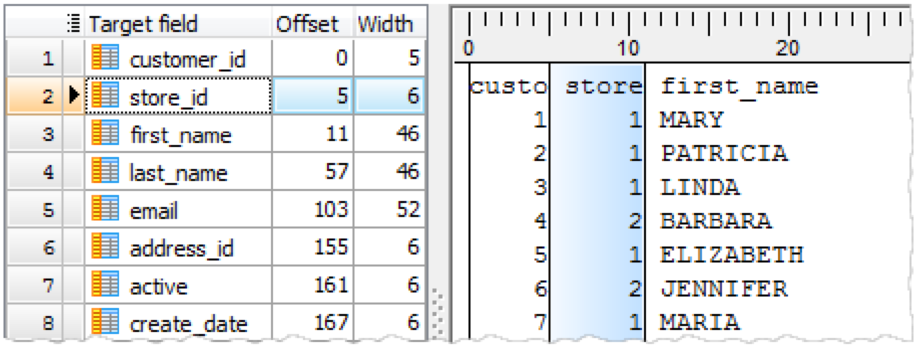
8.5.2.1 Map builder

To specify the accordance between source and target columns visually, use popup menu of the upper row to map source file columns to target table fields.



	customer_id	store_id	first_name	last_name	email	address_id	ac
▶	customer_id	store_id	first_name	last_name			
	1	1	MARY	SMITH		5	1
	2	1	PATRICIA	JOHNSON		6	1
	3	1	LINDA	WILLIAMS		7	1
	4	2	BARBARA	JONES		8	1
	5	1	ELIZABETH	BROWN		9	1
	6	2	JENNIFER	DAVIS	JENNIFER.DAVIS@sakilacustomer.org	10	1
	7	1	MARIA	MILLER	MARIA.MILLER@sakilacustomer.org	11	1
	8	2	SUSAN	WILSON	SUSAN.WILSON@sakilacustomer.org	12	1
	9	2	MARGARET	MOORE	MARGARET.MOORE@sakilacustomer.org	13	1

For text files define columns bounds first. To add a bound, double-click near the column data in the builder area. To map a column to a target field, select the field in the Target field list and then click between the bounds.



	Target field	Offset	Width
1	customer_id	0	5
2	store_id	5	6
3	first_name	11	46
4	last_name	57	46
5	email	103	52
6	address_id	155	6
7	active	161	6
8	create_date	167	6

```
custo store first_name
    1     1 MARY
    2     1 PATRICIA
    3     1 LINDA
    4     2 BARBARA
    5     1 ELIZABETH
    6     2 JENNIFER
    7     1 MARIA
```

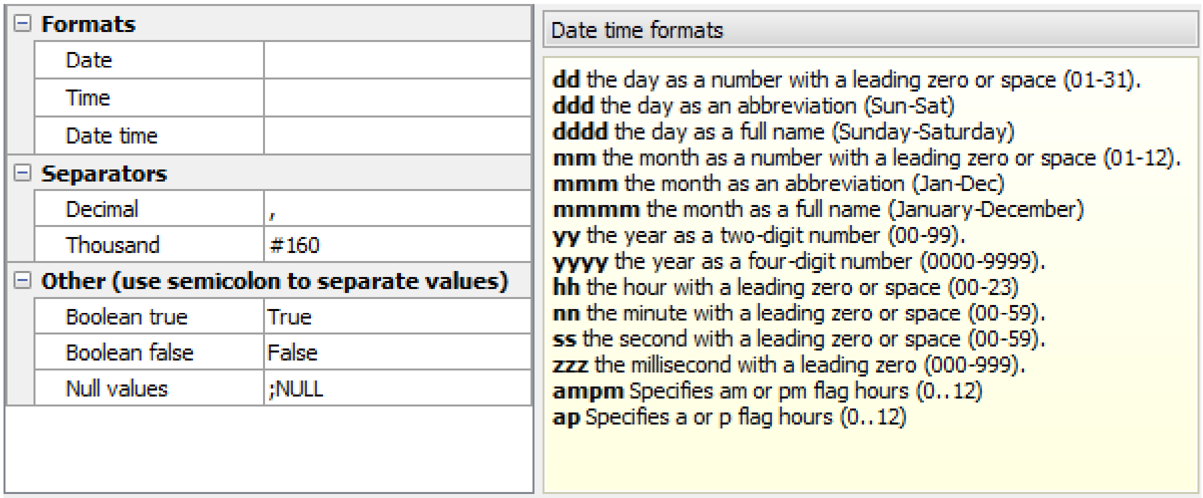
8.5.2.2 Data formats

Use the window fields to indicate format masks of the source data imported to the table. It allows the application to import data correctly.

The components of the date time format mask are represented at the window. Compose your date, time, and date time format mask of this components and separators. The following table contains some types of input fields and suggests masks to import them.

To import these input data correctly	Use these format masks
June 29	mmm dd
Jun 29, 2009	mmmm dd, yyyy
Tue Jun 14 16:50:49	ddd mmm dd hh:nn:ss
01/15/09 08:26 AM	mm/dd/yy h:nn ampm

You can also set decimal and thousand separators, and custom NULL,TRUE and FALSE values. If you have several values to be imported to NULL(TRUE, FALSE) value, use semicolons to separate them.



Formats

Date	
Time	
Date time	

Separators

Decimal	,
Thousand	#160

Other (use semicolon to separate values)

Boolean true	True
Boolean false	False
Null values	;NULL

Date time formats

dd the day as a number with a leading zero or space (01-31).
ddd the day as an abbreviation (Sun-Sat)
dddd the day as a full name (Sunday-Saturday)
mm the month as a number with a leading zero or space (01-12).
mmm the month as an abbreviation (Jan-Dec)
mmmm the month as a full name (January-December)
yy the year as a two-digit number (00-99).
yyyy the year as a four-digit number (0000-9999).
hh the hour with a leading zero or space (00-23)
nn the minute with a leading zero or space (00-59).
ss the second with a leading zero or space (00-59).
zzz the millisecond with a leading zero (000-999).
ampm Specifies am or pm flag hours (0..12)
ap Specifies a or p flag hours (0..12)

8.5.3 Customizing common options

On the wizard step you can set the number of records to import, whether the tool import all table records or only the specified number. In the second case you can set the number of records to skip.

Logging
This options group let you to manage logging of the import process.

Scripts
There are many cases where the import process is necessary to correct with additional scripts. So to disable table indexes before the importing, specify the corresponding scripts to be executed before and after the process.

The typical example of usage of the Before each table and After each table scripts is the import data to autoincrement columns of several tables. In this case it's neseccary to set the corresponding scripts:
SET IDENTITY_INSERT %table_name% ON
and
SET IDENTITY_INSERT %table_name% OFF
to be executed before and after import data to each table correspondingly.

Import mode

If the Update existing records option is turned ON, the records will be either updated or inserted: an UPDATE will be performed when a target row exists in the table and an INSERT is performed when the target row does not exist.

9 Database Tools

MS SQL Maestro provides a number of powerful tools for working with databases.

The following tools are available:

- **SQL Editor** 313
  Creates and executes SQL queries.

- **Visual Query Builder** 318
  Builds queries visually.

- **Script Runner** 363
  Executes SQL scripts to the database.

- **SQL Script Editor** 364
  Allows to edit and execute SQL scripts.

- **Extract Database Wizard** 372
  Extracts the database objects and data to the SQL script, which can be executed later to reserve the database structure and data.

- **Generate Database Report Wizard** 377
  Generates the database HTML or PDF report for structure of selected object in a whole or partially.

- **Backup Database** 366
  Allows to create a copy of the database.

- **BLOB Viewer** 379
  Displays a content of BLOB fields in different representations.

- **Diagram Viewer** 385
  Represents data from a table or a query as a diagram in various ways.

- **Data Analysis** 389
  Allows to slice and dice information efficiently according your business rules.

- **Report Designer** 394
  Prepares data for reading, viewing, and printing in a polished look.

- **Schema Designer** 401
  Allows to represent database tables and relationships as ER diagrams.

- **Process Browser** 406
  A very useful feature for DBAs to monitor the users' activity.

- **SQL Generator** 408
  Provides you with a set of simple SQL statements.

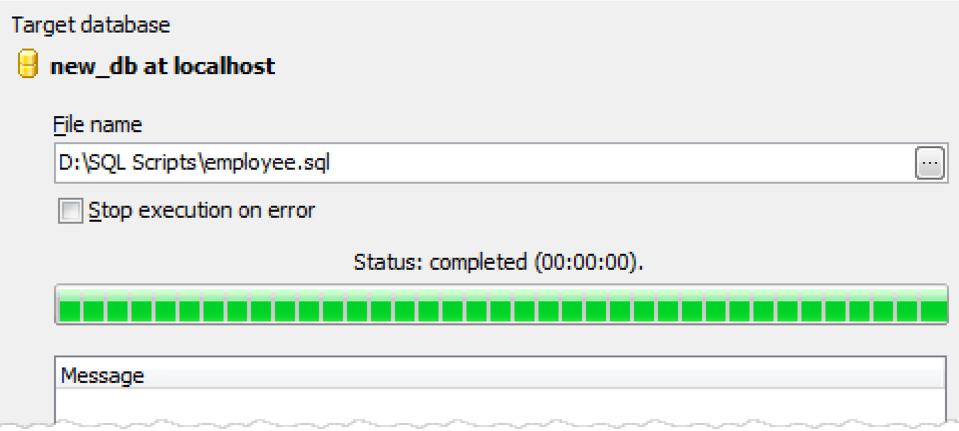
- Simple tools for **DML procedures** 409 and **Updatable views** 411 generation allow to create a bunch of CRUD procedures automatically.

9.1 Script Runner

Script Runner is designed for executing of SQL scripts that don't require modifications. The window can be invoked from the Tools menu or with the Execute script from file link of SQL Script Editor 364.

Script Runner allows to execute .sql files as well as archived scripts directly from .zip files. In case archived files this tool unpacks zip archives to temporary files by itself for further executing.The tool neither starts any implicit transactions before executing the script nor issues COMMIT or ROLLBACK commands after the executing.

To execute a script with Script Runner, set the file name and the Stop execution on error option value. This option allows to view all the execution errors (OFF). The specified script will be executed immediately on the database which name is represented at the top of the window.


Target database

new_db at localhost

File name: D:\SQL Scripts\employee.sql

☐ Stop execution on error

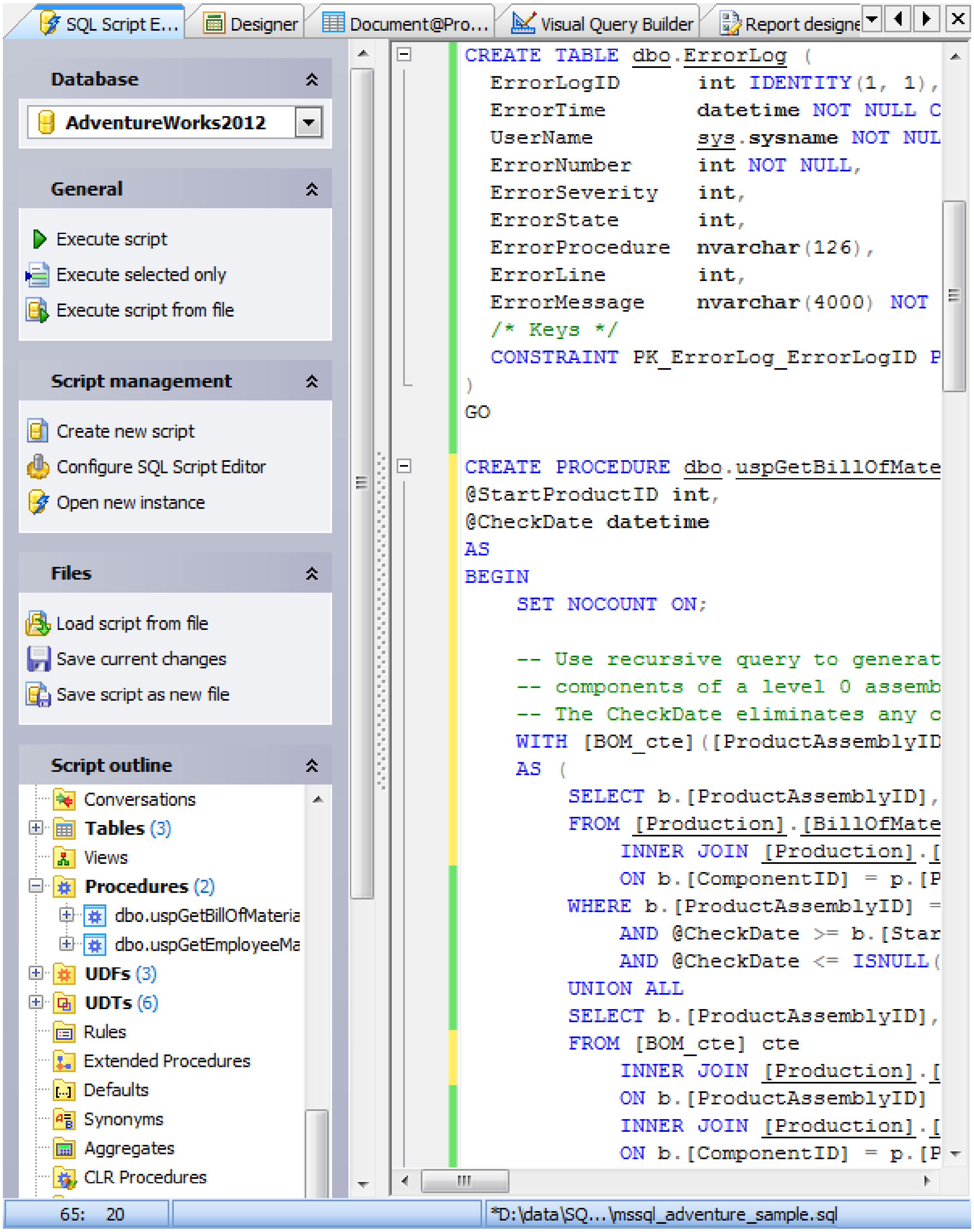
Status: completed (00:00:00).

9.2 SQL Script Editor

SQL Script Editor is designed for SQL scripts editing and executing. The editor does not display results of SELECT queries. To work with such queries' data, use SQL Editor 313. If you have a script that is ready to use, execute it with Script Runner 363. To open SQL Script Editor, select the Tools | SQL Script Editor main menu item.

To work with a script within SQL Script Editor, load it from an *.sql* file or type it in the editor area directly. To prevent mistakes in SQL syntax, the editor supports syntax highlighting, code completion and divides the script text into logical parts that can be individually collapsed or expanded (code folding). All the logical parts are represented at the Explorer at the Navigation bar. It allows you to transfer to the proper script fragment quickly by clicking the corresponding node in the tree.

SQL Script Editor allows you to execute the whole SQL script or only its selected part. To make the executing of a large script much faster, execute the script directly from a file with Script Runner 363. By default, if a user opens a file larger than 100K, SQL Script Editor will suggest him to execute the script file without opening it in the editor. This file size may be changed at the editor's options 432 tab.



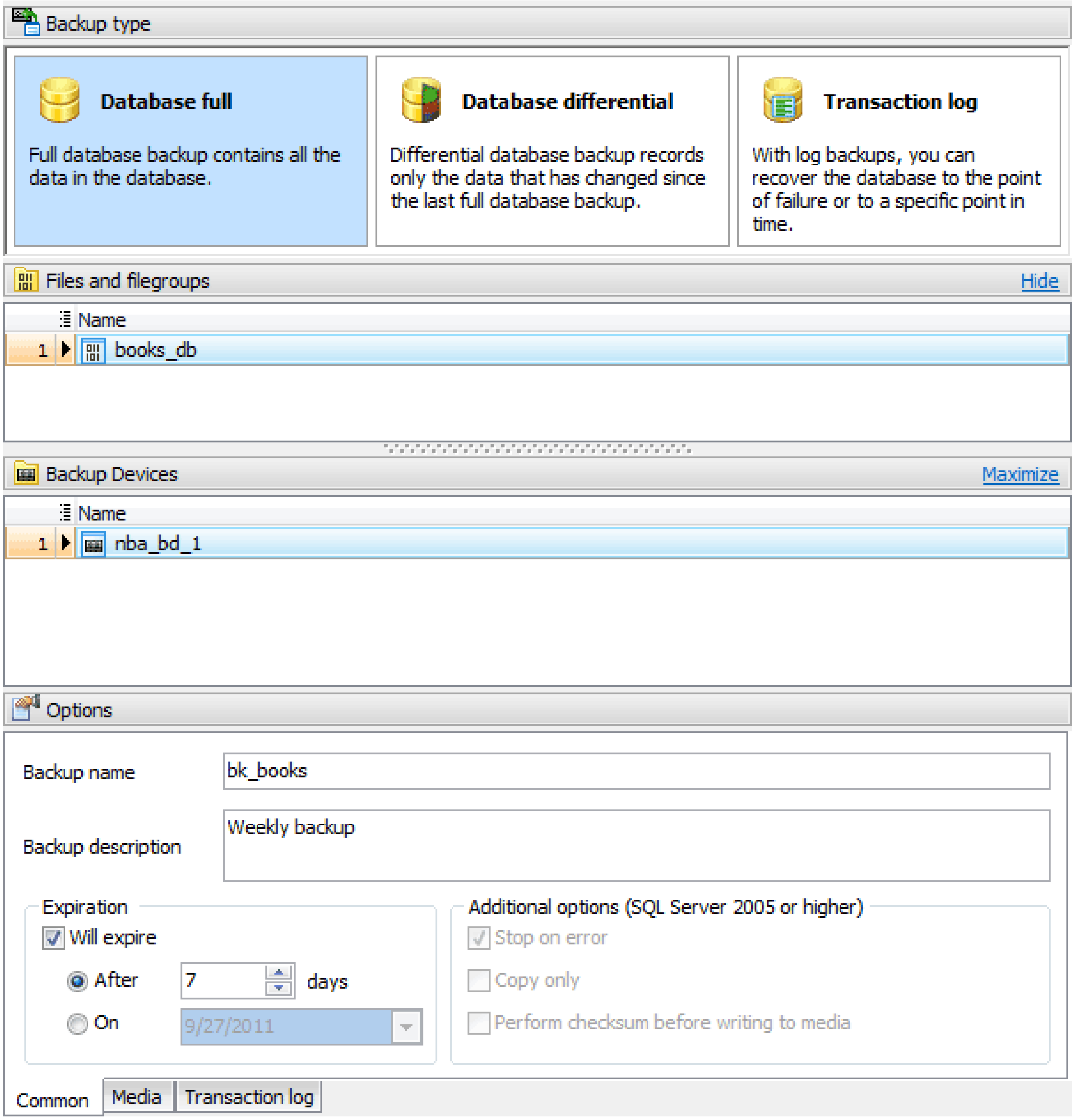
```
CREATE TABLE dbo.ErrorLog (
  ErrorLogID      int IDENTITY(1, 1),
  ErrorTime       datetime NOT NULL C
  UserName        sys.sysname NOT NUL
  ErrorNumber     int NOT NULL,
  ErrorSeverity   int,
  ErrorState      int,
  ErrorProcedure  nvarchar(126),
  ErrorLine       int,
  ErrorMessage    nvarchar(4000) NOT
  /* Keys */
  CONSTRAINT PK_ErrorLog_ErrorLogID P
)
GO

CREATE PROCEDURE dbo.uspGetBillOfMate
@StartProductID int,
@CheckDate datetime
AS
BEGIN
    SET NOCOUNT ON;

    -- Use recursive query to generat
    -- components of a level 0 assemb
    -- The CheckDate eliminates any c
    WITH [BOM_cte]([ProductAssemblyID
    AS (
        SELECT b.[ProductAssemblyID],
        FROM [Production].[BillOfMate
            INNER JOIN [Production].[
            ON b.[ComponentID] = p.[P
        WHERE b.[ProductAssemblyID] =
            AND @CheckDate >= b.[Star
            AND @CheckDate <= ISNULL(
        UNION ALL
        SELECT b.[ProductAssemblyID],
        FROM [BOM_cte] cte
            INNER JOIN [Production].[
            ON b.[ProductAssemblyID]
            INNER JOIN [Production].[
            ON b.[ComponentID] = p.[P
```

9.3 Backup Database

The need to back up databases on a regular basis is a major component of managing any production system. To backup database means to make copies of data which may be used to restore the original 370 after a data loss event. As is well known implementing a well-planned backup and restore strategy protects databases against data loss due to damage caused by a variety of failures.

Backup Database tool allows you create full database backups, differential database backups, and transaction log backups. Use buttons on the top of the window to select the type of backup you want to perform on the specified database. The tool supports only Simple Recovery model. To run this wizard select the Tools | Backup Database main menu item.

To find some explanation for the tools's parts and their elements, see the Backup options 367 topic.



Database full
Full database backup contains all the data in the database.

Database differential
Differential database backup records only the data that has changed since the last full database backup.

Transaction log
With log backups, you can recover the database to the point of failure or to a specific point in time.

Files and filegroups

#	Name
1	books_db

Backup Devices

#	Name
1	nba_bd_1

Backup name: bk_books
Backup description: Weekly backup

Expiration
☑ Will expire
● After 7 days
○ On 9/27/2011

Additional options (SQL Server 2005 or higher)
☑ Stop on error
☐ Copy only
☐ Perform checksum before writing to media

9.3.1 Backup options

Backup Database tool allows you create the various types of data and differential backups. Use buttons on the top of the window to select the type of backup you want to perform on the specified database.

Backup type	**Available for**	**Restrictions**
Database Full	Databases, files, and filegroups	On the master database, only full backups are possible. Under the Simple Recovery Model, file and filegroup backups are available only for read-only filegroups.

Database Differential	Databases, files, and filegroups	Under the Simple Recovery Model, file and filegroup backups are available only for read-only filegroups.
Transaction Log	Transaction logs	Transaction log backups are not available for the Simple Recovery Model.

Files and File Groups part represents selected filegroups or files you want to back up. To manage them, use the appropriated section on the Navigation bar or popup menu of the File and File Group box. For more information about file and file group's properties see Files 208 and File Groups 211.

Select Backup Devices to which to write the data. Microsoft SQL Server 2005 can back up databases, transaction logs, and files to disk and tape devices. To manage them, use the appropriated section on the Navigation bar or popup menu of the Backup Devices box. For more information about their properties see Backup Devices 263.

Common options part allows you to specify Backup name and Backup description. You can also choose one of the following Expiration options.

After	Specify the number of days that must elapse before this backup set expires and can be overwritten. This value can be from 0 to 99999 days; a value of 0 days means that the backup set will never expire.
On	Specify a specific date when the backup set expires and can be overwritten.

Additional options (Available only for Microsoft SQL Server 2005)

☑ Stop on error
Instructs BACKUP whether to continue despite encountering errors such as invalid checksums.

☑ Copy only specifies that the backup not affect the normal sequence of backups. A copy-only does not affect the overall backup and restore procedures for the database. You can create a copy-only backup for any type of backup. The effect of this option varies with the general backup type, as follows:

- A data backup taken with the option cannot be used as a base backup for differential backups. Differential backups taken later will behave as if the copy-only backup does not exist.

- A differential backup is unaffected by the option.

- A log backup taken using the option does not truncate the transaction log.

☑ Perform checksum before writing to media specifies that:
Prior to writing a page to the backup media, BACKUP verifies the page (checksum or torn page), if this information is present on the page.

Regardless of whether page checksums are present, BACKUP generates a separate backup checksum for the backup streams. Restore operations can optionally use the backup checksum to validate that the backup is not corrupt. The backup checksum is stored on the backup media, not on the database pages. The backup checksum can optionally be used at restore time,

> **Note:** Using backup checksums may affect workload and backup throughput.

Use Media options to define backup write method. Possible values are: Append to existing media (appends to the existing file), Overwrite existing media (overwrites the file).

Media name
Define the media name for the entire backup media set. If name is specified, it must match the previously specified media name already existing on the backup volumes. If it is not specified, there is no verification check of the media name.

Media description
You can specify the description of the media set in this field.

Media password
Enter the password for the media set to overwrite the backup only by reformatting.
Note, if the backup media is password protected, SQL Server does not write to the media unless the media password is supplied. This check is not overridden by the SKIP option. Password-protected media may be overwritten only by reformatting it.

☑ Check media set name, password and expiration date
If checked, instructs the BACKUP statement to check these elements for all backup sets on the media before allowing them to be overwritten.

Use Transaction log options (Available only if Transaction Log Backup selected) part to specify that the log not be truncated and cause the database engine to attempt the backup regardless of the state of the database (Backup the tail of the log, and leave the database in the restoring state). This option allows backing up the log in situations where the database is damaged.

With the Truncate the transaction log option, the database must be online.

Use the Execute link on the Navigation bar to start the backup process.

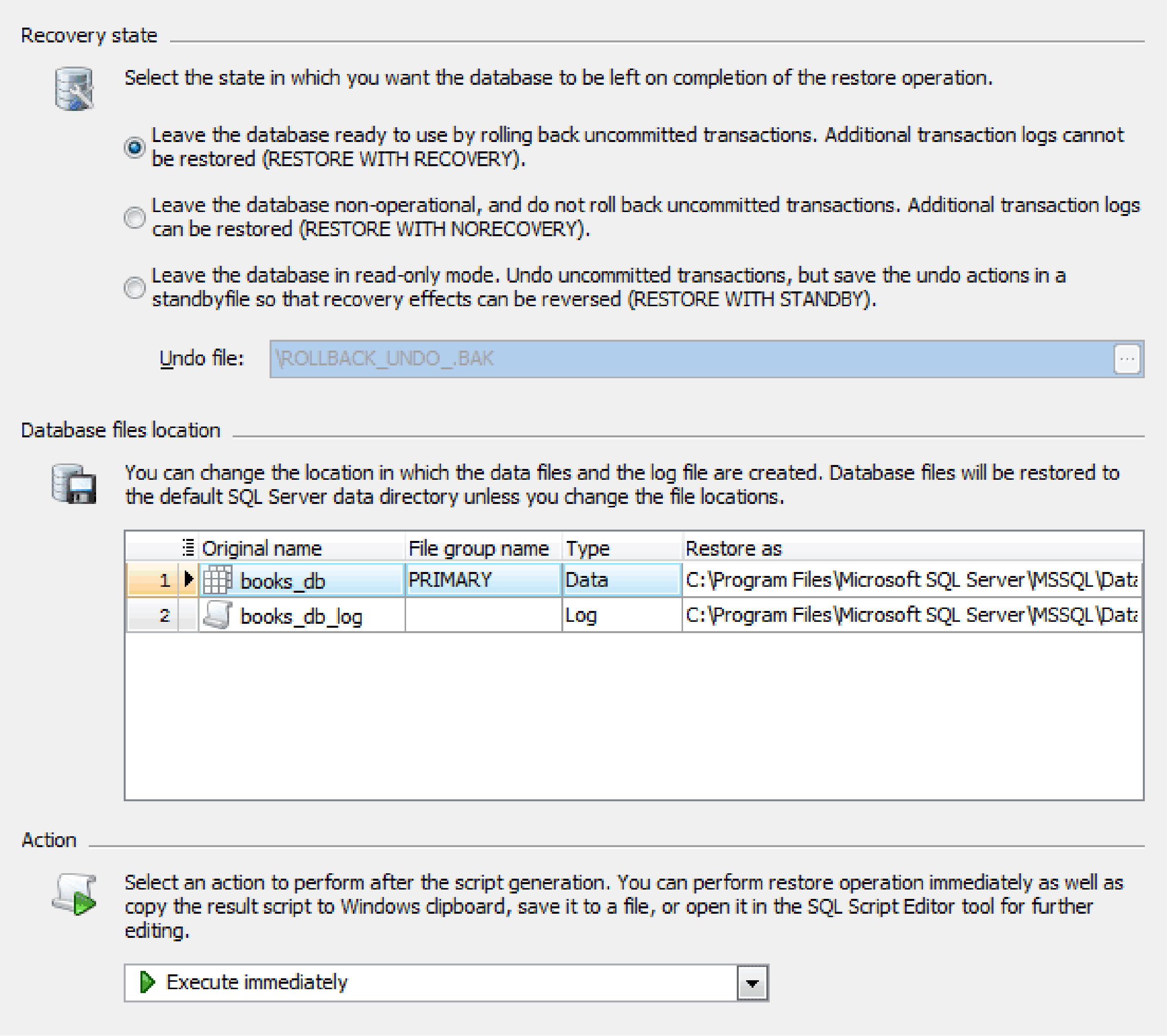
9.4 Restore Database

A database backup 366 may be restored to the existing database or a new one on the same (or a different) server. With the Restore Database tool you can restore full and differential backups using the Simple Recovery model. To invoke this wizard, select the Tools | Restore Database main menu item.

First you need to connect to the server where the backup is to be restored, run the wizard and specify the database for which the backup has to be restored. To restore the backup to a new database, select the Create new database option and specify the connection settings to the database to be created.

After that select the backup to be used for the restoring. By default, the Backup Devices part displays the default backup device. To add other devices, use the corresponding button. The Backups part represents a list of all backups stored on the selected backup devices. Select the backup to restore from the list by checking the box near the backup name.

Now select the final restore state, define the location in which the data files and log files are created, and choose the action to perform after the script generation.



Recovery state

Select the state in which you want the database to be left on completion of the restore operation.

● Leave the database ready to use by rolling back uncommitted transactions. Additional transaction logs cannot be restored (RESTORE WITH RECOVERY).

○ Leave the database non-operational, and do not roll back uncommitted transactions. Additional transaction logs can be restored (RESTORE WITH NORECOVERY).

○ Leave the database in read-only mode. Undo uncommitted transactions, but save the undo actions in a standbyfile so that recovery effects can be reversed (RESTORE WITH STANDBY).

Undo file: \ROLLBACK_UNDO_.BAK

Database files location

You can change the location in which the data files and the log file are created. Database files will be restored to the default SQL Server data directory unless you change the file locations.

	Original name	File group name	Type	Restore as
1	books_db	PRIMARY	Data	C:\Program Files\Microsoft SQL Server\MSSQL\Dat
2	books_db_log		Log	C:\Program Files\Microsoft SQL Server\MSSQL\Dat

Action

Select an action to perform after the script generation. You can perform restore operation immediately as well as copy the result script to Windows clipboard, save it to a file, or open it in the SQL Script Editor tool for further editing.

Execute immediately

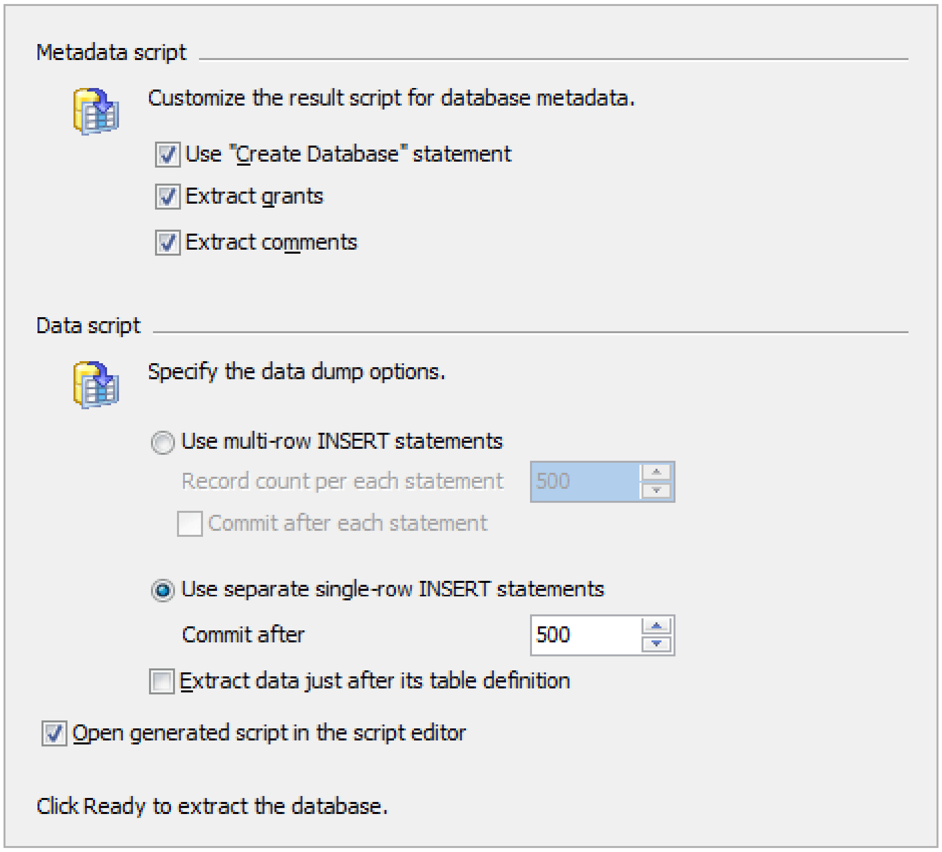
9.5 Extract Database

Using Extract Database Wizard you can extract database objects and data to a SQL script, e.g. for backup purposes. To run this wizard select the Tools | Extract Database main menu item.

Use the More... button to save the extract configuration for future use or to load the previously saved configuration for faster extract.

- Selecting database to extract and the target script file name 372
- Selecting database objects to extract their structure 373
- Selecting database objects to extract their data 374
- Customizing script options 375

See also: Get SQL Dump Wizard 351



Metadata script

Customize the result script for database metadata.

☑ Use "Create Database" statement

☑ Extract grants

☑ Extract comments

Data script

Specify the data dump options.

○ Use multi-row INSERT statements

Record count per each statement 500

☐ Commit after each statement

● Use separate single-row INSERT statements

Commit after 500

☐ Extract data just after its table definition

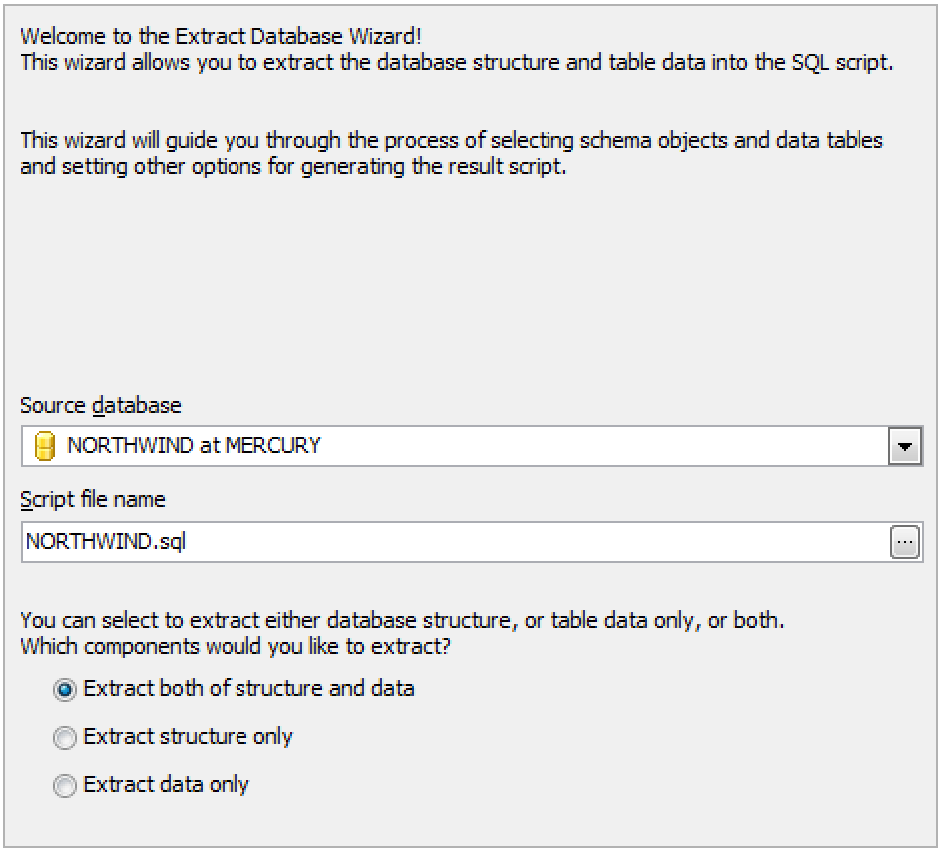
☑ Open generated script in the script editor

Click Ready to extract the database.

9.5.1 Selecting the database and the target file name

Select the source database to extract and set the target script file name.

Select the components to be extracted: object definitions, table data or both.



Welcome to the Extract Database Wizard!
This wizard allows you to extract the database structure and table data into the SQL script.

This wizard will guide you through the process of selecting schema objects and data tables and setting other options for generating the result script.

Source database: NORTHWIND at MERCURY

Script file name: NORTHWIND.sql

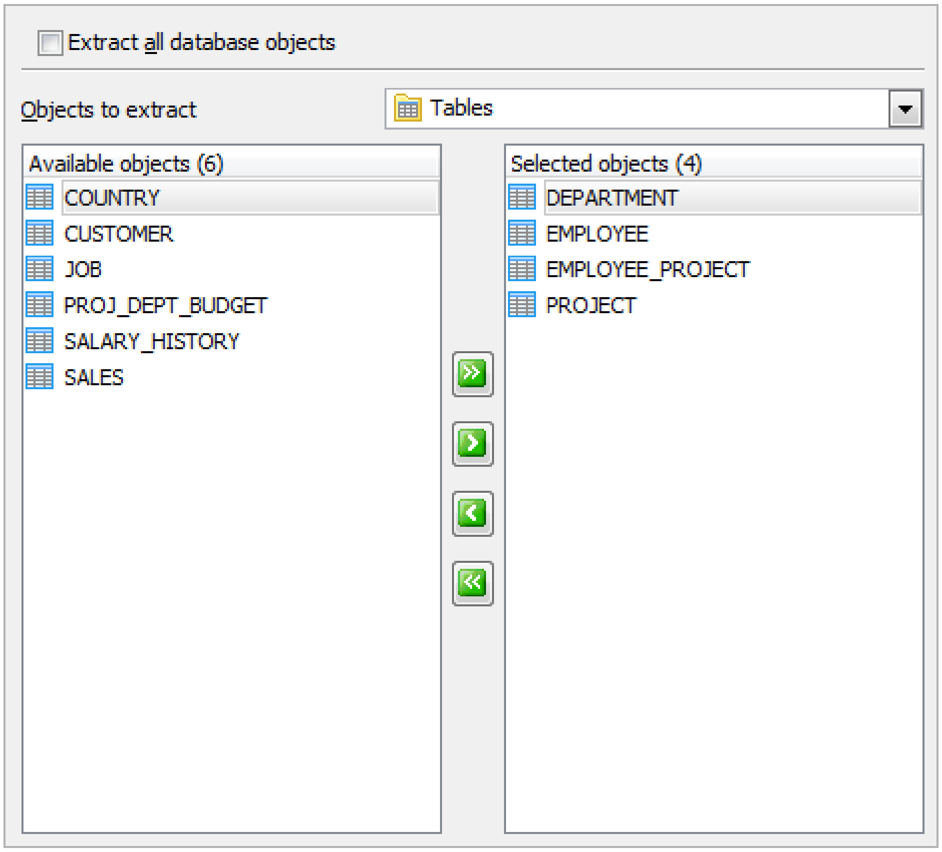
You can select to extract either database structure, or table data only, or both.
Which components would you like to extract?

- ● Extract both of structure and data
- ○ Extract structure only
- ○ Extract data only

9.5.2 Selecting objects to extract their structure

Select the database object to be extracted or check the Extract all database objects option to extract all objects from the database.

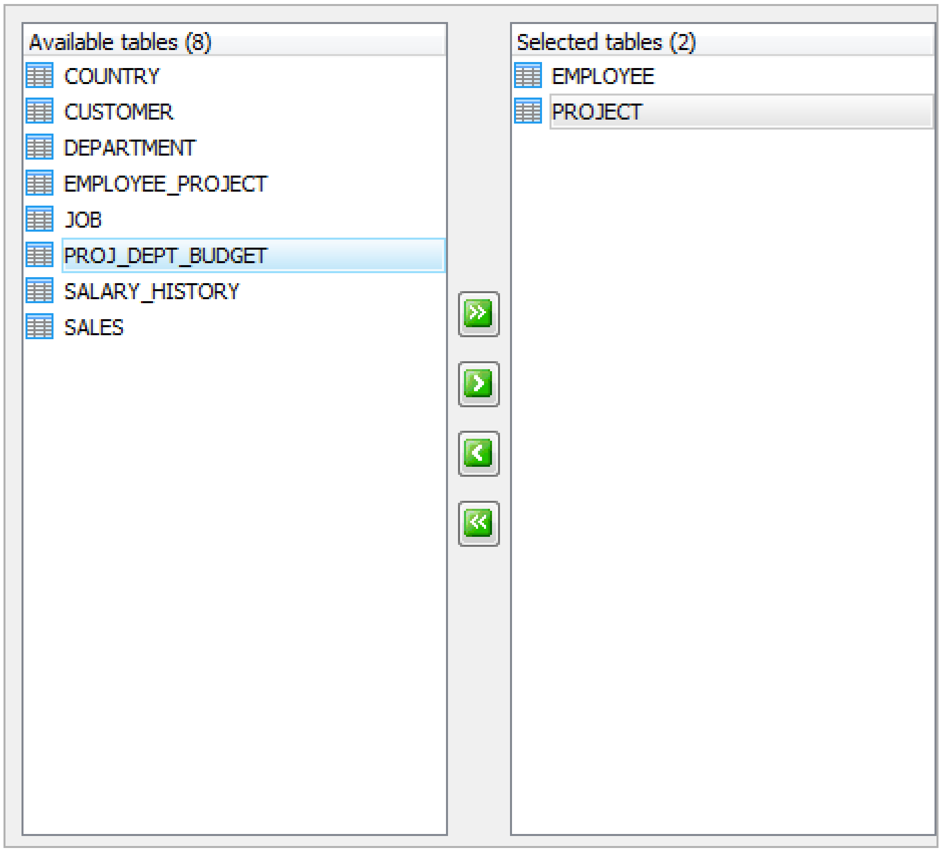
Note: this step is not available if you select the extract data only mode when selecting components to be extracted 372.



9.5.3 Selecting objects to extract their data

Select the tables to extract their data by moving them from the Available tables list to the Selected tables list.

Note: this step is not available if you select the extract structure only mode when selecting components to be extracted 372.



9.5.4 Customizing script options

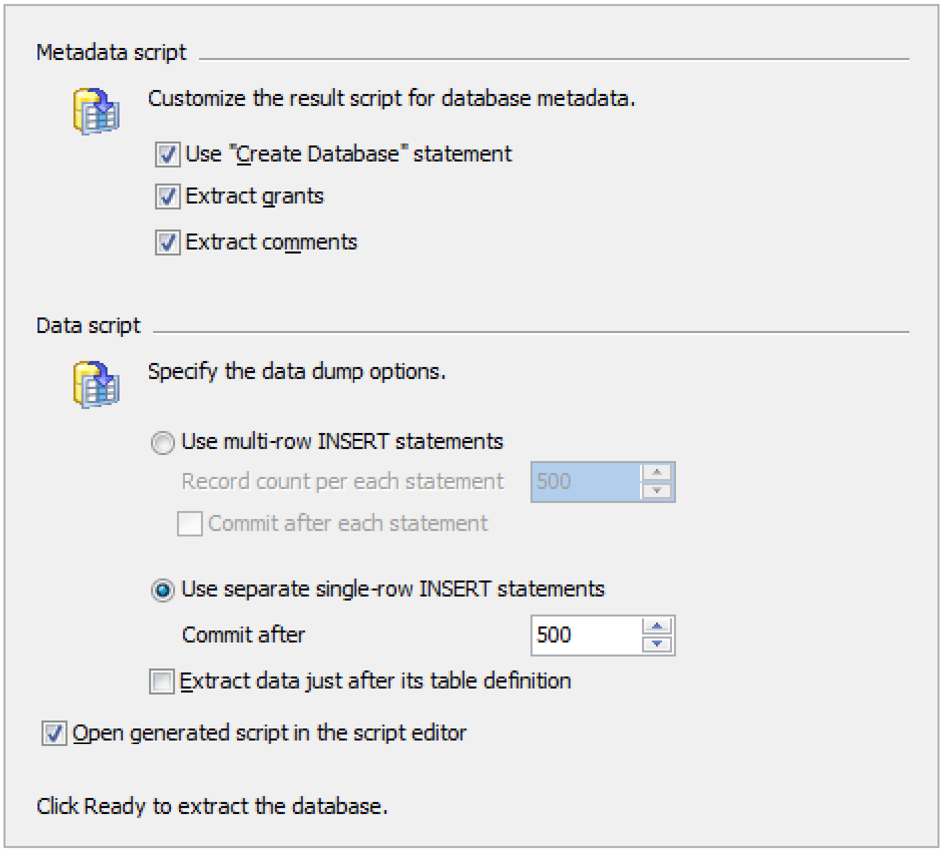
Adjust the result script for database metadata according to your needs. To include/exclude the following statements into the script, check the corresponding boxes: *CREATE DATABASE*, *DROP DATABASE IF EXISTS*, *DROP IF EXISTS* statements for database objects, and foreign key checks.

Select the data dump mode to be used (Multi-row INSERT statements or separate single-row INSERT statements) and specify commits' frequency.

You can also set extract data after its table definition.

Fill the Open generated script in the script editor box to load the result script to SQL Script Editor 364.

Click the Ready button to start the extraction process.



Metadata script

Customize the result script for database metadata.

- ☑ Use "Create Database" statement
- ☑ Extract grants
- ☑ Extract comments

Data script

Specify the data dump options.

- ○ Use multi-row INSERT statements
  - Record count per each statement: 500
  - ☐ Commit after each statement
- ● Use separate single-row INSERT statements
  - Commit after: 500
- ☐ Extract data just after its table definition

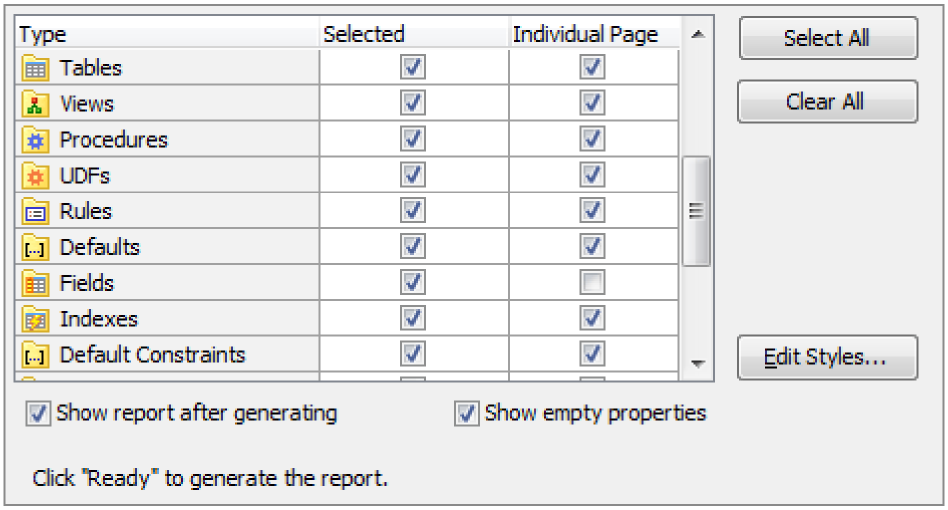
☑ Open generated script in the script editor

Click Ready to extract the database.

9.6 Generate Database Report

With the help of Generate Database Report Wizard you can create HTML or PDF report for the structure of the selected object in the whole or partially. To run this wizard select the Tools | Generate Database Report main menu item.

- Selecting reporting elements and setting other report options 377
- Specifying reporting objects and editing styles 377



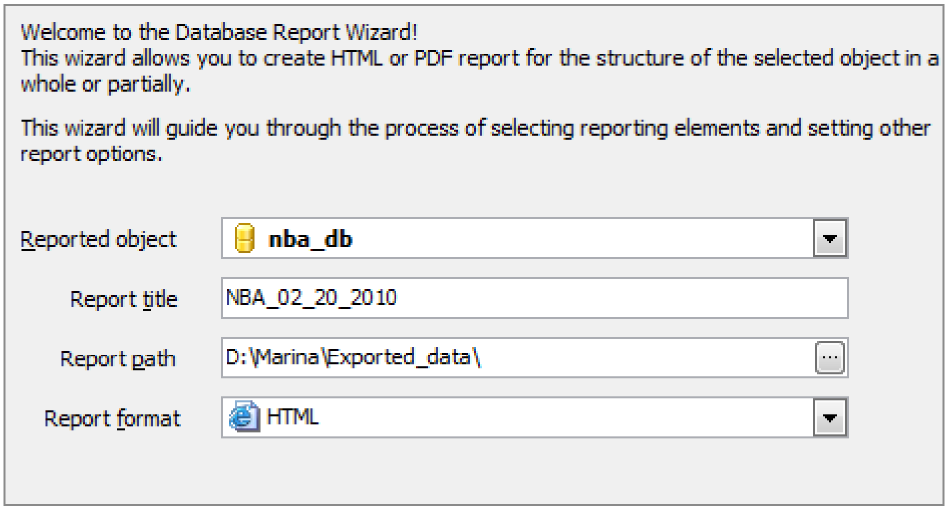
Type	Selected	Individual Page
Tables	☑	☑
Views	☑	☑
Procedures	☑	☑
UDFs	☑	☑
Rules	☑	☑
Defaults	☑	☑
Fields	☑	☐
Indexes	☑	☑
Default Constraints	☑	☑

☑ Show report after generating ☑ Show empty properties

Click "Ready" to generate the report.

9.6.1 Selecting reporting elements and setting other report options

Select the report object and the report format items, set the report title and the report path options in the respective boxes.



Welcome to the Database Report Wizard!
This wizard allows you to create HTML or PDF report for the structure of the selected object in a whole or partially.

This wizard will guide you through the process of selecting reporting elements and setting other report options.

Reported object: **nba_db**
Report title: NBA_02_20_2010
Report path: D:\Marina\Exported_data\
Report format: HTML

9.6.2 Reporting objects and editing styles options

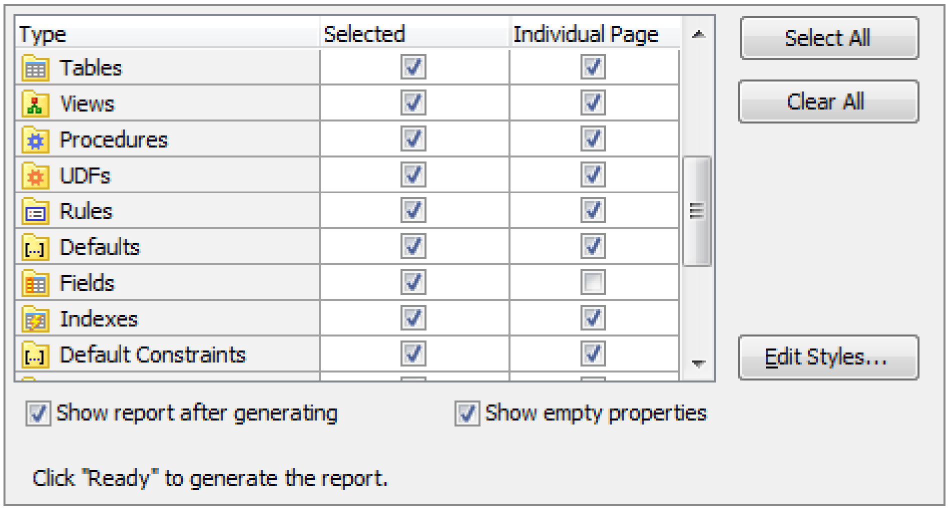
This step allows you to select the essential objects to report and to specify the output format and style using Report Style Editor 378.

☑ Show report after generating

If checked, opens the result files in the associated program after making the report.

☑ Show empty properties

If checked, allows you to report objects even if they are empty.



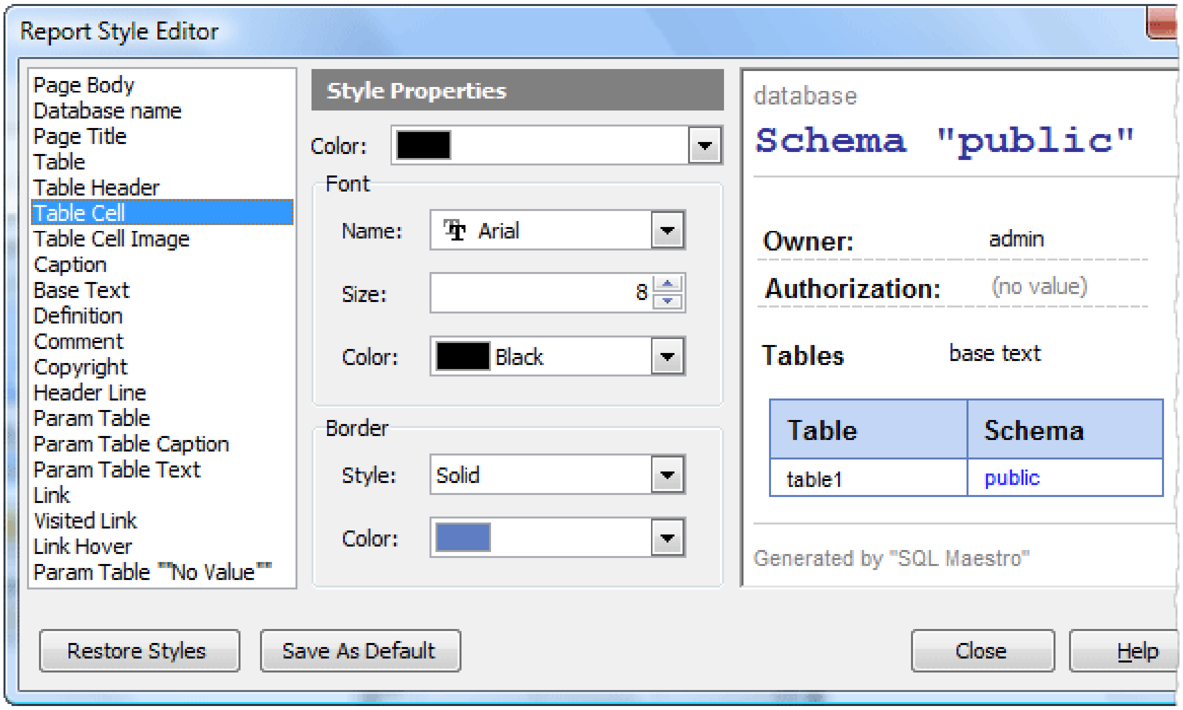
Type	Selected	Individual Page
Tables	☑	☑
Views	☑	☑
Procedures	☑	☑
UDFs	☑	☑
Rules	☑	☑
Defaults	☑	☑
Fields	☑	☐
Indexes	☑	☑
Default Constraints	☑	☑

☑ Show report after generating ☑ Show empty properties

Click "Ready" to generate the report.

9.6.3 Editing database report style

Using Report Style Editor you can specify style properties of a report including font size, color and name for different elements.



Report Style Editor

- Page Body
- Database name
- Page Title
- Table
- Table Header
- Table Cell
- Table Cell Image
- Caption
- Base Text
- Definition
- Comment
- Copyright
- Header Line
- Param Table
- Param Table Caption
- Param Table Text
- Link
- Visited Link
- Link Hover
- Param Table ""No Value""

Style Properties

Color: ■

Font

Name: Arial

Size: 8

Color: ■ Black

Border

Style: Solid

Color: ■

database

Schema "public"

Owner: admin

Authorization: (no value)

Tables base text

Table	Schema
table1	public

Generated by "SQL Maestro"

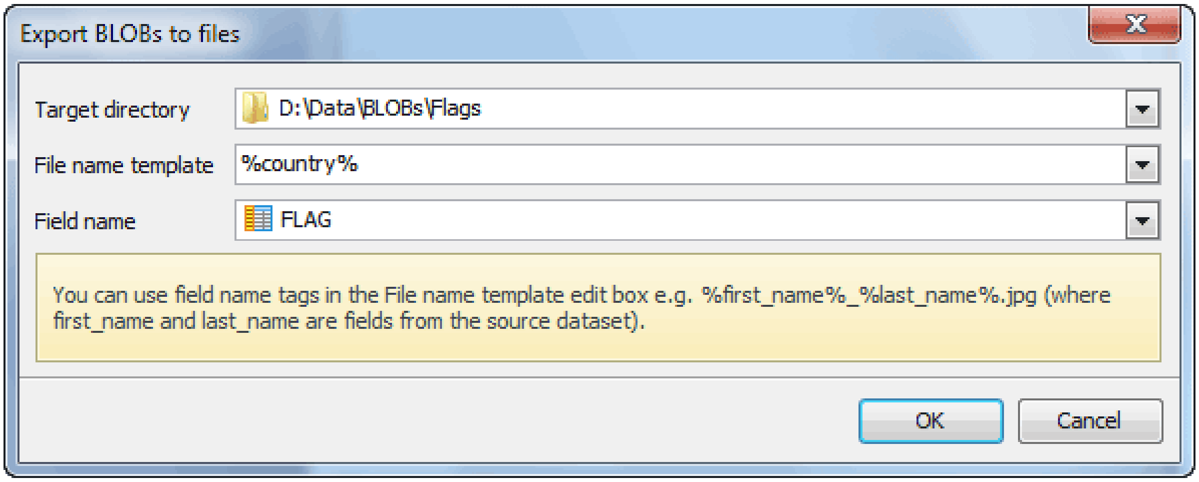
9.7 BLOB Viewer

BLOB Viewer allows you to view the content of the BLOB fields in various representations.

- Viewing BLOB field as hexadecimal dump 379
- Viewing BLOB field as plain text 380
- Viewing BLOB field as graphical image 381
- Viewing BLOB field as HTML 382
- Viewing BLOB field as PDF 383

See also: BLOB Editor 339

BLOB Viewer also allows you to save all BLOBs from a table or view to a given directory. Just click Save to files on the Navigation bar and fill all fields in the Export BLOBs window shown below. You can use table columns' names enclosed in % as a file name template.



Target directory: D:\Data\BLOBs\Flags

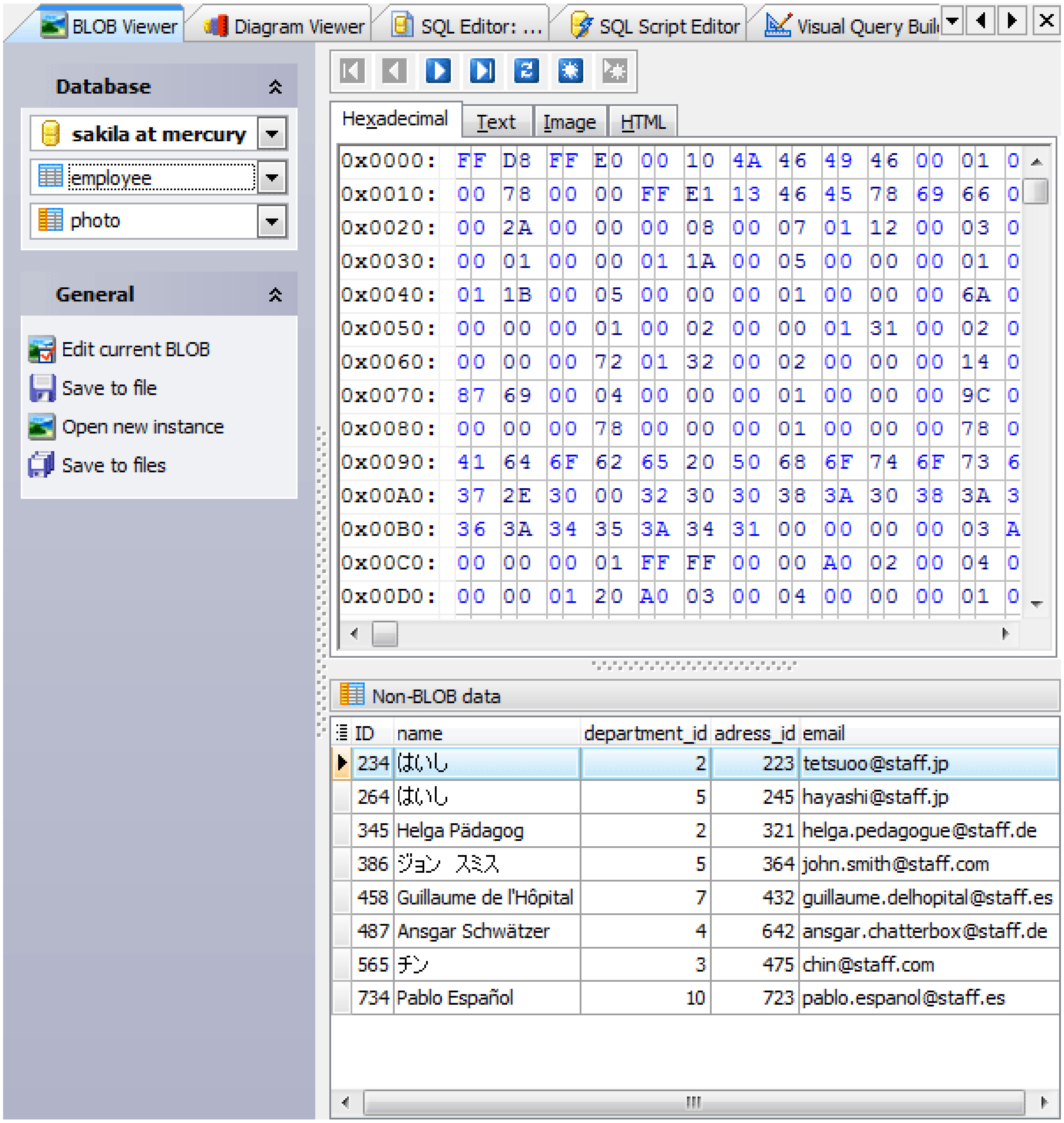
File name template: %country%

Field name: FLAG

You can use field name tags in the File name template edit box e.g. %first_name%_%last_name%.jpg (where first_name and last_name are fields from the source dataset).

9.7.1 Viewing as hexadecimal dump

The Hexadecimal panel allows you to view data in hexadecimal mode.

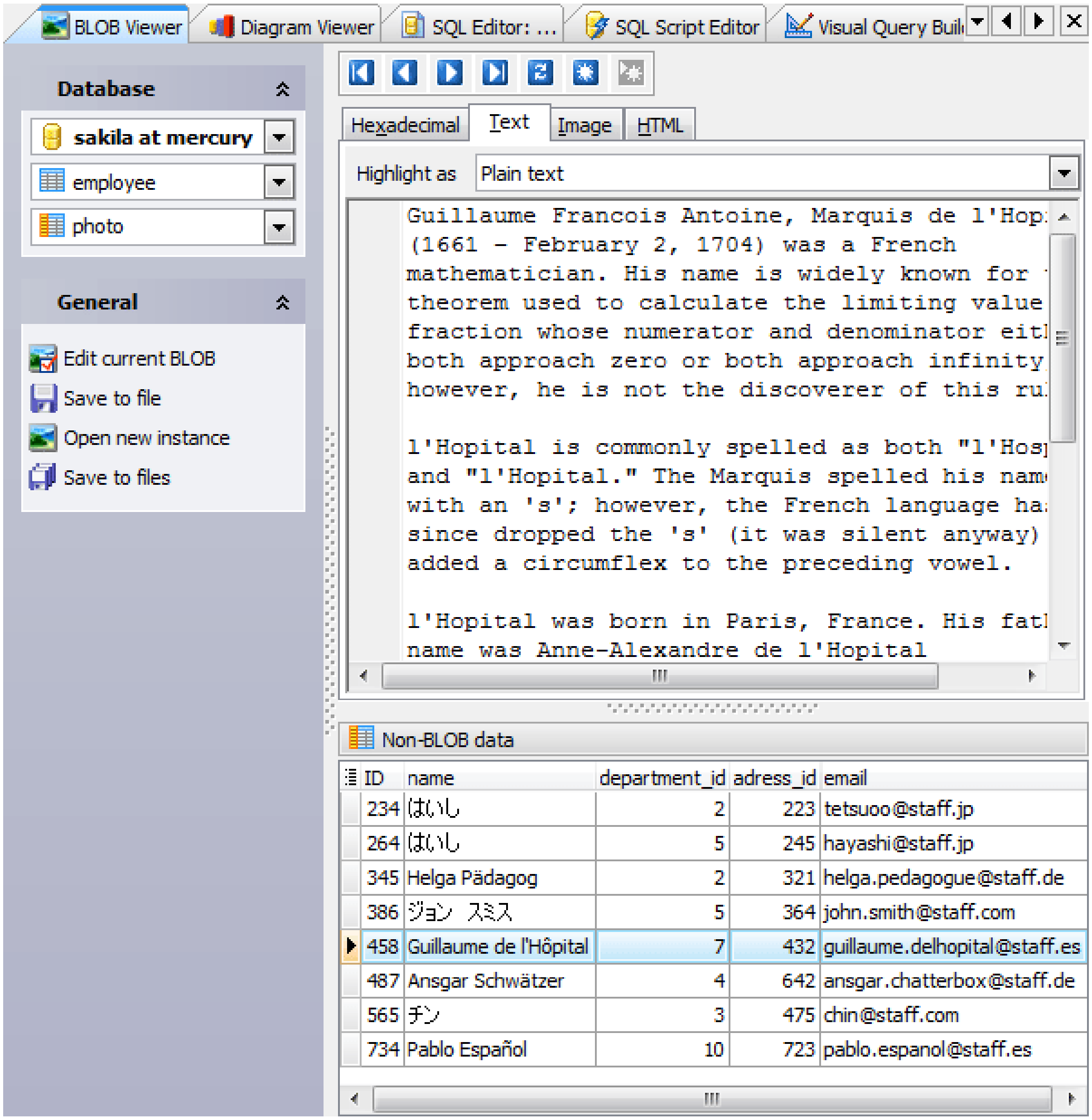


```
0x0000: FF D8 FF E0 00 10 4A 46 49 46 00 01 0
0x0010: 00 78 00 00 FF E1 13 46 45 78 69 66 0
0x0020: 00 2A 00 00 00 08 00 07 01 12 00 03 0
0x0030: 00 01 00 00 01 1A 00 05 00 00 00 01 0
0x0040: 01 1B 00 05 00 00 00 01 00 00 00 6A 0
0x0050: 00 00 00 01 00 02 00 00 01 31 00 02 0
0x0060: 00 00 00 72 01 32 00 02 00 00 00 14 0
0x0070: 87 69 00 04 00 00 00 01 00 00 00 9C 0
0x0080: 00 00 00 78 00 00 00 01 00 00 00 78 0
0x0090: 41 64 6F 62 65 20 50 68 6F 74 6F 73 6
0x00A0: 37 2E 30 00 32 30 30 38 3A 30 38 3A 3
0x00B0: 36 3A 34 35 3A 34 31 00 00 00 00 03 A
0x00C0: 00 00 00 01 FF FF 00 00 A0 02 00 04 0
0x00D0: 00 00 01 20 A0 03 00 04 00 00 00 01 0
```

ID	name	department_id	adress_id	email
234	はいし	2	223	tetsuoo@staff.jp
264	はいし	5	245	hayashi@staff.jp
345	Helga Pädagog	2	321	helga.pedagogue@staff.de
386	ジョン スミス	5	364	john.smith@staff.com
458	Guillaume de l'Hôpital	7	432	guillaume.delhopital@staff.es
487	Ansgar Schwätzer	4	642	ansgar.chatterbox@staff.de
565	チン	3	475	chin@staff.com
734	Pablo Español	10	723	pablo.espanol@staff.es

9.7.2 Viewing as plain text

The Text panel allows you to view data as simple text. For your convenience several types of text highlighting are available (*Plain text, HTML, JScript, CSS, PHP, XML, SQL,* and *SQLite DDL*). The popup menu of the panel provides you to Find or Replace a necessary text fragment.



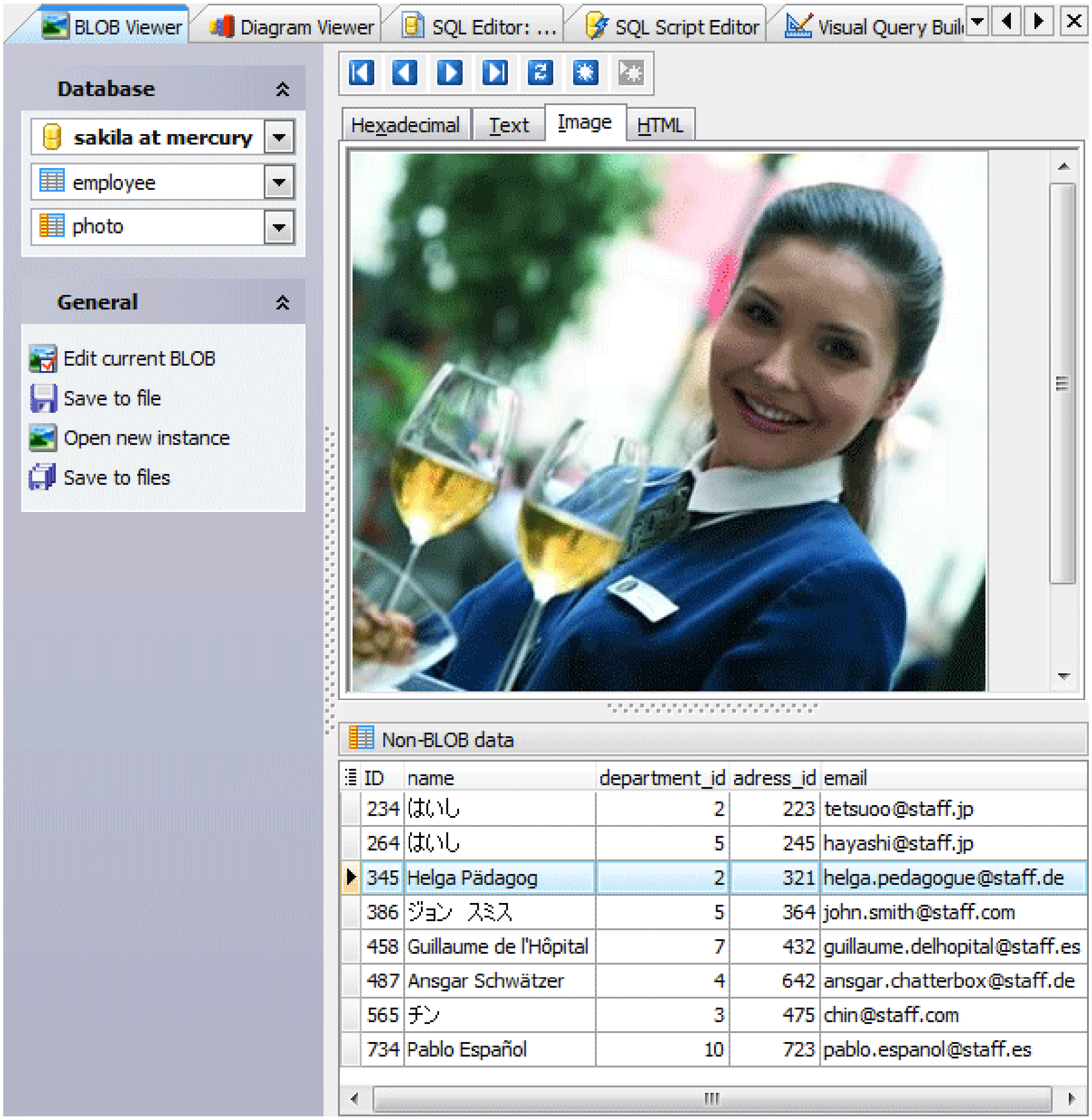
Highlight as: Plain text

```
Guillaume Francois Antoine, Marquis de l'Hop
(1661 - February 2, 1704) was a French
mathematician. His name is widely known for
theorem used to calculate the limiting value
fraction whose numerator and denominator eit
both approach zero or both approach infinity
however, he is not the discoverer of this ru

l'Hopital is commonly spelled as both "l'Hos
and "l'Hopital." The Marquis spelled his nam
with an 's'; however, the French language ha
since dropped the 's' (it was silent anyway)
added a circumflex to the preceding vowel.

l'Hopital was born in Paris, France. His fat
name was Anne-Alexandre de l'Hopital
```

ID	name	department_id	adress_id	email
234	はいし	2	223	tetsuoo@staff.jp
264	はいし	5	245	hayashi@staff.jp
345	Helga Pädagog	2	321	helga.pedagogue@staff.de
386	ジョン スミス	5	364	john.smith@staff.com
458	Guillaume de l'Hôpital	7	432	guillaume.delhopital@staff.es
487	Ansgar Schwätzer	4	642	ansgar.chatterbox@staff.de
565	チン	3	475	chin@staff.com
734	Pablo Español	10	723	pablo.espanol@staff.es

9.7.3 Viewing as image

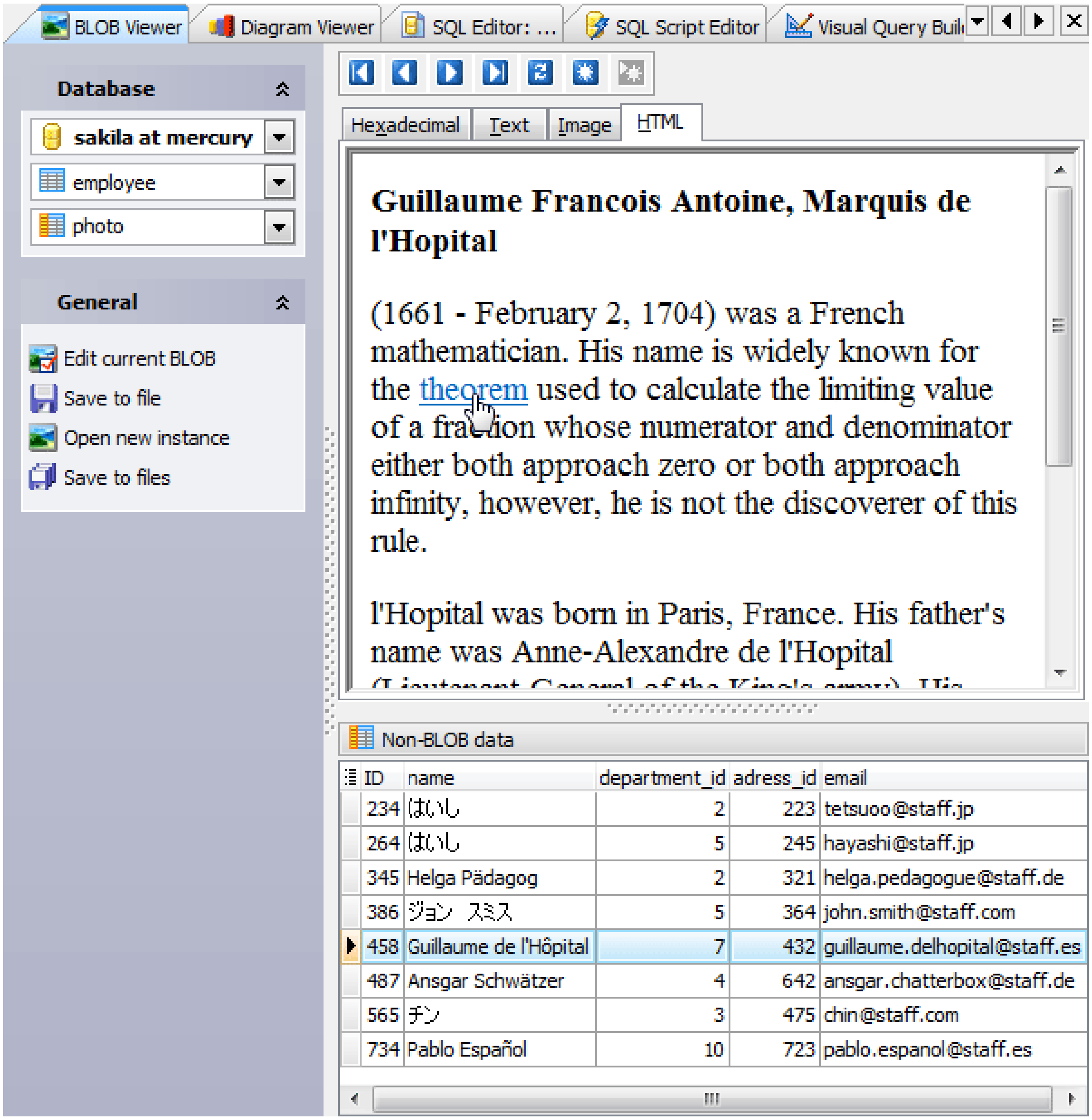
The Image panel displays field data as image.



ID	name	department_id	adress_id	email
234	はいし	2	223	tetsuoo@staff.jp
264	はいし	5	245	hayashi@staff.jp
345	Helga Pädagog	2	321	helga.pedagogue@staff.de
386	ジョン スミス	5	364	john.smith@staff.com
458	Guillaume de l'Hôpital	7	432	guillaume.delhopital@staff.es
487	Ansgar Schwätzer	4	642	ansgar.chatterbox@staff.de
565	チン	3	475	chin@staff.com
734	Pablo Español	10	723	pablo.espanol@staff.es

9.7.4 Viewing as HTML

The HTML panel displays field data as HTML.



Guillaume Francois Antoine, Marquis de l'Hopital

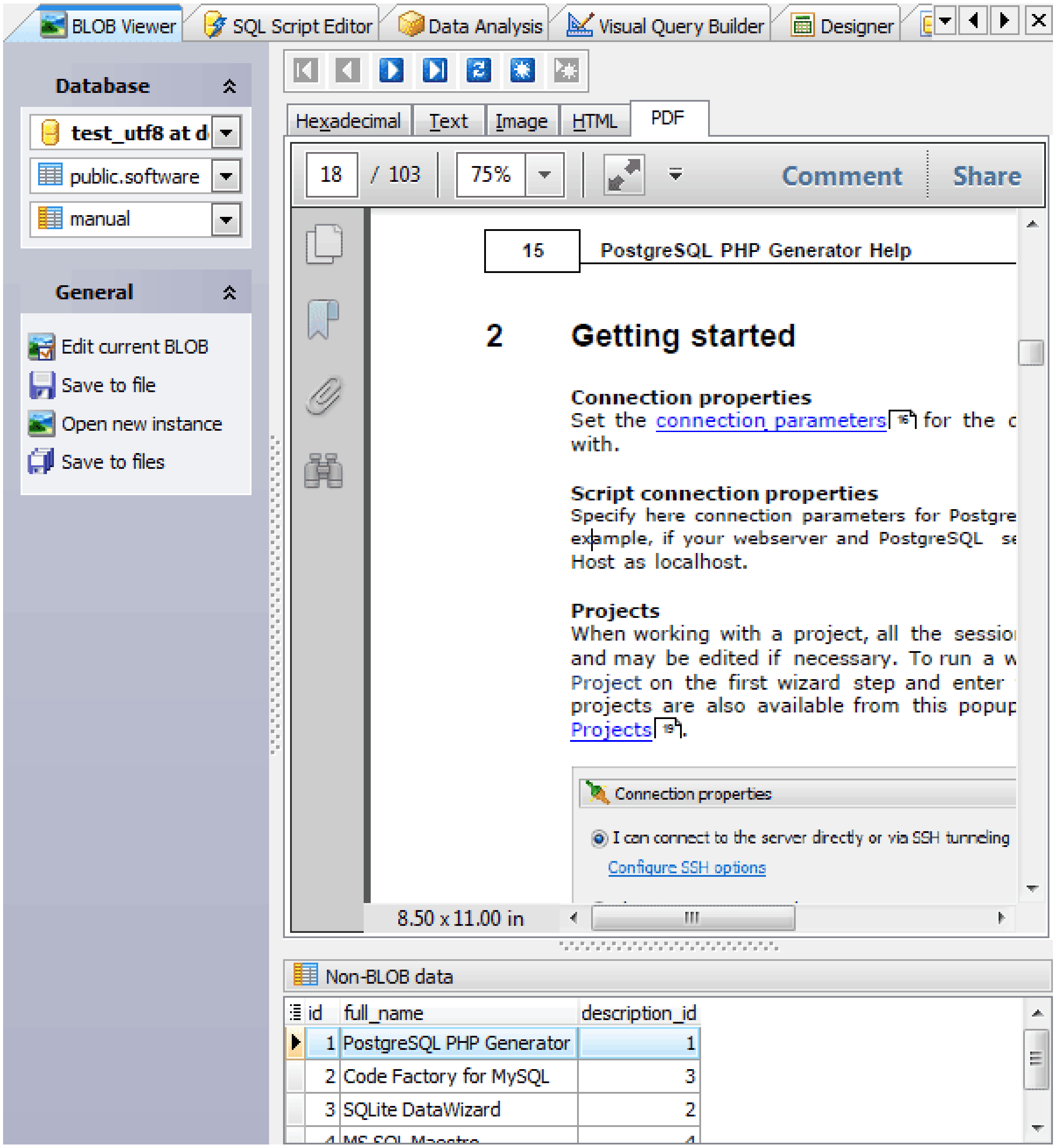
(1661 - February 2, 1704) was a French mathematician. His name is widely known for the theorem used to calculate the limiting value of a fraction whose numerator and denominator either both approach zero or both approach infinity, however, he is not the discoverer of this rule.

l'Hopital was born in Paris, France. His father's name was Anne-Alexandre de l'Hopital

ID	name	department_id	adress_id	email
234	はいし	2	223	tetsuoo@staff.jp
264	はいし	5	245	hayashi@staff.jp
345	Helga Pädagog	2	321	helga.pedagogue@staff.de
386	ジョン スミス	5	364	john.smith@staff.com
458	Guillaume de l'Hôpital	7	432	guillaume.delhopital@staff.es
487	Ansgar Schwätzer	4	642	ansgar.chatterbox@staff.de
565	チン	3	475	chin@staff.com
734	Pablo Español	10	723	pablo.espanol@staff.es

9.7.5 Viewing as PDF

The PDF panel allows you to browse PDF data stored in the database.



test_utf8 at d

public.software

manual

18 / 103

75%

15 PostgreSQL PHP Generator Help

2 Getting started

Connection properties
Set the connection parameters for the c
with.

Script connection properties
Specify here connection parameters for Postgre
example, if your webserver and PostgreSQL se
Host as localhost.

Projects
When working with a project, all the sessio
and may be edited if necessary. To run a w
Project on the first wizard step and enter
projects are also available from this popup
Projects.

Connection properties

● I can connect to the server directly or via SSH tunneling

Configure SSH options

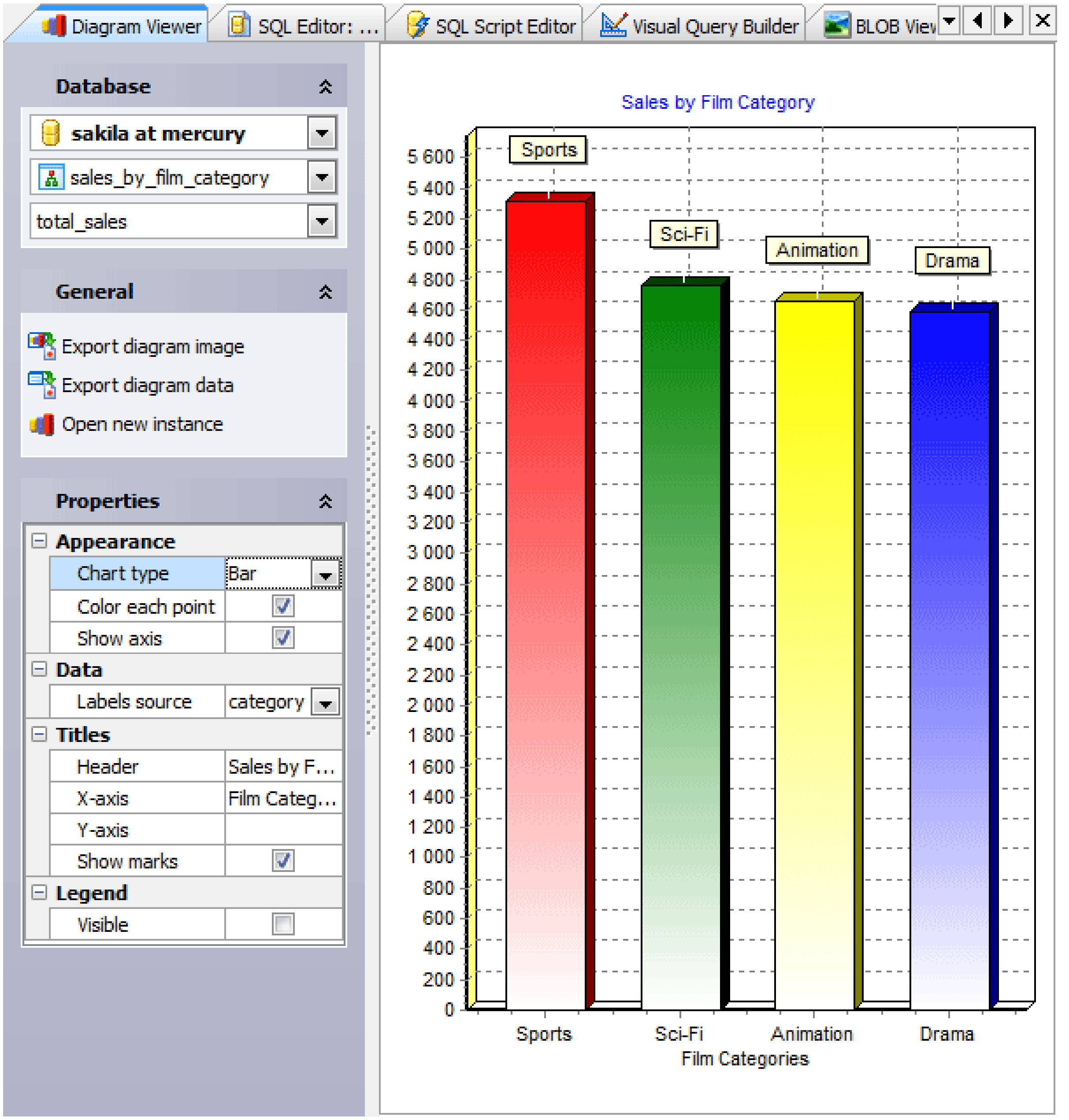
8.50 x 11.00 in

id	full_name	description_id
1	PostgreSQL PHP Generator	1
2	Code Factory for MySQL	3
3	SQLite DataWizard	2
4	MS SQL Maestro	4

9.8 Diagram Viewer

Diagram Viewer is a tool for representing data from a table or a query as a diagram in various ways. This means you can build a diagram represented as bars, lines, areas, points or pies, colored or not, with axis visible or not; specify axis labels source, diagram header and more. The Diagram Viewer also has the Export diagram image 387 and the Export diagram data features implemented, with a lot of formats supported.

- Customizing diagram options 386
- Exporting diagram as a graphical image 387

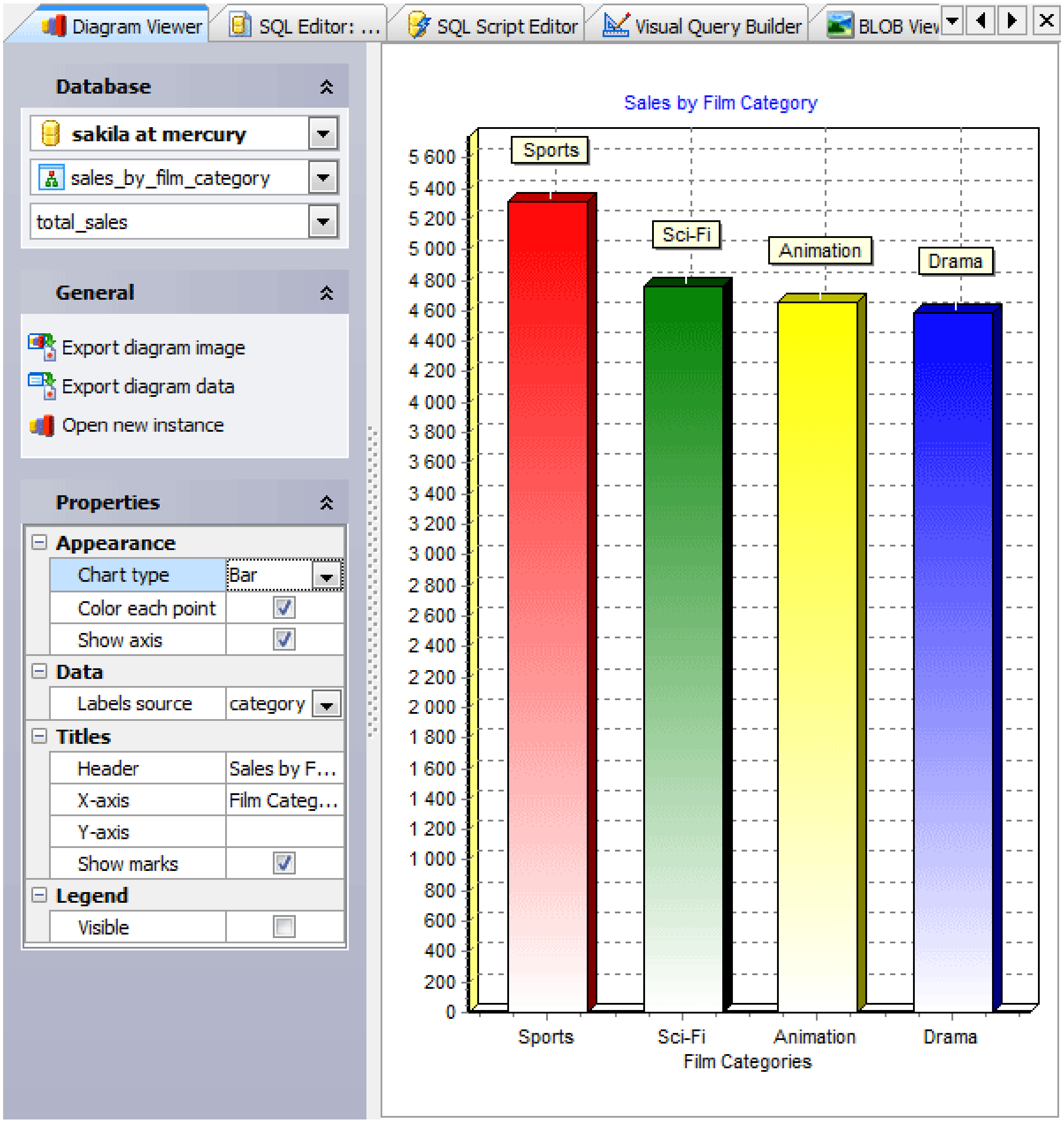


Database: sakila at mercury
sales_by_film_category
total_sales

Appearance	
Chart type	Bar
Color each point	☑
Show axis	☑
Data	
Labels source	category
Titles	
Header	Sales by F...
X-axis	Film Categ...
Y-axis	
Show marks	☑
Legend	
Visible	☐

9.8.1 Customizing diagram properties

To build a diagram in Diagram Viewer, you should select the source field(s) to be represented in the diagram first. Only numeric types of fields can be used in the diagram, and each selected field corresponds to a separated diagram series. Fields are selected by checking items in the third combo box from the top in the Database group of the Navigation Bar. If the combo box is empty then either data source is not yet selected or it contains no numeric fields.



Database

sakila at mercury

sales_by_film_category

total_sales

Properties

Appearance

Chart type: Bar

Color each point: ☑

Show axis: ☑

Data

Labels source: category

Titles

Header: Sales by F...

X-axis: Film Categ...

Y-axis:

Show marks: ☑

Legend

Visible: ☐

Diagram Viewer provides a special control for customizing the diagram properties. This control is located in the Properties group of the Navigation Bar and consists of four separate subgroups:

Appearance
Contains properties responsible for major diagram appearance:

- Chart type - defines a way of how the diagram will be represented: as bars, lines, areas, points, pies, or fast lines
- Color each points - if checked, each bar, point, line or sector of the diagram has an individual color; if not checked, all the points are colored red
- Show axis - defines if the diagram has the axis and background grid or not

Data
Contains the Labels source property which allows you to specify the field for X-axis labels as well as for diagram point marks .

Titles
Contains properties for defining titles for different parts of the diagram:

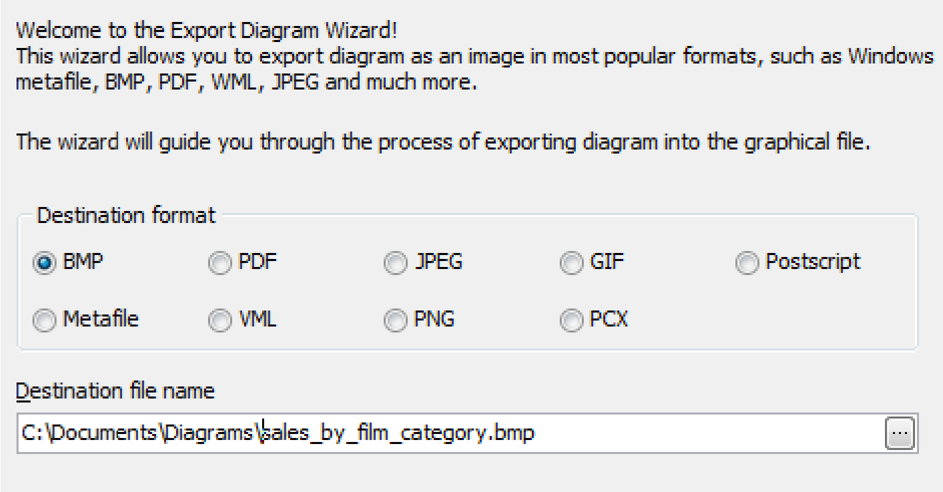
- Header - defines the title appeared on the top of the diagram
- X-axis and Y-axis - define the titles for diagram axis
- Show marks - defines if the diagram point marks are visible or not

Legend
The only Visible property of this subgroup specifies whether the legend rectangle should be represented on the right side of the diagram or not.

9.8.2 Exporting diagram image

Diagram Viewer provides an ability to export current diagram to a file as graphical image. This ability is constituted in Export Diagram Wizard which can be invoked by the Export diagram image item of the Navigation Bar.

Select the desired graphical format in the Destination format radio group and specify the file name in the Destination file name box.



Welcome to the Export Diagram Wizard!
This wizard allows you to export diagram as an image in most popular formats, such as Windows metafile, BMP, PDF, WML, JPEG and much more.

The wizard will guide you through the process of exporting diagram into the graphical file.

Destination format

● BMP ○ PDF ○ JPEG ○ GIF ○ Postscript
○ Metafile ○ VML ○ PNG ○ PCX

Destination file name
C:\Documents\Diagrams\sales_by_film_category.bmp

Set the destination width and height by the corresponding spin edits. Check or uncheck the Keep aspect ratio option to keep the image ratio for exported image or not. Check the Open exported image in associated program option to view the image after the export is done.

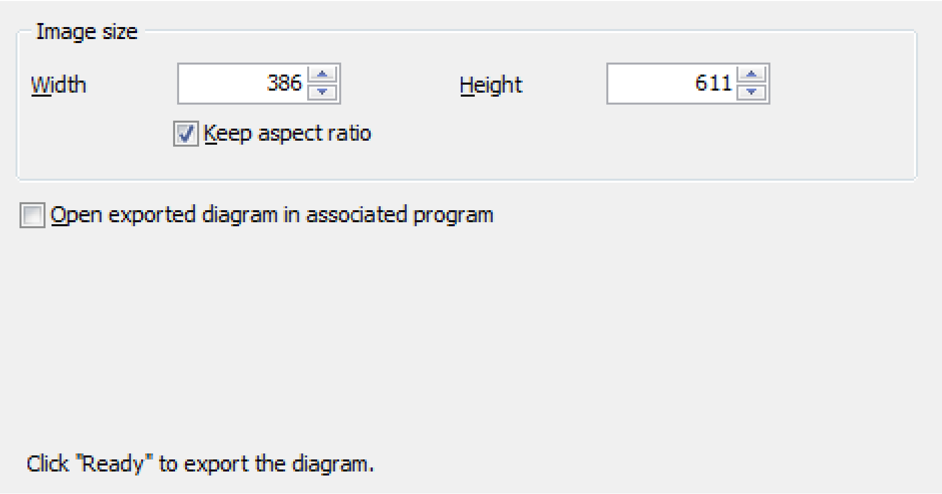


Image size

Width 386

Height 611

☑ Keep aspect ratio

☐ Open exported diagram in associated program

Click "Ready" to export the diagram.

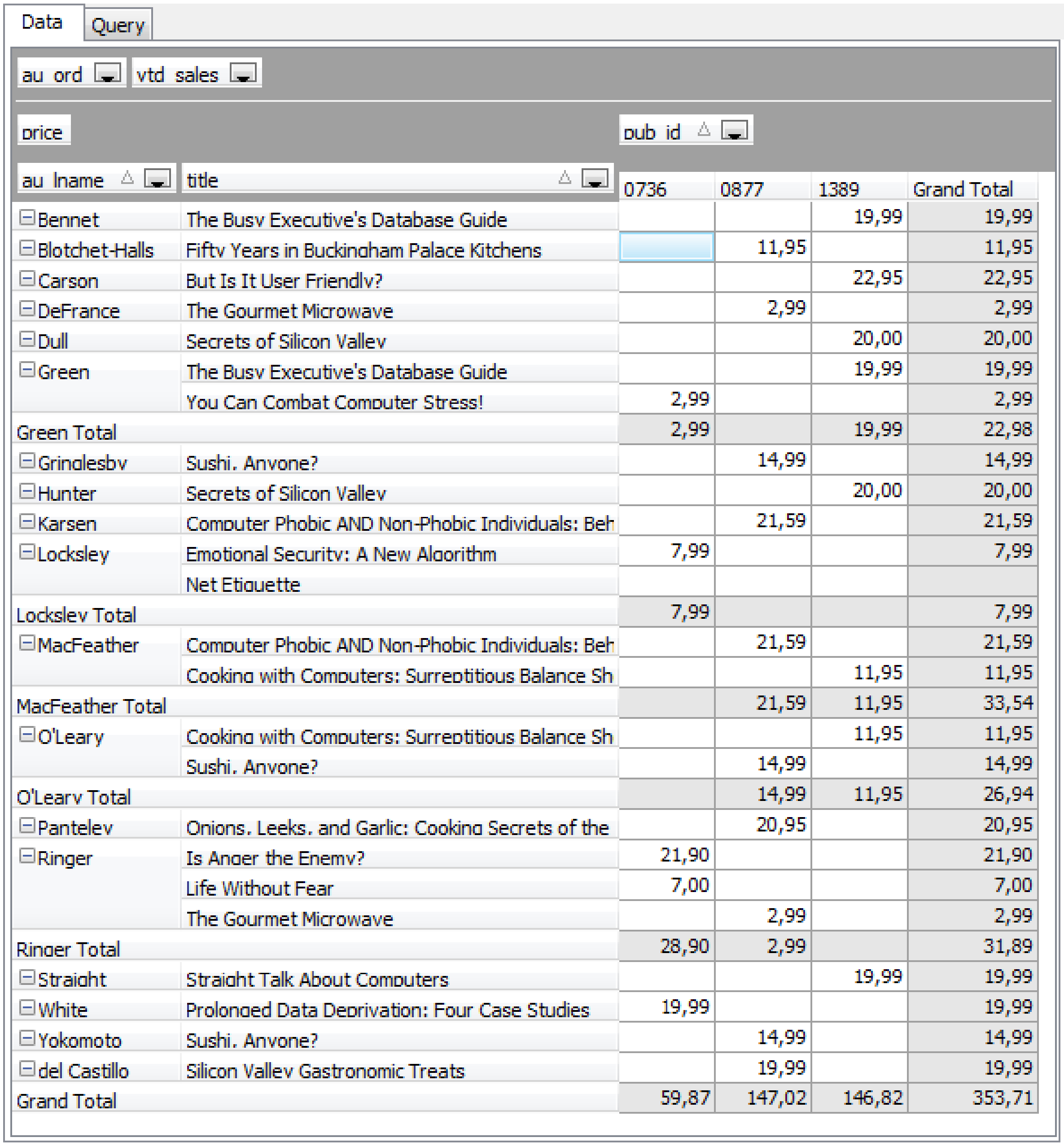
9.9 Data Analysis

Data Analysis is a tool to define a multidimensional model with analytic calculations to analyze information also called OLAP cube. Such cubes could effectively be re-oriented. So the tool allows you to view data in various ways, such as displaying all the cities down the page and all the products across a page and then immediately view it in another way. Because this re-orientation involves re-summarizing very large amounts of data, this new view of the data has to be generated efficiently to avoid wasting the analyst's time, i.e. within seconds, rather than the hours a relational database and conventional report-writer might have taken. It allows you to focus on business rules rather than creating dozens and dozens of reports. To run Data Analysis, choose Tools | Data Analysis main menu item.

To get an OLAP cube:

- Input SELECT query 390 in the Query window or load it from the .cub file.
- Manage report data 391 in the Data window.

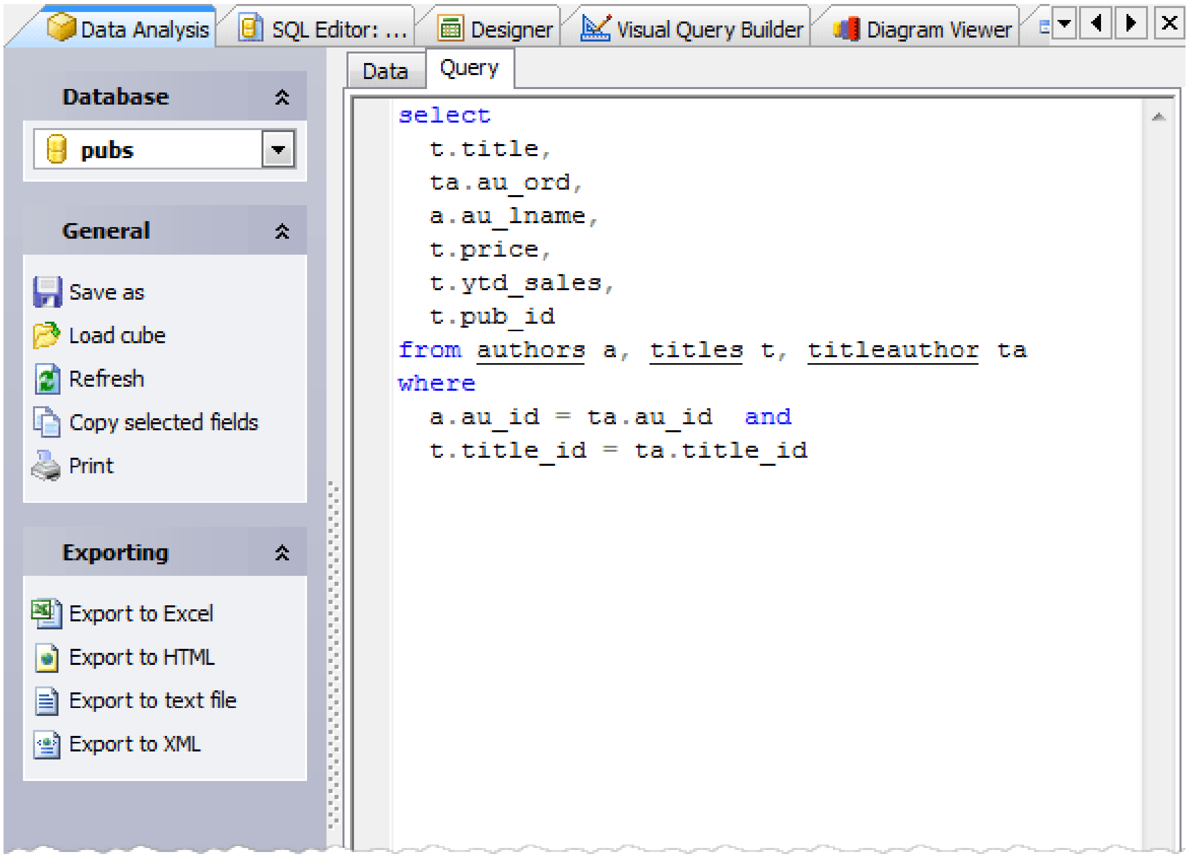
The Data Analysis tool allows you to save the created OLAP cube to .cub file, print it, or export it to Excel, HTML, XML, and text file.



au_lname	title	0736	0877	1389	Grand Total
Bennet	The Busy Executive's Database Guide			19,99	19,99
Blotchet-Halls	Fifty Years in Buckingham Palace Kitchens		11,95		11,95
Carson	But Is It User Friendly?			22,95	22,95
DeFrance	The Gourmet Microwave		2,99		2,99
Dull	Secrets of Silicon Valley			20,00	20,00
Green	The Busy Executive's Database Guide			19,99	19,99
	You Can Combat Computer Stress!	2,99			2,99
Green Total		2,99		19,99	22,98
Gringlesby	Sushi, Anyone?		14,99		14,99
Hunter	Secrets of Silicon Valley			20,00	20,00
Karsen	Computer Phobic AND Non-Phobic Individuals: Beh		21,59		21,59
Locksley	Emotional Security: A New Algorithm	7,99			7,99
	Net Etiquette				
Locksley Total		7,99			7,99
MacFeather	Computer Phobic AND Non-Phobic Individuals: Beh		21,59		21,59
	Cooking with Computers: Surreptitious Balance Sh			11,95	11,95
MacFeather Total			21,59	11,95	33,54
O'Leary	Cooking with Computers: Surreptitious Balance Sh			11,95	11,95
	Sushi, Anyone?		14,99		14,99
O'Leary Total			14,99	11,95	26,94
Panteley	Onions, Leeks, and Garlic: Cooking Secrets of the		20,95		20,95
Ringer	Is Anger the Enemy?	21,90			21,90
	Life Without Fear	7,00			7,00
	The Gourmet Microwave		2,99		2,99
Ringer Total		28,90	2,99		31,89
Straight	Straight Talk About Computers			19,99	19,99
White	Prolonged Data Deprivation: Four Case Studies	19,99			19,99
Yokomoto	Sushi, Anyone?		14,99		14,99
del Castillo	Silicon Valley Gastronomic Treats		19,99		19,99
Grand Total		59,87	147,02	146,82	353,71

9.9.1 Input SELECT query

To get an OLAP cube 391, enter SELECT query as a snowflake schema, represented by centralized fact tables (with numeric data) which are connected to multiple dimensions (the numeric data to be categorized by). Input the query text in the SQL Editor area directly or use "drag-n-drop" operation SQL Editor or Visual Query Builder areas and the Query tab of Data Analisis.

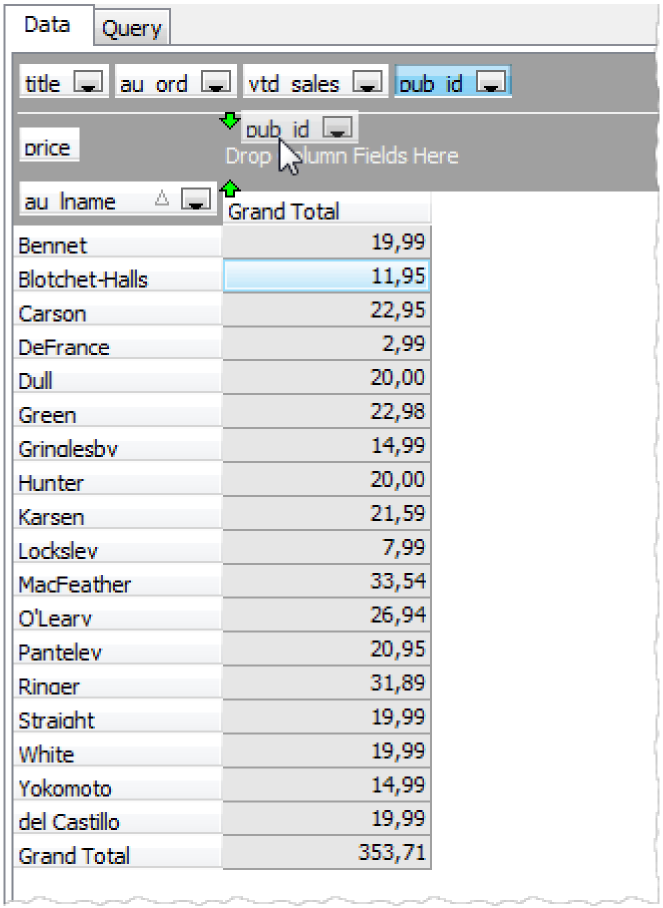


```sql
select
  t.title,
  ta.au_ord,
  a.au_lname,
  t.price,
  t.ytd_sales,
  t.pub_id
from authors a, titles t, titleauthor ta
where
  a.au_id = ta.au_id  and
  t.title_id = ta.title_id
```

9.9.2 Managing report data

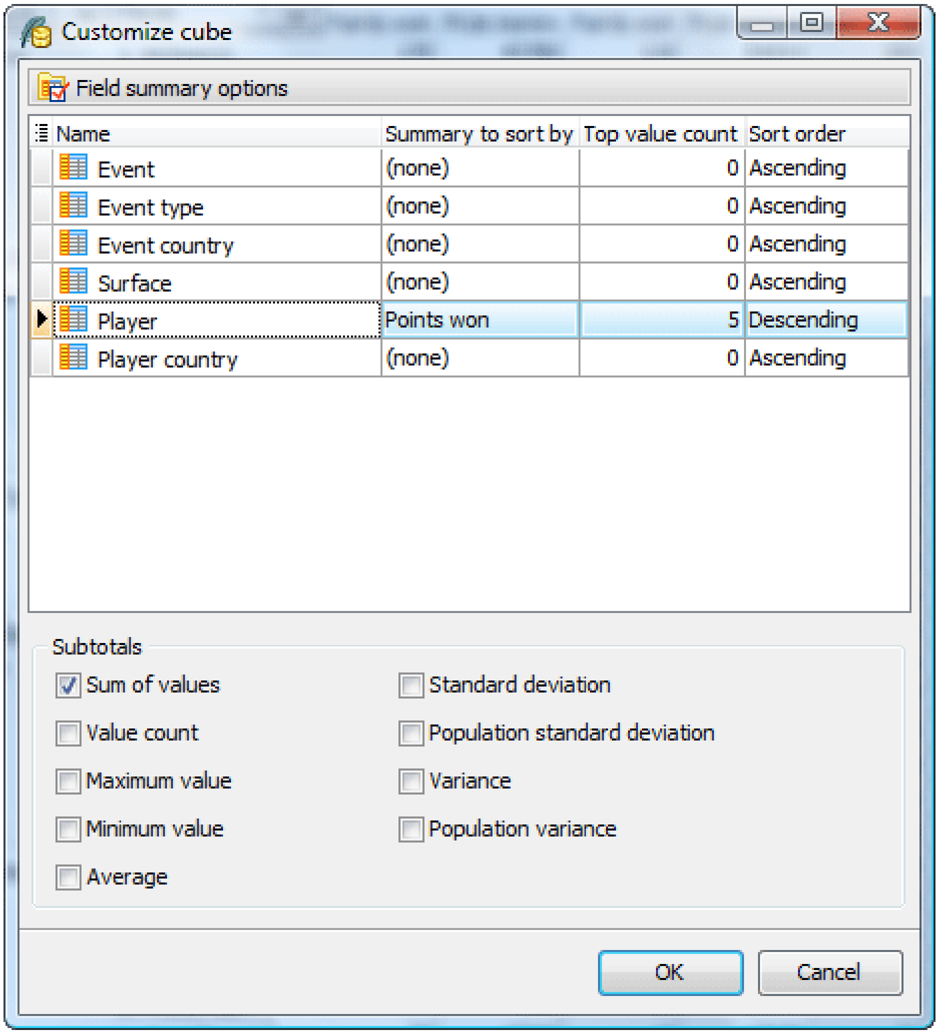
The Data tab allows you to manipulate the created OLAP cube appearance. At the beginning all the query 390 columns are arranged at the top of the tab. Put them according to your business rules: drag numeric columns to be filtered and summarized corresponding to the chosen columns and rows to the Data Fields area; place necessary columns to Column Fields / Row Fields areas respectively.

Note: Use for the Data Fields area only numerical columns.



au_lname	Grand Total
Bennet	19,99
Blotchet-Halls	11,95
Carson	22,95
DeFrance	2,99
Dull	20,00
Green	22,98
Gringlesby	14,99
Hunter	20,00
Karsen	21,59
Locksley	7,99
MacFeather	33,54
O'Leary	26,94
Panteley	20,95
Ringer	31,89
Straight	19,99
White	19,99
Yokomoto	14,99
del Castillo	19,99
Grand Total	353,71

To set the aggregates calculated on the numeric columns, use the Customize cub window opened with the corresponding link at the Navigation bar. The window provides you also with an ability to specify columns the summary to be sorted by, the sort order and the max number of records represented in grid.



Name	Summary to sort by	Top value count	Sort order
Event	(none)	0	Ascending
Event type	(none)	0	Ascending
Event country	(none)	0	Ascending
Surface	(none)	0	Ascending
Player	Points won	5	Descending
Player country	(none)	0	Ascending

Subtotals

☑ Sum of values
☐ Value count
☐ Maximum value
☐ Minimum value
☐ Average
☐ Standard deviation
☐ Population standard deviation
☐ Variance
☐ Population variance

9.10 Report Designer

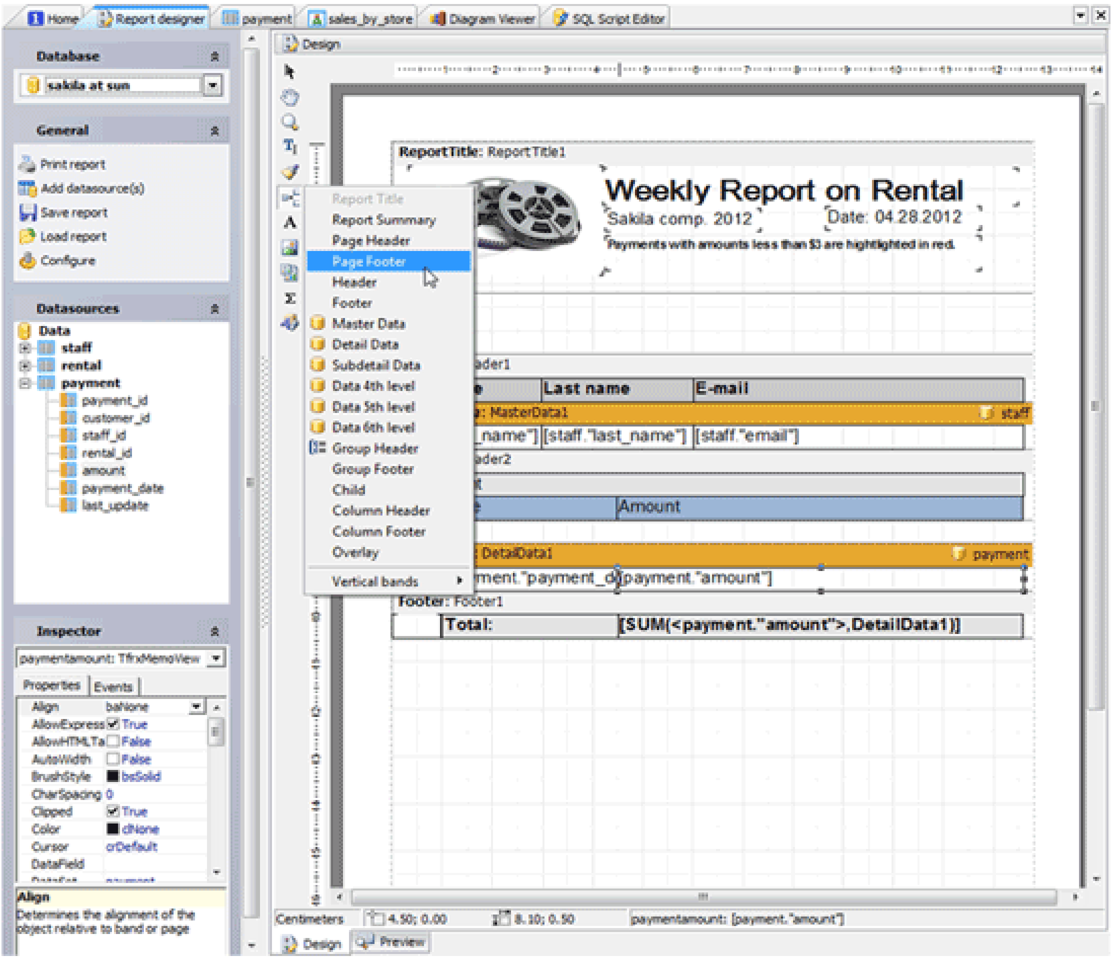
Report Designer allows you to create database reports, define reports appearance in your own style, equip it with master-detail data views, aggregate functions, and images and control the result with the ability of simultaneous previewing. To run Report Designer, choose Tools | Report Designer main menu item.

To create a report, you need to:

- specify data sources 396 for the data to be used in the report;
- add all necessary objects 396 to the report template;
- set the objects' format within the Inspector window 398.

The prepared report pages are available immediately at the Preview window where you can browse it, save to it an .smr file, or print.

Report Designer in Action

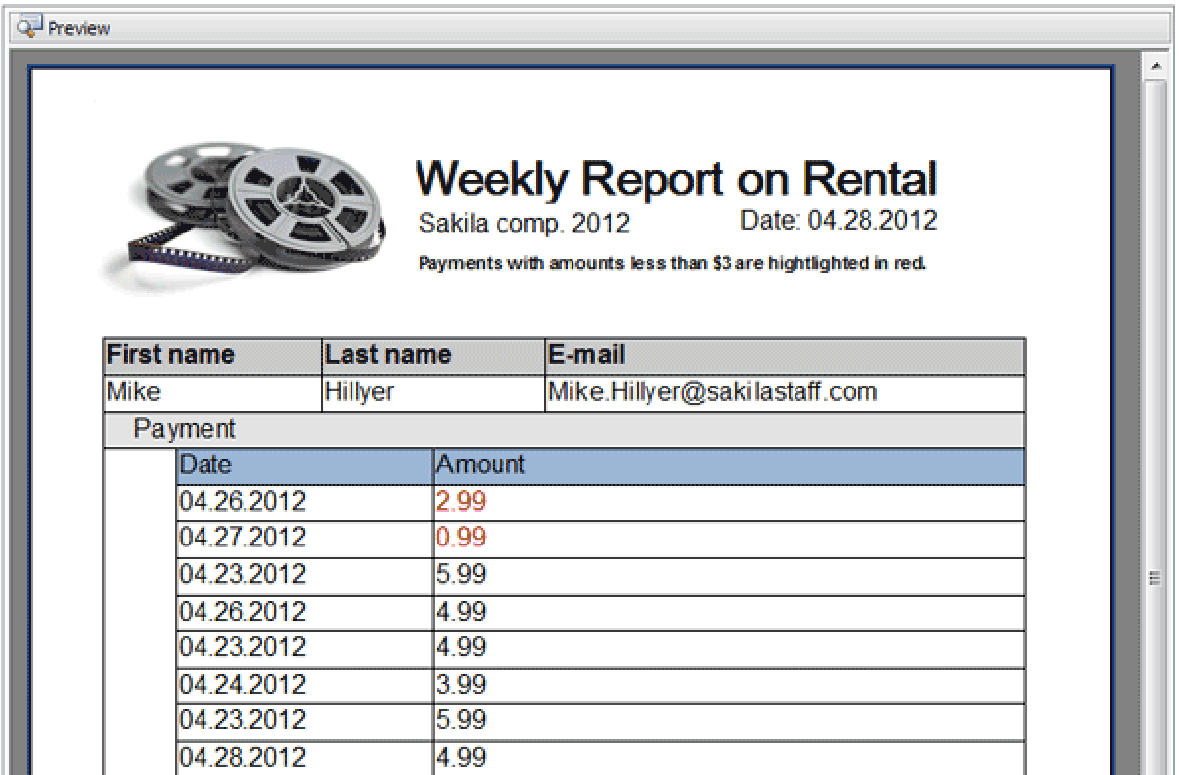


paymentamount: [payment."amount"]

Align bsNone
AllowExpress ☑ True
AllowHTMLTa ☐ False
AutoWidth ☐ False
BrushStyle bsSolid
CharSpacing 0
Clipped ☑ True
Color clNone
Cursor crDefault
DataField

Align
Determines the alignment of the object relative to band or page

Report Preview



Weekly Report on Rental

Sakila comp. 2012 Date: 04.28.2012

Payments with amounts less than $3 are hightlighted in red.

First name	Last name	E-mail
Mike	Hillyer	Mike.Hillyer@sakilastaff.com

Payment

Date	Amount
04.26.2012	2.99
04.27.2012	0.99
04.23.2012	5.99
04.26.2012	4.99
04.23.2012	4.99
04.24.2012	3.99
04.23.2012	5.99
04.28.2012	4.99

9.10.1 Designer Tools and Objects

A blank report is presented as a paper page. At any place on the page, a user is able to add objects, which can display different information (such as text and/or graphics), as well as to define report's appearance. There is a possibility to use rulers and a grid with a specified size in the Design tab. To enable/disable these options, follow the Configure link at the Navigation Bar and check the corresponding boxes.

Datasources
To use content of a table (view) column data in a report,
- check the necessary database is selected as Database at the Navigation Bar;
- drag the table which data to be used in the report to the Datasources pane at the Navigation Bar;
- drag the necessary column from this pane and drop it to the necessary location on a report page.

Designer tools:
Select tool
The standard tool to select objects, modify their sizes, etc.

Hand tool
The tool allows dragging a report page.

Zoom tool
When the button is pressed, clicking on the left button doubles the zoom (adds 100%), while clicking the right one zooms out by 100%. When holding the left mouse button while dragging,the selected area would be zoomed.

Edit text tool
Clicking on the text object allows editing its contents right on the report page. If you hold the left mouse button when moving the cursor, the text object appears in the selected place, and then its editor launches.

Copy format tool
The button becomes enabled when the text object is selected. When clicking on the text object with the left button, it copies formatting, which has the previously selected text object, into the object.

Available objects:
Band objects allow to specify where, when, and how to display data and information in reports. Bands are used for logically placing the objects it contains at a location on the output page. Insert Band adds an area with definite behaviour according to it's type such basic bands as Header, Footer, Title, and Summary, and databands whose allow to print data from database tables such as Master Data, Detail Data, etc.

Text object displays one or several text lines within the rectangular area.

Picture object displays a graphic file in BMP, JPEG, ICO, WMF, or EMF format.

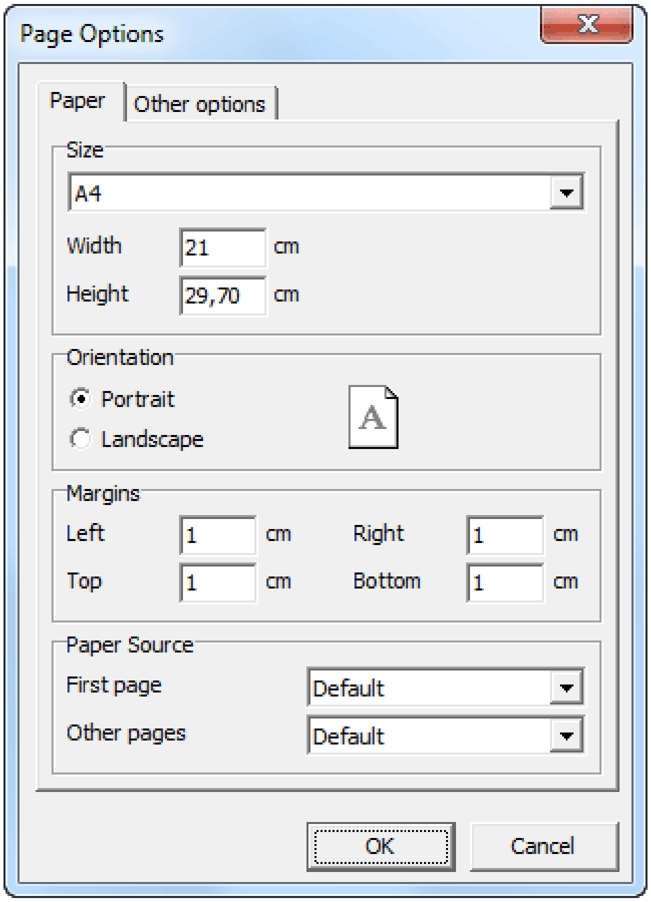
Subreport object allows inserting an additional report design page inside the basic one.

System text displays service information (date, time, page number, etc), as well as aggregate values.

Draw object represents different geometrical figures (diagonal line, rectangle, rounded rectangle, ellipse, triangle, and diamond).

Page options

This dialogue allows you to set the page settings of the prepared. To invoke the window, use the Edit... link of the page blank space popup menu. The dialogue has two pages: Paper and Other options. On the Paper page, you can select size and alignment of paper, as well as set margins. In Paper source drop-down lists you can select a printer tray for the first page and the rest of the report pages.



Size
A4

Width 21 cm

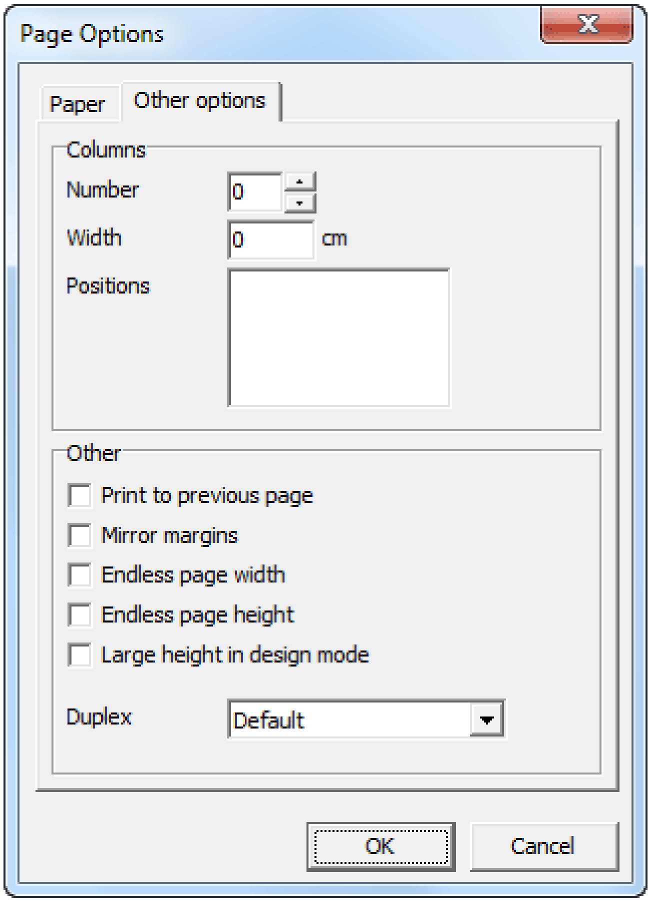
Height 29,70 cm

Orientation
● Portrait
○ Landscape

Margins
Left 1 cm
Right 1 cm
Top 1 cm
Bottom 1 cm

Paper Source
First page Default
Other pages Default

On Other Options you can set the number of columns for multi-column reports' printing. The current settings are displayed in the designer.



Columns

Number: 0

Width: 0 cm

Positions

Other

☐ Print to previous page

☐ Mirror margins

☐ Endless page width

☐ Endless page height

☐ Large height in design mode

Duplex: Default

The Print to previous page flag allows you to print pages, beginning from blank space of the previous page. This option can be used in case when a report template consists of several pages or when printing batch (composite) reports.

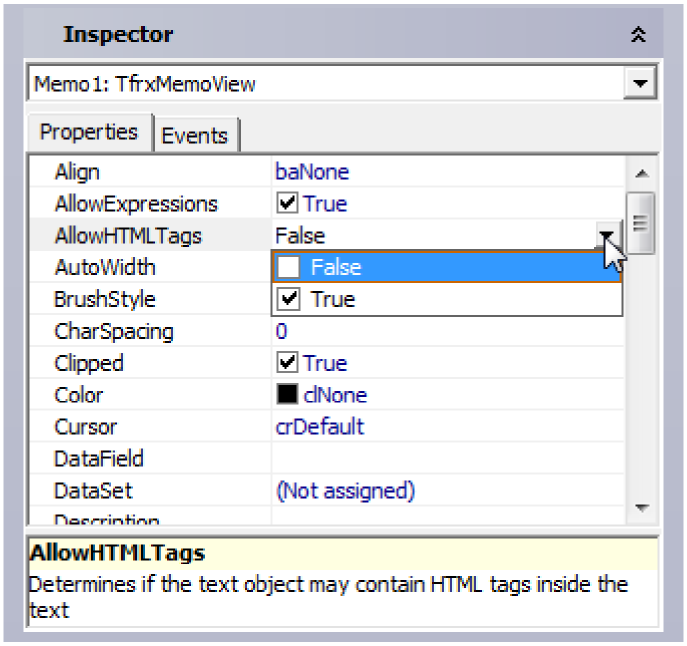
The Mirror margins option switches right and left margins of page for even pages during previewing or printing a report.

The Endless page width & height option increases page's sizes depending on number of data records on the page (when running a report). In this case you will see one big page in the preview window instead of several standard size pages.

The Large height in design mode option increases page’s height several times more. This feature can be useful if many bands are located in the page, and must be used when working with the overlay band. This only effects the page height in design mode.

9.10.2 Object Inspector

Object Inspector pane allows you to specify the appearance of each report object in detail. To setup object properties, select it at the Design area or select it from the popup menu at the top of the pane. Now all the properties of the object are available for editing. The most of properties are provided by a set of available values. The description of the selected option is displayed at the bottom of the pane.



Memo1: TfrxMemoView

Property	Value
Align	baNone
AllowExpressions	☑ True
AllowHTMLTags	False
AutoWidth	
BrushStyle	
CharSpacing	0
Clipped	☑ True
Color	■ clNone
Cursor	crDefault
DataField	
DataSet	(Not assigned)
Description	

AllowHTMLTags
Determines if the text object may contain HTML tags inside the text

Below you can find a brief description of several options.

Align - set here the align option of the object according to the list.

AllowExpressions - enables the ability to display not only a static text, but expressions as well.

AllowHTMLTags - Enables using some simple HTML tags inside the text of an object. This option is disabled by default.Here is the list of supported tags:
<b> - bold text;
<i> - text in italic;
<u> - underlined text;
<sub> - subscript;
<sup> - superscript;
<font color> - font color;
<nowrap> - text which does not get broken up when using WordWrap, but gets transported wholly.

Font: there are abilities to specify the charset, font color, font name, and font size, and also set the bold, italic, underline, strike out attributes.

Frame: You can set as the color, the style and the shadow for all the frame, as well as for each frame line.

BrushStyle – type of object filling.

CharSpacing – space between symbols in pixels.

GapX, GapY – text indents from object's left and top boundaries (in pixels).

LineSpacing – space between lines (in pixels).

ParagraphGap – the first paragraph line indent (in pixels).

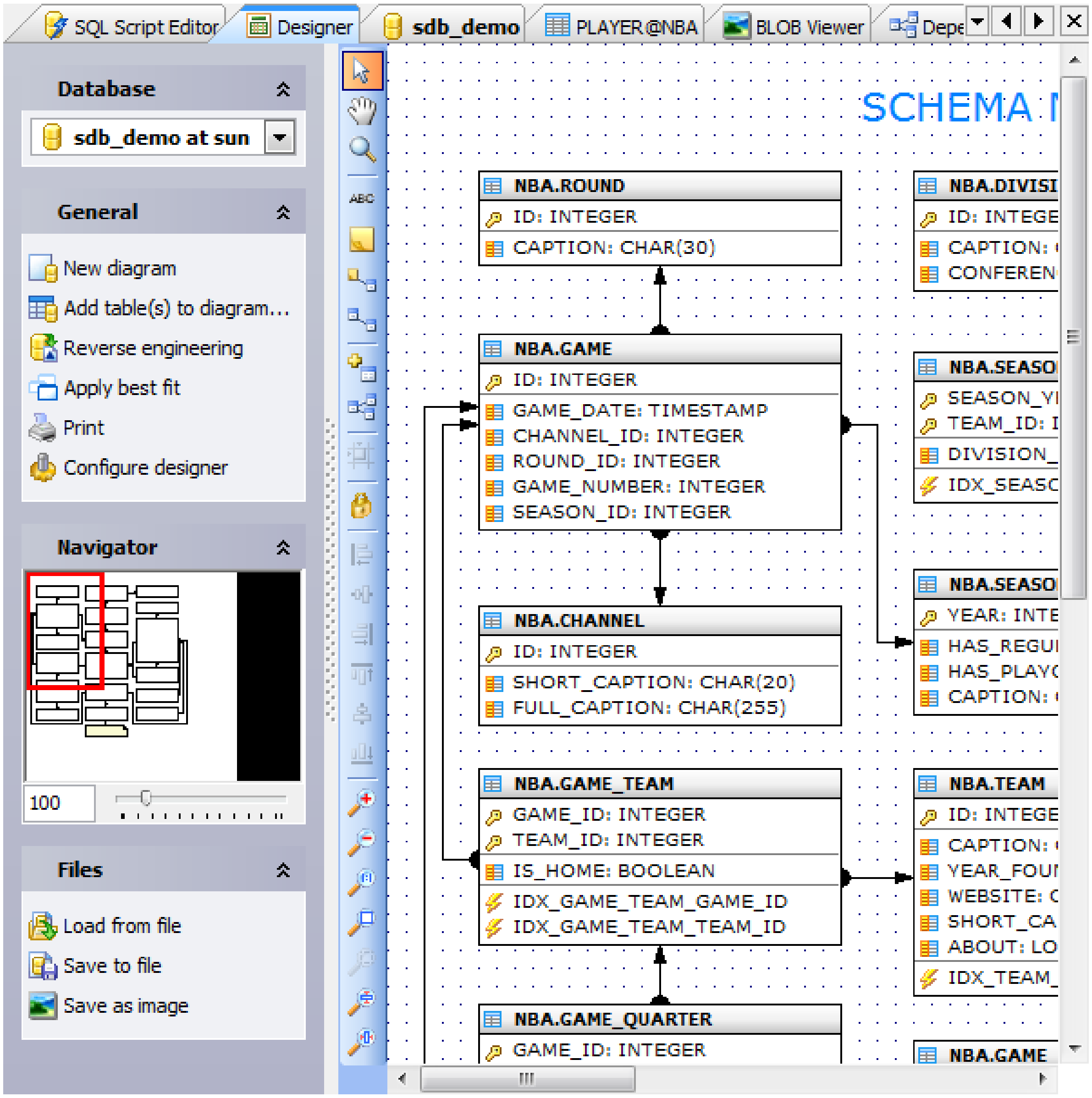
Rotation - specify the angle of the text rotation in the field.

WordWrap - if it is disabled, the long lines will be simply cut off.

9.11 Schema Designer

Schema Designer allows you to create physical Entity Relationship Diagram that will represent objects in your Microsoft SQL database. A diagram represents the tables of your database and the relationships between them. The tool is intended for reverse engineering and database modification in an easy and powerful way. It helps you to simplify database maintenance.

A diagram of your database can help you define operational aspects of your application logic that you might otherwise overlook. Also, a well-defined data diagram that accurately represents your tasks can be helpful in orienting employees to goals and operations. The data diagram can also serve as an invaluable communications tool for both internal and external constituents.



Below you can find answers for the following questions:

- How can I add a table to a diagram? 402
- How can I add a relationship between tables? 402
- How can I delete a diagram object? 402
- How can I work with diagram objects? 403

See also: Designer Navigation bar 403, Schema Designer Toolbox 403.

Adding a table

You can add an existing Microsoft SQL table to the diagram using popup menu in the working area, or with the corresponding link on the Navigation bar 403.

To create a new table, use the appropriate item of the popup menu in the working area. The table will be created in the current database.

Tables also may be dragged on the diagram from Explorer and the similar to Explorer tools like Object Manager and Object Browser.

Moreover, Designer provides you with a possibility to represent all the tables and relationships existing in the database automatically (the Reverse engineering link of the Navigation bar). At that the database contents will be represented on a diagram in the most compact and vivid manner.

All the diagram objects are available for editing. Just double click the object (table or relationship) to view/edit its properties within the corresponding editor.

Adding a relationship between tables

At adding to a diagram tables that reference on each other, the relationships between tables are represented automatically. The Schema Designer tool also allows you to add new foreign keys to the diagram tables. Thereto you can do the following.

Select a table (child table)

- Use the Create new... item of the popup menu to launch Foreign Key Properties 90 window.
- Specify there properties of the relationship been created.

Moreover you can add a reference graphically:

- Choose the Create relation tool on the Toolbox 403. Your mouse cursor will change its appearance.
- Then click on the table (child table) that will have foreign key and then click on the second table (parent table) whose primary key will be referred by the new foreign key.
- Specify properties of the relationship been created in the Foreign Key Properties 90 window.

With the Create new… item of the popup menu you can also add a new field, an index, a trigger, etc. to the selected table. For more information about object properties see: Field Editor 83, Index Editor 87, Create Trigger Wizard 99.

Deleting of the diagram objects

To hide a table (several tables) or a relationship between tables, select the objects and click Remove selection link of the popup menu or Navigation bar. You can also use the **Del** key for this purpose.

It’s also possible to physically delete a table/foreign key from the database: just select the object to delete and use the appropriate item of the popup menu.

Editing of a diagram appearance
Movement of a table/several tables along the diagram is realized with dragging or pressing **Ctrl**+arrows. You can use **Shift**+arrows to change width/height of table/several tables representation.

Designer also allows you to edit shape of the line representing foreign key relations/ logical relations. In order to break the line you should

- Select the relationship.
- Press **Ctrl** and click on the necessary line section to create a new node.
- Position the node by dragging.

You can also delete a node on the line. Thereto

- Select the relationship.
- Press **Alt** and click the node to delete. In that case the near nodes will be united by a straight line.

9.11.1 Designer Navigation Bar

The Navigation Bar of Schema Designer provides you the following opportunities:

Use the Database drop-down list to move around your Microsoft SQL databases.

There are also links for adding a New diagram or an existing table to diagram quickly.

Reverse engineering
The link provides you to create a new diagram with all the database tables and

Apply best fit
Use the link to dispose tables on the diagram in the most clear manner.

Remove selection
The link cancels current object selection.

Use Print to see the print preview of the diagram.

Certainly, it's possible to customize Schema Designer with Configure designer. For more information see Schema Designer Customization 437.

The Navigator part allows you to adjust the scale of the diagram and the position of the visible part.

Besides the Navigation bar allows you to Load a diagram from file, Save to file, and Save as image (Bitmap, GIF and JPEG formats are supported).

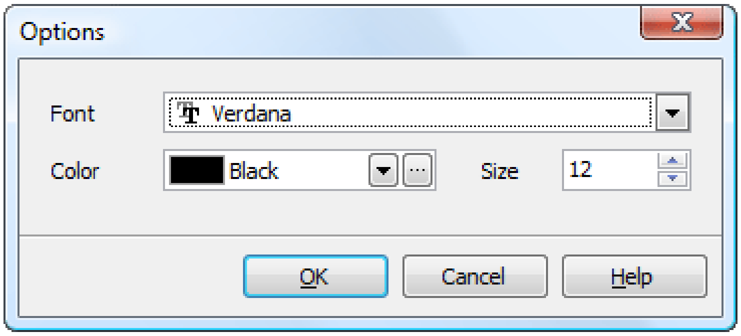
9.11.2 Schema Designer Toolbox

The toolbox is located on the left side of the Schema Designer's working area.

Move
The tool is intended for selection of diagram objects. Use the tool then click anywhere inside of the object. Double click opens the corresponding Object Editor.

To select multiple objects, use the tool then click and drag a selection rubber-band so that the rubber-band box encompasses the objects you want to select, and then release the mouse button.
To add objects to the list of already selected objects again, use the Move tool then click anywhere inside of the object holding the **Shift** button. To quick launch of the tool, use **M** shortcut.

Use Create text box to add title and comments on your diagram. Click on the necessary place and double-click on the appeared box to enter a text. You can also tune up the text font, color and size with Text options of the box popup menu. To quick launch of the tool, use **XX** shortcut.


Options

Font: Verdana

Color: Black

Size: 12

Moreover you can add notes and also links between them and diagram elements using Create note and Create link to note links. To quick launch of the tools, use **N** and **L** shortcuts accordingly.

Lock
The tool to locking/unlocking diagram objects. This feature prevents your diagram from unforeseen changes: when the diagram is locked, you can neither move/resize/delete existing objects nor add new ones.

Hand tool moves a diagram within its window. To quick launch of the tool, use **H** shortcut.

Zoom magnifies and reduces the view of a diagram. To zoom out, hold the Alt key. To quick launch of the tool, use **Z** shortcut.

There are also tools allowing to Create table and Create relation directly from the Designer. To quick launch of the tool, use **T** and **R** shortcut.

Below you can find toolset for aligning the selected objects by left and right edges, by horizontal and vertical centers, tops and bottoms.

Click the Zoom in button in the options bar to magnify to the next preset percentage.

When the image has reached its maximum magnification level, the command is dimmed.

Click the Zoom out button in the options bar to reduce to the previous preset percentage. When the image has reached its maximum reduction level, the command is dimmed.

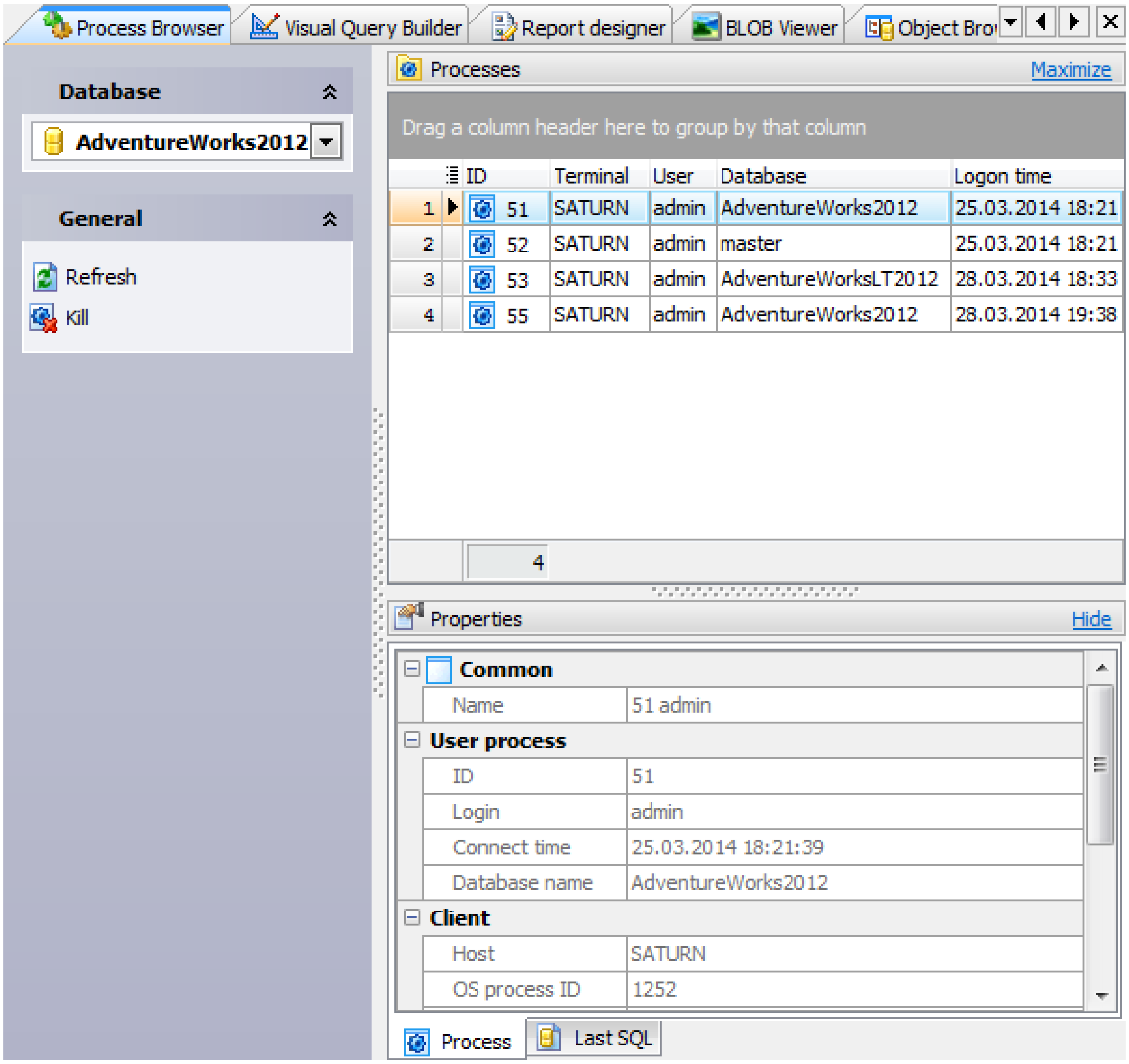
Click the Zoom 1:1 button to display a diagram at 100%.

Pay attention to the Fit diagram function, that pick-up properly scaling factor to display your diagram fully. For your convenience the Fit selected, Fit height, and Fit width were added.

9.12 Process Browser

Process Browser is very useful tool for DBAs who want to monitor the users' activity (in fact, there are potentially thousands of sessions in a database at any one time). You can view details for each session (such as login, connect time, database name, client host, last SQL statement executed and more) as well as group and filter processes by client host, connected user, database, client application, etc.

To access the Process Browser window, select the corresponding item from the Tools menu.



	ID	Terminal	User	Database	Logon time
1	51	SATURN	admin	AdventureWorks2012	25.03.2014 18:21
2	52	SATURN	admin	master	25.03.2014 18:21
3	53	SATURN	admin	AdventureWorksLT2012	28.03.2014 18:33
4	55	SATURN	admin	AdventureWorks2012	28.03.2014 19:38

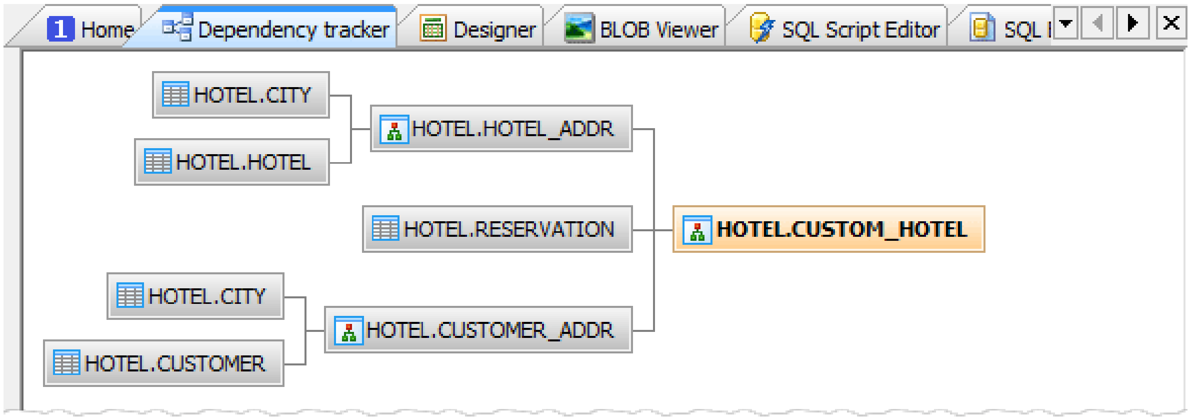
Common	
Name	51 admin
User process	
ID	51
Login	admin
Connect time	25.03.2014 18:21:39
Database name	AdventureWorks2012
Client	
Host	SATURN
OS process ID	1252

9.13 Dependency tracker

Dependency tracker is a tool to browse all-level dependencies of a schema object (table, view, function, etc). To display dependencies of an object, drag and drop it from the Explorer tree (or Object Manager, Object Browser) to the tracker's working area.

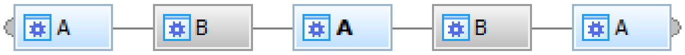
This tool allows you to see the way any database object is involved in the net of scheme dependencies. The selected object is displayed in a highlighted rectangle in the center of the working area. The right side of this area represents objects depending on the selected one. The left side represents objects on which the selected object depends.

To highlight all occurrences of an object in the diagram, click the appropriate rectangle. Double click an object to change the tracker focus to this object. Right click a rectangle to display a popup menu with common operations related to the appropriate object.



The recursive dependencies are marked with a semicircle. This means marked object depends on itself directly or via other objects.

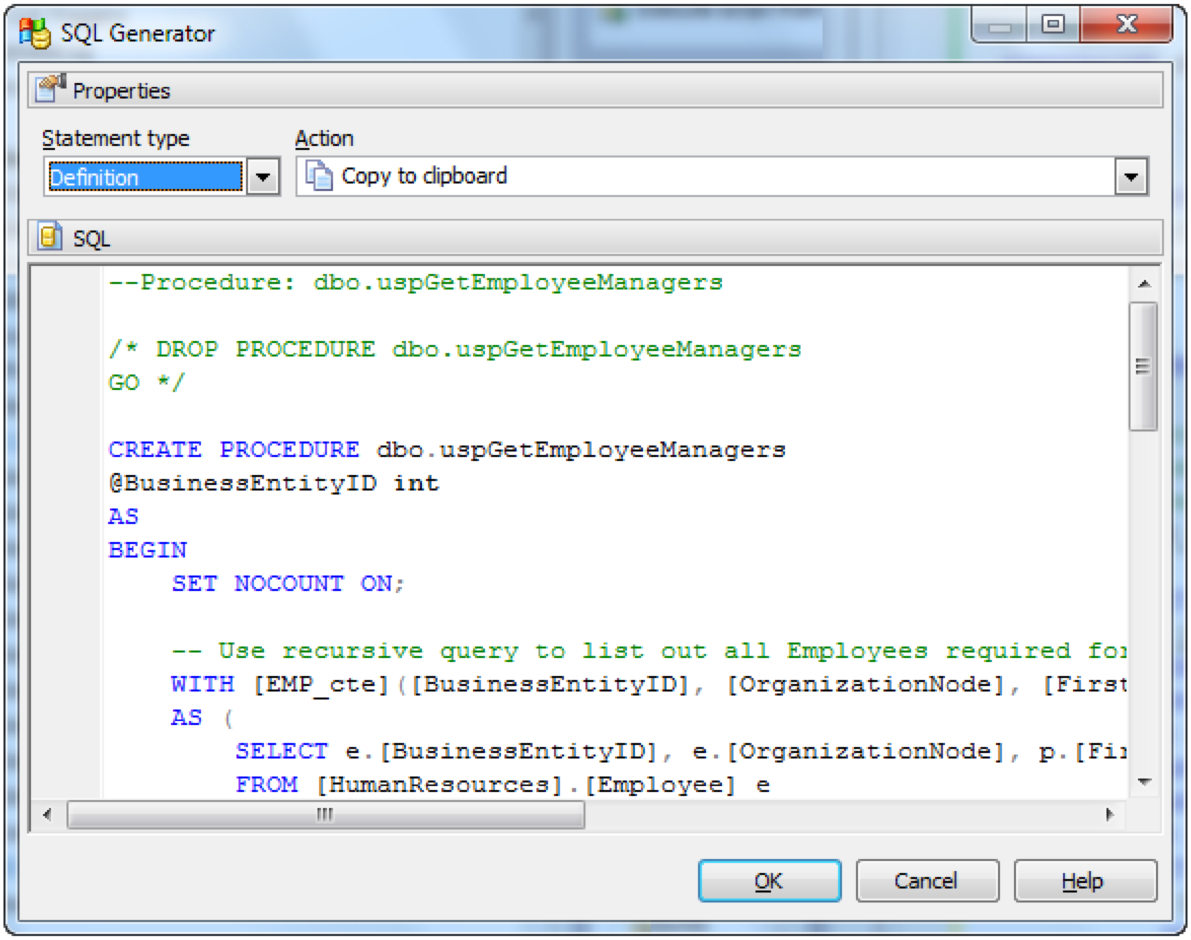
Example
Suppose we have two procedures A and B calling each other. The tracker displays these dependencies in the following way:



9.14 SQL Generator

Among other features MS SQL Maestro provides you with SQL Generator, a tool to create simple SQL statements. Just choose a database object, select statement type (Definition, Select, Insert, Update, or Delete) and the destination device (Clipboard, File, SQL Editor, SQL Script Editor).

The SQL Generator window can be invoked from the Explorer tree.



Statement type: Definition
Action: Copy to clipboard

```
--Procedure: dbo.uspGetEmployeeManagers

/* DROP PROCEDURE dbo.uspGetEmployeeManagers
GO */

CREATE PROCEDURE dbo.uspGetEmployeeManagers
@BusinessEntityID int
AS
BEGIN
    SET NOCOUNT ON;

    -- Use recursive query to list out all Employees required fo
    WITH [EMP_cte]([BusinessEntityID], [OrganizationNode], [First
    AS (
        SELECT e.[BusinessEntityID], e.[OrganizationNode], p.[Fi
        FROM [HumanResources].[Employee] e
```

9.15 DML procedures generation

MS SQL Maestro allows you to create DML (also known as CRUD) procedures automatically. CRUD is an acronym for the four essential database operations: Create, Read, Update, and Delete. The application designer has many choices for accomplishing the CRUD operations but the most efficient choice in terms of Microsoft SQL performance is to create a set of stored procedures to perform the operations.

The reasons for using DML Procedures instead of allowing ad hoc SQL statements are:

- The best possible performance

After the first use of each stored procedure, the plan for executing the procedure is cached in the server's procedure cache. For subsequent invocations of the stored procedure, the plan is reused. This avoids the parsing and optimization steps with their overhead.

- Removing of the SQL code from the other layers of the application

By removing the SQL statements from the application code, all the SQL can be kept in the database.

- Preventing of SQL injection attacks

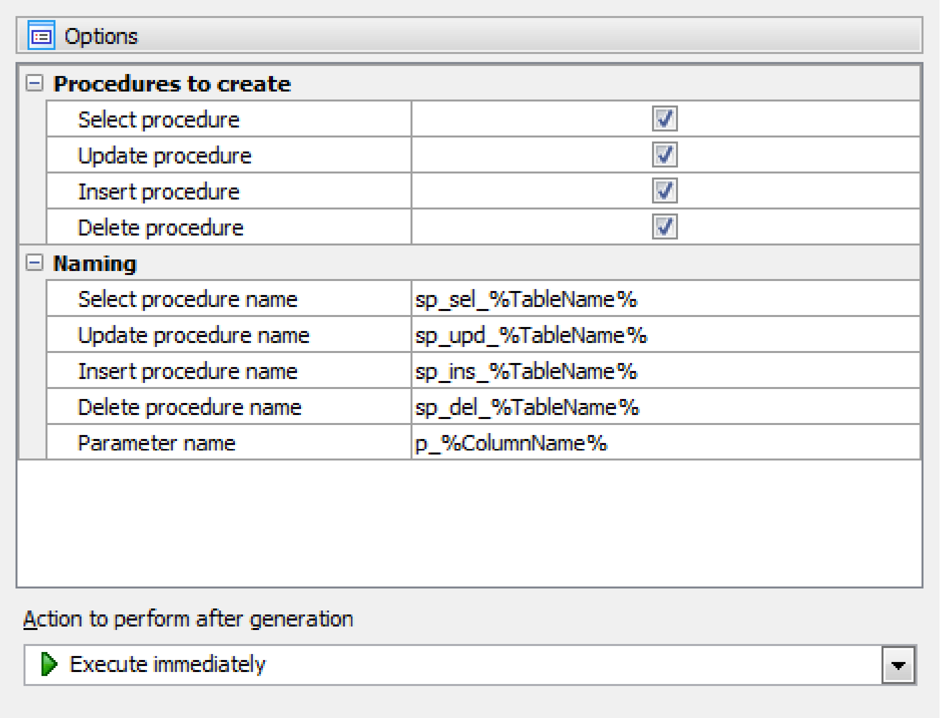
Anytime a client application uses string concatenation to create SQL statements, there is a possibility of a SQL injection attack. In short, these attacks involve clever entry of SQL in the data entry fields of an application in such a way that the SQL statements executed are different from the ones intended by the programmer. They require that the application developer is careless about not cleaning any user input to prevent the attack.

- Preventing of casual table browsing and modifications

If an application uses ad hoc SQL statements, the users of the application must have the required permissions on the database tables. Once they are given permission on the tables, they can work with them in any application that can read and manipulate the data such as Excel, Word and various report writers. Casual examination of the data and even updates that bypass the application's business rules become possible. Stored procedures have long been used to prevent casual browsing and updates. This is implemented by granting permission to execute the CRUD stored procedures to the users and revoking permission to access the tables directly.

To generate DML procedure,

- select the Object | Generate DML procedures... main menu item (to create procedures for several tables) or use the corresponding popup menu item of the table's node at the Explorer tree (to create procedures for one table).
- Specify tables the procedures will be created for (in case of several tables).
- Uncheck the operations the procedures will not be created for. By default the procedures are generated for inserting, reading, updating, and deleting of table data.
- Adjust templates of procedures names.
- Select the action to perform after the generation. The created definitions can be copied to Clipboard, saved to a file, sent to SQL Script Editor or executed immediately.



Procedures to create

Select procedure	☑
Update procedure	☑
Insert procedure	☑
Delete procedure	☑

Naming

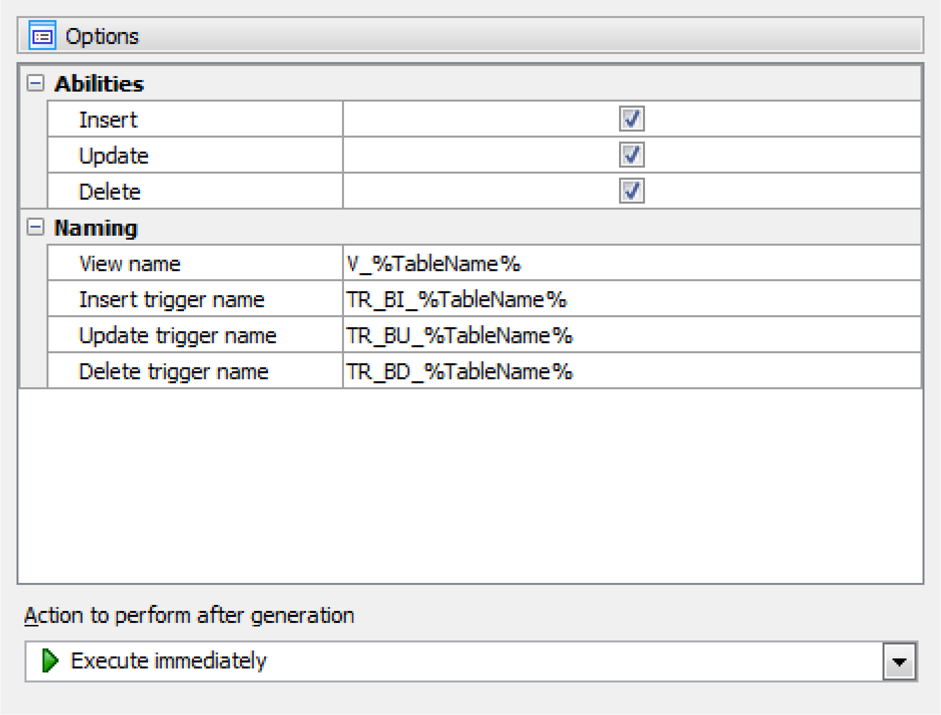
Select procedure name	sp_sel_%TableName%
Update procedure name	sp_upd_%TableName%
Insert procedure name	sp_ins_%TableName%
Delete procedure name	sp_del_%TableName%
Parameter name	p_%ColumnName%

Action to perform after generation: Execute immediately

9.16 Generation of updatable views

To generate updatable view,

- select the Object | Generate updatable views... main menu item (to create views for several tables) or use the corresponding popup menu item of the table's node at the Explorer tree (to create a view for one table).
- Specify tables the views will be created for (in case of several tables).
- Specify the abilities to be available on working with the view data. By default the views are generated for inserting, updating, and deleting of table data.
- Adjust the name templates of views and corresponding triggers.
- Select the action to perform after the generation. The created definitions can be copied to Clipboard, saved to a file, sent to SQL Script Editor or executed immediately.

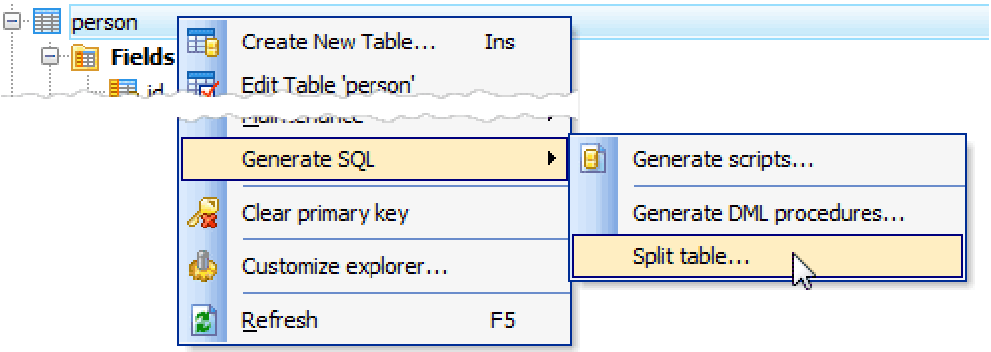


Abilities	
Insert	☑
Update	☑
Delete	☑
Naming	
View name	V_%TableName%
Insert trigger name	TR_BI_%TableName%
Update trigger name	TR_BU_%TableName%
Delete trigger name	TR_BD_%TableName%

Action to perform after generation: Execute immediately

9.17 Split table

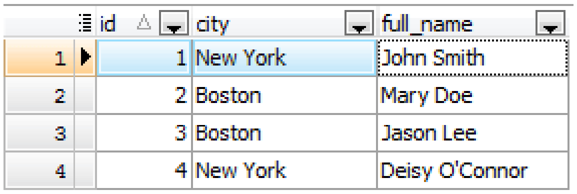
It's not an uncommon situation when new requirements arise, or when you need to enforce referential integrity on a set of columns, and the best decision is to split a table into two separate tables. MS SQL Maestro provides you with Split Table Wizard, a simple tool to generate a bunch of SQL scripts to modify the primary table, to create a secondary table with a primary key, and to transfer data from the primary table to the secondary one without duplicating of data. To invoke the wizard, follow the corresponding link of the Generate SQL section of popup menu of the selected table at the Explorer tree.



Let's see the wizard in action on the example of a table with the following SQL definition:

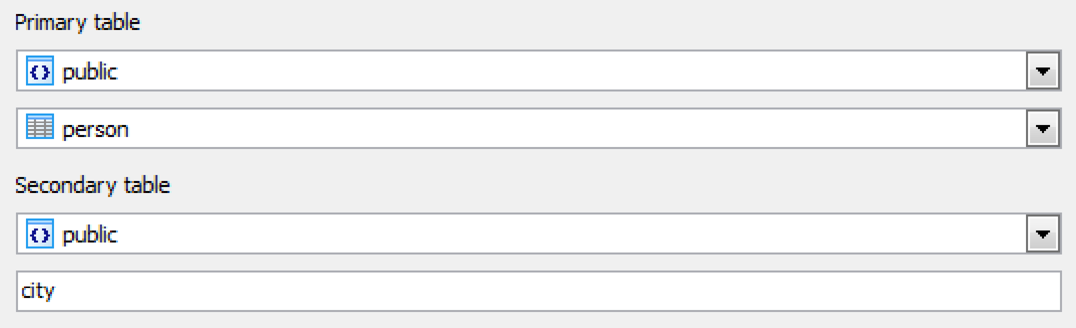
```
CREATE TABLE person (
  id          integer NOT NULL,
  city        varchar(30) NOT NULL,
  full_name   varchar(30) NOT NULL,
  /* Keys */
  CONSTRAINT  person_pkey
    PRIMARY KEY (id)
);
```

The table stores sample data:



	id	city	full_name
1	1	New York	John Smith
2	2	Boston	Mary Doe
3	3	Boston	Jason Lee
4	4	New York	Deisy O'Connor

To enforce the referential integrity, we specify 'city' as secondary table:



Primary table

public

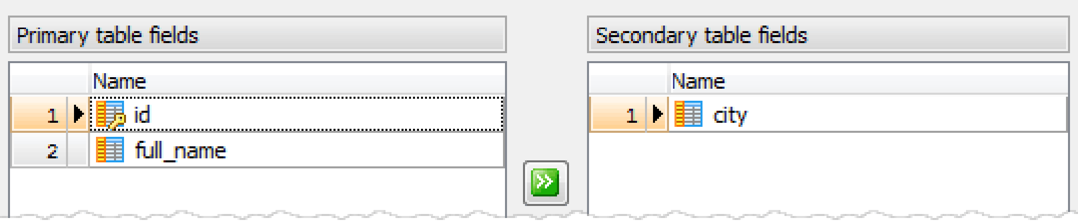
person

Secondary table

public

city

The primary table must contain now only 'id' and' full_name' columns. The field 'city_id' will be added to the table automatically.



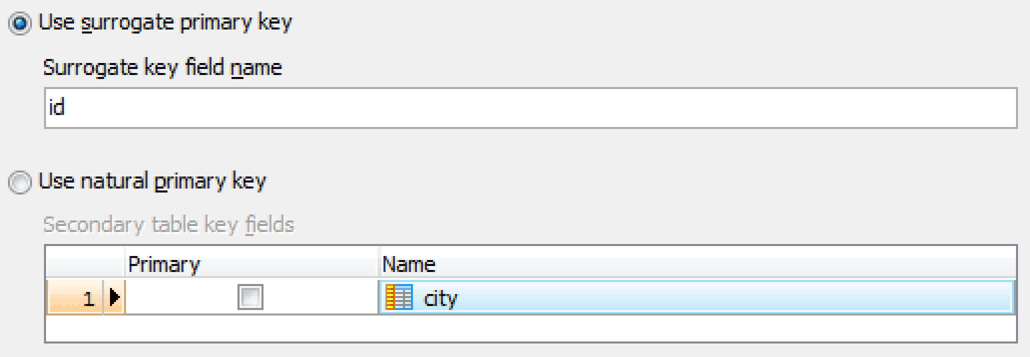
Primary table fields

	Name
1	id
2	full_name

Secondary table fields

	Name
1	city

Now we have to specify what kind of primary key to be created for the secondary table: surrogate or natural. We create create the 'city' table with a surrogate primary key.



● Use surrogate primary key

Surrogate key field name

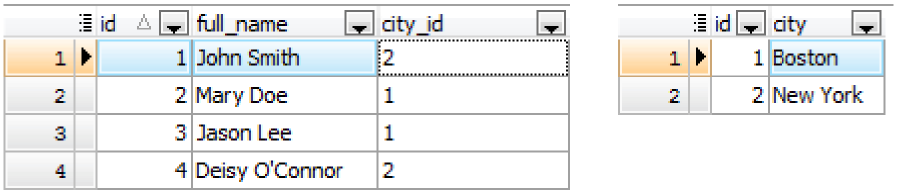
id

○ Use natural primary key

Secondary table key fields

	Primary	Name
1	☐	city

Then click Ready and get the following tables:

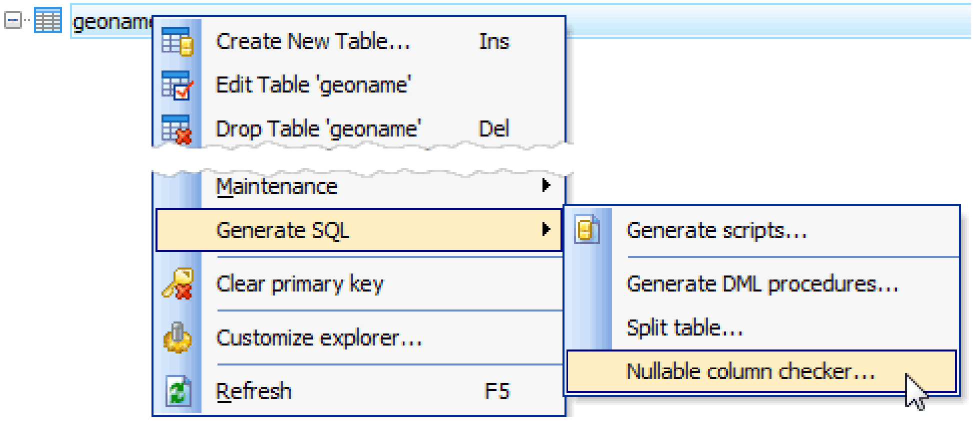


	id	full_name	city_id
1	1	John Smith	2
2	2	Mary Doe	1
3	3	Jason Lee	1
4	4	Deisy O'Connor	2

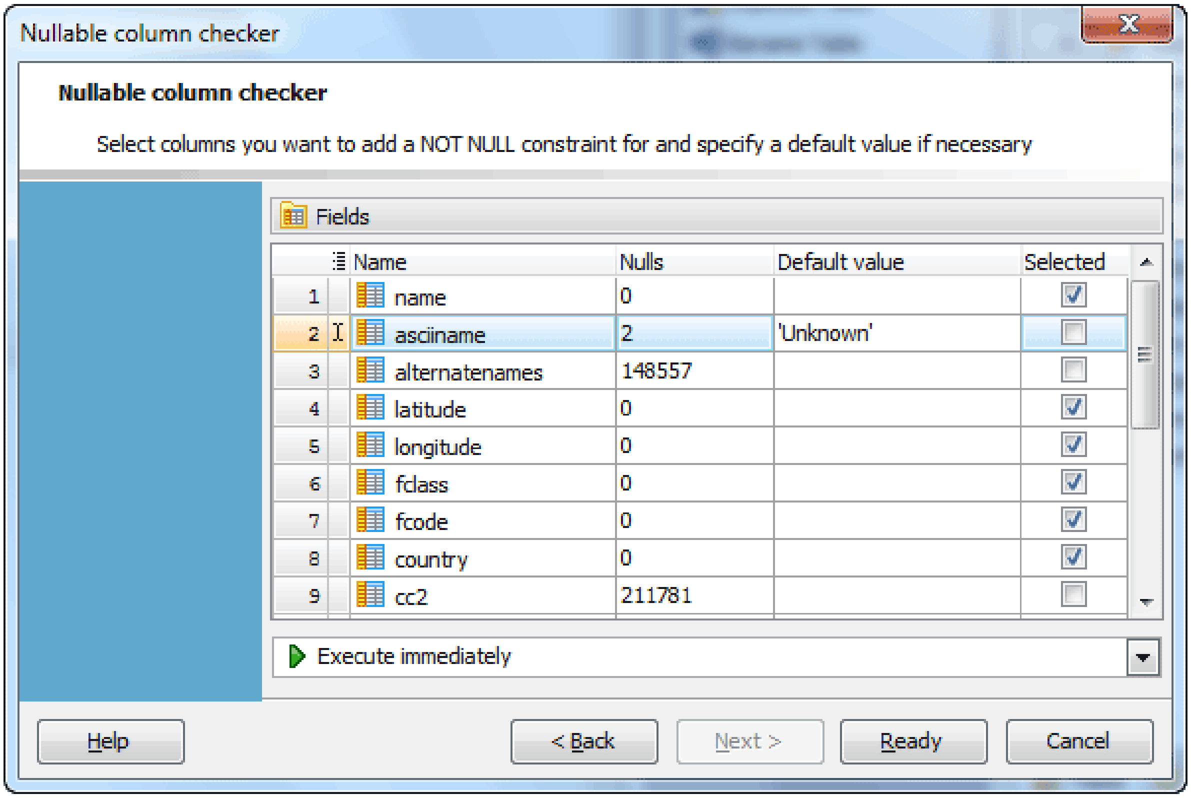
	id	city
1	1	Boston
2	2	New York

9.18 Nullable Column Checker

Nullable Column Checker allows you to refactor your database schema by enforcing NOT NULL constraints to all necessary table columns. It suggests candidates for NOT NULL columns among columns of the selected table and generates SQL script to replace all NULL values of selected columns with specified default values and to add the NOT NULL constraint to these columns. To invoke the wizard, follow the corresponding link of the Generate SQL section of popup menu of the selected table at the Explorer tree.



To get SQL scripts enforcing NOT NULL constraints to columns of an existing table, select the necessary columns, specify the default values to be used instead of existing columns NULLs and select the action to perform after the generation. The created scripts can be copied to Clipboard, saved to a file, sent to SQL Script Editor 364 or executed immediately.



Nullable column checker

Select columns you want to add a NOT NULL constraint for and specify a default value if necessary

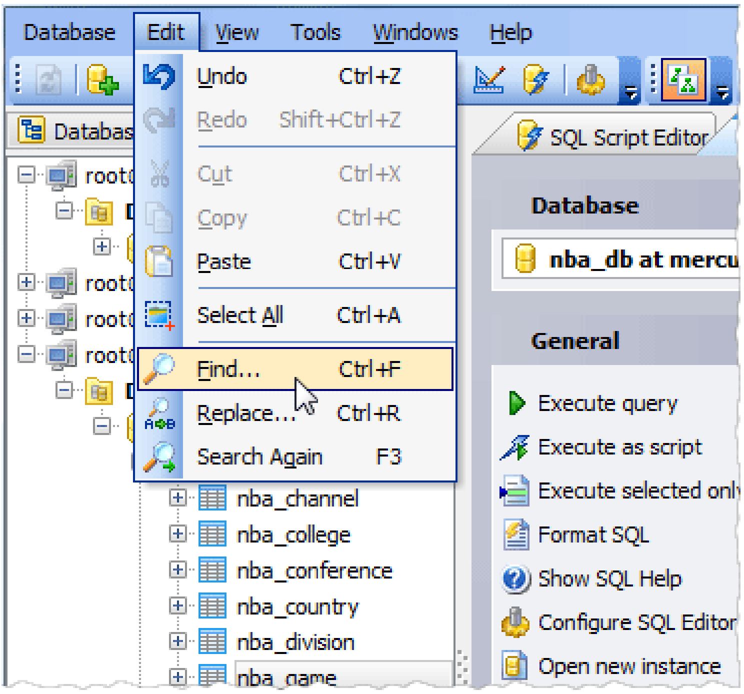
	Name	Nulls	Default value	Selected
1	name	0		☑
2	asciiname	2	'Unknown'	☐
3	alternatenames	148557		☐
4	latitude	0		☑
5	longitude	0		☑
6	fclass	0		☑
7	fcode	0		☑
8	country	0		☑
9	cc2	211781		☐

Execute immediately

9.19 Dialogs

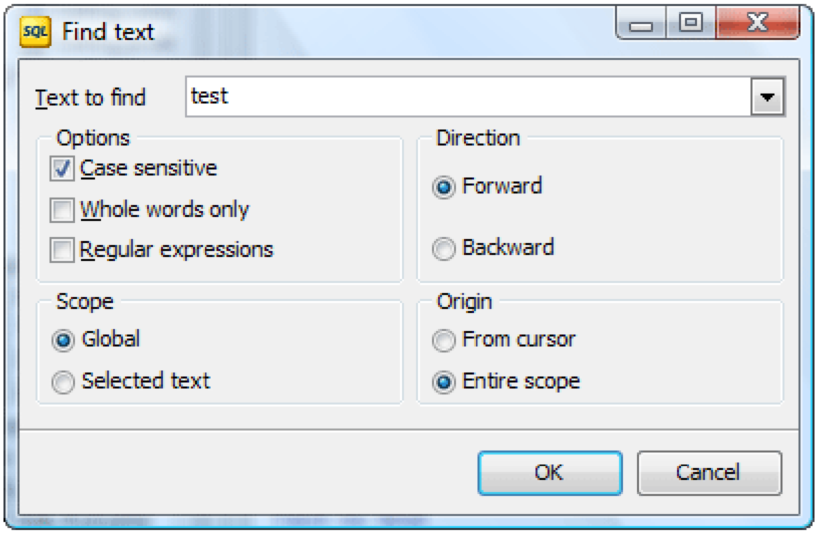
MS SQL Maestro provides two dialogs for searching and replacing text in the editor areas of the database tools. Both of them are available through the popup menu of the editor area.

- Find Text dialog 416
- Replace Text dialog 417



9.19.1 Find Text dialog

The Find Text dialog is provided for quick search for certain text.



Text to find: test

Options
☑ Case sensitive
☐ Whole words only
☐ Regular expressions

Direction
● Forward
○ Backward

Scope
● Global
○ Selected text

Origin
○ From cursor
● Entire scope

Text to find
Enter a search string or click the down arrow next to the input box to select from a list of previously entered search strings.

☑ Case sensitive
Differentiates uppercase from lowercase when performing a search.

☑ Whole words only
Searches for words only. (With this option off, the search string might be found within longer words.)

☑ Regular expressions
Recognizes regular expressions in the search string.

Forward
Searches from the current position to the end of the file. Forward is the default.

Backward
Searches from the current position to the beginning of the file.

Global
Searches the entire file, in the direction specified by the Direction setting. Global is the default scope.

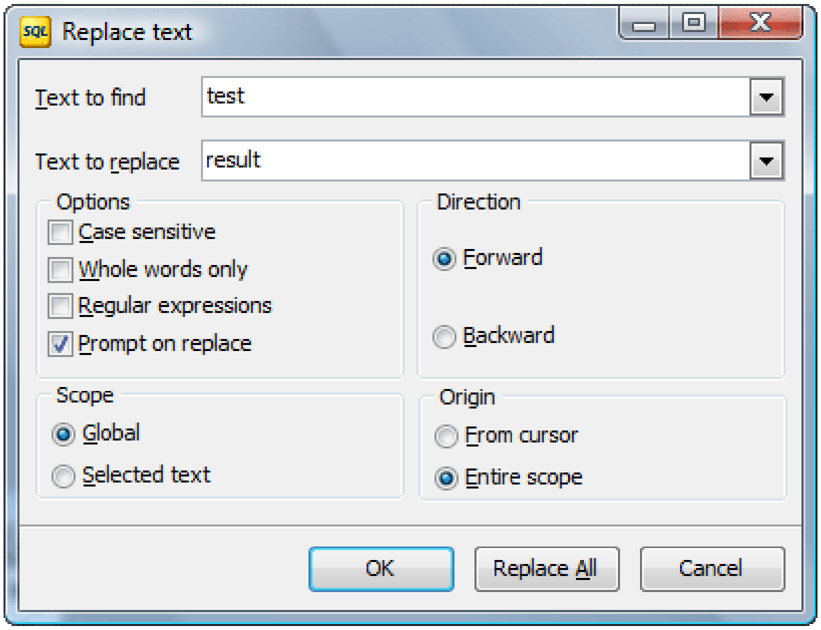
Selected text
Searches within the selected text only, in the direction specified by the Direction setting. You can use the mouse or block commands to select a block of text.

From cursor
The search starts at the cursor's current position, and then proceeds either forward to the end of the scope, or backward to the beginning of the scope depending on the Direction setting. From Cursor is the default setting.

Entire scope
The search covers either the entire block of selected text or the entire file (no matter where the cursor is), depending upon the Scope options.

9.19.2 Replace Text dialog

The Replace Text dialog is provided for searching and replacing text in the editor window.



Text to find: test

Text to replace: result

Options
☐ Case sensitive
☐ Whole words only
☐ Regular expressions
☑ Prompt on replace

Direction
● Forward
○ Backward

Scope
● Global
○ Selected text

Origin
○ From cursor
● Entire scope

Text to find

Enter a search string. To select from a list of previously entered search strings, click the down arrow next to the input box.

Text to replace

Enter the replacement string. To select from a list of previously entered search strings, click the down arrow next to the input box. To replace the text with nothing, leave this input box blank.

☑ Case sensitive

Differentiates uppercase from lowercase when performing a search.

☑ Whole words only

Searches for words only. (With this option off, the search string might be found within longer words.)

☑ Regular expressions

Recognizes specific regular expressions in the search string.

☑ Prompt on replace

Prompts you before replacing each occurrence of the search string. When Prompt on replace is off, the editor automatically replaces the search string.

Forward

Searches from the current cursor position, to the end of the file. Forward is the default Direction setting.

Backward

Searches from the current cursor position, to the beginning of the file.

Global

Searches the entire file, in the direction specified by the Direction setting. Global is the default scope.

From cursor
The search starts at the cursor's current position, and proceeds either forward to the end of the scope, or backward to the beginning of the scope depending on the Direction setting. From cursor is the default Origin setting.

Entire scope
The search covers either the entire block of selected text or the entire file (no matter where the cursor is in the file), depending upon the Scope options.

Replace All
Click Replace all to replace every occurrence of the search string. If you check Prompt on replace, the Confirm dialog box appears on each occurrence of the search string.

10 Server Maintenance

MS SQL Maestro provides graphical wizard interface for native Microsoft SQL server maintenance operations. To make your work with the server easier, MS SQL Maestro also provides some graphical tools for working with the server as a whole.

The following services and server tools are available:

- Attach Database 422
- Detach Database 421

10.1 Detach Database

Detaching a database removes it from the instance of SQL Server but leaves the database intact within its data files and transaction log files. These files can be used afterwards to attach the database to any instance of SQL Server, including the server from which the database was detached.

To run Detach Database Wizard, select the Database | Detach Database main menu item.

- Selecting server to attach the database to 421
- Specifying additional parameters 421

See also: Attach Database 422

10.1.1 Selecting server to detach the database from

Set the necessary connection properties.

Host name
You can select or type the server instance name.

You can use the current Windows user account for authentication, if this account is added to the sever as login. For this you need to select the Windows Authentication radio item.

You can also specify a server login for authentication, if you select SQL Server Authentication. The Login name and Password text areas become enabled. You have to type a valid login and password there.

10.1.2 Specifying additional parameters

Specify options according to your needs. The detailed description is given below.

Database Name
Defines the name of the database to be detached.

☑ Update Statistics
Specifies whether to skip or run UPDATE STATISTIC. To skip Update statistics, you have to leave this field unchecked. To explicitly run UPDATE STATISTICS, you need to check this field. By default, Update statistics is performed to update the information about the data in the tables and indexes in the Microsoft SQL Server 2005 Database Engine. Performing update statistics is useful for databases that are to be moved to read-only media.

Connections
Defines the number of connections to the database. You can end all active transactions in this database selecting this field and clicking the red cross on the right.

☑ For Replication
Shows whether a database is involved with replication.

The Status column displays the current database state.

10.2 Attach Database

You can attach a copied or detached SQL Server database. In SQL Server 2005, full-text files that are part of a database are attached with the database. Generally, attaching a database places it in the same state as it was in when it was detached or copied. However, in SQL Server 2005, attach and detach operations both disable cross-database ownership chaining for the database.

To run Attach Database Wizard, select the Database | Attach Database main menu item.

- Selecting server to attach the database to 422
- Specifying additional parameters 422

See also: Detach Database 421

10.2.1 Selecting server to attach the database to

You need to specify server to attach and the connection options.

Host name
You can select or type server instance name.

You can use the current Windows user account for authentication if this account is added to the sever as login. For this you need to select the Windows Authentication radio item.

You can also specify any server login for authentication, if you select SQL Server Authentication. The Login name and Password text areas become enabled. You have to type a valid login and password there.

Check the Create a single profile option to set the database name manually and create a single profile for this database. Use this option if the SHOW DATABASES Microsoft SQL command is forbidden on your database server (e.g. for security reasons).

10.2.2 Specifying additional parameters

Specify options according to your needs. The detailed description is given below.

MDF File To Attach
You should specify a MDF file to attach.

Attach As
This field defines the name of the new database. Database names must be unique within an instance of SQL Server.

Owner
Specify the owner in the field. By default, only the owner of an object can perform various operations with the object. In order to allow other users to operate it, privileges must be granted. (However, users that have the superuser attribute can always access any object.)

☑ Rebuild Log

Specifies that the database is created by attaching an existing set of the operating system files. This option is limited to read/write databases. If one or more transaction log files are missing, the log file is rebuilt.

Service Broker Option

Controls Service Broker options on the database. Service Broker options are enable broker, new broker and error broker conversations.

Enable broker	Specifies that Service Broker is enabled for the specified database.
New broker	Creates a new *service_broker_guid* in *sys.databases* and the restored database.
Error broker conversations	Ends all conversations with an error that indicates a copy of the broker has been created.

11 Options

MS SQL Maestro allows you to customize the way it works within the Options dialog. To open the dialog, select the Tools | Options main menu item.

The window allows you to customize the options grouped by the following sections:

- **Application** 425
  General MS SQL Maestro options: environment style, confirmations, window restrictions, explorer tree, SQL Editor, Visual Query Builder, etc.

- **Editors & Viewers** 447
  Customizing of all the SQL editors - SQL Editor, SQL Script Editor, etc.

- **Appearance** 455
  Customizing program interface - bars, trees, menus, etc.

Besides, the Options dialog allows you to export all program settings to a **.reg* file for future use, e.g. on another PC (see Export Settings 464 for details).

It is a good idea to check through these settings before you start working with MS SQL Maestro. You may be surprised at all the things you can adjust and configure!

11.1 Application

The Application section allows you to customize common rules of MS SQL Maestro behavior. The section consists of several tab; follow the links to find out more about each of them.

11.1.1 Preferences

User interface area allow you to select your favorite UI style according to your preferences.

☑ Display splash screen at startup
Displays the splash screen on MS SQL Maestro startup.

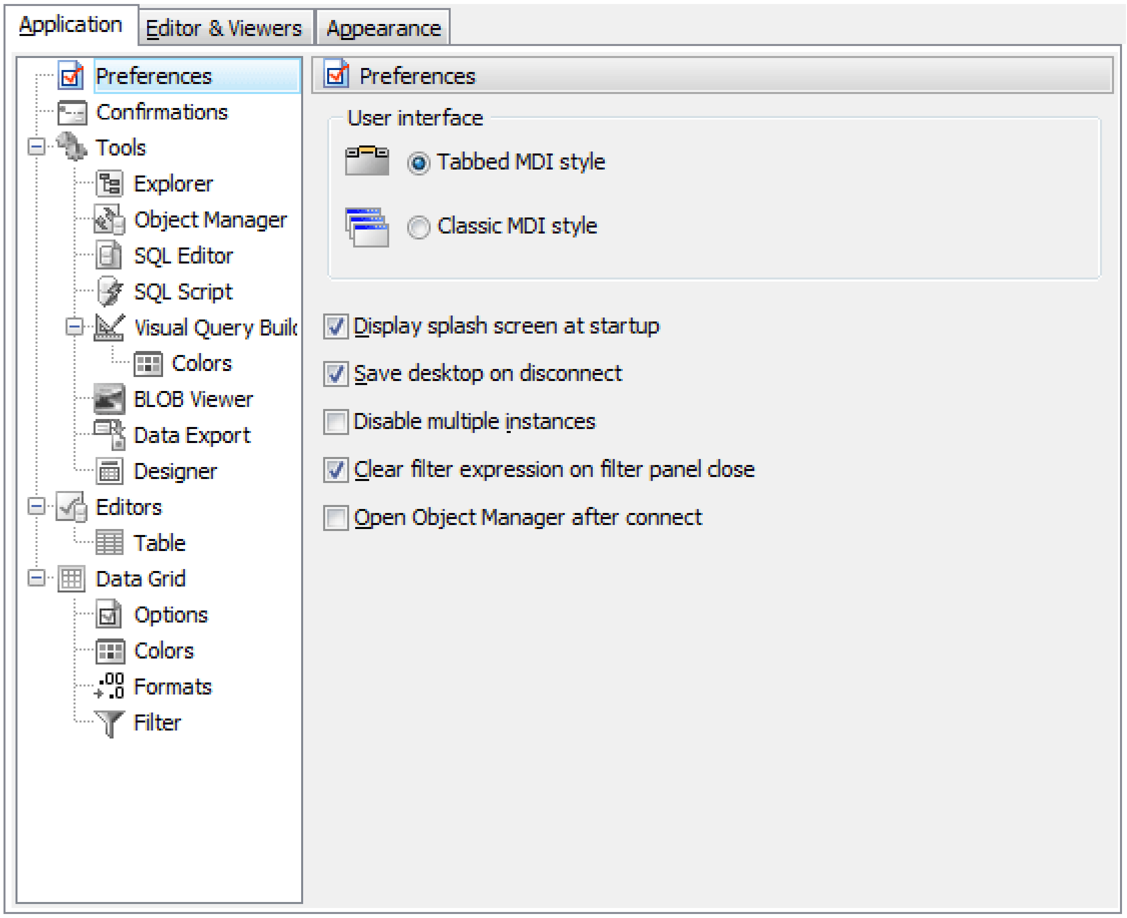
☑ Save desktop on disconnect
Saves all the database windows and their positions on disconnecting from the database.

☑ Disable multiple instances
Prohibits running multiple instances of MS SQL Maestro.

☑ Open Object Manager after connect
Opens the Object Manager window after connection is established.

☑ Clear filter expression on filter panel close
Clears the filter applied to the explorer tree and all the instances of Object Manager

after the filter panel is closed.



User interface

● Tabbed MDI style

○ Classic MDI style

☑ Display splash screen at startup

☑ Save desktop on disconnect

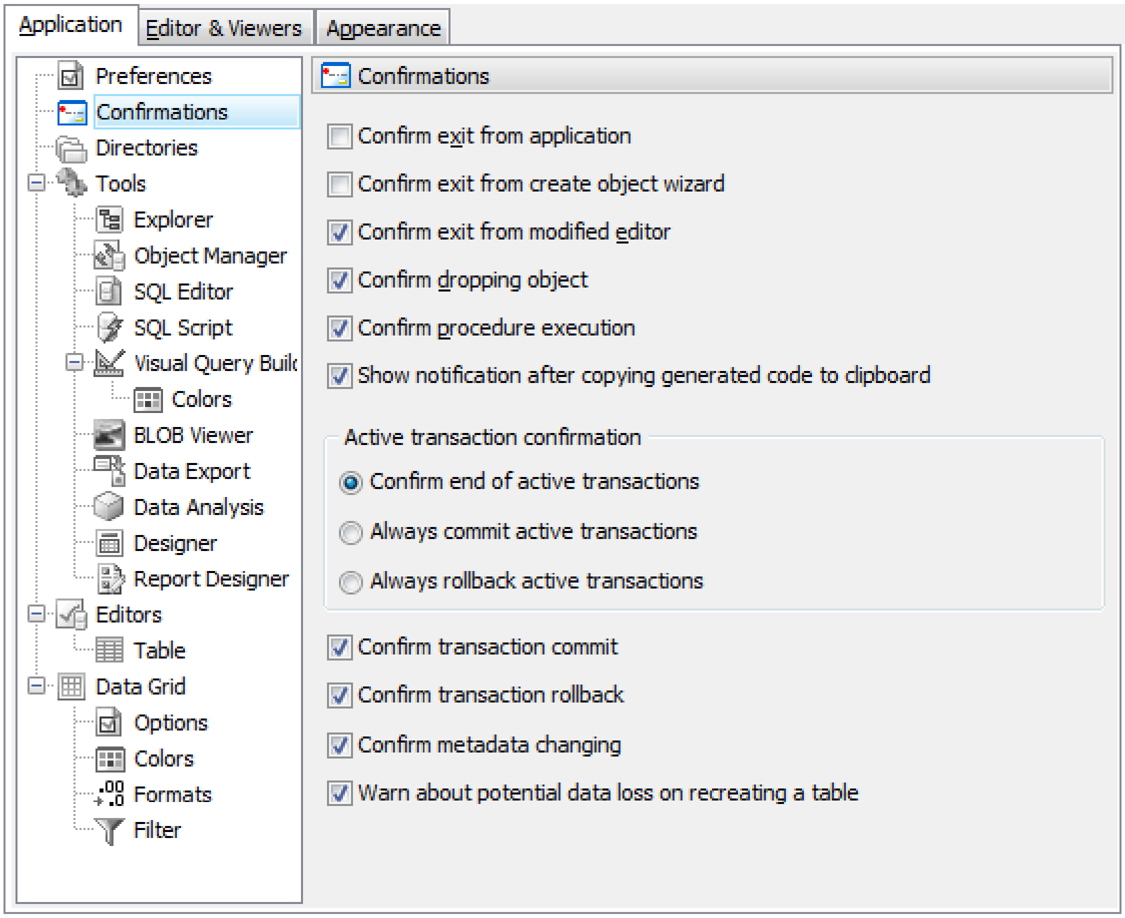
☐ Disable multiple instances

☑ Clear filter expression on filter panel close

☐ Open Object Manager after connect

11.1.2 Confirmations

Use this tab to manage application confirmations.



Confirmations

☐ Confirm exit from application
☐ Confirm exit from create object wizard
☑ Confirm exit from modified editor
☑ Confirm dropping object
☑ Confirm procedure execution
☑ Show notification after copying generated code to clipboard

Active transaction confirmation

● Confirm end of active transactions
○ Always commit active transactions
○ Always rollback active transactions

☑ Confirm transaction commit
☑ Confirm transaction rollback
☑ Confirm metadata changing
☑ Warn about potential data loss on recreating a table

☑ Confirm exit from Create Object Wizard

If this option is checked, the program requires confirmation each time you want to exit the Create Object Wizard.

☑ Confirm exit from modified editor

If this option is checked, the program asks you to confirm exit from the editor, if you have made any changes.

☑ Confirm dropping object

If this option is checked, the program requires confirmation for dropping database object.

☑ Confirm exit from application

If this option is checked, the program requires confirmation when you want to exit <% PRODUCT_NAME%.

☑ Transaction confirmation

Select whether you will be prompted to commit or rollback active transaction or MS SQL Maestro will commit or rollback transactions without asking.

☑ Confirm metadata changing

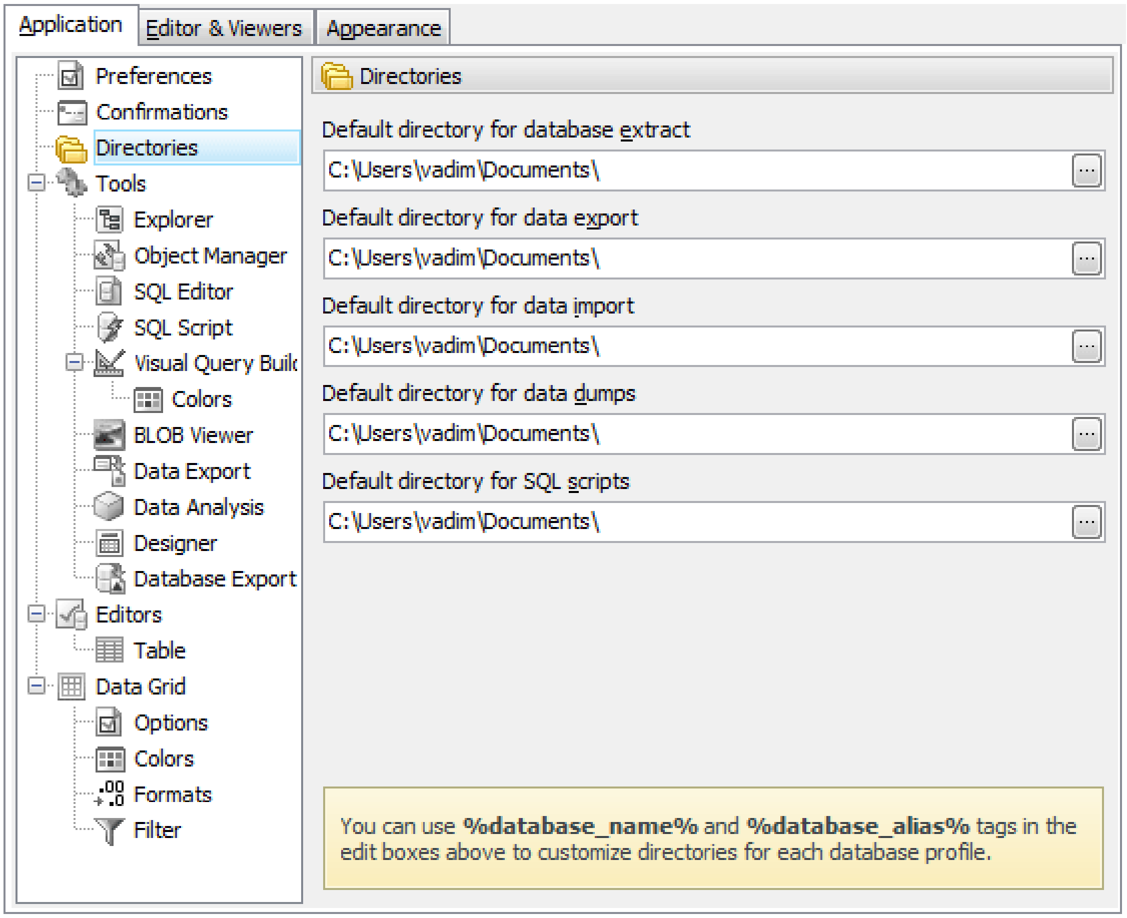
If this option is checked, the program requires confirmation for changing metadata.

11.1.3 Directories

The tab allows you to specify default directories to be used on database profiles creating. You can use such variables as %database_name%, %database_alias%, and %user_name%.

Example:
To store all SQL scripts in folders sorted by databases in the "C:\SQL Scripts\" directory, specify the default directory for SQL scripts as follows:

C:\SQL Scripts\%database_name%



Default directory for database extract: C:\Users\vadim\Documents\

Default directory for data export: C:\Users\vadim\Documents\

Default directory for data import: C:\Users\vadim\Documents\

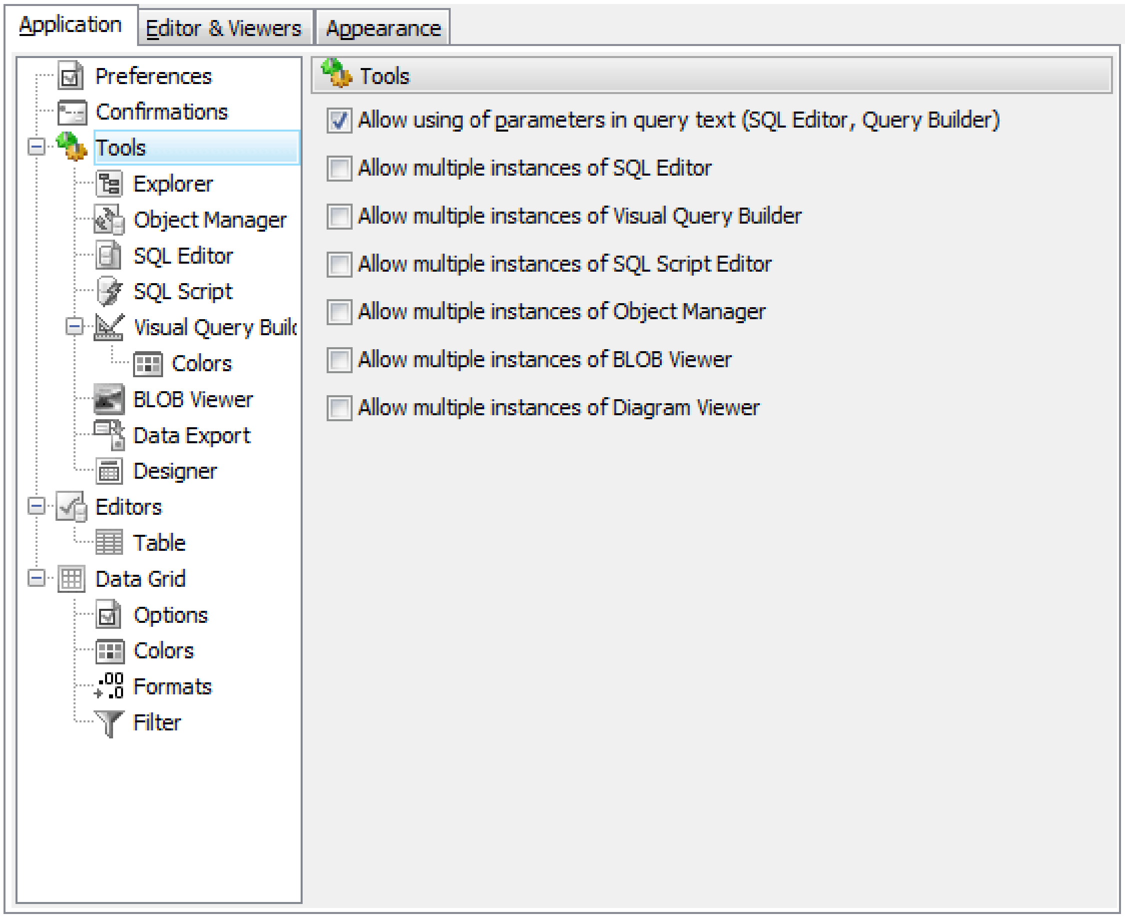
Default directory for data dumps: C:\Users\vadim\Documents\

Default directory for SQL scripts: C:\Users\vadim\Documents\

You can use **%database_name%** and **%database_alias%** tags in the edit boxes above to customize directories for each database profile.

11.1.4 Tools

Below you will find a detailed decryption of the following tools options.



☑ Allow using of parameters in query text (SQL Editor, Query Builder)

☐ Allow multiple instances of SQL Editor

☐ Allow multiple instances of Visual Query Builder

☐ Allow multiple instances of SQL Script Editor

☐ Allow multiple instances of Object Manager

☐ Allow multiple instances of BLOB Viewer

☐ Allow multiple instances of Diagram Viewer

☑ Allow using of parameters in query text

Check this option to be able to use query parameters in SQL Editor 313 and Visual Query Builder 318.

☑ Allow multiple instances of SQL Editor

Check this option to be able to use multiple instances of SQL Editor 313 simultaneously.

☑ Allow multiple instances of Visual Query Builder

Check this option to be able to use multiple instances of Visual Query Builder 318 simultaneously.

☑ Allow multiple instances of SQL Script Editor

Check this option to be able to use multiple instances of SQL Script Editor 364 simultaneously.

☑ Allow multiple instances of Object Manager

Check this option to be able to use multiple instances of Object Manager simultaneously.

☑ Allow multiple instances of BLOB Viewer

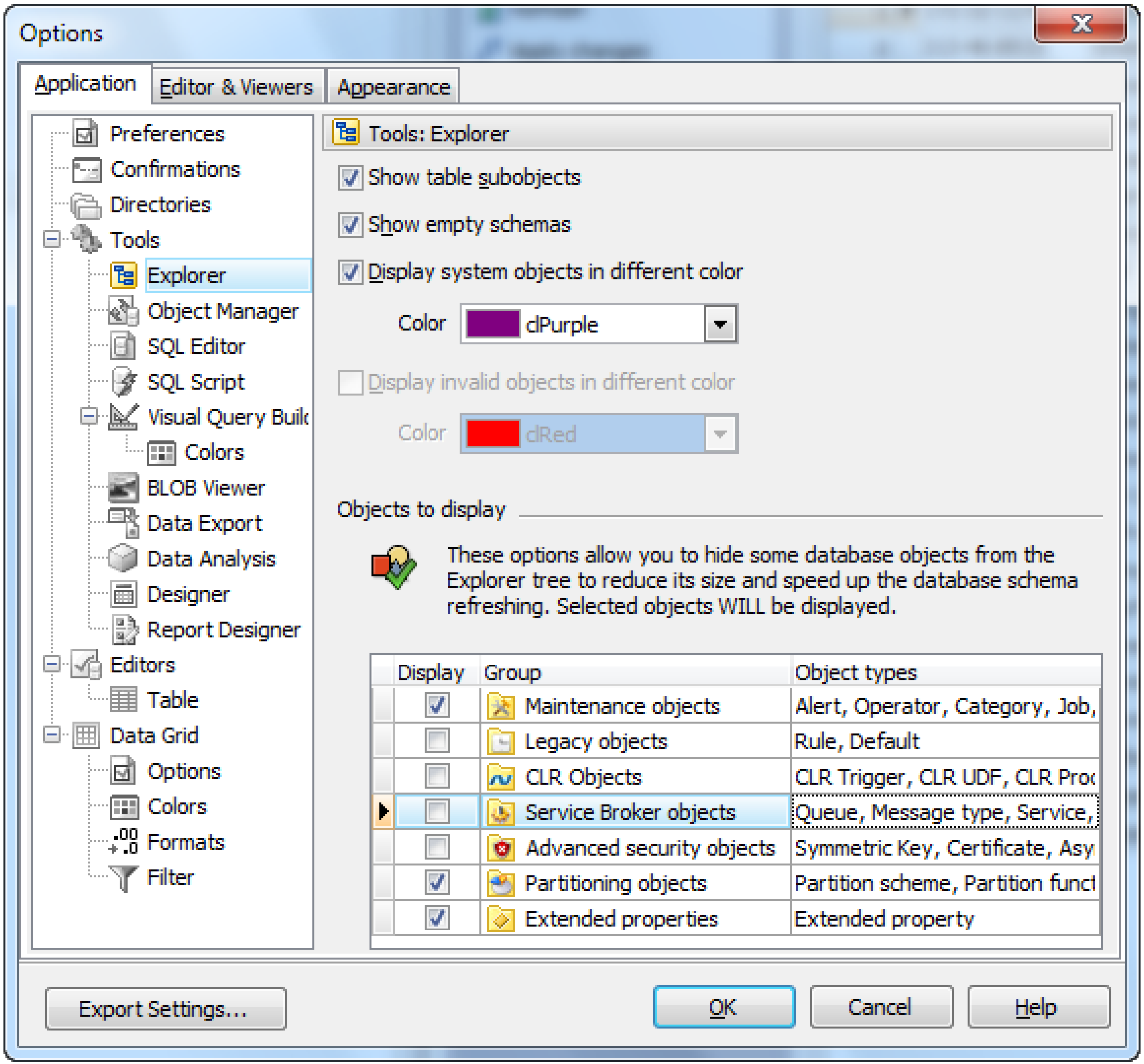
Check this option to be able to use multiple instances of BLOB Viewer 379 simultaneously.

☑ Allow multiple instances of Diagram Viewer

Check this option to be able to use multiple instances of Diagram Viewer 385 simultaneously.

11.1.4.1 Explorer

Below you will find a detailed decryption of the following explorer options.



Tools: Explorer

☑ Show table subobjects

☑ Show empty schemas

☑ Display system objects in different color

Color: clPurple

☐ Display invalid objects in different color

Color: clRed

Objects to display

These options allow you to hide some database objects from the Explorer tree to reduce its size and speed up the database schema refreshing. Selected objects WILL be displayed.

Display	Group	Object types
☑	Maintenance objects	Alert, Operator, Category, Job,
☐	Legacy objects	Rule, Default
☐	CLR Objects	CLR Trigger, CLR UDF, CLR Proc
☐	Service Broker objects	Queue, Message type, Service,
☐	Advanced security objects	Symmetric Key, Certificate, Asy
☑	Partitioning objects	Partition scheme, Partition funct
☑	Extended properties	Extended property

☑ Show table subobjects
Shows/hides table subobjects (fields and indexes) in the explorer tree.

☑ Sort profiles by aliases
Sorts profile aliases alphabetically in the explorer tree.

☑ Expand the "Tables" node after connection
Shows all database tables in the explorer tree after connecting to the database.

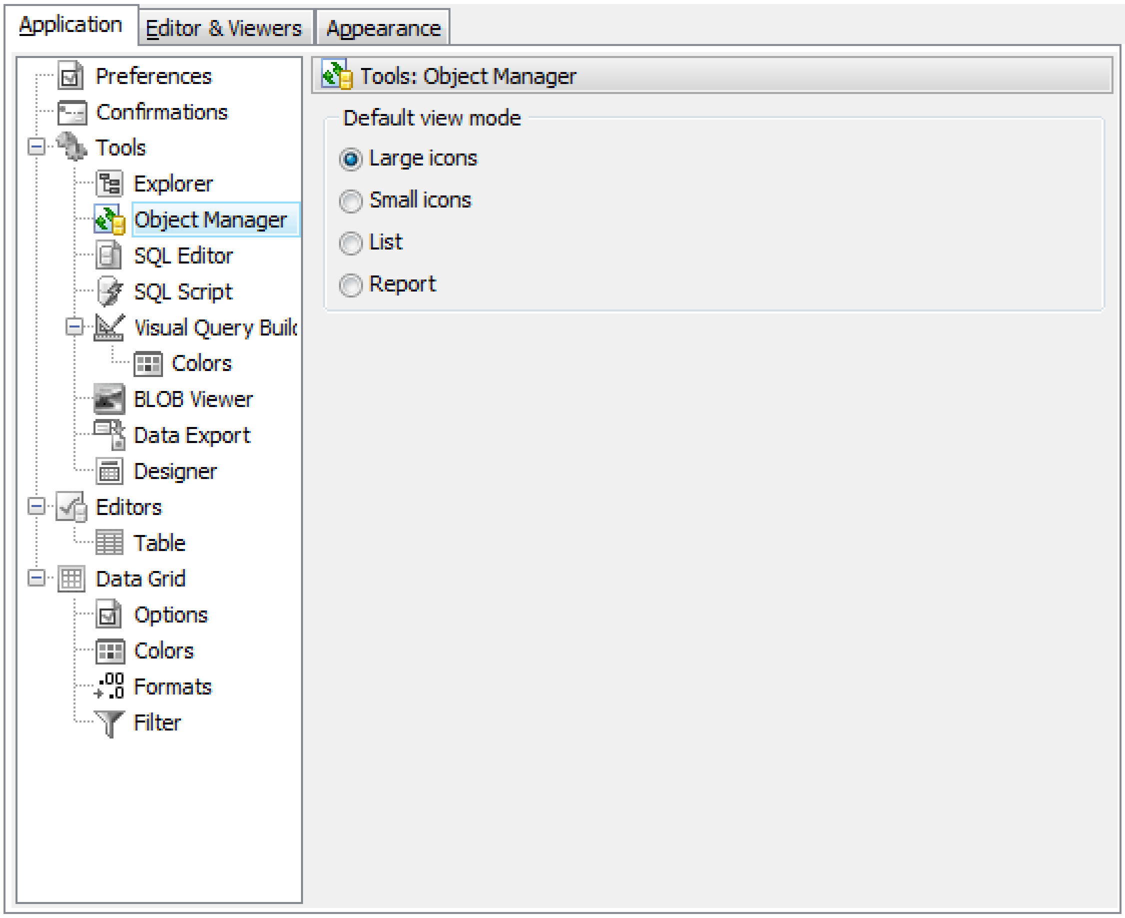
☑ Expand the "Queries" node after connection
Shows all database queries in the explorer tree after connecting to the database.

☑ Display system objects in different color
Represents all system objects in selected color.

You can also exclude/include rarely used objects from/to the Explorer tree. Manage object groups to be displayed at Explorer with corresponding checkboxes.

11.1.4.2 Object Manager

Below you will find a detailed decryption of the following Object Manager options.



Tools: Object Manager

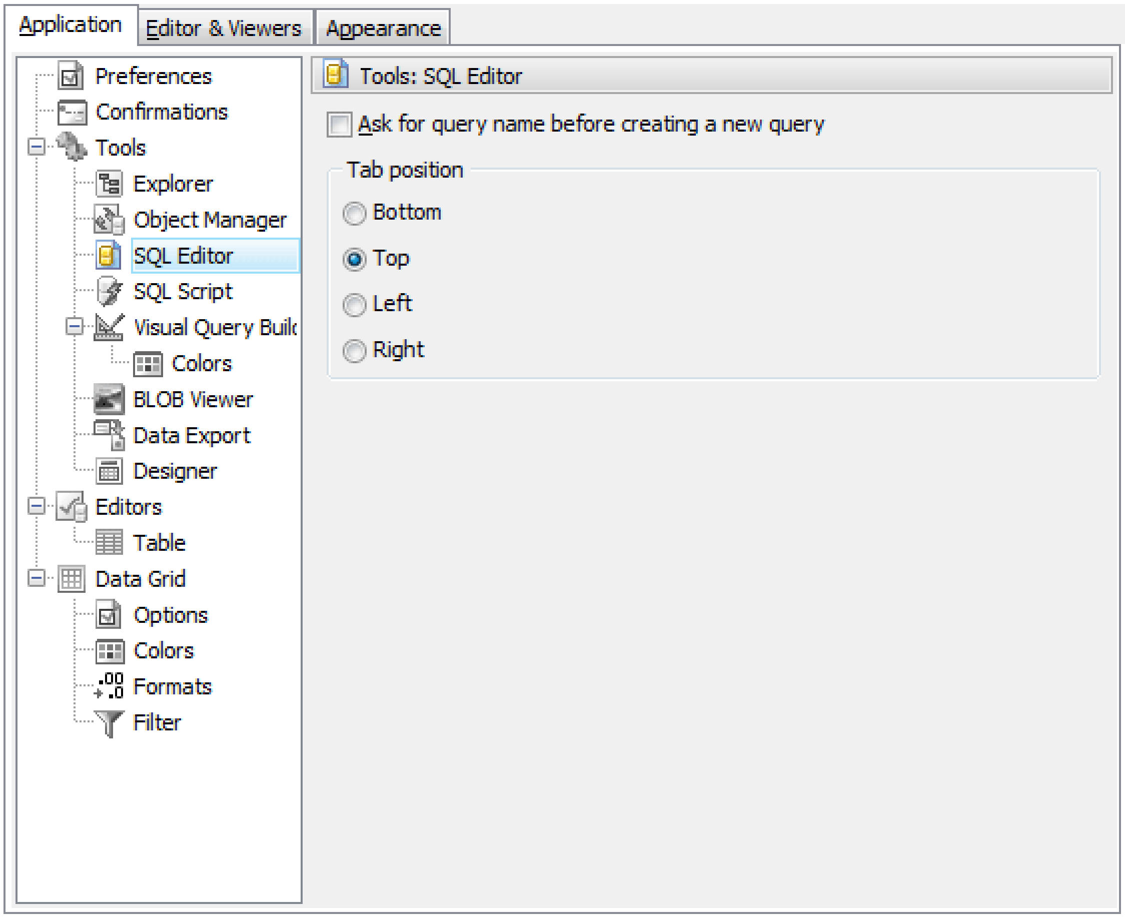
Default view mode

● Large icons
○ Small icons
○ List
○ Report

Default view mode
Defines which view mode (*large icons*, *small icons*, *list* or *report*) is applied to Object Manager by default.

11.1.4.3 SQL Editor

Below you will find a detailed decryption of the following SQL Editor options.



☐ Ask for query name before creating a new query

Tab position
○ Bottom
● Top
○ Left
○ Right

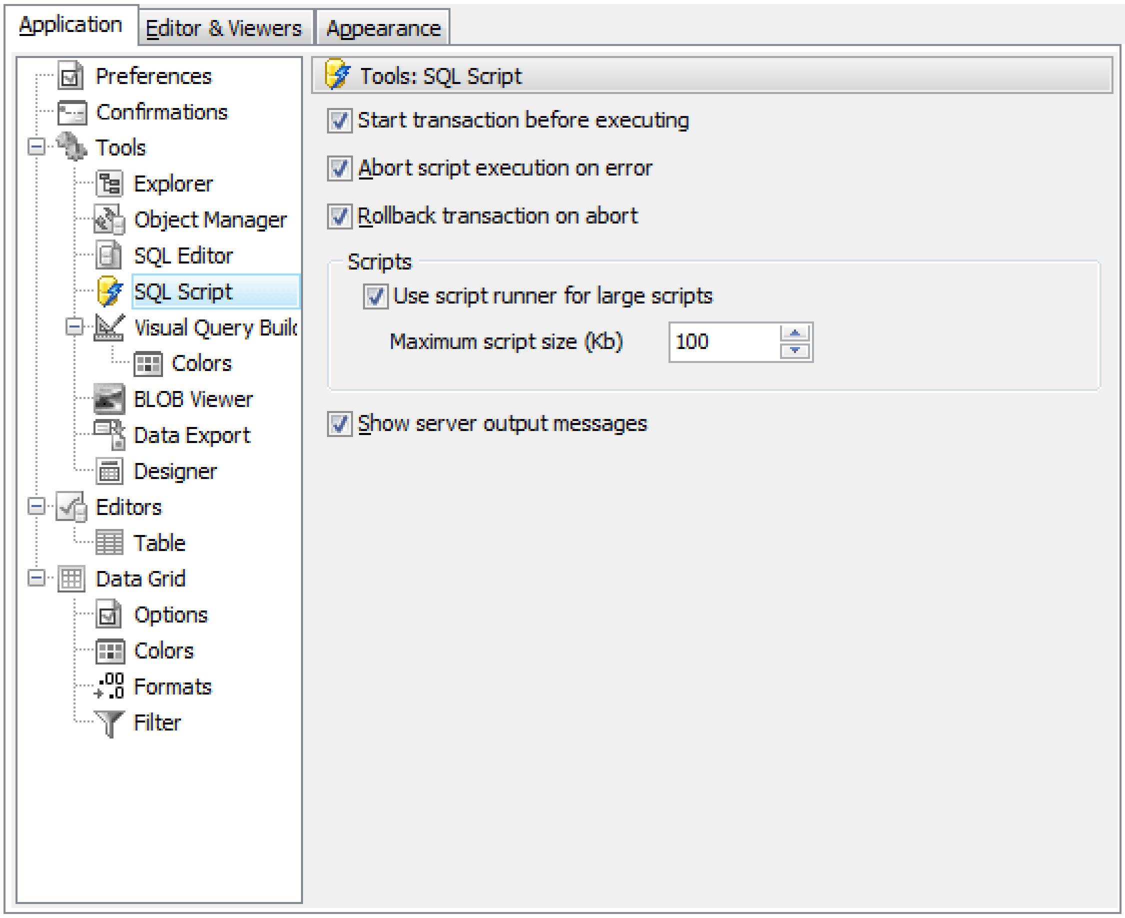
☑ Ask for query name before creating a new query
If this option is checked, SQL Editor 313 asks for a query name each time you create a new query.

☑ Auto commit
Check the option to execute queries in autocommit mode (default value) or leave it blank to manage transactions manually.

You can also select position of query tabs.

11.1.4.4 SQL Script Editor

Below you will find a detailed decryption of the following SQL Script Editor options.



☑ Start transaction before executing

☑ Abort script execution on error

☑ Rollback transaction on abort

Scripts

☑ Use script runner for large scripts

Maximum script size (Kb) 100

☑ Show server output messages

☑ Abort script execution on error
If this option is checked, script execution aborts when an error occurs.

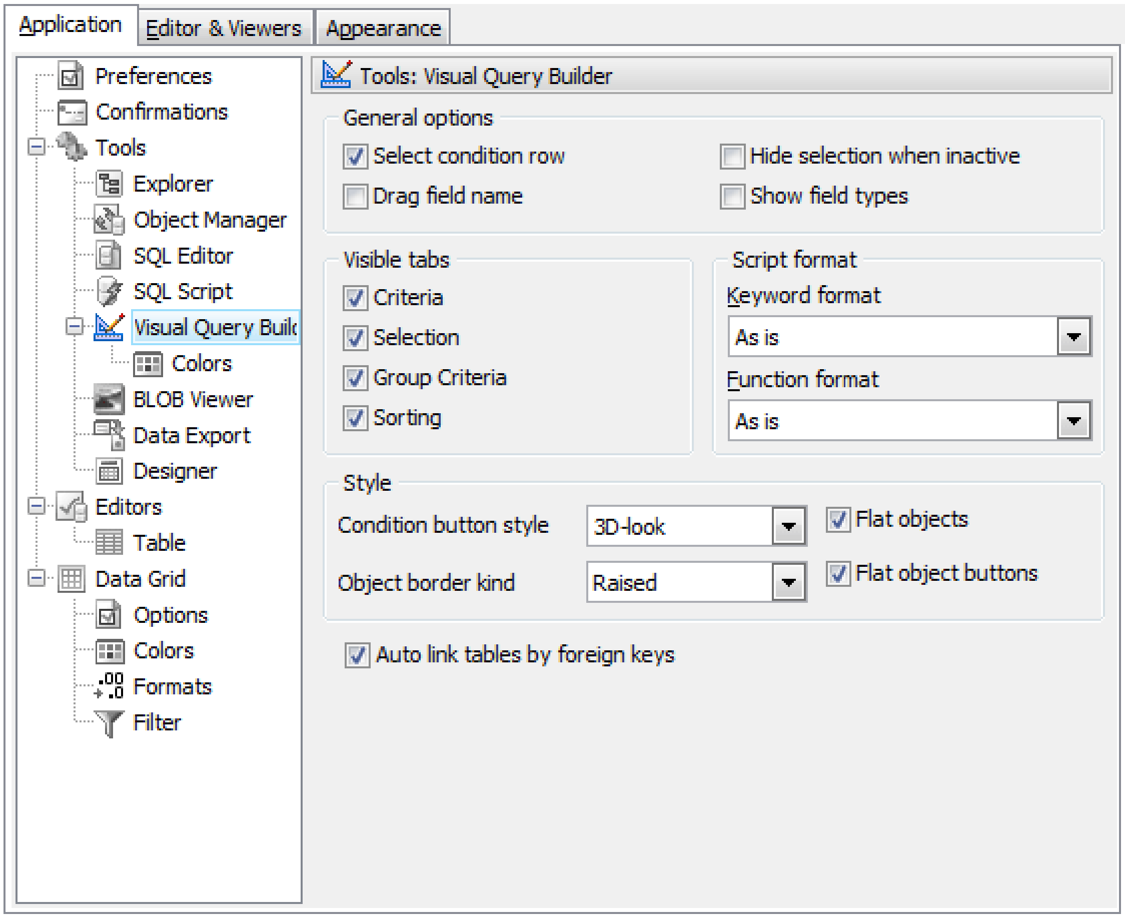
☑ Rollback transaction on abort
This option evokes automatic rollback on script execution abort.

☑ Use script runner for large scripts
Check the box to execute large script in the fastest way. You can change the maximum size of a script to execute without script runner.

☑ Show server output messages
Turn the option ON to see warning messages generated by the server.

11.1.4.5 Query Builder

Below you will find a detailed decryption of the following Query Builder options.



Tools: Visual Query Builder

General options

☑ Select condition row
☐ Hide selection when inactive
☐ Drag field name
☐ Show field types

Visible tabs

☑ Criteria
☑ Selection
☑ Group Criteria
☑ Sorting

Script format

Keyword format: As is
Function format: As is

Style

Condition button style: 3D-look
Object border kind: Raised
☑ Flat objects
☑ Flat object buttons

☑ Auto link tables by foreign keys

☑ Select condition row

Displays the selected condition in different row on the Criteria and Grouping Criteria tabs of Visual Query Builder [318].

☑ Drag field name

Displays the dragged field name in the Builder area.

☑ Hide selection when inactive

Hides the selection when the query builder is inactive.

☑ Show field types

Displays the field type next to the field in the table box.

Visible tabs

These options specify which the query builder tabs are available and which are not. Check them to make the appropriate tabs visible.

Script format

These options specify the case formatting of keywords and functions in query text on the Edit tab. As is saves the original case, Uppercase sets all the keywords/functions to upper case, Lowercase sets all the keywords/functions to lower case, and First upper sets the first letters of all keywords/functions to upper case.

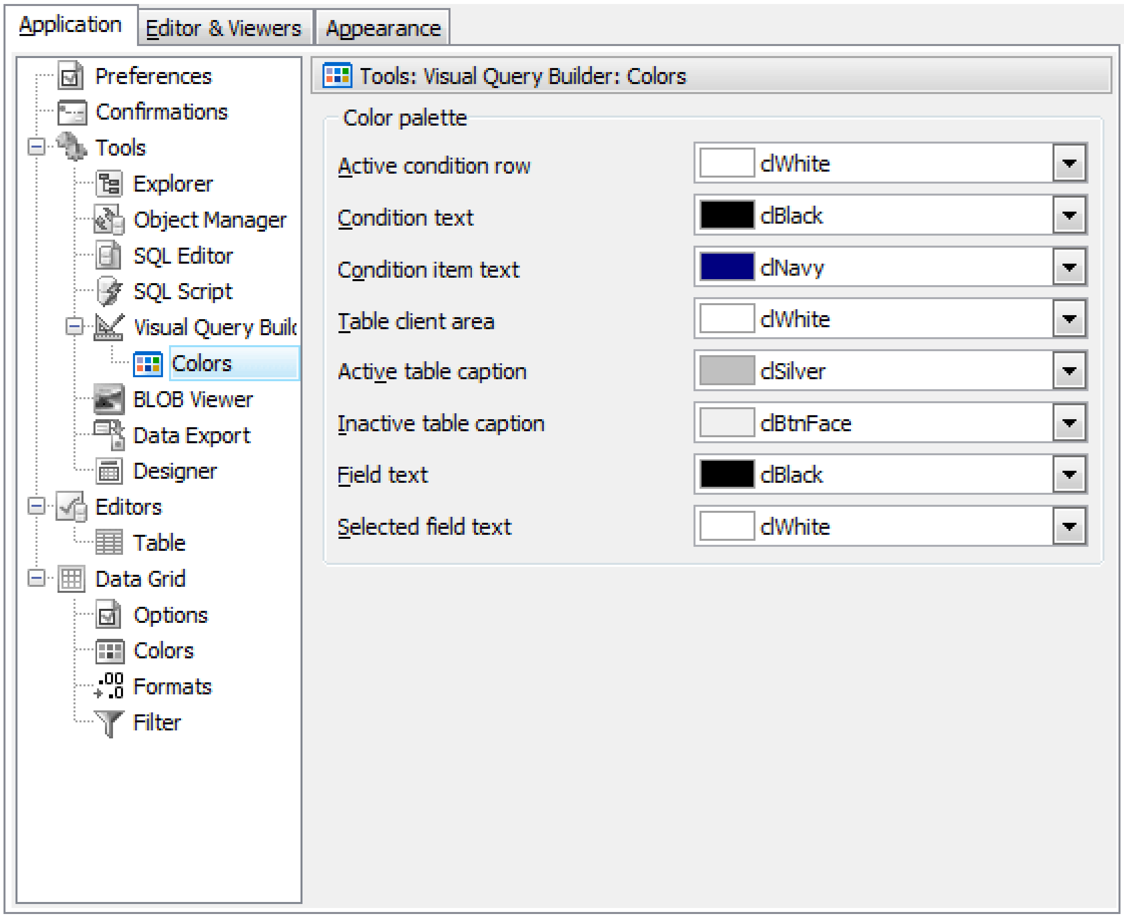
Style
These options specify how different the Query Builder objects look like - 3D, flat, etc.

☑ Auto link tables by foreign keys
If tables that have foreign key reference are placed in Query Builder, in diagram they will be auto linked.END

Colors
These options define colors of the different Query Builder elements: condition row, active caption, table client area, etc. Click an item to select a color for the appropriate Query Builder element.

11.1.4.5.1 Colors

The tab is provided to editing of the Query Builder color schema. Customize colors for all editor element according to your preferences.

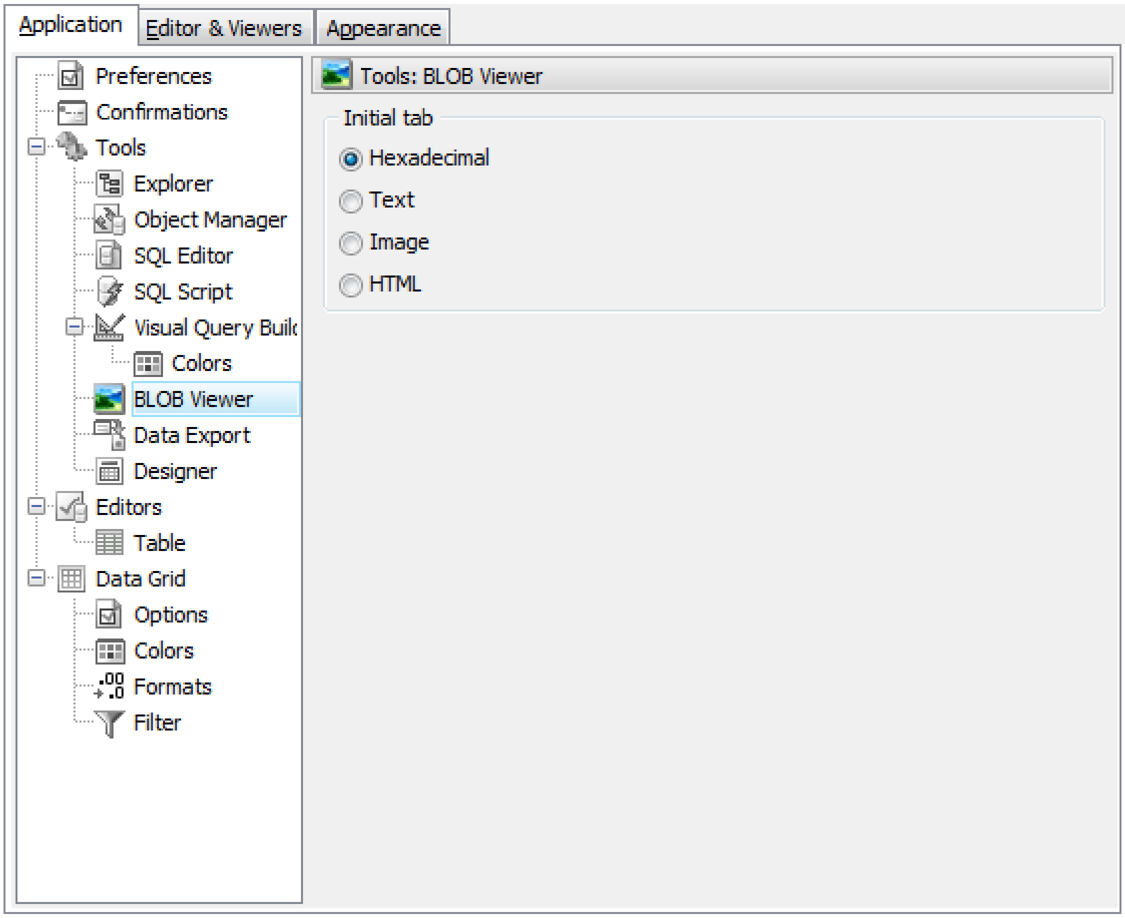


Color palette

Active condition row	clWhite
Condition text	clBlack
Condition item text	clNavy
Table client area	clWhite
Active table caption	clSilver
Inactive table caption	clBtnFace
Field text	clBlack
Selected field text	clWhite

11.1.4.6 BLOB Viewer

Below you will find a detailed decryption of the following BLOB Viewer 379 options.



Tools: BLOB Viewer

Initial tab

● Hexadecimal

○ Text

○ Image

○ HTML

Initial tab
Specifies which tab of BLOB Viewer should be active when it is initially opened.

11.1.4.7 Data Export

This window allows you to customize formats applied to exported data. Edit the format masks to adjust the result format in the way you need.

In *numeric* formats using digit placeholder (# or 0) you can specify the format of number. For example, integer 1234567890 with # ### ##0 integer format is represented like 1 234 567 890. The locations of the leftmost '0' before the decimal point in the format string and the rightmost '0' after the decimal point in the format string determine the range of digits that are always present in the output string.

Conversion and their description for *date*, *time* and *date time* format:

dd	day of the month, represented by 1 or 2 symbols. For example, the first day of month is 1
DD	day of the month, represented only by 2 symbols. For example, the first day of month is 01
mm	minutes

MM	month
yy	year, represented by 2 symbols. For example, 2006 year will be 06
yyyy	year, represented by 4 symbols. For example, 2006
h	hour, represented by 1 or 2 symbols. For example, 2
hh	hour, represented only by 2 symbols. For example, 02

"**.**", "**,**", "**:**" symbols can be placed as separators in format masks for *integer, float, date, time, date time, currency* types.

☑ Auto save format strings
Auto saves format strings when they are changed (e.g. at the Adjusting data formats step in Export Data Wizard 344).

Using Set defaults button, you can set default values of data formats.

11.1.4.8 Database Designer

The following topic provides a shot Database Designer 401 retrofitting's description.

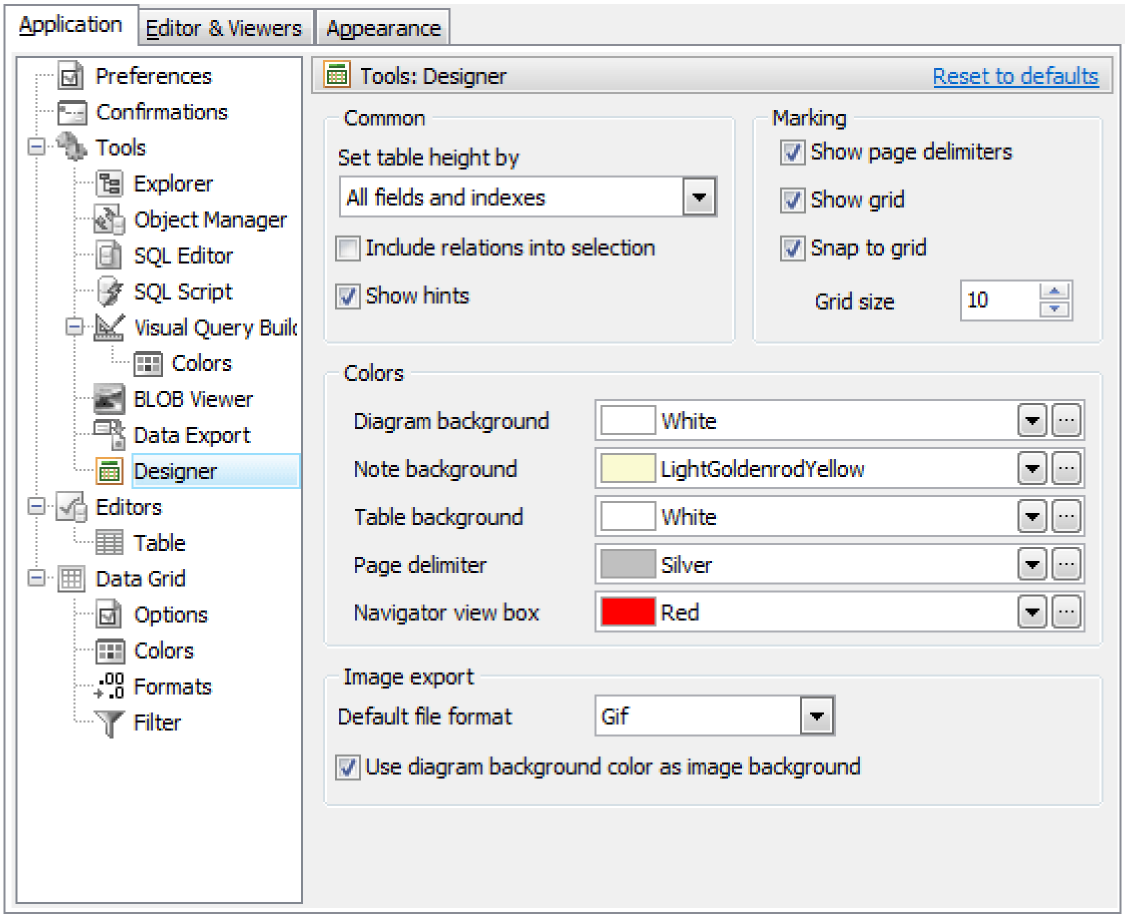
The Set table height option determines what the table attributes will be represented at the diagram (Caption only, Key fields, All fields or All fields and indexes).

Frequently it is necessary to line up some tables and objects on the diagram in a column. One of the ways that you may find convenient to do this is to take advantage of the grid. You can enable showing of the grid by checking the Show grid clause at the Marking options. Then diagram will be covered by points disposed on the same interval between.

If you enable snap to grid function by using the Snap to grid checkbox, when you move the table or other object, its upper left corner "snaps" to the nearest grid point. This feature will assist you to line up objects both horizontally and vertically. Diagram objects moved on a diagram can be automatically snapped to the grid points even if the grid is not displayed on the diagram.

Interval between grid points can be changed as grid size.

The Colors options are to attune the Designer appearance to your taste. Here you change diagrams and tables background colors, the colors of page delimiter and Navigator view box.



Tools: Designer

Common

Set table height by: All fields and indexes

☐ Include relations into selection

☑ Show hints

Marking

☑ Show page delimiters

☑ Show grid

☑ Snap to grid

Grid size: 10

Colors

Diagram background	White
Note background	LightGoldenrodYellow
Table background	White
Page delimiter	Silver
Navigator view box	Red

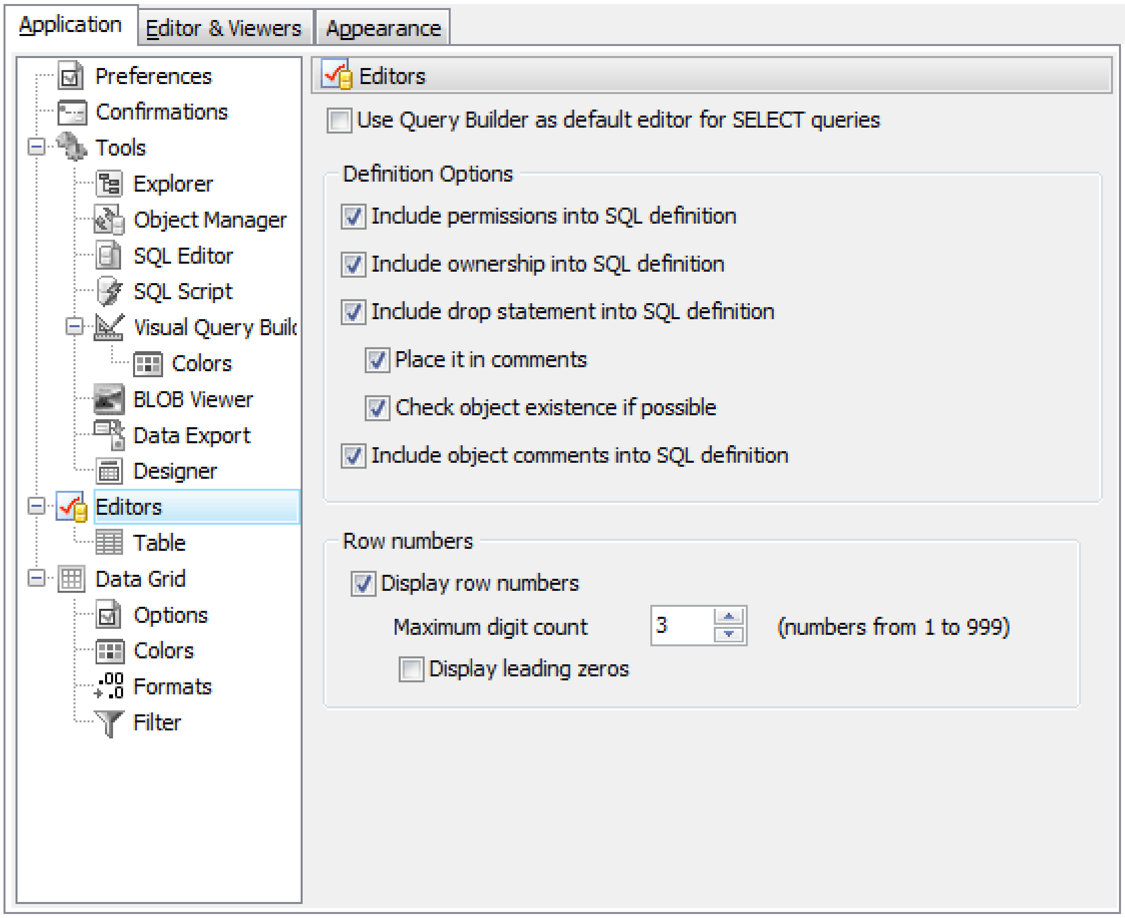
Image export

Default file format: Gif

☑ Use diagram background color as image background

11.1.5 Object Editors

Below you will find a detailed description of the following object editors options.



Editors

☐ Use Query Builder as default editor for SELECT queries

Definition Options

☑ Include permissions into SQL definition

☑ Include ownership into SQL definition

☑ Include drop statement into SQL definition

☑ Place it in comments

☑ Check object existence if possible

☑ Include object comments into SQL definition

Row numbers

☑ Display row numbers

Maximum digit count 3 (numbers from 1 to 999)

☐ Display leading zeros

☑ Open each object editor in a new window

With this option checked a new child window opens each time you open an object for editing, otherwise the edited object is being changed in the existing object editor (except the existing object editor is in modified state).

☑ Use Query Builder as default editor for SELECT queries

With this option enabled all the SELECT queries will be opened in Visual Query Builder 318 instead of SQL Editor 313.

☑ Include permissions

If checked, the SQL definition includes all the GRANT statements, which are applied to the object. Let's assume that a user named 'john' has rights to read the data from the table 'customers' from the schema 'public', and a user named 'michael' can delete data from that table. In this case the script includes itself the following forms:

GRANT SELECT ON public.cutomers to john;

GRANT DELETE ON public.cutomers to michael;

Note: Please take a look at the topic Grant (Transact-SQL Reference) to learn more about security management in Microsoft SQL server.

☑ Include ownership

If checked, the SQL definition includes the object owner specification. For example, if a user named '*john*' is the owner of the table '*customers*' from the schema '*public*', the script contains the following statement:

```
ALTER AUTHORIZATION
  ON OBJECT :: [public].cutomers
  TO [john]
GO
```

☑ Include drop statement
If checked, the SQL definition includes the drop statement.

☑ Place it in comments
With this option drop statement will be placed in comments of the SQL definition.

☑ Include object comments into SQL definition
With this option enabled comments that are specified for the object and object subitems are placed in SQL definition.

Row numbers
This options group allows you to manage the row numbering of the subobjects lists such as fileds, indexes, parameters and so on.
To enable/disable the numbering, use Display row numbers checkbox. You can set the number columns width with Maximum digit count. (I.e. for the value '3' the max column number will be 999).
For uniformity you can use the Display leading zeros option. With this option enabled and maximum digit count '3' you numbering column will be of the form: '001, 002, 003, ...'.

11.1.5.1 Table

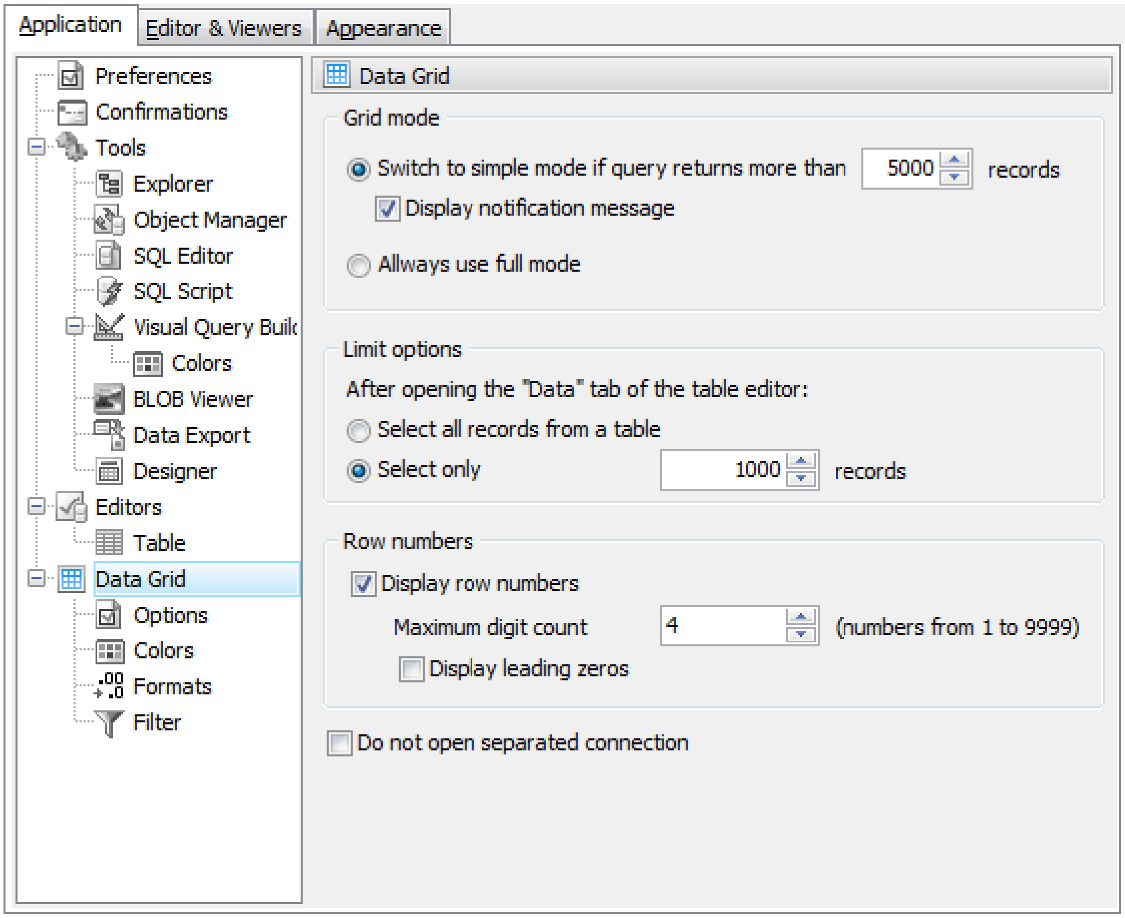
Initial tab
Specifies which tab of Table Editor 79 should be active when it is initially opened.

☑ Retrieve record count before loading data in the data grid
With this option enabled all the SELECT COUNT(*)... query is executed before loading data in the data grid.

Default field type
Specifies the field type appearing in Field Editor 83 by default.

11.1.6 Data Grid

Below you will find a detailed description of the following data grid options.



Data Grid

Grid mode

● Switch to simple mode if query returns more than 5000 records

☑ Display notification message

○ Allways use full mode

Limit options

After opening the "Data" tab of the table editor:

○ Select all records from a table

● Select only 1000 records

Row numbers

☑ Display row numbers

Maximum digit count 4 (numbers from 1 to 9999)

☐ Display leading zeros

☐ Do not open separated connection

MS SQL Maestro provides you with two grid modes 329 of viewing data:
- Fool grid mode allows you to group, filter and sort data in a usual way.
- Simple mode is provided for working with large records number. For data fetching speed-up, filtering, sorting, and grouping features are not enabled in this mode.

You can use notification message to indicate simple mode.

Set the number of records to switch to simple mode automatically or select Always use full mode.

Limit options
Allows you either to select all records from table after opening the Data tab, or select only specified number of rows on one page with an ability to rotate pages and view all data.

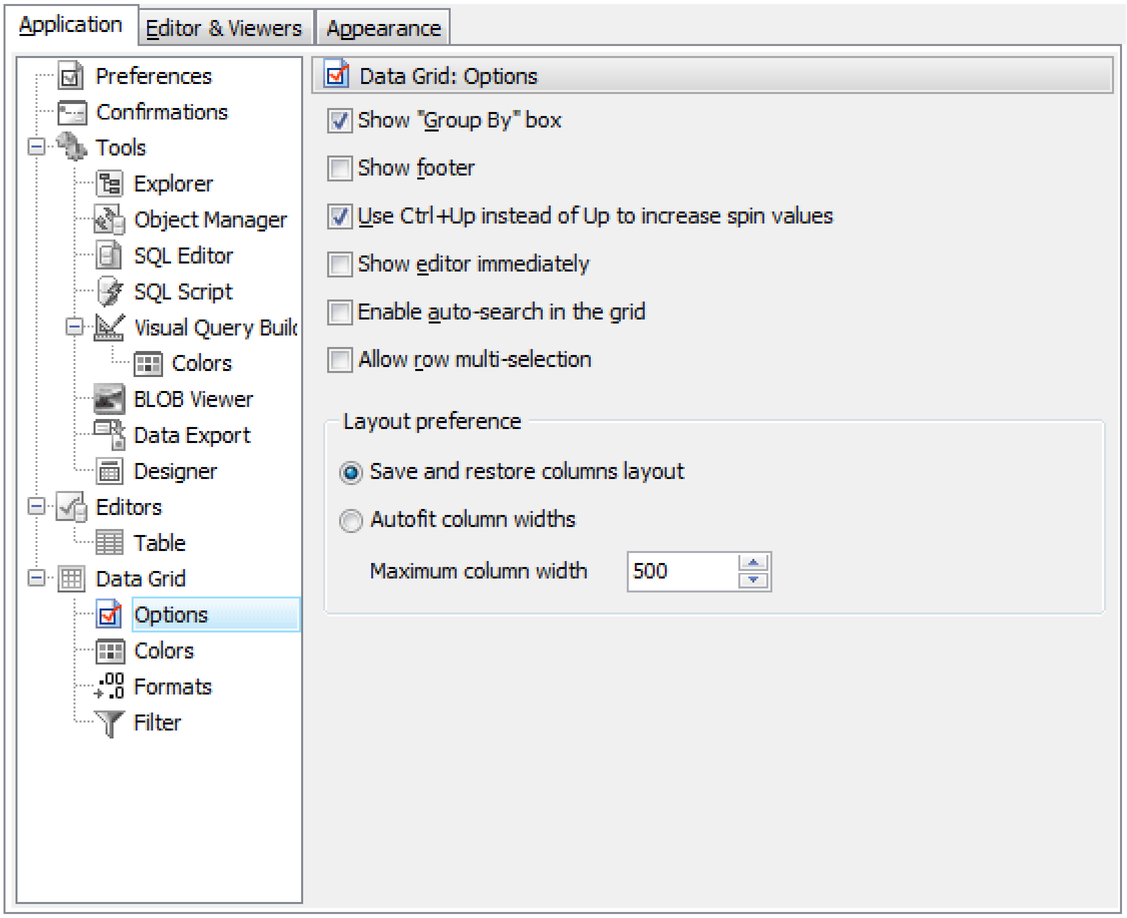
Row numbers
This options group allows you to manage grid rows numbering.
To enable/disable the numbering, use Display row numbers checkbox. You can set the number columns width with Maximum digit count. (I.e. for the value '3' the max column number will be 999).
For uniformity you can use the Display leading zeros option. With this option enabled and maximum digit count '3' you numbering column will be of the form: '001, 002, 003, ...'.

☑ Do not open separated connection

With this option enabled a new connections for fetching data is not opened. This gives you an ability to work with data a little bit faster, because time for opening a new connections is not demanded.

11.1.6.1 Options

Below you will find a detailed decryption of the data grid options.



Data Grid: Options

☑ Show "Group By" box

☐ Show footer

☑ Use Ctrl+Up instead of Up to increase spin values

☐ Show editor immediately

☐ Enable auto-search in the grid

☐ Allow row multi-selection

Layout preference

● Save and restore columns layout

○ Autofit column widths

Maximum column width 500

☑ Show "Group By" box

Shows the box on the top of the grid view for grouping data by fields.

☑ Show footer

Shows the footer on the bottom of the grid view.

☑ Use Ctrl+Up instead of Up to increase spin values

Allows you to use Ctrl+Up and Ctrl+Down key combinations for editing the spin for numeric fields.

☑ Show editor immediately

Allows editing the cell value right after the cell is clicked.

☑ Enable auto-search in the grid

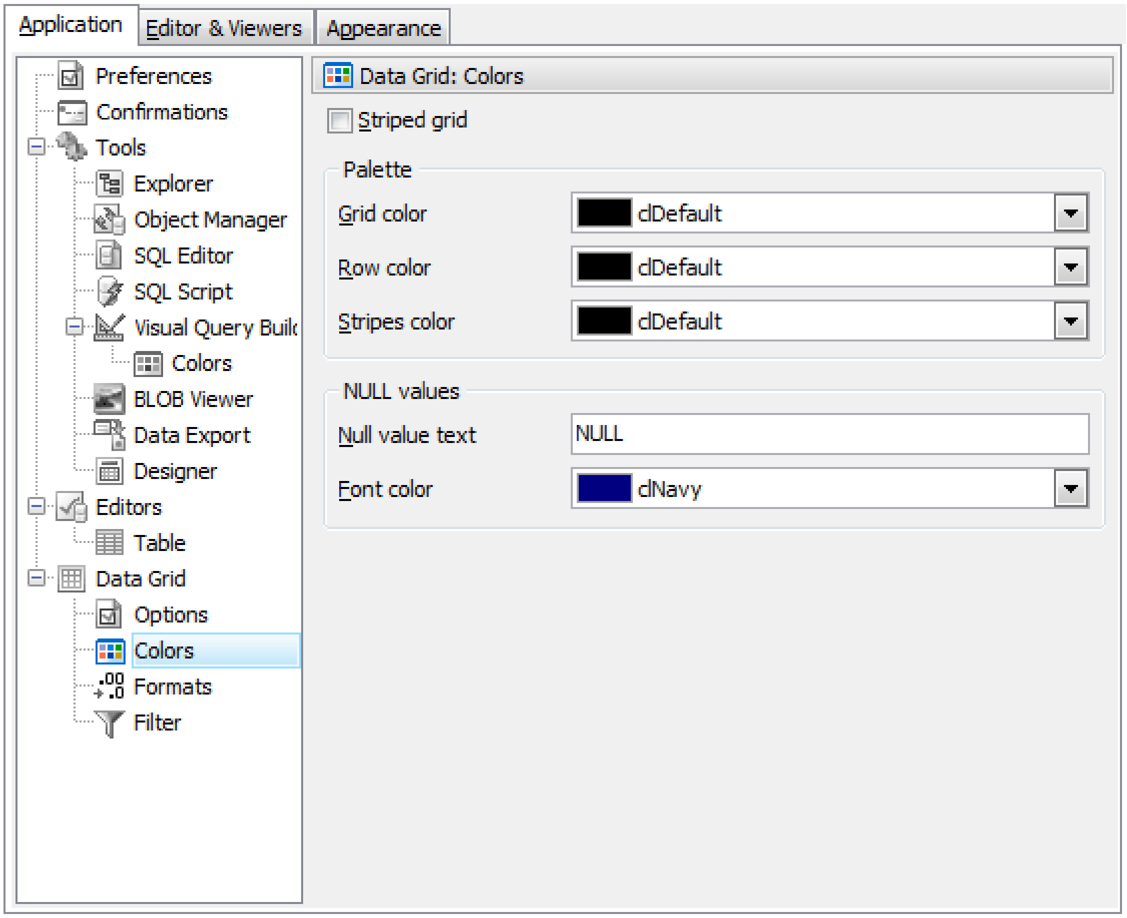
Allows you to search records in the grid by the first letters.

☑ Allow row multi-selection
Allows you to select multiple records using the Ctrl and Shift keys.

Layout preference
Select whether MS SQL Maestro should remember the column positions for the grids or fit them automatically.

11.1.6.2 Colors

Below you will find a detailed decryption of the following colors options.



☐ Striped grid

Palette

Grid color: clDefault

Row color: clDefault

Stripes color: clDefault

NULL values

Null value text: NULL

Font color: clNavy

☑ Striped grid
Displays the odd grid rows in a different color defined by the Stripes color option.

Grid color
Defines the background color of the data grid.

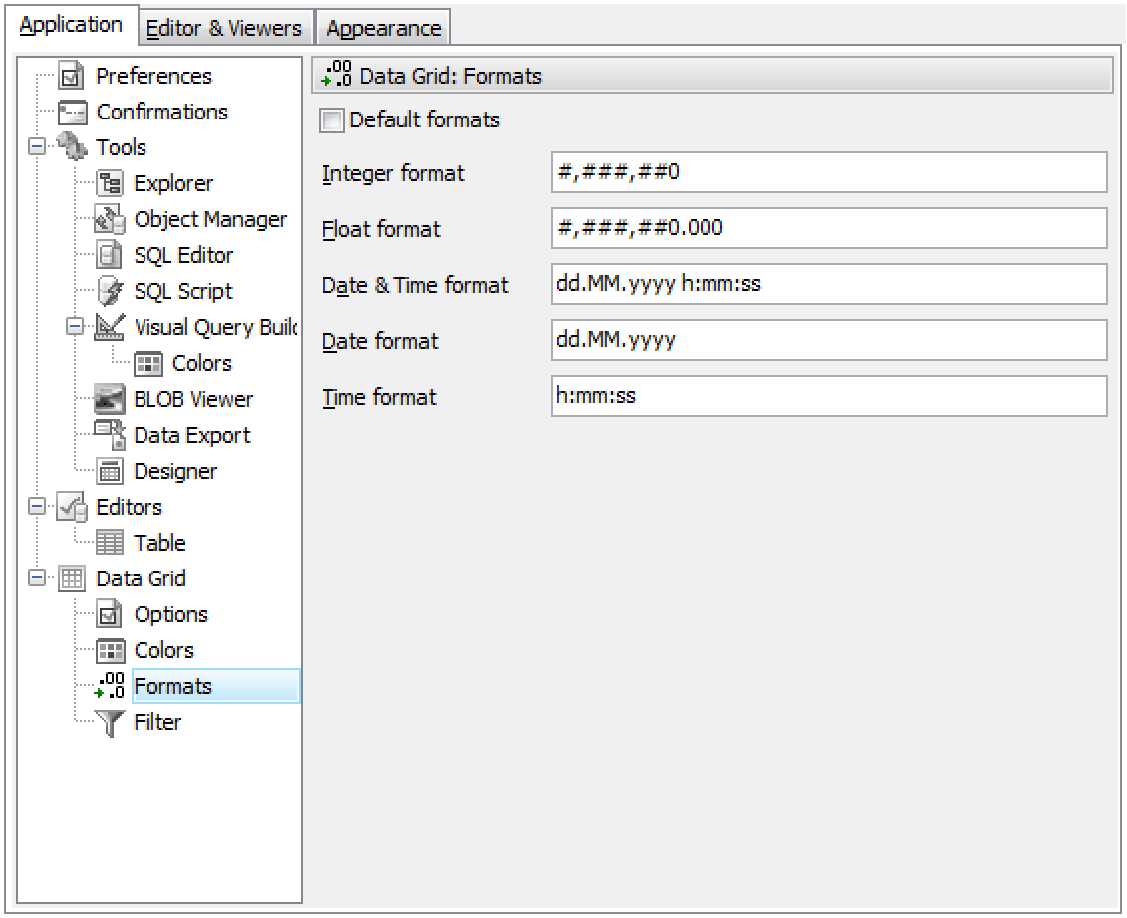
Row color
Defines the color of the selected row in the data grid.

Stripes color
Defines the color of the odd rows if the Striped Grid option is on.

Null values
Use Null value text to define the text that stand for the NULL value and use Font color to set the color for displaying this text.

11.1.6.3 Formats

Below you will find a detailed decryption of the following formats options.



☐ Default formats

Integer format: #,###,##0

Float format: #,###,##0.000

Date & Time format: dd.MM.yyyy h:mm:ss

Date format: dd.MM.yyyy

Time format: h:mm:ss

This window allows you to customize formats applied to data in grid. Edit the format masks to adjust the result format in the way you need.

In *numeric* formats using digit placeholder (# or 0) you can specify the format of number. For example, integer 1234567890 with # ### ##0 integer format is represented like 1 234 567 890. The locations of the leftmost '0' before the decimal point in the format string and the rightmost '0' after the decimal point in the format string determine the range of digits that are always present in the output string.

Conversion and their description for *date*, *time* and *date time* format:

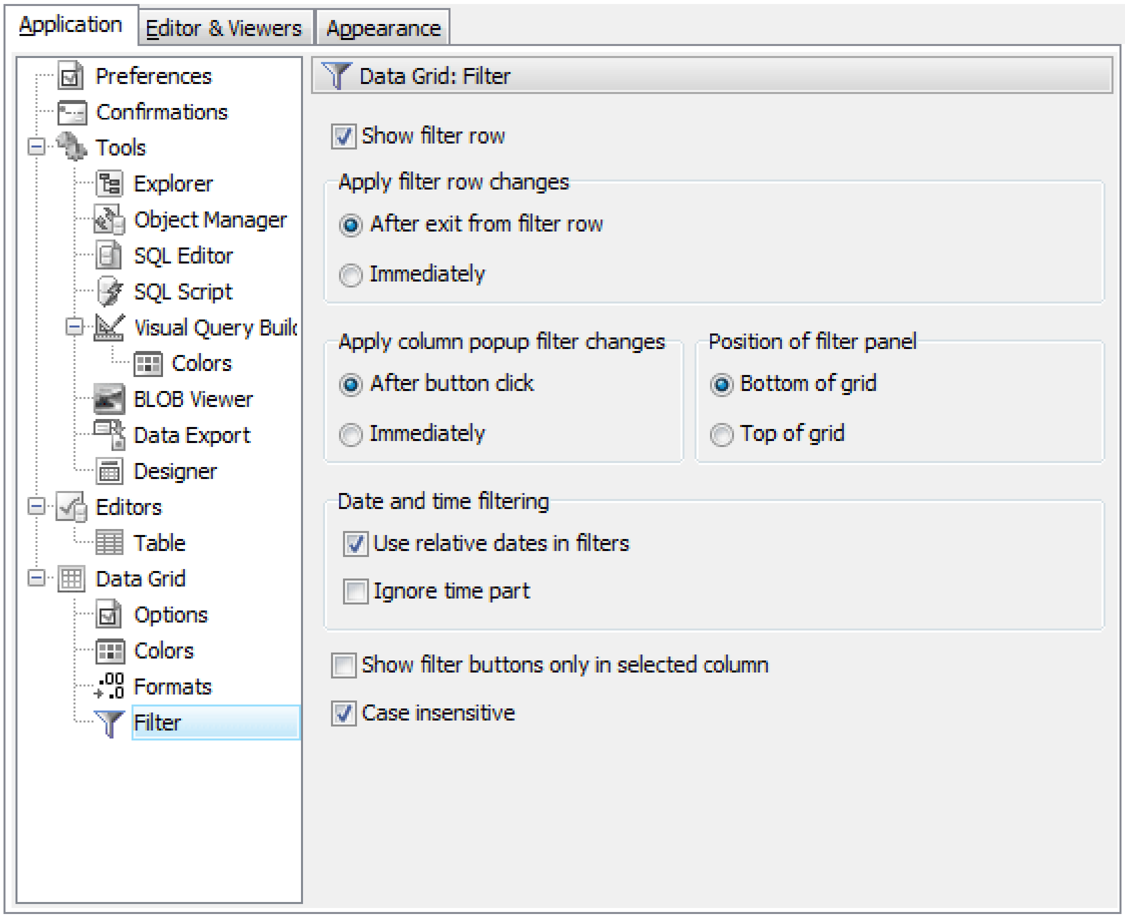
dd	day of the month, represented by 1 or 2 symbols. For example, the first day of month is 1
DD	day of the month, represented only by 2 symbols. For example, the first

	day of month is 01
mm	minutes
MM	month
yy	year, represented by 2 symbols. For example, 2006 year will be 06
yyyy	year, represented by 4 symbols. For example, 2006
h	hour, represented by 1 or 2 symbols. For example, 2
hh	hour, represented only by 2 symbols. For example, 02

"**.**", "**,**", "**:**" symbols can be placed as separators in format masks for *integer, float, date, time, date time, currency* types.

11.1.6.4 Filter

These options allows you to customize data filtering in grids.



Data Grid: Filter

☑ Show filter row

Apply filter row changes
● After exit from filter row
○ Immediately

Apply column popup filter changes
● After button click
○ Immediately

Position of filter panel
● Bottom of grid
○ Top of grid

Date and time filtering
☑ Use relative dates in filters
☐ Ignore time part

☐ Show filter buttons only in selected column

☑ Case insensitive

☑ Show filter row

When checked the filter row is always represented in the data grid as an additional row.

Apply filter row changes and Apply column popup filter changes allows you to manage the speed of the data filtering. To speed-up the process select Immediately as the time the filter you are set will be applied.

Change the Position of filter panel and customize timestamp data filtering: check the Use relative dates in filters box to include in column popup filter such options as "Yesterday", "Today", "Tomorrow", "Last 30 day", "Last week", "Next week", and others; check the Ignore time part box to neglect time part of timestamp data under the filtering.

By default filter buttons are shown at all columns header, to show filter button only in selected column, check the corresponding option.

You can specify the case sensitivity of the grid filter with the Case insensitive checkbox (ON by default).

11.2 Editors & Viewers

The Editors & Viewers section allows you to set the parameters of viewing and editing the SQL statements within MS SQL Maestro.

- General 447
- Display 448
- SQL highlight 449
- PHP highlight 451
- XML highlight 450
- Code Insight 452
- Code Folding 453

See also: SQL Editor 313, SQL Script Editor 364, Visual Query Builder 318, Table Editor 79.

11.2.1 General

If the Auto indent option is checked, each new indention is the same as the previous when editing SQL text.

☑ Insert mode
If this option is checked, insert symbols mode is default on.

☑ Use syntax highlight
Enables syntax highlight in the object editor window.

☑ Always show links
If this option is checked, hyperlinks are displayed in the editor window. To open a link click it with the **Ctrl** button pressed.

☑ Show line numbers
If this option is checked, line numbers are displayed in the editor window.

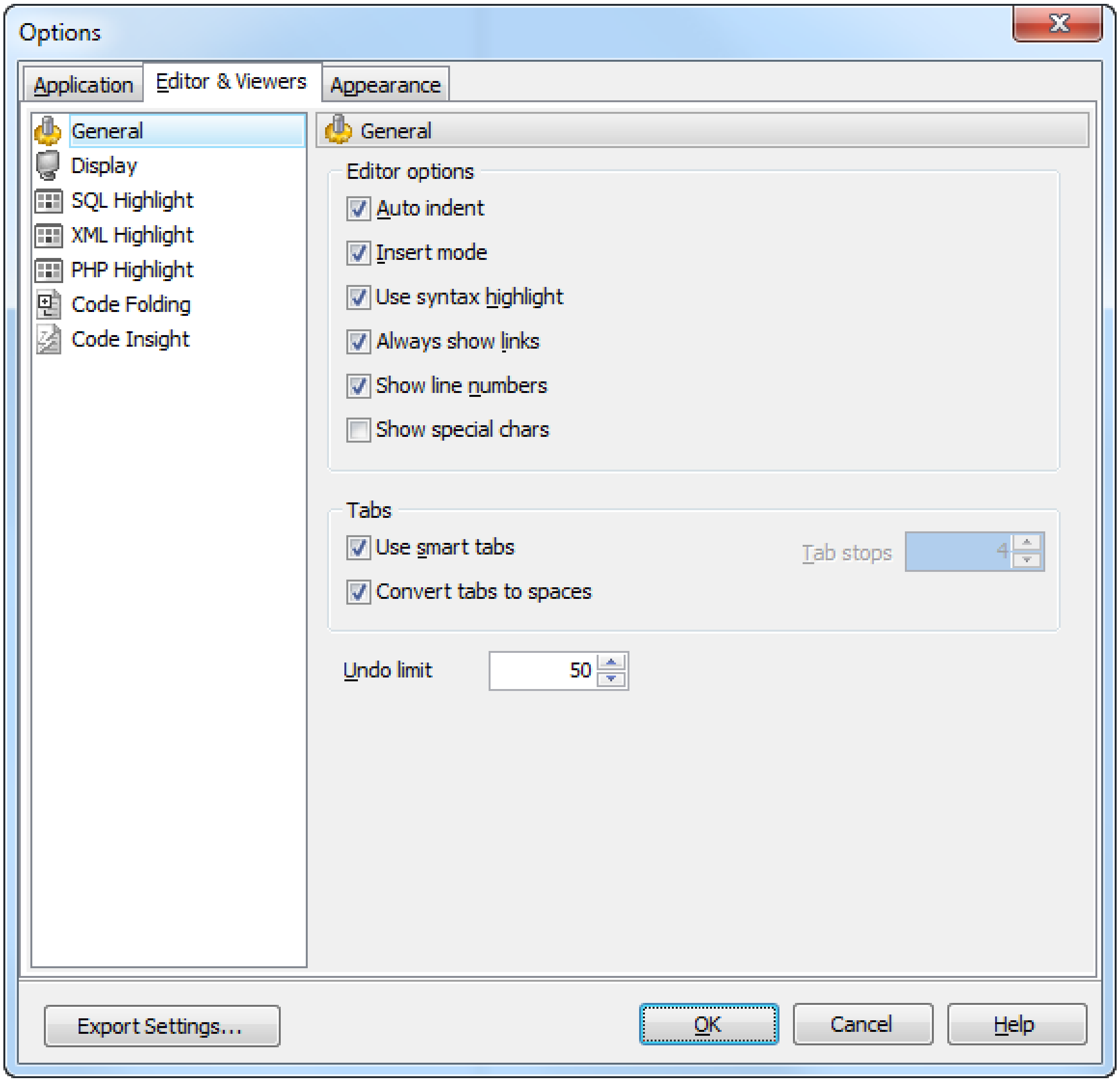
☑ Show special chars
If this option is checked, special chars (like line breaks) are displayed in the editor window.

☑ Use smart tabs
With this option on the number of tab stops is calculated automatically, depending on the previous line tab.

☑ Convert tabs to spaces
If this option is checked, each time you press the Tab key, the appropriate number of spaces will be added to the edited text.

Tab Stops
Defines the tab length, used when editing text.

Undo Limit
Defines the maximum number of changes possible to be undone.



Editor options
☑ Auto indent
☑ Insert mode
☑ Use syntax highlight
☑ Always show links
☑ Show line numbers
☐ Show special chars

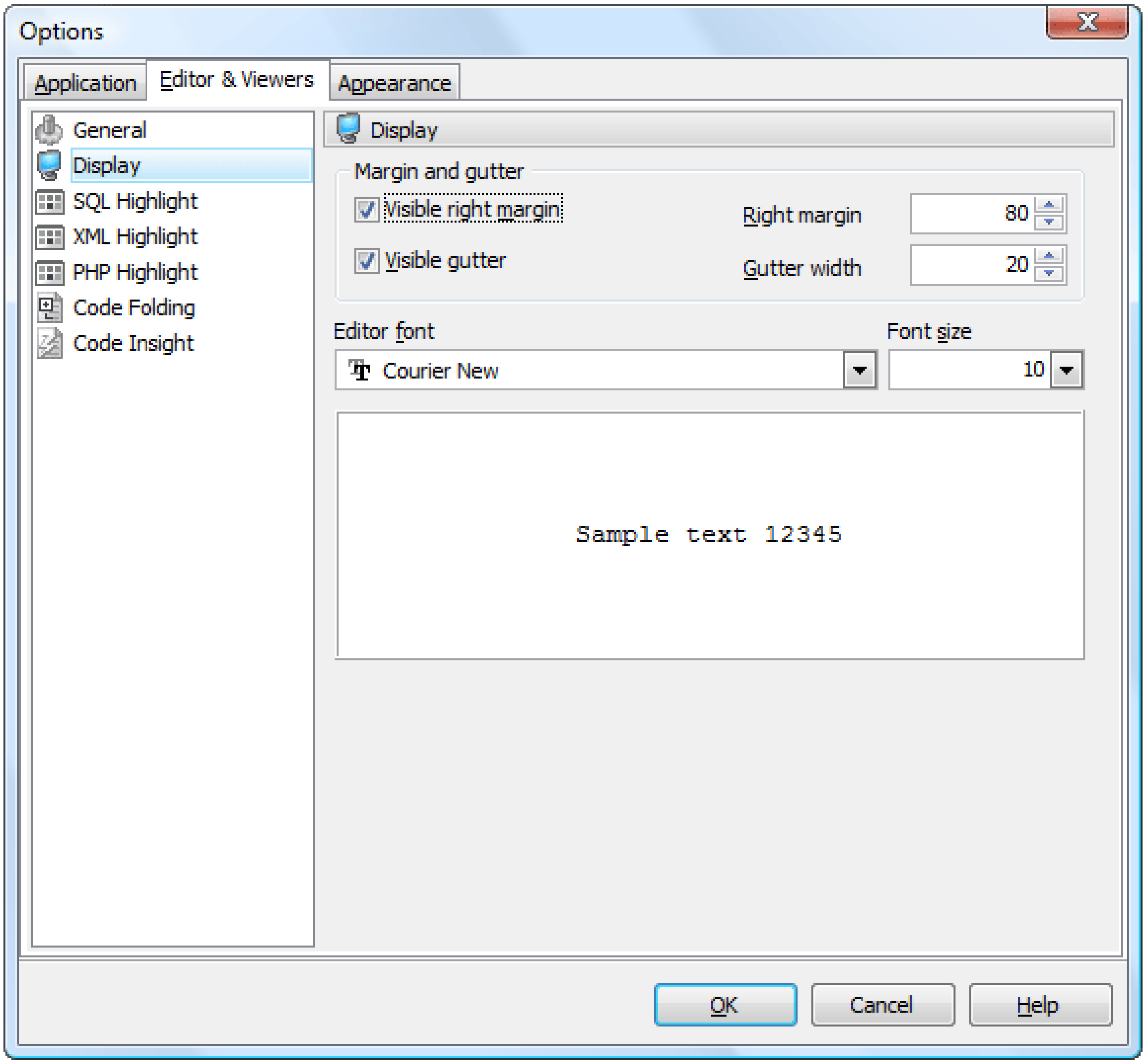
Tabs
☑ Use smart tabs
☑ Convert tabs to spaces
Tab stops 4

Undo limit 50

11.2.2 Display

You can disable/enable the right text margin and the gutter of the editor area, set the position of the right text margin as Right margin, and the Gutter width.

Use the Editor font and Font size to define the font used in all program editors and viewers. The panel below displays the sample of the selected font.



Display

Margin and gutter

☑ Visible right margin

Right margin: 80

☑ Visible gutter

Gutter width: 20

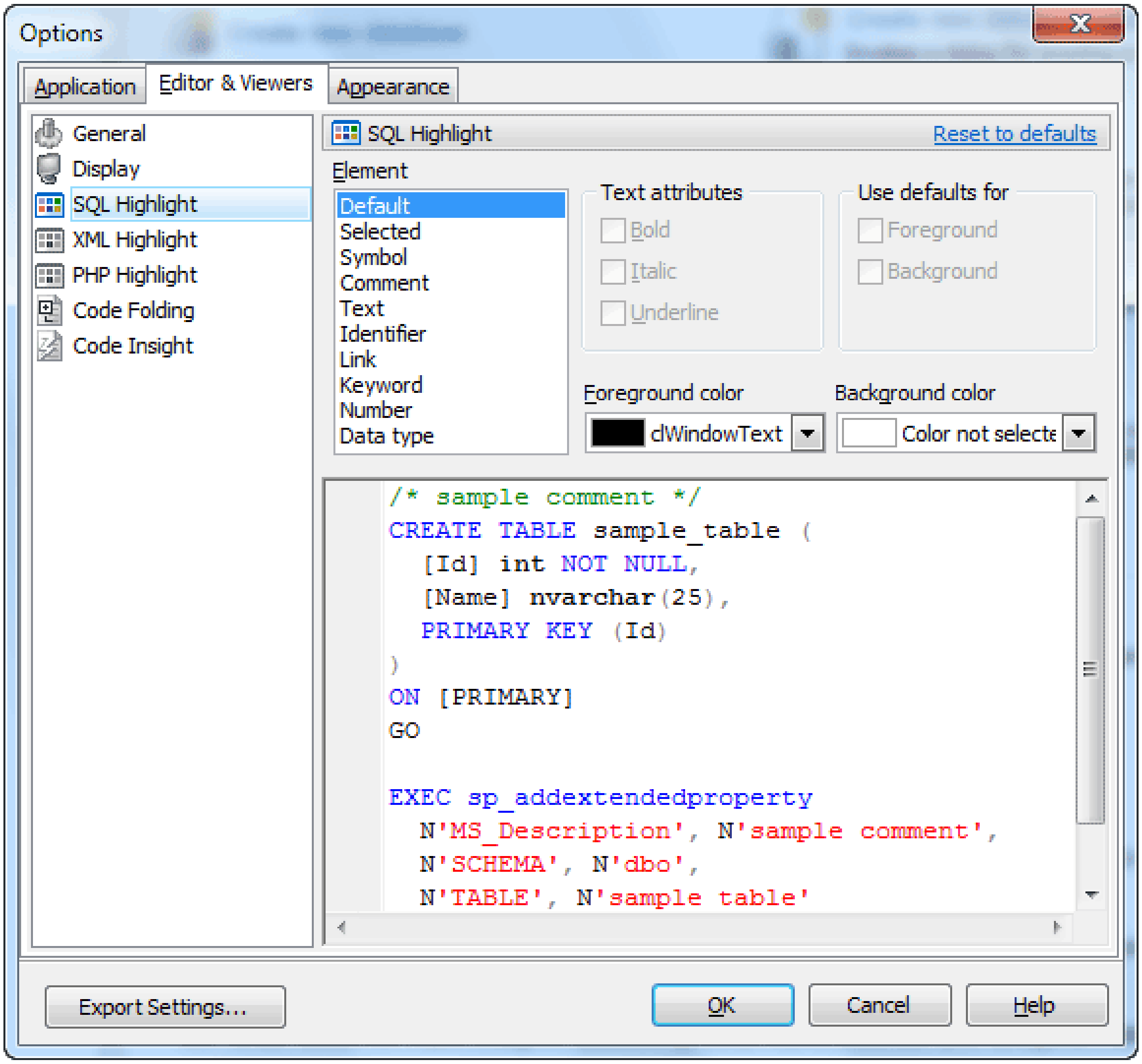
Editor font: Courier New

Font size: 10

```
Sample text 12345
```

11.2.3 SQL highlight

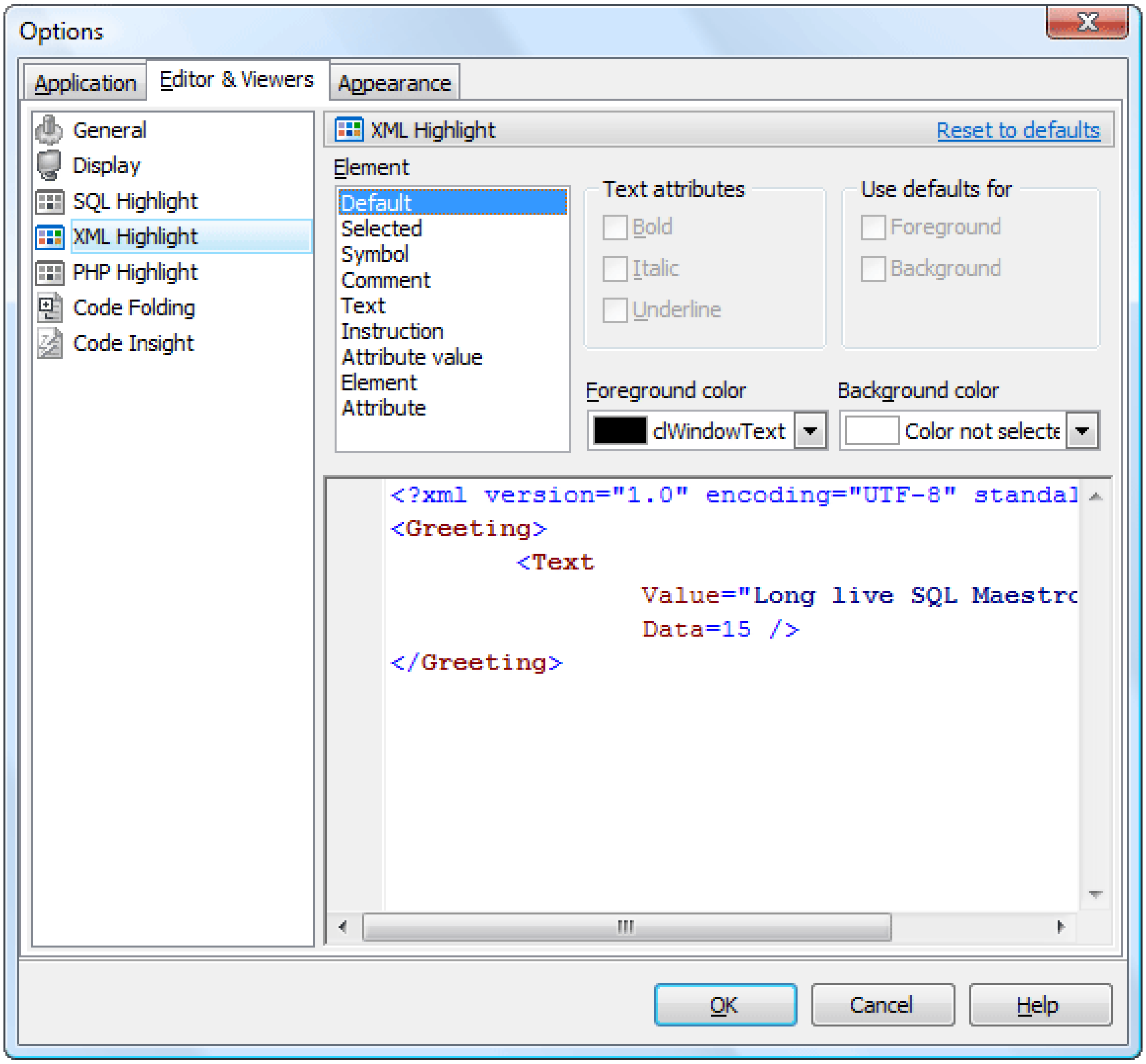
Use the SQL highlight item to customize syntax highlight in all SQL editors and viewers, e.g. in SQL Editor, Query Builder, Table Editor and others. Select the text element from the list, e.g. *comment* or *SQL keyword* and adjust its foreground color, background color and text attributes according to your preferences.



```
/* sample comment */
CREATE TABLE sample_table (
  [Id] int NOT NULL,
  [Name] nvarchar(25),
  PRIMARY KEY (Id)
)
ON [PRIMARY]
GO

EXEC sp_addextendedproperty
  N'MS_Description', N'sample comment',
  N'SCHEMA', N'dbo',
  N'TABLE', N'sample_table'
```

11.2.4 XML highlight

Use the XML highlight item to customize XML syntax highlight for the text representation of BLOBs in BLOB Viewer/Editor. Select the text element from the list, e.g. attribute or attribute value and adjust its foreground color, background color and text attributes according to your wishes.



Element

- Default
- Selected
- Symbol
- Comment
- Text
- Instruction
- Attribute value
- Element
- Attribute

Text attributes

☐ Bold
☐ Italic
☐ Underline

Use defaults for

☐ Foreground
☐ Background

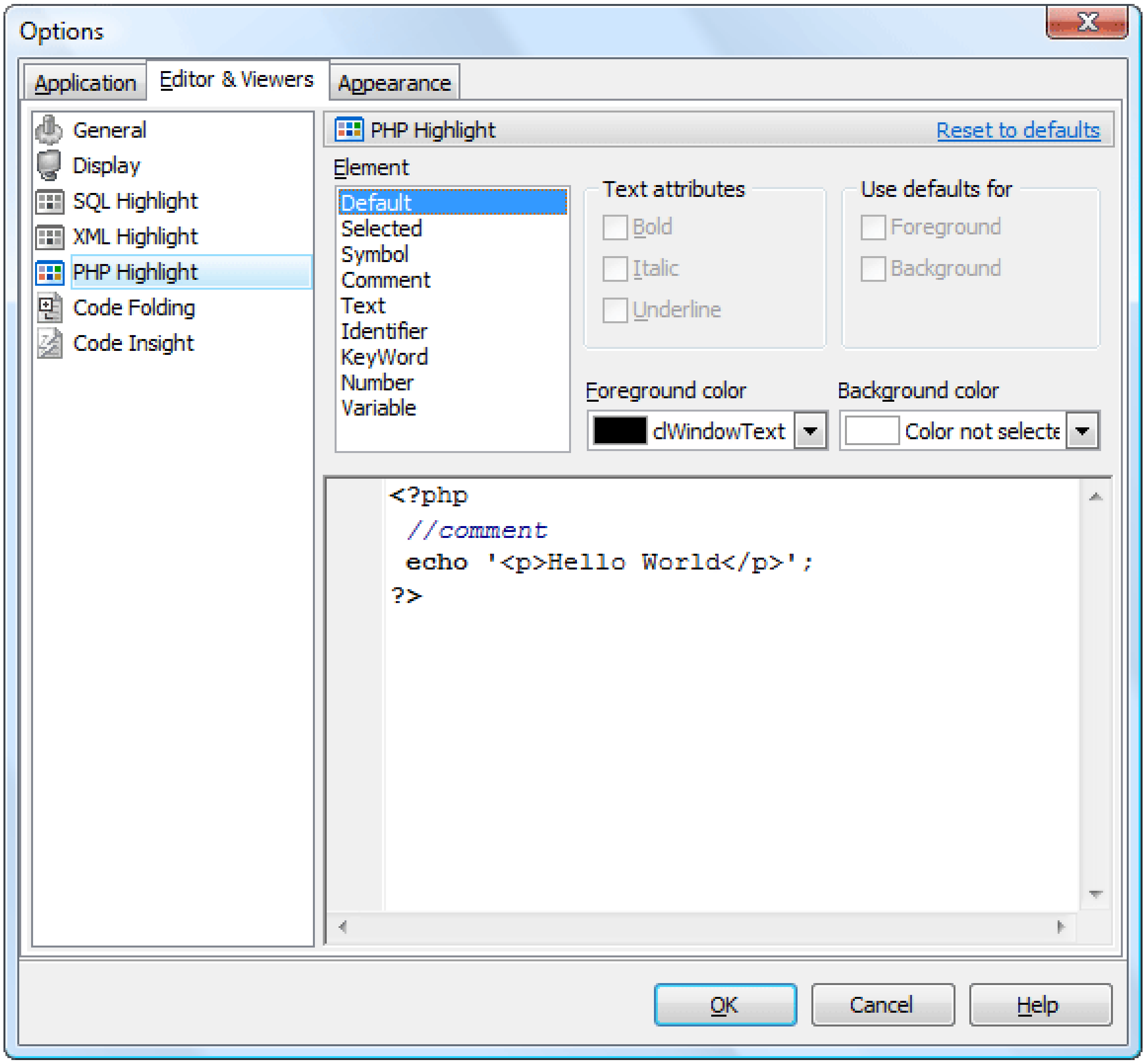
Foreground color: clWindowText

Background color: Color not selecte

```
<?xml version="1.0" encoding="UTF-8" standal
<Greeting>
        <Text
                Value="Long live SQL Maestro
                Data=15 />
</Greeting>
```

11.2.5 PHP highlight

Use the PHP highlight item to customize PHP syntax highlight for the text representation of BLOBs in BLOB Viewer/Editor. Select the text element from the list (e.g. Keyword, Comment, Identifier), and adjust its foreground color, background color and text attributes according to your wishes.



Element: Default

☐ Bold
☐ Italic
☐ Underline

☐ Foreground
☐ Background

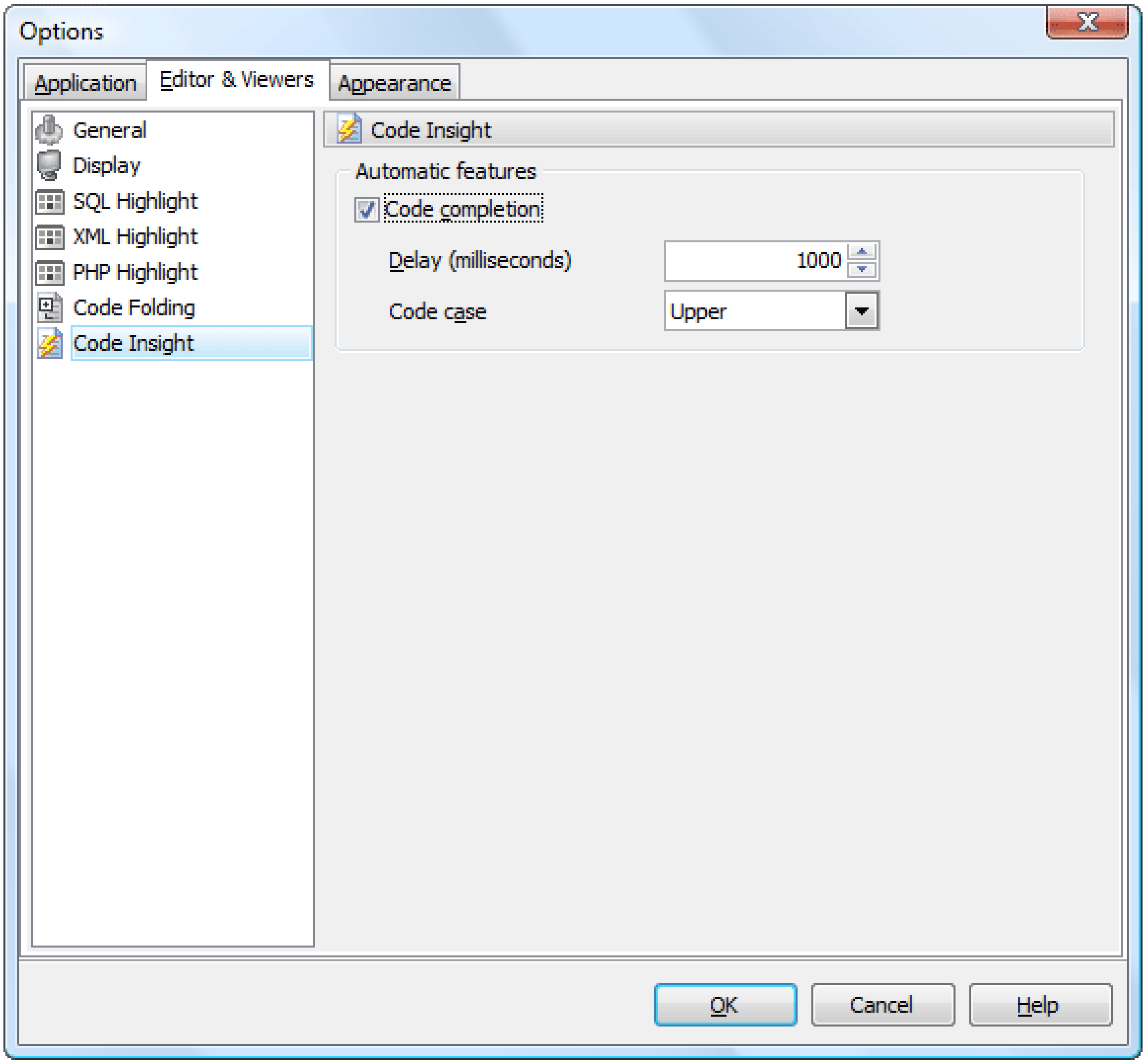
Foreground color: clWindowText

Background color: Color not selecte

```
<?php
 //comment
 echo '<p>Hello World</p>';
?>
```

11.2.6 Code Insight

You can disable/enable the code completion with the corresponding option and also set the time is appears as Delay, and case of words inserted automatically.



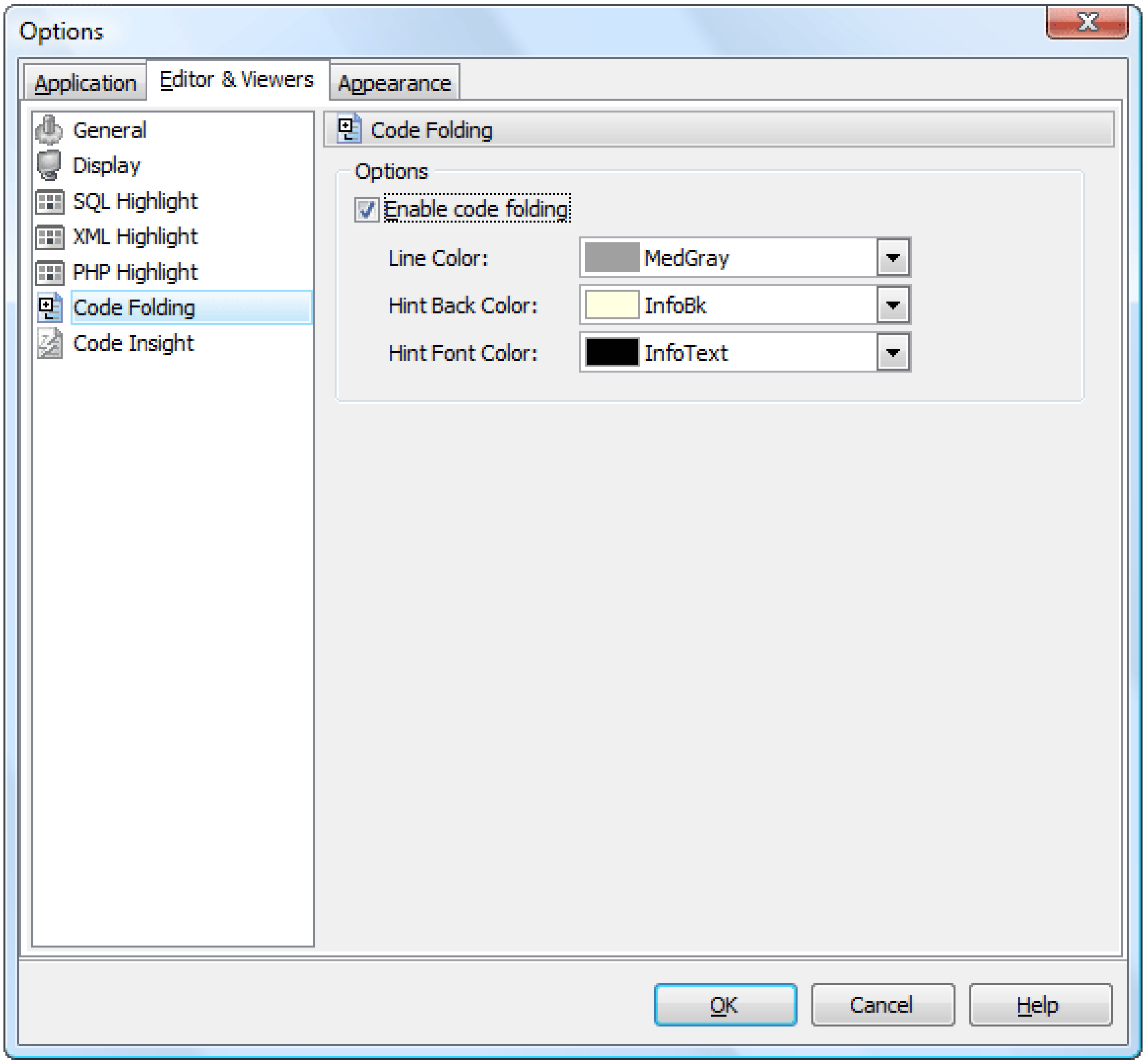
☑ Code completion

Delay (milliseconds) 1000

Code case Upper

11.2.7 Code Folding

The Code Folding item group makes it possible both to view the whole text and to divide it into logical parts (regions). Each part can be collapsed and extended. In extended mode the whole text is displayed (set by default), in collapsed mode the text is hidden behind one text line denoting the first line of the collapsed region.



☑ Enable code folding

Line Color: MedGray

Hint Back Color: InfoBk

Hint Font Color: InfoText

You can enable/disable code folding in SQL editors and viewers and customize the colors of its items.

11.3 Appearance

The Appearance section allows you to customize the application interface style to your preferences.

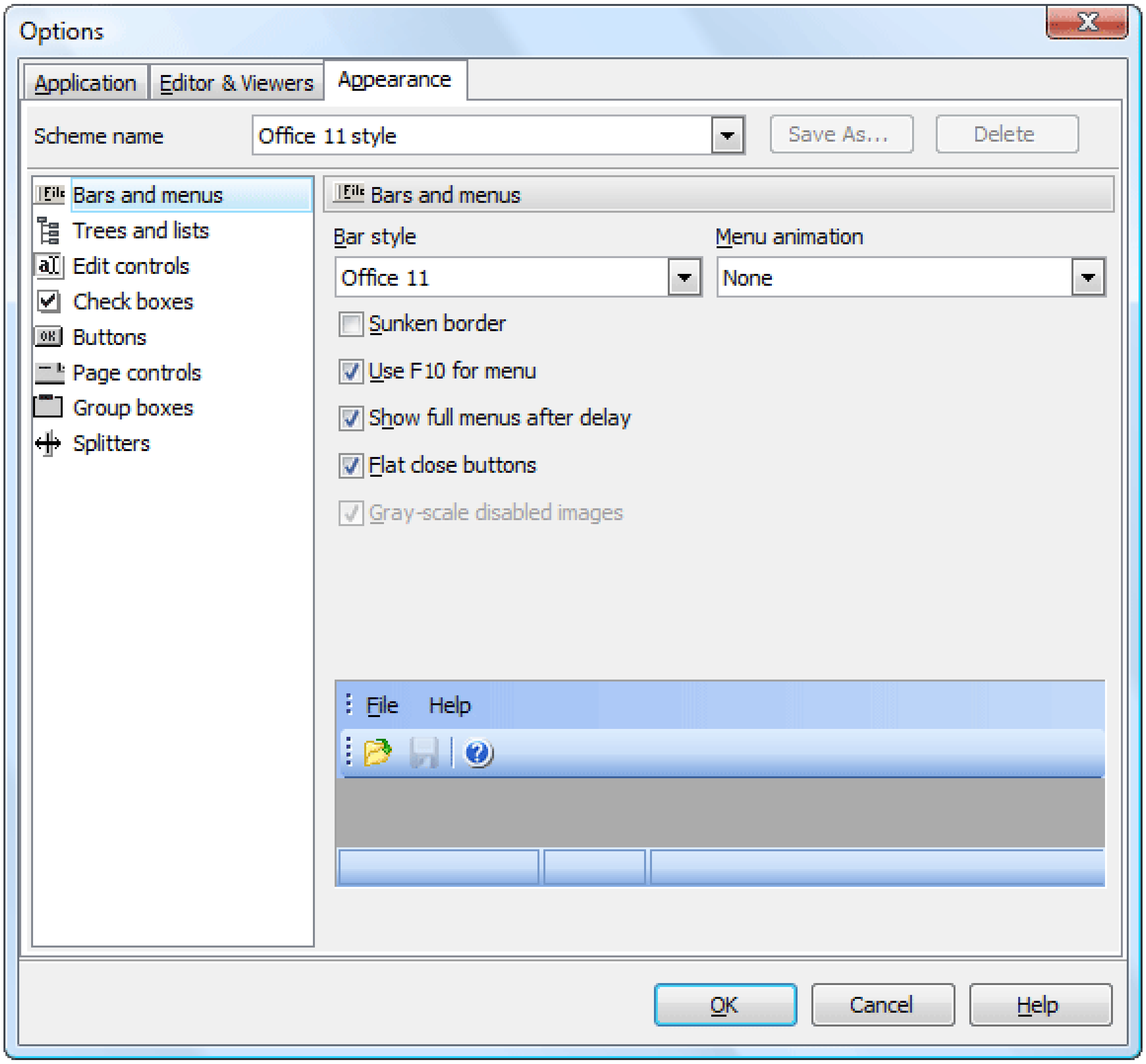
Use the Scheme name box to select the interface scheme you prefer: *Office XP style*, *Windows XP native style*, etc. You can create your own interface schemes by customizing any visual options (Bars and menus, Trees and lists, Edit controls, Check boxes, Buttons, etc.) and clicking the Save As button. All the customized options are displayed on the sample panel.

- Bars and menus 455
- Trees and lists 456
- Edit controls 457
- Check boxes 458
- Buttons 459
- Page controls 460
- Group boxes 461
- Splitters 462

11.3.1 Bars and menus

Use the Bars and menus item to customize MS SQL Maestro toolbars style and menus animation.

The item allows you to select Bar style and menu animation from the corresponding drop-down lists and to enable or disable such options as sunken border, F10 key for opening menu, viewing full menus after delay, flat close buttons, gray-scale images.



Scheme name: Office 11 style

Bars and menus

Bar style: Office 11

Menu animation: None

☐ Sunken border

☑ Use F10 for menu

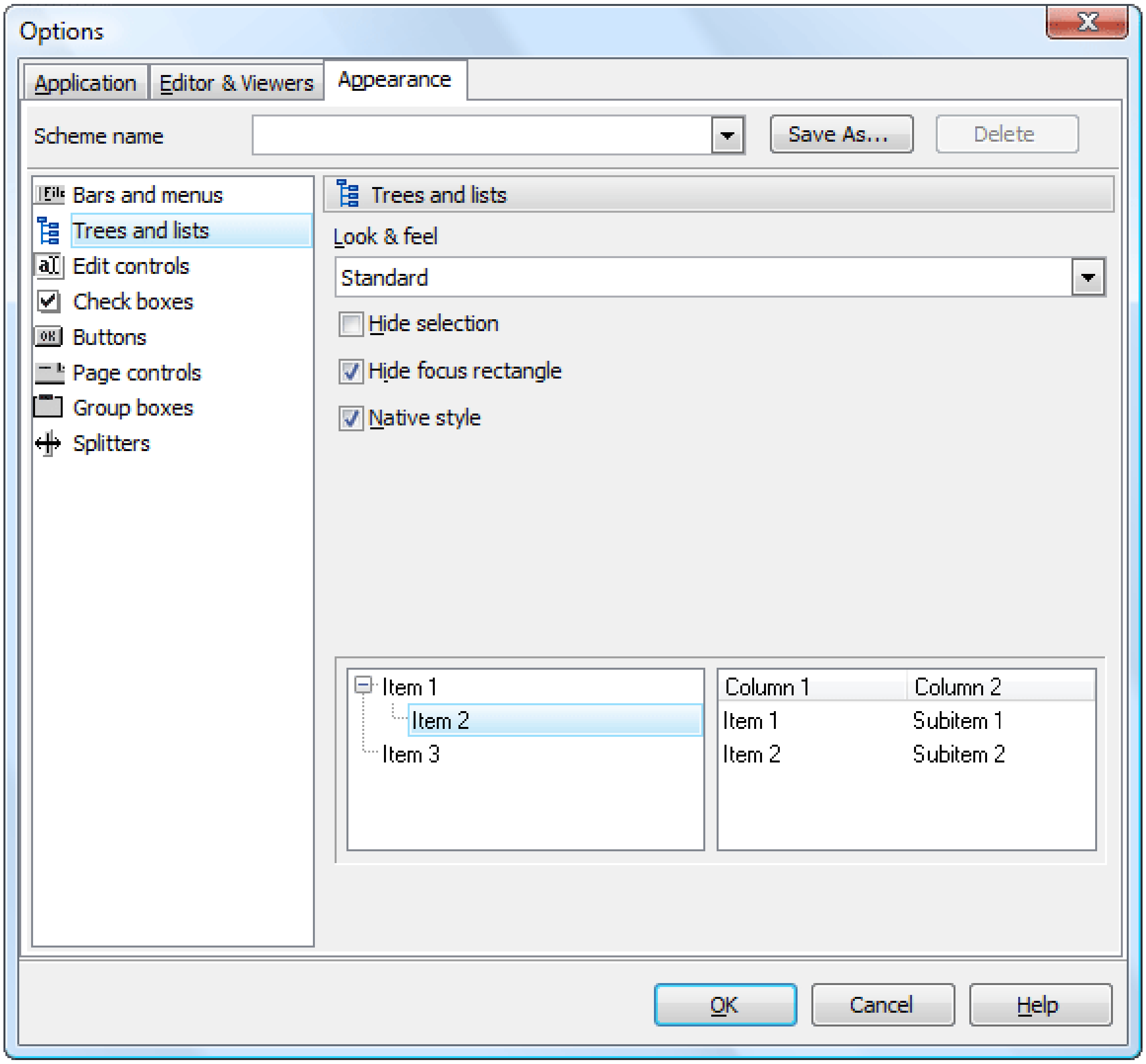
☑ Show full menus after delay

☑ Flat close buttons

☑ Gray-scale disabled images

11.3.2 Trees and lists

Use the Trees and lists item to select various tree view options. Use the item to select *standard*, *flat* or *ultraflat* styles, check or uncheck the *hide selection*, *hide focus rectangle* and *native style* options.



Look & feel: Standard

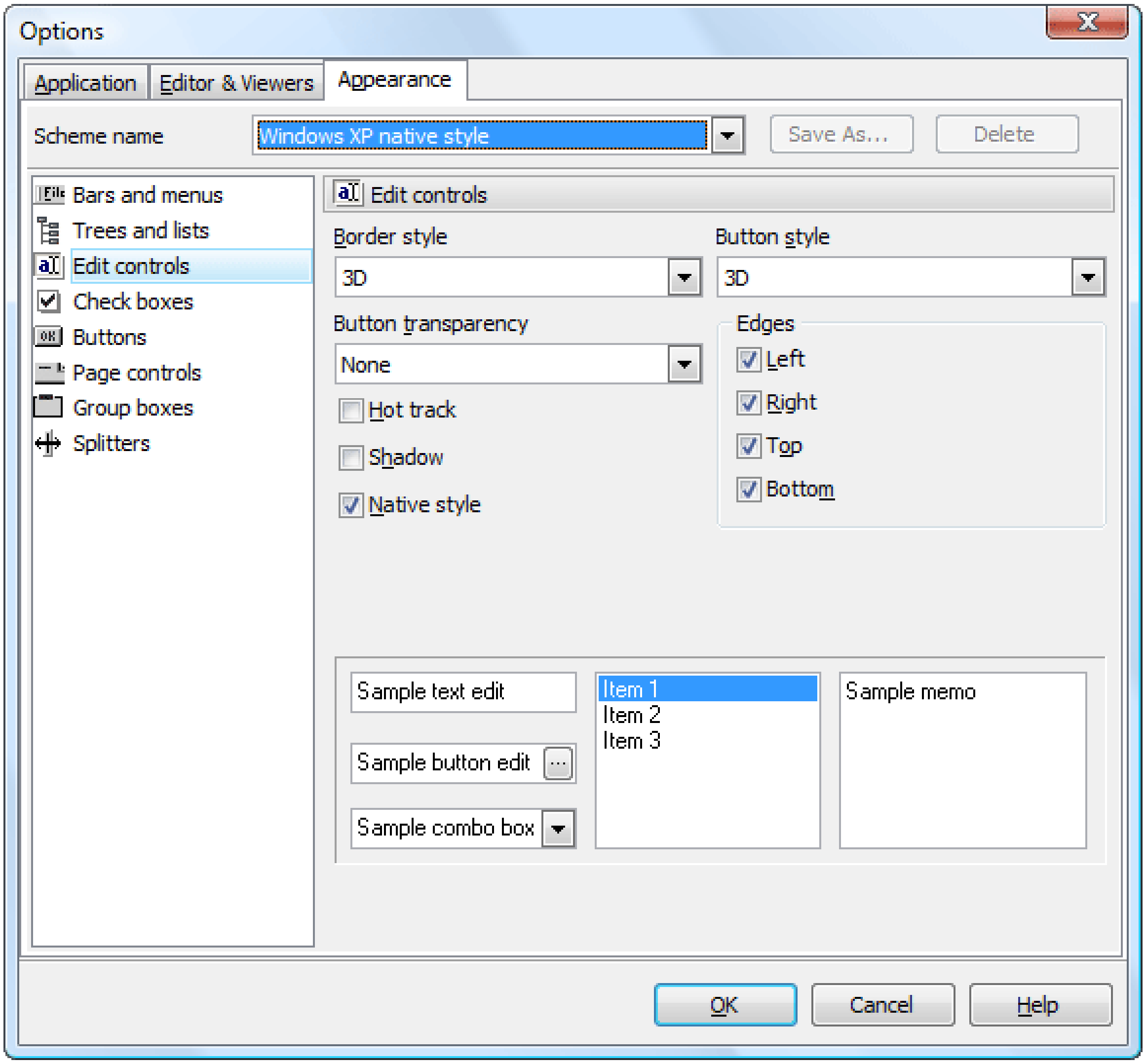
☐ Hide selection

☑ Hide focus rectangle

☑ Native style

11.3.3 Edit controls

Use the Edit controls item to customize the appearance of different MS SQL Maestro edit controls. The tab allows you to select the edit controls border style, button style and transparency, enable/disable hot tracks, shadows, native style and customize edges. It is also possible to define samples for the text edit, button edit and combo box controls.



Scheme name: Windows XP native style

Border style: 3D

Button style: 3D

Button transparency: None

☐ Hot track

☐ Shadow

☑ Native style

Edges

☑ Left

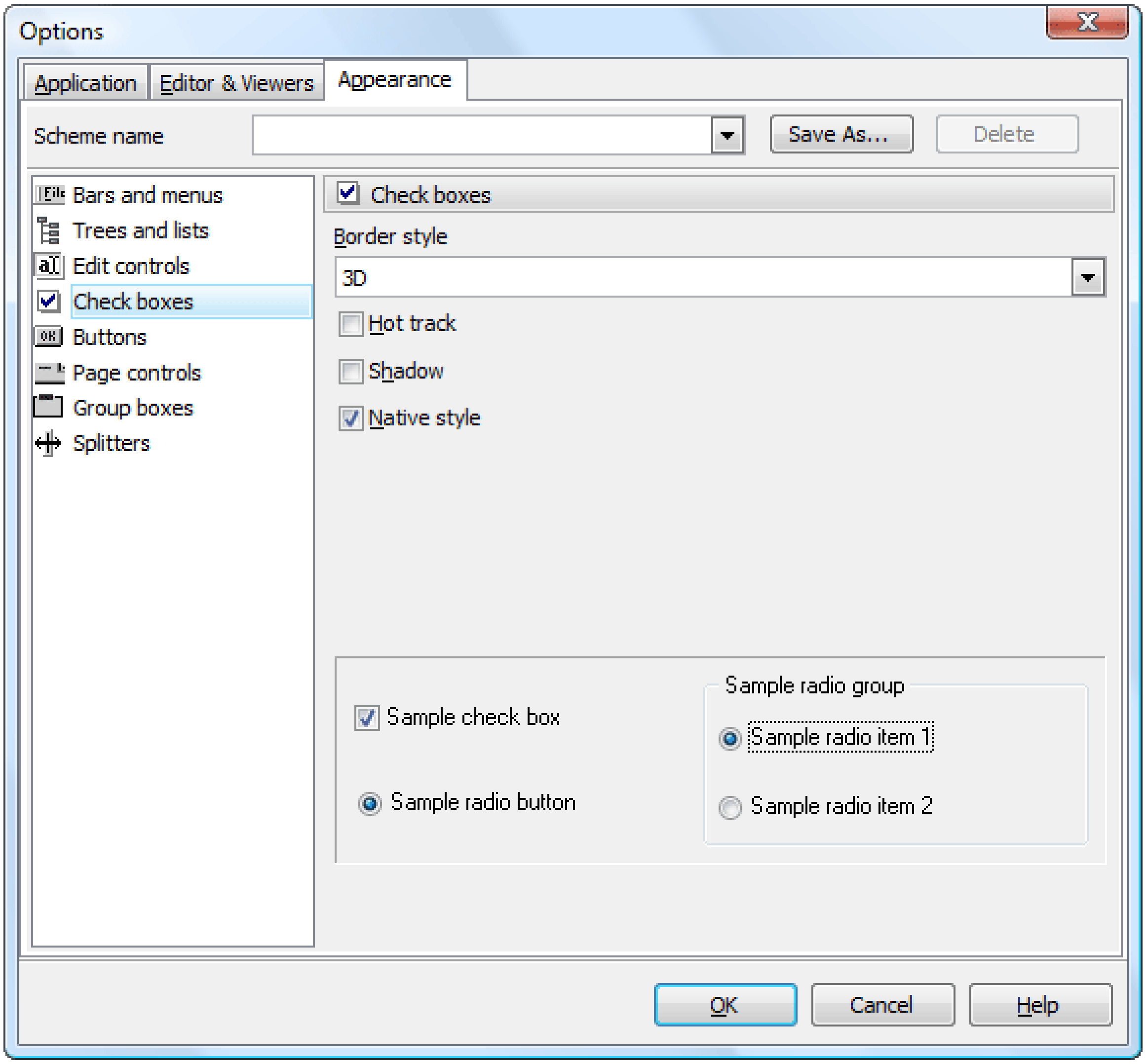
☑ Right

☑ Top

☑ Bottom

11.3.4 Check boxes

The Check boxes item allows you to customize the appearance of check boxes and radio buttons. The tab allows you to customize the appearance of check boxes: set border style, enable/disable hot tracks, shadows, native style. It is also possible to define samples for check boxes and radio buttons.



☑ Check boxes

Border style: 3D

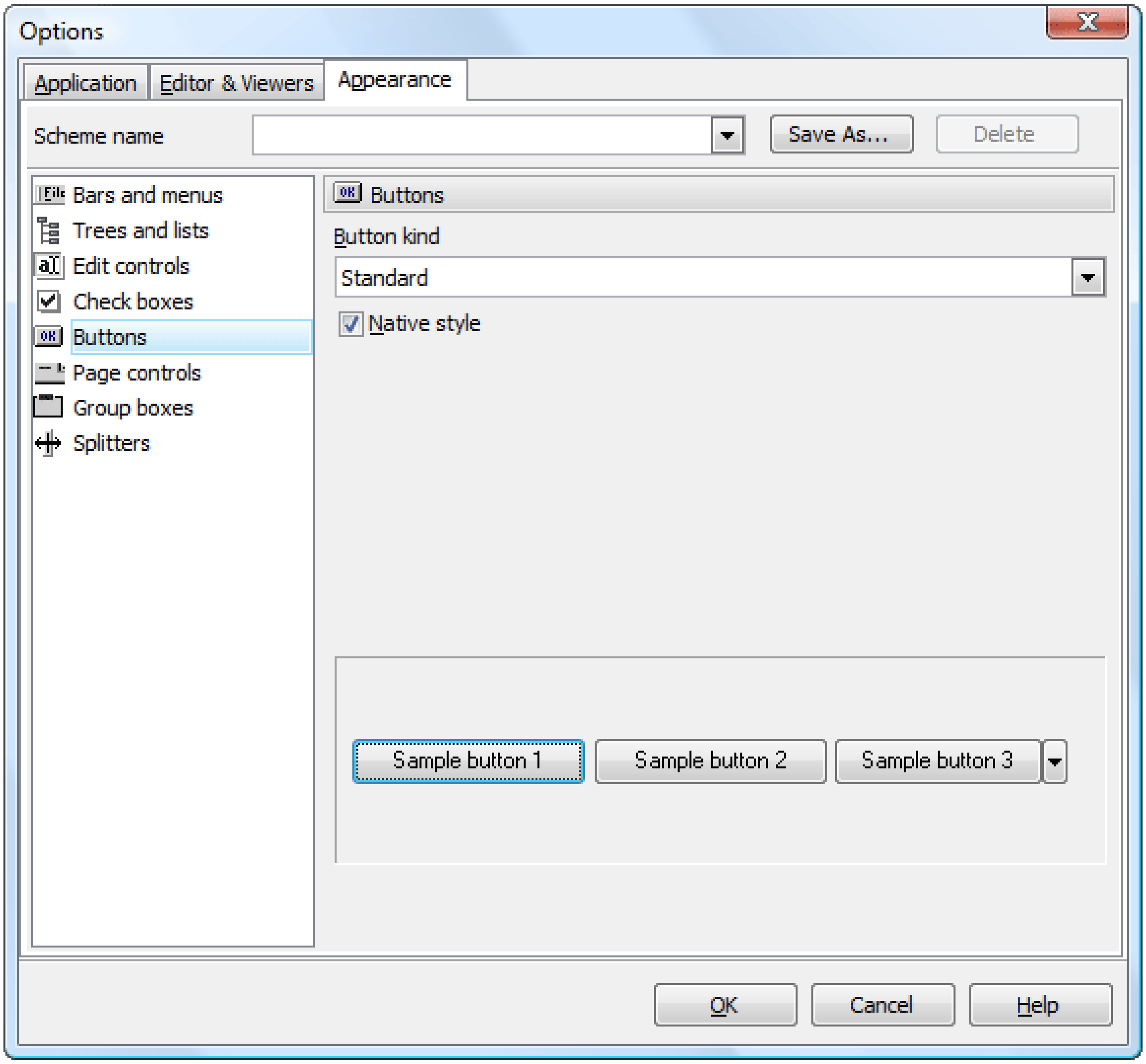
☐ Hot track

☐ Shadow

☑ Native style

11.3.5 Buttons

Use the Buttons item to customize MS SQL Maestro buttons. The tab allows you to adjust the appearance of buttons and define sample buttons as well.

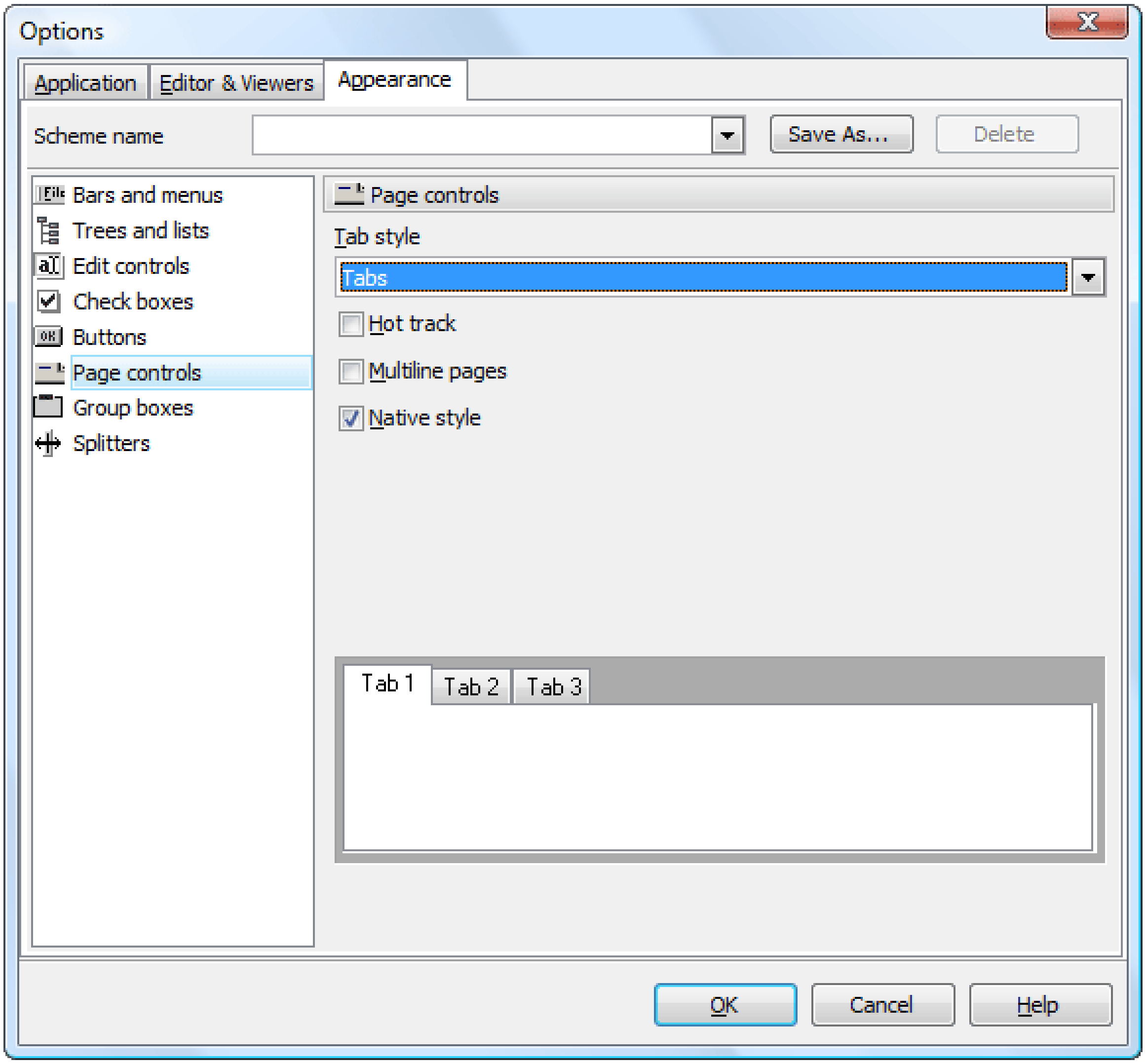


Button kind: Standard

☑ Native style

11.3.6 Page controls

The Page controls item allows you to customize the style of all MS SQL Maestro page controls. The tab allows you to select tab styles, enable/disable hot track, multi-line pages and native style.



Tab style: Tabs

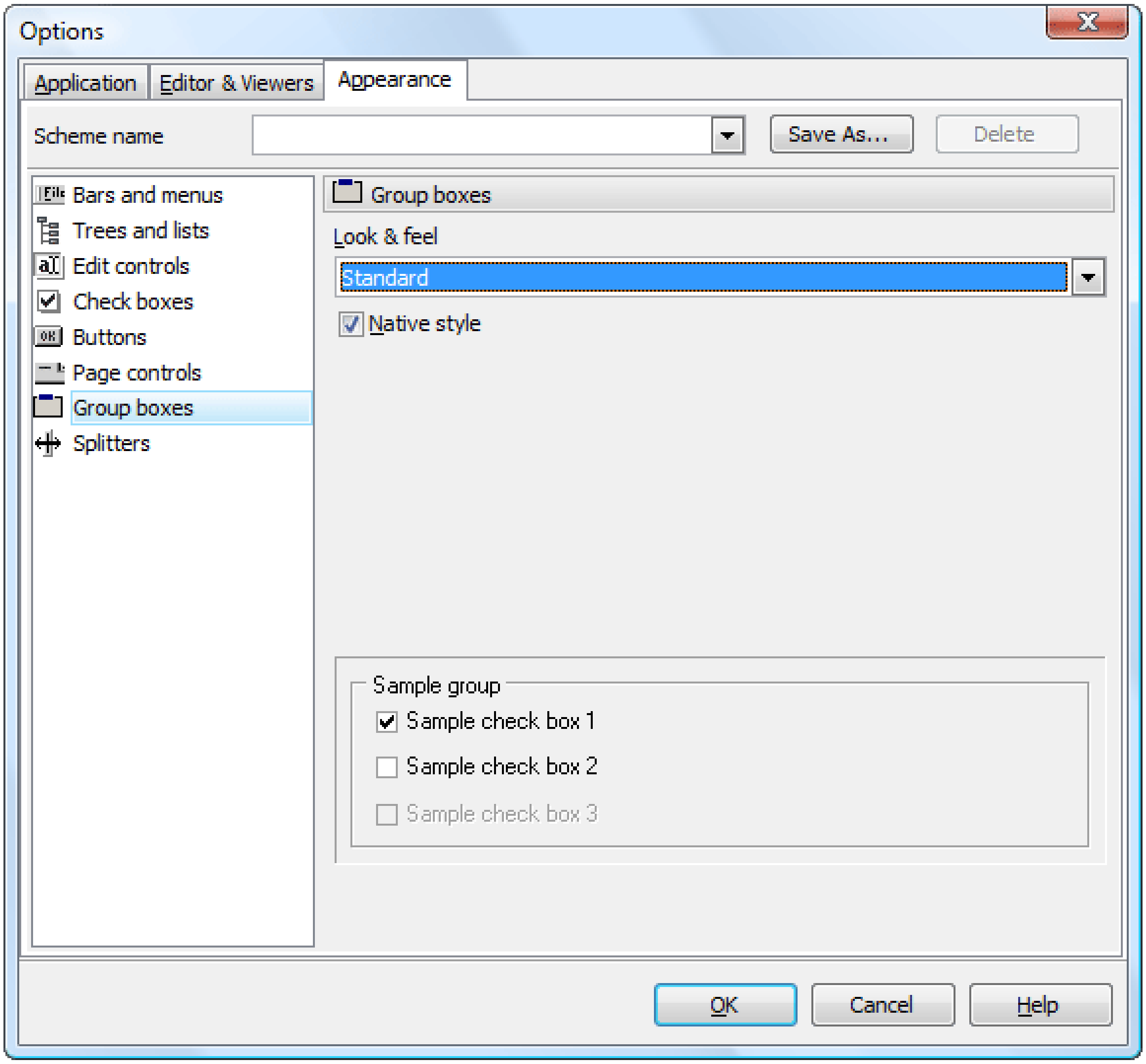
☐ Hot track

☐ Multiline pages

☑ Native style

11.3.7 Group boxes

Use the Group boxes item to customize all MS SQL Maestro group boxes according to your preferences. Use tab to apply styles for group boxes, enable/disable native style and define samples.



Scheme name:

Look & feel: Standard

☑ Native style

Sample group

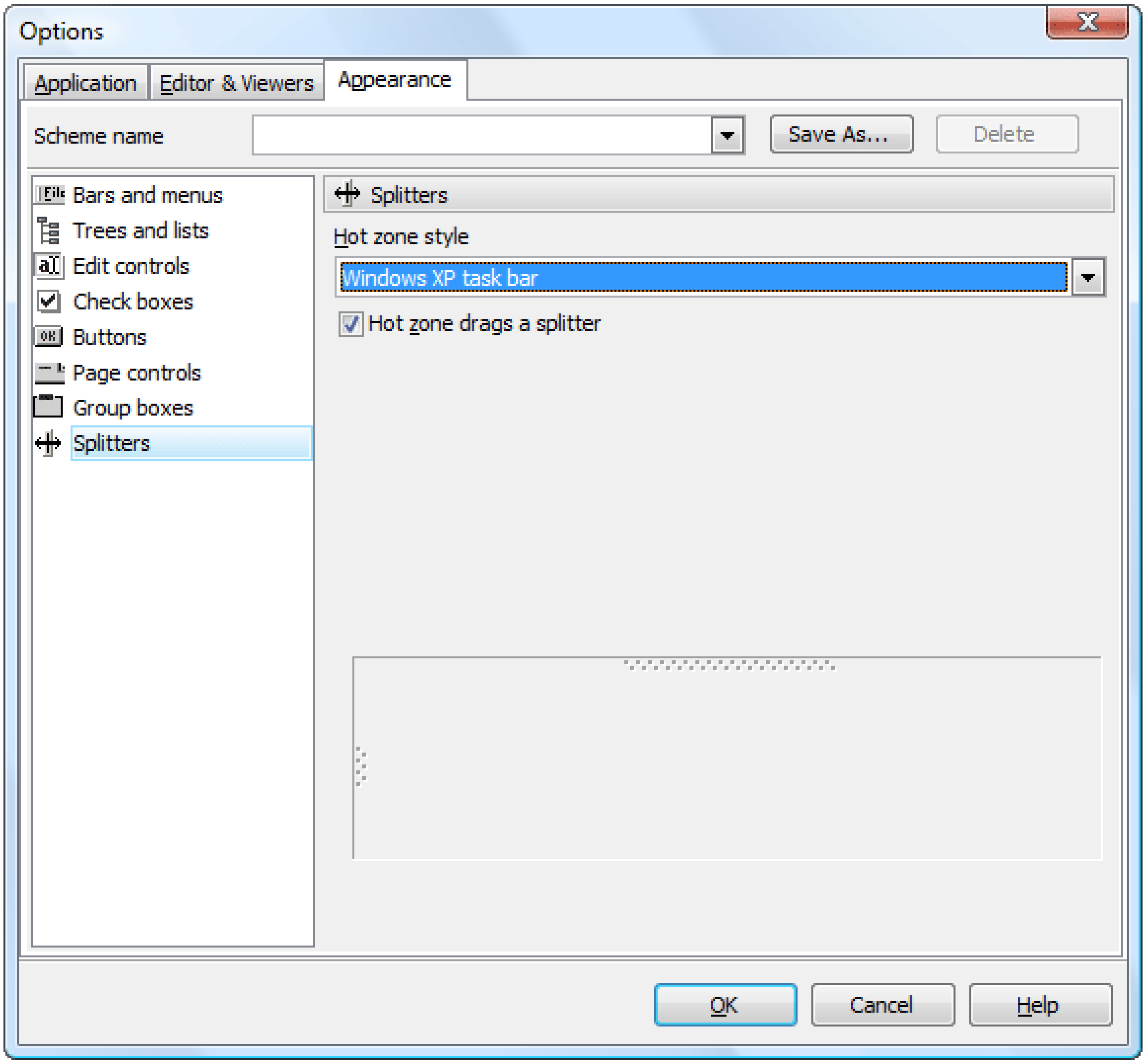
☑ Sample check box 1

☐ Sample check box 2

☐ Sample check box 3

11.3.8 Splitters

Use the Splitters item to customize all MS SQL Maestro splitters according to your preferences. Use the tab to select hot zone style (*Windows XP task bar*, *Media Player 8*, *Media Player 9*, *Simple* or *none*) and specify the Hot zone drags a splitter option.



Options

Scheme name

Splitters

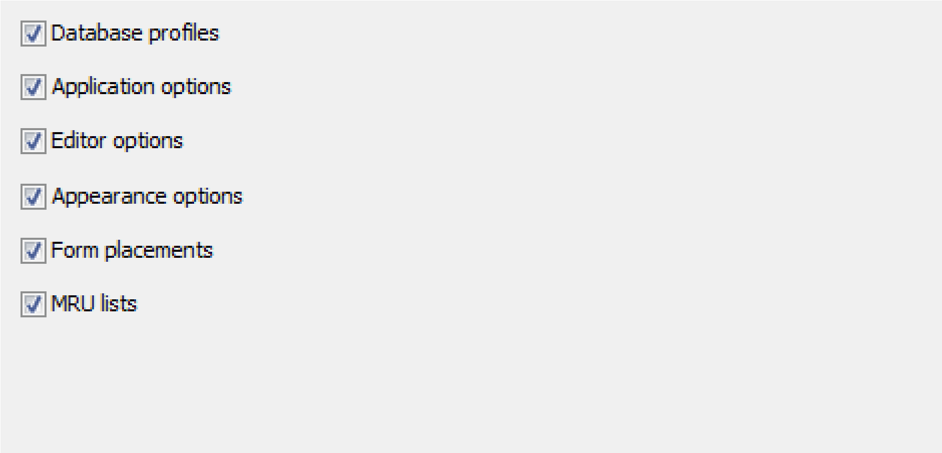
Hot zone style: Windows XP task bar

☑ Hot zone drags a splitter

11.4 Export Settings

Export Settings Wizard allows you to export all or partial MS SQL Maestro settings to single **.reg* file which can be applied to the application of MS SQL Maestro installed on another machine or used to backup previous settings. To run the wizard, select the Tools | Options main menu item and click Export Settings in the Options 424 dialog.

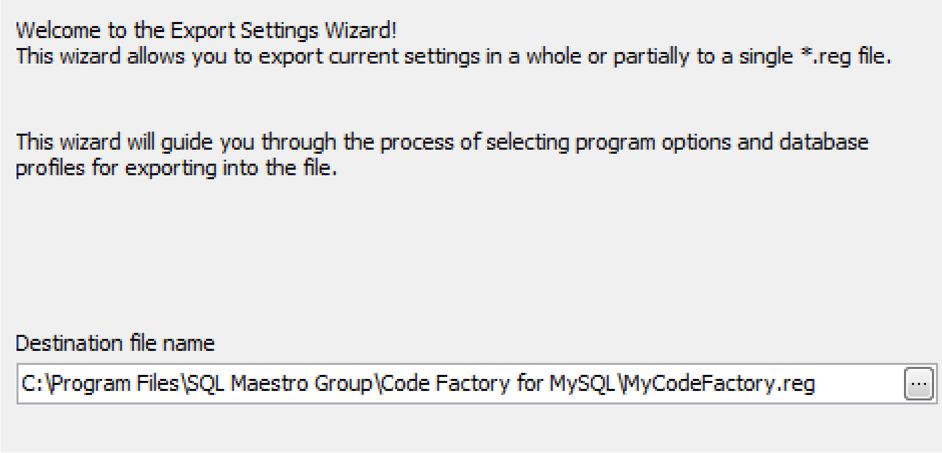
- Specifying destination file to save settings to 464
- Specifying settings categories to save 464
- Select database profiles to save 465
- Saving settings 465



☑ Database profiles
☑ Application options
☑ Editor options
☑ Appearance options
☑ Form placements
☑ MRU lists

11.4.1 Specifying destination file

Specify a **.reg* file to extract MS SQL Maestro setting to.



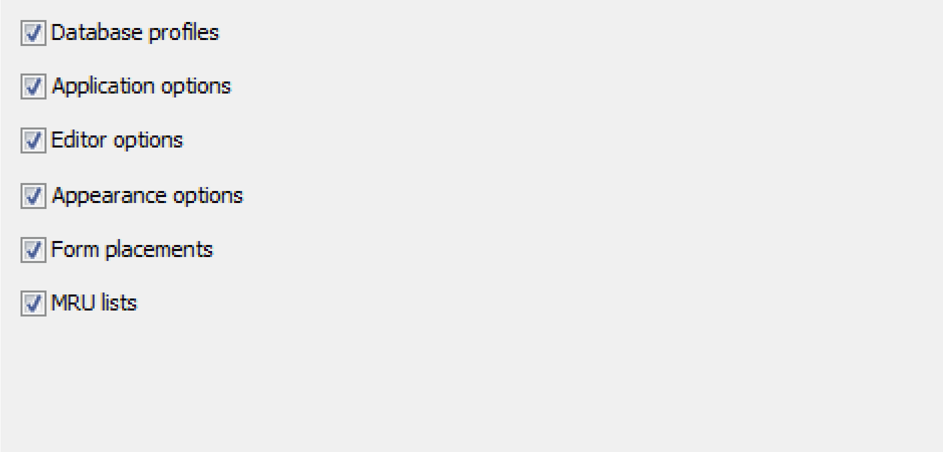
Welcome to the Export Settings Wizard!
This wizard allows you to export current settings in a whole or partially to a single *.reg file.

This wizard will guide you through the process of selecting program options and database profiles for exporting into the file.

Destination file name
C:\Program Files\SQL Maestro Group\Code Factory for MySQL\MyCodeFactory.reg

11.4.2 Selecting setting categories

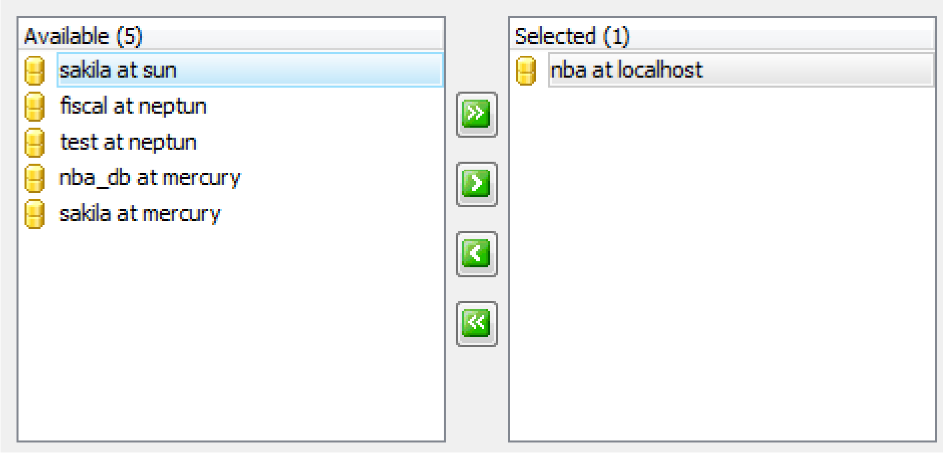
The options of this step specify the information saved to the result file, e.g. Database profiles, Application options 425, etc.



- ☑ Database profiles
- ☑ Application options
- ☑ Editor options
- ☑ Appearance options
- ☑ Form placements
- ☑ MRU lists

11.4.3 Selecting database profiles

Select database profiles to save their settings by moving them from the Available Databases list to the Selected Databases one.



Available (5)

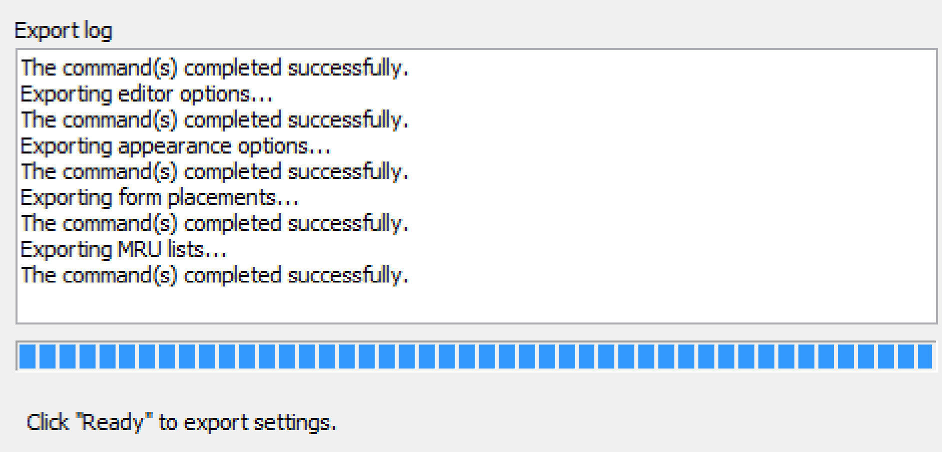
- sakila at sun
- fiscal at neptun
- test at neptun
- nba_db at mercury
- sakila at mercury

Selected (1)

- nba at localhost

11.4.4 Saving settings

Click the Ready button to start the extracting. The process log is displayed in the Export log box.



```
The command(s) completed successfully.
Exporting editor options...
The command(s) completed successfully.
Exporting appearance options...
The command(s) completed successfully.
Exporting form placements...
The command(s) completed successfully.
Exporting MRU lists...
The command(s) completed successfully.
```

Click "Ready" to export settings.

Index