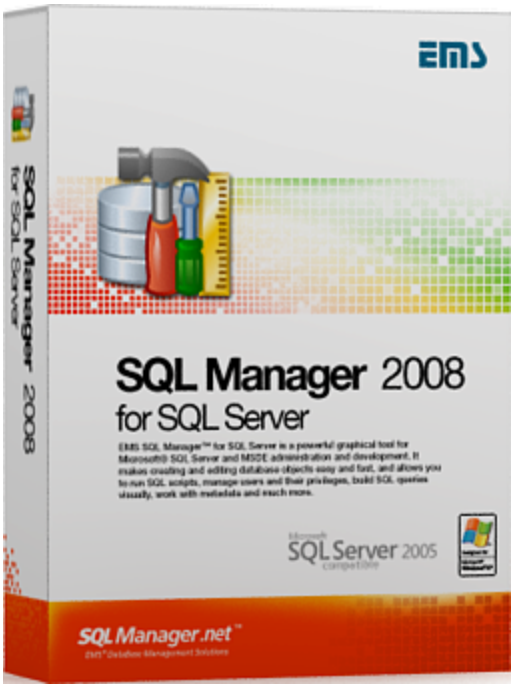
SQL Manager.net™

EMS® Database Management Solutions



SQL Manager 2008 for SQL Server User's Manual

© 2008 EMS Database Management Solutions

SQL Manager 2008 for SQL Server
User's Manual

© 2008 EMS Database Management Solutions
All rights reserved.

This manual documents EMS SQL Manager for SQL Server, version 3.3.x.x

No parts of this work may be reproduced in any form or by any means - graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems - without the written permission of the publisher.

Products that are referred to in this document may be either trademarks and/or registered trademarks of the respective owners. The publisher and the author make no claim to these trademarks.

While every precaution has been taken in the preparation of this document, the publisher and the author assume no responsibility for errors or omissions, or for damages resulting from the use of information contained in this document or from the use of programs and source code that may accompany it. In no event shall the publisher and the author be liable for any loss of profit or any other commercial damage caused or alleged to have been caused directly or indirectly by this document.

Use of this documentation is subject to the following terms: you may create a printed copy of this documentation solely for your own personal use. Conversion to other formats is allowed as long as the actual content is not altered or edited in any way.

Document generated on: 25.09.2008

Table of Contents

Part I Welcome to SQL Manager for SQL Server! 20

What's new 21
System requirements 22
Feature Matrix 23
Installation 24
Registration 25
How to register SQL Manager 27
License Agreement 29
Version history 33
EMS SQL Manager FAQ 52
Other EMS Products 58

Part II Getting started 66

Selecting style and language 68
First time started 70
Using Desktop Panel 72
Database navigation 74
Working with database objects 76
Using context menus 78
- Host context menu 78
- Database context menu 79
- Object context menu 82

Switching windows 85

Part III Database Explorer 88

Managing database registration info 91
Connecting to databases 94
Operations with database objects 97
Selecting multiple objects 100
Using tabs for database navigation 102
Recently opened objects 104
Creating projects 106
Searching within the tree 109
SQL Assistant 112
Configuring Database Explorer 114
Managing Favorite queries 116

© 2008 EMS Database Management Solutions

Part IV Database management 120

Register Host wizard 123
- **Setting connection parameters** 123
- **Selecting databases** 125
- **Setting specific options** 125

Create Database wizard 128
- **Setting database name** 128
- **Setting connection properties** 129
- **Defining database files** 130
- **Defining transaction log files** 132
- **Setting recovery model and collation** 133
- **Defining filestream files and filegroups** 135
- **Viewing result SQL statement** 136

Register Database wizard 138
- **Setting connection parameters** 138
- **Setting specific options** 140

Database Registration Info 143
- **Editing connection properties** 144
- **Setting common database options** 145
- **Setting display options** 147
- **Setting default directories** 148
- **Setting log options** 149
- **Find Option** 150

Database Properties 152
- **Information** 154
- **Options** 154
  - Recovery options 156
  - ANSI options 158
  - SQL options 160
  - Automation options 162
  - Cursor options 163
  - Advanced 165
- **Data files** 166
- **Log files** 168
- **Filegroups** 169
- **Filestream files and filegroups** 171
- **Find Option** 172

Server Properties 174
- **General options** 175
- **Memory options** 176
- **Processor options** 178
- **Security options** 180
- **Active Directory** 182
- **Connection settings** 184
- **Database Settings** 186
- **Misc Server Settings** 188
- **Find Option** 190

Part V Database objects management 194

New Object 196

© 2008 EMS Database Management Solutions

Duplicate Object Wizard 198
Selecting the source database 198
Selecting object to duplicate 199
Selecting destination database or server 200
Modifying the new object's definition 201
Schemas 203
Schema Editor 204
Using Navigation bar 204
Editing schema definition 205
Schema Objects 206
Tables 206
New table 207
Using Navigation bar 207
Setting table properties 208
Specifying fields 210
Table Editor 211
Using Navigation bar 212
Managing fields 214
Changing fields order 215
Managing foreign keys 215
Managing checks 216
Managing indexes 217
Managing triggers 218
Working with table data 219
Table properties 220
Fields 222
Field Editor 223
Setting field name and type 223
Setting field identity 225
Setting XML properties 226
Setting field defaults 227
Setting field description 228
Foreign Keys 229
Foreign Key Editor 230
Using Navigation bar 230
Editing foreign key definition 231
Checks 232
Check Editor 233
Using Navigation bar 233
Editing check definition 234
Indexes 235
Index Editor 236
Using Navigation bar 236
Editing index definition 237
Defining index storage 240
Viewing index info 241
Triggers 242
Trigger Editor 243
Using Navigation bar 243
Editing trigger definition 244
Views 246
View Editor 247
Using Navigation bar 247

© 2008 EMS Database Management Solutions

Editing view definition 248
Managing fields 249
Managing indexes 250
Managing triggers 251
Working with data 252
Procedures 253
Procedure Editor 255
Using Navigation bar 255
Editing procedure definition 256
Browsing procedure parameters 258
Specifying input parameters 259
Executing procedure/function 260
UDTs 260
UDT Editor 261
Using Navigation bar 262
Editing UDT definition 262
UDFs 265
UDF Editor 267
Using Navigation bar 267
Editing UDF definition 268
Browsing UDF parameters 269
Synonyms 270
Synonym Editor 271
Using Navigation bar 271
Editing synonym definition 272
Aggregates 273
Aggregate Editor 274
Using Navigation bar 275
Editing aggregate definition 275
XML Schema Collections 277
XML Schema Collection Editor 278
Using Navigation bar 279
Editing XML Schema Collection 279
Defaults 280
Default Editor 281
Using Navigation bar 281
Editing default definition 282
Rules 283
Rule Editor 284
Using Navigation bar 284
Editing rule definition 285
Queues 286
Queue Editor 287
Using Navigation bar 287
Editing queue definition 288
Non-schema Objects 290
Assemblies 290
Assembly Editor 291
Using Navigation bar 291
Editing assembly definition 292
DDL Triggers 294
DDL Trigger Editor 295
Using Navigation bar 295
Editing DDL trigger definition 296

© 2008 EMS Database Management Solutions

Partition Functions **298**
Partition Function Editor 299
Using Navigation bar 299
Editing partition function definition 300
Partition Schemes **302**
Partition Scheme Editor 303
Using Navigation bar 303
Editing partition scheme definition 304
Full-Text Catalogs **305**
Full-Text Catalog Editor 306
Using Navigation bar 307
Editing full-text catalog definition 307
Managing full-text indexes 308
Symmetric Keys **310**
Symmetric Key Editor 311
Using Navigation bar 311
Editing symmetric key definition 312
Asymmetric Keys **314**
Asymmetric Key Editor 315
Using Navigation bar 315
Editing asymmetric key definition 316
Certificates **317**
Certificate Editor 318
Using Navigation bar 318
Editing certificate definition 319
Message Types **321**
Message Type Editor 322
Using Navigation bar 322
Editing Message Type definition 323
Contracts **324**
Contract Editor 325
Using Navigation bar 325
Editing Contract definition 325
Services **327**
Service Editor 328
Using Navigation bar 328
Editing Service definition 329
Routes **329**
Route Editor 330
Using Navigation bar 331
Editing Route definition 331
Remote Service Bindings **332**
Remote Service Binding Editor 333
Using Navigation bar 334
Editing Remote Service Binding definition 334
Database-level principals **336**
Users **336**
User Editor 337
Using Navigation bar 337
Editing user definition 338
Defining user membership 339
Roles **339**
Role Editor 340
Using Navigation bar 341

© 2008 EMS Database Management Solutions

Editing role definition ... 341
Defining role membership ... 342

Part VI Query management tools 346

SQL Editor ... 348
Using Navigation bar ... 349
Working with SQL Editor area ... 351
Using the context menu ... 352
Viewing query plan ... 354
Using object links ... 355
Executing queries ... 356
Viewing query logs ... 357
Favorites editor ... 358
Merging queries ... 360
Visual Query Builder ... 362
Using Navigation bar ... 363
Working with diagram area ... 365
Joining two objects ... 367
Setting criteria ... 369
Setting output fields ... 371
Setting grouping criteria ... 373
Setting sorting parameters ... 374
Working with the editor area ... 376
Query execution ... 377
Viewing query plan ... 379
Query parameters ... 382
Input parameters dialog ... 382

Part VII Data management 386

Data View ... 387
Grid View ... 387
Grouping data ... 389
Filtering records ... 390
Using the context menu ... 392
Working in multi-level mode ... 393
Create Grid Level wizard ... 396
Specifying master level ... 397
Defining source for detail level ... 397
Binding master and detail levels ... 399
Query parameterization ... 400
Setting additional parameters ... 401
Working in card view mode ... 402
Column Summary ... 403
Form View ... 404
Print Data ... 405
BLOB View ... 406
Navigation within BLOB Editor ... 407
Editing as Hexadecimal ... 408
Editing as Text ... 409
Editing as Rich Text ... 410
Editing as Image ... 410
Editing as HTML ... 411

© 2008 EMS Database Management Solutions

Applying changes 412
Custom Filter 414
Filter Builder dialog 416
Invoking the Filter Builder dialog 416
Adding a new condition 418
Setting filter criteria 418
Setting filter operator 419
Setting filter criteria values 420
Adding a new group 421
Setting group operator 422
Applying filter conditions 423
Part VIII Import/Export Tools 426
Export Data Wizard 427
Setting destination file name and format 428
Selecting fields for export 429
Adjusting data formats 430
Setting header and footer 431
Setting format-specific options 432
Excel options 433
Data format 433
Fields 434
Options 434
Styles 435
Extensions 436
Hyperlinks 437
Notes 437
Charts 438
Cells 440
Merged Cells 441
Advanced 442
Access options 443
Word / RTF options 444
Base Styles 445
Strip Styles 446
HTML options 447
Preview 448
Basic 449
Multi-file 450
Advanced 451
PDF options 452
TXT options 453
CSV options 454
XML options 455
MS Excel 2007 options 456
MS Word 2007 options 458
ODF options 461
Setting common export options 464
Exporting data 466
Import Data Wizard 467
Selecting source file name and format 468
Selecting data source 469

© 2008 EMS Database Management Solutions

Setting fields correspondence **470**
Excel 471
Access 472
DBF 473
XML Datapacket 474
TXT 475
CSV 476
HTML 477
XML Generic 478
MS Excel/Word 2007, ODF 479
Adjusting data formats **480**
Setting advanced field formats **481**
Setting import mode **482**
Customizing bulk insert options **485**
Customizing common options **486**
Importing data **487**
Export as SQL Script **489**
Defining the type of destination server **490**
Setting destination file name **491**
Setting BLOB options **492**
Selecting fields to export **493**
Editing table definition **494**
Setting export options **495**
Exporting as SQL Script **496**
Bulk Insert Wizard **498**
Selecting a table or view and the data file **499**
Setting bulk insert options **500**
Customizing data file format **502**
Viewing the result SQL statement **504**
Running bulk insert **504**
Part IX Database tools **508**
Common DB Tools **509**
Dependency Tree **509**
Using Navigation bar 511
Viewing dependency tree 511
Visual Database Designer **513**
Using Navigation bar and context menu 516
Diagram Navigator 518
Adding objects to diagram 519
Removing objects from diagram 521
Creating new table 521
Creating relationships 522
Working with multiple pages 523
Reverse engineering 523
Printing diagram 525
Print Preview dialog 525
Print Setup dialog 526
Saving/Loading diagram 528
Setting diagram options 529
SQL Script Editor **530**
Using Navigation bar 532
Using the context menu 534

© 2008 EMS Database Management Solutions

Using Script Explorer 535
Script execution 537
Extract Database Wizard 538
Selecting source database 540
Specifying destination file name 540
Setting extraction mode 541
Setting BLOB options 542
Selecting objects for structure extraction 543
Selecting server objects for extraction 544
Selecting objects for data extraction 545
Customizing script options 546
Start of extraction process 548
Print Metadata 549
Using Navigation bar 550
Printing options 552
HTML Report Wizard 553
Selecting database and directory 555
Selecting object types 555
Specifying CSS 556
Setting additional report options 557
Creating HTML report 558
Reports management 559
Create Report Wizard 560
Specifying database name and report options 561
Selecting report bands 562
Selecting report style 563
Specifying paper settings 564
Specifying margins 564
Specifying other page settings 565
Report Designer 566
Adding dialog form 568
Adding database and query components 569
Adding report data 571
Viewing the report 573
Saving the report 575
Report Viewer 575
Using Navigation bar 576
SQL Monitor 577
Using Navigation bar 579
Working with SQL Monitor 579
Specific DB Tools 581
Attach Database 581
Setting connection properties 583
Defining database files and general options 584
Attaching database 585
Detach Database 585
Setting connection properties 587
Selecting the database 588
Detaching database 589
Backup Database 590
Selecting database to backup 591
Setting backup type and options 592
Selecting database files to backup 594
Selecting backup devices 595

© 2008 EMS Database Management Solutions

Setting schedule options ... 596
Setting media options ... 597
Setting advanced backup options ... 600
Setting transaction log backup options ... 601
Setting data transfer options ... 603
Running database backup ... 603
Restore Database ... 604
Selecting destination database ... 606
Setting restore type and recovery options ... 607
Selecting backup set to restore ... 609
Selecting files to restore ... 611
Setting advanced restore options ... 612
Running database restore ... 614
Shrink Database ... 615
Selecting database and setting shrink mode ... 616
Defining shrink action for database ... 617
Defining shrink action for files ... 619
Shrinking database ... 620
Indices Management ... 620
Selecting database for indices management ... 622
Selecting tables ... 623
Specifying actions and setting options ... 623
Editing and executing the result script ... 625
Update Statistics ... 626
Selecting database for statistics update ... 628
Selecting tables and views ... 628
Specifying update statistics options ... 629
Selecting indexes ... 631
Editing and executing the result script ... 631
Check Database ... 632
Selecting database to check ... 634
Specifying arguments and options ... 634
Editing and executing the result script ... 636
Grant Manager ... 637
Using Navigation bar and context menu ... 638
Managing database-specific privileges ... 641
Managing column permissions ... 642
Viewing effective permissions ... 643

Part X Server tools 648

Server-level principals ... 649
Logins ... 650
Login Manager ... 650
Login Editor ... 652
Setting login parameters ... 652
Allocating server roles ... 656
Defining server permissions ... 657
Defining endpoint permissions ... 659
Defining database permissions ... 660
User mapping ... 661
Server Roles ... 662
SQL Server objects and services ... 665
Backup Devices ... 666

© 2008 EMS Database Management Solutions

Backup devices manager ... 668
Backup Device properties ... 669
New backup device ... 670
Database Snapshots ... 671
Managing database snapshots ... 673
Database snapshot properties ... 674
Linked Servers ... 674
Linked servers manager ... 677
Linked Server Editor ... 678
Configuring Linked server ... 678
Security settings ... 679
Setting server options ... 680
Linked Server content ... 682
DTS Packages ... 682
DTS Packages manager ... 684
DTS Package properties ... 685
Jobs ... 685
Jobs manager ... 688
Job Editor ... 689
Editing job definition ... 689
Managing job steps ... 690
Managing job schedules ... 694
Managing notifications ... 695
Managing target servers ... 696
Job History ... 697
Viewing job history ... 699
Filtering job history list ... 700
Alerts ... 701
Alerts manager ... 704
Alert Editor ... 705
Using Navigation bar ... 705
Editing alert definition ... 706
Setting additional properties ... 709
Managing notifications ... 709
Operators ... 710
Operators manager ... 713
Operator Editor ... 714
Using Navigation bar ... 714
Editing operator definition ... 715
Setting additional properties ... 715
Managing alert notifications ... 716
Managing job notifications ... 717
Schedules ... 717
Shared schedules manager ... 720
Schedule Editor ... 721
Using Navigation bar ... 721
Editing schedule definition ... 722
Viewing scheduled jobs ... 724
Proxies ... 724
Proxies manager ... 727
Proxy Editor ... 728
Using Navigation bar ... 728
Editing proxy definition ... 729
Viewing proxy jobs ... 732

© 2008 EMS Database Management Solutions

Credentials **732**
Credential Editor 734
Using Navigation bar 734
Editing credential definition 735
Endpoints **736**
Endpoint Editor 737
Using Navigation bar 738
Editing endpoint definition 738
Setting protocol options 740
Setting payload options 742
DDL Triggers **744**
DDL Trigger Editor 745
Using Navigation bar 746
Editing DDL trigger definition 747
Target Servers **748**
Target Servers manager 749
Server Log Viewer **750**
Using Navigation bar 752
Working with Server Log Viewer 752
Activity Monitor **753**
Using Navigation bar 755
Working with Activity Monitor 755

Part XI Personalization **760**

SQL Manager Options **761**
Environment Options **761**
Preferences 763
Full mode activation 764
Confirmations 765
Windows 767
Tools 768
Timeouts 770
DB Explorer 770
Naming Rules 772
Object Editors 773
SQL Editor 774
SQL Monitor 776
SQL Script 777
Query Builder 778
Style & Color Palette 779
Visual Database Designer 780
Color Palette 782
Print Metadata 782
Data Export 783
SQL Server Reference 784
Fonts 785
Grid 787
Data Options 789
Print Data 791
Color & Formats 792
Advanced 794
Column Options 796
Localization 797

© 2008 EMS Database Management Solutions

Find Option ... 799
Editor Options ... 800
General ... 801
Display ... 804
Color ... 807
Quick Code ... 808
SQL Formatter ... 810
Key Mapping ... 812
Spell Checking ... 814
Find Option ... 815
Visual Options ... 816
Bars and menus ... 818
Trees and lists ... 820
Edit controls ... 821
Check boxes ... 823
Buttons ... 825
Page controls ... 826
Group boxes ... 828
Splitters ... 830
Navigation bar ... 831
Find Option ... 833
Save Settings ... 834
Specifying destination file ... 835
Selecting settings ... 836
Selecting databases ... 836
Saving settings ... 837
Localization ... 839
Localization Editor ... 840
Select Program Language ... 841
Keyboard Templates ... 843
Find Option dialog ... 846
Part XII External tools 850
External Tools dialog ... 852
External Tool Info editor ... 854
Part XIII How to... 856
Work with databases ... 857
Create Database ... 857
Drop Database ... 859
Register Database ... 862
Unregister Database ... 865
Connect to Database ... 867
Disconnect from Database ... 870
Attach Database ... 873
Detach Database ... 874
Backup Database ... 876
Restore Database ... 877
Shrink Database ... 879
Manage indices ... 880
Update Statistics ... 882

© 2008 EMS Database Management Solutions

Check Database **883**
Edit database registration info **885**
Edit database properties **886**
Edit server properties **888**
Work with database objects **890**
Work with Schemas **890**
Create Schema 890
Edit Schema 891
Drop Schema 892
Work with Tables **893**
Create Table 893
Edit Table 894
Drop Table 895
Manage table fields 896
Add Table Field 896
Edit Table Field 897
Drop Table Field 898
Manage foreign keys 899
Add Foreign Key 899
Edit Foreign Key 899
Drop Foreign Key 900
Manage table checks 901
Create Check 901
Edit Check 902
Drop Check 902
Manage table indexes 903
Create Index 903
Edit Index 904
Drop Index 904
Manage table triggers 905
Create Trigger 905
Edit Trigger 906
Drop Trigger 906
Work with Views **907**
Create View 907
Edit View 908
Drop View 910
Work with Procedures **911**
Create Procedure 911
Edit Procedure 912
Execute Procedure 914
Drop Procedure 915
Work with User-Defined Types **917**
Create UDT 917
Edit UDT 918
Drop UDT 919
Work with User-Defined Functions **920**
Create UDF 920
Edit UDF 921
Drop UDF 923
Work with Synonyms **924**
Create Synonym 924
Edit Synonym 925
Drop Synonym 926

© 2008 EMS Database Management Solutions

Work with Aggregates **927**
Create Aggregate 927
Edit Aggregate 928
Drop Aggregate 930
Work with XML Schema Collections **931**
Create XML Schema Collection 931
Edit XML Schema Collection 932
Drop XML Schema Collection 933
Work with Defaults **935**
Create Default 935
Edit Default 936
Drop Default 937
Work with Rules **938**
Create Rule 938
Edit Rule 939
Drop Rule 940
Work with Assemblies **941**
Create Assembly 941
Edit Assembly 942
Drop Assembly 943
Work with DDL Triggers **944**
Create DDL Trigger 944
Edit DDL Trigger 945
Drop DDL Trigger 946
Work with Partition Functions **948**
Create partition function 948
Edit partition function 949
Drop partition function 950
Work with Partition Schemes **951**
Create partition scheme 951
Edit partition scheme 952
Drop partition scheme 953
Work with Full-Text Catalogs **955**
Create full-text catalog 955
Edit full-text catalog 956
Drop full-text catalog 957
Work with Symmetric Keys **958**
Create symmetric key 958
Edit symmetric key 959
Drop symmetric key 960
Work with Asymmetric Keys **961**
Create asymmetric key 961
Edit asymmetric key 962
Drop asymmetric key 964
Work with Certificates **965**
Create certificate 965
Edit certificate 966
Drop certificate 967
Work with Users **968**
Create User 968
Edit User 969
Drop User 970
Work with Roles **971**
Create Role 971

© 2008 EMS Database Management Solutions

Edit Role 972
Drop Role 973
Manage Server security 975
Add server login 975
Edit login properties 976
Drop login 977
Allocate server role 979

Part XIV Appendix 982

Viewing object DDL structure 982
Editing object description 984
Browsing object dependencies 985
Setting object permissions 986
Changing Metadata window 987
Tip of the Day 989
Find Text dialog 990
Replace Text dialog 992
Format specifiers 995
Language Info Editor 998
Using templates 999
Supported file formats 1000
To-Do List 1002
Window List 1004
Customize toolbars and menus 1005
SQL Manager shortcuts 1009

© 2008 EMS Database Management Solutions

Part I

1 Welcome to SQL Manager for SQL Server!

EMS SQL Manager 2008 for SQL Server is a powerful tool for SQL Server administration and development. SQL Manager 2008 for SQL Server works with all SQL Server versions from 7 to 2005 and supports all the latest features including new SQL Server 2005 permission system, CLR objects, DDL triggers, XML columns, endpoints and so on. It offers plenty of powerful tools for experienced users to satisfy all their needs. SQL Manager has a new state-of-the-art graphical user interface with well-described wizard system, so clear in use that even a newbie will not be confused with it.

Visit our web-site for details: http://www.sqlmanager.net/

Key features:

- Full support of SQL Server 2005
- Support of Unicode data
- Rapid database management and navigation
- Easy management of all SQL Server objects
- Advanced data manipulation tools
- Effective security management
- Excellent visual and text tools for query building
- Impressive data export and import capabilities
- Visual Database Designer to handle database structure in a few clicks
- Easy-to-use wizards performing SQL Server administrative tasks
- Powerful tools to make your work with SQL Server as easy as it can be
- Report designer with clear in use report construction wizard
- New state-of-the-art graphical user interface

Product information:

Homepage:	http://www.sqlmanager.net/en/products/mssql/manager
Support Ticket System:	http://www.sqlmanager.net/support
Register on-line at:	http://www.sqlmanager.net/en/products/mssql/manager/buy

© 2008 EMS Database Management Solutions

1.1 What's new

Version	Release date
SQL Manager 2008 for SQL Server 3.3.0.1	September 25, 2008

What's new in SQL Manager 2008 for SQL Server?

- The following SQL Server 2008 features are now supported:
  - Compressed Tables and Indexes.
  - Field and file group filestream.
  - Filtered indexes.
- Query Builder.
  - When a table is added into a query, its foreign key connections to other tables participating
  - In some cases renaming a table and moving to the SQL tab resulted in the "Cannot bind JOIN
- DB Explorer
  - The possibility to reorder hosts using drag-and-drop is implemented.
  - The User role and Application role objects now have different icons.
  - Pressing the Edit Object button or selecting the corresponding context menu item with multi
  - The warning that the database compatibility level is not maximized can be disabled now.
  - SQL Assistant. The Edit objects command didn't work with server objects. Fixed now.
- Database Registration info. The "Schema-organized database tree" option is added. Disabling this option makes SQL Manager create the object tree without schema division.
- Database Properties. Database owner information is now available.
- The ability to edit the result of a query containing JOIN statements is implemented.
- The seed and increment parameters of the IDENTITY field can now be negative.
- Restore Database wizard. It is now possible to disconnect all users from the existing database wh
- HTML Report. The possibility to change font size for the DDL section is added.
- Changing metadata window can now be closed by pressing the ESC button.
- The Access Violation error appeared sometimes when deleting User-defined data type. Fixed now.
- Queue Editor. If the Rollback operation was made in the Changing metadata window while editing
- Database object comments containing national symbols were displayed improperly if their collation
- Extract Database. Bugs concerning wrong statements order and incomplete object definition, and
- Code Completion window. Some system functions used to be duplicated in the list. Fixed now.
- Code Parameters. The list of parameters didn't appear if the object name was put into square bra
- Visual Database Designer. Some printers couldn't print field and table icons. Fixed now.
- Table Editor. The Property tab couldn't be refreshed. Fixed now.
- An attempt to connect to SQL Server 2005 database with compatibility level 8 led to an error; co
- If different languages that use different decimal separators were set at the first and the last tabs
- Data Grid. An error occurred when editing data of the date format with month as text (e.g. dd-M
- Wrong shortcuts were displayed for a number of commands from the Quick Code context menu. Fi
- Other small improvement and bugfixes.

See also:

Version history

© 2008 EMS Database Management Solutions

1.2 System requirements

System requirements for SQL Manager 2008 for SQL Server

- 300-megahertz (MHz) Intel® Pentium II-compatible or faster processor; 600-megahertz (MHz) or faster processor recommended
- Microsoft® Windows NT4 with SP4 or later, Microsoft® Windows 2000, Microsoft® Windows 2000 Server, Microsoft® Windows XP, Microsoft® Windows 2003 Server, Microsoft® Windows Vista
- 64MB RAM or more; 128MB or more recommended
- 50MB of available HD space for program installation
- Super VGA (800x600) or higher-resolution video adapter and monitor; Super VGA (1024x768) or higher-resolution video adapter and monitor recommended
- Microsoft® Mouse or compatible pointing device
- Microsoft® Data Access Components (MDAC) or SQL Server Native Client
- Possibility to connect to any local or remote Microsoft® SQL Server™

© 2008 EMS Database Management Solutions

1.3 Feature Matrix

The **FREE** *Lite version* of **SQL Manager 2008 for SQL Server** does not include all features of the *Full version* and has some limitations concerning the number of the databases that can be registered and the number of data manipulation and server maintenance tools. The detailed feature matrix is given below.

Note that when using the **FREE** *Lite version* of **SQL Manager 2008 for SQL Server** you can activate a 30-day period of fully-functional usage. After the period expires, you will be able to continue using the Lite version.

For more information on activating the Full version features see Full Mode activation.

© 2008 EMS Database Management Solutions

1.4 Installation

If you are **installing SQL Manager 2008 for SQL Server for the first time** on your PC:

- download the SQL Manager 2008 for SQL Server distribution package from the download page available at our site;
- unzip the downloaded file to any local directory, e.g. *C:\unzipped*;
- run *MSManagerFullSetup.exe* from the local directory and follow the instructions of the installation wizard;
- after the installation process is completed, find the SQL Manager shortcut in the corresponding group of Windows Start menu.

If you want to **upgrade an installed copy of SQL Manager 2008 for SQL Server** to the latest version:

- download the executable file of SQL Manager 2008 for SQL Server from the download page available at our site;
- unzip the downloaded file to any local directory, e.g. *C:\unzipped*;
- close SQL Manager application if it is running;
- replace the previous version of SQL Manager by copying the unzipped executable file to the *C:\...\%program_directory%* location;
- run SQL Manager 2008 for SQL Server using its shortcut in the corresponding group of Windows Start menu.

Also you can use the full distribution package to upgrade your current version of SQL Manager 2008 for SQL Server. In this case, you should repeat the steps you have made for the first-time installation. Note that the full distribution package is larger than a single executable file.

See also:

SQL Manager FAQ

License agreement

© 2008 EMS Database Management Solutions

1.5 Registration

To make it easier for you to purchase our products, we have contracted with share-it! registration service. The share-it! order process is protected via a secure connection and makes on-line ordering by credit/debit card quick and safe. The following information about share-it! is provided for your convenience.

Share-it! is a global e-commerce provider for software and shareware sales via the Internet. Share-it! accepts payments in US Dollars, Euros, Pounds Sterling, Japanese Yen, Australian Dollars, Canadian Dollars or Swiss Franks by Credit Card (Visa, MasterCard/EuroCard, American Express, Diners Club), Bank/Wire Transfer, Check or Cash.

If you have ordered EMS software online and would like to review your order information, or if you have questions about ordering, payments, or shipping procedures, please visit our Customer Care Center, provided by share-it!

Please note that all of our products are delivered via ESD (Electronic Software Delivery) only. After purchase you will be able to immediately download the registration keys or passwords and download links for archives of full versions. Also you will receive a copy of registration keys or passwords by e-mail. Please make sure to enter a valid e-mail address in your order. If you have not received the keys within 2 hours, please, contact us at sales@sqlmanager.net.

Product distribution	
SQL Manager 2008 for SQL Server (Business license) + 1-Year Maintenance*	Register Now!
SQL Manager 2008 for SQL Server (Business license) + 2-Year Maintenance*	
SQL Manager 2008 for SQL Server (Business license) + 3-Year Maintenance*	
SQL Manager 2008 for SQL Server (Non-commercial license) + 1-Year Maintenance*	
SQL Manager 2008 for SQL Server (Non-commercial license) + 2-Year Maintenance*	
SQL Manager 2008 for SQL Server (Non-commercial license) + 3-Year Maintenance*	
SQL Manager 2008 for SQL Server (Trial version)	Download Now!
SQL Manager 2008 for SQL Server Lite	Download Now!

*__EMS Maintenance Program__ provides the following benefits:

- Free software bug fixes, enhancements, updates and upgrades during the maintenance period
- Free unlimited communications with technical staff for the purpose of reporting Software failures
- Free reasonable number of communications for the purpose of consultation on operational aspects of the software

© 2008 EMS Database Management Solutions

After your maintenance expires you will not be able to update your software or get technical support. To protect your investments and have your software up-to-date, you need to renew your maintenance.

You can easily reinitiate/renew your maintenance with our on-line, speed-through Maintenance Reinstatement/Renewal Interface. After reinitiating/renewal you will receive a confirmation e-mail with all the necessary information.

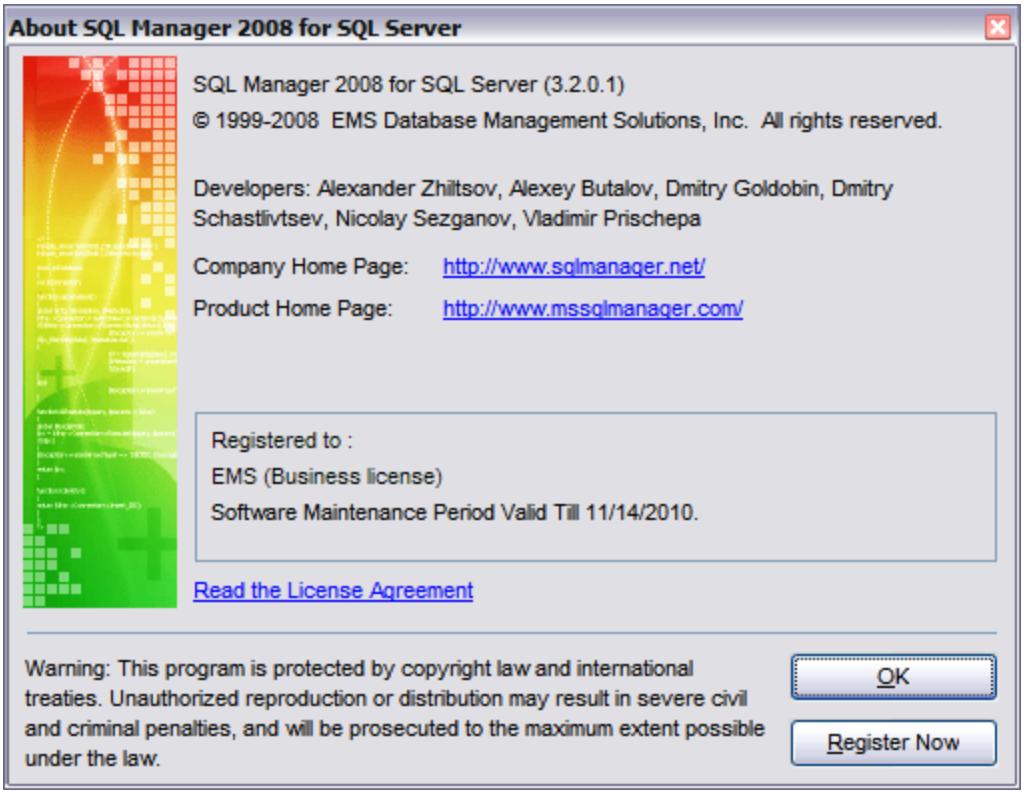
See also:

How to register SQL Manager

© 2008 EMS Database Management Solutions

1.6 How to register SQL Manager

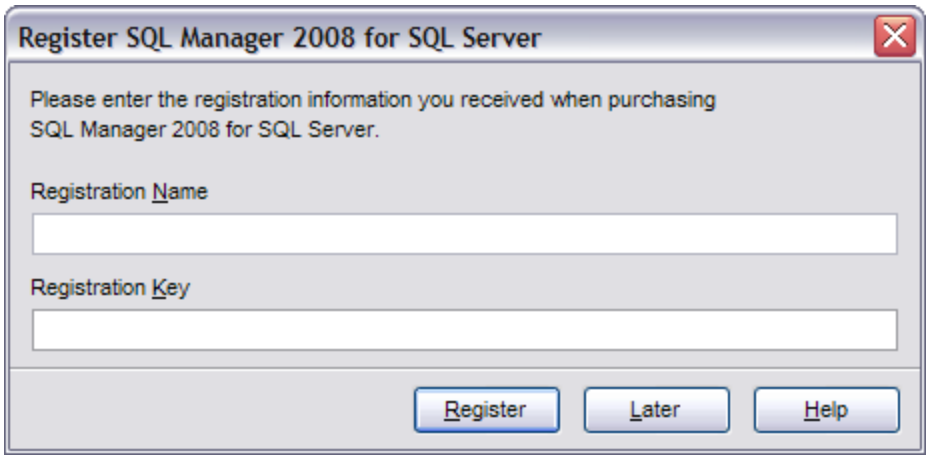
If you have not registered your copy of SQL Manager 2008 for SQL Server yet, you can do it by pressing the **Register Now** button and entering your registration information in the **Register SQL Manager 2008 for SQL Server** dialog.



To register your newly purchased copy of EMS SQL Manager 2008 for SQL Server, perform the following steps:

- receive the notification letter from **Share-it!** with the registration info;
- enter the **Registration Name** and the **Registration Key** from this letter;
- make sure that the registration process has been completed successfully – check the registration information in the **About SQL Manager 2008 for SQL Server** dialog (use the **Help | About** menu item to open this dialog).

© 2008 EMS Database Management Solutions



See also:

Registration

License Agreement

© 2008 EMS Database Management Solutions

1.7 License Agreement

- License Agreement for SHAREWARE versions
- License Agreement for FREEWARE versions

License Agreement for SHAREWARE versions

© 1999-2008 EMS Database Management Solutions, Inc.
All rights reserved.

IMPORTANT: PLEASE READ THIS AGREEMENT CAREFULLY BEFORE USING THE SOFTWARE.

END USER LICENSE AGREEMENT
EMS Database Management Solutions, Inc. ("EMS") agrees to provide the user ("USER") with a copy of this software product ("SOFTWARE"), and grants the USER a limited license to use the SOFTWARE. This LICENSE AGREEMENT ("LICENSE") defines what the USER may do with the SOFTWARE, and contains limitations on warranties, liabilities and remedies. This LICENSE may be revoked by EMS at any time without notice if the USER fails to comply with the terms of this LICENSE. The copyright and all other rights in the SOFTWARE shall remain with EMS.

LICENSE OF UNREGISTERED SOFTWARE
An unregistered copy of the SOFTWARE ("UNREGISTERED SOFTWARE") may be used by the USER for evaluation purposes for a period of thirty (30) days following the initial installation of the UNREGISTERED SOFTWARE. At the end of the trial period ("TRIAL PERIOD"), the USER must either register the SOFTWARE or remove it from his system. The UNREGISTERED SOFTWARE may be freely copied and distributed to other users for their evaluation.

LICENSE OF REGISTERED SOFTWARE
A registered copy of the SOFTWARE ("REGISTERED SOFTWARE") allows the USER to use the SOFTWARE only on a single computer or network and only by a single user at a time. If the USER wishes to use the SOFTWARE for more than one user, the USER will need a separate license for each individual user. The USER is allowed to make one copy of the REGISTERED SOFTWARE for back-up purposes.

DISTRIBUTION OF UNREGISTERED SOFTWARE
The uninstalled, UNREGISTERED SOFTWARE may be freely copied and distributed to other users provided the USER complies with the following requirements. If the USER offers this uninstalled, UNREGISTERED SOFTWARE for download ("SHAREWARE SITE USER"), then the SHAREWARE SITE USER agrees to:
(1) Immediately replace this version of the uninstalled, UNREGISTERED SOFTWARE with a new version of this SOFTWARE if a new version is released by EMS, or
(2) Delete this version of the UNREGISTERED SOFTWARE immediately upon written email notice by EMS.

TERM OF LICENSE
This LICENSE shall continue for as long as the USER uses the REGISTERED SOFTWARE and/or distributes the UNREGISTERED SOFTWARE according to the terms of this agreement. However, this LICENSE will terminate if the USER fails to comply with any of

its terms or conditions. The USER agrees, upon termination, to destroy all copies of the REGISTERED and/or UNREGISTERED SOFTWARE. The limitations of warranties and liability set forth in this LICENSE shall continue in force even after termination.

ACCEPTANCE OF THIS LICENSE AGREEMENT
By downloading and/or installing this SOFTWARE, the USER agrees to the terms of this LICENSE.

DISCLAIMER OF WARRANTY AND LIABILITY
THE SOFTWARE AND THE ACCOMPANYING FILES ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, EMS DISCLAIMS ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, ANY IMPLIED WARRANTIES OF PERFORMANCE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NONINFRINGEMENT. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, IN NO EVENT SHALL EMS BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL OR INCIDENTAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, DAMAGES FOR LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTION OR LOSS OF BUSINESS INFORMATION) ARISING OUT OF THE USE OF OR INABILITY TO USE THE SOFTWARE.

ANY LIABILITY OF EMS WILL BE LIMITED EXCLUSIVELY TO PRODUCT REPLACEMENT OR REFUND OF ORIGINAL PURCHASE PRICE.

OTHER RESTRICTIONS
The USER may not rent, lease, sublicense, translate, disassemble, reverse engineer, or de-compile the SOFTWARE, or modify or merge the SOFTWARE with any part of the software in another program. This LICENSE may not be assigned or otherwise transferred without the prior written consent of EMS.

INVALID PROVISIONS
If any provision of this LICENSE shall be declared invalid or unenforceable, the remaining provisions of this LICENSE shall remain in full force and effect to the fullest extent permitted by law. In such event, each provision of this LICENSE which is invalid or unenforceable shall be replaced with a provision as similar in terms to such invalid or unenforceable provision as may be possible which is legal and enforceable.

ENTIRE AGREEMENT
This LICENSE is the entire agreement between EMS and the USER, and supersedes any other agreement, oral or written, and may not be changed except by a written signed agreement.

License Agreement for FREEWARE versions

© 1999-2008 EMS Database Management Solutions, Inc.
All rights reserved.

IMPORTANT: PLEASE READ THIS AGREEMENT CAREFULLY BEFORE USING THE SOFTWARE.

END USER LICENSE AGREEMENT
EMS Database Management Solutions, Inc. ("EMS") agrees to provide the user ("USER") with a copy of this software product ("SOFTWARE"), and grants the USER a limited license to use the SOFTWARE. This LICENSE AGREEMENT ("LICENSE") defines what the USER may

do with the SOFTWARE, and contains limitations on warranties, liabilities and remedies. This LICENSE may be revoked by EMS at any time without notice if the USER fails to comply with the terms of this LICENSE. The copyright and all other rights in the SOFTWARE shall remain with EMS.

LICENSE OF SOFTWARE
This FREEWARE edition of SOFTWARE is free and may be used by the USER for any legal purposes for an unlimited period on unlimited number of computers without any restrictions. The FREEWARE edition of SOFTWARE may have some functional restrictions and does NOT contain any ADWARE or SPYWARE.

DISTRIBUTION OF SOFTWARE
The FREEWARE edition of SOFTWARE may be freely copied and distributed to other users without any restrictions.

TERM OF LICENSE
This LICENSE shall continue for as long as the USER uses the SOFTWARE and/or distributes the SOFTWARE according to the terms of this agreement. However, this LICENSE will terminate if the USER fails to comply with any of its terms or conditions. The limitations of warranties and liability set forth in this LICENSE shall continue in force even after termination.

ACCEPTANCE OF THIS LICENSE AGREEMENT
By downloading and/or installing this SOFTWARE, the USER agrees to the terms of this LICENSE.

DISCLAIMER OF WARRANTY AND LIABILITY
THE SOFTWARE AND THE ACCOMPANYING FILES ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, EMS DISCLAIMS ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, ANY IMPLIED WARRANTIES OF PERFORMANCE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NONINFRINGEMENT. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, IN NO EVENT SHALL EMS BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL OR INCIDENTAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, DAMAGES FOR LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTION OR LOSS OF BUSINESS INFORMATION) ARISING OUT OF THE USE OF OR INABILITY TO USE THE SOFTWARE.

OTHER RESTRICTIONS
The USER may not rent, lease, sublicense, translate, disassemble, reverse engineer, or de-compile the SOFTWARE, or modify or merge the SOFTWARE with any part of the software in another program. This LICENSE may not be assigned or otherwise transferred without the prior written consent of EMS.

INVALID PROVISIONS
If any provision of this LICENSE shall be declared invalid or unenforceable, the remaining provisions of this LICENSE shall remain in full force and effect to the fullest extent permitted by law. In such event, each provision of this LICENSE which is invalid or unenforceable shall be replaced with a provision as similar in terms to such invalid or unenforceable provision as may be possible which is legal and enforceable.

ENTIRE AGREEMENT
This LICENSE is the entire agreement between EMS and the USER, and supersedes any other agreement, oral or written, and may not be changed except by a written signed

© 2008 EMS Database Management Solutions

agreement.

© 2008 EMS Database Management Solutions

1.8 Version history

Product name	Version	Release date
SQL Manager 2008 for SQL Server	**Version 3.2.0.1**	April 10, 2008
SQL Manager 2008 for SQL Server	**Version 3.1.0.1**	January 18, 2008
SQL Manager 2008 for SQL Server	**Version 3.0.0.1**	November 28, 2007
SQL Manager 2005 for SQL Server	**Version 2.7.0.1**	June 13, 2007
SQL Manager 2005 for SQL Server	**Version 2.6.0.1**	January 23, 2007
SQL Manager 2005 for SQL Server	**Version 2.5.0.1**	September 29, 2006
SQL Manager 2005 for SQL Server	**Version 2.4.0.1**	June 13, 2006
SQL Manager 2005 for SQL Server	**Version 2.3.0.1**	April 24, 2006
SQL Manager 2005 for SQL Server	**Version 2.2.0.1**	February 14, 2006
SQL Manager 2005 for SQL Server	**Version 2.1.0.1**	November 17, 2005
SQL Manager 2005 for SQL Server	**Version 2.0.5.1**	October 31, 2005
SQL Manager 2005 for SQL Server	**Version 2.0.0.1**	October 14, 2005
MS SQL Manager	**Version 1.8.5.1**	June 28, 2005
MS SQL Manager	**Version 1.8.0.1**	December 16, 2004
MS SQL Manager	**Version 1.7.0.1**	September 15, 2004
MS SQL Manager	**Version 1.6.0.1**	May 31, 2004
MS SQL Manager	**Version 1.5.0.1**	April 14, 2004
MS SQL Manager	**Version 1.4.0.1**	February 24, 2004
MS SQL Manager	**Version 1.3.0.1**	January 14, 2004
MS SQL Manager	**Version 1.2.0.1**	November 17, 2003
MS SQL Manager	**Version 1.1.0.1**	October 14, 2003
MS SQL Manager	**Version 1.0.0.1**	October 1, 2003

Full version history is available at
http://www.sqlmanager.net/products/mssql/manager/news

Version 3.2.0.1

- Service Broker support. Now all Service Broker objects are supported: message types, contracts, queues, services, routes, remote service bindings
- SQL Server 2008 support:
  - new data types support: date, datetime2, datetimeoffset, time, hierarchyid, geometry, geography;
  - table types support implemented. Table types are displayed at the UDTs branch within the Explorer tree;
  - Backup Database wizard now supports new backup parameters such as compression, buffer count, max transfer size;
  - logon triggers support implemented. Logon triggers are created in the Server DDL Trigger Editor. New DDL trigger events are currently supported as well.
- The 'Same queries for all database' option is added for the SQL Editor. With this option enabled SQL Editor stores all queries in a shared repository so that switching to another database does not cause loading that database queries (applying that option does not affect the currently opened copies of the SQL Editor). The values of the option can be changed freely without any risk to lose the query repository content.
- SQL editors:
  - some functions work quicker now;

© 2008 EMS Database Management Solutions

  - now you can change the size of the Code Completion list;
  - the 'Search next' feature now works even if the 'Find dialog' was opened for another SQL Editor.
- Job History tool. Jobs can now be edited, stopped, run and deleted directly within the Job History form
- Server Log Viewer. Time in the date field is now displayed with milliseconds
- Now on attempt to connect to database the compatibility level is checked with the prompt for changing it, if needed
- The 'Invalid date' error occurred when entering a date with different languages set at the first and last tabs of the Control Panel | Regional and Language Options dialog. Fixed now
- In some rare cases the 'Multiple-step operation generated errors' error used to occur when executing a query containing XML fields. Fixed now
- Activity Monitor. After refreshing the content, filtering options used to be reset. Fixed now
- Dependency tab. The dependencies used to be displayed improperly for some types and procedures. Fixed now
- Location of toolbars within the program main window used to be reset. Fixed now
- Print Metadata. The table field comments were not displayed. Fixed now
- Visual Database Designer. The default settings for the diagram were not applied to its first tab. Fixed now
- Export Data Wizard. Now you can export all results of the multi-recordset query
- BLOB Editor used to allow editing data even in read-only levels. Fixed now
- Fixed the error occurred on attempt to connect to a database using the password which contained punctuation marks
- Many other small improvements and bugfixes

Version 3.1.0.1

- Implemented support for server objects in Extract Database, Print Metadata, Duplicate Object, New Object, SQL Script Explorer tools
- Grant Manager: it is now possible to view effective permissions on objects
- SQL Editor: a number of functions now work faster
- SQL Editor: fixed several bugs concerning incorrect work of code folding, 'List index out of bounds' and 'Access Violation' errors, cases of incorrect recognition of table aliases in queries, non-applicability of background color
- SQL Editor: keyboard templates do not trigger in comments any more
- Database Registration Info: added the 'Always use client cursor to get data' option; this option should be enabled for remote server connections, e.g. upon connection via Internet
- Print Metadata: added the possibility to interrupt report generation
- Data Grid: sub-levels based on a SQL query can now be edited in the Load All Rows mode as well, if the query is built by one table and contains key fields
- Data Grid: upon insertion of text into a grid cell, Shift-Ins resulted in the duplicating field dialog. Fixed now
- View Editor: added a capability for editing queries in Query Builder
- Job Editor: added commands for running tasks, stopping task execution, viewing task execution results
- Indices Management: now it is possible to select several indexes and set parameters for all of them simultaneously
- SQL Editor: Unicode was not supported upon inputting parameters for queries containing parameters' declaration. Fixed now

© 2008 EMS Database Management Solutions

- SQL Editor: when working with several instances of SQL Editor, Favorite queries were not always saved correctly. Fixed now
- SQL Editor: in some cases, after server connection was lost, no attempt to reconnect was made. Fixed now
- Upon duplicating full-text catalogs an error message appeared stating that the query could not be executed inside a transaction. Fixed now
- SQL Script: the 'Cannot create file' error used to occur on attempt to save the editor content to a file. Fixed now
- Extract Database: the generated CREATE DATABASE statement did not contain operators for creating files and file groups. Fixed now
- Server Log Viewer: in some cases the 'Field Date is not found' error occurred while opening a log. Fixed now
- Restore Database wizard: when saving master options to a template, in some cases database file names were not written into the template. Fixed now
- Keyboard Templates: when using the Paste command in a template body, applying the template resulted in insertion of the first word only, instead of the entire text from the clipboard. Fixed now
- Dependency Tree: if the tree building was interrupted, rebuilding was not initiated upon subsequent selection of an object. Fixed now
- Other minor improvements and bug-fixes

Version 3.0.0.1

- Added several new tools for working with the server: Server Log Viewer, Activity Monitor, Shrink Database wizard, Check Database wizard, Indices management and Update statistics wizards, server roles management
- Implemented support for a number of additional SQL Server capabilities: support of Symmetric keys, Asymmetric keys, Certificates; support of Full-text catalogs and Full-text indices; support of server DDL triggers; support of Endpoints; support of Database Snapshots; support of Shared Schedules, Proxies and Target Servers in Jobs; support of new login properties, such as password policy, enable/disable; creating Certificate-mapped and Asymmetric key-mapped logins; capability of mapping logins to database users in Login Editor; assigning credentials to SQL Server Logins
- Support of Unicode in SQL Editor, Query Builder, Grid View, Import/Export Data wizards
- All wizards, including Import/Export Data wizards, Extract Database wizard, Backup/Restore Database wizards run in background mode now, allowing one to continue working with the program
- New advanced SQL Editor of the Visual Studio 2005 kind with Code Folding, UNICODE support, code completion, syntax highlight and other useful features
- Enhanced Import Data Wizard - now data can be imported up to 10 times faster; also added support of the following source file formats: HTML, XML Document, Open Document Format (ODF), Open XML Format (MS Word 2007, MS Excel 2007)
- Added the Permissions tab to object editors, making it possible to grant privileges on objects to users and roles
- Improved DB Explorer in several respects: added a menu for generation of common SQL statements for database objects (Script to SQL Editor, Script to Clipboard) - it is now possible to get CREATE, ALTER, DROP, SELECT statements for each database object; DB Explorer displays server objects (objects of the server level) as well; object status (e.g. enabled/disabled) is indicated by corresponding icons; added Tasks submenu to the context menu, items of the submenu provide access to basic operations that can be performed over the selected object/database/server; added the Search Panel
- Now it is possible to save frequently used queries within the Favorite Queries node of

© 2008 EMS Database Management Solutions

DB Explorer; Favorite Queries can be stored either locally in Windows Registry (in this case they are only available for local users), or in a special table of the database (therefore, the queries can be accessed by all database users)
- Enhanced Visual Database Designer: added a new diagram object (Comment); possibility to select the specification level for displaying objects (only object names, only fields); upon Reverse Engineering one can select schemas to add their tables to the diagram
- Improved Export Data Wizard: now works faster; added several output file formats: XML Document, Open Document Format (ODF), Open XML Format (MS Word 2007, MS Excel 2007)
- Advanced Table Editor: added a possibility to view and edit table properties within a separate tab
- Added the Find Option tool which allows one to find the required option in the scope of all program settings, by a word or an expression from the option title
- Enhanced reports management system
- It is now possible to execute functions directly in Function Editor window
- Extract Database wizard: object dependencies are now handled correctly
- Backup/Restore Database wizards: implemented support of several additional backup/restore parameters applied to SQL Server 2005 (copy-only backup, verify backup when finished, checksum support)

Scroll to top

Version 2.7.0.1

- Database properties: information about the database is now displayed: the size of the entire database, data, indexes, transaction log, etc.; the date of the recently performed full, differential, transaction log backup
- Database properties: possibility to rename the database added
- Database properties: it is now possible to open this dialog for disconnected databases and for databases in the offline mode
- Database properties: it has become possible to change the name and location of database files (the latter is applied to SQL Server 2005 only)
- DB Explorer: it is now possible to view brief table information in SQL Assistant
- It is now possible to modify the schema/owner in object editors
- Incremental search has been added to a number of object lists
- Data View: for string fields of 80 and more characters in length an editor similar to that used for editing TEXT fields is implemented in the Form View; also, the "Word wrap in string editor" option is added
- Table Editor, Fields tab: the Data Type column now contains the basic type on which the current data type is based
- SQL Editor: when several copies of SQL Editor are used and the content of one of them is being saved, the changes in the content of the other ones are also considered
- Export as SQL Script: you are now prompted to confirm the result file replacement if the file already exists
- HTML Report: an error used to occur when building a report for SQL Server 2000 users ("Invalid object name sys.schemas"). Fixed now
- Fixed the "List index out of bounds" error that occurred on Step 2 of Export Data Wizard with German localization applied
- An error used to occur upon attempt to add the IDENTITY field to an existing table. Fixed now
- When working on Windows Vista, with Native Client installed and SQL authentication used, it was impossible to connect to the server. Corrected now
- Trigger Editor: it was impossible to change the 'Enabled' property for encrypted

© 2008 EMS Database Management Solutions

triggers when using the editor
- In SQL Editor the name of the object was not displayed as a link when the default schema was not [dbo] for the current user and the name of the object of this schema did not contain the [dbo] schema name
- Restore Database: there was an error when specifying the backup file on selecting backup set to restore from (in the "Backup of database" mode). Fixed now
- Extract Database: dependencies between Views and User-Defined Functions were not considered in the generated script. Corrected now
- Grid: in the Load Visible Rows mode the Blank and the Non Blank filters did not work. Fixed now
- After aligning windows, e.g. by title vertical, another object selection in the object editor caused changing the position and the size of the form. Fixed now
- Visual Database Designer: the Primary keys that were created for several fields were not displayed. Fixed now
- Other minor improvements and bug-fixes

Scroll to top

Version 2.6.0.1

- SQL Script: ability to execute selected statements and statements under cursor added
- The identity field property can now be edited
- Reorder Fields: field order can now be changed no matter if the table has other dependent objects or not
- Table Editor: if Object Explorer is turned off, Table Editor opens quicker. Object Explorer can be disabled within the Object Editors section of the Environment Options dialog
- Table Editor, Fields tab: the Identity column added
- Extract Database: view triggers used to be extracted with empty bodies. Fixed now
- SQL Editor, Query Builder, Procedure Editor: when executing a query asynchronously, CPU load could reach 100%. Fixed now
- The DDL of assemblies is now represented as binary code
- Dependency Tree: with the 'Show subobjects' option enabled, the dependencies were displayed not quite correctly. Fixed now
- Dependency Tree: object icons were not printed. Fixed now
- SQL Script: the SQL Script window does not close upon disconnect from the database
- SQL Script: the error list was not cleared when loading a file. Fixed now
- Editor Options: the ability to set default colors added
- Index Editor: when creating an index, you need not specify the data space; the data space of the table is used by default
- SQL Editor: when switching between queries, the cursor and the scrolling bar position were not saved. Fixed now
- DDL Trigger Editor: wrong script used to be generated when disabling/enabling an existing trigger. Fixed now
- Data View: mouse-wheel scrolling for the Form View is implemented
- Visual Options: the 'Menus show recently used commands first' option added
- Environment Options | DB Explorer: the 'Refresh object on showing in SQL Assistant' option is added
- Add Grid Level: with the ‘Show tables related by foreign keys only’ mode enabled, loading tables used to take much time on SQL Server 2005. Fixed now
- It was impossible to create a TIMESTAMP field with a NOT NULL restriction. Fixed now
- To-Do List: on selecting a disconnected database in the list the ‘Invalid authorization specification’ error message used to appear. Fixed now

© 2008 EMS Database Management Solutions

- In some rare cases when executing several queries simultaneously in asynchronous mode, the 'Access Violation' error used to appear. Fixed now
- DB Explorer: a newly created object was not listed in the 'Recent Objects' drop-down menu
- View Editor, Triggers tab: if a view had more than one trigger, the type for the second and the succeeding triggers used to be displayed incorrectly. Fixed now
- Visual Database Designer: in some rare cases when opening a chart the 'Could not convert variant of type (OleStr) into (Double)' error occurred. Fixed now
- Export as SQL Script: the values of DATETIME and binary types used to be returned incorrectly for the sql_variant fields. Fixed now
- Import Data Wizard: when importing from DBF format, the default values for the Boolean type were 'False' and 'True' (instead of the required 'F' and 'T'), so it was necessary to change them into the appropriate ones while processing the data. Fixed now
- The Changing Metadata window can now be closed with the Esc button applying no changes
- Environment Options | Windows: the 'Number of open editors is restricted' option did not function. Fixed now

Scroll to top

Version 2.5.0.1

- Database Registration Info: an option that allows one to skip refreshing objects on connection to database is added
- SQL Editor, SQL Script, Procedure Editor: information messages are displayed correctly now; to enable the messages, switch on the corresponding options within the Tools and the SQL Script sections of the Environment Options dialog)
- French localization added
- Grid: items that allow switching between grid tabs are added to the context menu
- Grid: in some cases for the master-detail views related by decimal or money type field in the "Load all rows" mode the records were not displayed. Fixed now
- Grid: filtering of DATETIME type data used to work incorrectly in SQL Filter mode. Fixed now
- Grid: the "Word wrap in memo editor" option added (Environment Options | Grid | Advanced)
- Grid: formats set in Grid | Color & Formats were not applied to detail levels. Fixed now
- Linked Servers: Now this feature works fine with SQL Server version 7.0
- Export as SQL Script: 'COMPUTED BY' fields were not excluded from the INSERT statements. Fixed now
- Export as SQL Script: sql_variant type data were always represented as strings in the result script. Fixed now
- CLR User-defined data types were not shown in the DB Explorer tree. Fixed now
- When duplicating a table with data into another database, the TEXT and NTEXT type data were not copied. Fixed now
- When duplicating a procedure, its name remained. Fixed now
- DB Explorer: after adding an object into the project it did not appear within the Projects node. Fixed now
- Trigger Editor: the comments preceding the CREATE TRIGGER statement were not displayed in trigger DDL. Fixed now
- SQL Editor: for queries returning more than one dataset the value of Fetched records is changed now in the status bar on switching between the grid tabs
- On executing a procedure with parameters, those of DATETIME type containing the time part were not saved between execution sessions. Fixed now

© 2008 EMS Database Management Solutions

- Visual Database Designer: after resetting all toolbars and opening Visual Database Designer, the toolbar used to disappear from the form. Fixed now
- The 'Reset all Toolbars and menus' command did not work properly for DB Explorer toolbar. Fixed now
- Now it is possible to resize the To-Do Item Editor window
- Dependency Tree: the 'Save as image' button is now available on the Navigation bar
- When creating a table without the NOT NULL flag for the Primary key, the constraint was not applied automatically, which caused wrong script generation and resulted in error upon its execution. Fixed now

Scroll to top

Version 2.4.0.1

- Export as SQL Script: now you can specify the DBMS for which the script should be generated
- BLOB Editor now supports BINARY and VARBINARY field types
- Linked Servers: now you can see the tables and views contained by a linked server
- Extract Database: now views and procedures are arranged in accordance with the object dependencies in the target script
- DB Explorer: the 'Add to Projects' command implemented; now objects can be added by specifying them in the corresponding dialog, and not only by dragging into the Projects folder
- When duplicating an object into another schema, for some objects its definition used to change incorrectly in DDL generated. Fixed now
- Grid: the 'Expand All' and 'Collapse All' context menu items added
- Grid: the 'Save column sorting' option is added for the 'Save and restore layout' mode
- Grant Manager: now User Editor and Role Editor can be called directly from this window
- Naming Rules: XML index support implemented
- CPU usage significantly decreased
- For the fields allowing NULL value the 'Null' sign is indicated in table definition now
- SQL Editor: for the queries which allow data insert the ability to import data into the table implemented
- SQL Editor: the Code Completion tool did not start automatically. Fixed now
- SQL Editor: in some cases the Results tab was not displayed. Fixed now
- SQL Editor: in some rare cases the 'Range Check Error' message appeared during query execution
- SQL Editor: upon opening object editor with the Ctrl+Enter shortcut used, the scheme was ignored. Fixed now
- SQL Editor: the Ctrl+Shift+Enter shortcut for pointing out the corresponding item in the DB Explorer tree worked incorrectly. Fixed now
- SQL Editor: the Code Completion tool used to work incorrectly with the "[" and the "]" symbols. Fixed now
- SQL Editor: after dragging objects from DB Explorer to SQL Editor the Undo command worked improperly in some rare cases
- SQL Editor: the 'Select All' command did not work correctly on text with long lines (more than 4096 symbols)
- Restore Database: it was impossible to use logical backup device containing the space character(s) in its name. Fixed now
- Job schedule, daily frequency: fixed the bug with improper handling of Daily frequency control (minutes used to be interpreted as hours and vice versa)
- The size and Scale for input parameter type could not be set when creating new partition functions. Fixed now

© 2008 EMS Database Management Solutions

- Table Editor, Data tab: fixed the bug with incorrect displaying of the bit-typed fields in the 'SQL filter' mode
- The dependencies were not displayed for XML schema collections. Fixed now
- To avoid script errors, the name of XML schema collection is always quoted now when creating an XML-typed field
- Import Data Wizard: when setting column correspondence for import from Excel, selecting a new column caused the cursor to move to the first column in the grid. Fixed now
- Database Registration Info: when applying changes, the originality of database aliases used not to be checked. Fixed now
- Visual Database Designer: switching to another visual scheme did not take any effect in context menu style. Fixed now

Scroll to top

Version 2.3.0.1

- Editor Options: added the Key Mapping section that allows you to set shortcuts for SQL editors
- XML Schema collections support is implemented
- Query Builder: query execution plan is now displayed as a graphical diagram
- Query Builder: queries containing user-defined functions are now supported
- Dependency Tree: the diagram can now be printed
- SQL Editor: now it is possible to rename and move query tabs
- SQL Editor: ability to execute statement under cursor added
- SQL Editor: query execution time is now displayed in the hh:mm:ss format
- SQL Editor: the timer in status bar area used to stay running on after aborting query execution. Fixed now
- SQL editors: now you can set a source code fragment as a comment with a single click
- SQL editors: next Bookmark, Previous Bookmark context menu items added
- SQL editors: with 'Trim trailing spaces' option on, an attempt to use Code Complete used to clear the whole line. Fixed now
- SQL editors: dragging the field names from SQL Assistant works perfectly now
- When opening Grant Manager window, it was impossible to edit statement permissions without forced refresh. Fixed now
- HTML Report: now you can include user and role information into reports
- HTML Report: ability to change header/footer of the report added
- Extract Database: now provides an option adding the CREATE DATABASE statement to the script
- Extract Database: server version is included into comments for the script now
- Environment Options: now it is possible to set refreshing objects timeout. Refresh of the objects can be terminated after timeout expired if there is an active transaction that made any changes to metadata
- The open/save backup file dialogs now look more like standard ones in behaviour
- Import Data Wizard: with the 'Use transactions...' option disabled, in the 'Commit changes manually' mode you are now prompted for an action on transaction
- Import Data Wizard: ability to resize the wizard window added
- Import Data Wizard: the 'Division by zero' error used to appear after importing data with the 'Commit changes manually' option switched on. Fixed now
- Import Data Wizard: having specified user-defined fields to import, the rest of them were filled with empty strings; so that the default values defined for the fields were not applied. Fixed now
- Trigger Editor: when altering a trigger, the schema/owner name was not added to the

© 2008 EMS Database Management Solutions

trigger name in the generated script, that in some cases raised errors. Fixed now
- Foreign key/check constraint editors: the 'Not for replication' option added
- Visual Database Designer: it was impossible to drop table via the corresponding context menu item when just a field, not the table itself, was selected. Fixed now
- Visual Database Designer: when dragging a table of a non-default schema (SQL Server 2005) from DB Explorer to the working area, an error used to appear and the table was not added. Fixed now
- Grid: the 'Paste Cell' context menu item did not work in the Lite version. Fixed now
- Grid: in some rare cases the data of of BIGINT fields were displayed improperly. Fixed now
- Grid: with the 'Column auto-width' option enabled, BLOB field columns used to become too wide. Fixed now
- SQL Script: confirmation of abort script execution added
- Restore Database: it was impossible to type in backup file name in the corresponding window. Fixed now
- Restore Database: prompt for database registration used to appear even if restoration was unsuccessful. Fixed now
- Job Editor: fixed the bug with wrong script generation when you saved the job in which the created schedule was deleted
- DB Explorer: it was unable to open index or trigger via the Recent Objects menu. Fixed now
- DB Explorer: the context menu used to stay on top when switching between applications. Fixed now
- DB Explorer: after table duplication the indices and triggers of the new table were not displayed in the tree. Fixed now
- Export Data Wizard: with the 'Export empty tables' option disabled, on attempt to export an empty table there appeared an error containing no text. Fixed now
- Export Data Wizard: in some cases, when exporting data to MS Excel, the 'Data format \ Styles' options used to work incorrectly. Fixed now
- Export Data Wizard: when exporting data to MS Excel, the user-adjusted Boolean True format used to be ignored. Fixed now
- Detach Database: having used the 'Clear connections' function, the database was still displayed in the DB Explorer tree and further attempts to access its objects returned errors. Fixed now
- Many other small improvements and bug-fixes

Scroll to top

Version 2.2.0.1

- SQL Editor: query execution plan is now displayed as a graphical diagram
- SQL Formatter: formats SELECT, INSERT, UPDATE, DELETE queries (available from SQL Editor context menu)
- Grid: when pasting records from the clipboard, you can now set correspondences between clipboard columns and query columns
- Form View: Memo-fields editor is extended (can be customized within the Grid | Advanced section of the Environment Options dialog)
- Blob View: 'Word Wrap' option for text added
- Grid: 'Set to Empty String' and other similar operations do not result in posting records in the table any more
- SQL Script: the progress of script execution is now displayed and the execution process can be stopped
- SQL Script: several windows can now be opened simultaneously
- SQL Script: SQL Script used to hang when a large (>10Mb) script was edited and

© 2008 EMS Database Management Solutions

Enable Parsing was on. Fixed now
- SQL Script: now some additional statement types can be parsed
- SQL Editor: query execution plan can now work with SQL Server 2005
- SQL Editor: now there are no limitations for the size of edited scripts
- Field Editor: default constraints now can be renamed
- DTS Packages: now you can remove imported packages
- Table Data: not all the records added with the following COMMIT in "Use transactions in Data tab of object editors, SQL Editor and Query Builder" mode (Environment Options | Tools) were displayed. Fixed now
- Extract Database: with the "Insert COMMIT after each block" option on, COMMIT was added without corresponding BEGIN TRANSACTION. Fixed now
- Extract Database: empty NOT NULL image fields used to be displayed incorrectly. Now fixed
- Extract Database: dbo name was missing when generating INSERT INTO [dbo].[table_name]
- Export as SQL Script: with the "Insert COMMIT after each block" option on, COMMIT was added without corresponding BEGIN TRANSACTION. Fixed now
- Dependency Tree: wrong dependency order was shown for Assemblies, Aggregates, Partition Functions and Partition Schemas: the parent object used to be displayed after the dependent one. Fixed now
- Visual Database Designer: changing the table name and editing the fields did not affect the currently opened graphical diagram. Fixed now
- Reorder Columns, Duplicate Table, Extract Metadata: tables with timestamp/rowversion fields are processed correctly now
- Switching program interface language took a long time if there were any databases connected. Fixed now
- Database Properties: if the "auto file growth" option was off for a DB file, it could not be turned on. Fixed now
- Attach Database: it was impossible to set login and password using SQL Authentication. Fixed now
- Attach Database: the 'File name' field did not have the Explorer button to call the "Open File" dialog. Fixed now
- UDF Editor: options of the 'For new columns only' group are processed properly now
- DB Explorer: it was impossible to duplicate triggers and indices. Fixed now
- DB Explorer: after dropping a trigger or index it was still displayed in the tree
- Query Builder stores queries between sessions now
- The 'Don't fill server lists in connection forms' option is added; the option disables retrieving information about all SQL servers available on the local network because in some rare cases it might take a long time
- The border of the SQL Editor window is now painted according to the current visual scheme
- Procedure Editor: if the FLOAT type was specified for a procedure parameter and had a value equal to 0, it used to be processed as NULL. Fixed now
- Restore Database: sometimes due to the improper size of the dialog the directory tree at the Select Backup Set step was not seen. Fixed now
- The corresponding documentation page did not open for some forms when pressing F1. Fixed now
- Synonym Editor, Aggregate Editor: it was impossible to create an object when the <Default> scheme was selected because of wrong script generation. Fixed now
- Create Database wizard: if log and data files had the same names when creating a database, the log file was not created. Fixed now
- Aggregate Editor: if a parameter does not contain the at ("@") sign, it is added automatically now

© 2008 EMS Database Management Solutions

- Aggregate Editor: turning on the Max Size option disables the Size field now
- Fixed the bug with the Code Completion drop-down list which could be partially covered with Windows Taskbar
- On continuous holding F5 button to refresh the objects in DB Explorer they used to double. Fixed now
- Table/Field Editor: identity seed is now displayed properly for values greater than 65536
- Check Editor: the "Check existing data" option is not cleared anymore after compiling the current check
- After renaming the object its name was not refreshed in DB Explorer context menu. Fixed now
- For some forms the Refresh operation could not be performed with F5 key used. Fixed now

Scroll to top

Version 2.1.0.1

- Partition functions and partition schemes are now supported
- Added the ability to work with linked servers
- German localization is implemented (Michael Albrecht)
- Export/Import wizards: added the "Save Current Settings as Default" drop-down menu and the "Reset Saved Settings" option in Templates
- New editor options added: trim trailing spaces, Drag and drop text, Group undo, Keep caret horizontal position
- Grid: large queries were opened very slowly because the switch to Grid Mode Load All Rows was executed after the grid was filled. Fixed now
- Export Data Wizard: added the Finish button that runs the Wizard at any moment
- Export as SQL Script and Extract Database wizards gave values of datetime type in the formats predefined in Windows settings, which sometimes caused execution errors. Fixed now
- When creating a CLR user-defined data type, correct script is generated now
- Unicode data are now displayed correctly for the system language for non-Unicode programs
- Object editors: when pressing F5 in the subobjects lists, the 'Access Violation' error occurred. Fixed now
- Window List: added the 'Close window' item to the context menu
- The "Don't save query text" option available back in version 1.x of the program is now restored in SQL Editor
- Grid: adding level bound by several fields caused “Invalid column name” error. Fixed now
- Save Settings: multi-line keyboard templates were blank when imported from a file. Fixed now
- Keyboard templates: the #Date tag gave the wrong date. Fixed now
- Printing now works correctly in Aggregate, DDL Trigger and Synonym editors
- Index editor: clicking the header of selected fields list returned the 'Access violation' error if the list was empty. Fixed now
- SQL Script is now saved with any content of the editor
- Allow parameters in SQL (Environment Options | Tools) for currently opened Query Builder and SQL Editor windows can now be activated; also, when running queries asynchronously, the values of parameters were not taken. Fixed now
- Other small improvements and bug-fixes

Scroll to top

© 2008 EMS Database Management Solutions

Version 2.0.5.1

- When closing the program it gave a number of errors if no other programs were running. Now fixed
- Field default values are now displayed correctly now when working with SQL Server 2005
- SQL Editor: text search did not work when called from the context menu. Fixed now
- SQL Editor: sometimes the Limit bar was displayed for grid. Fixed now
- SQL Editor: code completion across databases did not work by table alias with owner (schema) not specified. Fixed now
- SQL Editor: code completion by table aliases did not work after 10-15 table lines down the text. Fixed now
- SQL Editor: when adding a new tab and trying to delete a row, the "Access Violation" error emerged. Fixed now
- Database Registration Info: Display Options were not saved after closing/restarting the program. Fixed now
- Now the size of stored procedure and UDF parameters is shown correctly
- SQL Script: the "Access Violation" error arose upon calling Code Completion if there were no connected databases. Fixed now
- DB Explorer: now the parameter information of some stored procedure types is displayed correctly in SQL Assistant
- Grid: filtering by value that had an apostrophe (') character did not function. Fixed now

Scroll to top

Version 2.0.0.1

Basic features:

Latest SQL Server versions support
- Full compatibility with all SQL Server versions from 7 to 2005
- Full support of SQL Server 2005 permission system
- DDL Triggers, Synonyms and Aggregates support
- Typed XML columns

Significantly improved user interface
- New well-described Wizard System to perform complex tasks easily
- Fast access to most essential actions using Navigation bars in most of the windows
- New Windows Toolbar to switch windows easily, like in Windows Task Bar
- Office 11 and Windows XP visual schemes support

Improved Database Explorer
- Creating tabs for any tree node to access it in one-click
- Work with multiple selected objects at a time
- Much more convenient toolbar and context menu

New powerful tools to perform SQL Server administrative tasks
- Server Properties and Database Properties windows
- DTS Packages management tool

Visual Database Designer to handle database structure in a few clicks
- Creating diagram of existing database using Reverse engineering tool
- Dividing diagram into different subject areas located on separate pages
- Navigating through diagram using preview navigation tool
- Creating/editing/dropping tables and table fields visually setting links between them
- Saving as image or printing database diagram
- Two diagram notations (IDEF1x and IE)

© 2008 EMS Database Management Solutions

- A lot of visual customization options

New powerful data management tools

- Plenty of useful tools such as data grouping, summarizing, fast searching and filtering
- Master-Detail View to work with two linked tables at the same time
- Table and Card Views to see data the way you like
- Ability to copy and paste selected records
- Advanced printing system
- Fully customizable interface

Improved report management

- The new Create Report Wizard allows you to create a simple report in a few clicks
- Managing reports in the same way as if they were database objects: accessing reports directly through DB Explorer tree

and more...

Scroll to top

Version 1.8.5.1

- Support of Microsoft® SQL Server 2005
- When an error arises in SQL Editor or in the Changing Metadata window, the cursor is placed to the proper place now
- Creating table: enhanced editing abilities in the Fields grid
- Editor Options: added the 'Convert Tab character to spaces' option
- SQL Editor: implemented the 'Don't save query text' option
- SQL Editor: added an ability to reset current execution point - the corresponding item is added to the context menu
- Table Editor: confirmation request was not displayed upon deletion of table subobjects even if the corresponding option was on. Now fixed
- Query Builder: the WHERE part was parsed incorrectly if there were several IN operators - the list was duplicated
- Timeouts did not work in SQL Script
- Sometimes when disconnecting from a database, its objects were refreshed if viewing table subobjects was off
- Grid: the time values were not shown on attempt to enter the edit mode for datetime fields
- When trying to open a procedure with number (e.g. proc;1), there occurred an error saying that the procedure was not found in the database
- The program hanged when using keyboard templates if Windows clipboard had the hash (#) sign
- To-Do List: upon editing an item a second copy of the item used to appear in the list. Fixed now
- Table Editor, the Dependency tab: tables with foreign keys referring to a currently opened table were not shown in the dependent objects list
- If the Clustered checkbox was off when creating an index, keyword NONCLUSTERED was not added to the script, which led to creating a clustered primary key
- Shortcuts for tabs did not work. Now fixed
- Resolved the issue when the state of the main form window was not restored when running the application
- It was impossible to edit fields having a dot (.) character or the 'on' word in their name. Now fixed
- Code Completion did not work with table aliases which had the dollar ($) sign in their name, e.g. "TableName$". Now fixed
- Editor Options: decreased the minimum possible value for Quick Code delay
- Other small improvements and bug-fixes

Scroll to top

© 2008 EMS Database Management Solutions

Version 1.8.0.1

- Implemented the ability to generate names for the table subobjects on their creating automatically; you can customize the name templates or disable this option using the Naming Rules section of the Environment Options dialog
- Extract Metadata: changed the sequence of some operator types to register possible dependencies between objects; included support for identity and calculated fields; added an ability to generate the IF EXISTS... DROP statements
- SQL Editor: now it is possible to view the query results under the same tab with the query text
- There is a new possibility to disable the Changing Metadata window using the corresponding option within the Confirmations section of the Environment Options dialog
- Query Builder: the "Access Violation" error emerged on attempt to add a criterion and close the form. Fixed now
- Table Data: the 'DataD: Cannot perform this operation on a closed dataset' error does not appear anymore
- On creating a unique index you can now select whether to create an index or a constraint
- Grid: added the possibility to set format for the money fields within the Grid tab of the Environment Options dialog
- Table Data: fixed the bug when the Grid layout was not saved
- The 'Make hidden Table Explorer visible on new table window' option is substituted for 'Hide Table Explorer'; a new button is added to the toolbar of Table Editor to show/hide the explorer
- If a field was indexed, you could not change its type or the NOT NULL constraint - the script error arose. Fixed now
- Extract Metadata & Export as Insert: when extracting BLOB data, the final bytes with 0x00 value were cut. Fixed now
- Export Data Wizard: when the 'Current record only' option was selected the cursor jumped to the first record and the first record was exported. Fixed now
- When running the program, it was impossible to shut down Windows. Fixed now
- Print Metadata: fixed the bug when the Primary Key and Unique columns of table fields were not filled out
- Now the 'Help | Send bug report to' main menu item allows you to access our Support Ticket System
- The 'Class TSynEdit not found' error arose on attempt to open the localization form. Fixed now

Scroll to top

Version 1.7.0.1

- Implemented a new form for creating tables: now it allows you to define all table fields before table creation
- Import Data Wizard: added a possibility of defining the error log file name
- Backup/Restore: while working with large databases, the operation was canceled by timeout in 30 seconds. Now the problem is solved
- Fixed the following bugs concerned with Code Completion: processor loading was raised after the window "code completion" had been opened with a non-default localization used; the focus bounced back to the top of the list after one started typing; the view was not canceled in case a space character was entered or the cursor was moved after typing a letter or a number; there was a problem with

© 2008 EMS Database Management Solutions

re-opening the code completion window after inserting a keyword
- Export as Insert: in float fields the dot (.) character is used as a fractional part separator now
- The bug that caused only tree-choice of backup filename in case of using non-logical device is fixed now
- Wrong script was generated when a Not Null field with a Unique constraint and without default value had been added. Fixed now
- After a field with a unique non-constrained index was changed, wrong script was generated. Fixed now
- Available permissions were defined incorrectly for UDF. Fixed now
- Edit login: the password could not be cleared. Fixed now
- Confirmation is requested now on attempt to delete query pages
- Integer field changes (if the field was not in edit mode) were saved incorrectly. Fixed now
- The font size changes in Environment Options were not applied to menus and toolbars. Fixed now
- Corrected the "RecordIndex out of range" error that used to appear when editing database file list
- The Row Guid flag could not be changed for fields with type "uniqueidentifier". Fixed now
- Sometimes after upgrade to a new product version the language settings could not be changed. Fixed now
- Export to XML: if table field names contained spaces, the summary file would not open. Fixed now
- Changing Metadata window: now the editor font is determined by the database properties instead of the defaults used previously

Scroll to top

Version 1.6.0.1

- Significantly improved the Export data dialog: now you can add notes, hyperlinks and charts based on exported data to Excel documents; you can also define graphical styles for column captions, data, footer, and header for export to RTF documents; ability to define strip styles is also included
- Significantly improved Import Data Wizard to make most of EMS QuickImport 2 component suite new abilities available in MS SQL Manager: import from XML format is available now; a possibility of importing data from different Excel sheets is implemented; a possibility of importing formulas from Excel is added
- We have implemented a new service - Backup Devices Manager: you can access this tool using the Services menu
- SQL Editor: now you can use the Shift+Ctrl+Space key combination to view function parameters after you enter a function name in your code; the Shift+Ctrl+P key combination can be used to insert names of the function parameters into your code
- Duplicate Table: in this version you can copy the table metadata and data to another schema/database
- Added a new localization - Chinese (Simplified)
- The Database Metadata log now contains only records that are executed completely and not rolled back
- The SQL Editor log now contains only statements that are executed successfully. You can disable this feature using the corresponding option within the SQL Editor section of the Environment Options dialog
- Now the program correctly displays all objects within the Dependencies tab
- If a table contained a calculated field with more than 4000-char length, then the 'List

© 2008 EMS Database Management Solutions

index is out of bounds' error appeared. Now fixed
- Form View: NULL values of date fields were displayed as 30.12.1899. Now fixed

Scroll to top

Version 1.5.0.1

- Implemented two new services - Backup Database and Restore Database: you can access these tools through the Services menu
- Data Grid: now you can filter data in the 'Load Visible Rows' mode; the program exports only the filtered data
- Improved the reporting tools (Print Metadata and Print Data tabs of the program windows): added powerful filters for exporting data to different formats, such as XLS, RTF, HTML, BMP, JPEG and TIFF
- Now the program compiles large procedures, views, user-defined functions and triggers without losing spaces in the object body
- Now you can create fields flagged with the Not Null constraint without any problems
- SQL Script: fixed a bug that caused the "Control 'tlMessages' has no parent window" error after successful script compilation
- Now the program correctly creates fields with the Collation and the Default parameters defined; correct DDL is also built for tables that contain fields with these parameters
- SQL Script: now the program correctly saves scripts upon pressing the Ctrl+S shortcut
- Fixed a bug that caused high processor loading when typing text in SQL Editor if the selected interface language was different from default
- Now the program correctly displays connection errors that appear when connecting to the database
- If the database contained two procedures with names different only in character case, then the 'Index is out of bounds' error appeared. Now fixed
- Fixed the 'Index is out of bounds' error that appeared on attempt to create a job schedule
- If an operation caused many errors on the server, then the program returned only the last one; now all errors are returned
- SQL Script: fixed a bug that caused incorrect word wrapping in the Messages window
- Fixed a bug with a button on the Form View tab that shows BLOB content
- View Editor: fixed a bug that sometimes caused the 'List index is out of bounds' error when switching between the Description and the Data tabs
- Extract Metadata: now the program correctly extracts values of BIT fields
- Small improvements and bug-fixes

Scroll to top

Version 1.4.0.1

- Implemented four new services: Jobs, Job History, Alerts, and Operators; these services are available through the respective items of the Services menu
- Added Romanian localization to the installation package
- SQL Editor: the grid did not show results of any queries after executing query that returned empty dataset in asynchronous mode. Now fixed
- The 'Class Tcxbutton not found' error appeared when trying to open the Keyboard Templates dialog. Now fixed
- SQL Editor: Fixed bug with displaying number of affected rows when executing queries in asynchronous mode
- SQL Script: The Rollback on Abort option works correctly now

© 2008 EMS Database Management Solutions

- Added the vertical scrollbar to the stored procedure parameters dialog; now you can define the procedure parameters without any difficulties
- Fixed the bug giving the 'Insufficient key column information for updating or refreshing' error when refreshing query results
- Fixed several bugs in the Code Insight system; now the system correctly works with queries that contain aliases
- Export data: now the Export dialog automatically adds the file extension to the file name
- Now you can call the Replace Text dialog using the Ctrl+R keyboard shortcut in SQL Editor and in SQL Script editors

Scroll to top

Version 1.3.0.1

- Implemented the threaded query execution: now you can stop query execution at any time. To enable this feature, check the "Asynchronous query execution" option within the Tools tab of the Environment Options dialog
- SQL Editor: now you can view the query execution plans. You can view the plans by pressing the corresponding button on the SQL Editor toolbar, or by checking the "Explain Query" option within the SQL Editor section of the Environment Options dialog
- Now grids automatically switch to the "Load Visible Rows" mode on opening queries with a large amount of result rows
- Now the DDL tab and the Extract Metadata Expert show object owners that differ from "dbo"
- We have fixed a bug with stored procedure execution
- Now the "Refresh" item of the DB Explorer context menu includes the type of object to refresh
- Now the table explorer is correctly refreshed after deleting the table subobjects in the DB Explorer
- Some small improvements and bug-fixes

Scroll to top

Version 1.2.0.1

- Added the Services menu which contains two database services - Attach Database and Detach Database
- Now you can run stored procedures directly from Procedure Editor
- Now MS SQL Manager supports operation timeouts. You can setup the timeout values using the Timeouts panel available within the Tools section of the Environment Options dialog
- Added the "Try to open server cursor" option to the SQL Editor section of the Environment Options dialog
- The "Field 'Name' not found" error appeared on registering database for some servers. Now fixed
- Now MS SQL Manager correctly opens tables with large NUMERIC and DECIMAL fields
- Fixed a bug in the Code Completion system. Now it correctly displays view fields
- Fixed a bug with comments on altering procedures and functions
- The "Cannot focus a disabled or invisible window" error appeared in some cases. Now fixed
- SQL Editor: fixed a bug when the "There is no active transaction" message appeared during query execution, with transactions disabled
- Query Builder: fixed a bug with dragging and dropping tables from the explorer panel to the workspace

© 2008 EMS Database Management Solutions

- SQL Script: fixed bug in building the object tree, if the script contains objects, which names differ only by char cases
- Editor Options: now you can customize colors for SQL functions and SQL data types
- Now SQL Editor saves the cursor position for every script, which is currently opened
- Added 'Copy' and 'Paste' items to the grid context menu
- Now MS SQL Manager does not show the _WA_Sys_xxx indices
- A large amount of bug-fixes in Query Builder
- Some small bug-fixes and improvements

Scroll to top

Version 1.1.0.1

- Implemented an opportunity to view dependencies between database objects: this feature is accessible through the Dependencies tab of all Object Editors
- Considerably increased connection speed
- Now you can set almost all database options using the Database Properties dialog
- Implemented Default Editor and Rule Editor
- Now object names containing unavailable symbols are enclosed in square brackets []
- Now text editors are displayed much faster than in the previous version
- Fixed a bug with connection to database with SQL Authentication if the 'Password' field was not empty
- The "Cannot sort a row of size XXXX, which is greater than the allowable maximum of 8094." message appeared upon connection to databases containing large objects. Now fixed
- Fixed error "Field 'sql2' not found" that used to occur on connection to databases containing large triggers
- Fixed the "Invalid column name 'Collation'" error that appeared on connection to the database if the 'Collation' property was set to a case-sensitive value
- Now you need to confirm record deletion on pressing the Delete key in the grid
- Fixed a bug with displaying object comments
- The 'Login prompt before connection' option available in the Database Info dialog works properly now
- All windows with text editors are displayed correctly now
- Now View Editor generates correct update script if there is a comment before the view definition
- Small bug-fixes and improvements

Scroll to top

Version 1.0.0.1

Basic features:

- Easy management of all MS SQL objects: creating/dropping databases; creating/dropping/altering tables, views, procedures, etc.
- Creating/dropping/altering all the table subobjects
- Viewing/editing data in tables
- Support of all MS SQL field types
- Multiple database connections
- Powerful Login Manager and Grant Manager
- SQL Editor with syntax highlight
- Visual Query Builder
- Powerful BLOB Viewer/Editor that provides several ways of viewing BLOB data
- Export data to 14 popular formats, including MS Access, Excel, RTF (Word), HTML, etc.

© 2008 EMS Database Management Solutions

- Import data from most popular formats, including MS Excel, Access, etc.
- Customizable program interface
- Detailed Help System and User's Guide

and more...

Scroll to top

See also:

What's new

© 2008 EMS Database Management Solutions

1.9 EMS SQL Manager FAQ

Please read this page attentively if you have questions about **EMS SQL Manager 2008 for SQL Server**.

Table of contents

Product questions
- What is EMS SQL Manager for SQL Server?
- What is the difference between Full/Lite editions of EMS SQL Manager for SQL Server?
- What do I need to start working with EMS SQL Manager for SQL Server?
- What is the difference between the Export/Import functions in SQL Manager and the Data Export/Import utilities?
- What is the difference between the Query Builder module in SQL Manager and the SQL Query for SQL Server utility?
- What is the difference between the Extract Database function in SQL Manager for SQL Server and the DB Extract for SQL Server standalone utility?

Common questions
- I cannot connect to SQL Server. Why?
- How can I connect to a named instance of SQL Server 2000 in EMS SQL Manager for SQL Server?
- I can access my server via an alternative port. I have searched for a setting in EMS SQL Manager for SQL Server to alter the connection port, but have not yet been able to locate this. How would I go about altering this port?
- I can't modify DDL. Why?
- How can I customize data formats in grid?
- I need to perform some changes in database objects of my test database and then make the same changes on master database. Are there any tools for this purpose in SQL Manager for SQL Server?
- When I create database objects, their names are always converted to lower case. How I can prevent it?
- I am trying to create a report in Report Designer, but I can't get access to the table data: 'Band data source' list is empty.
- How can I speed up my work with large tables?
- I have a table with appr. 1000 records only, and a large number of fields. Opening this table on the 'Data' tab takes too much time.
- I get error 'Timeout expired' when I try to execute a query in SQL Editor or when I perform some operations with database objects.

Export/Import questions
- I'm trying to export data from a table, but TEXT fields are not exported.
- What is the difference between the "Extract Database" and "Export as SQL Script" functions?
- How can I change the default directory where exported data are saved?

Question/answer list

Product questions

Q: What is EMS SQL Manager for SQL Server?

© 2008 EMS Database Management Solutions

A: EMS SQL Manager for SQL Server is a powerful graphical tool for Microsoft® SQL Server™ administration and development. SQL Manager works with all SQL Server versions from 7 to 2005 and supports all of the latest SQL Server features. It offers plenty of powerful tools for experienced users to satisfy all their needs. SQL Manager makes creating and editing MS SQL Server database objects easy and fast, and allows you to run SQL scripts, manage users and their privileges, build SQL queries visually, extract, print and search metadata, export data to 14 available formats and import them from most popular formats, view and edit BLOB fields, and much more. SQL Manager 2008 for SQL Server has a new state-of-the-art graphical user interface with well-described wizard system, so clear in use that even a newbie will not be confused with it.

Q: What is the difference between Full/Lite editions of EMS SQL Manager for SQL Server?

A: These editions of SQL Manager for SQL Server differ in price and features. To register SQL Manager for SQL Server, see the Purchase page, and to learn about the difference in features please go to our Feature Matrix page.

Q: What do I need to start working with EMS SQL Manager for SQL Server?

A: First of all, you must have an opportunity to connect to some local or remote SQL Server 7, MS SQL Server 2000, MSDE, SQL Server 2005 (except Compact Edition) to work with EMS SQL Manager for SQL Server. You can download SQL Server from http://www.microsoft.com/downloads/. Besides, you need your workstation to satisfy the system requirements of SQL Manager 2008 for SQL Server.

Q: What is the difference between the Export/Import functions in SQL Manager and the Data Export/Import utilities?

A: The Data Export/Import for SQL Server utilities include some additional features which are not available in SQL Manager, such as:

- export/import data from/to several tables simultaneously;
- export/import data from/to tables selected from different databases on one host;
- a command-line utility to export/import data using the configuration file with all the export/import options.

Q: What is the difference between the Query Builder module in SQL Manager and the SQL Query for SQL Server utility?

A: First of all, SQL Query for SQL Server works faster as it is a much lighter product. Besides, it provides additional features for query building, e.g.:

- keeping query history which allows you to rollback to any edited query;
- various interface improvements for more productive and easier work.

Q: What is the difference between the Extract Database function in SQL Manager for SQL Server and the DB Extract for SQL Server standalone utility?

A: The DB Extract for SQL Server utility includes some additional features which are not available in SQL Manager, such as:

- extracting metadata and/or data from several databases on one host;
- a console application for performing extract in one-touch;
- faster extraction speed.

Scroll to top

Common questions

© 2008 EMS Database Management Solutions

Q: I cannot connect to SQL Server. Why?
A: There are several reasons for this issue:
1. Please note that if Microsoft® SQL Server™ is installed as a named instance, you should enter the name of your machine and the instance name in the following format: *computer_name**sqlserver_instance_name* (e.g. "*MYCOMPUTER**SQLEXPRESS* ").
2. If you connect to the remote server, make sure that the connection is not blocked by a firewall. Ask your system administrator to open the appropriate ports.
3. Make sure that SQL Server is started and you have a valid login to access the server.
4. Make sure that the client protocols are correctly tuned. User SQL Server Configuration Manager on connection to SQL Server 2005 (it is installed with Native Client) or SQL Server Client Network Utility (*%Windows_directory%**system32**cliconfg.exe*) on connection to earlier versions.
5. Make sure that remote connection is enabled on the server and the server protocols are correctly configured. Use SQL Server Surface Area Configuration on SQL Server 2005 and SQL Server Network Utility (*%Program_files%*\Microsoft SQL Server\80\Tools\Binn\svrnetcn.exe) on earlier server versions.
6. If you use a named instance of the server with dynamic TCP/IP port, make sure that the SQL Browser service is running.
7. If you connect remotely, try using server IP address instead of its name to ensure that DNS works well.

Q: How can I connect to a named instance of SQL Server 2000 in EMS SQL Manager for SQL Server?
A: If you have troubles when connecting to a named instance of SQL Server 2000, please try the following:
- run SQL Server Client Network Utility (*%Windows_directory%**system32**cliconfg.exe*) on the client computer where EMS SQL Manager for SQL Server is running;
- proceed to the second (Alias) tab of the Utility;
- add a new server alias with Server Name in the following format: *computer_name**sqlserver_instance_name*, and TCP/IP as Network library;
- use the server alias name as the host name in the Database Registration Info dialog of EMS SQL Manager.

Q: I can access my server via an alternative port. I have searched for a setting in EMS SQL Manager for SQL Server to alter the connection port, but have not yet been able to locate this. How would I go about altering this port?
A: To connect to SQL Server 2005 running on an alternative port, do the following:
- run SQL Server Configuration Manager (it is installed with Native Client) on the client computer where SQL Manager is running;
- select the SQL Native Client Configuration | Aliases node;
- add a new server alias and set the port number, TCP/IP as the protocol and the server name;
- use the server alias name as the host name in the Database Registration Info dialog of EMS SQL Manager.

To connect to SQL Server 2000 (or earlier) running on an alternative port, do the following:
- run SQL Server Client Network Utility (*%Windows_directory%**system32**cliconfg.exe*) on the client computer where SQL Manager is running;
- proceed to the second (Alias) tab of the Utility;
- add a new server alias and set the server name, port number and TCP/IP as Network library;
- use the server alias name as the host name in the Database Registration Info dialog of

© 2008 EMS Database Management Solutions

EMS SQL Manager.

Q: I can't modify DDL. Why?
A: The DDL tab of the SQL Manager editors is read-only. It displays object structure as SQL text and reflects the operations you perform over the object under other editor tabs. To modify an object, you can copy the text to the clipboard and edit it using SQL Script Editor. For more details refer to Viewing object DDL structure.

Q: How can I customize data formats in grid?
A: You can customize all display formats: integer, float, date, time and datetime using the Color & Formats page of the Environment Options dialog.

Q: I need to perform some changes in database objects of my test database and then make the same changes on master database. Are there any tools for this purpose in SQL Manager for SQL Server?
A: The Database Registration Info dialog provides the Logs section where you can enable logging metadata changes performed over the database and/or SQL queries executed in SQL Editor.

Q: When I create database objects, their names are always converted to lower case. How I can prevent it?
A: Please try to switch off the "Convert created objects' names to lower case" option within the Object Editors section of the Environment Options dialog.

Q: I am trying to create a report in Report Designer, but I can't get access to the table data: 'Band data source' list is empty.
A: It is recommended to use Create Report wizard which is run by right-clicking the Reports node in DB Explorer and selecting the 'New Report...' context menu item. The wizard will create all necessary data sources.
If you still want to use Report Designer, to get the data source in a report, you need to add database and query components to the report.

Q: How can I speed up my work with large tables?
A: For your convenience and to speed up your work, the Data Grid allows customizing a number of data display parameters. Here are the most important of them (accessible through the Grid | Data Options section of the Environment Options dialog):
- *Limit options in table and view editors*. The 'Select all records of a table' option will enable you to see all table records without extra references to the server, yet in case of large tables or low speed connection channel the data may be fetched with huge delays and the incoming traffic might grow considerably. This mode is recommended when working with local databases or in a private network. The 'Select only' mode restricts the maximum number of records returned after the query. A man cannot process a massive amount of information at once. Hence, we came up with this mode. This mode speeds up table data viewing considerably, prevents hanging and connection timeout. It is recommended to work with large tables, in case of low speed connection channels and when the traffic volume is of importance. This is the default mode. When in this mode, enabling the 'Use SQL sorting in data view' and 'Use SQL filter in data view' options comes really helpful.
- *Default Grid Mode*. This option defines whether the requested rows will be loaded in the Grid all at once ('Load all rows'), or in parts ('Load visible rows') as the user scrolls down table data. The first mode increases the query opening time, but speeds up scrolling. In the second mode the query opens very fast, but there might be delays when navigating the grid.

© 2008 EMS Database Management Solutions

- *Use SQL sorting in data view*. With this option enabled, when the user sets data sorting, a new request is sent to the server, and the result is shown anew in the Grid. Enabling this option is very helpful with the 'Select only' mode enabled, but it increases the traffic. When this option is disabled, sorting is made on the client computer without any references to the server but only loaded data are sorted. That is, if you set 'Select only 1,000 records', then only these records will be sorted.
- *Use SQL filter in data view*. With this option enabled, the data will be sorted on the server to which a new request is sent, and the result will be shown anew in the Grid. Enabling this option is also helpful when the 'Select only' mode is on, but it increases the traffic. When this option is disabled, the filtering is made on the client computer without any references to the server but only loaded data are filtered. That is, if you set 'Select only 1,000 records', then only these records will be filtered.

We recommend that you set the following option values to achieve maximum efficiency when working with large tables:

- *Select only* – ON
- *Load visible rows* – ON
- *Use SQL sorting in data view* – ON
- *Use SQL filter in data view* – ON

Q: I have a table with appr. 1000 records only, and a large number of fields. Opening this table on the 'Data' tab takes too much time.

A: You need to set the Grid Mode for the table to 'Load Visible Rows'. Please right click within the table grid and select the 'Grid Mode' | 'Load Visible Rows' context menu item.

Q: I get error 'Timeout expired' when I try to execute a query in SQL Editor or when I perform some operations with database objects.

A: You need to increase timeout values within the Tools | Timeouts section of the Environment Options dialog, or set them to 0 (unlimited).

Scroll to top

Export/Import questions

Q: I'm trying to export data from a table, but TEXT fields are not exported.

A: Fields of types TEXT, BINARY, IMAGE, etc. are not exported by default. You should select these fields manually at the Selecting fields for export step.

Q: What is the difference between the "Extract Database" and "Export as SQL Script" functions?

A: Export as SQL Script is intended for exporting table data that will be inserted into a database system other than SQL Server. Use Extract Database Wizard to copy metadata and/or data to a database on SQL Server afterwards.

Q: How can I change the default directory where exported data are saved?

A: Follow the steps below to change the default directory:

1. Right-click the database alias in DB Explorer and select the 'Database Registration Info...' context menu item (you can also find this item in the 'Database' main menu) to open the Database Registration Info dialog.
2. Proceed to the Directories section within the dialog.
3. Set the 'Default directory for Export Data'.

Scroll to top

© 2008 EMS Database Management Solutions

If you still have any questions, contact us at our Support Center.

© 2008 EMS Database Management Solutions

1.10 Other EMS Products

Quick navigation

MySQL

Microsoft SQL

PostgreSQL

InterBase / FireBird

Oracle

IBM DB2

DBISAM

Tools & components

MySQL

SQL Management Studio for MySQL
EMS SQL Management Studio for MySQL is a complete solution for database administration and development. SQL Studio unites the must-have tools in one powerful and easy-to-use environment that will make you more productive than ever before!

SQL Manager for MySQL
Simplify and automate your database development process, design, explore and maintain existing databases, build compound SQL query statements, manage database user rights and manipulate data in different ways.

Data Export for MySQL
Export your data to any of 15 most popular data formats, including MS Access, MS Excel, MS Word, PDF, HTML and more.

Data Import for MySQL
Import your data from MS Access, MS Excel and other popular formats to database tables via user-friendly wizard interface.

Data Pump for MySQL
Migrate from most popular databases (MySQL, PostgreSQL, Oracle, DB2, InterBase/Firebird, etc.) to MySQL.

Data Generator for MySQL
Generate test data for database testing purposes in a simple and direct way. Wide range of data generation parameters.

DB Comparer for MySQL
Compare and synchronize the structure of your databases. Move changes on your development database to production with ease.

DB Extract for MySQL
Create database backups in the form of SQL scripts, save your database structure and table data as a whole or partially.

SQL Query for MySQL
Analyze and retrieve your data, build your queries visually, work with query plans, build charts based on retrieved data quickly and more.

Data Comparer for MySQL
Compare and synchronize the contents of your databases. Automate your data migrations from development to production database.

Scroll to top

Microsoft SQL

© 2008 EMS Database Management Solutions

SQL Management Studio for SQL Server
EMS SQL Management Studio for SQL Server is a complete solution for database administration and development. SQL Studio unites the must-have tools in one powerful and easy-to-use environment that will make you more productive than ever before!

SQL Manager for SQL Server
Simplify and automate your database development process, design, explore and maintain existing databases, build compound SQL query statements, manage database user rights and manipulate data in different ways.

Data Export for SQL Server
Export your data to any of 15 most popular data formats, including MS Access, MS Excel, MS Word, PDF, HTML and more

Data Import for SQL Server
Import your data from MS Access, MS Excel and other popular formats to database tables via user-friendly wizard interface.

Data Pump for SQL Server
Migrate from most popular databases (MySQL, PostgreSQL, Oracle, DB2, InterBase/Firebird, etc.) to Microsoft® SQL Server™.

Data Generator for SQL Server
Generate test data for database testing purposes in a simple and direct way. Wide range of data generation parameters.

DB Comparer for SQL Server
Compare and synchronize the structure of your databases. Move changes on your development database to production with ease.

DB Extract for SQL Server
Create database backups in the form of SQL scripts, save your database structure and table data as a whole or partially.

SQL Query for SQL Server
Analyze and retrieve your data, build your queries visually, work with query plans, build charts based on retrieved data quickly and more.

Data Comparer for SQL Server
Compare and synchronize the contents of your databases. Automate your data migrations from development to production database.

Scroll to top

PostgreSQL

SQL Management Studio for PostgreSQL
EMS SQL Management Studio for PostgreSQL is a complete solution for database administration and development. SQL Studio unites the must-have tools in one powerful and easy-to-use environment that will make you more productive than ever before!

SQL Manager for PostgreSQL
Simplify and automate your database development process, design, explore and maintain existing databases, build compound SQL query statements, manage database user rights and manipulate data in different ways.

Data Export for PostgreSQL
Export your data to any of 15 most popular data formats, including MS Access, MS Excel, MS Word, PDF, HTML and more

© 2008 EMS Database Management Solutions

Data Import for PostgreSQL
Import your data from MS Access, MS Excel and other popular formats to database tables via user-friendly wizard interface.

Data Pump for PostgreSQL
Migrate from most popular databases (MySQL, SQL Server, Oracle, DB2, InterBase/Firebird, etc.) to PostgreSQL.

Data Generator for PostgreSQL
Generate test data for database testing purposes in a simple and direct way. Wide range of data generation parameters.

DB Comparer for PostgreSQL
Compare and synchronize the structure of your databases. Move changes on your development database to production with ease.

DB Extract for PostgreSQL
Create database backups in the form of SQL scripts, save your database structure and table data as a whole or partially.

SQL Query for PostgreSQL
Analyze and retrieve your data, build your queries visually, work with query plans, build charts based on retrieved data quickly and more.

Data Comparer for PostgreSQL
Compare and synchronize the contents of your databases. Automate your data migrations from development to production database.

Scroll to top

InterBase / Firebird

SQL Management Studio for InterBase/Firebird
EMS SQL Management Studio for InterBase and Firebird is a complete solution for database administration and development. SQL Studio unites the must-have tools in one powerful and easy-to-use environment that will make you more productive than ever before!

SQL Manager for InterBase/Firebird
Simplify and automate your database development process, design, explore and maintain existing databases, build compound SQL query statements, manage database user rights and manipulate data in different ways.

Data Export for InterBase/Firebird
Export your data to any of 15 most popular data formats, including MS Access, MS Excel, MS Word, PDF, HTML and more

Data Import for InterBase/Firebird
Import your data from MS Access, MS Excel and other popular formats to database tables via user-friendly wizard interface.

Data Pump for InterBase/Firebird
Migrate from most popular databases (MySQL, SQL Server, Oracle, DB2, PostgreSQL, etc.) to InterBase/Firebird.

Data Generator for InterBase/Firebird
Generate test data for database testing purposes in a simple and direct way. Wide range of data generation parameters.

© 2008 EMS Database Management Solutions

DB Comparer for InterBase/Firebird
Compare and synchronize the structure of your databases. Move changes on your development database to production with ease.

DB Extract for InterBase/Firebird
Create database backups in the form of SQL scripts, save your database structure and table data as a whole or partially.

SQL Query for InterBase/Firebird
Analyze and retrieve your data, build your queries visually, work with query plans, build charts based on retrieved data quickly and more.

Data Comparer for InterBase/Firebird
Compare and synchronize the contents of your databases. Automate your data migrations from development to production database.

Scroll to top

Oracle

SQL Management Studio for Oracle
EMS SQL Management Studio for Oracle is a complete solution for database administration and development. SQL Studio unites the must-have tools in one powerful and easy-to-use environment that will make you more productive than ever before!

SQL Manager for Oracle
Simplify and automate your database development process, design, explore and maintain existing databases, build compound SQL query statements, manage database user rights and manipulate data in different ways.

Data Export for Oracle
Export your data to any of 15 most popular data formats, including MS Access, MS Excel, MS Word, PDF, HTML and more.

Data Import for Oracle
Import your data from MS Access, MS Excel and other popular formats to database tables via user-friendly wizard interface.

Data Pump for Oracle
Migrate from most popular databases (MySQL, PostgreSQL, MySQL, DB2, InterBase/Firebird, etc.) to Oracle

Data Generator for Oracle
Generate test data for database testing purposes in a simple and direct way. Wide range of data generation parameters.

DB Comparer for Oracle
Compare and synchronize the structure of your databases. Move changes on your development database to production with ease.

DB Extract for Oracle
Create database backups in the form of SQL scripts, save your database structure and table data as a whole or partially.

SQL Query for Oracle
Analyze and retrieve your data, build your queries visually, work with query plans, build charts based on retrieved data quickly and more.

© 2008 EMS Database Management Solutions

Data Comparer for Oracle
Compare and synchronize the contents of your databases. Automate your data migrations from development to production database.

Scroll to top

DB2

SQL Management Studio for DB2
EMS SQL Management Studio for DB2 is a complete solution for database administration and development. SQL Studio unites the must-have tools in one powerful and easy-to-use environment that will make you more productive than ever before!

SQL Manager for DB2
Simplify and automate your database development process, design, explore and maintain existing databases, build compound SQL query statements, manage database user rights and manipulate data in different ways.

Data Export for DB2
Export your data to any of 15 most popular data formats, including MS Access, MS Excel, MS Word, PDF, HTML and more.

Data Import for DB2
Import your data from MS Access, MS Excel and other popular formats to database tables via user-friendly wizard interface.

Data Pump for DB2
Migrate from most popular databases (MySQL, PostgreSQL, Oracle, MySQL, InterBase/Firebird, etc.) to DB2

Data Generator for DB2
Generate test data for database testing purposes in a simple and direct way. Wide range of data generation parameters.

DB Comparer for DB2
Compare and synchronize the structure of your databases. Move changes on your development database to production with ease.

DB Extract for DB2
Create database backups in the form of SQL scripts, save your database structure and table data as a whole or partially.

SQL Query for DB2
Analyze and retrieve your data, build your queries visually, work with query plans, build charts based on retrieved data quickly and more.

Data Comparer for DB2
Compare and synchronize the contents of your databases. Automate your data migrations from development to production database.

Scroll to top

DBISAM

SQL Manager for DBISAM
Simplify and automate your database development process, design, explore and maintain existing databases, build compound SQL query statements, manage database user rights and manipulate data in different ways.

Scroll to top

© 2008 EMS Database Management Solutions

Tools & components

Advanced Data Export
Advanced Data Export Component Suite (for Borland Delphi and .NET) will allow you to save your data in the most popular office programs formats.

Advanced Data Import
Advanced Data Import™ Component Suite for Delphi® and C++ Builder® will allow you to import your data to the database from files in the most popular data formats.

Advanced PDF Generator
Advanced PDF Generator for Delphi gives you an opportunity to create PDF documents with your applications written on Delphi® or C++ Builder®.

Advanced Query Builder
Advanced Query Builder is a powerful component suite for Borland® Delphi® and C++ Builder® intended for visual building SQL statements for the SELECT, INSERT, UPDATE and DELETE clauses.

Advanced Excel Report
Advanced Excel Report for Delphi is a powerful band-oriented generator of template-based reports in MS Excel.

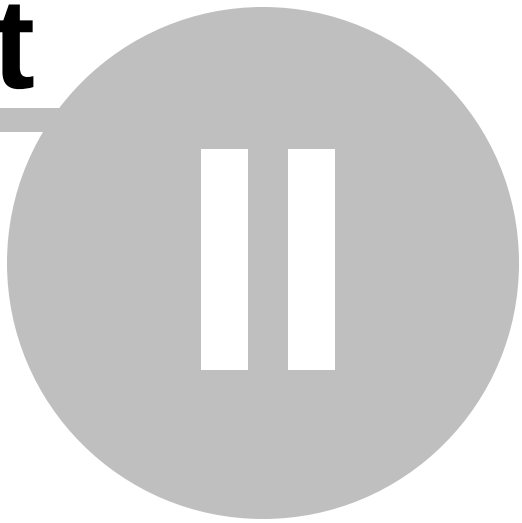
Advanced Localizer
Advanced Localizer™ is an indispensable component suite for Delphi® for adding multilingual support to your applications.

Source Rescuer
EMS Source Rescuer™ is an easy-to-use wizard application for Borland Delphi® and C++Builder® which can help you to restore your lost source code.

Scroll to top

© 2008 EMS Database Management Solutions

Part



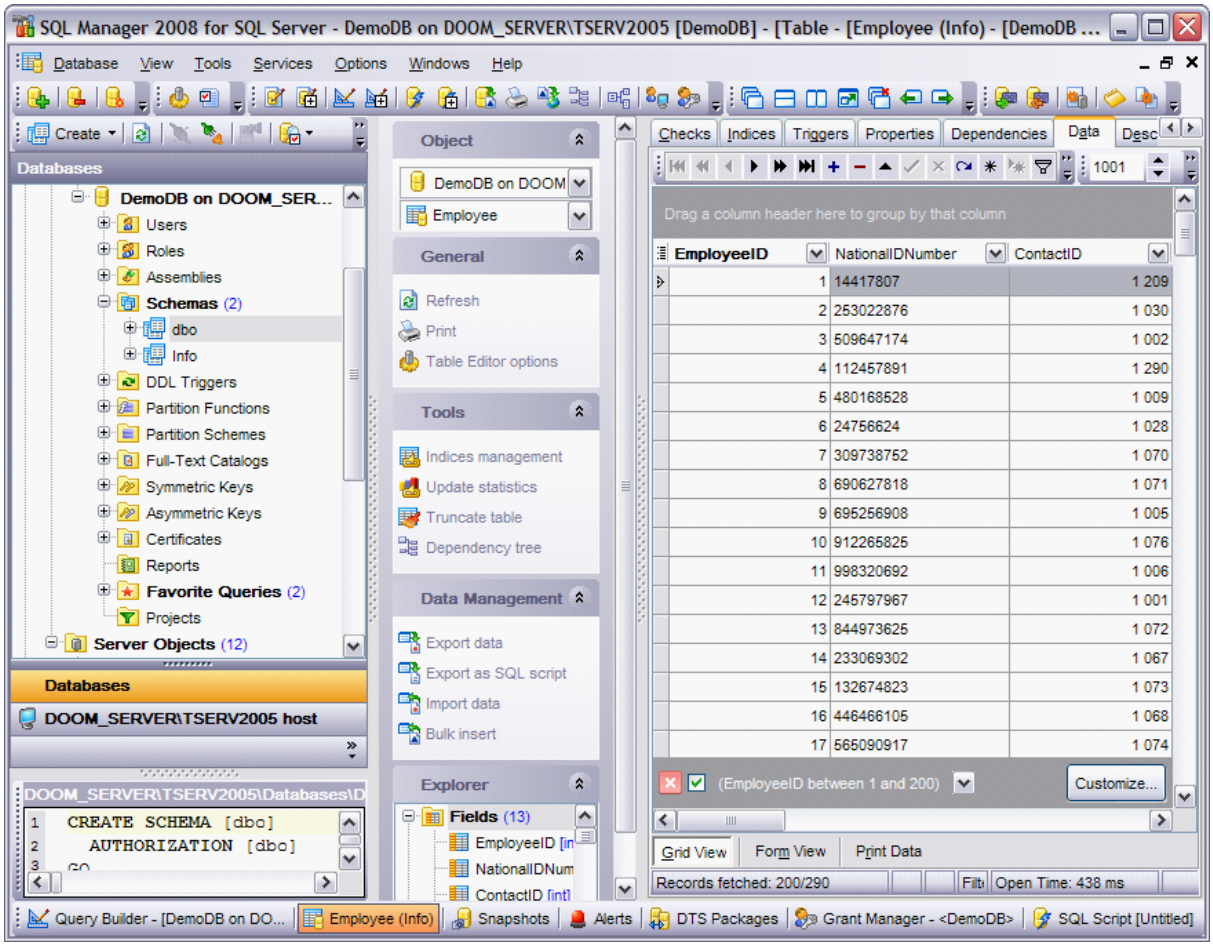
2 Getting started

SQL Manager 2008 for SQL Server provides you with an ability to contribute to efficient SQL Server administration and development using a variety of available tools easily and quickly.

The succeeding chapters of this document are intended to inform you about the tools implemented in SQL Manager 2008 for SQL Server. Please see the instructions below to learn how to perform various operations in the easiest way.

- Selecting style and language
- How the application looks when you start it for the first time
- Using Desktop Panel
- Database navigation
- Working with database objects
- Using context menus
- Switching between child windows

Enjoy your work with SQL Manager 2008 for SQL Server!



© 2008 EMS Database Management Solutions

See also:

Database Explorer
Database management
Database objects management
Query management tools
Data management
Import/Export tools
Database tools
Server tools
Personalization
External tools

© 2008 EMS Database Management Solutions

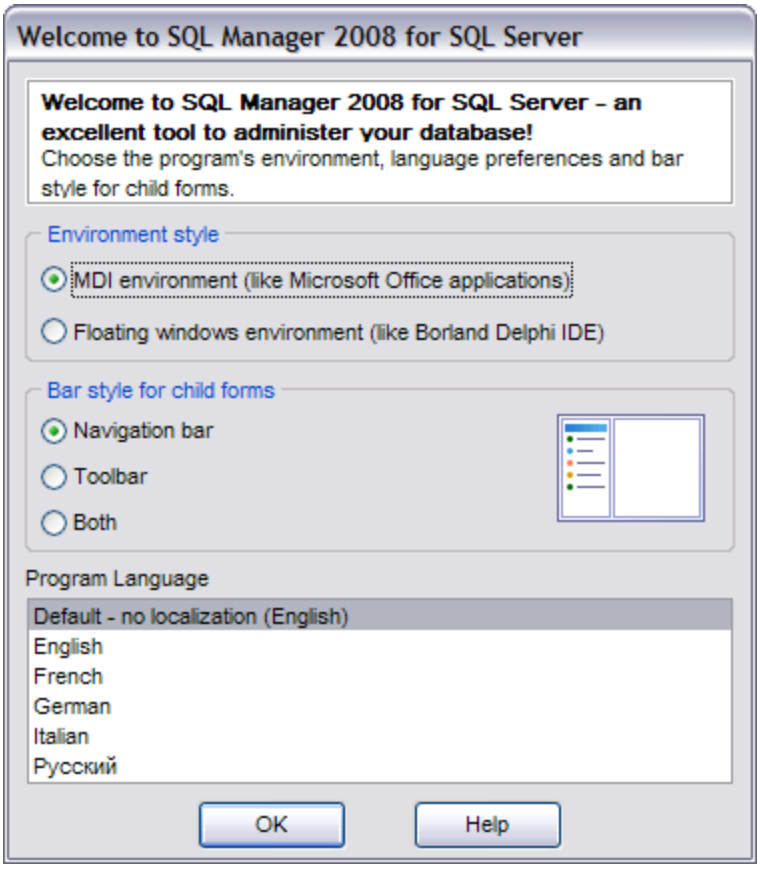
2.1 Selecting style and language

Before you start SQL Manager for the first time, you have to choose the environment style and the interface language. You can always change these settings using the **Environment Options** dialog (**Options | Environment Options...**) to configure the environment style and the Select Language dialog (**Options | Select Program Language...**) to change the program language.

Environment style
This switch allows you to define the main window behavior style - *MDI* (like in MS Office applications) or *Floating windows* (like Borland Delphi IDE).

Bar style for child forms
Here you can define the location of action buttons: within the *Navigation bar* (on the left) and/or on the *Toolbar*.

Program Language
Select the interface language from the list of available languages.



See also:
First time started
Using Desktop Panel

© 2008 EMS Database Management Solutions

Database navigation
Working with database objects
Using context menus
Switching windows

© 2008 EMS Database Management Solutions

2.2 First time started

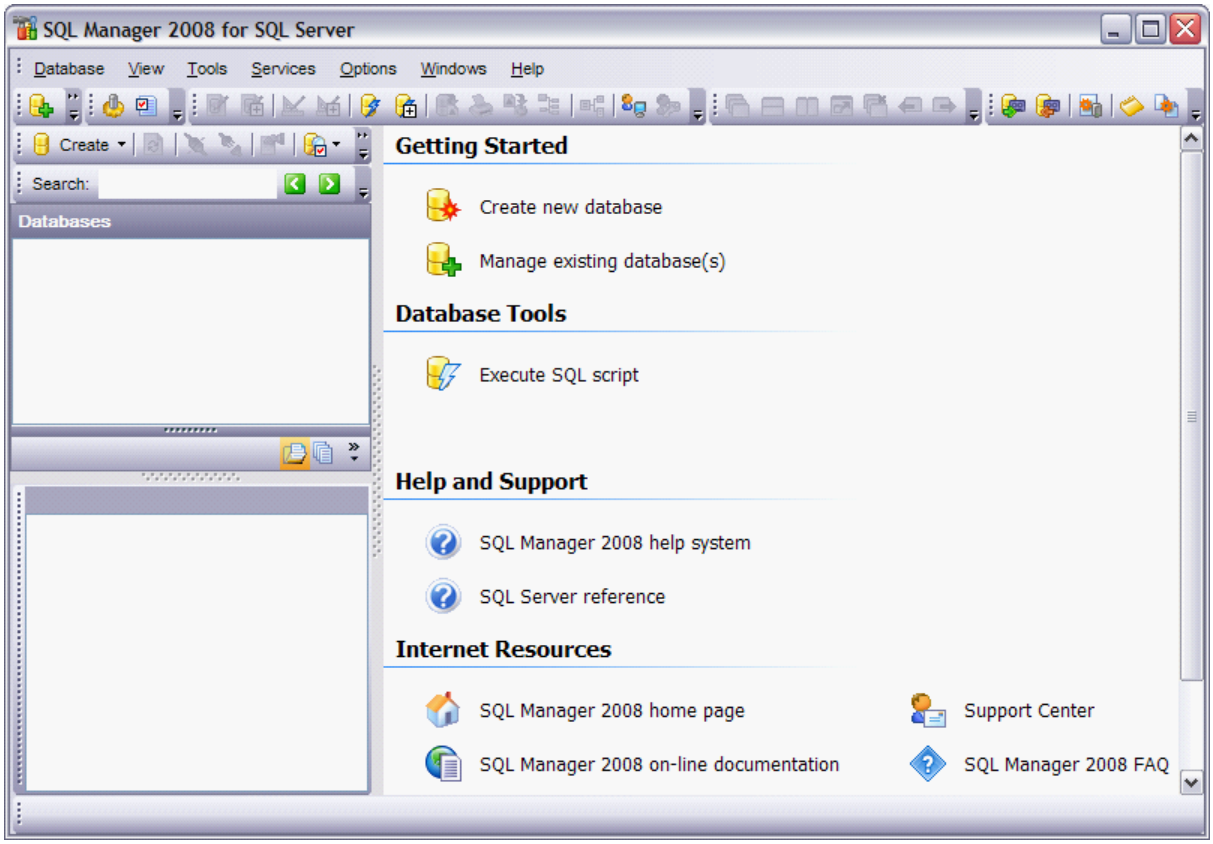
This is how **SQL Manager 2008 for SQL Server** looks when you start it for the first time. Use the Desktop panel to fulfill any of common tasks: Create a new database, Manage existing database(s), and several tasks which do not require database registration, i.e. Execute SQL script, accessing the **reference system** or using access available **Internet resources**.

The main menu allows you to perform various **Database** operations, open To-Do List and activate/deactivate Database Explorer, SQL Assistant and various toolbars within the **View** menu, manage your databases using items of the **Tools** and **Services** menus, customize the application using the **Options** menu, manage SQL Manager **Windows** using Window List and other tools, view the Tip of the Day and access Registration information and product documentation using the corresponding items available within the **Help** menu.

To start working with your SQL Server, you should first register the host using Register Host wizard. After that you need to register one or several databases using Register Database Wizard.

By default the corresponding **Register Host**, **Register Database** buttons are available on the toolbar and within the **Database** menu.

When the database connection settings are specified, you can set connection to your database and proceed to Database navigation, Database Objects management, working with SQL queries and other tools of SQL Manager.



© 2008 EMS Database Management Solutions

See also:

Selecting style and language
Using Desktop Panel
Database navigation
Working with database objects
Using context menus
Switching windows

© 2008 EMS Database Management Solutions

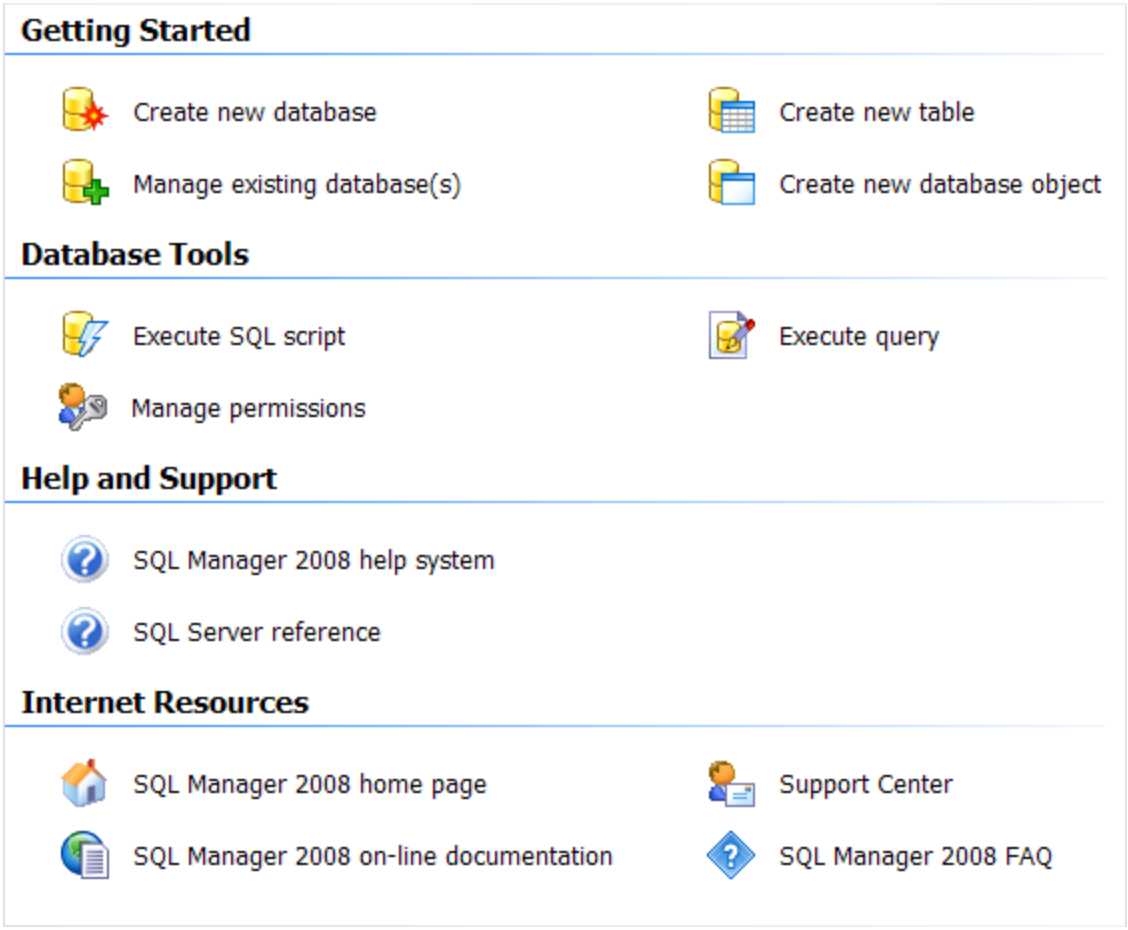
2.3 Using Desktop Panel

Desktop Panel is the area that is visible when no child windows are open in SQL Manager 2008 for SQL Server. The working area of **Desktop Panel** is divided into four sections: *Getting Started*, *Database Tools*, *Help and Support*, *Internet Resources*.

Using the **Desktop Panel** items you can:

- create a new SQL Server database;
- register existing database(s) to operate them afterwards in SQL Manager;
- create a new table within the current database using the New Table window (this item is available if there is at least one active database connection);
- create a new database object within the current database (this item is available if there is at least one active database connection);
- execute a script using SQL Script Editor;
- execute a SQL query (this item is available if there is at least one active database connection);
- grant permissions on database objects to SQL Server users using Grant Manager (this item is available if there is at least one active database connection);
- show this help file;
- use SQL Server reference;
- browse SQL Manager 2008 on-line documentation;
- visit SQL Manager 2008 Home Page;
- go to Technical Support Center;
- look through the Frequently Asked Questions page.

© 2008 EMS Database Management Solutions



See also:

Selecting style and language
First time started
Database navigation
Working with database objects
Using context menus
Switching windows

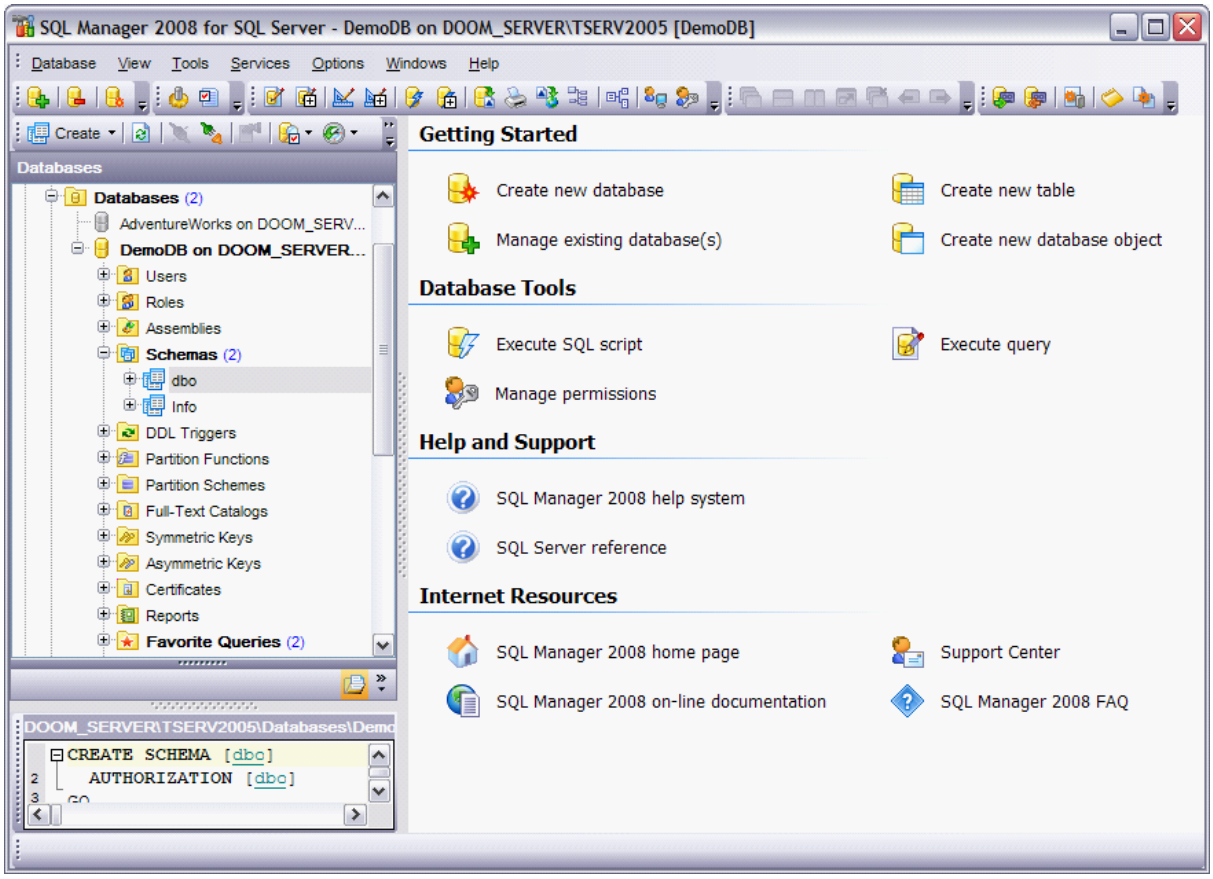
© 2008 EMS Database Management Solutions

2.4 Database navigation

After you have registered the required database(s), the corresponding alias(es) appear in the DB Explorer tree on the left. If the **Show Hosts** option is checked on the Environment Options | DB Explorer page, the host nodes are also displayed in the tree (alternatively, you can use the **Show Hosts** item of the Database context menu, or the drop-down menu of the **View Mode** toolbar button for the same purpose). If necessary, you can also specify that empty schemas should be hidden in the tree: use the corresponding option available on the DB Explorer page of the Environment Options dialog.

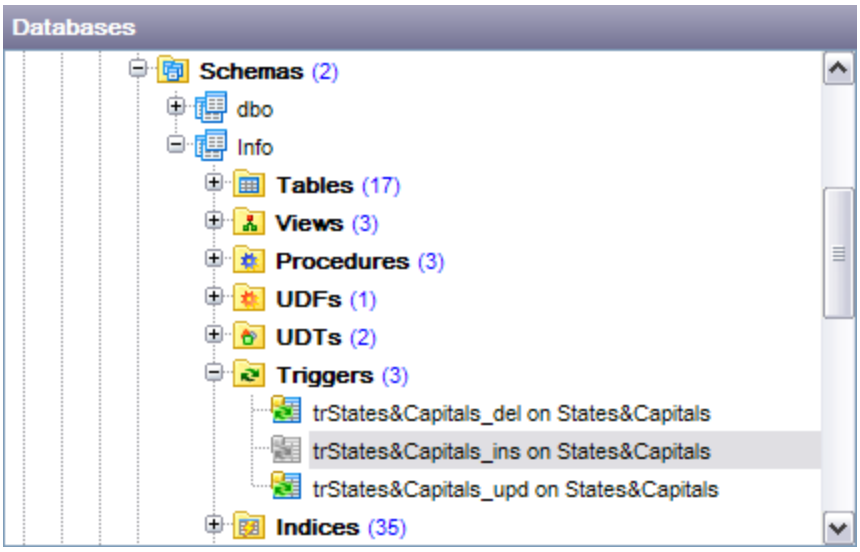
To connect to a database, simply double-click its alias (or select the database alias in DB Explorer and press **Enter**). If the connection is successful, the database node expands into a tree of objects. To select the types of objects to be expanded upon successful database connection, you can use the **Expand on connection** group of the Environment Options | DB Explorer page.

Now you can navigate within the database objects. Use SQL Assistant to get extended information about the currently selected object.



For your convenience objects having different status (e.g. enabled/disabled) are displayed with different icons in Database Explorer.

© 2008 EMS Database Management Solutions



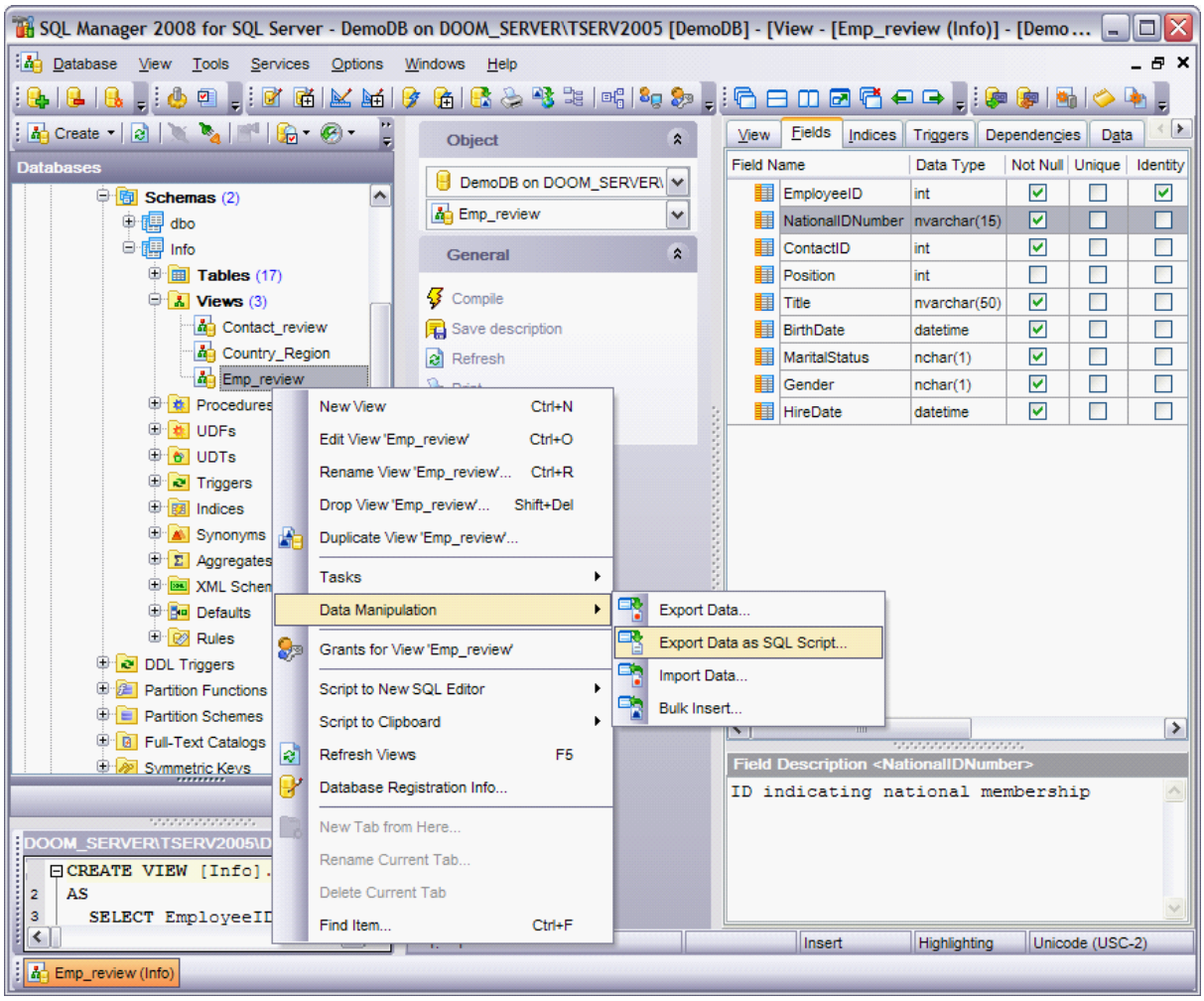
See also:

Selecting style and language
First time started
Using Desktop Panel
Working with database objects
Using context menus
Switching windows

© 2008 EMS Database Management Solutions

2.5 Working with database objects

The nodes of the DB Explorer tree allow you to access server objects and objects of the selected database respectively. If SQL Server you are connected to supports certain types of objects, their nodes will appear in the tree. Double-click an object to open it in the corresponding editor.
Right-click an object to display its context menu which allows you to perform various operations over the selected object or database.



If you want to use the DB Explorer tree for working with **table subobjects** (fields, indexes, Foreign keys, etc.), check the **Show table subobjects** option which is available within the **General options** group of the Environment Options | DB Explorer page (you can also use the **Show Table Subobjects** menu item in the drop-down menu of the **View Mode** toolbar button for the same purpose.)

See also:

Selecting style and language

© 2008 EMS Database Management Solutions

First time started
Using Desktop Panel
Database navigation
Using context menus
Switching windows

© 2008 EMS Database Management Solutions

2.6 Using context menus

The **context menus** are aimed at facilitating your work with SQL Manager 2008 for SQL Server: you can perform a variety of operations using context menu items.

Select an object in DB Explorer and right-click its alias to open the context menu.

- Host context menu
- Database context menu
- Object context menu

See also:

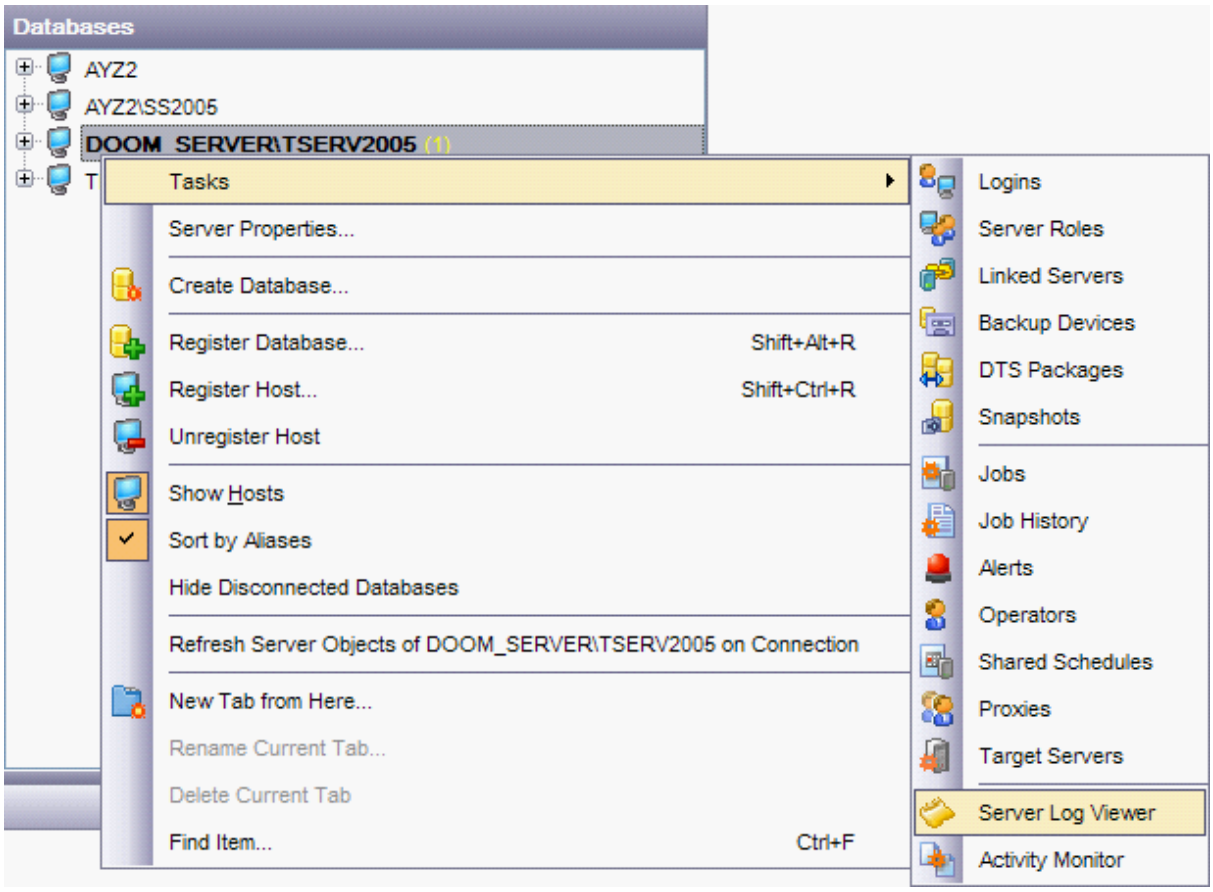
Selecting style and language
First time started
Using Desktop Panel
Database navigation
Working with database objects
Switching windows

2.6.1 Host context menu

The **context menu of a Host** in the DB Explorer tree allows you to:

- access server objects and common services available in the **Tasks** submenu;
- view/edit the selected host properties within the Server Properties dialog;
- create a new database at the selected host;
- register a new database using Register Database Wizard;
- register a new host using Register Host Wizard;
- unregister the selected Host;
- configure representation of hosts and databases in Database Explorer;
- create a new tab for the selected Host to access it through this tab quickly and/or manage the existing tab;
- search for an object within the tree.

© 2008 EMS Database Management Solutions



See also:

Database context menu

Object context menu

2.6.2 Database context menu

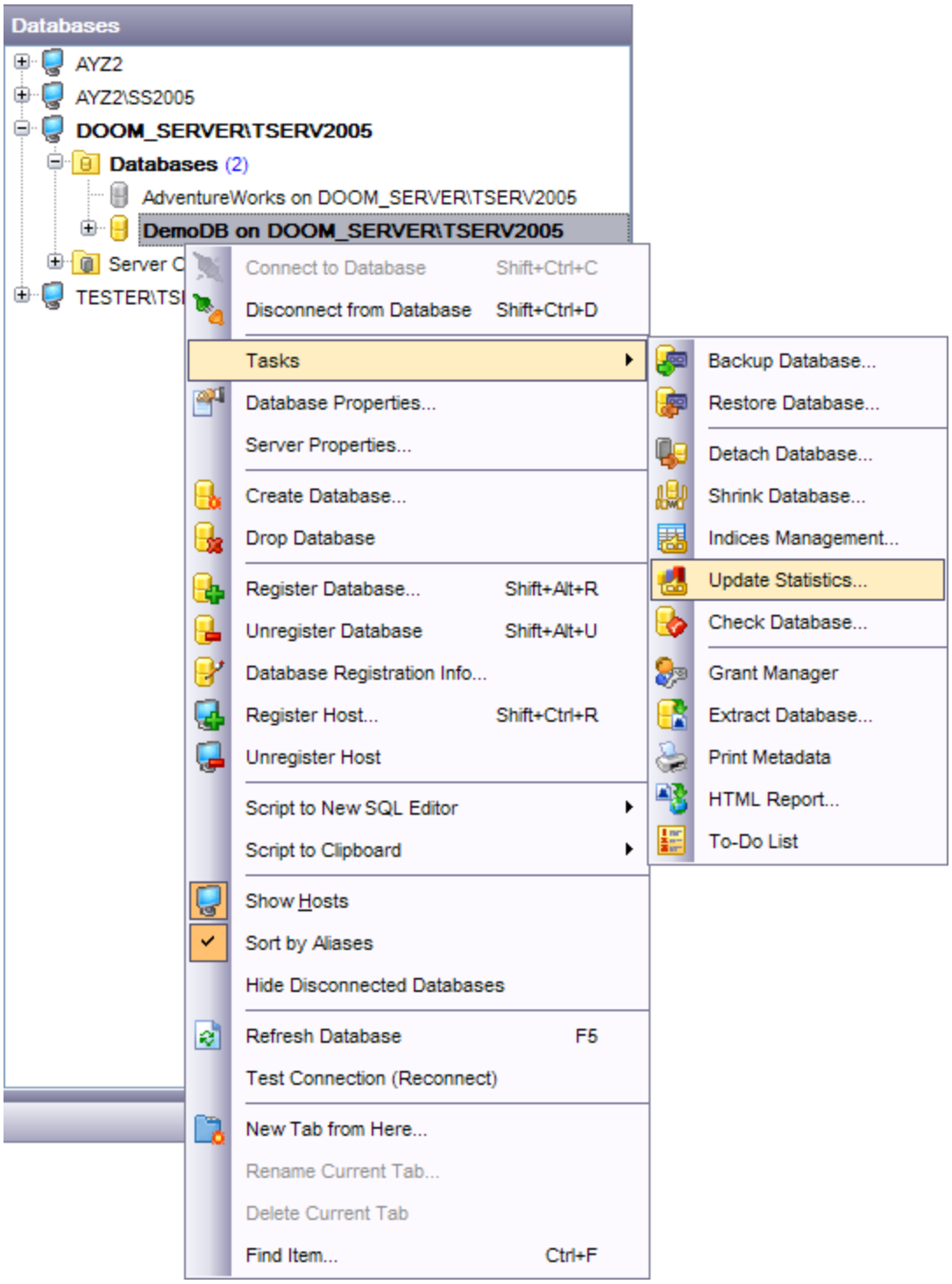
The **context menu of a registered database** in the DB Explorer tree allows you to:

- connect to the selected database (if connection to the database is not active yet);
- disconnect from the selected database (if connection to the database has been already activated);
- access database tools available in the **Tasks** submenu;
- view/edit the Database properties;
- view/edit the host properties within the Server Properties dialog;
- create a new database at the host where the selected database resides;
- drop the selected database;
- register a new database using Register Database Wizard;
- unregister the selected database;
- view/edit the selected database registration information within the Database Registration Info dialog;
- register a new host using Register Host Wizard;
- unregister the Host where the selected database resides;
- generate the database script and open it in SQL Editor;

© 2008 EMS Database Management Solutions

- generate the database script and copy its text to Windows clipboard;
- configure representation of hosts and databases in Database Explorer;
- refresh the selected database;
- test connection to the database (or reconnect if connection has been already established);
- create a new tab for the selected database to access it through this tab quickly and/or manage the existing tab;
- search for an object within the tree.

© 2008 EMS Database Management Solutions



See also:

Host context menu

Object context menu

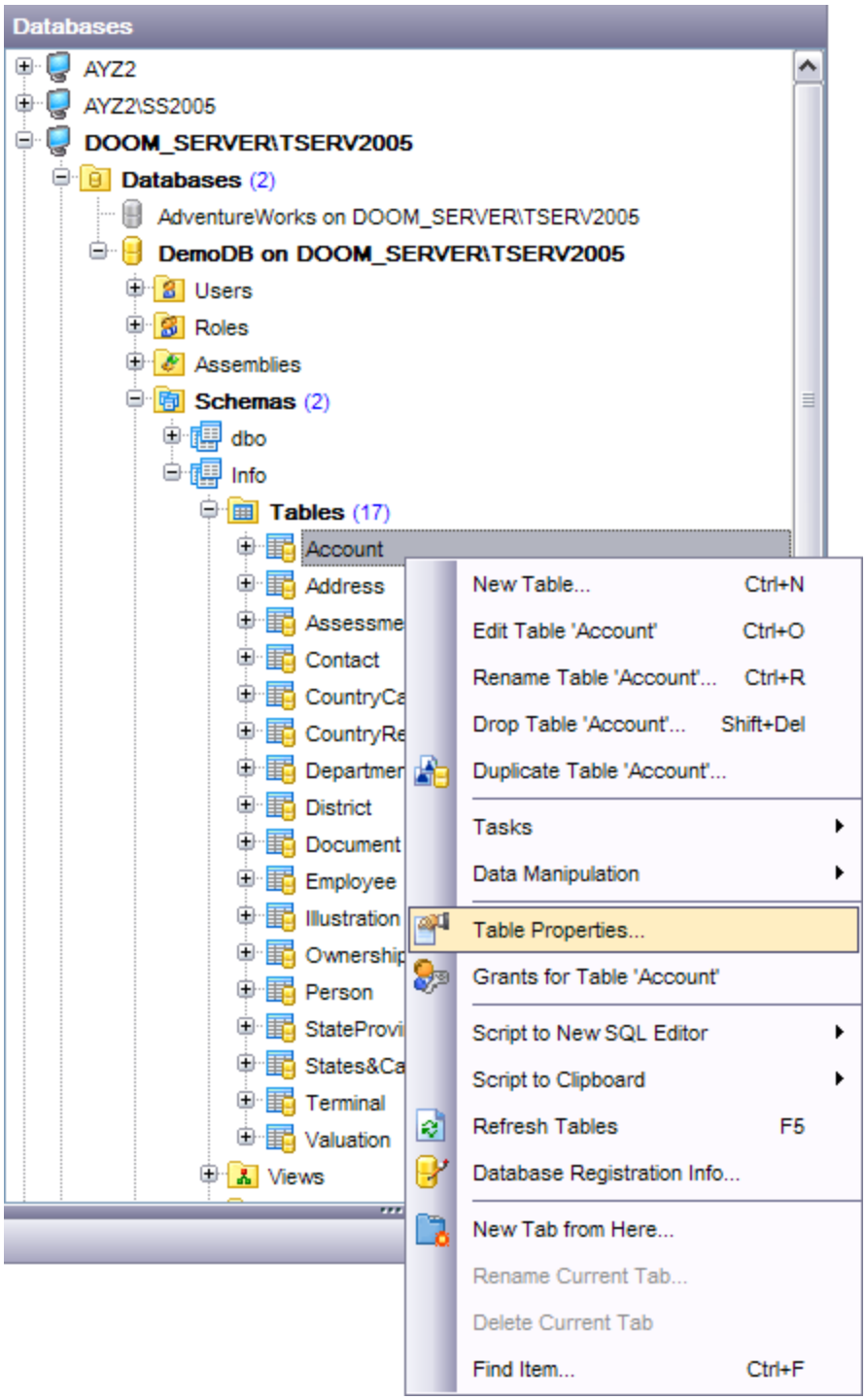
© 2008 EMS Database Management Solutions

2.6.3 Object context menu

The **context menu of an object** (e.g. *table*) in the DB Explorer tree allows you to:

- create a new database object of the same type;
- edit the selected object in its editor;
- rename the selected object;
- drop the selected object from the database;
- duplicate the selected object (create a new object with the same DDL structure and properties as the selected object has);
- access common **Tasks** applied to this object;
- perform data manipulation operations (for tables and views);
- view properties (for tables);
- define grants for the selected object;
- generate the object script and open it in SQL Editor;
- generate the object script and copy its text to Windows clipboard;
- refresh all objects of the selected object type;
- view/edit the database registration information within the Database Registration Info dialog (for database objects);
- create a new tab for the selected object to access it through this tab quickly and/or manage the existing tab;
- search for an object within the tree.

© 2008 EMS Database Management Solutions



See also:

Host context menu

Database context menu

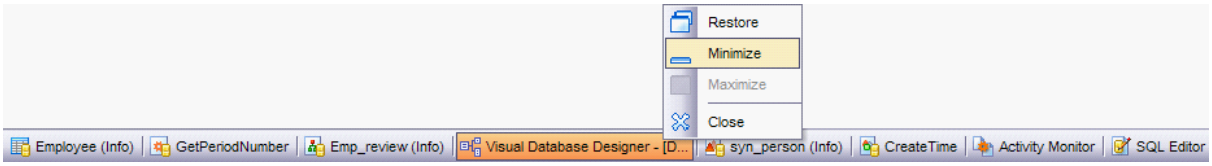
© 2008 EMS Database Management Solutions

© 2008 EMS Database Management Solutions

2.7 Switching windows

The **Windows Toolbar** of SQL Manager allows you to switch between child windows easily, like in Windows Task Bar.

To activate the window you need, simply click one of the window buttons. To perform some additional actions with the window, right-click its tab and select the corresponding menu item from the popup menu.



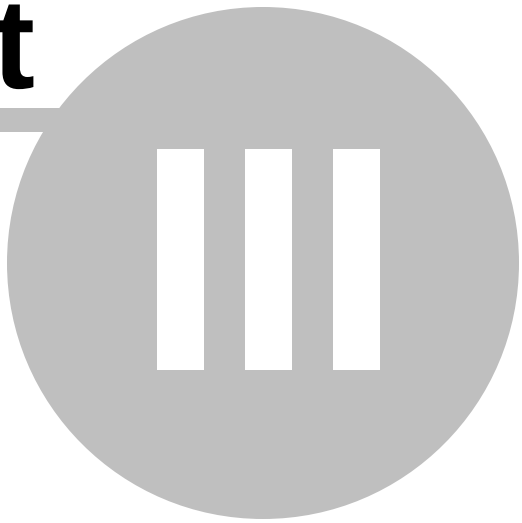
If you have multiple windows opened, you can also switch between them using the *Ctrl+Tab* shortcut.

See also:

Selecting style and language
First time started
Using Desktop Panel
Database navigation
Working with database objects
Using context menus

© 2008 EMS Database Management Solutions

Part



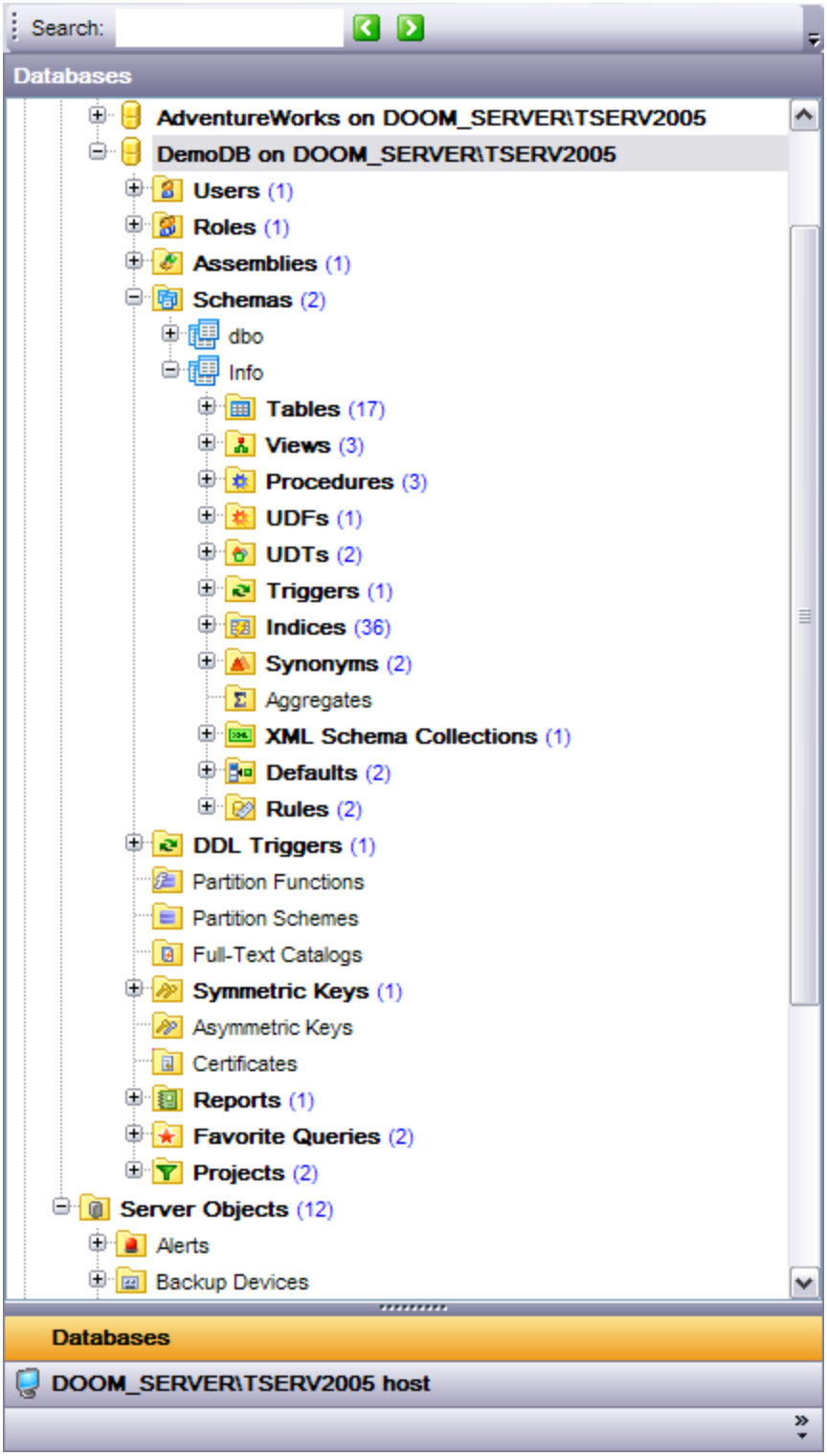
3 Database Explorer

Database Explorer (or **DB Explorer**) is the basic window of SQL Manager 2008 for SQL Server for navigation within databases and working with database and server objects. The tree-like structure of DB Explorer allows you to manage the databases, database and server objects and perform other everyday operations quickly and easily.
You can also reorder hosts order by using drag-and-drop.

The following list contains the most frequently used features provided by Database Explorer.

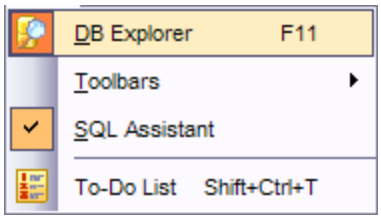
- Managing database registration info
- Connecting to databases
- Performing basic operations upon database objects
- Selecting multiple objects
- Navigating database objects using multiple tabs
- Easy access to recently opened objects
- Creating projects to work with selected objects only
- Searching within the tree
- Viewing extended information about database objects
- Configuring Database Explorer
- Managing Favorite queries

© 2008 EMS Database Management Solutions



To view/hide the Database Explorer window, use the **View | DB Explorer** main menu item or press the **F11** key.

© 2008 EMS Database Management Solutions



Use the *Ctrl+Shift+C* shortcut to collapse current **DB Explorer** tree branch and switch to the parent node.

Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**Yes**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

Getting started
Database management
Database objects management
Query management tools
Data management
Import/Export tools
Database tools
Server tools
Personalization
External tools

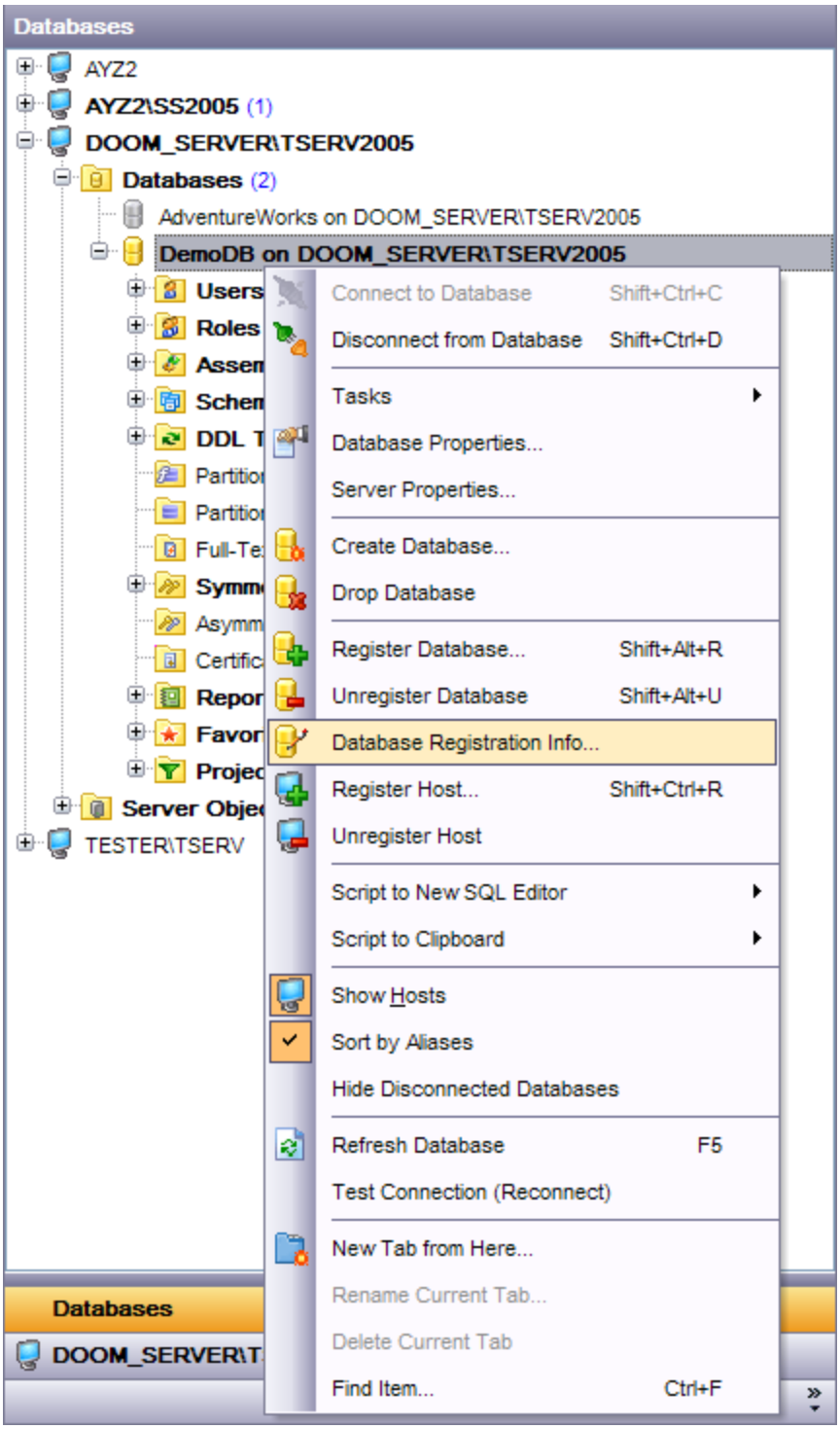
© 2008 EMS Database Management Solutions

3.1 Managing database registration info

After you have created and/or registered your database in SQL Manager 2008 for SQL Server, you can perform a number of operations with the database using the context menu.

If you need to view/edit the registration information of a database, right-click the database alias in DB Explorer and select the **Database Registration Info...** context menu item to open the Database Registration Info dialog.

© 2008 EMS Database Management Solutions



See also:

Connecting to databases

Operations with database objects

© 2008 EMS Database Management Solutions

Selecting multiple objects
Using tabs for database navigation
Recently opened objects
Creating projects
Searching within the tree
SQL Assistant
Configuring Database Explorer
Managing Favorite queries

© 2008 EMS Database Management Solutions

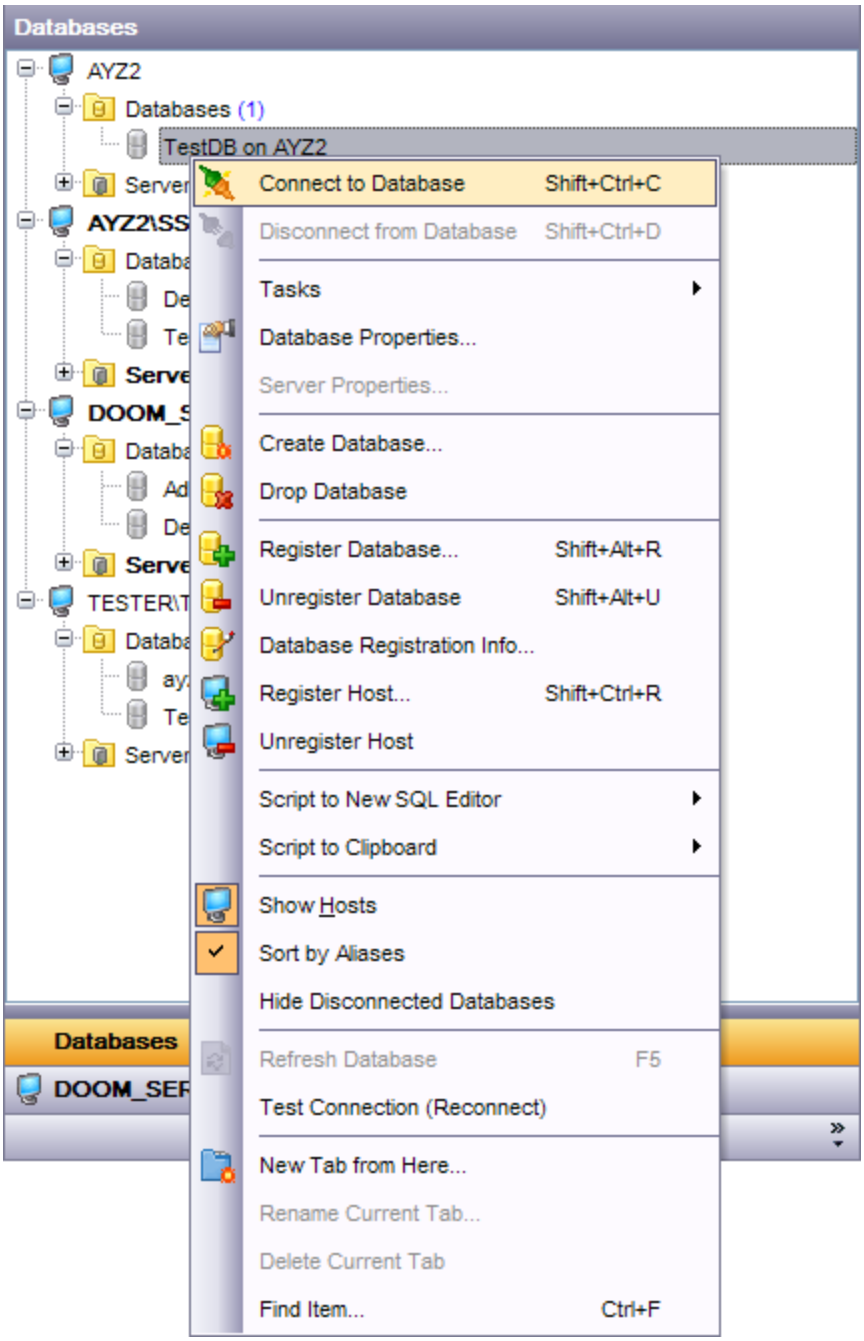
3.2 Connecting to databases

When the SQL Server host is registered, and the database registration is complete, you can establish connection to your database.

The simplest way to connect to a database is to double-click its alias in the Database Explorer tree. The same operation can be performed by selecting the **Connect to Database** item of the database alias context menu, or by using the **Database | Connect to Database** main menu item.

Alternatively, you can use the *Shift+Ctrl+C* shortcut to connect to your database.

© 2008 EMS Database Management Solutions



Note that you can change database aliases order by dragging within the **DB Explorer** tree.

See also:

Managing database registration info

Operations with database objects

© 2008 EMS Database Management Solutions

Selecting multiple objects
Using tabs for database navigation
Recently opened objects
Creating projects
Searching within the tree
SQL Assistant
Configuring Database Explorer
Managing Favorite queries

© 2008 EMS Database Management Solutions

3.3 Operations with database objects

Database Explorer allows you to perform various operations with database objects and server objects.

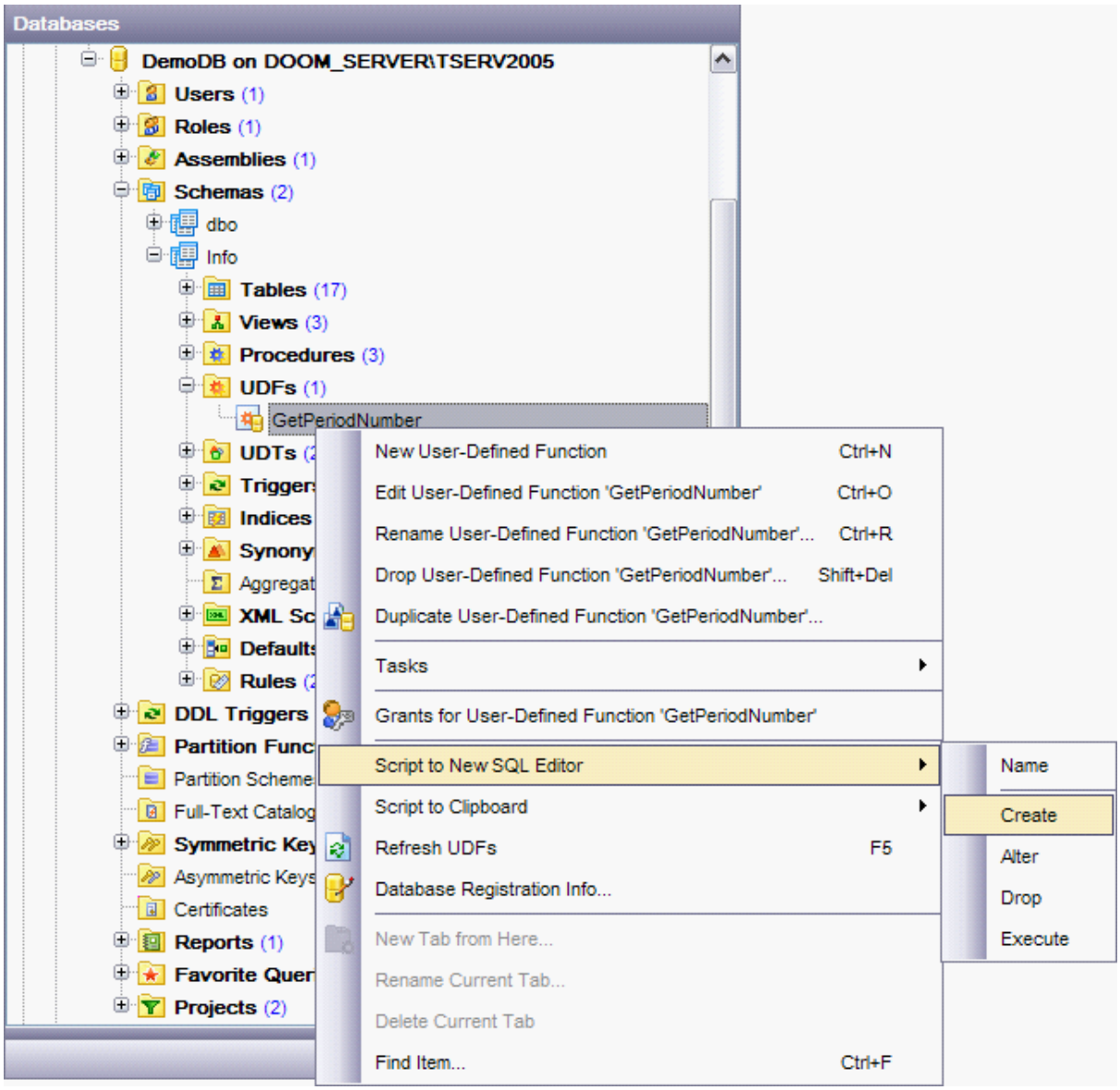
To open an object in its editor, you can double-click the object in the **DB Explorer** tree.

You can also right-click an object within the **DB Explorer** tree and use its context menu of to perform a number of operations:

- create a new object (the **New <object>...** item);
- edit currently selected object (the **Edit <object_name>...** item);
- rename currently selected object (the **Rename <object_name>...** item);
- drop the selected object from the database (the **Drop <object_name>...** item);
- duplicate the selected object (the **Duplicate <object_name>...** item);
- define grants for the selected object (the **Grants for <object_name>...** item).

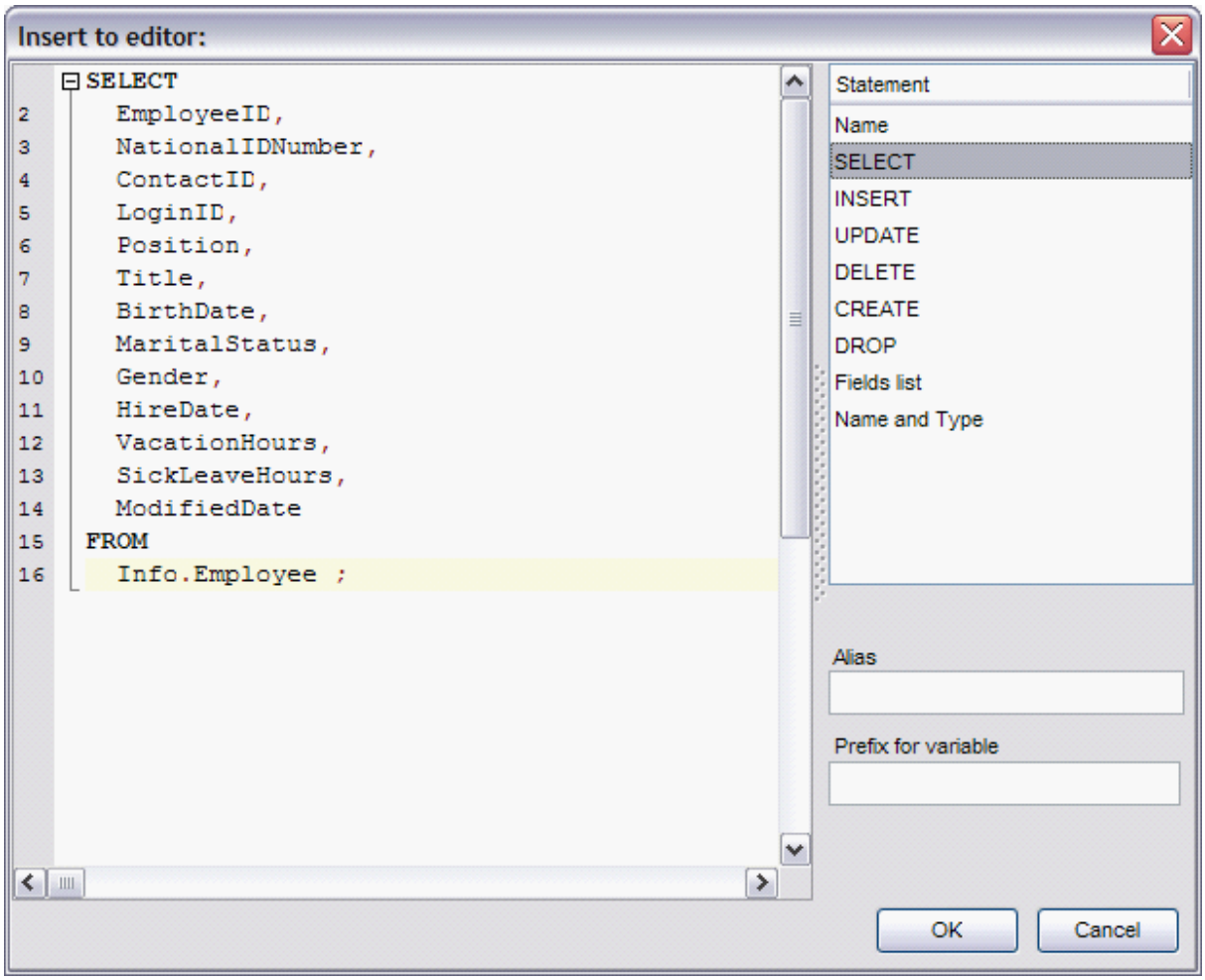
Note that the context menu contains object-specific items only when the object is currently selected in **DB Explorer**.

© 2008 EMS Database Management Solutions



Using drag-and-drop operations you can add objects to SQL Editor, Visual Query Builder or SQL Script Editor. For your convenience the **Insert to editor** dialog is implemented. The dialog allows you to specify the statement to be inserted into the editor. If necessary, you can edit the generated statement manually.

© 2008 EMS Database Management Solutions



See also:

Managing database registration info
Connecting to databases
Selecting multiple objects
Using tabs for database navigation
Recently opened objects
Creating projects
Searching within the tree
SQL Assistant
Configuring Database Explorer
Managing Favorite queries

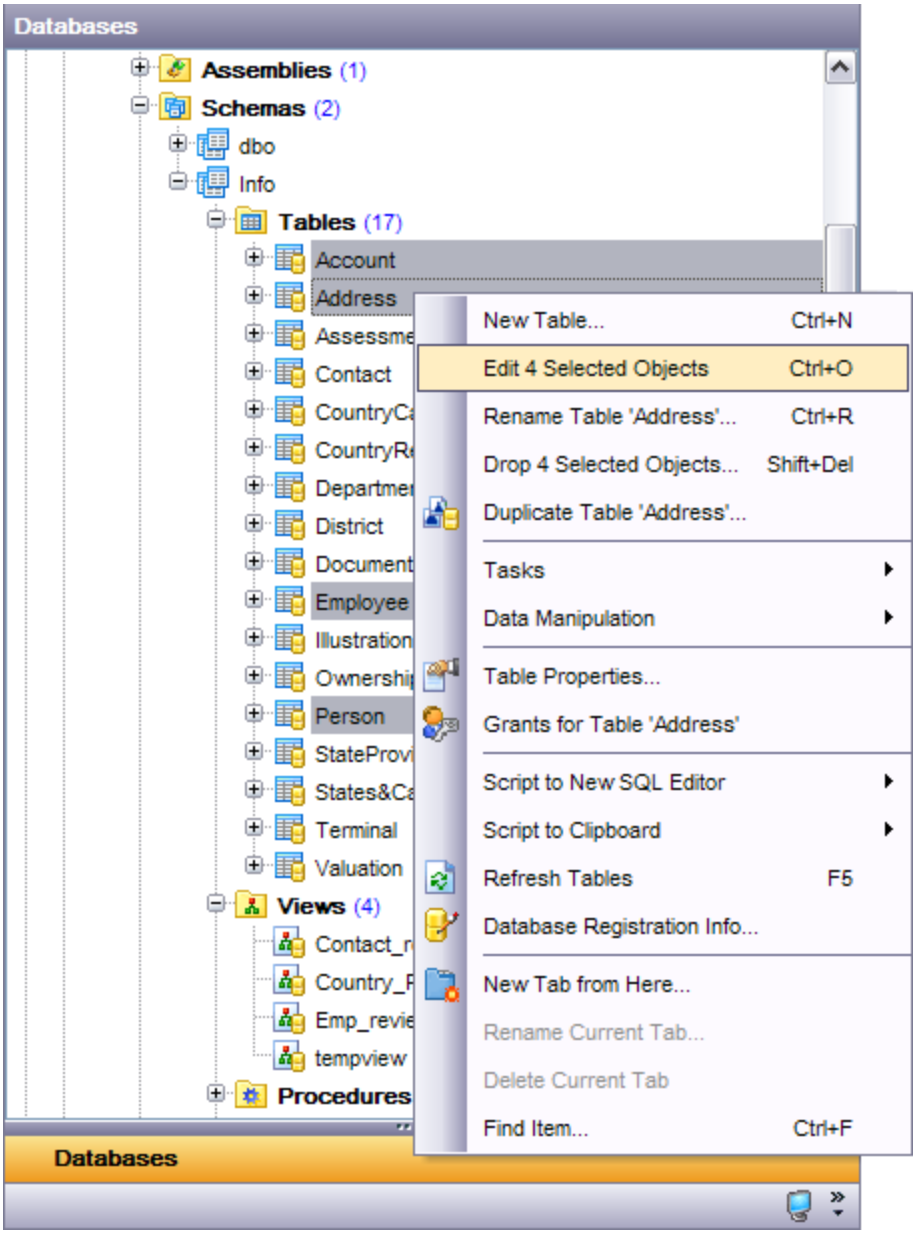
© 2008 EMS Database Management Solutions

3.4 Selecting multiple objects

You can select more than one object in **Database Explorer** by pressing the *Ctrl* or the *Shift* key and selecting multiple objects one by one.

The **context menu** of several selected objects allows you to:

- create a new database object of the same type;
- edit the selected objects (for database objects);
- drop the selected objects;
- perform other operations with the first of the selected objects (see Operations with database objects).



© 2008 EMS Database Management Solutions

See also:

Managing database registration info
Connecting to databases
Operations with database objects
Using tabs for database navigation
Recently opened objects
Creating projects
Searching within the tree
SQL Assistant
Configuring Database Explorer
Managing Favorite queries

© 2008 EMS Database Management Solutions

3.5 Using tabs for database navigation

To make your work with **Database Explorer** even more convenient, the capability of **working with several tabs** is implemented.

You can use tabs when you wish to work with a particular node of the DB Explorer tree only: with one specific schema, or with tables of some schema, or with a specific database project. Creating such tabs will minimize scrolling within large trees, you only need to switch between them with a single click on the corresponding tab.

Creating tabs

In order to create a new tab:

- right-click the node for which you wish to create a tab and select the **New Tab from Here...** context menu item.

Renaming tabs

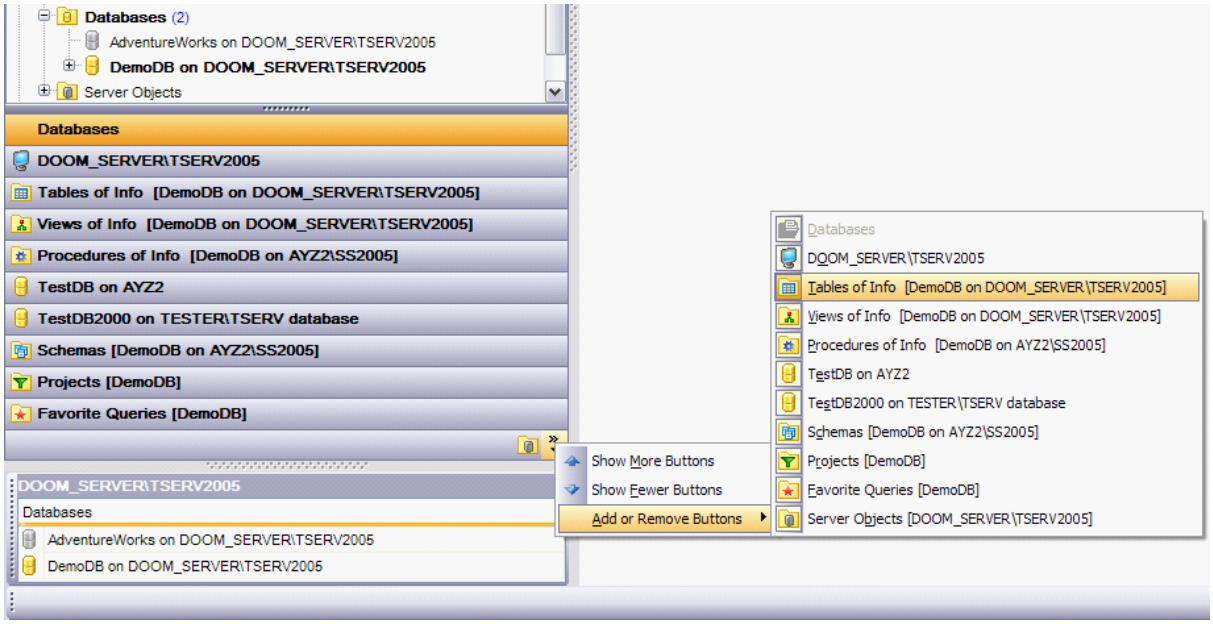
In order to rename a tab:

- switch to the tab by clicking its caption (there can be only one active tab, and it is differentiated by a different color);
- right-click within the DB Explorer area and select the **Rename Current Tab...** context menu item.

Removing tabs

In order to remove a tab:

- switch to the tab by clicking its caption (there can be only one active tab, and it is differentiated by a different color);
- right-click within the DB Explorer area and select the **Delete Current Tab** context menu item.



© 2008 EMS Database Management Solutions

Note: Navigation through the tabs is also possible with the help of the following shortcuts :

- *Ctrl+Shift+N* - move to the next tab;
- *Ctrl+Shift+P* - move to the previous tab.

See also:

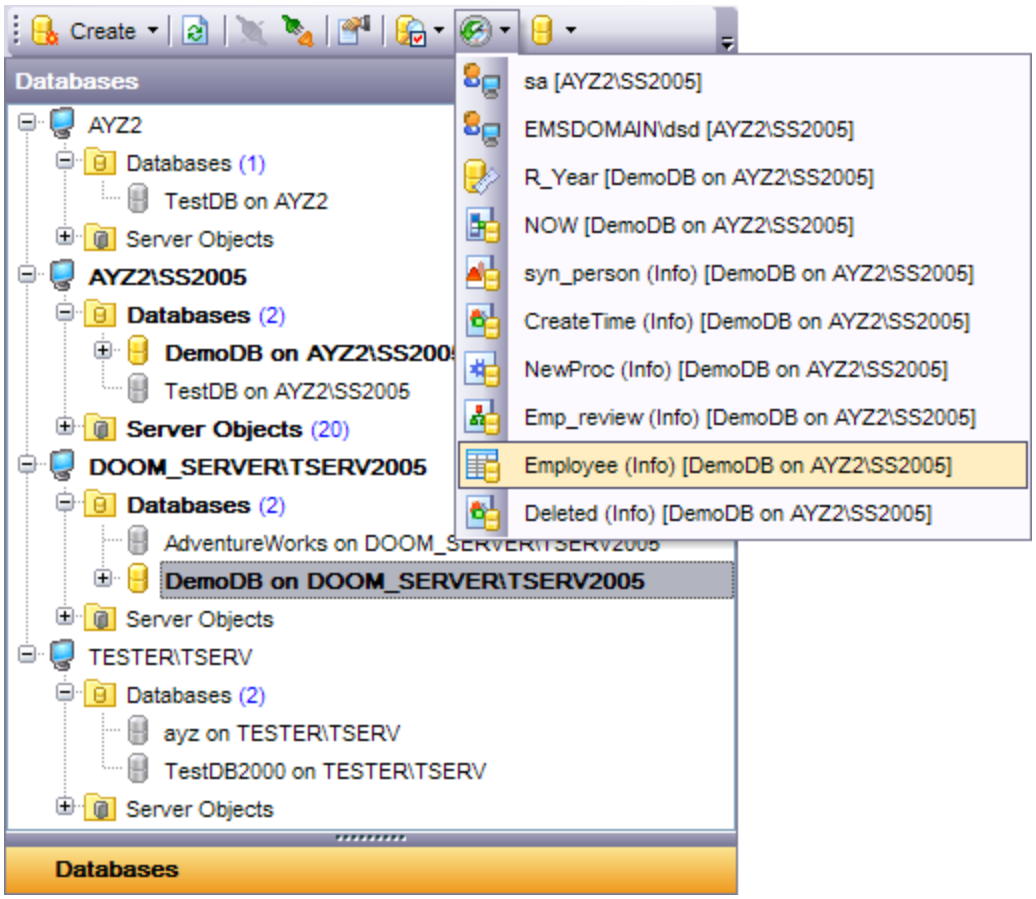
Managing database registration info
Connecting to databases
Operations with database objects
Selecting multiple objects
Recently opened objects
Creating projects
Searching within the tree
SQL Assistant
Configuring Database Explorer
Managing Favorite queries

© 2008 EMS Database Management Solutions

3.6 Recently opened objects

Use the **Recent Objects** button available on the DB Explorer toolbar to access the list of recently opened database objects (during the current session).

This list is common for all registered databases. Next to the object name the database name and the host are displayed. Select an object from this list to open it using its editor.



To change the number of objects that are considered 'recent', select the **Options | Environment Options** main menu item, proceed to the **Tools | DB Explorer** section within the **Environment Options** dialog, and set the **Recent objects count** option value (see Environment Options for details).

See also:

Managing database registration info
Connecting to databases
Operations with database objects
Selecting multiple objects
Using tabs for database navigation

© 2008 EMS Database Management Solutions

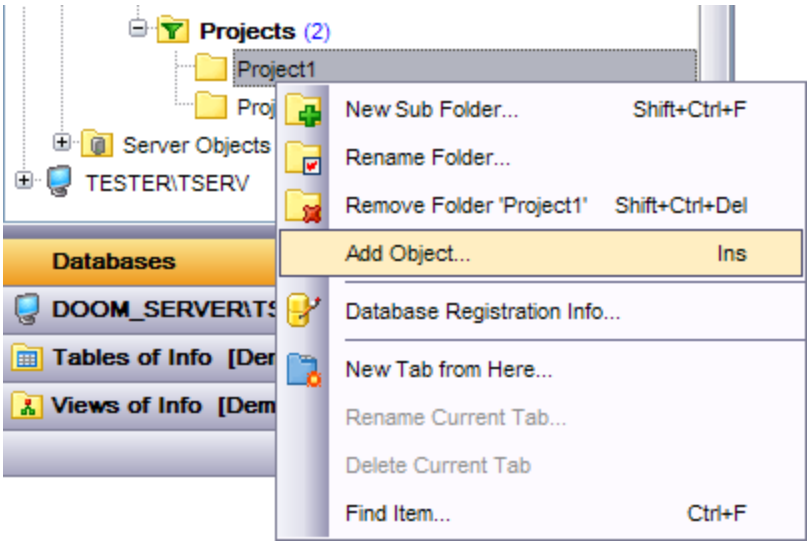
Creating projects
Searching within the tree
SQL Assistant
Configuring Database Explorer
Managing Favorite queries

© 2008 EMS Database Management Solutions

3.7 Creating projects

Use the **Projects** node for each database to work with the selected objects of this database only. You can place any object from the database tree here.

You can also create a separate tab for your project. See Using tabs for database navigation section for details.



Creating project folders

In order to create a new folder:

- create a folder (if necessary, you can create subfolders inside the project folder) by right-clicking the **Projects** node and selecting the **New Sub Folder...** context menu item;
- enter the folder name within the **New Folder** dialog.

Adding objects

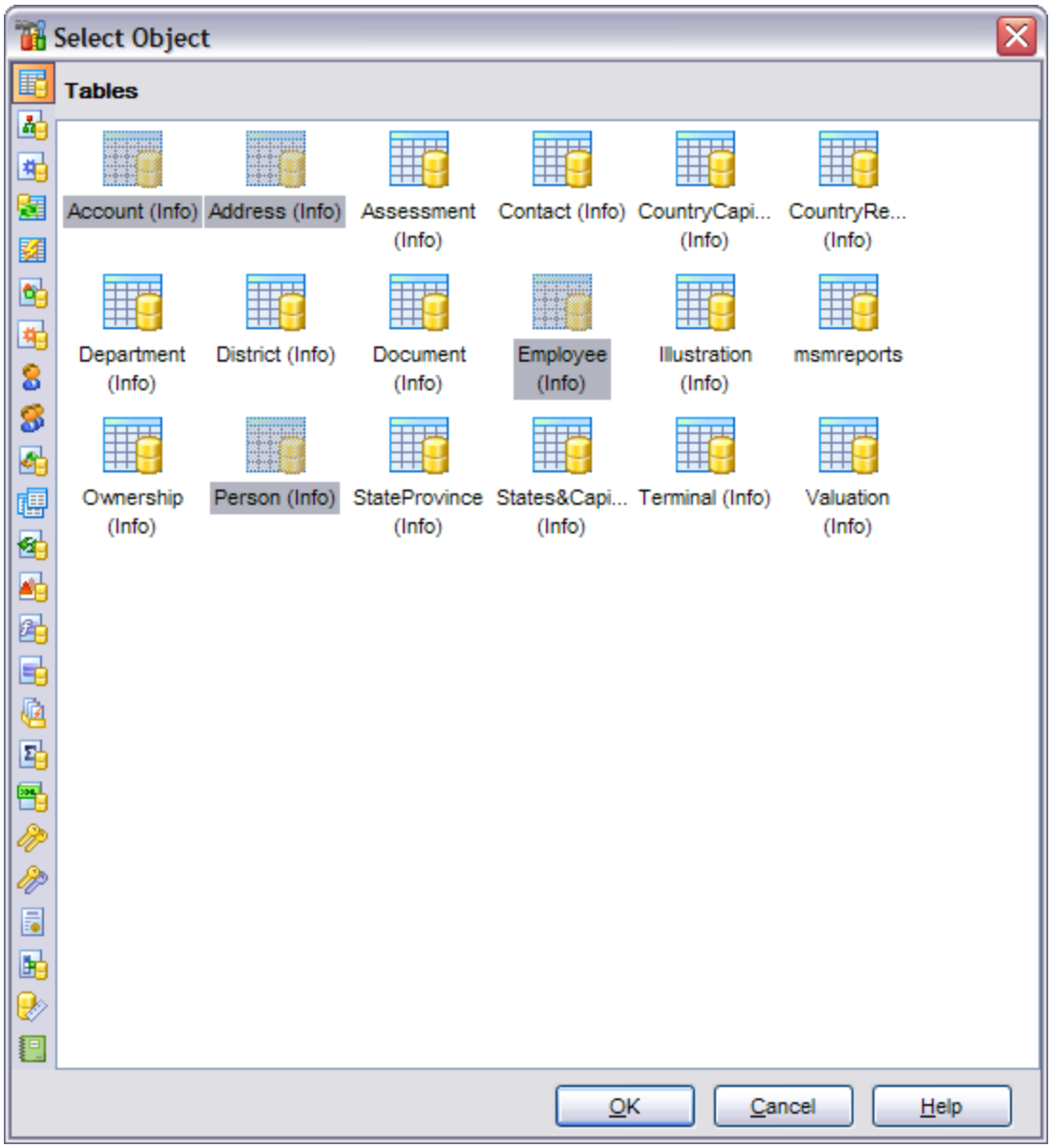
In order to add a new object to the project folder:

- extent the **Projects** node in DB Explorer;
- drag an object (or multiple objects) from the database tree to the project folder

or

- right-click the project folder and select the **Add Object...** context menu item, or use the *Ins* key;
- use the **Select Object** dialog to specify objects to be added to the project folder.

© 2008 EMS Database Management Solutions



Removing objects from the project

In order to remove an object from the project:

- right-click the object and select the **Remove <object_name> from Project** context menu item, or use the *Shift+Ctrl+Del* shortcut;
- confirm removing in the dialog window.

Note: This operation does not drop the object from the database, but only removes its alias from the project tree.

See also:

© 2008 EMS Database Management Solutions

Managing database registration info
Connecting to databases
Operations with database objects
Selecting multiple objects
Using tabs for database navigation
Recently opened objects
Searching within the tree
SQL Assistant
Configuring Database Explorer
Managing Favorite queries

© 2008 EMS Database Management Solutions

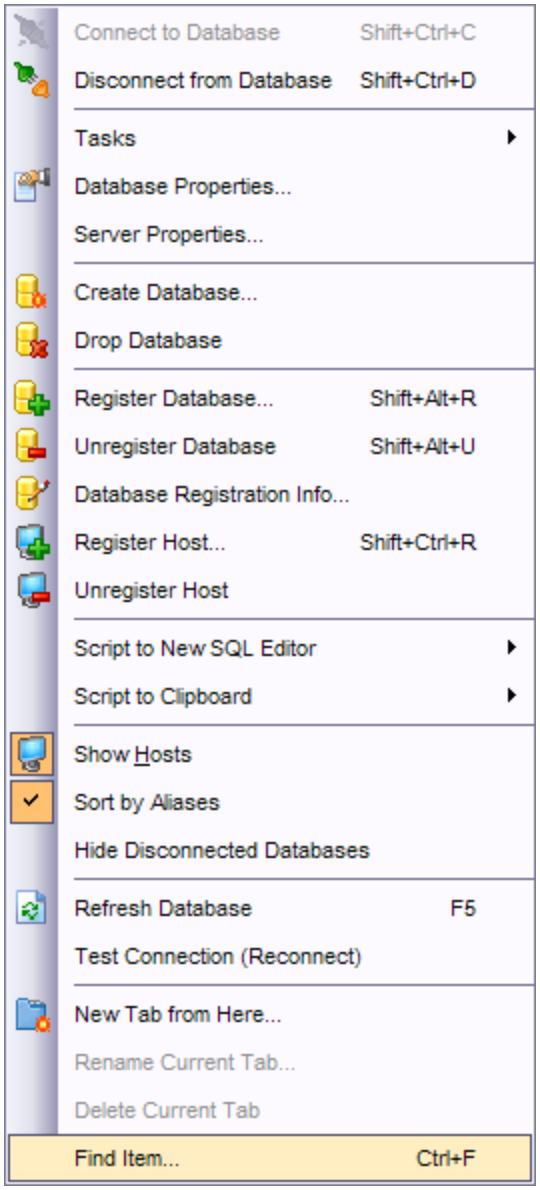
3.8 Searching within the tree

SQL Manager 2008 for SQL Server provides an ability to search for items within the **DB Explorer** tree. Searching for items may be useful if you have a lot of database objects, and it may be sometimes hard to find the one you need.

There are two search facilities implemented in SQL Manager for your convenience. You can search for items within the **DB Explorer** tree in either of the following ways:

- using the **Find Item** dialog

To call the **Find Item** dialog, right-click the **Database** alias or any of the database objects in the **DB Explorer** tree and select the **Find Item...** context menu item, or use the *Ctrl+F* key shortcut.



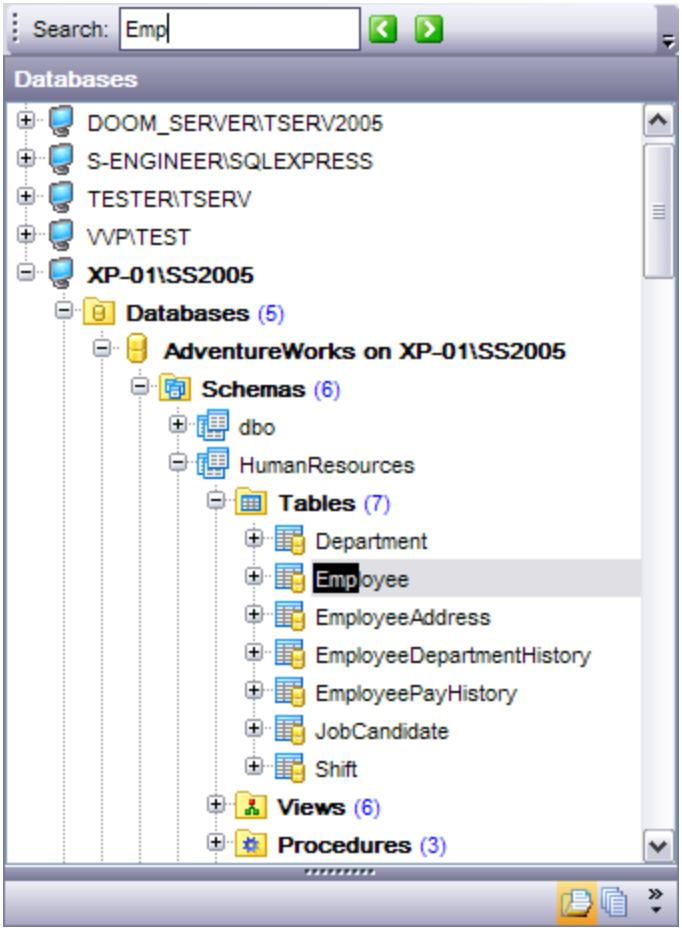
Note that you can specify whether the search will be performed within the entire tree or

© 2008 EMS Database Management Solutions

within the currently selected node only: use the **Search on categories** option available within the Tools | DB Explorer section of the Environment Options dialog.

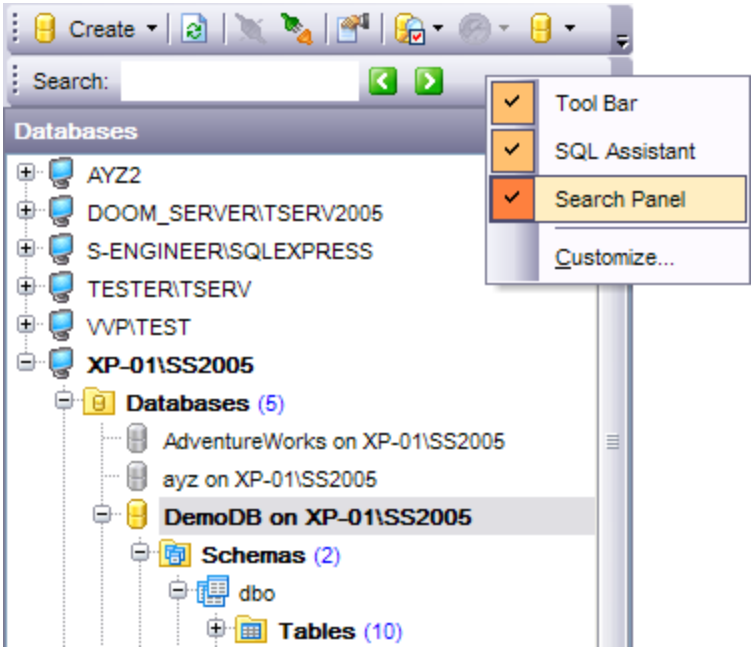
- using the **Search Panel**

Type in the first letters in the edit-box, and the corresponding object will be highlighted in the tree, as displayed in the picture below.



By default, the **Search Panel** is activated in the upper area of DB Explorer. To disable the panel, right-click within the panel and deselect the checkbox at the corresponding popup menu item.

© 2008 EMS Database Management Solutions



Hint: The **Search Panel** is dockable, i.e. you can drag it to any location within the **DB Explorer** form.

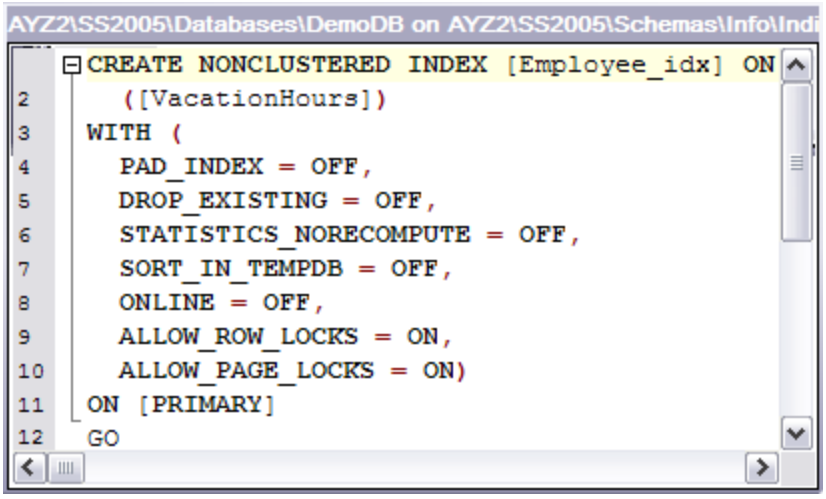
See also:

Managing database registration info
Connecting to databases
Operations with database objects
Selecting multiple objects
Using tabs for database navigation
Recently opened objects
Viewing query history
Creating projects
SQL Assistant
Configuring Database Explorer
Managing Favorite queries

© 2008 EMS Database Management Solutions

3.9 SQL Assistant

SQL Assistant which is located at the bottom of the **Database Explorer** window helps you to work with your database objects. Depending on the current selection in DB Explorer, the SQL Assistant area displays additional information about the selected object.



If you select a **database** in DB Explorer, SQL Assistant displays the list of the database *object groups* and *the number of objects* in each group.

If you select a **schema** in DB Explorer, SQL Assistant displays the schema *definition* (by default).

Selecting an **object group** in DB Explorer displays the list of the *objects* in SQL Assistant. Double-clicking the object name in **SQL Assistant** makes the object available for editing in the appropriate editor. The context menu of the object or group of objects (selected with the *Ctrl* or *Shift* keys pressed) allows you to edit or drop the selected objects.

If you select a **table** or a **view** in DB Explorer, SQL Assistant displays the list of the table subobjects (e.g. *fields* and their *types*) by default. What is displayed in **SQL Assistant** when a table or a view is selected in DB Explorer depends on the **Table Details** / **View Details** selection. Click the **View Mode** toolbar button and select the **Table Details | Show...** (or **View Details | Show...**) drop-down menu item, or use the context menu of SQL Assistant. Possible values are: *Show Fields*, *Show Foreign Keys*, *Show Checks*, *Show Indexes*, *Show Triggers*, *Show Table Info*, *Show DDL*, *Show Description* (for tables); *Show Fields*, *Show Indexes*, *Show Triggers*, *Show DDL*, *Show Description* (for views).

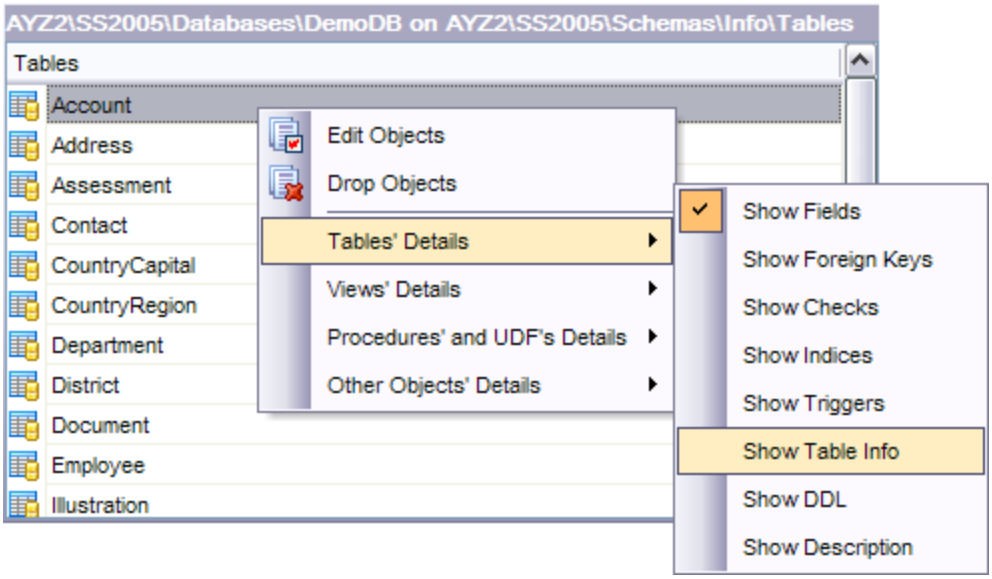
If you select a **procedure** or a **user-defined function** in DB Explorer, SQL Assistant lists its parameters by default. Use the **Procedure and UDF Details | Show...** context menu item within the SQL Assistant area to define the content of SQL Assistant when a procedure or a user-defined function is selected in DB Explorer. Possible values are: *Show Parameters*, *Show DDL*, *Show Description*.

Selecting other objects in **DB Explorer** displays the definition in **SQL Assistant** by

© 2008 EMS Database Management Solutions

default. Use the **Other Objects' Details | Show...** context menu item within the SQL Assistant area to define the content of SQL Assistant when an object is selected in DB Explorer. Possible values are: *Show DDL*, *Show Description*.

You can also use **SQL Assistant** to work with your queries quickly. You can drag-and-drop object aliases to the SQL Editor, Visual Query Builder or SQL Script Editor working area, in the same way as this operation is performed in **Database Explorer**.



See also:

Managing database registration info
Connecting to databases
Operations with database objects
Selecting multiple objects
Using tabs for database navigation
Recently opened objects
Creating projects
Searching within the tree
Configuring Database Explorer
Managing Favorite queries

© 2008 EMS Database Management Solutions

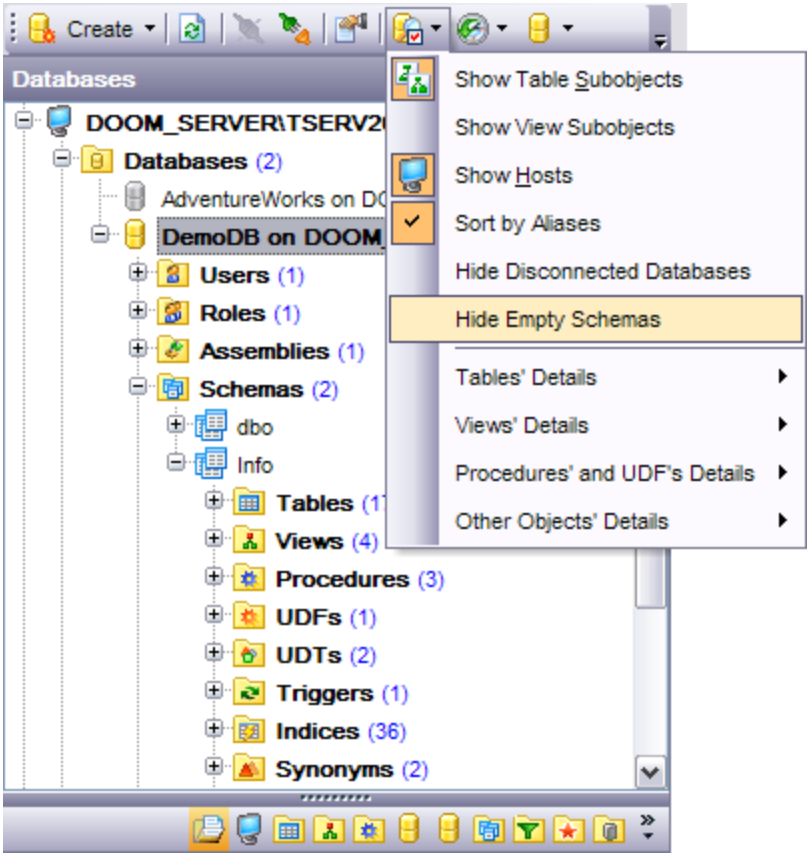
3.10 Configuring Database Explorer

Use the **View Mode** toolbar button to configure **Database Explorer** according to your needs.

The drop-down menu called upon clicking this button allows you to:

- show/hide table subobjects as child nodes of tables;
- show/hide view subobjects as child nodes of views;
- show/hide host nodes for registered databases;
- sort the list of databases by their aliases in the DB Explorer tree;
- show/hide disconnected databases;
- show/hide empty schemas;
- configure table/view/procedure/UDF/other objects' details for the SQL Assistant area.

Use the DB Explorer section of the Environment Options dialog (**Options | Environment Options...**) to see more options to configure **Database Explorer**.



See also:

Managing database registration info
Connecting to databases
Operations with database objects

© 2008 EMS Database Management Solutions

Selecting multiple objects
Using tabs for database navigation
Recently opened objects
Creating projects
Searching within the tree
SQL Assistant
Managing Favorite queries

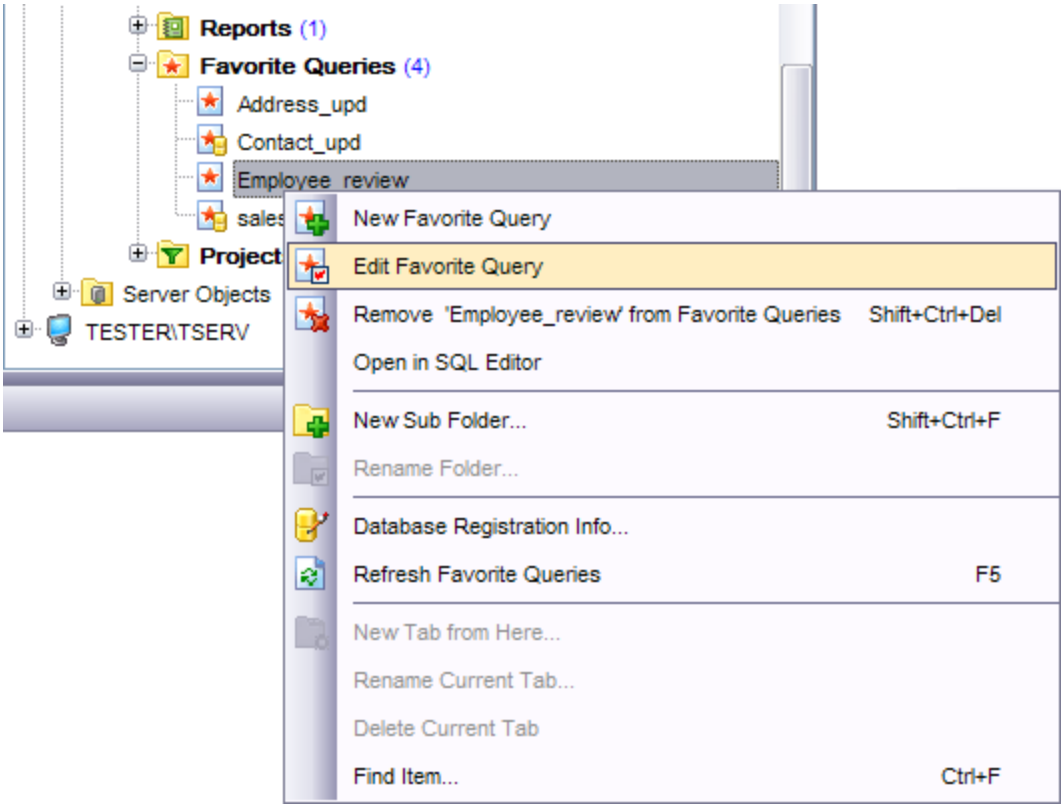
© 2008 EMS Database Management Solutions

3.11 Managing Favorite queries

Favorite Queries is a new feature of SQL Manager. Now you are provided with an opportunity to save the most frequently used SQL queries as Favorite Queries.

Use the **Favorite Queries** node of DB Explorer to access the list of your Favorite queries quickly. Queries stored in the database and those stored in Windows registry can be easily distinguished by their icons.

Using the context menu you can create a new Favorite query or edit an existing one using Favorites editor, open any of the existing queries in SQL Editor or remove a query if you don't need it any longer.



You can also create a separate tab for your Favorite queries. See Using tabs for database navigation section for details.

See also:

Managing database registration info
Connecting to databases
Operations with database objects
Selecting multiple objects
Using tabs for database navigation

© 2008 EMS Database Management Solutions

Recently opened objects
Creating projects
Searching within the tree
SQL Assistant
Configuring Database Explorer

© 2008 EMS Database Management Solutions

Part

4 Database management

SQL Manager 2008 for SQL Server provides a number of tools you may need to manage your SQL Server databases.
Find the list of common database management operations for working in SQL Manager below.

Registering Hosts

In order to register a Host in SQL Manager 2008 for SQL Server:
- select the **Database | Register Host...** main menu item or use the corresponding **Register Host** toolbar button

or
- right-click the Host and select the **Register Host...** context menu item in the DB Explorer tree;
- set all the necessary options using Register Host wizard which guides you through the entire process of Host registration.

Unregistering Hosts

In order to unregister a Host in SQL Manager 2008 for SQL Server:
- select the Host to unregister in the DB Explorer tree;
- select the **Database | Unregister Host** main menu item or use the corresponding **Unregister Host** toolbar button

or
- right-click the Host and select the **Unregister Host** context menu item in the DB Explorer tree;
- confirm unregistering in the corresponding dialog window.

Viewing and editing Server Properties

In order to view/edit Server properties in SQL Manager 2008 for SQL Server:
- select the host or a database that resides on this host in the DB Explorer tree;
- right-click the host or the database alias and select the **Server Properties** context menu item.

Creating Databases

In order to create a Database in SQL Manager 2008 for SQL Server:
- select the **Database | Create Database** main menu item or use the corresponding **Create Database** toolbar button;
- set all the necessary options using Create Database wizard which guides you through the entire process of creating a new database.

Dropping Databases

In order to drop a Database in SQL Manager 2008 for SQL Server:
- select the Database to drop in the DB Explorer tree;
- select the **Database | Drop Database** main menu item;
- confirm dropping in the corresponding dialog window.

© 2008 EMS Database Management Solutions

Registering Databases

In order to register a single Database in SQL Manager 2008 for SQL Server:
- select the **Database | Register Database...** main menu item or use the corresponding **Register Database** toolbar button

or
- right-click the Database and select the **Register Database...** context menu item in the DB Explorer tree;
- set all the necessary options using Register Database wizard which guides you through the entire process of Database registration.

Unregistering Databases

In order to unregister a Database in SQL Manager 2008 for SQL Server:
- select the Database to unregister in the DB Explorer tree;
- select the **Database | Unregister Database** main menu item or use the corresponding **Unregister Database** toolbar button

or
- right-click the Database and select the **Unregister Database** context menu item in the DB Explorer tree;
- confirm unregistering in the corresponding dialog window.

Connecting to Databases

In order to connect to a Database in SQL Manager 2008 for SQL Server:
- select the Database to connect to in the DB Explorer tree;
- select the **Database | Connect to Database** main menu item or use the corresponding **Connect to Database** toolbar button

or
- right-click the Database and select the **Connect to Database** context menu item in the DB Explorer tree.

Disconnecting from Databases

In order to disconnect from a Database in SQL Manager 2008 for SQL Server:
- select the Database to disconnect from in the DB Explorer tree;
- select the **Database | Disconnect from Database** main menu item or use the corresponding **Disconnect from Database** toolbar button

or
- right-click the Database and select the **Disconnect from Database** context menu item in the DB Explorer tree.

Viewing and editing Database Registration Info

In order to view/edit Database Registration Info in SQL Manager 2008 for SQL Server:
- select the Database or any of its objects in the DB Explorer tree;
- select the **Database | Database Registration Info...** main menu item

or
- right-click the Database or any of its objects and select the **Database Registration Info...** context menu item in the DB Explorer tree.

Viewing and editing Database Properties

© 2008 EMS Database Management Solutions

In order to view/edit Database properties in SQL Manager 2008 for SQL Server:

- select the Database in the DB Explorer tree;
- right-click the Database and select the **Database Properties...** context menu item or use the corresponding **Properties** toolbar button

Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**Yes**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

Getting started
Database Explorer
Database objects management
Query management tools
Data management
Import/Export tools
Database tools
Server tools
Personalization
External tools

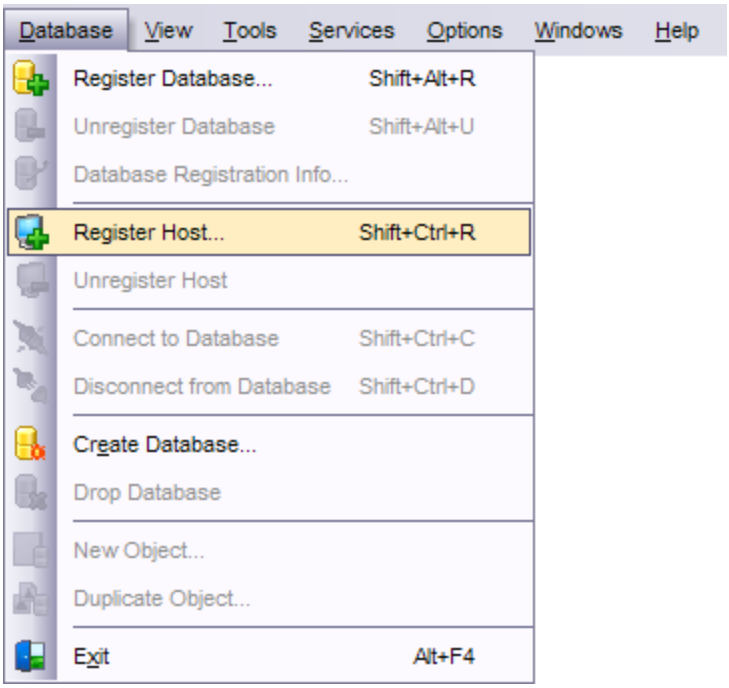
© 2008 EMS Database Management Solutions

4.1 Register Host wizard

Register Host wizard allows you to register a host in SQL Manager 2008 for SQL Server.

To start the wizard, select the **Database | Register Host...** main menu item, or use the **Register Host** button on the main toolbar. You can also use the *Shift+Ctrl+R* shortcut for the same purpose.

- Setting connection parameters
- Selecting databases
- Setting specific options

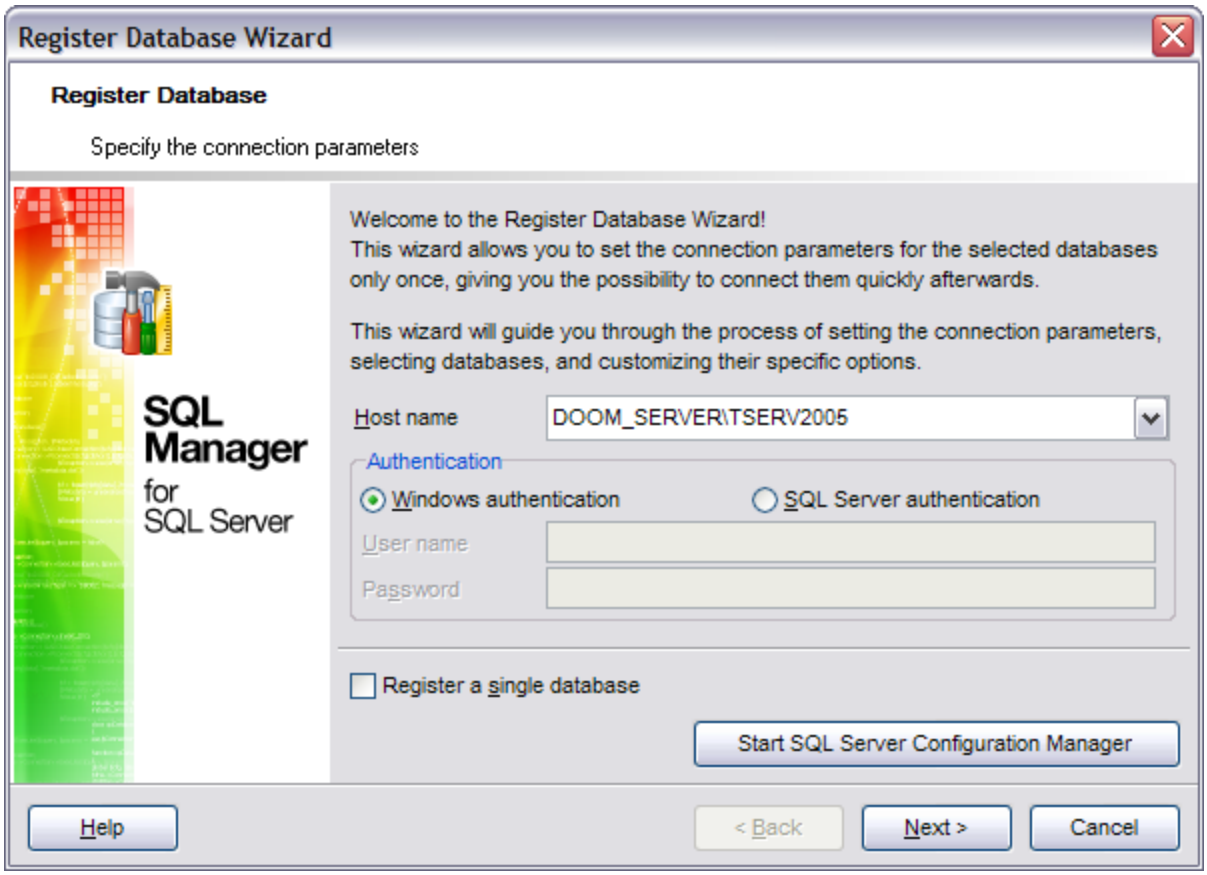


See also:
Create Database wizard
Register Database wizard
Database Registration Info
Database Properties
Server Properties

4.1.1 Setting connection parameters

Use this step of the wizard to set the necessary **connection parameters** for the new Host using the corresponding boxes and options: *Host name*, *Authentication*, *User name* and *Password*.

© 2008 EMS Database Management Solutions



Specify the host you are going to work with: type in the host name in the **Host name** field or select one in the drop-down list.
Please note that if Microsoft® SQL Server™ is installed as a named instance, you should enter the name of your machine and the instance name in the **Host name** field in the following format: *computer_name\sqlserver_instance_name* (e.g. "*MYCOMPUTER\SQLEXPRESS*").

Authentication type
Specify the type of Microsoft® SQL Server™ authentication to be used for the connection: *Windows* authentication or *SQL Server* authentication. It is strongly recommended to avoid using SQL Server authentication with "sa" as the login.

If 'SQL Server' has been selected as the *authentication type*, you should also provide *authorization* settings: **User name** and **Password**.

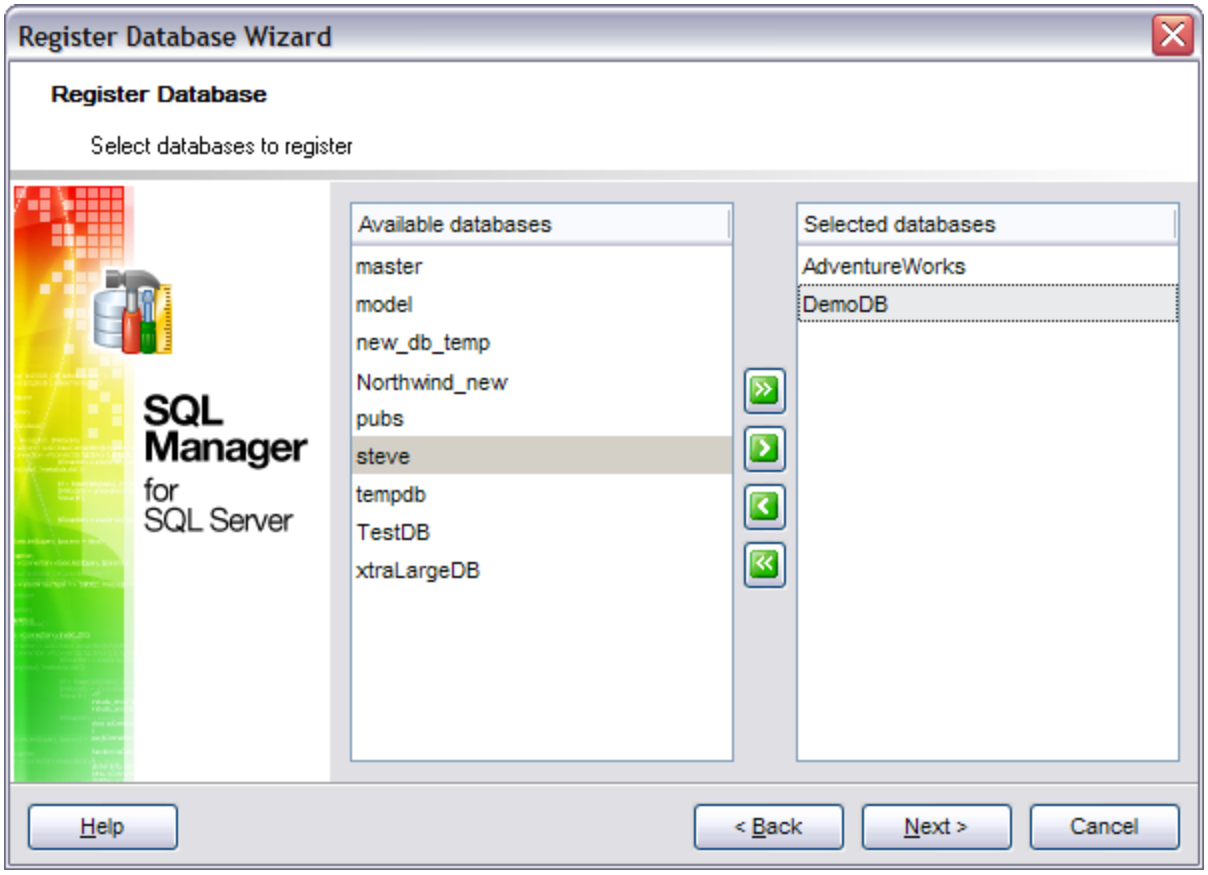
If necessary, you can **start SQL Server Configuration Manager** which is provided by Microsoft® for configuration management of SQL Server services, server/client protocols and client aliases.

Click the **Next** button to proceed to the Setting specific options step or to the Selecting databases step of the wizard, depending on whether the **Register a single database** option has been selected or not.

© 2008 EMS Database Management Solutions

4.1.2 Selecting databases

This step of the wizard allows you to select the database(s) that reside on the host for registration in SQL Manager.
To select a database, you need to move it from the **Available databases** list to the **Selected databases** list. Use the buttons or drag-and-drop operations to move the databases from one list to another.



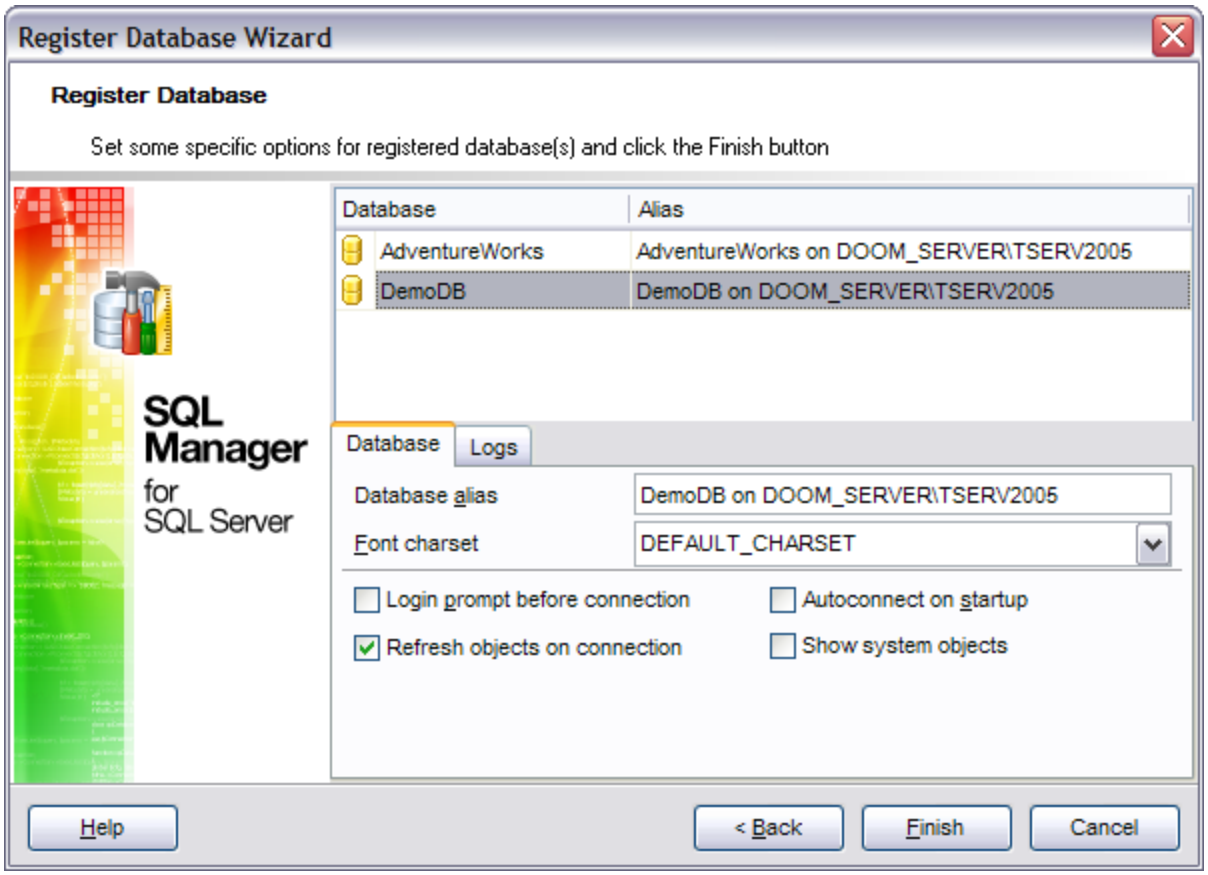
Click the **Next** button to proceed to the Setting specific options step of the wizard.

4.1.3 Setting specific options

This step of the wizard allows you to set the **database registration options** using the corresponding tabs: *General database options*, *Log settings*.

You can select several databases to set options for all the selected databases simultaneously (except for the aliases which must be unique for each individual database).

© 2008 EMS Database Management Solutions



The **Database** tab allows you to set common database registration options:

Database alias
Enter the alias to be used to display the database in DB Explorer and SQL Manager tools.

Font charset
Specify the character set to be used to display data in the grid.

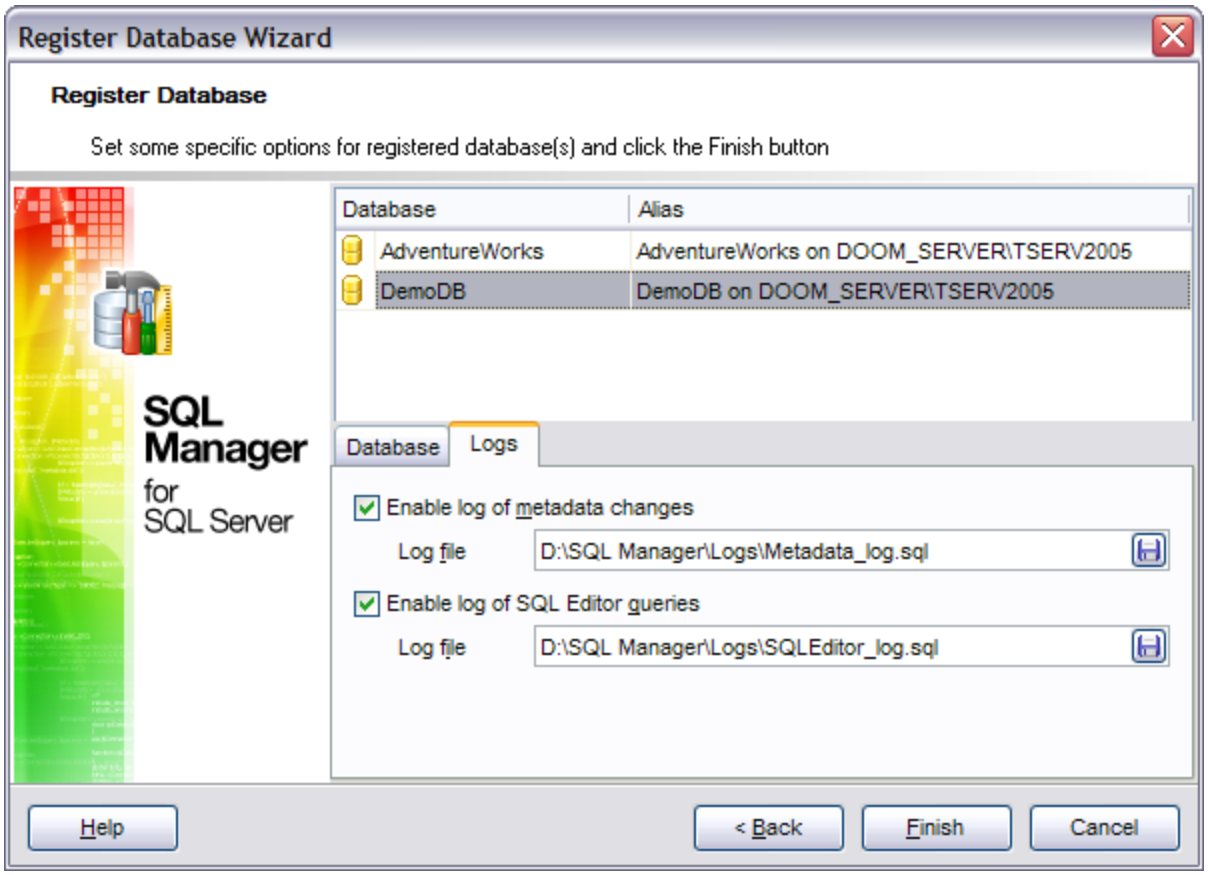
Login prompt before connection
Enables SQL Manager 2008 for SQL Server to prompt for user name and password each time you connect to the database.

Autoconnect at startup
With this option set, connection to the registered database is established automatically at application startup.

Refresh objects on connection
This option allows you to enable/disable refreshing objects on connection to the database. It is highly recommended to uncheck this option if your database contains many objects or if connection to the database is slow.

Show system objects
If this option is checked, SQL Server system objects will be displayed in DB Explorer.

© 2008 EMS Database Management Solutions



The **Logs** tab allows you to set log options for the database(s):

Enable log of metadata changes
Check this option if you wish to log metadata changes of your database(s) in a file.

Log file
This field is enabled if the **Enable log of metadata changes** option is selected. Type in or use the **Save as** button to specify the path to the **.sql* file to store the metadata logs.

Enable log of SQL Editor queries
Check this option if you wish to log your SQL Editor queries in a file.

Log file
This field is enabled if the **Enable log of SQL Editor queries** option is selected. Type in or use the **Save as** button to specify the path to the **.sql* file to store the logs of SQL queries.

Click the **Finish** button when done to start working with the newly registered host and database(s) in SQL Manager 2008 for SQL Server.

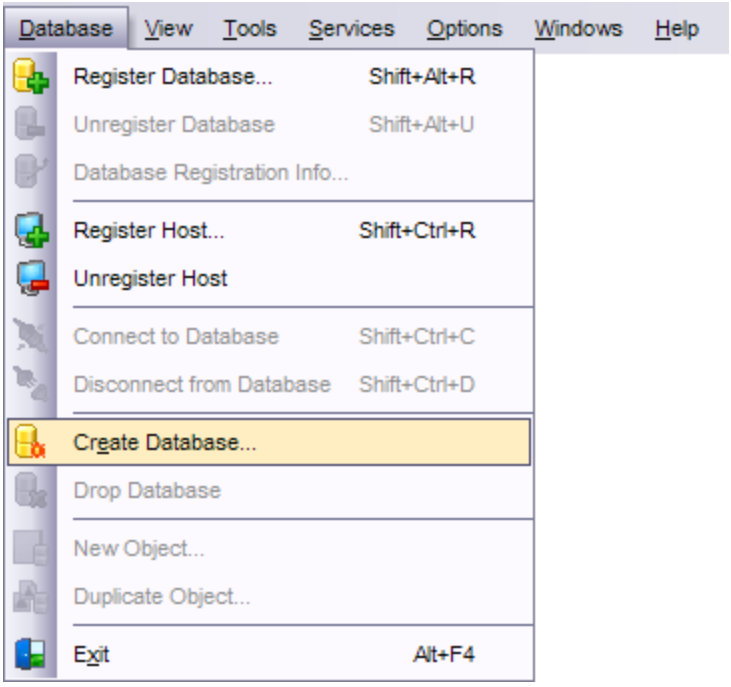
© 2008 EMS Database Management Solutions

4.2 Create Database wizard

Create Database wizard allows you to create a new database on your SQL Server host.

To start the wizard, select the **Database | Create Database...** main menu item, or use the **Create Database** button on the main toolbar.

- Setting database name
- Setting connection properties
- Defining database files
- Defining transaction log files
- Setting recovery model and collation
- Viewing result SQL statement

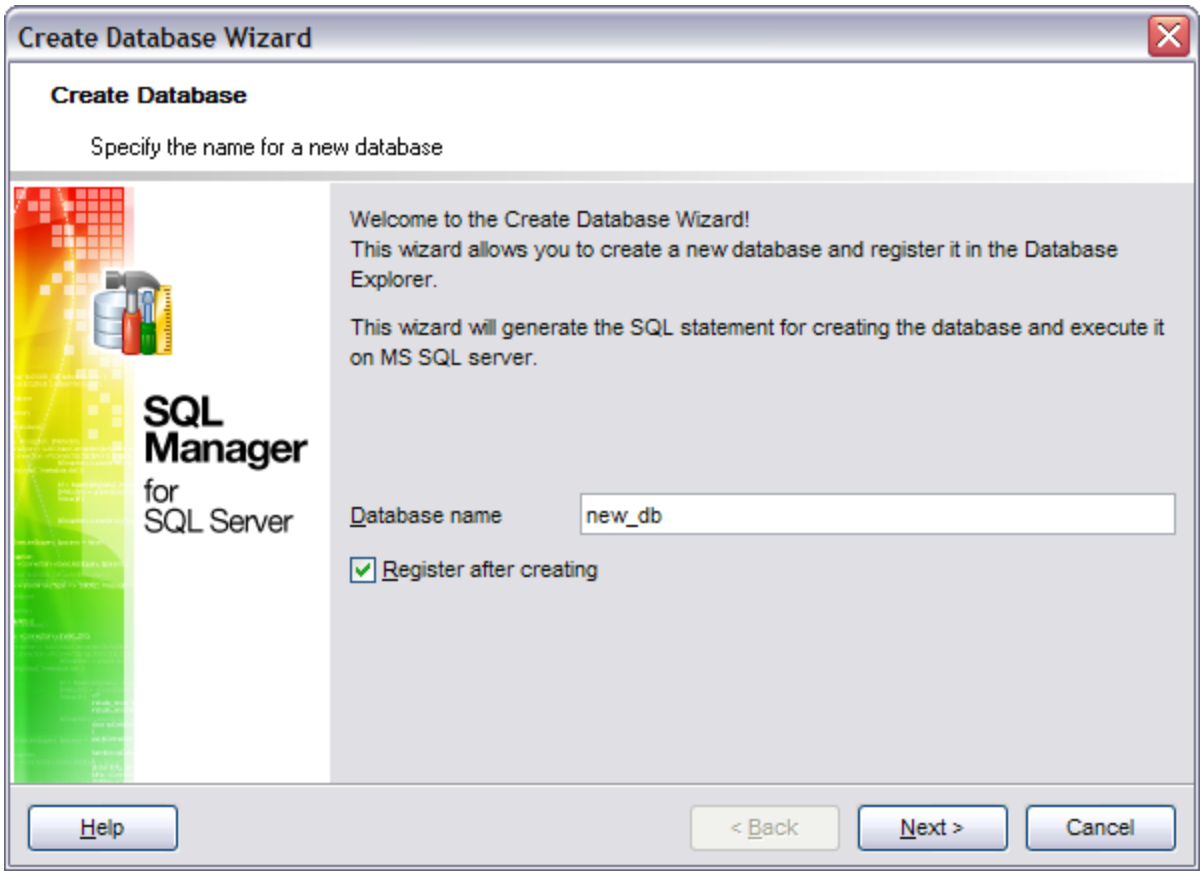


See also:
Register Host wizard
Register Database wizard
Database Registration Info
Database Properties
Server Properties

4.2.1 Setting database name

Use this step of the wizard to enter a **name** for the database being created.

© 2008 EMS Database Management Solutions



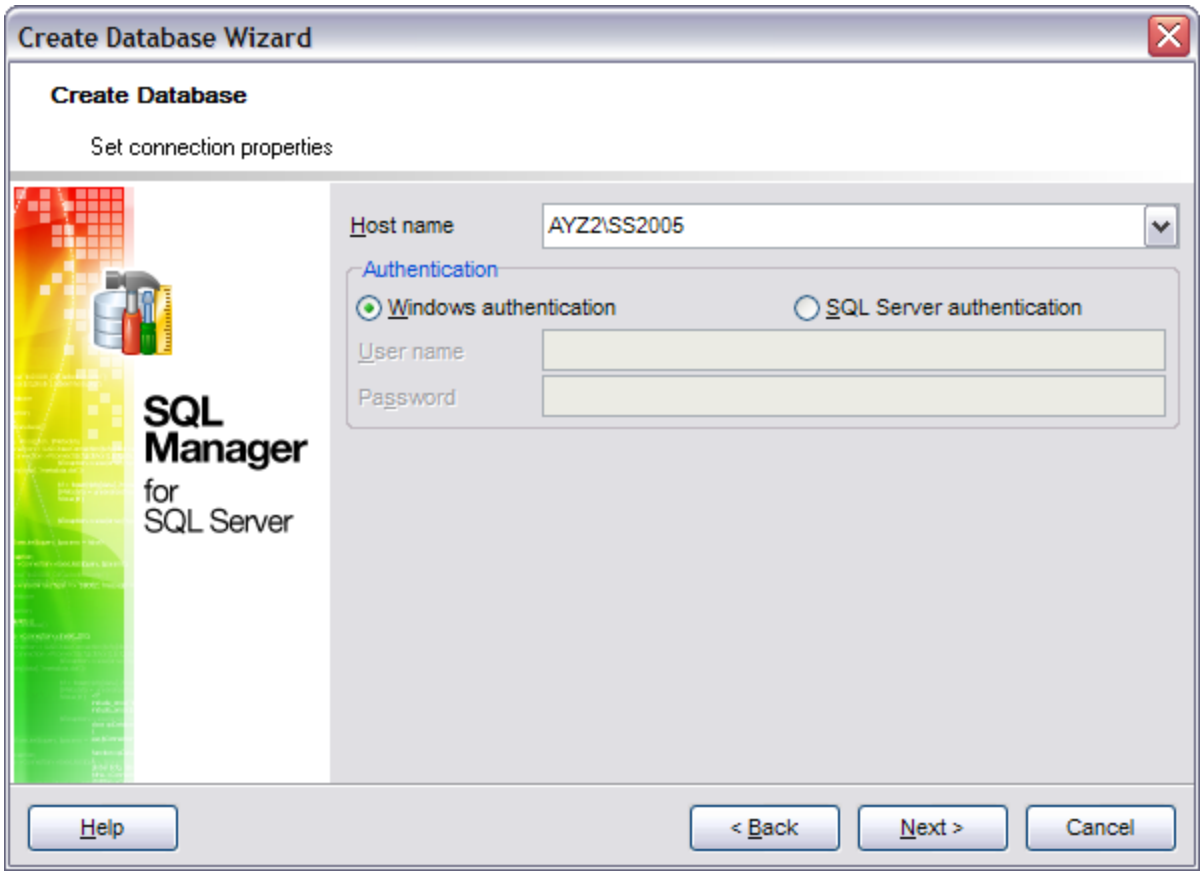
Register after creating
Check this option to register the newly created database in SQL Manager.

Click the **Next** button to proceed to the Setting connection properties step of the wizard.

4.2.2 Setting connection properties

Use this step of the wizard to set the necessary **connection parameters** for the database being created. Use the corresponding boxes and options: *Host name*, *Authentication*, *User name* and *Password*.

© 2008 EMS Database Management Solutions



Specify the host where the database being created will reside: type in the host name in the **Host name** field or select one in the drop-down list.
Please note that if Microsoft® SQL Server™ is installed as a named instance, you should enter the name of your machine and the instance name in the **Host name** field in the following format: *computer_name\sqlserver_instance_name* (e.g. "*MYCOMPUTER\SQLEXPRESS*").

Authentication type
Specify the type of Microsoft® SQL Server™ authentication to be used for the connection: *Windows* authentication or *SQL Server* authentication. It is strongly recommended to avoid using SQL Server authentication with "sa" as the login.

If 'SQL Server' has been selected as the *authentication type*, you should also provide *authorization* settings: **User name** and **Password**.

Click the **Next** button to proceed to the Defining database files step of the wizard.

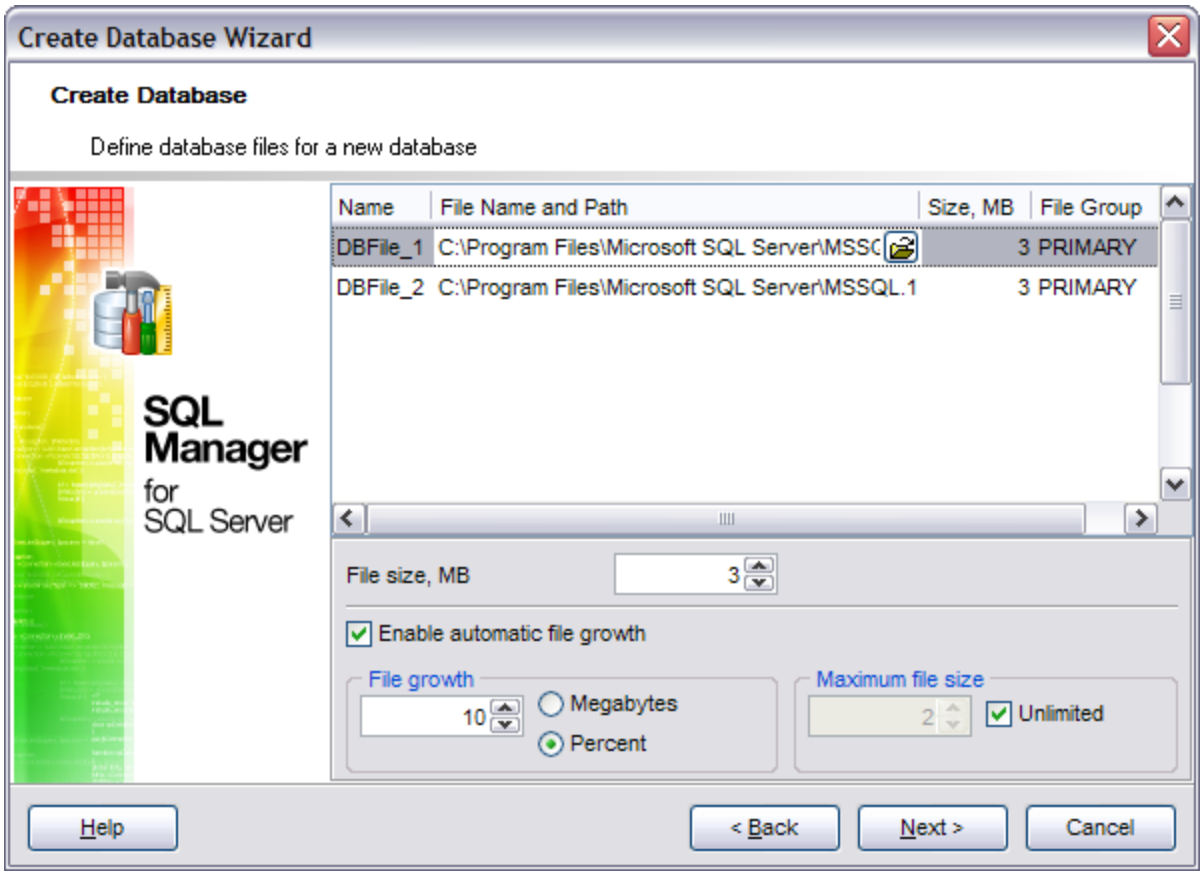
4.2.3 Defining database files

This step of the wizard allows you to define **database files** for the new database.

© 2008 EMS Database Management Solutions

Here you can specify that the disk files used to store the data portions of the database (data files) are defined explicitly, and fill the list of file items defining the data files for the primary file group.

To add a file, right-click in the main working area and select the **Add file** popup menu item.
To remove a file, right-click the file in the list and select the **Delete file** popup menu item.



The **Name** is a logical name used to refer to the physical file in all Transact-SQL statements. The logical file name must conform to the rules for SQL Server identifiers and must be unique within the database.

The **File Name and Path** is the name of the physical file including the directory path. The path must follow the rules for operating system file names.

File Group
The *Primary* file group contains the primary data file and any other files not specifically assigned to another file group. All pages for the system tables are allocated in the primary file group. *User-defined* file groups are any other file groups specified by the user.

File size
SQL Server 2005 files can grow automatically from their originally specified size. When you define a file, you can specify a **file growth** increment. Each time the file fills, it increases

© 2008 EMS Database Management Solutions

its size by the growth increment. If there are multiple files in a file group, they do not autogrow until all the files are full. Growth then occurs using a round-robin algorithm.

Each file can also have **maximum file size** specified. If **Unlimited** is specified, the file can continue to grow until it has used all available space on the disk.

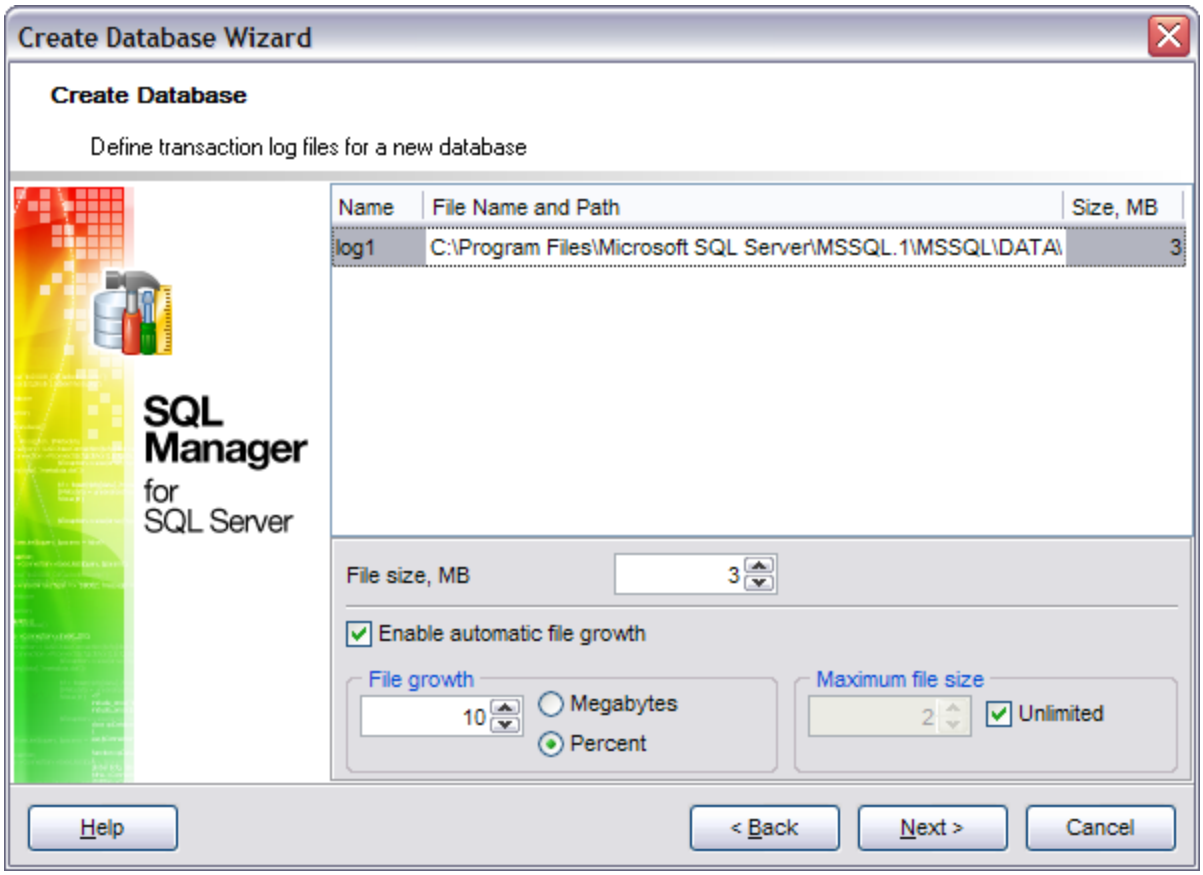
Click the **Next** button to proceed to the Defining transaction log files step of the wizard.

4.2.4 Defining transaction log files

This step of the wizard allows you to define **transaction log files** for the new database.

Log files hold all of the log information used to recover the database. There must be at least one log file for each database, although there can be more than one. The recommended file name extension for log files is **.ldf*.

To add a file, right-click in the main working area and select the **Add file** popup menu item.
To remove a file, right-click the file in the list and select the **Delete file** popup menu item.



© 2008 EMS Database Management Solutions

The **Name** is a logical name used to refer to the physical file in all Transact-SQL statements. The logical file name must conform to the rules for SQL Server identifiers and must be unique within the database.

The **File Name and Path** is the name of the physical file including the directory path. The path must follow the rules for operating system file names.

File size
SQL Server 2005 files can grow automatically from their originally specified size. When you define a file, you can specify a **file growth** increment. Each time the file fills, it increases its size by the growth increment. If there are multiple files in a file group, they do not autogrow until all the files are full. Growth then occurs using a round-robin algorithm.

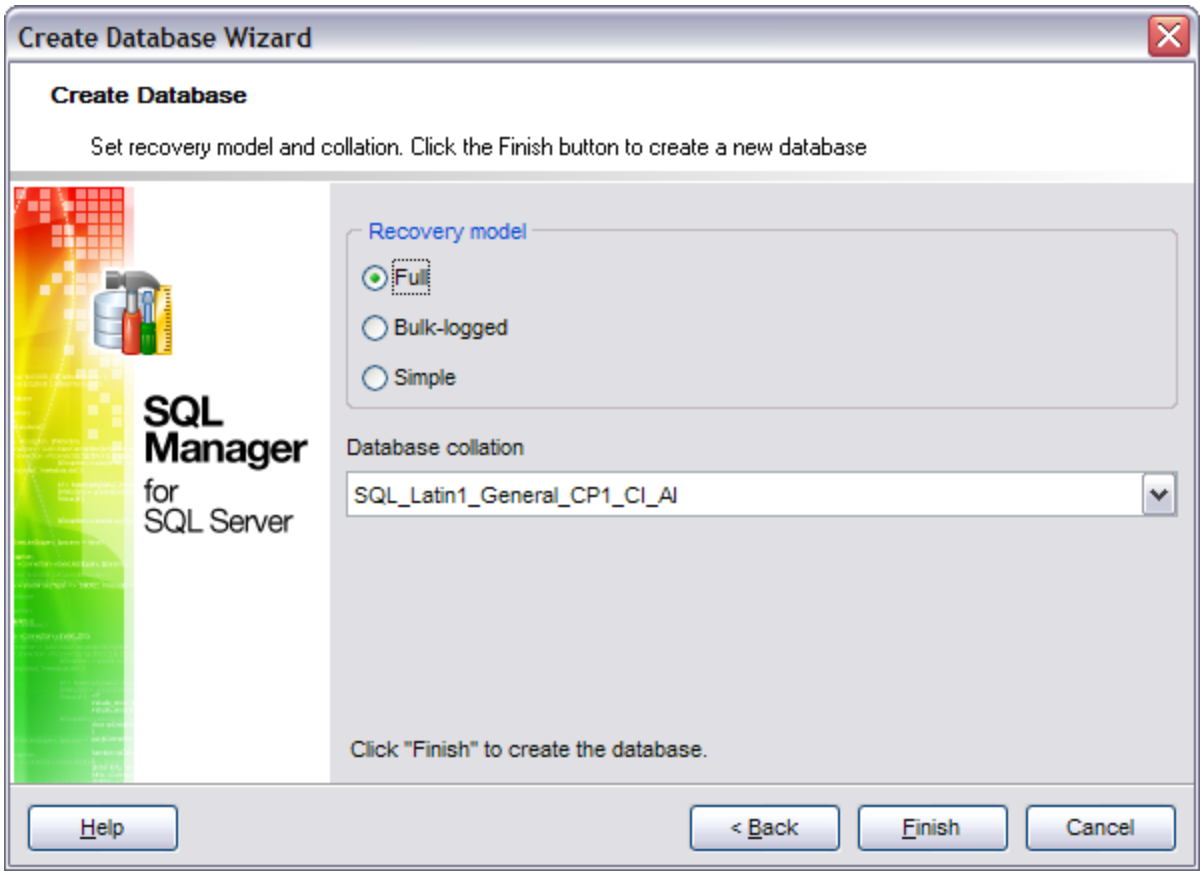
Each file can also have **maximum file size** specified. If **Unlimited** is specified, the file can continue to grow until it has used all available space on the disk.

Click the **Next** button to proceed to the Setting recovery model and collation step of the wizard.

4.2.5 Setting recovery model and collation

This step of the wizard allows you to set **recovery model** and **collation** for the new database.

© 2008 EMS Database Management Solutions



Recovery model

SQL Server recovery models are used for backup and restore operations. A *recovery model* is a database property that controls the basic behavior of backup and restore operations for a database: how transactions are logged, whether the transaction log requires backup, what kinds of restore operations are available, etc.

Full
When this recovery model is set, all transactions are fully logged and all the transaction log records are retained until after they are backed up. In the Enterprise Edition of SQL Server, the full recovery model allows a database to be recovered to the point of failure, assuming that the tail of the log has been backed up after the failure.

Bulk-logged
When this recovery model is set, most bulk operations are minimally logged (e.g. index creation and bulk loads), and other transactions are fully logged.

Simple
When this recovery model is set, most transactions are minimally logged: only the information required to ensure database consistency after a system crash or after restoring a data backup.

For detailed information concerning recovery models refer to SQL Server documentation.

© 2008 EMS Database Management Solutions

Database collation
Use the drop-down list to specify collation for the database being created. By default, the server collation is used.

When you are done, click the **Finish** button to view the result SQL statement.

4.2.6 Defining filestream files and filegroups

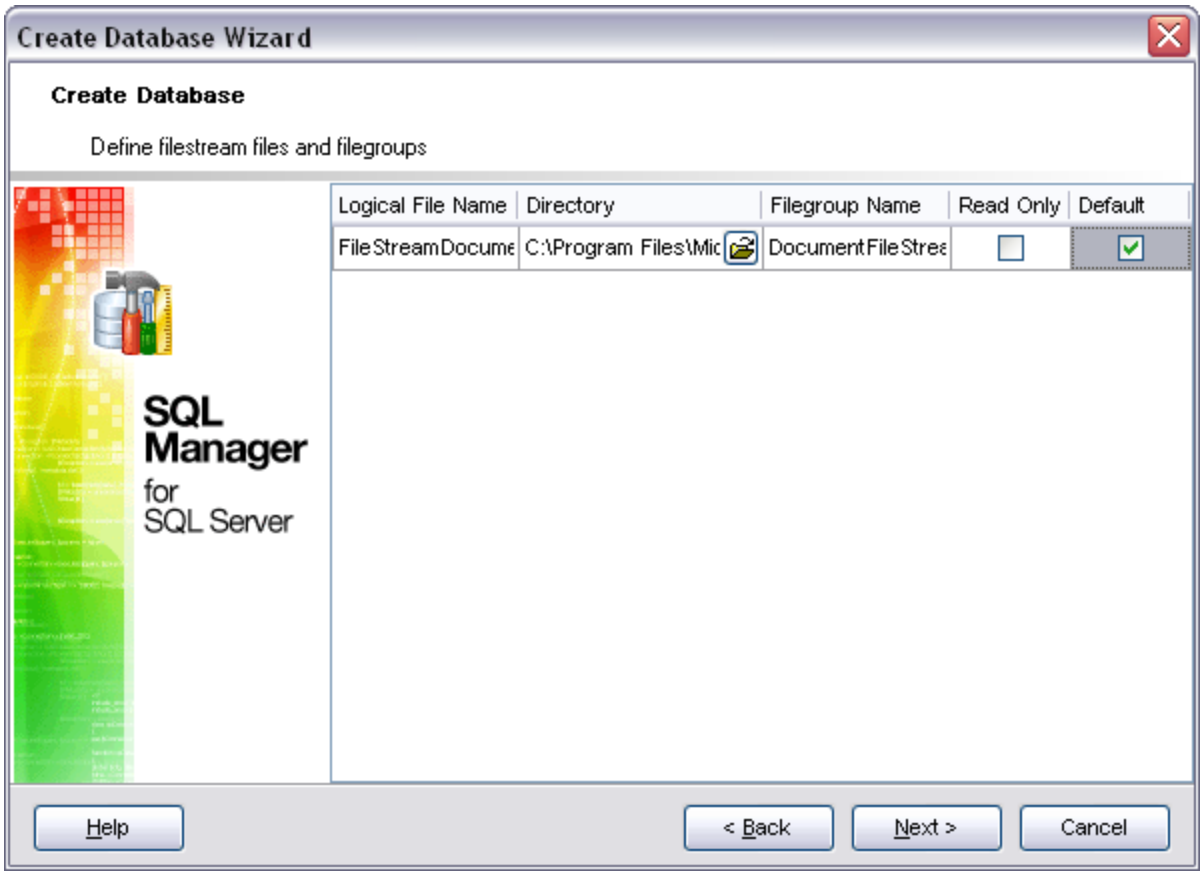
This step i only available only in the SQL Server 2008. Use this step to define **filestream files and filegroups** for the database.

To create a new Filestream filegroup select the **Add new filestream file** context menu item.

To create the new **Filestream filegroup** you need to specify:

Filegroup name
The name of the filestream filegroup.
Logical file name
The name of the logical file storing the data. This filename is not the name of the physical file.
Directory
The physical location of the data. The directory must be located on the server on which the database is registered.

© 2008 EMS Database Management Solutions



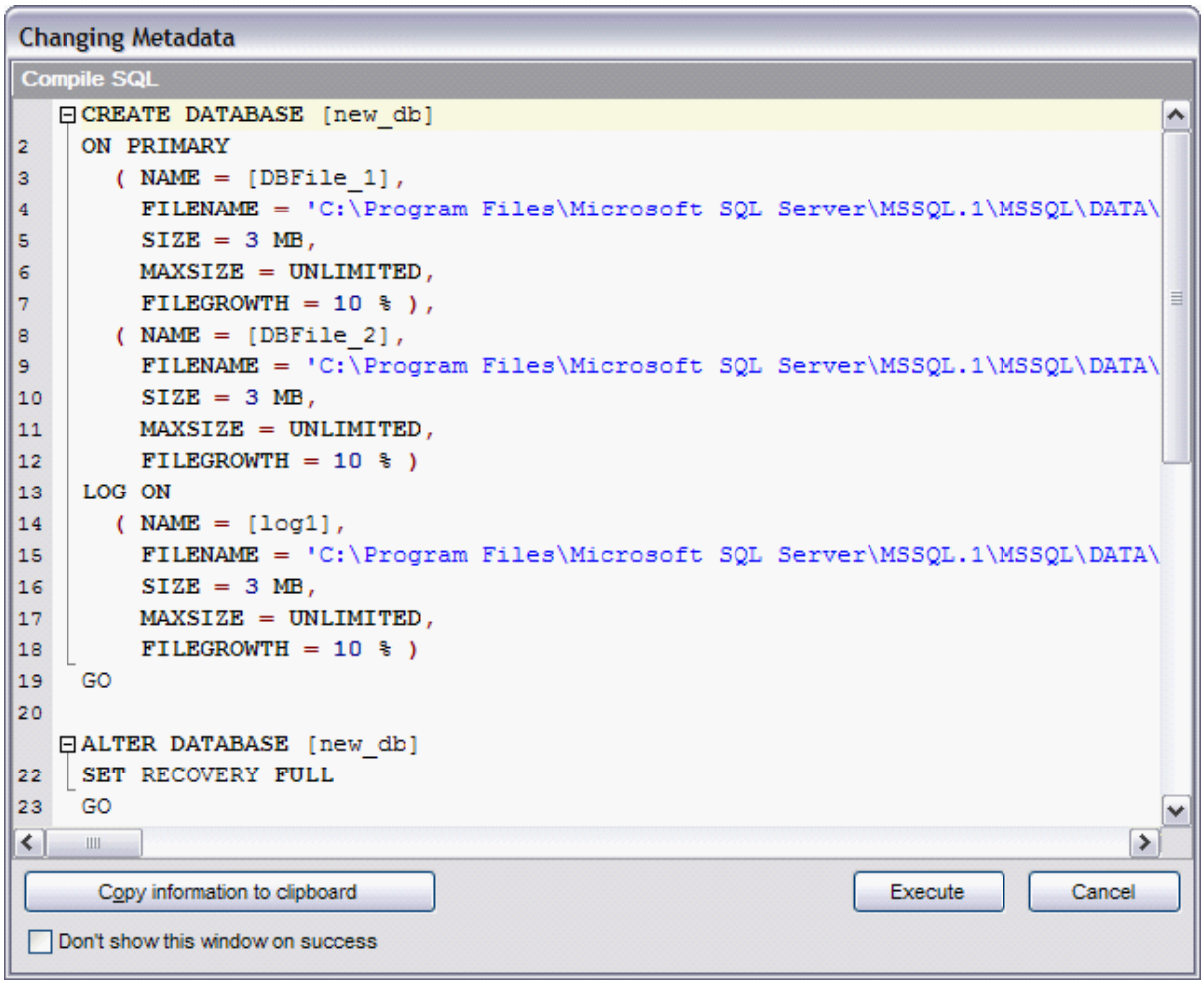
Read Only
Specifies that the filestream filegroup is read-only. Updates to objects that reside in this filestream filegroup are not allowed. The defalult fiestream filegroup can't be read-only.

Default
The filestream fields data will be stored in this filestream filegroup if none was specified for the table.

4.2.7 Viewing result SQL statement

View the result *CREATE DATABASE* statement issued for the new database creation. You can alter the script manually, if necessary. For more information see Changing Metadata window.

© 2008 EMS Database Management Solutions



If the **Register after creating** option was checked at the Setting database name step of the wizard, the Database Registration Info dialog will appear.

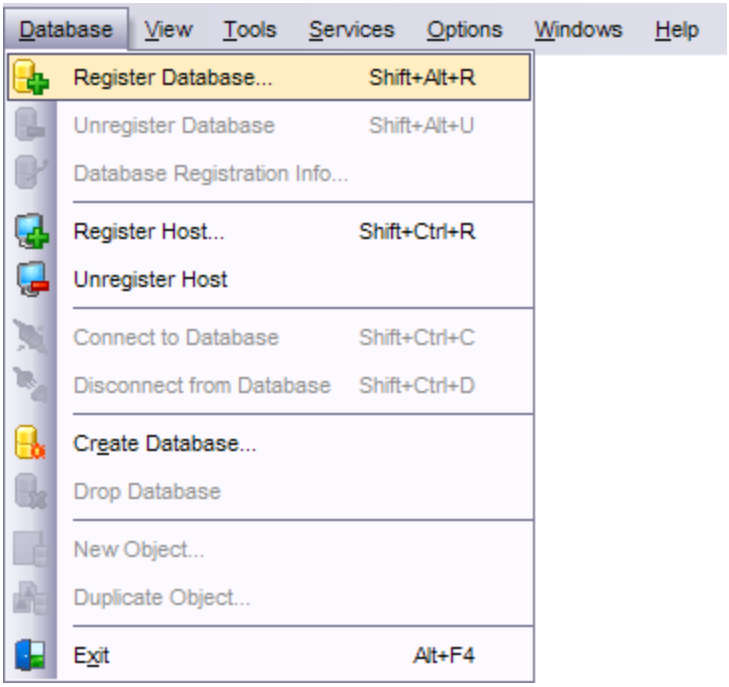
© 2008 EMS Database Management Solutions

4.3 Register Database wizard

Register Database Wizard allows you to register a single database.

To start the wizard, select the **Database | Register Database...** main menu item, or use the **Register Database** button on the main toolbar. You can also use the *Shift+Alt+R* shortcut for the same purpose.

- Setting connection parameters
- Setting specific options



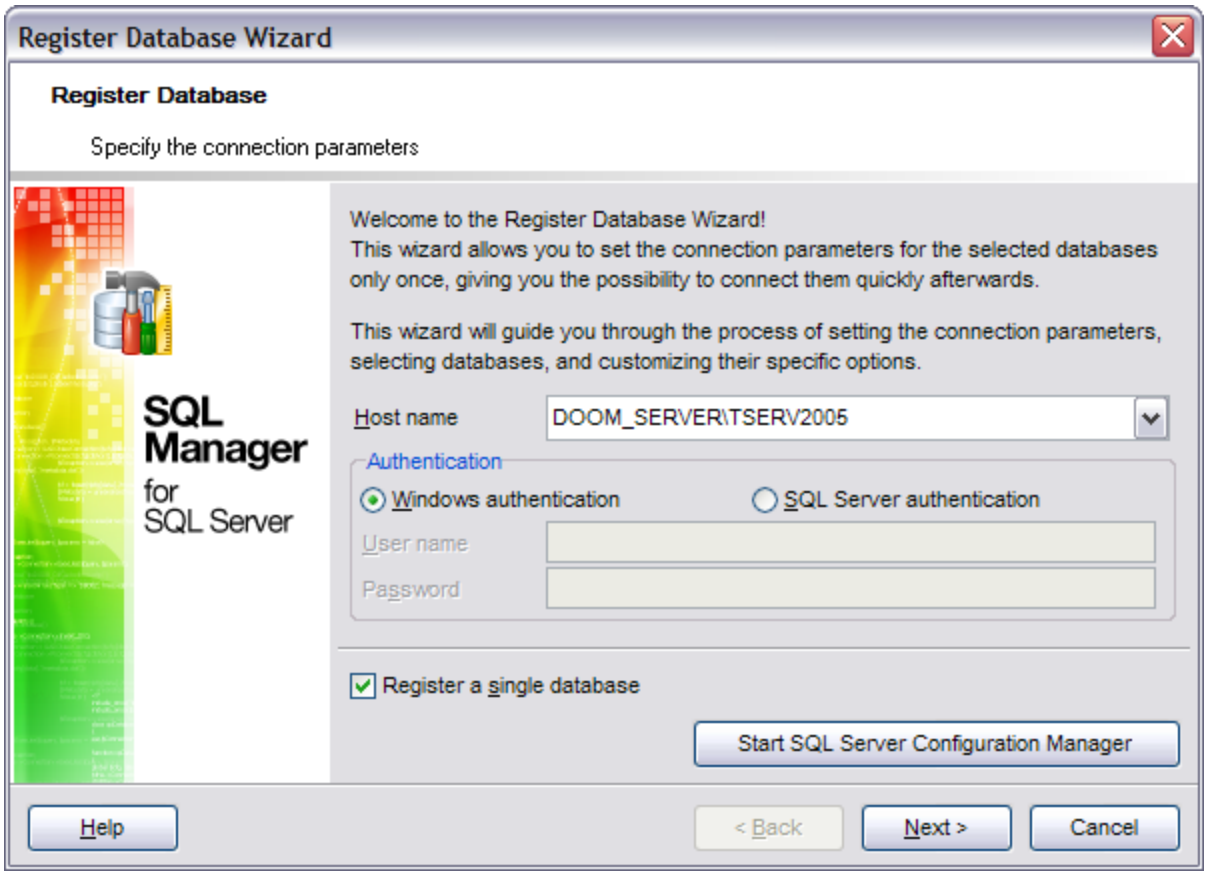
See also:

Register Host wizard
Create Database wizard
Database Registration Info
Database Properties
Server Properties

4.3.1 Setting connection parameters

Use this step of the wizard to set the necessary **connection parameters** for the new database using the corresponding boxes and options: *Host name*, *Authentication*, *User name* and *Password*.

© 2008 EMS Database Management Solutions



Specify the host where the database being registered resides: type in the host name in the **Host name** field or select one in the drop-down list.
Please note that if Microsoft® SQL Server™ is installed as a named instance, you should enter the name of your machine and the instance name in the **Host name** field in the following format: *computer_name\sqlserver_instance_name* (e.g. "*MYCOMPUTER\SQLEXPRESS*").

Authentication type
Specify the type of Microsoft® SQL Server™ authentication to be used for the connection: *Windows* authentication or *SQL Server* authentication. It is strongly recommended to avoid using SQL Server authentication with "sa" as the login.

If 'SQL Server' has been selected as the *authentication type*, you should also provide *authorization* settings: **User name** and **Password**.

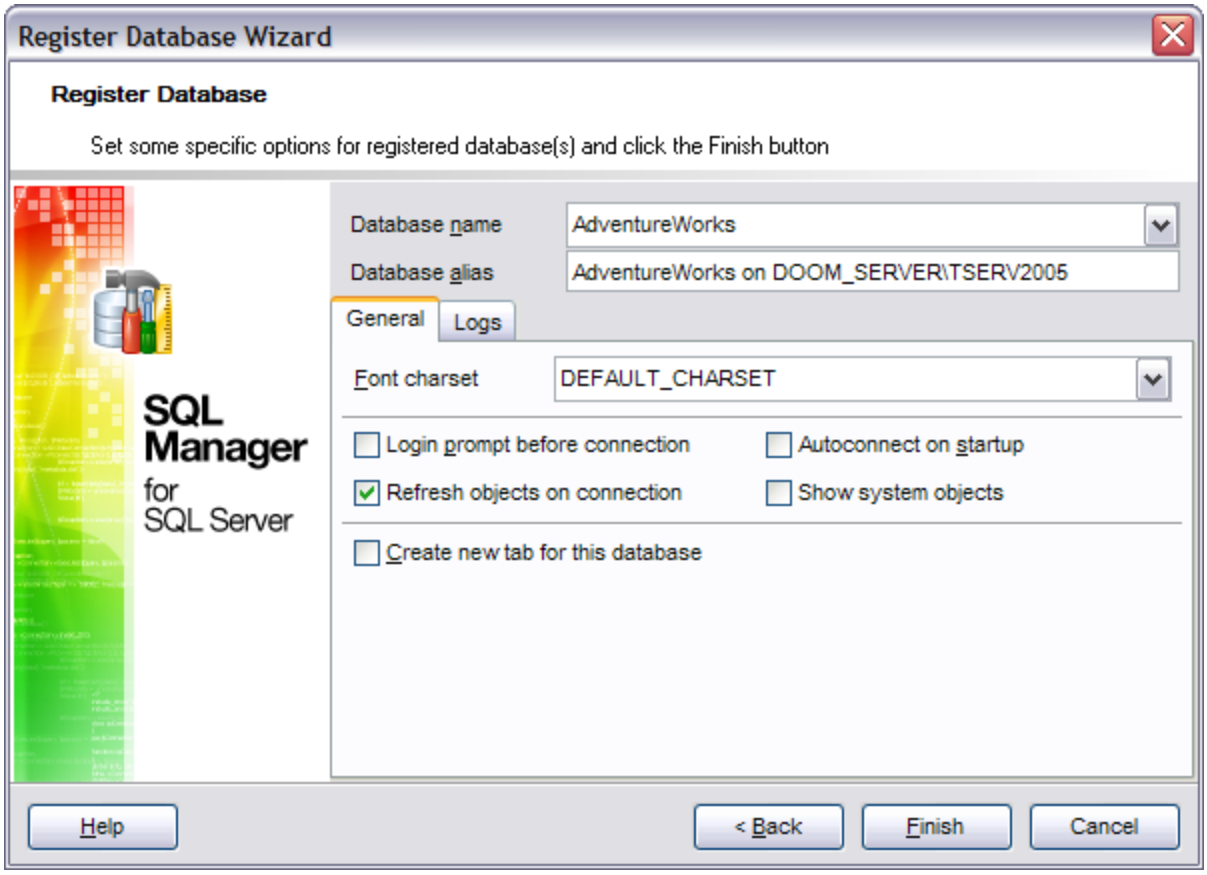
If necessary, you can **start SQL Server Configuration Manager** which is provided by Microsoft® for configuration management of SQL Server services, server/client protocols and client aliases.

Click the **Next** button to proceed to the Setting specific options step or to the Selecting databases step of the wizard, depending on whether the **Register a single database** option has been selected or not.

© 2008 EMS Database Management Solutions

4.3.2 Setting specific options

This step of the wizard allows you to set the **database registration options** using the corresponding tabs: *General database options*, *Log settings*.



The **General** tab allows you to set common database registration options:

Database name
Type in or use the drop-down list to select the database to be registered.

Database alias
Enter the alias to be used to display the database in DB Explorer and SQL Manager tools.

Font charset
Specify the character set to be used to display data in the grid.

Login prompt before connection
Enables SQL Manager 2008 for SQL Server to prompt for user name and password each time you connect to the database.

Autoconnect at startup

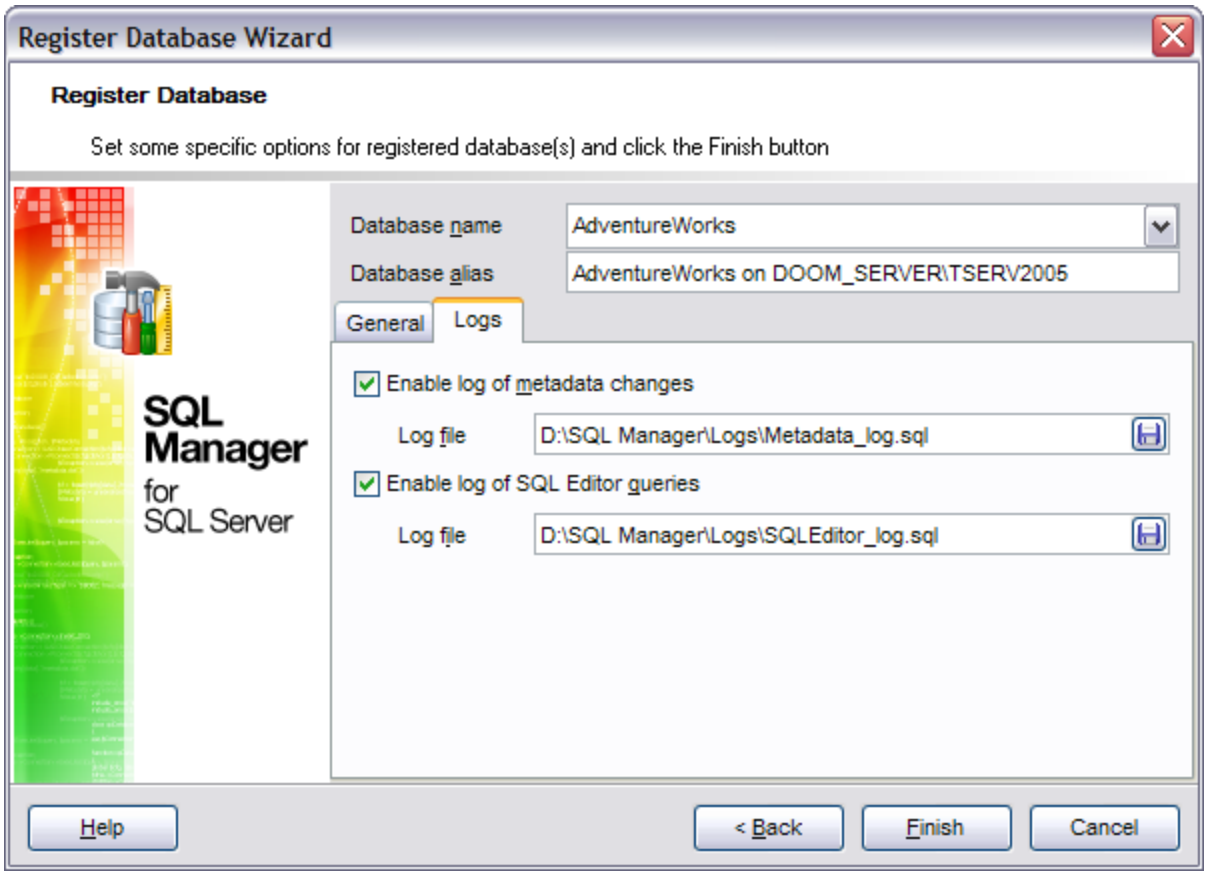
© 2008 EMS Database Management Solutions

With this option set, connection to the registered database is established automatically at application startup.

Refresh objects on connection
This option allows you to enable/disable refreshing objects on connection to the database. It is highly recommended to uncheck this option if your database contains many objects or if connection to the database is slow.

Show system objects
If this option is checked, SQL Server system objects will be displayed in DB Explorer.

Create new tab for this database
If this option is checked, the database will be displayed within a separate tab in DB Explorer.



The **Logs** tab allows you to set log options for the database:

Enable log of metadata changes
Check this option if you wish to log metadata changes of your database in a file.

Log file
This field is enabled if the **Enable log of metadata changes** option is selected. Type in or use the **Save as** button to specify the path to the **.sql* file to store the metadata logs.

© 2008 EMS Database Management Solutions

Enable log of SQL Editor queries
Check this option if you wish to log your SQL Editor queries in a file.

Log file
This field is enabled if the **Enable log of SQL Editor queries** option is selected. Type in or use the **Save as** button to specify the path to the **.sql* file to store the logs of SQL queries.

Click the **Finish** button when done to start working with the newly registered database in SQL Manager 2008 for SQL Server.

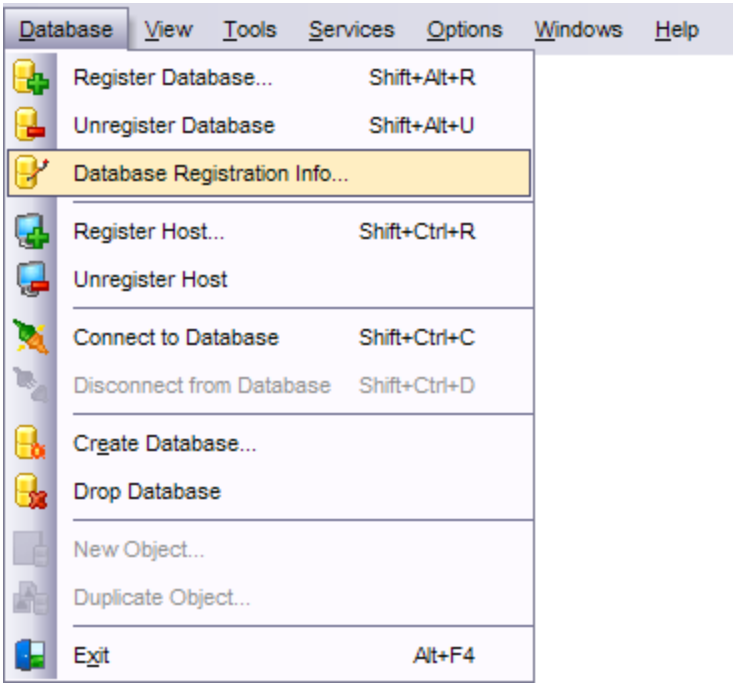
© 2008 EMS Database Management Solutions

4.4 Database Registration Info

Use the **Database Registration Info** dialog to view and edit the registration properties of the database.

To open the dialog, select the database or any of its objects in the DB Explorer tree, then select the **Database | Database Registration Info...** main menu item, or right-click the database alias in DB Explorer and use the **Database Registration Info...** context menu item. You can also use the **Database Registration Info...** button on the main toolbar.

- Editing connection properties
- Setting database options
- Setting display options
- Setting default directories
- Setting log options
- Find Option



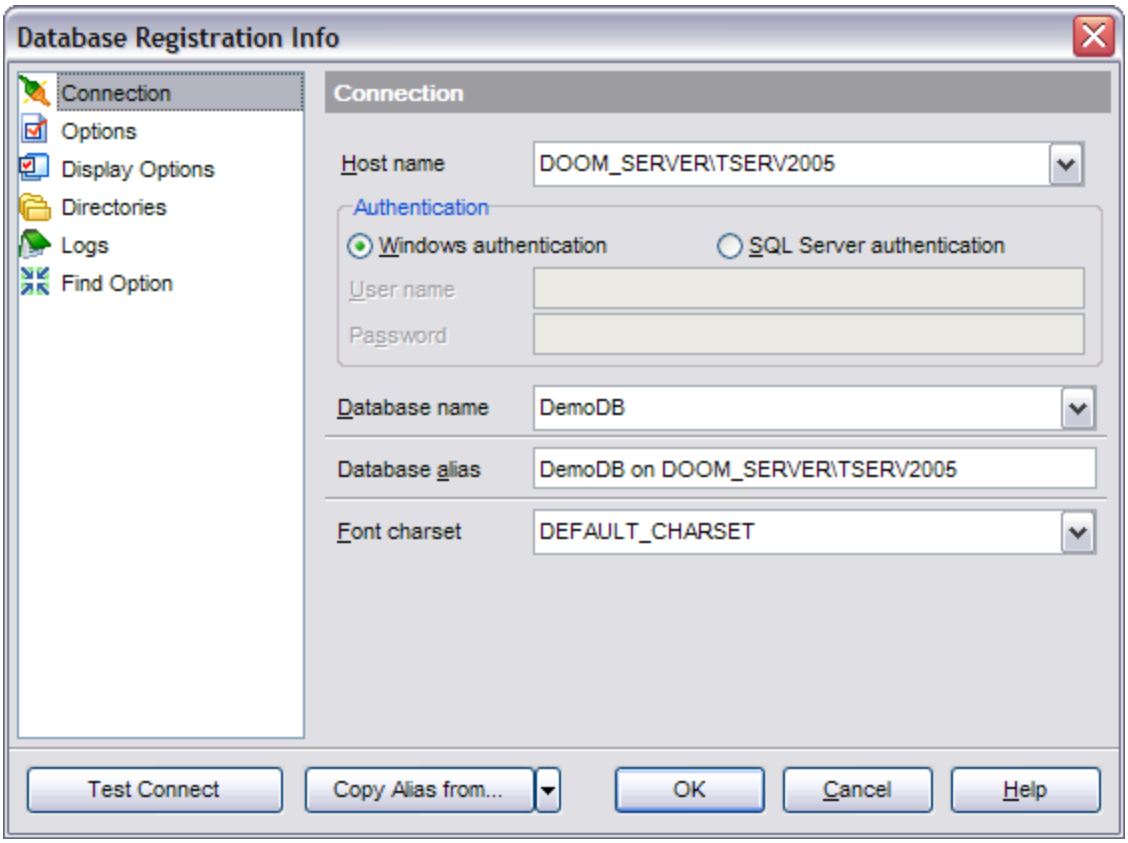
See also:

Register Host wizard
Create Database wizard
Register Database wizard
Database Properties
Server Properties

© 2008 EMS Database Management Solutions

4.4.1 Editing connection properties

The **Connection** section of the **Database Registration Info** dialog allows you to view and/or edit the connection properties in the corresponding boxes: *Host name*, *Authentication*, *User name*, *Password*, *Database name*, *Database alias*, *Font charset*.



Host name
Stores the name of the Host where the database resides.

Authentication type
Specifies the type of Microsoft® SQL Server™ authentication used for the connection:
Windows authentication
Select this type if you wish to use Windows Integrated Security for the login.

SQL Server authentication
Select this type if you wish to use SQL Server Authentication for the login. SQL Server Authentication stores logins and passwords in SQL Server and does not depend upon Windows. You can select SQL Server Authentication only if SQL Server Authentication has been enabled on the server.

User name
If necessary, edit the User name used to access the database (SQL Server authentication).

© 2008 EMS Database Management Solutions

Password
If necessary, edit the Password used to access the database (SQL Server authentication).

Database name
Stores the name of the database.

Database alias
Stores the database alias which is displayed in the DB Explorer tree and SQL Manager tools.

Font charset
Stores the character set used to display data in the grid.

Once you have specified the connection properties, you can check whether it is possible to establish connection to the database: click the **Test Connect** button for this purpose. If connection is successful, you will get the *'Connected!'* message; otherwise an error message will be returned.

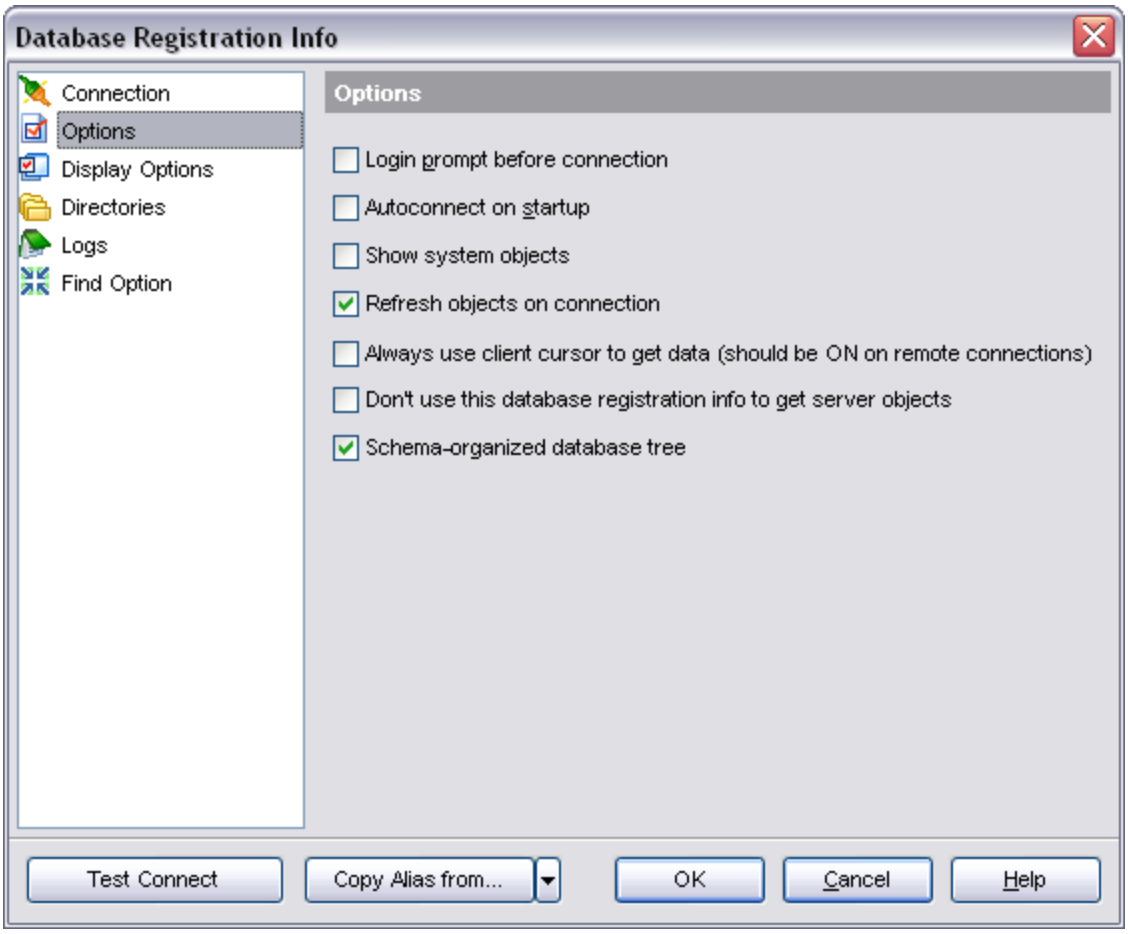
The **Copy Alias from...** menu allows you to select the alias of a previously registered database and use it for the newly created/configured database.

See also:
Setting common database options
Setting display options
Setting default directories
Setting log options
Find Option

4.4.2 Setting common database options

The **Options** section of the **Database Registration Info** dialog allows you to set various options for the database.

© 2008 EMS Database Management Solutions



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

Login prompt before connection
Enables SQL Manager 2008 for SQL Server to prompt for user name and password each time you connect to the database.

Autoconnect at startup
Check this option to specify that SQL Manager 2008 for SQL Server automatically establishes connection to the registered database at application startup.

Show system objects
This option determines whether SQL Server system objects are displayed in the DB Explorer tree.

Refresh objects on connection
This option allows you to enable/disable refreshing objects on connection to the database. It is highly recommended to uncheck this option if your database contains many objects or if connection to the database is slow.

Always use client cursor to get data
This option sets cursor to client. This might be useful for remote server connections, e.g.

© 2008 EMS Database Management Solutions

upon connection via Internet.

Don't use this database registration info to get server objects
If this option is enabled, the registration info of this database will not be used to establish server connection and get server objects (when expanding the **Server Objects** node in DB Explorer).

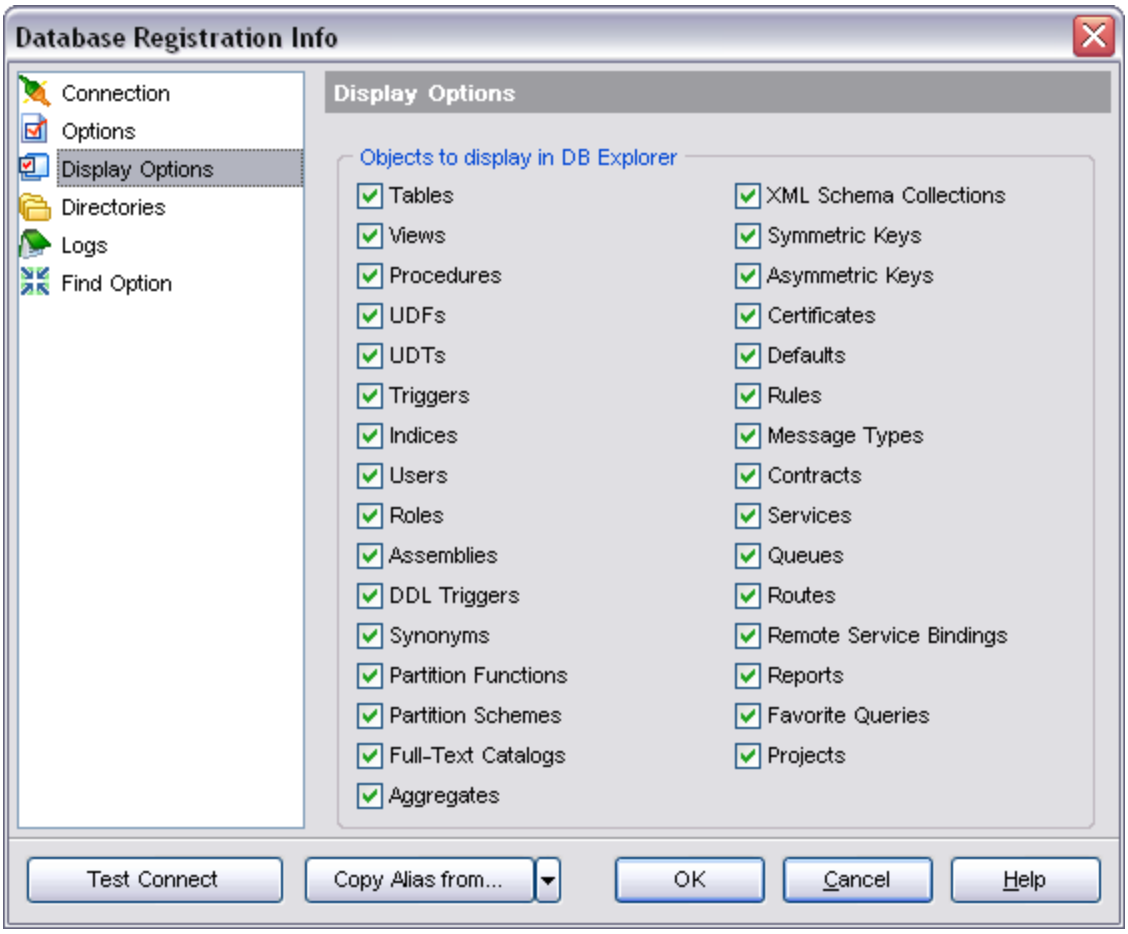
Schema-organized database tree
Disable this option to make the manager create the object tree without schema division.

See also:
Editing connection properties
Setting display options
Setting default directories
Setting log options
Find Option

4.4.3 Setting display options

The **Display Options** section of the **Database Registration Info** dialog allows you to specify which objects will be displayed in the Database Explorer tree.

© 2008 EMS Database Management Solutions



For your convenience the *Select All* and *Deselect All* functions are implemented in the context menu of the objects list area.

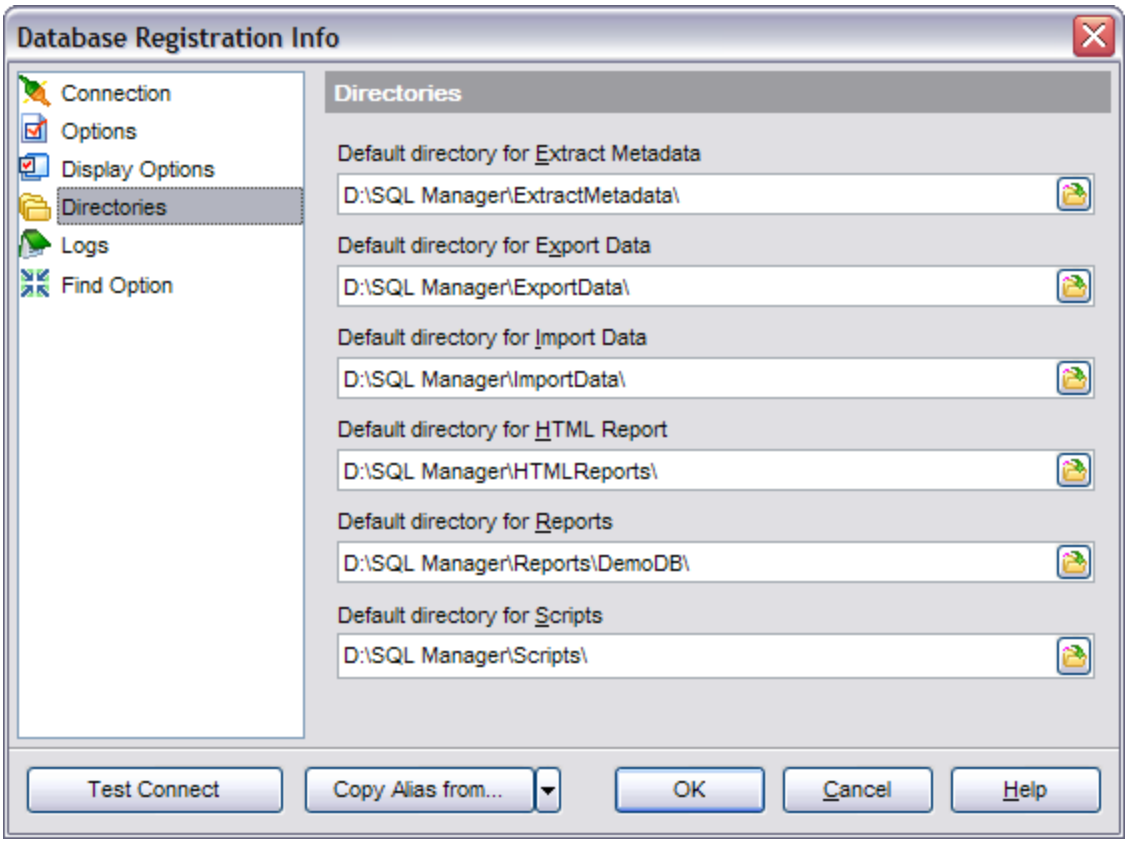
See also:

Editing connection properties
Setting common database options
Setting default directories
Setting log options
Find Option

4.4.4 Setting default directories

The **Directories** section of the **Database Registration Info** dialog allows you to set the directories to be used by default for database extract, data export, data import, saving HTML reports, creating reports operations.

© 2008 EMS Database Management Solutions



See also:
Editing connection properties
Setting common database options
Setting display options
Setting log options
Find Option

4.4.5 Setting log options

The **Logs** section of the **Database Registration Info** dialog allows you to specify log file names for metadata changes logging and SQL query logging, if necessary.

Logging can be useful when you are going to move the changes made in the development database to the production database.

Enable log of metadata changes
Check this option if you wish to log metadata changes of your database in a file.

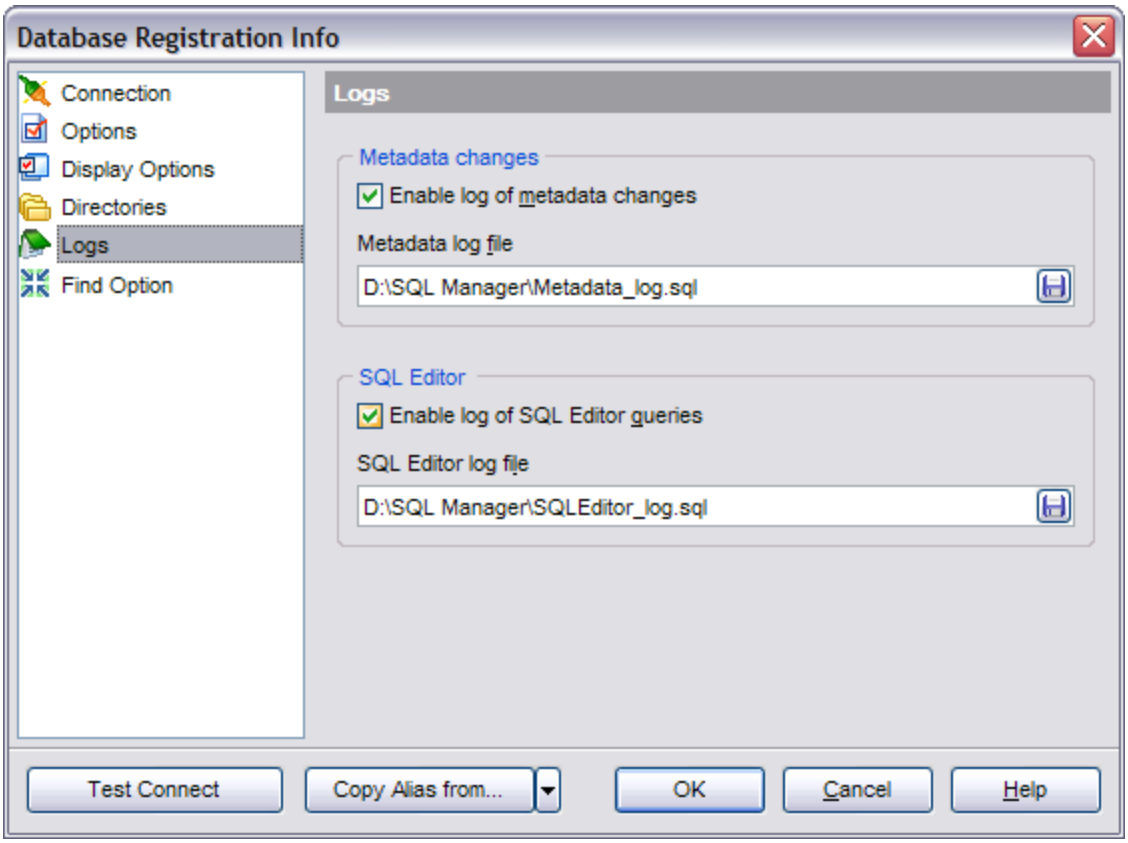
Metadata log file
This field is enabled if the **Enable log of metadata changes** option is selected. Type in or use the **Save as** button to specify the path to the **.sql* file to store the metadata

© 2008 EMS Database Management Solutions

logs.

Enable log of SQL Editor queries
Check this option if you wish to log your SQL Editor queries in a file.

SQL Editor log file
This field is enabled if the **Enable log of SQL Editor queries** option is selected. Type in or use the **Save as** button to specify the path to the **.sql* file to store the logs of SQL queries.



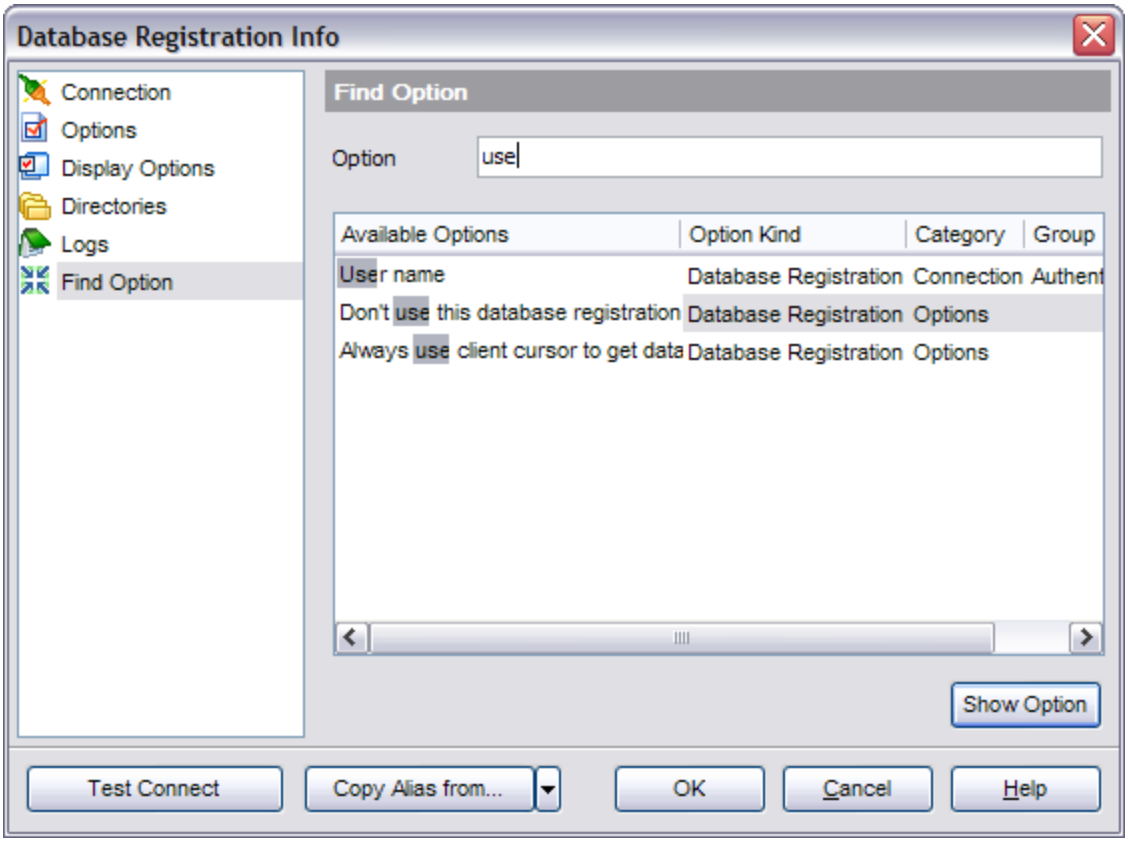
See also:
Editing connection properties
Setting common database options
Setting display options
Setting default directories
Find Option

4.4.6 Find Option

The **Find Option** section allows you to search for options available within the **Database Registration Info** dialog easily and quickly.

© 2008 EMS Database Management Solutions

Option
In this field you can enter the name of the option to search for within the database registration options.



The **Available options** area lists all options of the Database Registration category according to the specified name. Select the required option in the list and click **Show Option** to open the corresponding section where you can view/edit the value of this option.

See also:

Editing connection properties
Setting common database options
Setting display options
Setting default directories
Setting log options

© 2008 EMS Database Management Solutions

4.5 Database Properties

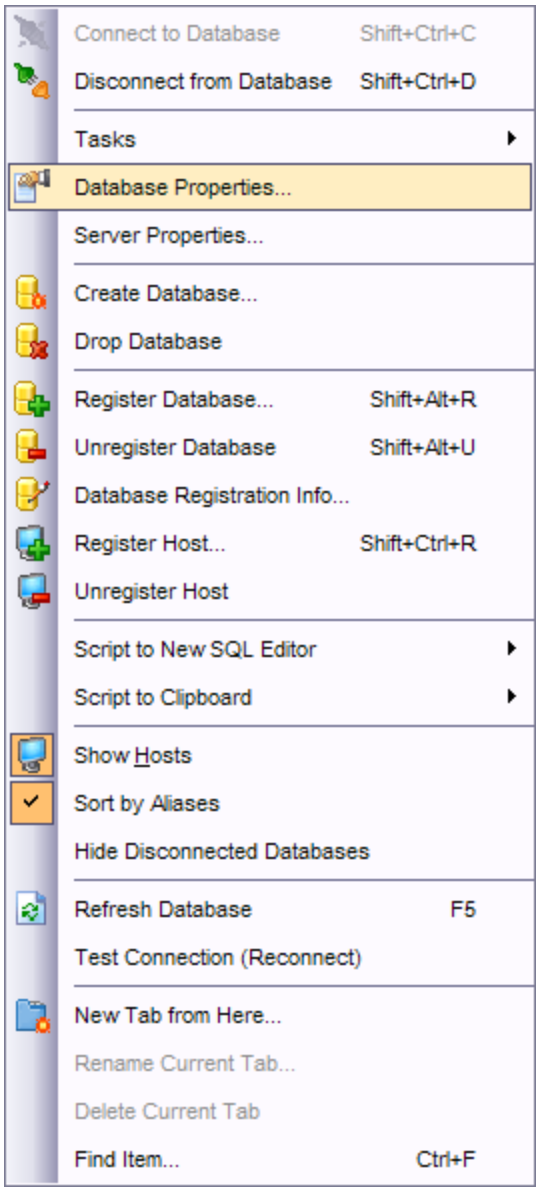
The **Database Properties** dialog allows you to view/edit a number of properties which can be changed to optimize database performance.

To open the dialog, right-click the Database in DB Explorer and select the **Database Properties...** context menu item, or use the **Properties** button on the main toolbar.

A number of database-level options that determine the characteristics of the database can be set for each database. Only the system administrator, database owner, members of the **sysadmin** and **dbcreator** fixed server roles and **db_owner** fixed database roles can modify these options. These options are unique to each database and do not affect other databases.

- Common database options
- Recovery options
- ANSI options
- SQL options
- Automation options
- Cursor options
- Advanced database options
- Data files
- Transaction log files
- Filegroups
- Find Option

© 2008 EMS Database Management Solutions



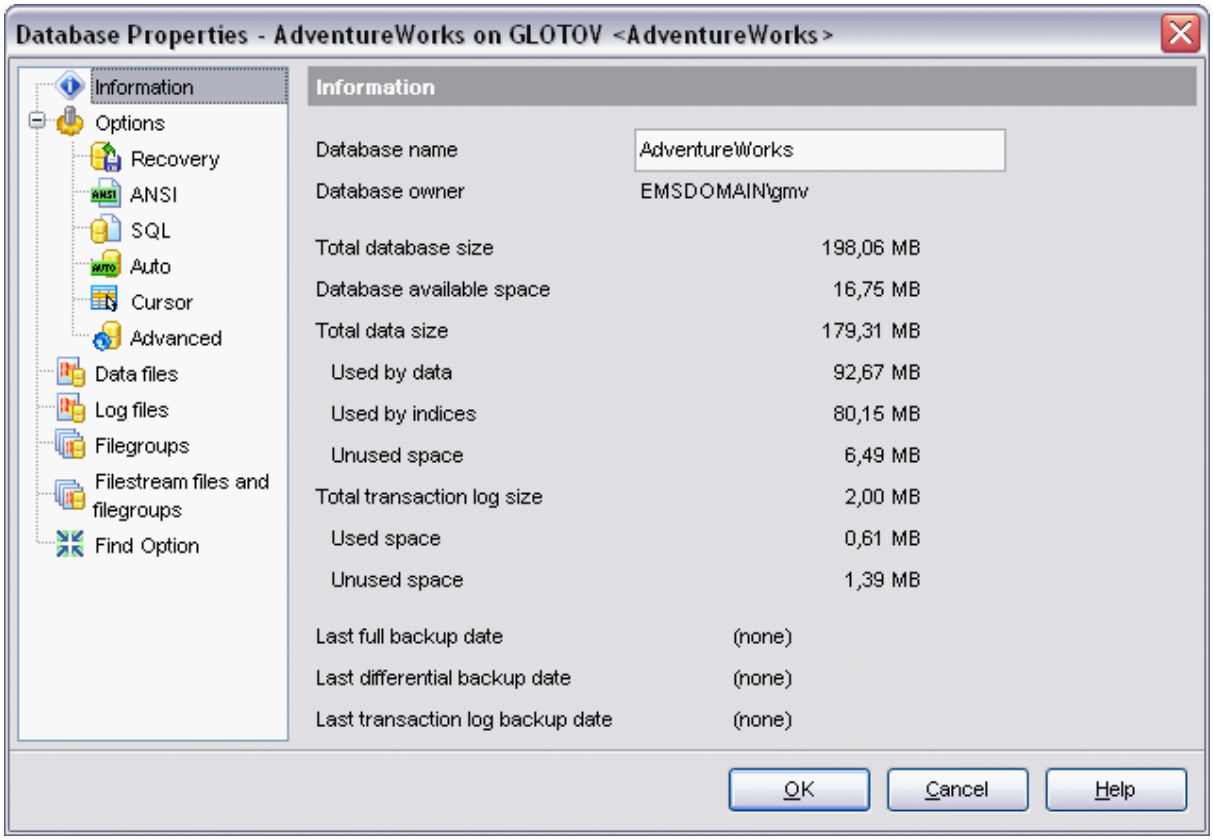
See also:

Register Host wizard
Create Database wizard
Register Database wizard
Database Registration Info
Server Properties

© 2008 EMS Database Management Solutions

4.5.1 Information

The **Information** section of the **Database Properties** dialog allows you to view the database summary: *Database name, Database owner, Total database size*, *Database available size*, *Total data size* (*used by data*, *used by indices*, *unused space*), *Total transaction log size* (*used space*, *unused space*), *Last full/differential/transaction log backup dates*.



See also:
Options
Data files
Log files
Filegroups
Find Option

4.5.2 Options

The **Options** section of the **Database Properties** dialog allows you to view/edit options pertaining to the database in groups: *General*, *Recovery*, *ANSI*, *SQL*, *Auto*, *Cursor*, *Advanced*.

- Recovery options

© 2008 EMS Database Management Solutions

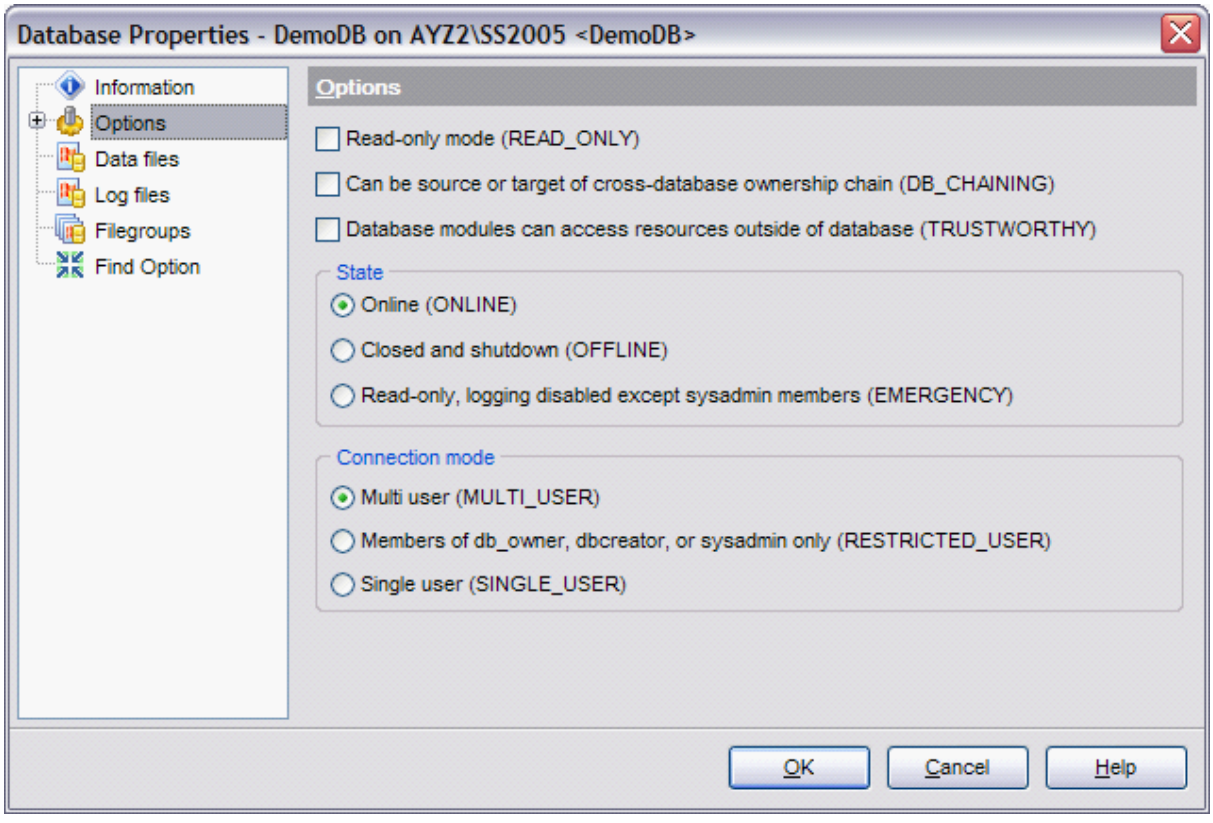
- ANSI options
- SQL options
- Automation options
- Cursor options
- Advanced

General options

Read-only mode (READ_ONLY)
When *READ_ONLY* is specified, the database is in read-only mode. Users can retrieve data from the database, but cannot modify the data, because a read-only database does not allow data modifications:
a. Automatic recovery is skipped at system startup.
b. Shrinking the database is not possible.
c. No locking takes place in read-only databases, which can result in faster query performance.

Can be source or target of cross-database ownership chain (DB_CHAINING)
When this option is checked, *DB_CHAINING* is enabled and the database can be the source or target of a cross-database ownership chain. When this option is unchecked, the database cannot participate in cross-database ownership chaining. You can change this option for user databases, but not for system databases.

Database modules can access resources outside of database (TRUSTWORTHY)
When this option is on, database modules (for example, views, user-defined functions, or stored procedures) that use an impersonation context are allowed to access resources outside of the database.



© 2008 EMS Database Management Solutions

State (ONLINE | OFFLINE | EMERGENCY)

When *ONLINE* is specified, the database is open and available for use (enabled by default).

When *OFFLINE* is specified, the database is closed and shutdown, and marked offline. The database cannot be modified while it is in the offline state.

When *EMERGENCY* is specified, the database is marked *READ_ONLY*, logging is disabled, and only members of the **sysadmin** fixed server role can access the database. EMERGENCY is primarily used for troubleshooting purposes. For example, a database marked as suspect can be set to the EMERGENCY state which allows the system administrator to copy the database contents elsewhere.

Connection mode (MULTI_USER | RESTRICTED_USER | SINGLE_USER)

MULTI_USER allows all users with the appropriate permissions to connect to the database (enabled by default).

RESTRICTED_USER allows only members of the **db_owner** fixed database role and **dbcreator** and **sysadmin** fixed server roles to connect to the database, but it does not limit their number. All connections to the database are terminated in the timeframe specified by the corresponding clause of the *ALTER DATABASE* statement. Once the database has transitioned to the *RESTRICTED_USER* state, connection attempts by unqualified users are refused.

SINGLE_USER allows one user at a time to connect to the database. All other user connections are broken. The timeframe for breaking the connection is controlled by the termination clause of the *ALTER DATABASE* statement. New connection attempts are refused. The database remains in the *SINGLE_USER* mode even if the user who set the option logs off. At that point, a different user (but only one) can connect to the database.

See also:

Information
Data files
Log files
Filegroups
Find Option

4.5.2.1 Recovery options

Recovery Model (RECOVERY FULL | BULK_LOGGED | SIMPLE)

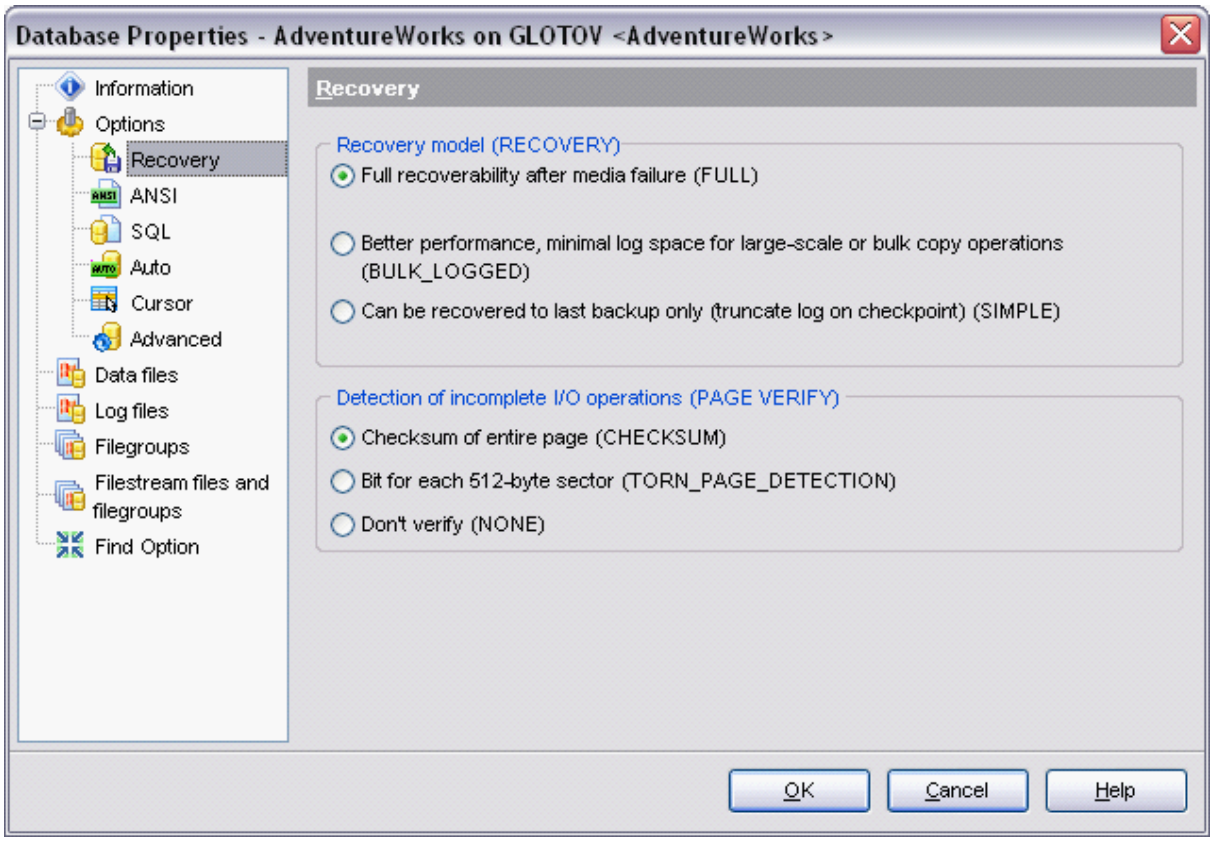
When *FULL* is specified, database backups and transaction log backups are used to provide full recoverability from media failure. All operations, including bulk operations such as *SELECT INTO*, *CREATE INDEX*, and bulk loading data, are fully logged.

When *BULK_LOGGED* is specified, logging for all *SELECT INTO*, *CREATE INDEX*, and bulk loading data operations is minimal and therefore requires less log space. In exchange for better performance and less log space usage, the risk of exposure to loss is greater than with full recovery.

When *SIMPLE* is specified, the database can be recovered only to the last full

© 2008 EMS Database Management Solutions

database backup or last differential backup.



Detection of Incomplete I/O Operations (PAGE_VERIFY CHECKSUM | TORN_PAGE_DETECTION | NONE)

PAGE_VERIFY specifies three options to discover incomplete I/O transactions caused by disk I/O errors. Disk I/O 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.

If *CHECKSUM* is specified, a checksum is taken over the contents of the entire page and stored in the page header when a page is written to disk. When a page is read from disk, the checksum is recomputed and compared to the checksum value stored in the page header. In SQL Server 2005, the database engine distinguishes between an I/O error detected by the operating system (error 823) and an I/O error detected by SQL Server *PAGE_VERIFY CHECKSUM* option (error 824). Should the values not match, error message 824 is reported to both the SQL Server error log and the NT Event Viewer.

If *TORN_PAGE_DETECTION* is specified, a bit is reversed for each 512-byte sector in the 8-kilobyte (KB) database page when the page is written to disk. If a bit is in the wrong state when the page is later read, the page was written incorrectly; a torn page is detected.

If *NONE* is specified, *PAGE_VERIFY* is set to OFF and future data page writes will not contain a *CHECKSUM* or *TORN_PAGE_DETECTION* bit and the page will not be verified at read time even if a *CHECKSUM* or *TORN_PAGE_DETECTION* bit is present.

© 2008 EMS Database Management Solutions

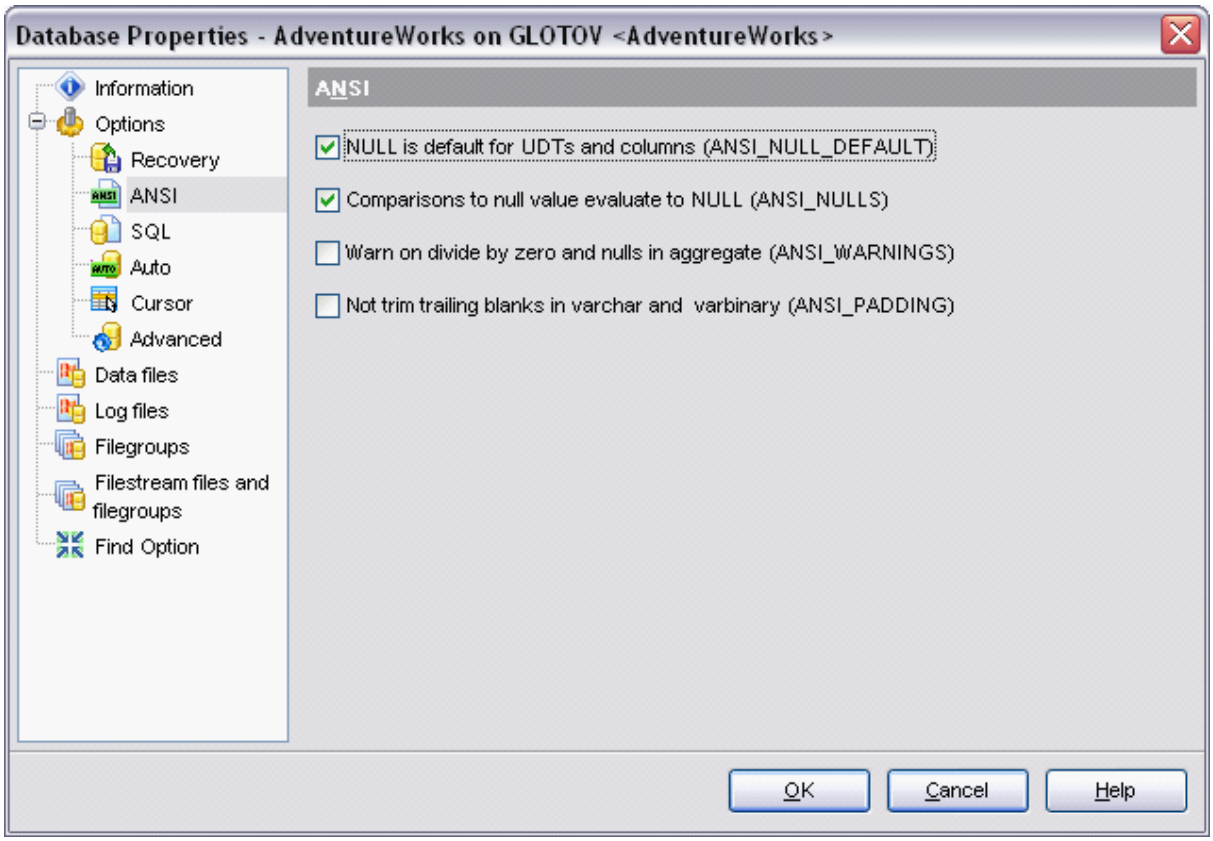
See also:
ANSI options
SQL options
Automation options
Cursor options
Advanced

4.5.2.2 ANSI options

NULL is default for UDTs and columns (ANSI_NULL_DEFAULT)
This option allows the user to control the database default nullability. When *NULL* or *NOT NULL* is not specified explicitly, a user-defined data type or a column definition uses the default setting for nullability.
Nullability is determined by session and database settings. When this option is enabled, all user-defined data types or columns that are not explicitly defined as *NOT NULL* during a *CREATE TABLE* or *ALTER TABLE* statement default to allowing null values. Columns that are defined with constraints follow constraint rules regardless of this setting.

Comparisons to null value evaluate to NULL (ANSI_NULLS)
When this option is enabled, all comparisons to a null value evaluate to *NULL* (unknown). When disabled, comparisons of non-Unicode values to a null value evaluate to *TRUE* if both values are *NULL*.

© 2008 EMS Database Management Solutions



Warn on divide by zero and nulls in aggregate (ANSI_WARNINGS)
When this option is enabled, errors or warnings are issued when conditions such as "divide by zero" occur or null values appear in aggregate functions. When disabled, no warnings are raised when null values appear in aggregate functions, and null values are returned when conditions such as "divide by zero" occur.
ANSI_WARNINGS must be set to ON when you create or manipulate indexes on computed columns or indexed views.

Not trim trailing blanks in varchar and varbinary (ANSI_PADDING)
When this option is enabled, trailing blanks in character values inserted into *varchar* or *nvarchar* columns and trailing zeros in binary values inserted into *varbinary* columns are not trimmed. Values are not padded to the length of the column. When disabled, the trailing blanks (for *varchar* or *nvarchar*) and zeros (for *varbinary*) are trimmed. This setting affects only the definition of new columns.
Char(n) and *binary(n)* columns that allow nulls are padded to the length of the column when *ANSI_PADDING* is set to ON, but trailing blanks and zeros are trimmed when *SET ANSI_PADDING* is OFF. *Char(n)* and *binary(n)* columns that do not allow nulls are always padded to the length of the column.

See also:
Recovery options
SQL options

© 2008 EMS Database Management Solutions

Automation options
Cursor options
Advanced

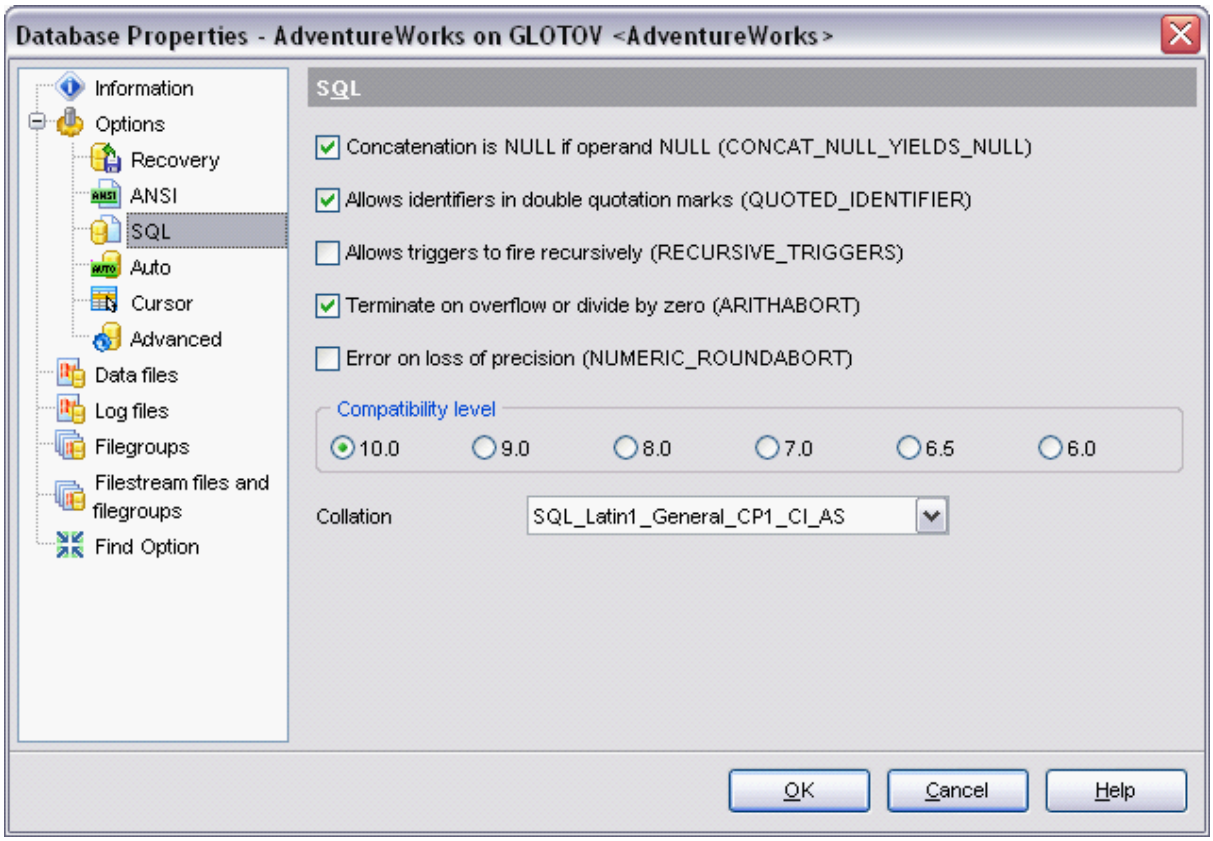
4.5.2.3 SQL options

Concatenation is NULL if operand NULL (CONCAT_NULL_YIELDS_NULL)
When this option is enabled, if one of the operands in a concatenation operation is *NULL*, the result of the operation is *NULL*. For example, concatenating the character string "This is" and *NULL* results in the value *NULL*, rather than the value "This is". When disabled, concatenating a null value with a character string yields the character string as the result; the null value is treated as an empty character string.

Allows identifiers in double quotation marks (QUOTED_IDENTIFIER)
When this option is enabled, identifiers can be delimited by double quotation marks and literals must be delimited by single quotation marks. All strings delimited by double quotation marks are interpreted as object identifiers. Quoted identifiers do not have to follow the Transact-SQL rules for identifiers. They can be keywords and can include characters not generally allowed in Transact-SQL identifiers. If a single quotation mark (') is part of the literal string, it can be represented by double quotation marks ("). When disabled, 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.

Allow triggers to fire recursively (RECURSIVE_TRIGGERS)
When this option is enabled, triggers are allowed to fire recursively. When disabled, triggers cannot be fired recursively.

© 2008 EMS Database Management Solutions



Terminate on overflow or divide by zero (ARITHABORT)
When this option is enabled, an overflow or divide-by-zero error causes the query or batch to terminate. If the error occurs in a transaction, the transaction is rolled back. When disabled, a warning message is displayed if one of these errors occurs, but the query, batch, or transaction continues to process as if no error occurred.

Error on loss of precision (NUMERIC_ROUNDABORT)
If this option is enabled, an error is generated when loss of precision occurs in an expression. When disabled, losses of precision do not generate error messages and the result is rounded to the precision of the column or variable storing the result.

The **Compatibility Level** option group controls the behavior of an instance of Microsoft® SQL Server™, setting behavior to match a specified version of Microsoft SQL Server. Possible values are: *9.0*, *8.0*, *7.0*, *6.5*, *6.0*.

Collation can be either a Windows collation name or a SQL collation name. By default, the server collation is used.

See also:

Recovery options
ANSI options
Automation options

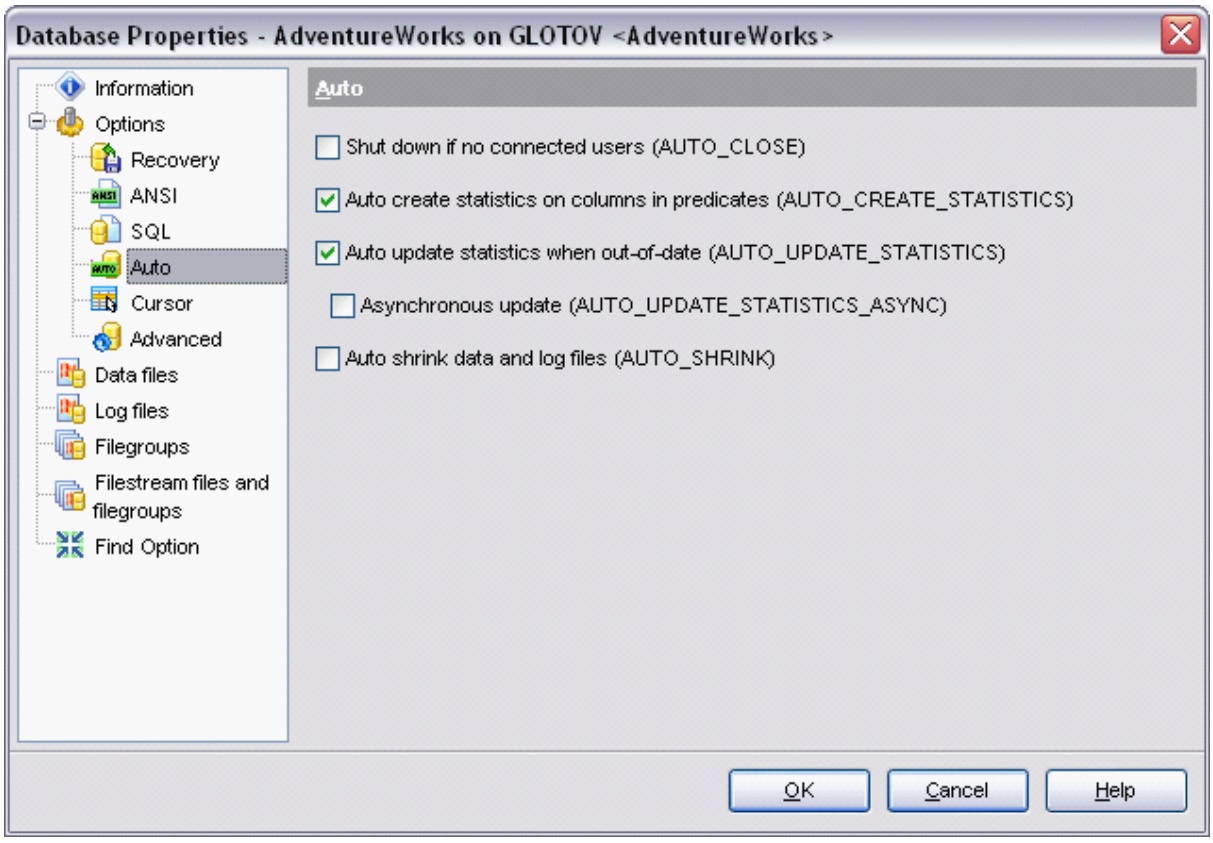
© 2008 EMS Database Management Solutions

Cursor options
Advanced

4.5.2.4 Automation options

Shut down if no connected users (AUTO_CLOSE)
When this option is enabled, the database is closed and shut down when the last user of the database exits and all processes in the database complete, thereby freeing any resources. The database reopens automatically when a user tries to use the database again. When disabled, the database remains open even if no users are currently using the database.

Auto create statistics on columns in predicates (AUTO_CREATE_STATISTICS)
When this option is enabled, statistics are automatically created on columns used in a predicate. Adding statistics improves query performance because the SQL Server query optimizer can better determine how to evaluate a query. If the statistics are not used, SQL Server automatically deletes them. When disabled, statistics are not automatically created; instead, statistics can be manually created.



Auto update statistics when out-of-date (AUTO_UPDATE_STATISTICS)
When this option is enabled, existing statistics are automatically updated when the statistics become out-of-date because the data in the tables have been changed. When disabled, existing statistics are not automatically updated; instead, statistics

© 2008 EMS Database Management Solutions

can be manually updated.

Asynchronous update (AUTO_UPDATE_STATISTICS_ASYNC)
When this option is enabled, queries do not wait for the statistics to be updated before compiling. Instead, the out-of-date statistics are put on a queue for updating by a worker thread in a background process. The query and any other concurrent queries compile immediately by using the existing out-of-date statistics. Because there is no delay for updated statistics, query response times are predictable; however, the out-of-date statistics may cause the query optimizer to choose a less-efficient query plan. Queries that start after the updated statistics are ready will use those statistics. This may cause the recompilation of cached plans that depend on the older statistics version.

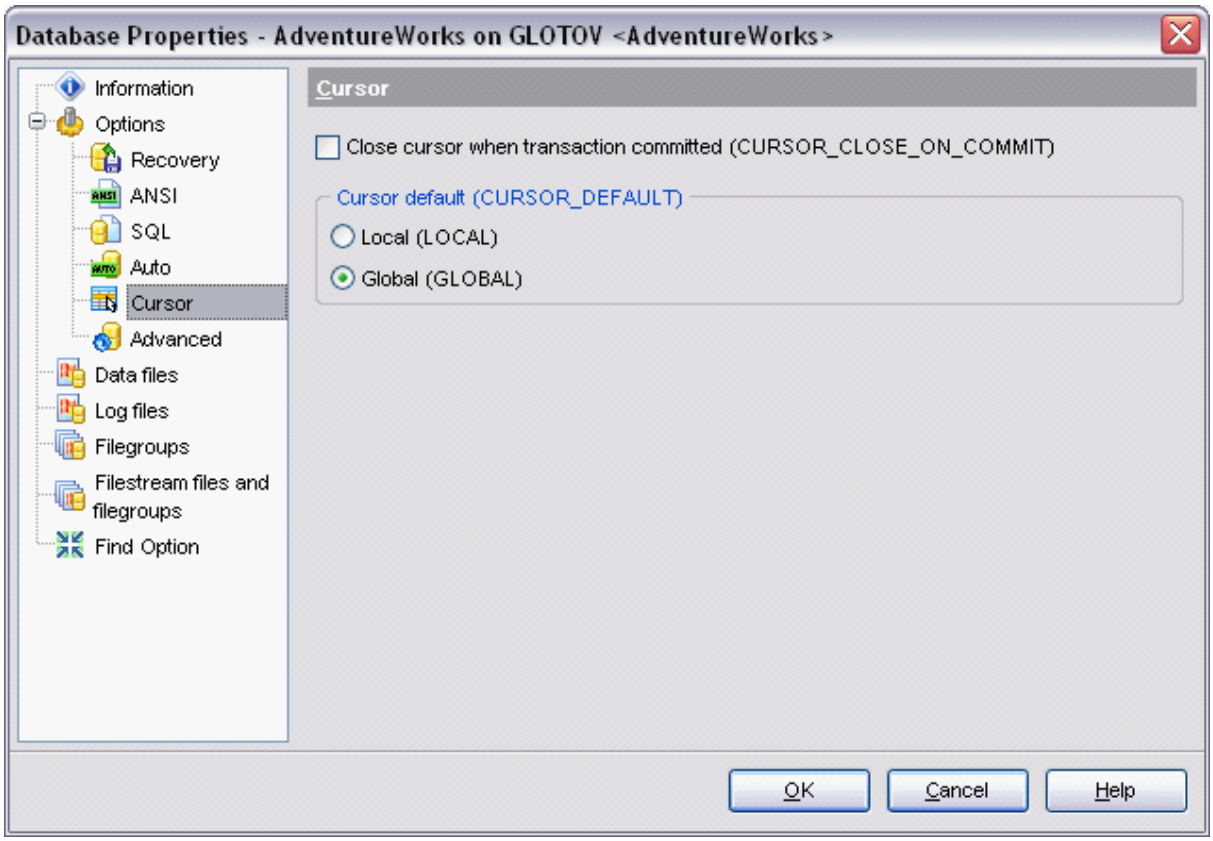
Auto shrink data and log files (AUTO_SHRINK)
When this option is enabled, the database files are candidates for periodic shrinking. Both data files and log files can be shrunk automatically by SQL Server. *AUTO_SHRINK* reduces the size of the transaction log only if the database is set to *SIMPLE* recovery model or if the log is backed up. When disabled, the database files are not automatically shrunk during periodic checks for unused space.

See also:
Recovery options
ANSI options
SQL options
Cursor options
Advanced

4.5.2.5 Cursor options

Close cursor when transaction committed (CURSOR_CLOSE_ON_COMMIT)
When this option is enabled, any open cursors are closed automatically (in compliance with SQL-92) when a transaction is committed. When disabled, cursors remain open across transaction boundaries, closing only when the connection is closed or when they are explicitly closed.

© 2008 EMS Database Management Solutions



Cursor default (CURSOR_DEFAULT LOCAL | GLOBAL)

When *LOCAL* is set, and a cursor is not defined as *GLOBAL*, the scope of the cursor is local to the batch, stored procedure, or trigger in which the cursor was created. The cursor name is valid only within this scope. The cursor can be referenced by local cursor variables in the batch, stored procedure, or trigger, or a stored procedure OUTPUT parameter. The cursor is implicitly deallocated when the batch, stored procedure, or trigger terminates, unless it was passed back in an OUTPUT parameter. If it is passed back in an OUTPUT parameter, the cursor is deallocated when the last variable referencing it is deallocated or goes out of scope.

See also:

Recovery options
ANSI options
SQL options
Automation options
Advanced

© 2008 EMS Database Management Solutions

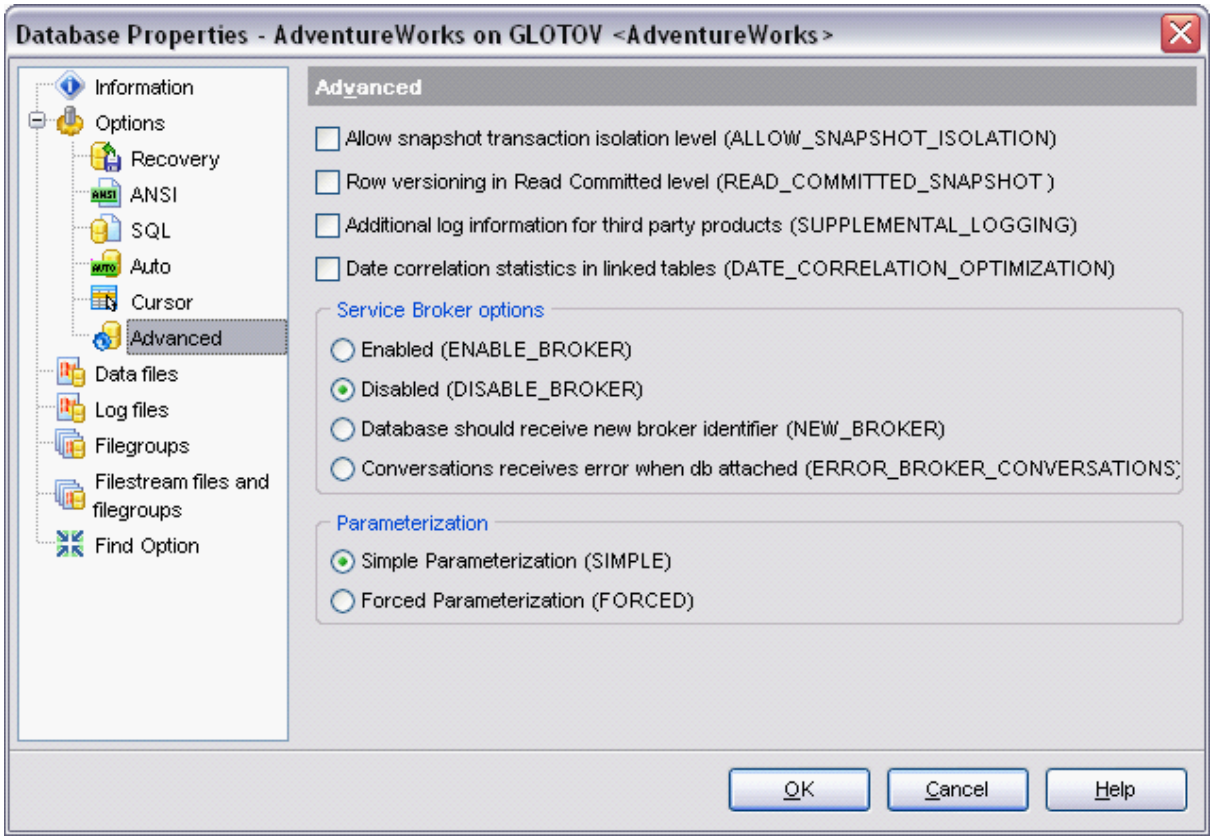
4.5.2.6 Advanced

Allow snapshot transaction isolation level (ALLOW_SNAPSHOT_ISOLATION)
If this option is enabled, SQL Server will execute snapshot transactions against the specified database when transactions are explicitly set to the *SNAPSHOT* transaction isolation level.

Row versioning in Read Comitted level (READ_COMMITTED_SNAPSHOT)
If this option is enabled, transactions setting the read-committed isolation level use row versioning instead of locking.

Additional log information for third party products (SUPPLEMENTAL_LOGGING)
When this option is enabled, additional information is added to the log for third-party products. Setting this option on will increase log space usage and may impact performance.

Date correlation statistics in linked tables (DATE_CORRELATION_OPTIMIZATION)
If this option is enabled, SQL Server maintains correlation statistics between any two tables in the database that have *DATETIME* columns and are linked by a one-column foreign key constraint.



Service Broker options

ENABLE_BROKER specifies that Service Broker is enabled for the specified database. The *IS_BROKER_ENABLED* flag is set to TRUE in the sys.databases catalog view and message delivery is started.

© 2008 EMS Database Management Solutions

- *DISABLE_BROKER* specifies that Service Broker is disabled for the specified database. The *IS_BROKER_ENABLED* flag is set to FALSE in the sys.databases catalog view and message delivery is stopped.
- *NEW_BROKER* specifies that the database should receive a new broker identifier. Because the database is considered to be a completely new service broker, all existing conversations in the database are immediately removed without producing end dialog messages.
- *ERROR_BROKER_CONVERSATIONS* specifies that conversations in the database should receive an error message when the database is attached. This allows your applications to perform normal clean up for existing conversations.

Parametrization

- *Simple Parametrization*: queries are parameterized on the basis of the default behavior of the database.
- *Forced Parametrization*: SQL Server parameterizes all queries in the database.

See also:

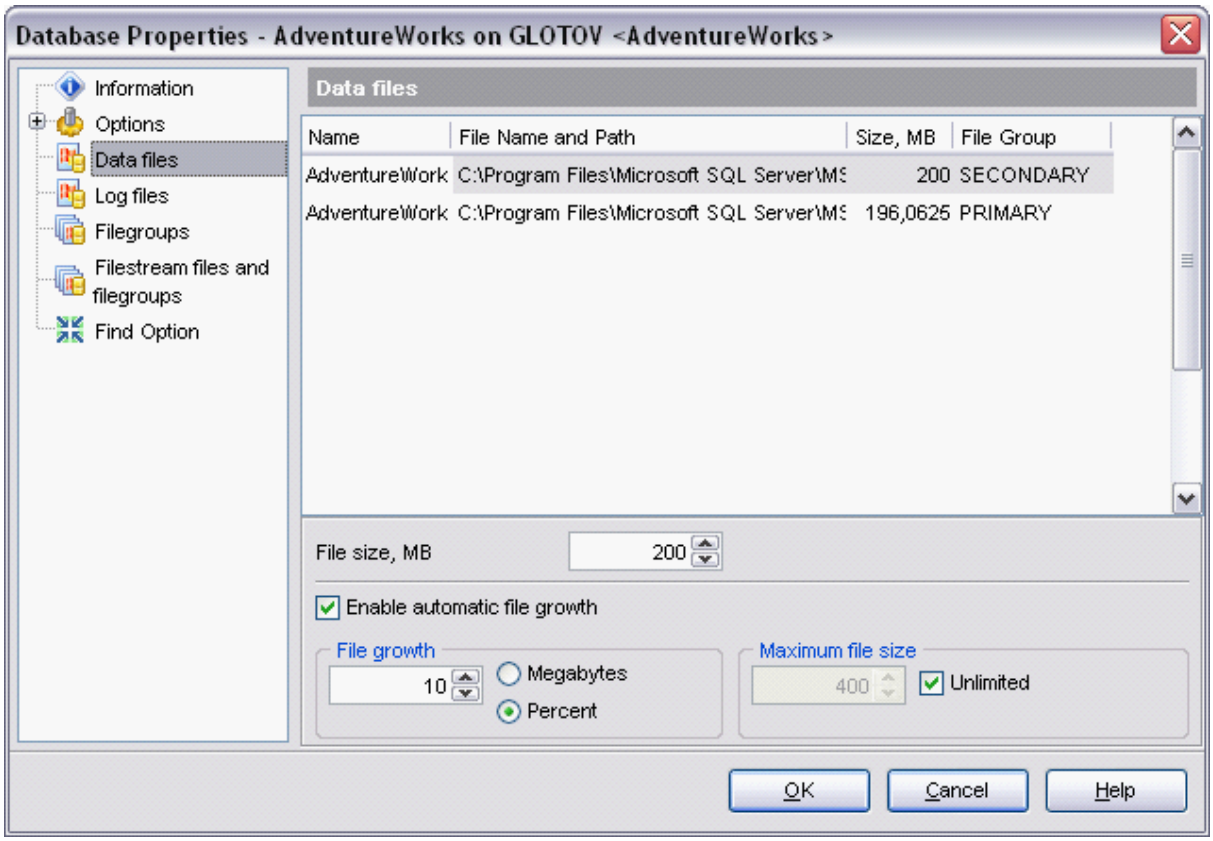
Recovery options
ANSI options
SQL options
Automation options
Cursor options

4.5.3 Data files

The **Data files** section of the **Database Properties** dialog allows you to specify that the disk files used to store the data portions of the database (data files) are defined explicitly, and fill the list of file items defining the data files for the primary file group.

To add a file, right-click in the main working area and select the **Add file** popup menu item.
To remove a file, right-click the file in the list and select the **Delete file** popup menu item.

© 2008 EMS Database Management Solutions



The **Name** is a logical name used to refer to the physical file in all Transact-SQL statements. The logical file name must conform to the rules for SQL Server identifiers and must be unique within the database.

The **File Name and Path** is the name of the physical file including the directory path. The path must follow the rules for operating system file names.

File Group
The *Primary* file group contains the primary data file and any other files not specifically assigned to another file group. All pages for the system tables are allocated in the primary file group. *User-defined* file groups are any other file groups specified by the user.

File size
SQL Server 2005 files can grow automatically from their originally specified size. When you define a file, you can specify a **file growth** increment. Each time the file fills, it increases its size by the growth increment. If there are multiple files in a file group, they do not autogrow until all the files are full. Growth then occurs using a round-robin algorithm.

Each file can also have **maximum file size** specified. If ☑ **Unlimited** is specified, the file can continue to grow until it has used all available space on the disk.

See also:

© 2008 EMS Database Management Solutions

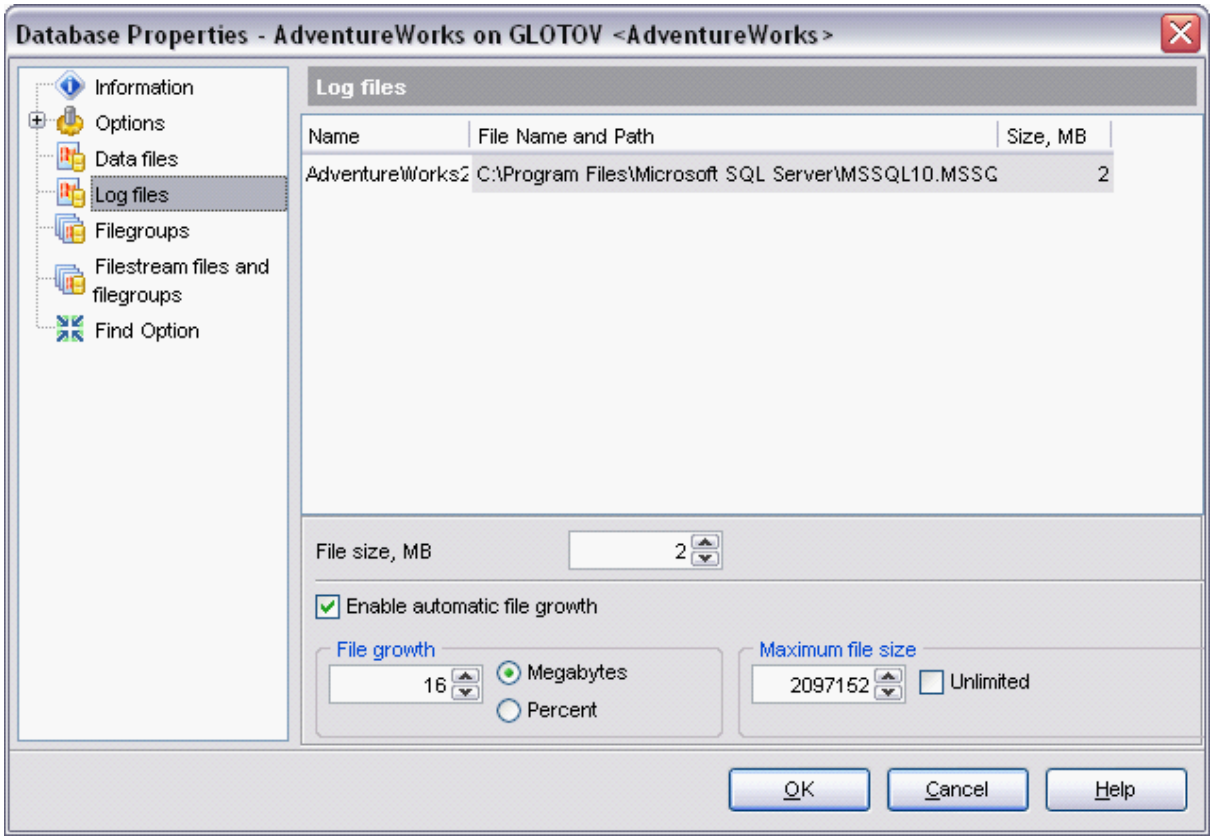
Information
Options
Log files
Filegroups
Find Option

4.5.4 Log files

The **Log files** section of the **Database Properties** dialog allows you to manage **transaction log files** for the database.

Log files hold all of the log information used to recover the database. There must be at least one log file for each database, although there can be more than one. The recommended file name extension for log files is **.ldf*.

To add a file, right-click in the main working area and select the **Add file** popup menu item.
To remove a file, right-click the file in the list and select the **Delete file** popup menu item.



The **Name** is a logical name used to refer to the physical file in all Transact-SQL statements. The logical file name must conform to the rules for SQL Server identifiers and

© 2008 EMS Database Management Solutions

must be unique within the database.

The **File Name and Path** is the name of the physical file including the directory path. The path must follow the rules for operating system file names.

File size
SQL Server 2005 files can grow automatically from their originally specified size. When you define a file, you can specify a **file growth** increment. Each time the file fills, it increases its size by the growth increment. If there are multiple files in a file group, they do not autogrow until all the files are full. Growth then occurs using a round-robin algorithm.

Each file can also have **maximum file size** specified. If **Unlimited** is specified, the file can continue to grow until it has used all available space on the disk.

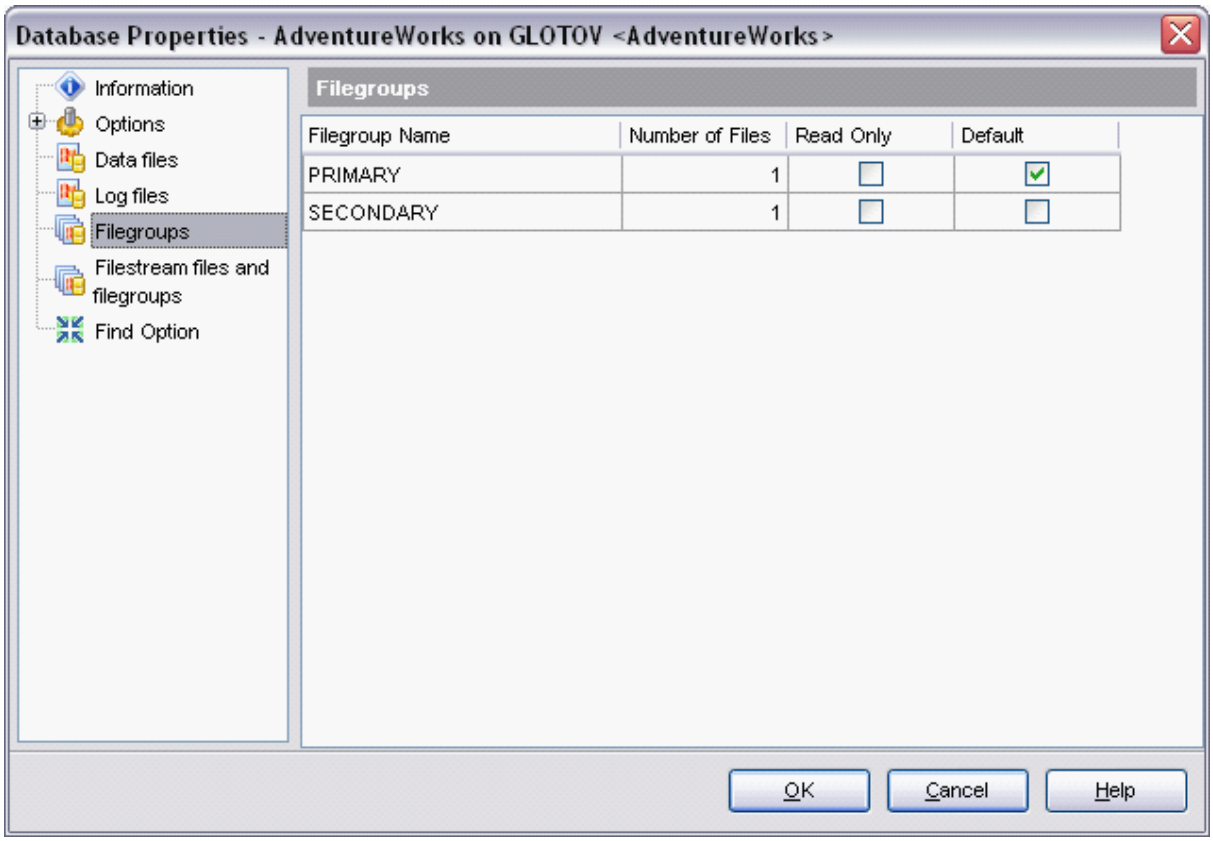
See also:
Information
Options
Data files
Filegroups
Find Option

4.5.5 Filegroups

The **Filegroups** section of the **Database Properties** dialog allows you to manage **filegroups** for the database.

Each database has a *PRIMARY* filegroup which contains the primary data file and any secondary files that are not put into other filegroups. User-defined filegroups can be created to group data files together for administrative and data allocation/placement purposes.

© 2008 EMS Database Management Solutions



All data files are stored in filegroups:

- *PRIMARY*: the filegroup that contains the primary file. All system tables are allocated to the primary filegroup.
- *User-defined*: any filegroup that is specifically created by the user when first creating or later altering the database.

Read Only

Specifies that the filegroup is read-only. Updates to objects that reside in this filegroup are not allowed. The *PRIMARY* filegroup cannot be marked read-only.

Default

When objects are created in the database without specifying to which filegroup they belong, they are assigned to the default filegroup. At any time, exactly one filegroup is designated as the default filegroup. The files in the default filegroup must be large enough to hold any new objects not allocated to other filegroups. Initially, the *PRIMARY* filegroup is the default filegroup.

See also:

Information
Options
Data files
Log files

© 2008 EMS Database Management Solutions

Find Option

4.5.6 Filestream files and filegroups

The **Filestream files and filegroups** section of the **Database Properties** dialog allows you to manage **filestream filegroups** for the database.

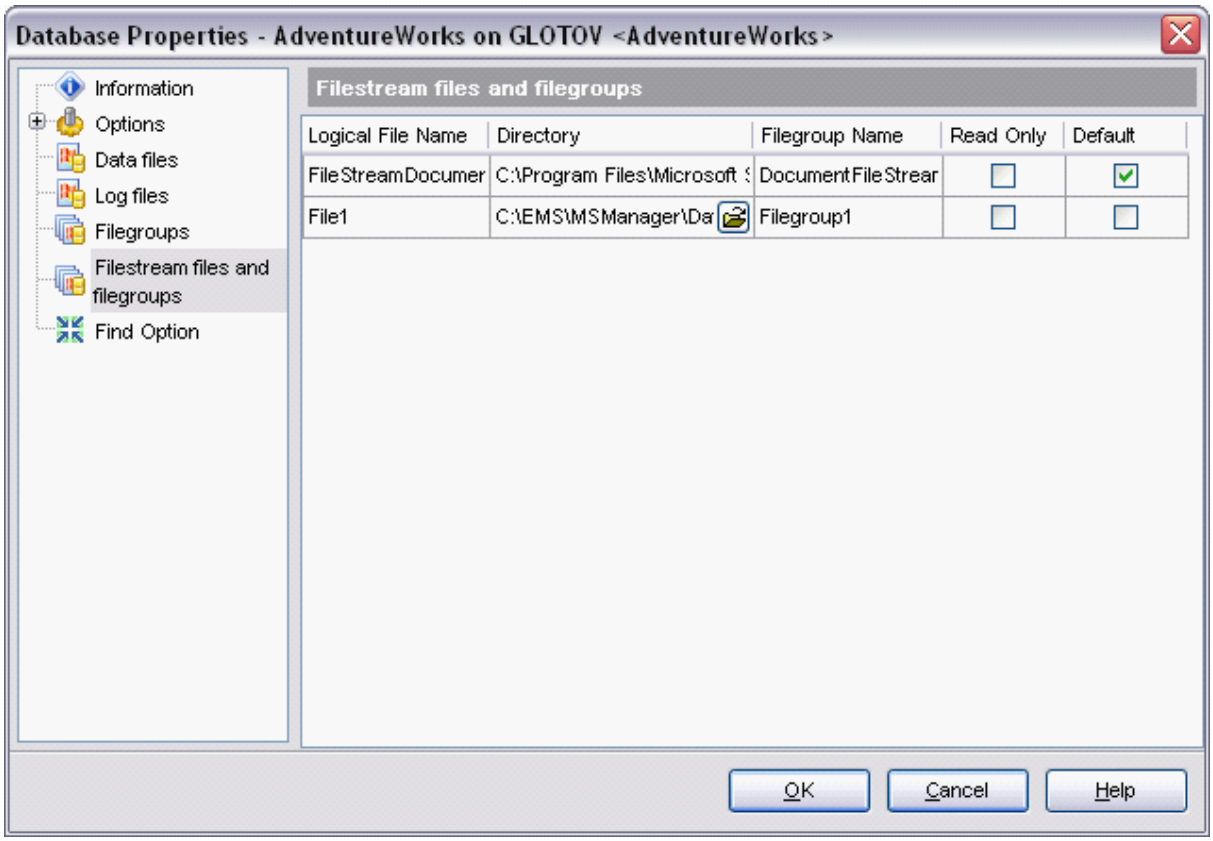
Filestream is used to store the varbinary(max) data in the file system instead of the tables. This storage type is limited only by the volume size of the file system.

Use the context menu to add a new filestream or to delete the selected one.

To create the new **Filestream filegroup** you need to specify:

Filegroup name
The name of the filestream filegroup.
Logical file name
The name of the logical file storing the data. This filename is not the name of the physical file.
Directory
The physical location of the data. The directory must be located on the server on which the database is registered.

© 2008 EMS Database Management Solutions



Read Only
Specifies that the filestream filegroup is read-only. Updates to objects that reside in this filestream filegroup are not allowed. The defalult fiestream filegroup can't be read-only.

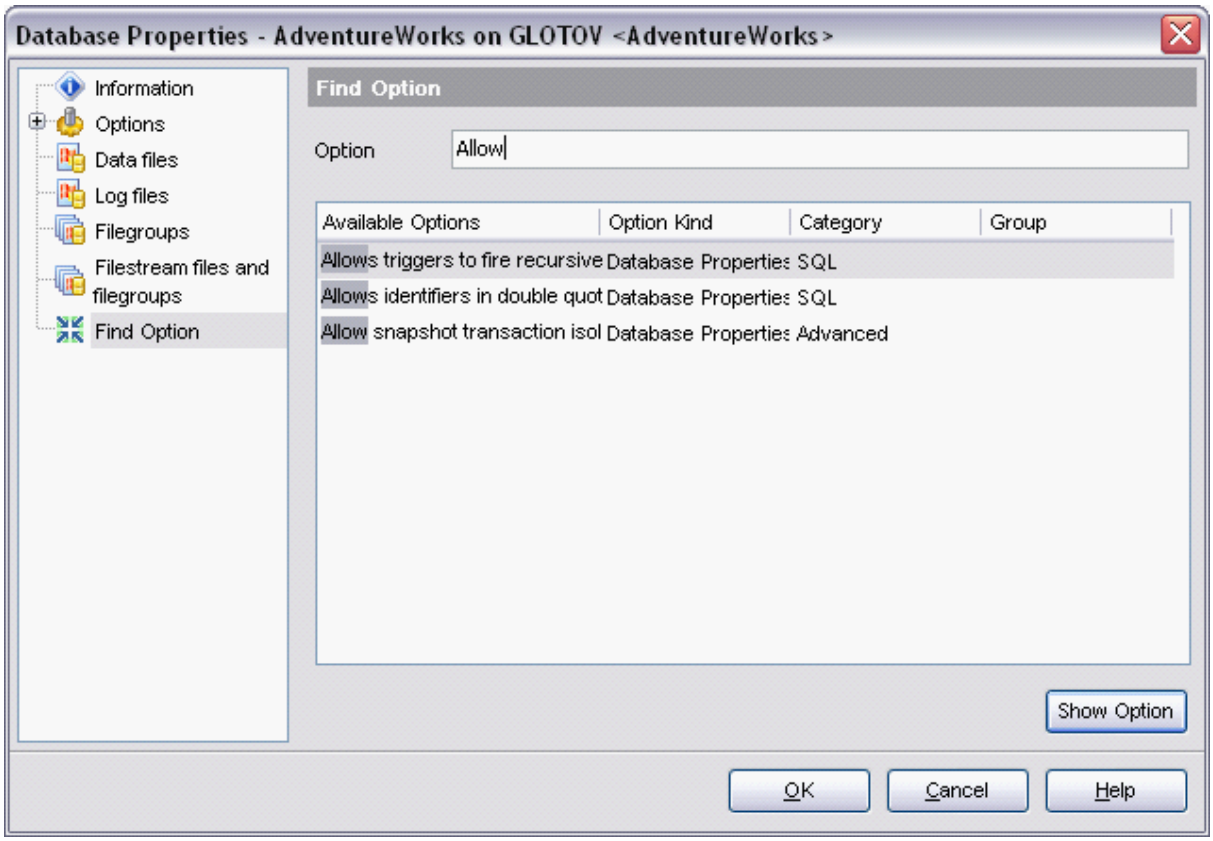
Default
The filestream fields data will be stored in this filestream filegroup if none was specified for the table.

4.5.7 Find Option

The **Find Option** section allows you to search for options available within the **Database Properties** dialog easily and quickly.

Option
In this field you can enter the name of the option to search for within the database properties.

© 2008 EMS Database Management Solutions



The **Available options** area lists all options of the Database Properties category according to the specified name. Select the required option in the list and click **Show Option** to open the corresponding section where you can view/edit the value of this option.

See also:

Information
Options
Data files
Log files
Filegroups

© 2008 EMS Database Management Solutions

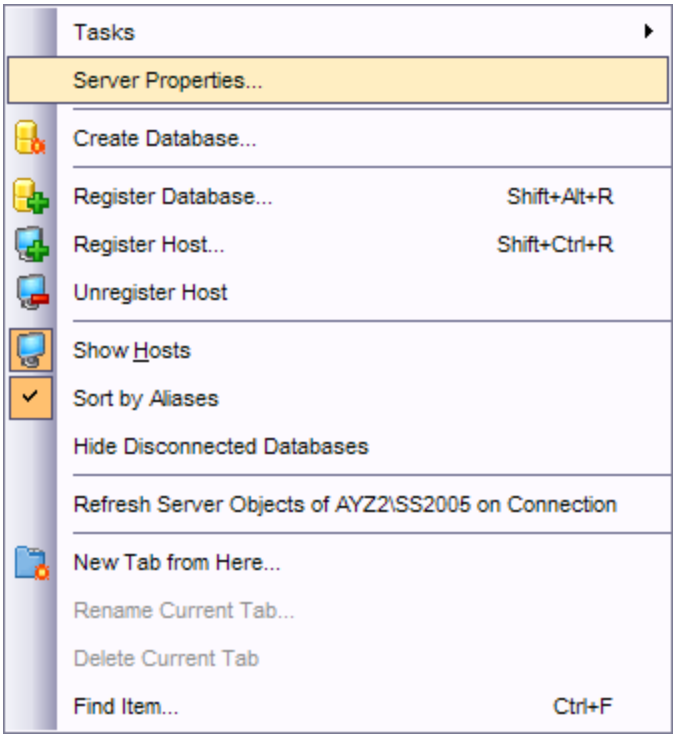
4.6 Server Properties

The **Server Properties** dialog allows you to view/edit a number of server options which can be changed to optimize Microsoft® SQL Server performance.

- General options
- Memory options
- Processor options
- Security options
- Active Directory
- Connection settings
- Database Settings
- Misc Server Settings
- Find Option

Configured values
This option displays the configured values for options on this page. If you change these values, select **Running Values** to see whether the changes have taken effect. If they have not, the instance of SQL Server must be restarted first.

Running values
This option displays the currently running values for options on this page. These values are read-only.



See also:
Register Host wizard

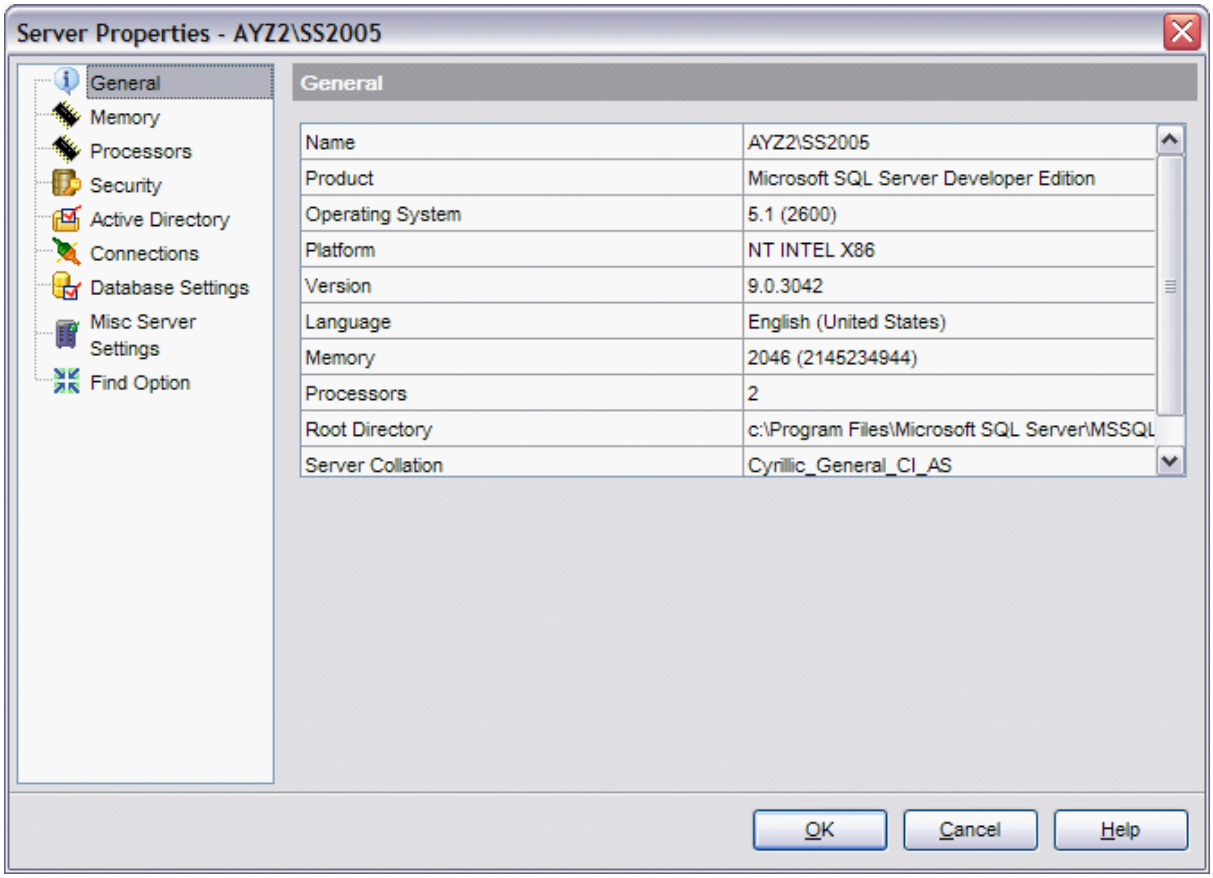
© 2008 EMS Database Management Solutions

Create Database wizard
Register Database wizard
Database Registration Info
Database Properties

4.6.1 General options

The **General** section of the **Server Properties** dialog allows you to take a brief look at the general server configuration properties: *Name*, *Product*, *Operating System*, *Platform*, *Version*, *Language*, *Memory*, *Processors*, *Root Directory*, *Server Collation*.

These properties are read-only.



See also:
Memory options
Processor options
Security options
Active Directory
Connection settings

© 2008 EMS Database Management Solutions

Database Settings
Misc Server Settings
Find Option

4.6.2 Memory options

The **Memory** section of the **Server Properties** dialog allows you to configure the instance of Microsoft® SQL Server by setting options pertaining to memory usage in groups: *Server memory options*, *Other memory options*.

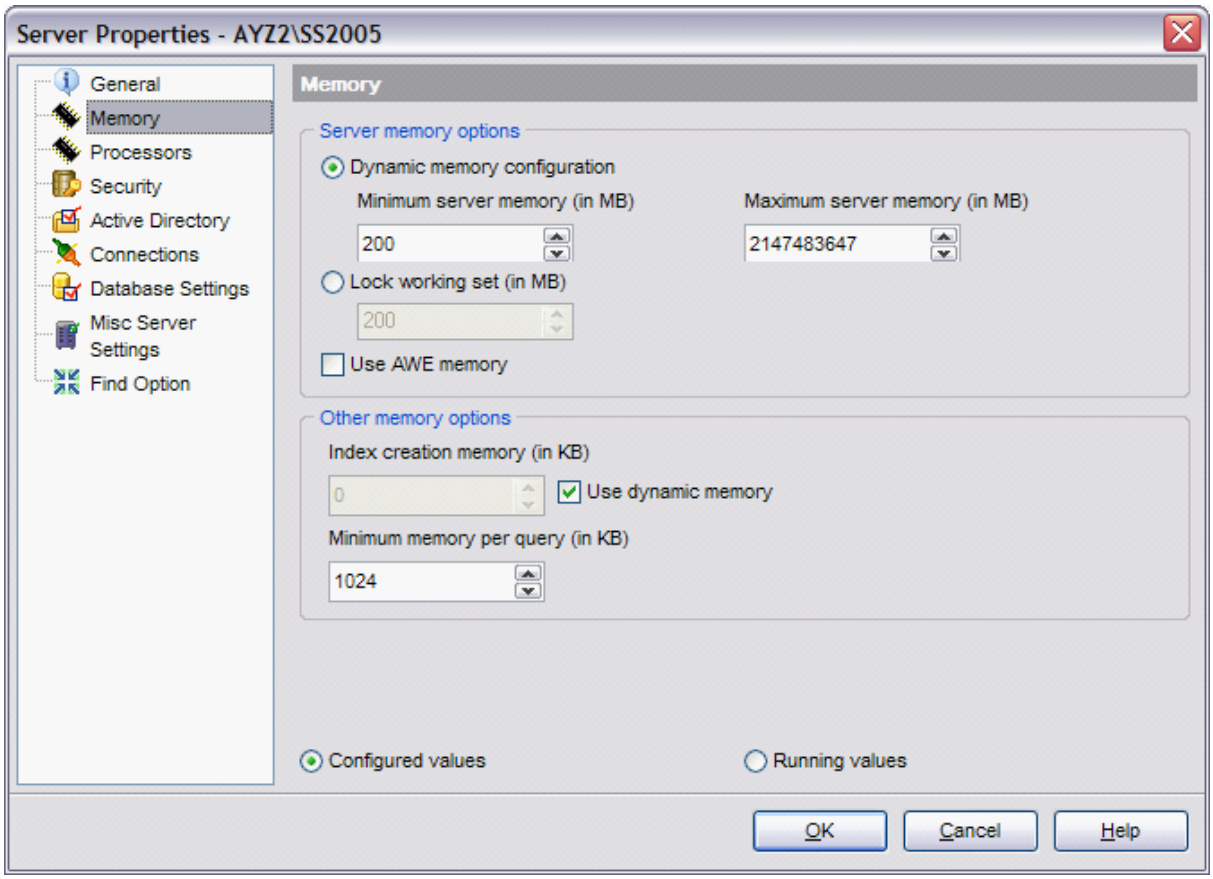
Server memory options

Dynamic memory configuration
Use the two server memory options, *Minimum server memory* and *Maximum server memory*, to reconfigure the amount of memory (in megabytes) in the buffer pool used by the instance of SQL Server.
By default, SQL Server can change its memory requirements dynamically on the basis of available system resources. The default setting for *Minimum server memory* is *0*, and the default setting for *Maximum server memory* is *2147483647*. The minimum amount of memory that can be specified for *Maximum server memory* is *16 MB*.

Locking working set
Allowing SQL Server to use memory dynamically is recommended; however, you can set the memory options manually and restrict the amount of memory that SQL Server can access.

Use AWE memory
Under Windows Server 2003, SQL Server 2005 can use Address Windowing Extensions (AWE) memory to further assist in balancing its own memory requirements with those of the operating system. This balancing between SQL Server and the operating system is subject to the constraints of the *Minimum server memory* and *Maximum server memory* options.

© 2008 EMS Database Management Solutions



Other memory options

In SQL Server 2000/2005 the **Index creation memory** option controls the maximum initial amount of memory allocated for index creation. If additional amount of memory is required for index creation operation, and it is available, the server will use additional memory thus exceeding the setting of this option. If additional memory is not available, the memory that has been already allocated will be used for index creation.

Use the **Minimum memory per query** option to specify the minimum amount of memory (in kilobytes) that will be allocated for execution of a query. For example, if **Minimum memory per query** is set to *2,048 KB*, the query is guaranteed to get at least that much total memory. You can set **Minimum memory per query** to any value from *512* up to *2,147,483,647 KB (appr. 2 GB)*. The default value is *1,024 KB*.

See also:

General options
Processor options
Security options
Active Directory
Connection settings

© 2008 EMS Database Management Solutions

Database Settings
Misc Server Settings
Find Option

4.6.3 Processor options

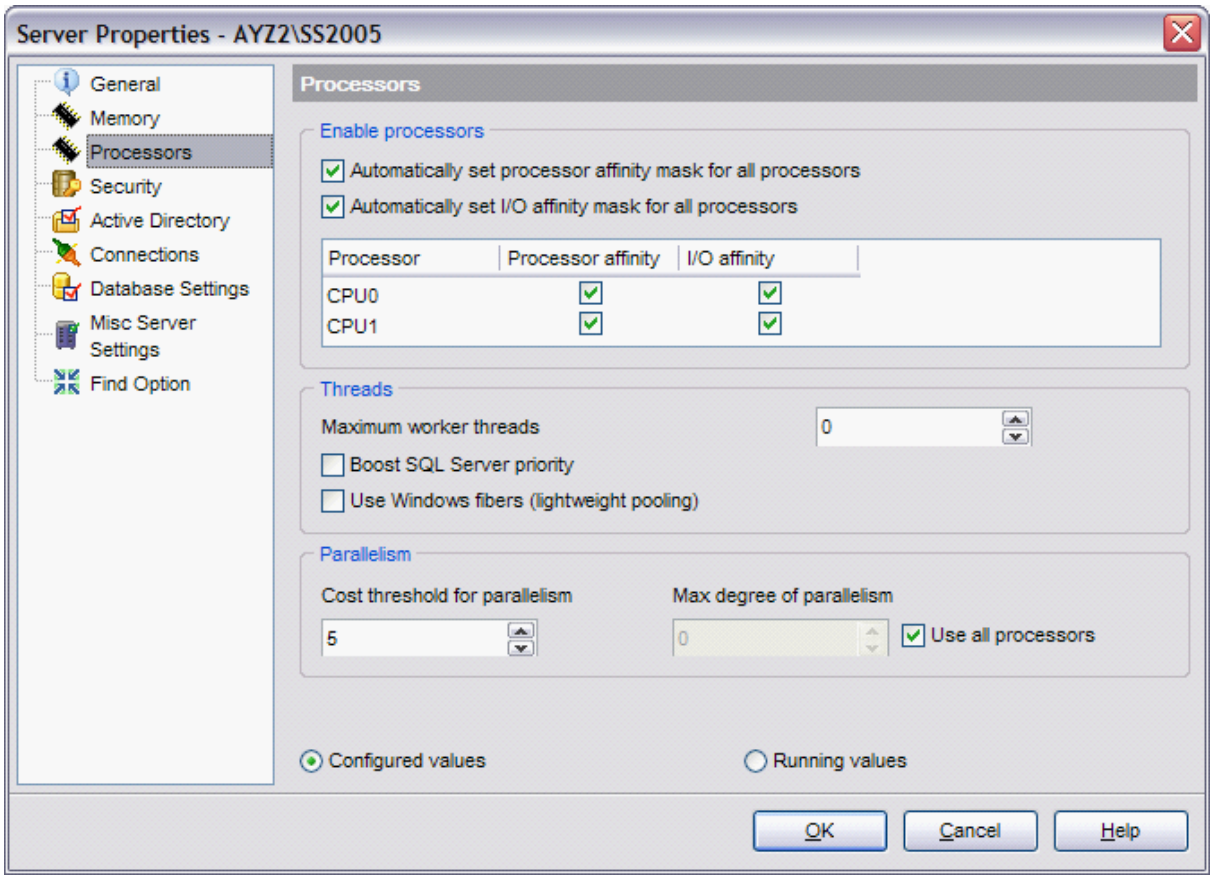
The **Processors** section of the **Server Properties** dialog allows you to configure the instance of Microsoft® SQL Server by setting options pertaining to processors usage in groups: *Enable processors*, *Threads*, *Parallelism*.

Enable processors

To carry out multitasking, it is possible to distribute process threads among different processors. Although efficient from an operating system point of view, this activity can reduce SQL Server performance under heavy system loads, as each processor cache is repeatedly reloaded with data. Assigning processors to specific threads can improve performance under these conditions by eliminating processor reloads and reducing thread migration across processors (thereby reducing context switching); such an association between a thread and a processor is called *processor affinity*.

You can enable any or both two affinity mask options supported by SQL Server 2005: **Automatically set processor affinity mask** (also known as **CPU affinity mask**) and **Automatically set I/O affinity mask for all processors**.

© 2008 EMS Database Management Solutions



Threads

Use the **Maximum worker threads** option to configure the number of working threads available to Microsoft® SQL Server processes. SQL Server uses the native thread services of the Microsoft® Windows 2000 and Windows Server 2003 operating systems so that one or more threads support each network that SQL Server supports simultaneously, another thread handles database checkpoints, and a pool of threads handles all users.

Use the **Boost SQL Server priority** option to specify whether SQL Server should run at a higher Windows 2000 or Windows 2003 scheduling priority than other processes on the same computer. If you set this option to 1, SQL Server runs at a priority base of 13 in the Windows 2000 or Windows Server 2003 scheduler. The default is 0, which is a priority base of 7.

Select the **Use Windows fibers (lightweight pooling)** option to provide a means of reducing the system overhead associated with the excessive context switching sometimes seen in symmetric multiprocessing (SMP) environments. When excessive context switching is present, lightweight pooling can provide better throughput by performing the context switching inline, thus helping to reduce user/kernel ring transitions.

Parallelism

Use the **Cost threshold for parallelism** option to specify the threshold at which SQL

© 2008 EMS Database Management Solutions

Server creates and runs parallel plans for queries. SQL Server creates and runs a parallel plan for a query only when the estimated cost to run a serial plan for the same query is higher than the value set in **Cost threshold for parallelism**. The cost refers to an estimated elapsed time in seconds required to run the serial plan on a specific hardware configuration. Only set **Cost threshold for parallelism** on symmetric multiprocessors.

When SQL Server 2005 runs on a computer with more than one CPU, it detects the best degree of parallelism for each instance of a parallel query execution. You can use the **Max degree of parallelism** option to limit the number of processors to use in parallel plan execution. The default value of *0* indicates that all available processors are used. Set **Max degree of parallelism** to *1* to suppress parallel plan generation. Set the value to a number greater than *1* (up to a maximum of *64*) to restrict the maximum number of processors used by a single query execution. If a value greater than the number of available processors is specified, or in case the **Use all processors** option is selected, the actual number of available processors is used. If the computer has only one processor, the **max degree of parallelism** value is ignored.

See also:

General options
Memory options
Security options
Active Directory
Connection settings
Database Settings
Misc Server Settings
Find Option

4.6.4 Security options

The **Security** section of the **Server Properties** dialog allows you to configure the instance of Microsoft® SQL Server by setting options pertaining to server security in groups: *Server Authentication*, *Login auditing*, *Server Proxy Account*, *Other security options*.

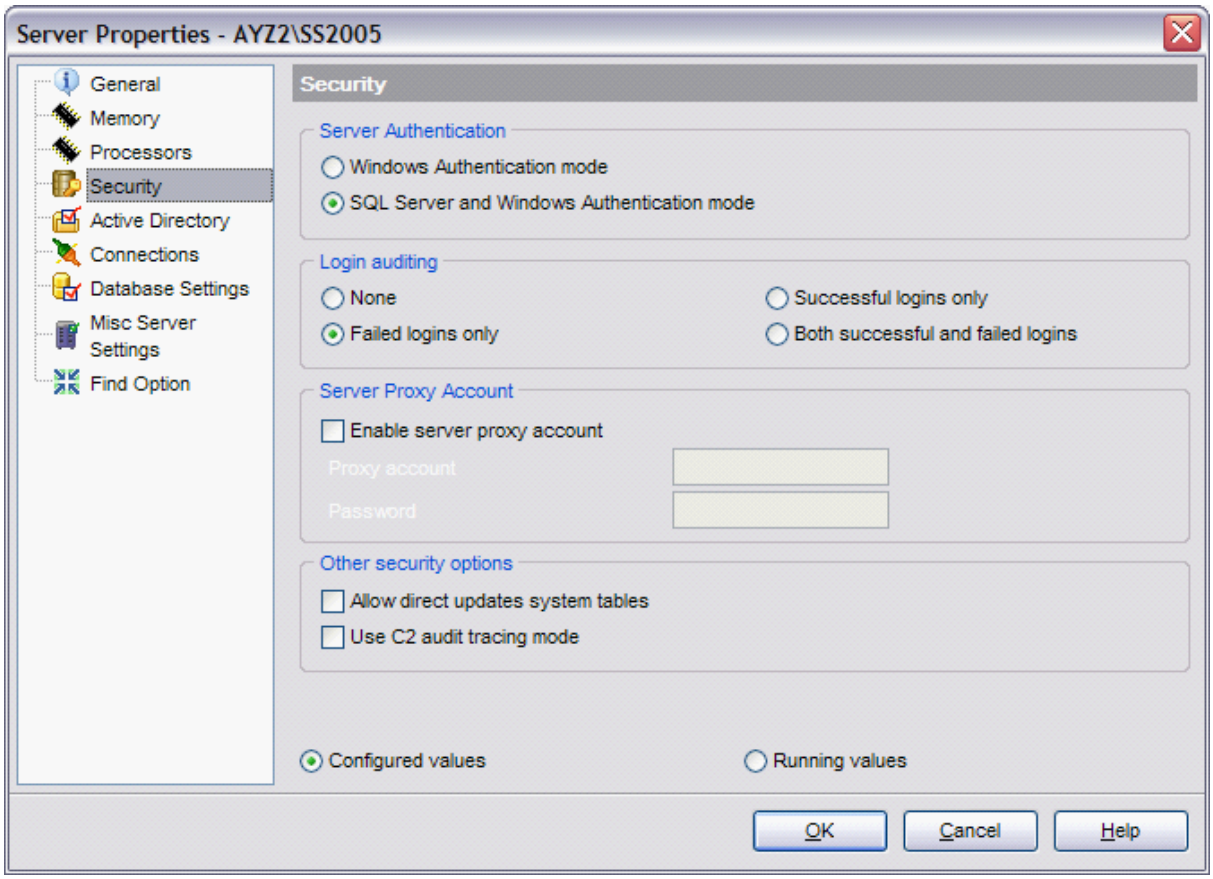
Server Authentication

Windows Authentication mode
When a user connects through a Microsoft® Windows user account, SQL Server validates the account name and password using information in the Windows operating system. This is the default authentication mode, and is much more secure than the mixed mode. Windows Authentication utilizes Kerberos security protocol, provides password policy enforcement in terms of complexity validation for strong passwords, provides support for account lockout, and supports password expiration.

SQL Server and Windows Authentication mode
This option enables the mixed mode and allows users to connect using either Windows Authentication or SQL Server Authentication. Users who connect through a Windows user

© 2008 EMS Database Management Solutions

account can make use of trusted connections that are validated by Windows.



The **Login auditing** property exposes SQL Server Authentication logging behavior:

- *None* (does not log authentication attempts)
- *Successful logins only* (logs successful authentication)
- *Failed logins only* (logs failed authentication)
- *Both successful and failed logins* (logs all authentication attempts regardless of success or failure)

Server Proxy account

This option allows SQL Server users that do not belong to the *sysadmin* fixed server role to execute *xp_cmdshell* commands and to own SQL Server jobs. To get access to this feature, you must enter **Proxy account** name and **Password**.

Note that this option is only available when you use MS SQL Server 2005. The login used by the server proxy account should have the least privileges required to perform the intended work. Excessive privileges for the proxy account could be used by a malicious user to compromise your system security.

Other security options

© 2008 EMS Database Management Solutions

Allow direct updates system tables
Turn this option on to modify system tables directly.
Note: System tables should not be changed directly by any user. It is not recommended to modify system tables with *DELETE*, *UPDATE*, *INSERT* statements or user-defined triggers.

Use C2 audit tracing mode
Selecting this option will configure the server to record both failed and successful attempts to access statements and objects. This information can help you profile system activity and track possible security policy violations.

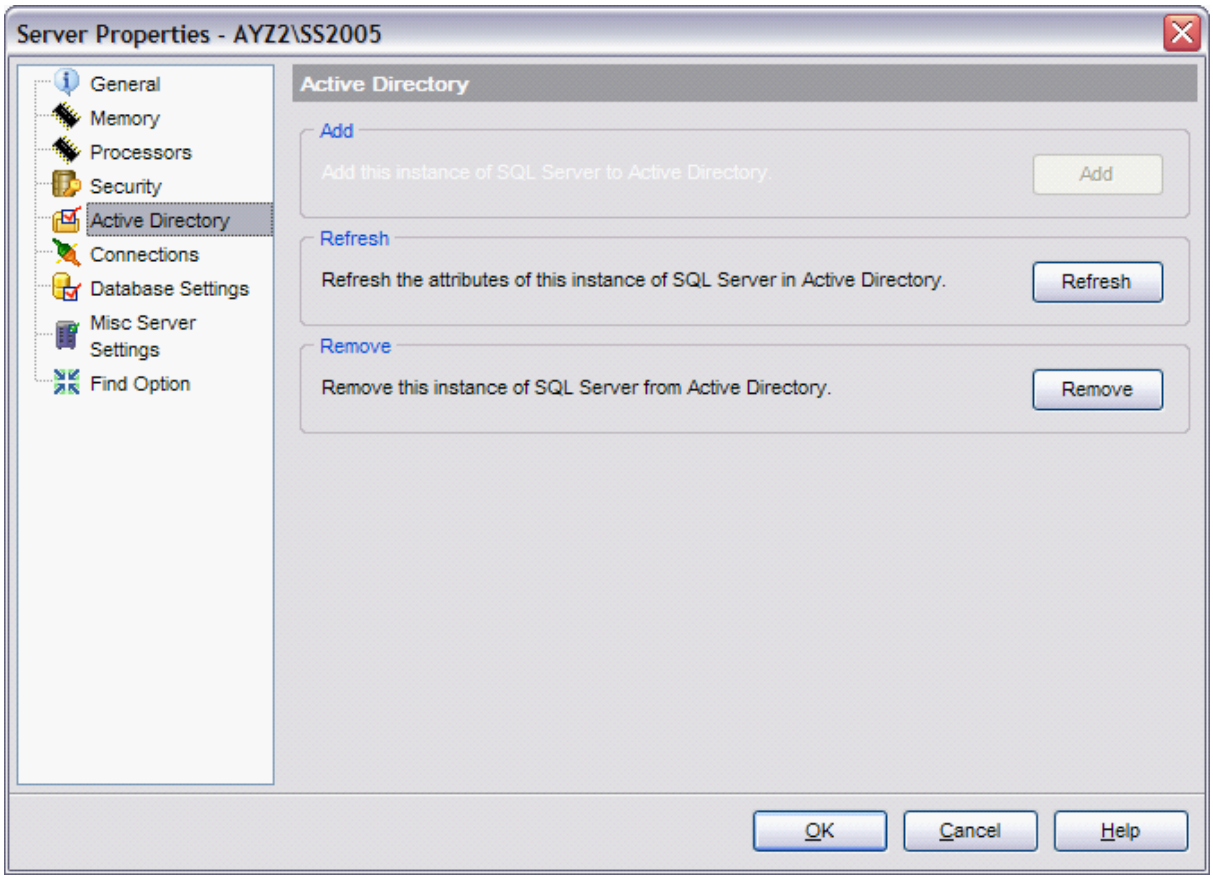
See also:
General options
Memory options
Processor options
Active Directory
Connection settings
Database Settings
Misc Server Settings
Find Option

4.6.5 Active Directory

The **Active Directory** section of the **Server Properties** dialog allows you to configure the instance of Microsoft® SQL Server by managing its Active Directory: *Add*, *Refresh*, *Remove*.

Add SQL Server to Microsoft Active Directory when you want users to be able to use Active Directory to find this instance of SQL Server. This may be useful in large organizations with many instances of SQL Server.

© 2008 EMS Database Management Solutions



Add

Add this instance of SQL Server to Active Directory.
Connects to Active Directory and registers the instance of Microsoft® SQL Server in the Active directory.

Refresh

Refresh the attributes of this instance in Active Directory.
Updates Active Directory with the current Microsoft® SQL Server properties.

Remove

Remove this instance of SQL Server from Active Directory.
Unregisters the instance of Microsoft® SQL Server from the Active Directory.

See also:
General options
Memory options

© 2008 EMS Database Management Solutions

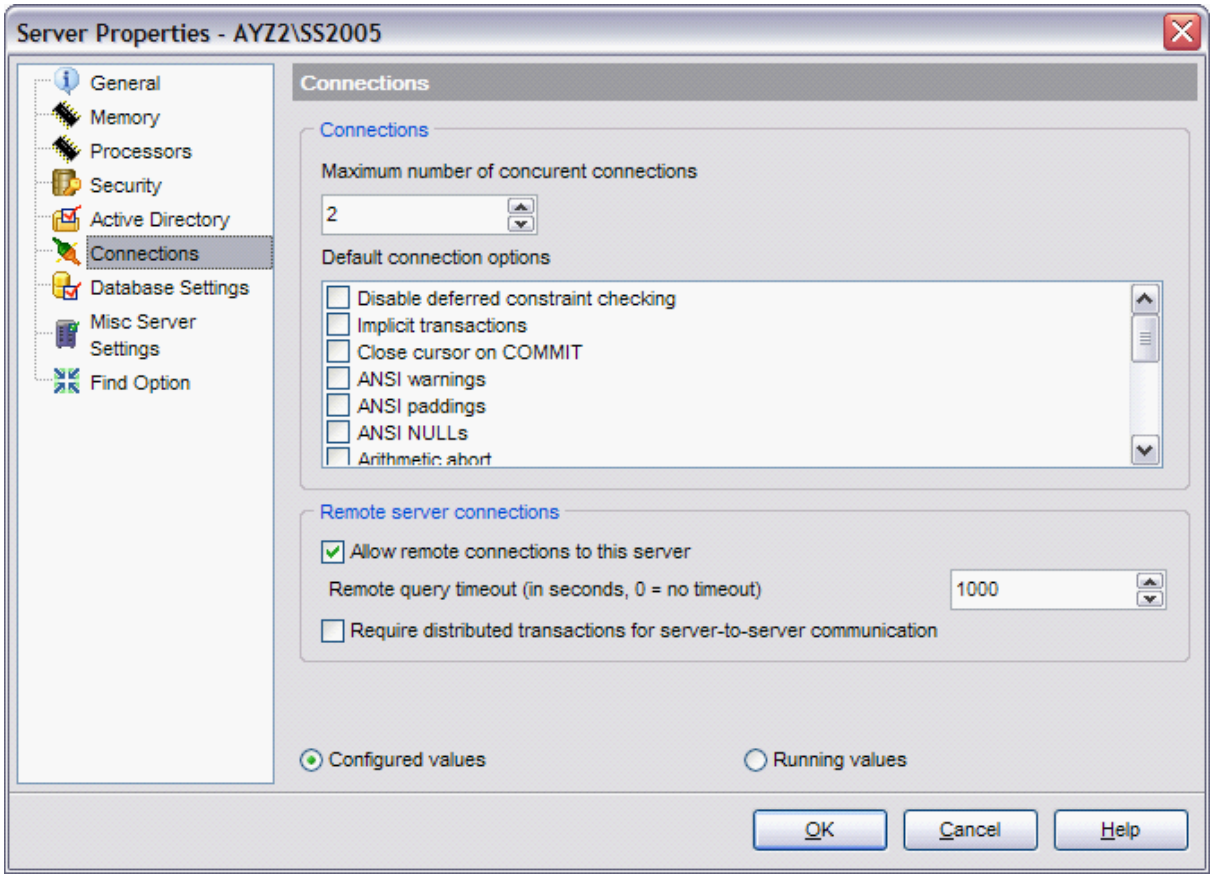
Processor options
Security options
Connection settings
Database Settings
Misc Server Settings
Find Option

4.6.6 Connection settings

The **Connections** section of the **Server Properties** dialog allows you to configure the instance of Microsoft® SQL Server by setting options pertaining to connection in groups: *Connections*, *Remote server connections*.

Connections

Maximum number of concurrent connections
If this option is set to a value other than zero, the number of connections that SQL Server will allow is limited. The default value is *0* (unlimited).



Default connection options
Specifies the default connection options as described in the table below:

© 2008 EMS Database Management Solutions

Configuration option	Description
Disable deferred constraint checking	Controls interim or deferred constraint checking.
Implicit transactions	Controls whether a transaction is started implicitly when a statement is run.
Cursor close on COMMIT	Controls behavior of cursors after a commit operation has been performed.
ANSI warnings	Controls truncation and NULL in aggregate warnings.
ANSI padding	Controls padding of fixed-length variables.
ANSI NULLs	Controls NULL handling when using equality operators.
Arithmetic abort	Terminates a query when an overflow or divide-by-zero error occurs during query execution.
Arithmetic ignore	Returns NULL when an overflow or divide-by-zero error occurs during a query.
Quoted identifier	Differentiates between single and double quotation marks when evaluating an expression.
No count	Turns off the message returned at the end of each statement that states how many rows were affected.
ANSI NULL default on	Alters the session's behavior to use ANSI compatibility for nullability. New columns defined without explicit nullability are defined to allow nulls.
ANSI NULL default off	Alters the session's behavior not to use ANSI compatibility for nullability. New columns defined without explicit nullability are defined not to allow nulls.
Concat NULL yields NULL	Returns NULL when concatenating a NULL value with a string.
Numeric round abort	Generates an error when a loss of precision occurs in an expression.
Xact abort	Rolls back a transaction if a Transact-SQL statement raises a run-time error.

Remote Server Connections

Allow remote connections to this server
Controls the execution of stored procedures from remote servers running instances of SQL Server. Selecting this check box has the same effect as setting the *sp_configureremote access* option to *1*. Clearing it prevents execution of stored procedures from a remote server.

Remote query timeout (in seconds, 0 = no timeout)
Specifies how long (in seconds) a remote operation may take before SQL Server times

© 2008 EMS Database Management Solutions

out. The default is 600 seconds, or a 10-minute wait.

Require distributed transactions for server-to-server communication
Protects the actions of a server-to-server procedure through a Microsoft® Distributed Transaction Coordinator (MS DTC) transaction.

See also:
General options
Memory options
Processor options
Security options
Active Directory
Database Settings
Misc Server Settings
Find Option

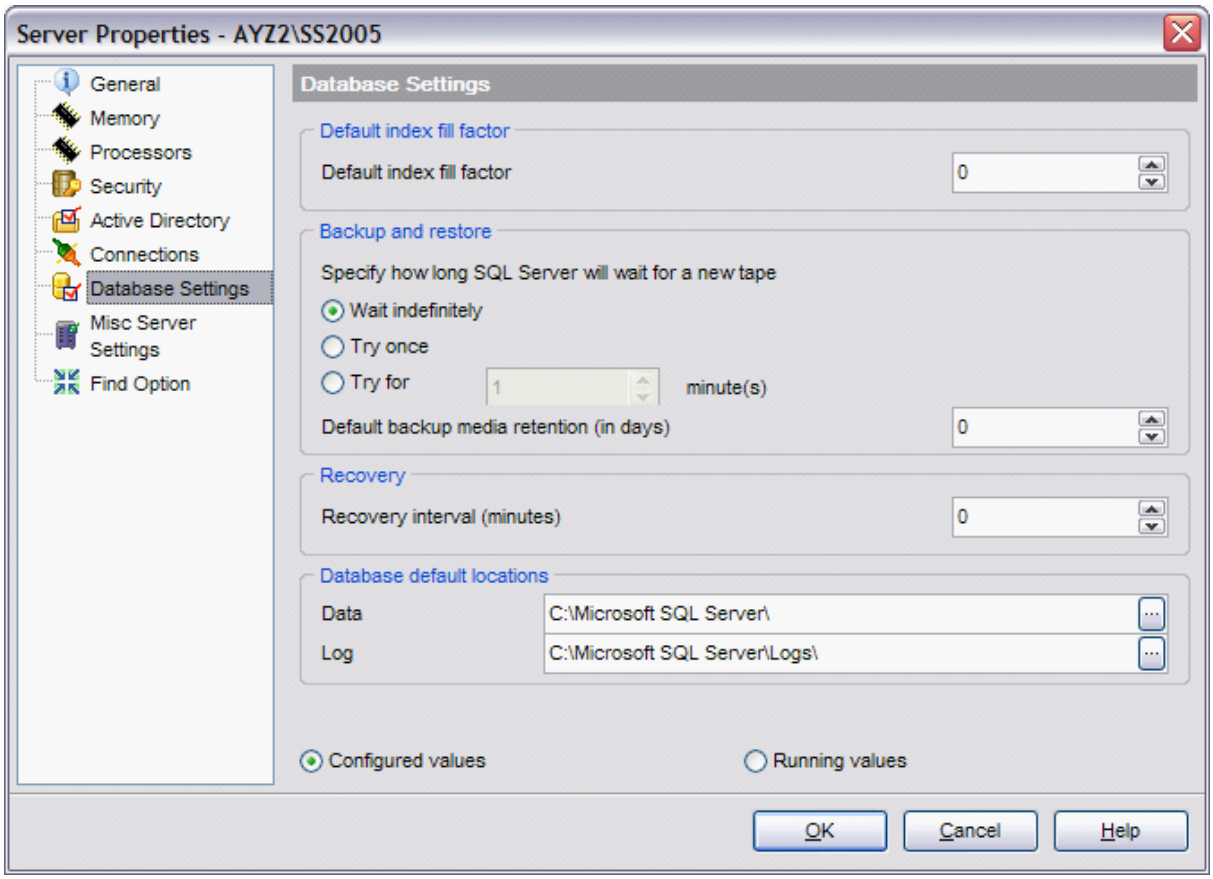
4.6.7 Database Settings

The **Database Settings** section of the **Server Properties** dialog allows you to configure the instance of Microsoft® SQL Server by setting options pertaining to database in groups: *Default index fill factor*, *Backup and restore*, *Recovery*, *Database default locations*.

Default index fill factor

This value specifies how full SQL Server should make each page when it creates a new index using existing data. The fill factor affects performance because SQL Server takes time to split pages when they fill up.
The default value is *0*; valid values range from *0* to *100*. A fill factor of *0* or *100* creates clustered indexes with full data pages and nonclustered indexes with full leaf pages, but it leaves some space within the upper level of the index tree. Fill factor values *0* and *100* are identical in all respects.
With low fill factor values specified, SQL Server creates indexes with pages that are not full. Each index takes more storage space, but there is more room for subsequent insertions without requiring page splits.

© 2008 EMS Database Management Solutions



Backup and restore

Wait indefinitely
Specifies that SQL Server will never time out while waiting for a new backup tape.

Try once
Specifies that SQL Server will time out if a backup tape is not available when needed.

Try for ... minute(s)
Specifies that SQL Server will time out if a backup tape is not available within the defined period of time.

Default backup media retention (in days)
Provides a system-wide default for the length of time to retain each backup medium after it has been used for a database or transaction log backup. This option helps protect backups from being overwritten until the specified number of days has elapsed.

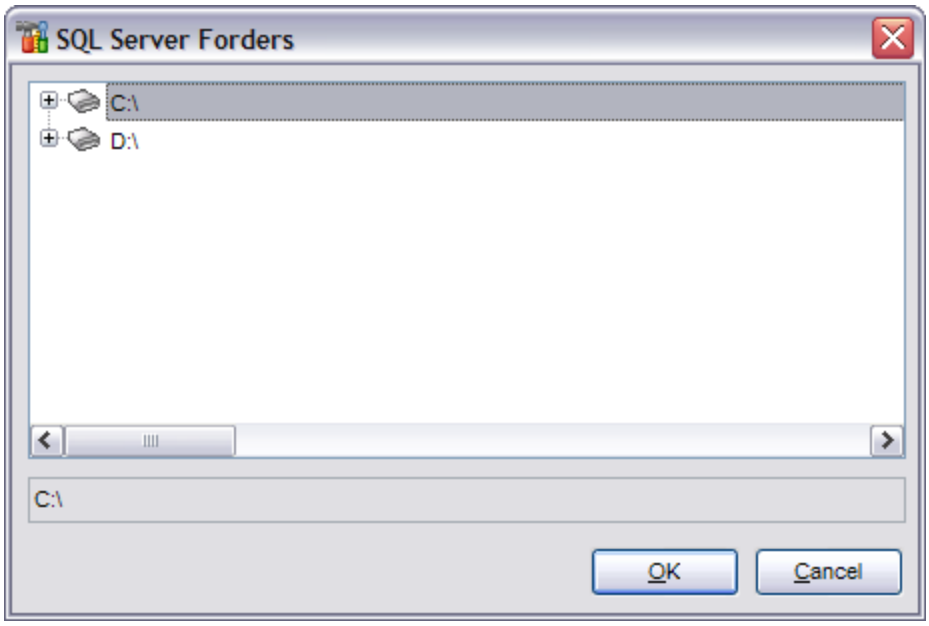
Recovery

Recovery interval (minutes)
Sets the maximum period of time to recover a database. The default is *0*, it stands for automatic configuration by SQL Server. In practice, this means a recovery time of less than one minute and a checkpoint approximately every one minute for active databases.

© 2008 EMS Database Management Solutions

Database default locations

Data
Specify the default location for data files: type in the path in the **Data** field or use the ellipsis button to browse for directory using the **SQL Server Folders** dialog.



Log
Specify the default location for log files: type in the path in the **Log** field or use the ellipsis button to browse for directory using the **SQL Server Folders** dialog.

See also:
General options
Memory options
Processor options
Security options
Active Directory
Connection settings
Misc Server Settings
Find Option

4.6.8 Misc Server Settings

The **Misc Server Settings** section of the **Server Properties** dialog allows you to configure the instance of Microsoft® SQL Server by setting miscellaneous server options in groups: *General settings*, *Two-digit year support*, *Network settings*.

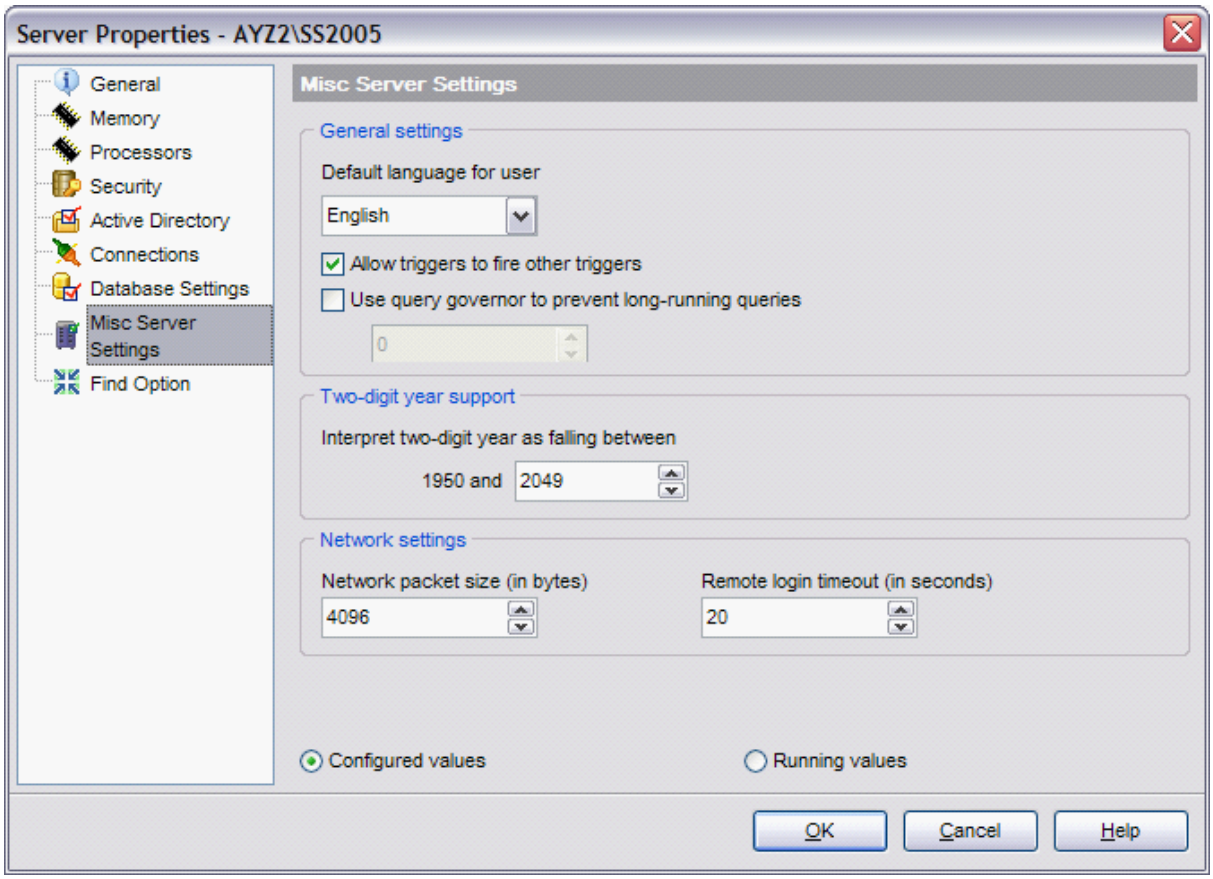
© 2008 EMS Database Management Solutions

General settings

Use the **Default language for user** drop-down list to select the default language for all newly created user logins.

Allow triggers to fire other triggers
Use this option to specify whether server-level triggers can be fired recursively.

Use query governor to prevent long-running queries
This option allows you to specify a limit on the time period in which a query can run. The query governor disallows execution of any query that has an estimated cost exceeding that value.



Two-digit year support

Interpret two-digit year as falling between 1950 and ...
Use this option to specify an integer from 1950 to 9999 that represents the cutoff year for interpreting two-digit years as four-digit years.

Use the **Network packet size** option to set the packet size (in bytes) used across the entire network.

Use the **Remote login timeout** option to specify the time interval (in seconds) to wait

© 2008 EMS Database Management Solutions

before returning from a failed attempt to log in to a remote server.

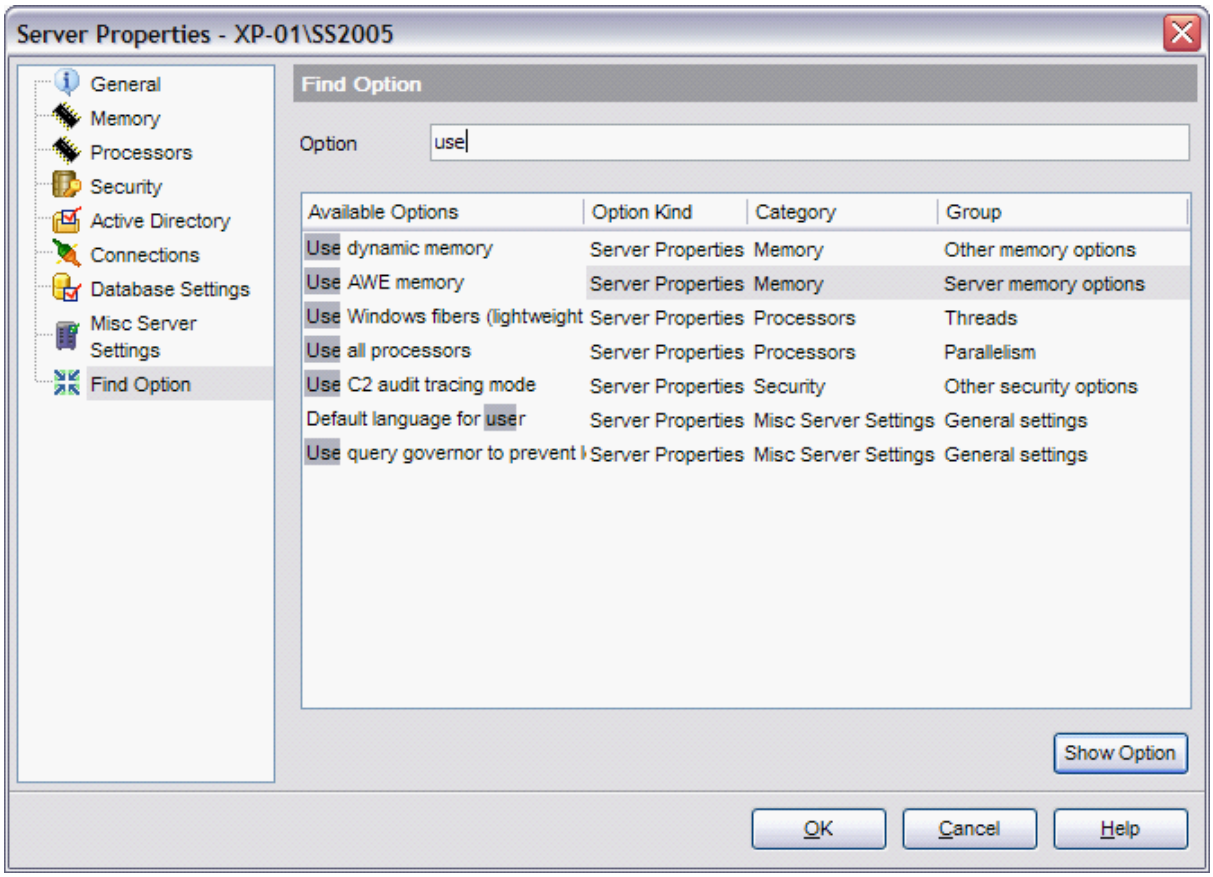
See also:
General options
Memory options
Processor options
Security options
Active Directory
Connection settings
Database Settings
Find Option

4.6.9 Find Option

The **Find Option** section allows you to search for options available within the **Server Properties** dialog easily and quickly.

Option
In this field you can enter the name of the option to search for within the server properties.

© 2008 EMS Database Management Solutions



The **Available options** area lists all options of the Server Properties category according to the specified name. Select the required option in the list and click **Show Option** to open the corresponding section where you can view/edit the value of this option.

See also:

General options
Memory options
Processor options
Security options
Active Directory
Connection settings
Database Settings
Misc Server Settings

© 2008 EMS Database Management Solutions

Part V

5 Database objects management

SQL Manager 2008 for SQL Server provides powerful tools to manage **database objects**.

To obtain detailed information concerning SQL Server database objects, refer to the official Microsoft® SQL Server™ documentation.

Note: before working with database objects in SQL Manager 2008 for SQL Server you should connect to the database first.

Creating Database Objects

To create a database object:
- select the **Database | New Object...** main menu item;
- select the type of object within the New Object Dialog;
- follow the steps of the wizard which guides you through the entire process of creating the object, or set the object properties using its editor - depending on the selected object type.

Note that you can also create a database object by selecting the appropriate context menu item of the DB Explorer tree or using the *Ctrl+N* shortcut.

Dropping Database Objects

To drop a database object:
- select the database object in the DB Explorer tree;
- right-click the object to call its context menu and select the **Drop <object type> <object name>** context menu item;
- confirm dropping in the dialog window.

Editing Database Objects

To edit a database object:
- select the database object in the DB Explorer tree;
- right-click the object to call its context menu and select the **Edit <object type> <object name>** context menu item, or double-click the object to open it in its editor.

When using an object editor, you can benefit from **tabs**. To switch between tabs, click on their respective tabs at the top of the main window. You can do it at anytime since views are absolutely independent.

To compile a newly created or edited object, you can use the **Compile** item available within the **Navigation bar** of the object editor.

Availability:

Full version (for Windows)	**Yes**

© 2008 EMS Database Management Solutions

Lite version (for Windows)	**Yes**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

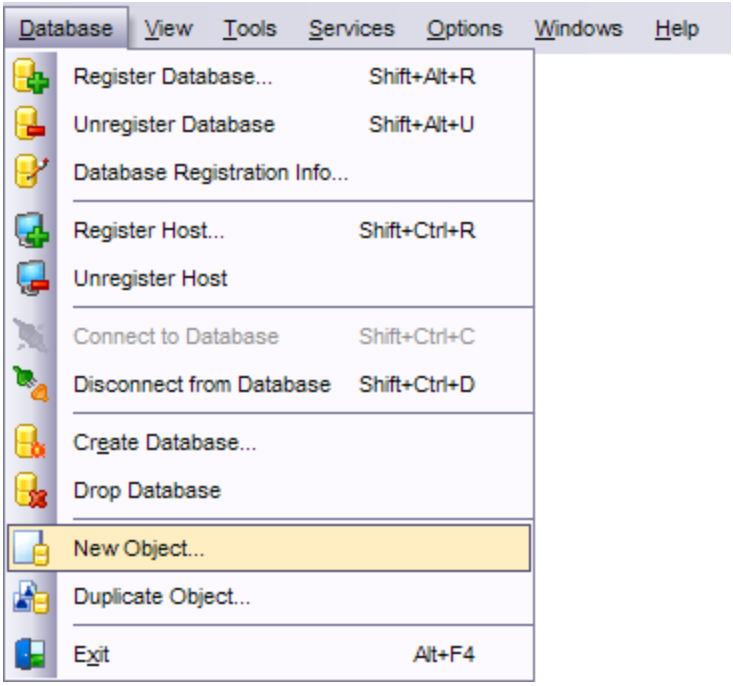
Getting started
Database Explorer
Database management
Query management tools
Data management
Import/Export tools
Database tools
Server tools
Personalization
External tools

© 2008 EMS Database Management Solutions

5.1 New Object

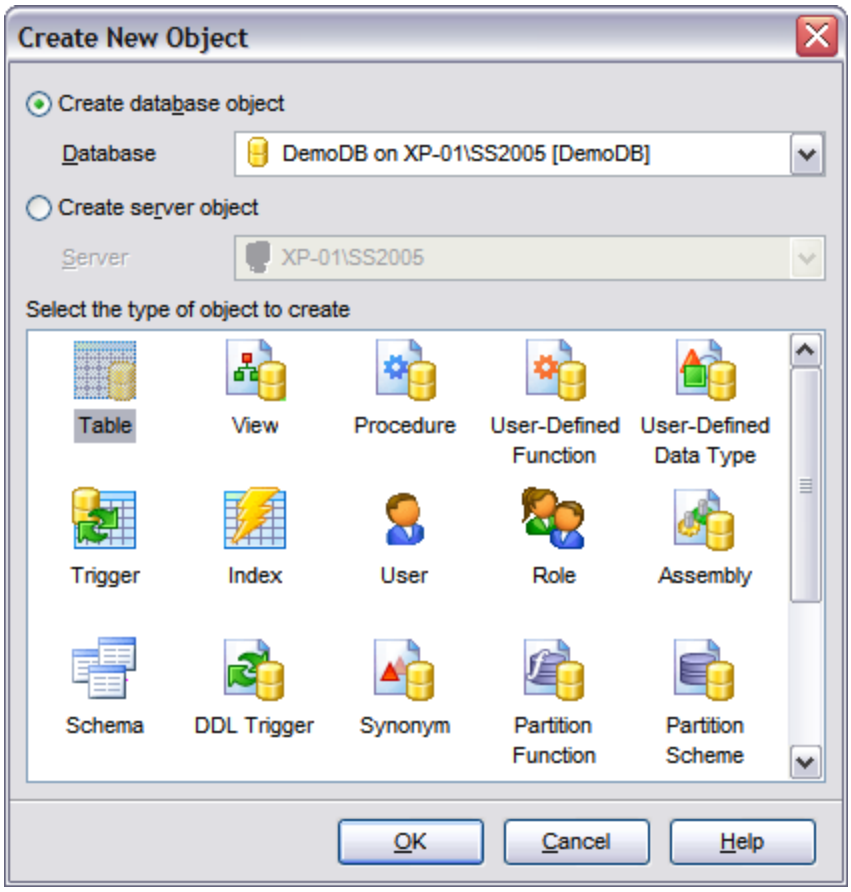
The **Create New Object** dialog allows you to select the type of the object to be created and run the appropriate wizard or editor.

To open the dialog, select the **Database | New Object...** main menu item.



Specify whether object of the database or server level is to be created. Use the **Database** or **Server** drop-down list to select the alias of the database or server where the new object should be created. Pick an object type icon and click **OK** to invoke the corresponding wizard or dialog.

© 2008 EMS Database Management Solutions



See also:

Duplicate Object Wizard
Schemas
Schema objects
Non-schema objects
Database-level principals

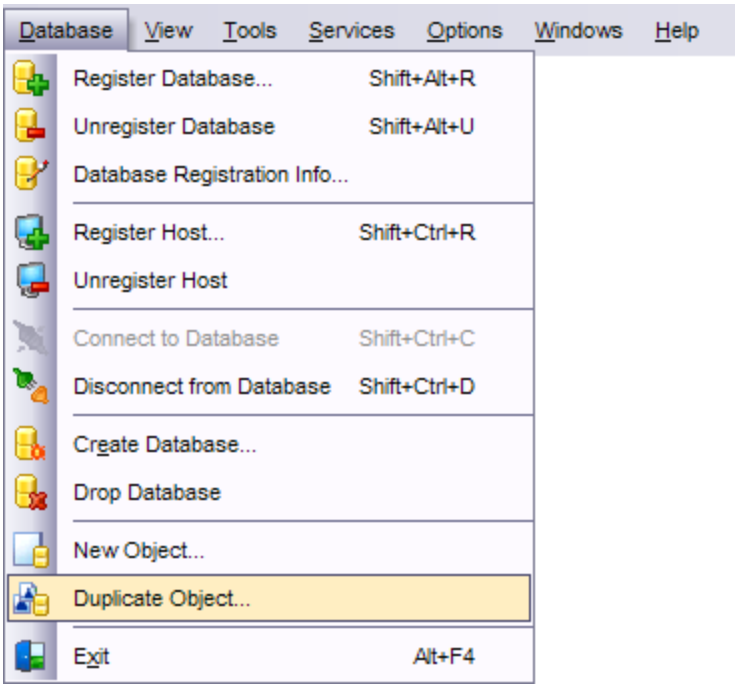
© 2008 EMS Database Management Solutions

5.2 Duplicate Object Wizard

Use the **Duplicate Object Wizard** to create a new database or server object of the same type and having the same properties as one of the existing ones.

To run the wizard, select the **Database | Duplicate Object...** main menu item, or right-click an object of the desired type in the DB Explorer tree and use the **Duplicate <object type> <object name>...** context menu item.

- Selecting the source database
- Selecting object to duplicate
- Selecting destination database
- Modifying the new object's definition



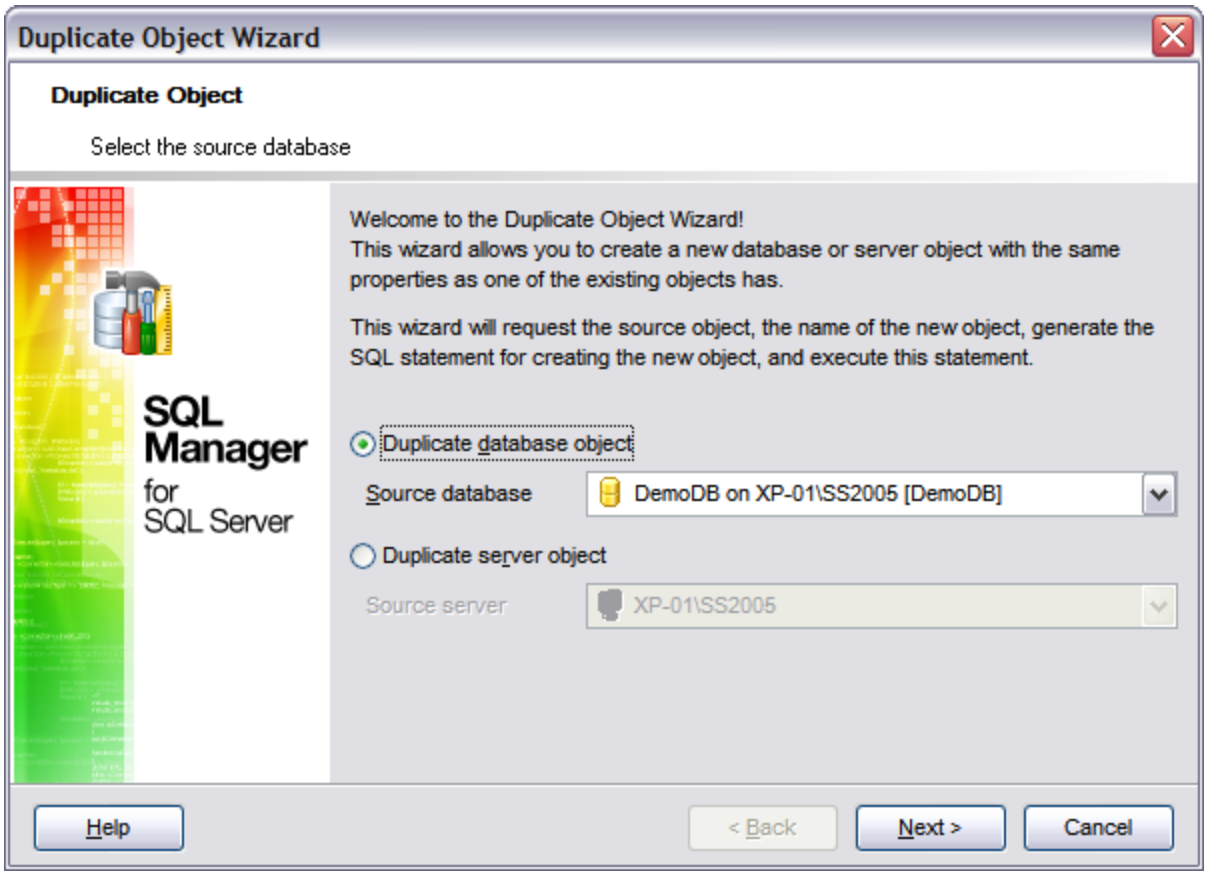
See also:

New Object dialog
Schemas
Schema objects
Non-schema objects
Database-level principals

5.2.1 Selecting the source database

This step of the wizard allows you to specify whether object of the database or server level is to be duplicated, and to select the **source database** / **server** containing the source object for duplication.

© 2008 EMS Database Management Solutions



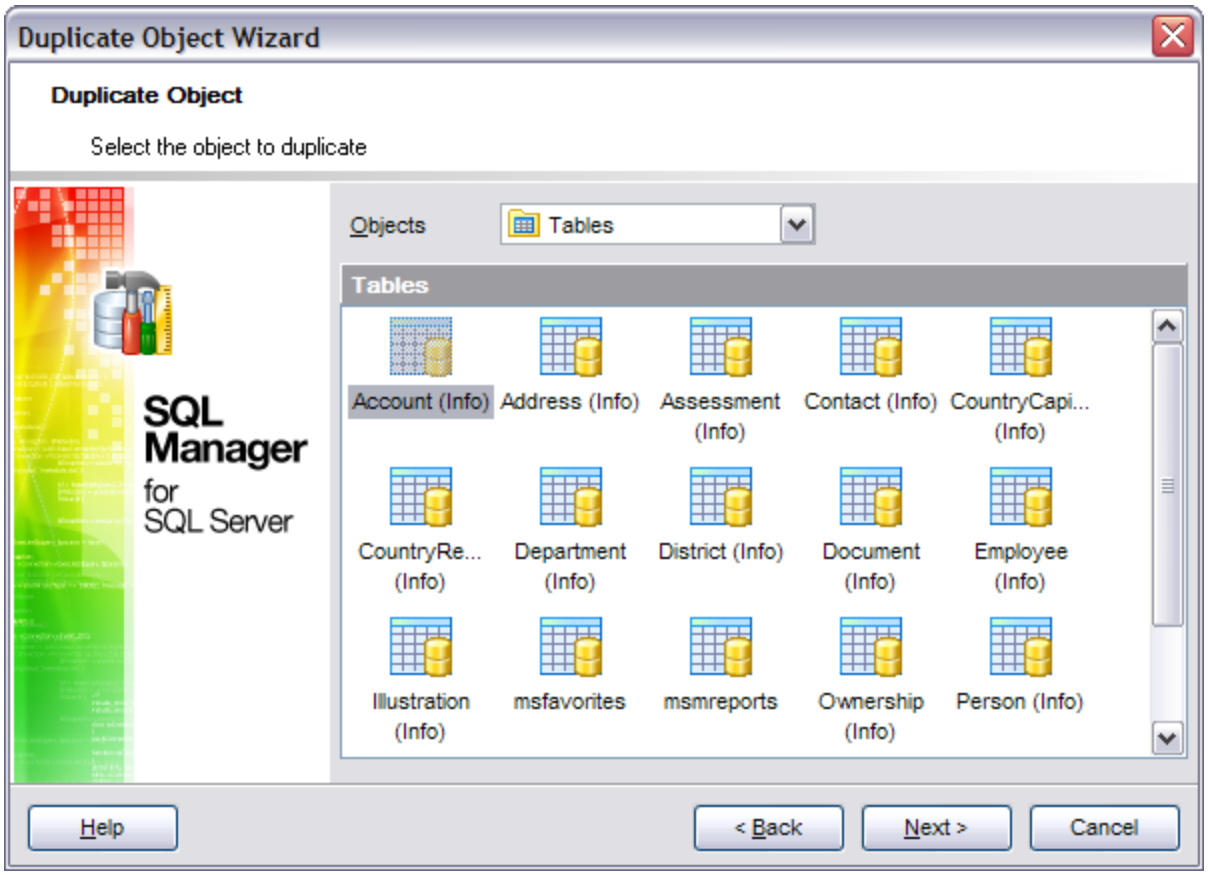
Click the **Next** button to proceed to the Selecting object to duplicate step of the wizard.

5.2.2 Selecting object to duplicate

Use the **Objects** drop-down menu to select the type of the object you intend to duplicate.

Select a database/server object to copy its properties to the new object.

© 2008 EMS Database Management Solutions



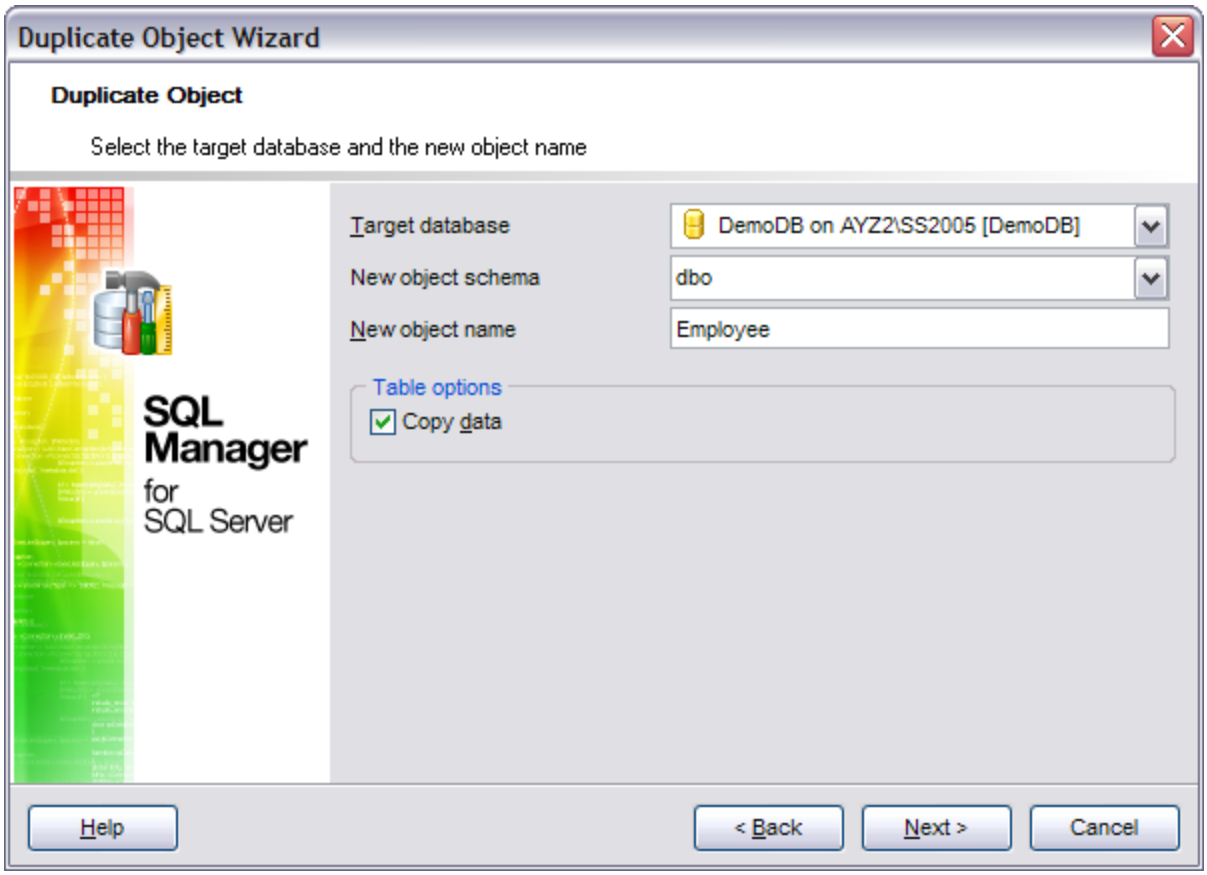
Click the **Next** button to proceed to the Selecting destination database step of the wizard.

5.2.3 Selecting destination database or server

Select the **target database** / **server** and **schema** (for a database object) to create the new object in, set the name of the new database/server object.

Check the ☑ **Copy data** option to copy data from the source table to the new one.

© 2008 EMS Database Management Solutions



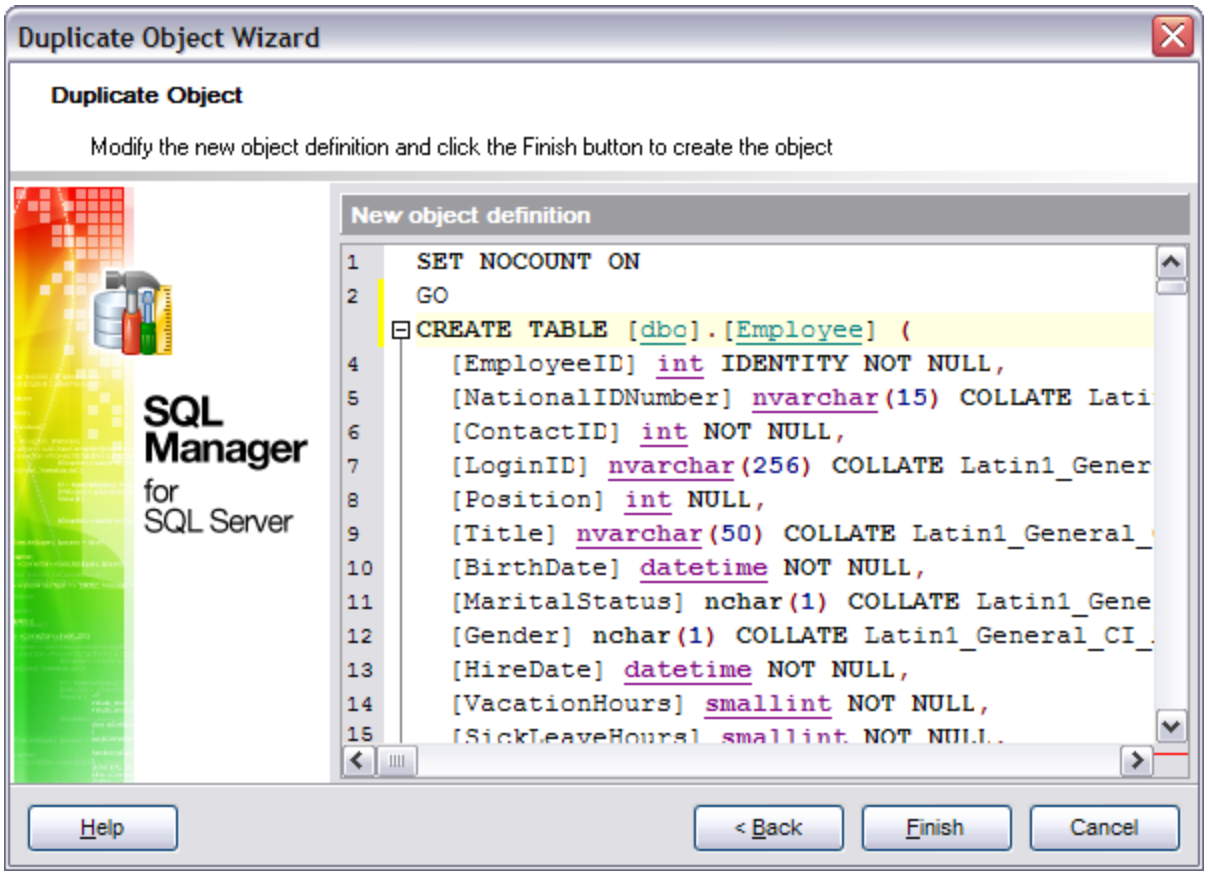
Click the **Next** button to proceed to the Modifying the new object's definition step of the wizard.

5.2.4 Modifying the new object's definition

This step of the wizard allows you to browse **the result SQL statement**.

If necessary, you can edit the definition of the new object.

© 2008 EMS Database Management Solutions



Click the **Finish** button to create the object.

© 2008 EMS Database Management Solutions

5.3 Schemas

A **schema** is essentially a namespace: it can be defined as a collection of database objects that form a single namespace. A namespace is a set in which each element has a unique name.

Creating Schemas

To create a new schema:
- select the **Database | New Object...** main menu item;
- select **Schema** in the Create New Object dialog;
- define schema properties using the appropriate tabs of Schema Editor.

Hint: To create a new schema, you can also right-click the **Schemas** node or any object within this node in the DB Explorer tree and select the **New Schema** item from the context menu.

To create a new schema with the same properties as one of existing schemas has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a schema in the DB Explorer tree and select the **Duplicate Schema <schema_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new schema in, and to edit the result SQL statement for creating the schema.

Editing Schemas

To edit an existing schema:
- select the schema for editing in the DB Explorer tree (type the first letters of the schema name for quick search);
- right-click the object and select the **Edit Schema <schema_name>...** context menu item, or simply double-click the schema;
- edit schema properties using the appropriate tabs of Schema Editor.

Dropping Schemas

To drop a schema:
- select the schema to drop in the DB Explorer tree;
- right-click the object and select the **Drop Schema <schema_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new schema;
Ctrl+O to edit the selected schema;
Shift+Del to drop the object from the database.

See also:

© 2008 EMS Database Management Solutions

New Object dialog
Duplicate Object Wizard
Schema objects
Non-schema objects
Database-level principals

5.3.1 Schema Editor

Schema Editor allows you to manage SQL Server schemas efficiently. It opens automatically when you create a new schema and is available on editing an existing one (see Create Schema and Edit Schema for details).

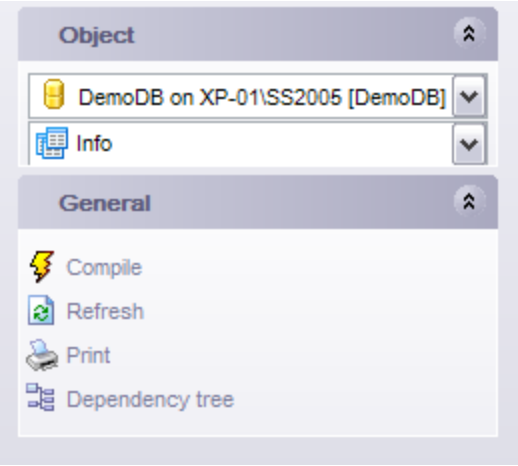
To open a schema in **Schema Editor**, double-click it in the DB Explorer tree.

- Using Navigation bar
- Editing schema definition
- Browsing object dependencies
- Editing schema description
- Viewing DDL definition

5.3.1.1 Using Navigation bar

The **Navigation bar** of **Schema Editor** allows you to:

- select the database where the schema is located;
- select the schema for editing;
- compile the schema (if it is being created/modified);
- save the schema description (if it has been modified);
- refresh the content of the active tab;
- print metadata of the schema;
- view the dependency tree for the schema;
- restore the default size of the editor window.



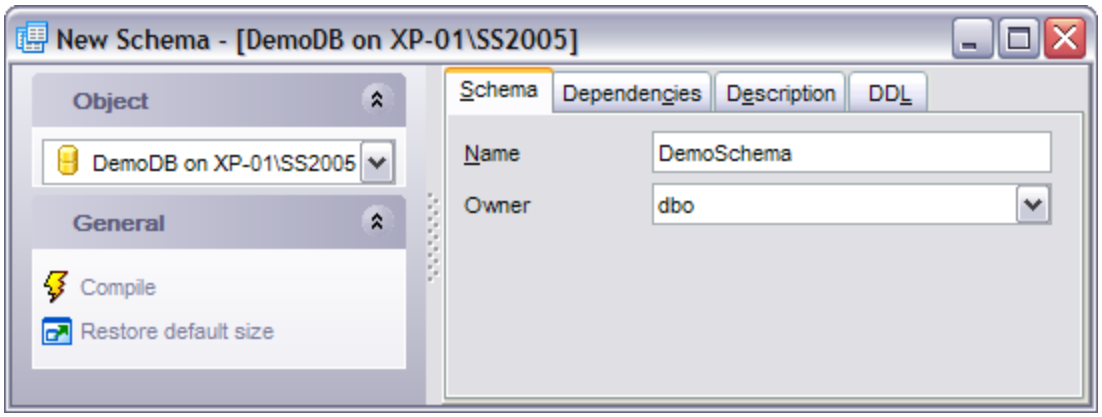
Depending on the current tab selection, the **Navigation bar** expands to one or more additional panes with tab-specific actions that can be useful for working with the schema.

© 2008 EMS Database Management Solutions

<< Back to Schema Editor

5.3.1.2 Editing schema definition

Use the **Schema** tab of **Schema Editor** to create/edit a schema and specify its properties.



Name
Specify the name by which the schema is identified within the database.

Owner
Specify the name of the database-level principal (user or role) that will own the schema. This principal may own other schemas, and may not use the current schema as its default schema.

© 2008 EMS Database Management Solutions

5.4 Schema Objects

A **schema** is a collection of logical structures of data, or **schema objects**. A schema is owned by a database user. Each user can own a single schema. Schema objects can be created and manipulated with SQL and include the following types of objects:

- Tables
- Views
- Procedures
- User-defined types
- User-defined functions
- Synonyms
- Aggregates
- XML Schema Collections
- Defaults
- Rules

Use the DB Explorer tree to navigate within the existing schemas and their objects.

See also:
New Object dialog
Duplicate Object Wizard
Schemas
Non-schema objects
Database-level principals

5.4.1 Tables

Relational databases store all their data in **tables**. A table is a data structure consisting of an unordered set of horizontal rows, each containing the same number of vertical columns. The intersection of an individual row and column is a field that contains a specific piece of information. Much of the power of relational databases comes from defining the relations among the tables.

Creating Tables

To create a new table:

- select the **Database | New Object...** main menu item;
- select **Table** in the Create New Object dialog;
- define table properties and fields using the appropriate tabs of Table Editor.

Hint: To create a new table, you can also right-click the **Tables** node or any object within this node in the DB Explorer tree and select the **New Table...** item from the context menu.

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

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

Alternatively, you can right-click a table in the DB Explorer tree and select the **Duplicate Table <table_name>...** context menu item.

© 2008 EMS Database Management Solutions

Duplicate Object Wizard allows you to select the database to create a new table in, and to edit the result SQL statement for creating the table.

Editing Tables

To edit an existing table (manage its fields, indexes, data, etc):
- select the table for editing in the DB Explorer tree (type the first letters of the table name for quick search);
- right-click the object and select the **Edit Table <table_name>** context menu item, or simply double-click the table;
- edit table subobjects and data using the appropriate tabs of Table Editor.

To change the name of a table:
- select the table to rename in the DB Explorer tree;
- right-click the table alias and select the **Rename Table <table_name>...** item from the context menu;
- edit the table name using the **Rename Object...** dialog.

Dropping Tables

To drop a table:
- select the table to drop in the DB Explorer tree;
- right-click the object and select the **Drop Table <table_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new table;
Ctrl+O to edit the selected table;
Ctrl+R to rename the table;
Shift+Del to drop the object from the database.

5.4.1.1 New table

The **New Table** window is a mode of Table Editor that opens automatically when you create a new table (see Create table for details) and allows you to create a new table, set table properties, specify table fields and edit table description.

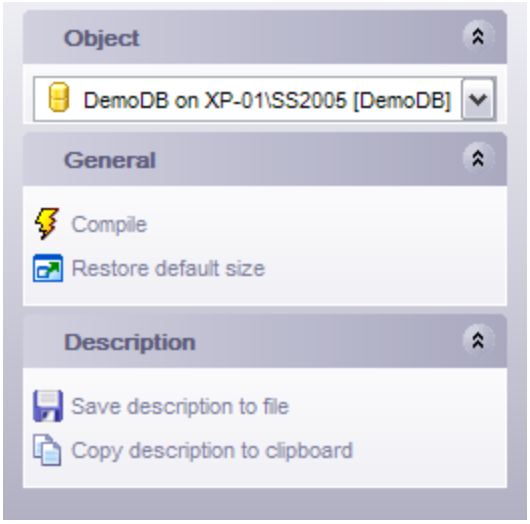
To call **Table Editor** for creating a new table, you can right-click the **Tables** node or any object within this node in the DB Explorer tree and use the *Ctrl+N* shortcut.
- Using Navigation bar
- Setting table properties
- Specifying table fields

5.4.1.1.1 Using Navigation bar

The **Navigation bar** of **Table Editor** (in the *New table* mode) allows you to:
- select the database to create a new table in;

© 2008 EMS Database Management Solutions

- compile the newly created table;
- save the table description (if it has been added) or copy the description text to clipboard;
- restore the default size of the editor window.



Depending on the current tab selection, the **Navigation bar** expands to one or more additional panes with tab-specific actions that can be useful for working with the table.

<< Back to creating new table

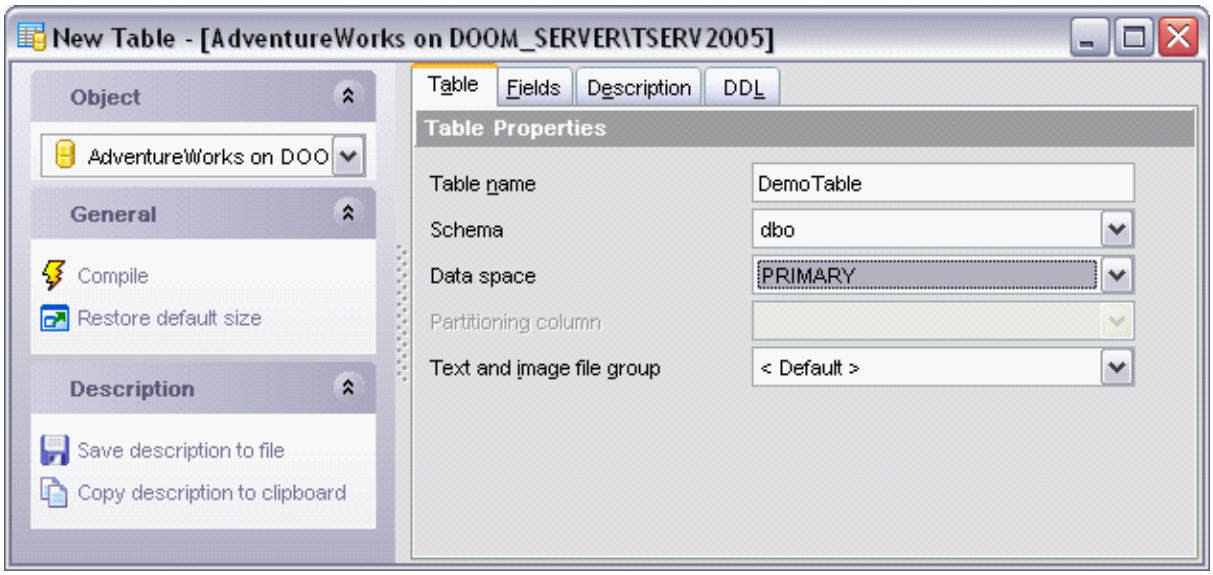
5.4.1.1.2 Setting table properties

Use the **Table** tab of **Table Editor** to create a table and specify its properties.

Table name
Enter a name for the new table. Note that table names must comply with the rules for identifiers and must be unique within the schema. A table name can contain a maximum of 128 characters.

Schema
Use the drop-down list to select the schema to which the table belongs. The default selection is the default schema for the current user.

© 2008 EMS Database Management Solutions



Data space
Use the drop-down list to specify the new table data storage (a file group or a partition scheme).

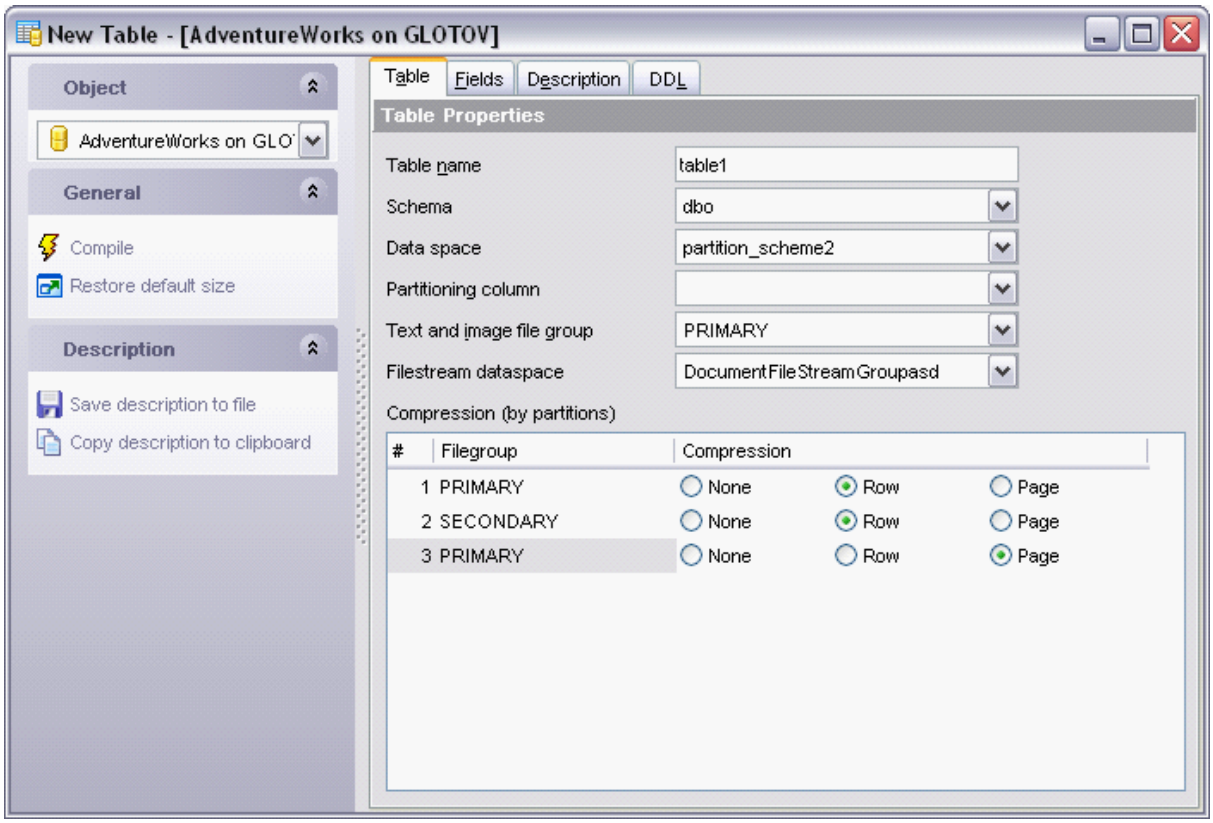
Partitioning column
This drop-down list is only available if a partition scheme is specified for **Data space**. Select a field that will be used as the partitioning column for the table.

Text and image file group
Use the drop-down list to select the file group where data of the *text* and *image* types will be stored.

If necessary, you can also use the **Description** area to supply a *description* for the table being created.

If the database is registered at SQL Server 2008, then the **Table Editor** will provide some additional table properties to define.

© 2008 EMS Database Management Solutions



Filestream dataspace
Select the needed Filestream filegroup from the drop-down list.

Compression (by partitions)
Use this part of the **Table Editor** to set the compression options for the table. Compression type can be different for each table partition. Set the partition compression to *Row* or *Page.* Set *None* to leave a partition uncompressed.

To compile the table, use the **Compile** item available within the Navigation bar.

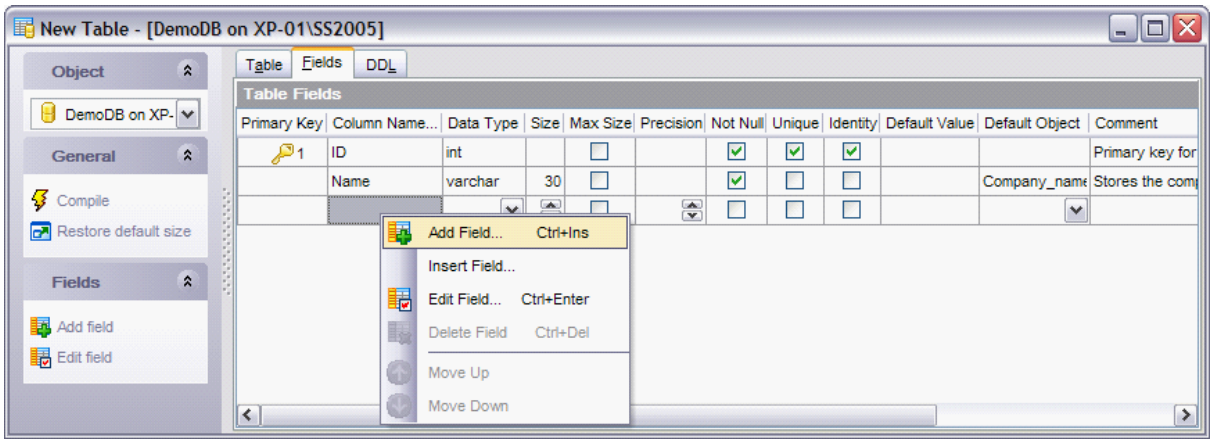
5.4.1.1.3 Specifying fields

The **Fields** tab is intended for setting up table fields. Double-click a field to open Field Editor for editing the field.

Right-click within the **Table Fields** area to display the context menu allowing you to *add*, *insert*, *edit* or *delete* fields.

Fields management tasks are also available through the Navigation bar of **Table Editor**.

© 2008 EMS Database Management Solutions



The **Table Fields** list provides the following attributes of each field of the new table:
Primary Key
Column Name
Data Type
Size
Max Size
Precision
Not Null
Unique
Identity
Default Value
Default Object
Comment

For details see Fields.

To compile the table, use the **Compile** item available within the Navigation bar.

5.4.1.2 Table Editor

Table Editor is the basic SQL Manager tool for working with tables. It opens automatically in the New table mode when you create a new table and is available on editing an existing one (see Create table and Edit table for details).

Table Editor allows you to create, edit and drop table's fields, indexes, foreign keys and other table subobjects, manage table data, properties and much more.

To open a table in **Table Editor**, double-click it in the DB Explorer tree.

- Using Navigation bar
- Managing table fields
- Changing fields order
- Managing table foreign keys
- Managing table checks
- Managing table indexes
- Managing table triggers

© 2008 EMS Database Management Solutions

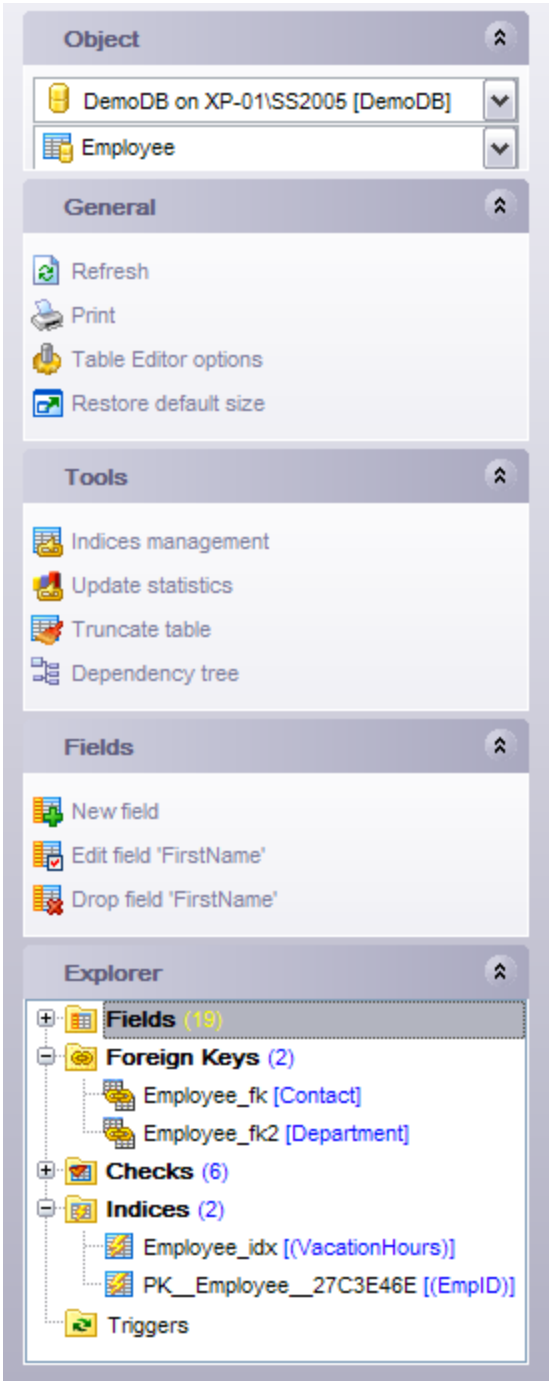
- Browsing object dependencies
- Working with table data
- Editing table description
- Viewing DDL definition
- Table properties

5.4.1.2.1 Using Navigation bar

The **Navigation bar** of **Table Editor** (in the *Edit table* mode) allows you to:

- select the database where the table is located;
- select the table for editing;
- compile the table (if it is being modified);
- save the table description (if it has been modified);
- refresh the content of the active tab;
- print metadata of the table;
- adjust Table Editor options;
- restore the default size of the editor window;
- manage table indexes using Indices Management Wizard;
- update table statistics using Statistics Update Wizard;
- truncate data of the table (with the *TRUNCATE TABLE* statement used);
- view the dependency tree for the table;
- browse the table subobjects using the Explorer tree.

© 2008 EMS Database Management Solutions



Depending on the current tab selection, the **Navigation bar** expands to one or more additional panes with tab-specific actions that can be useful for working with the object.

[<< Back to Table Editor]

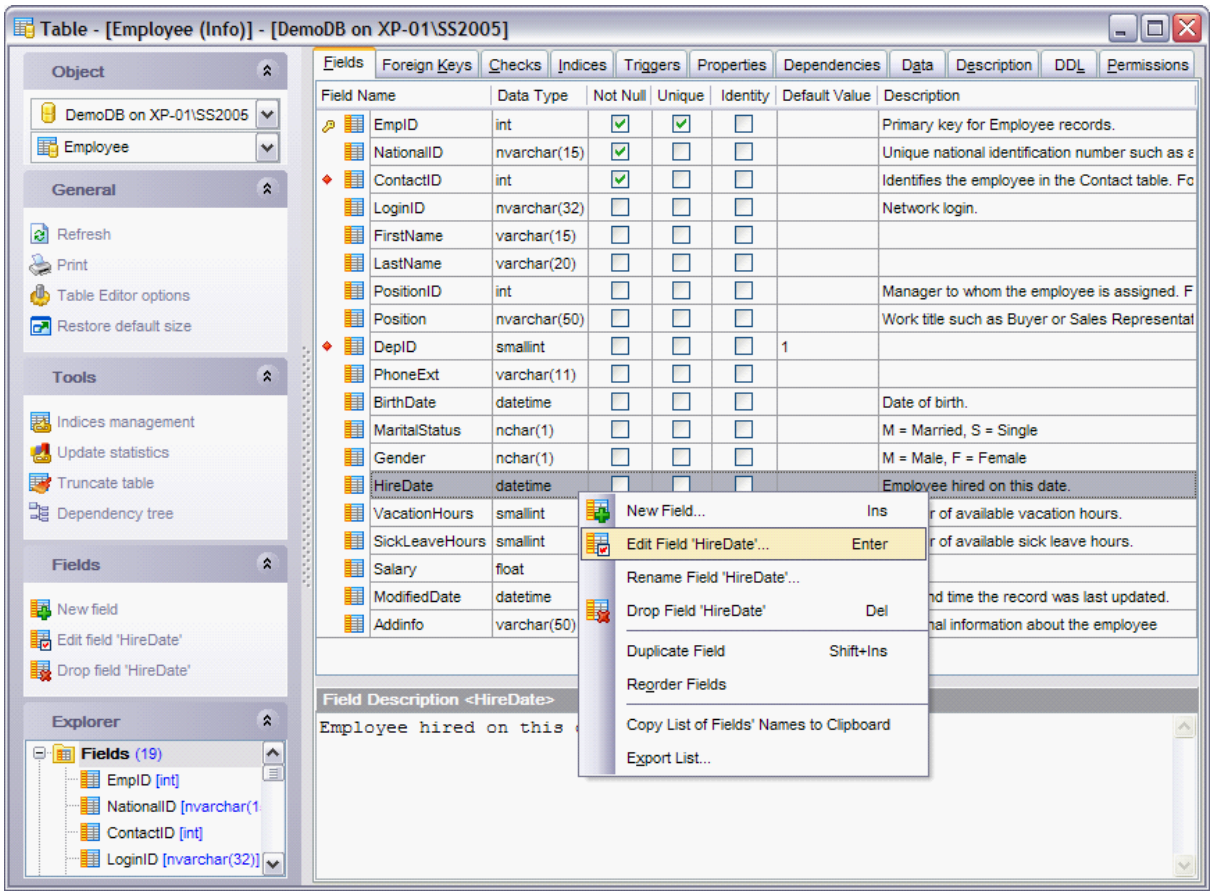
© 2008 EMS Database Management Solutions

5.4.1.2.2 Managing fields

The **Fields** tab is intended for managing table fields. Double-click a field to open Field Editor for editing the field.

Right-click a field to display the context menu allowing you to *create* new, *edit*, *drop*, *rename*, *duplicate* the selected field, or reorder fields of the table. Using the menu you can also export the list of the table fields to any of supported formats.

Fields management tasks are also available through the Navigation bar of **Table Editor**.



The **Fields** list provides the following attributes of each field of the table:
Field Name
Data Type
Not Null
Unique
Identity
Default Value

For details see Fields.

If necessary, you can also use the **Field Description** area to supply a *description* for each field.

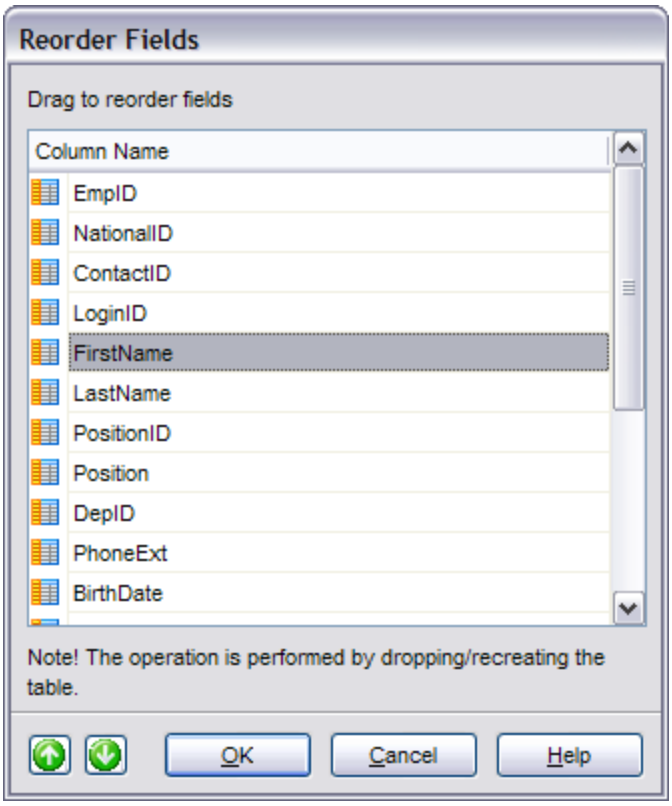
© 2008 EMS Database Management Solutions

5.4.1.2.3 Changing fields order

The **Reorder Fields** dialog allows you to change the fields order in the table.

To open this dialog, open the table in **Table Editor**, proceed to the Fields tab there, right-click within the **Fields** list and select the **Reorder Fields** item from the context menu.

To change the fields order, use the buttons or drag-and-drop operations within the list. Click the **OK** button to view the modification script in the Changing Metadata window and apply changes.



Important: This operation is performed by dropping/recreating the table.

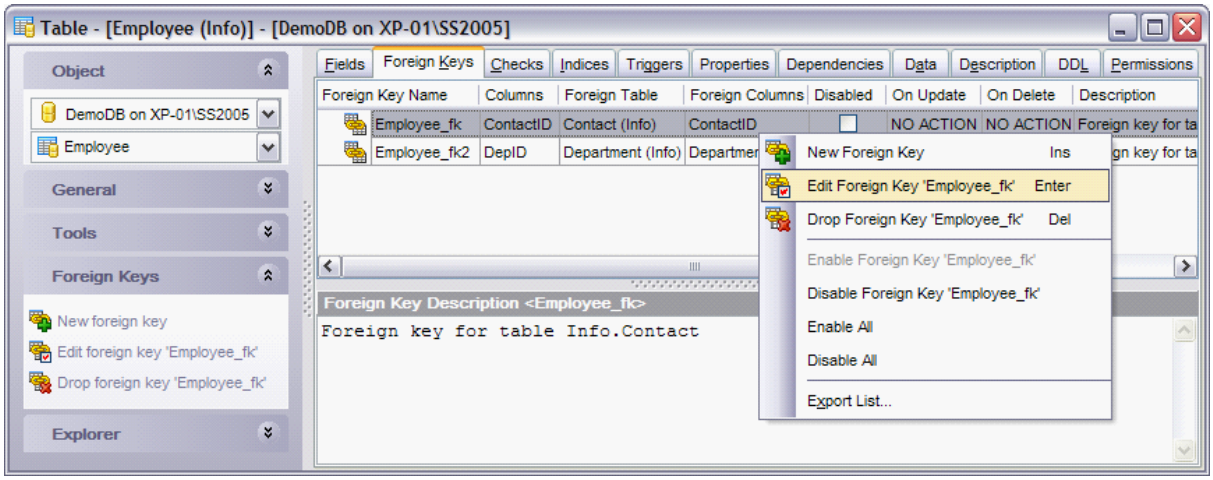
5.4.1.2.4 Managing foreign keys

The **Foreign Keys** tab is provided for managing table foreign keys. Double-click a foreign key to open Foreign Key Editor for editing the foreign key.

Right-click a foreign key to display the context menu allowing you to *create* new, *edit*, *drop*, *enable*/*disable* the selected foreign key, or *enable*/*disable* all foreign keys. Using the menu you can also export the list of the table foreign keys to any of supported formats.

© 2008 EMS Database Management Solutions

Foreign keys management tasks are also available through the Navigation bar of **Table Editor**.



The **Foreign Keys** list provides the following attributes of each foreign key of the table:
Foreign Key Name
Columns
Foreign Table
Foreign Columns
Disabled
On Update
On Delete

For details see Foreign Keys.

If necessary, you can also use the **Foreign Key Description** area to supply a *description* for each foreign key.

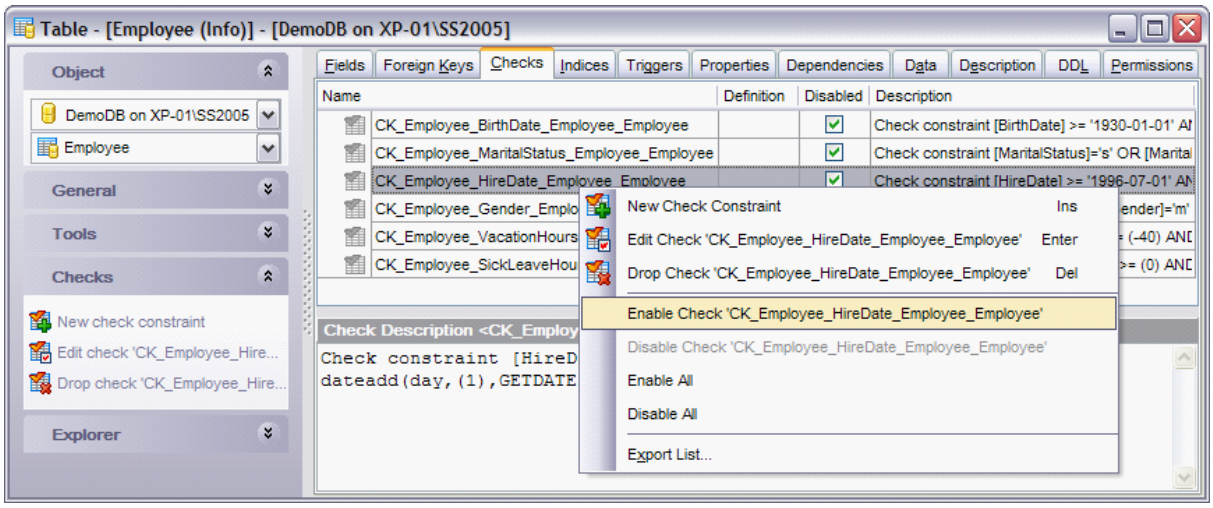
5.4.1.2.5 Managing checks

The **Checks** tab is provided for managing table check constraints. Double-click a check to open Check Editor for editing the check.

Right-click a check to display the context menu allowing you to *create* new, *edit*, *drop*, *enable*/*disable* the selected check, or *enable*/*disable* all checks. Using the menu you can also export the list of the table checks to any of supported formats.

Check constraints management tasks are also available through the Navigation bar of **Table Editor**.

© 2008 EMS Database Management Solutions



The **Checks** list provides the following attributes of each check constraint of the table:
Name
Definition

For details see Checks.

If necessary, you can also use the **Check Description** area to supply a *description* for each check constraint.

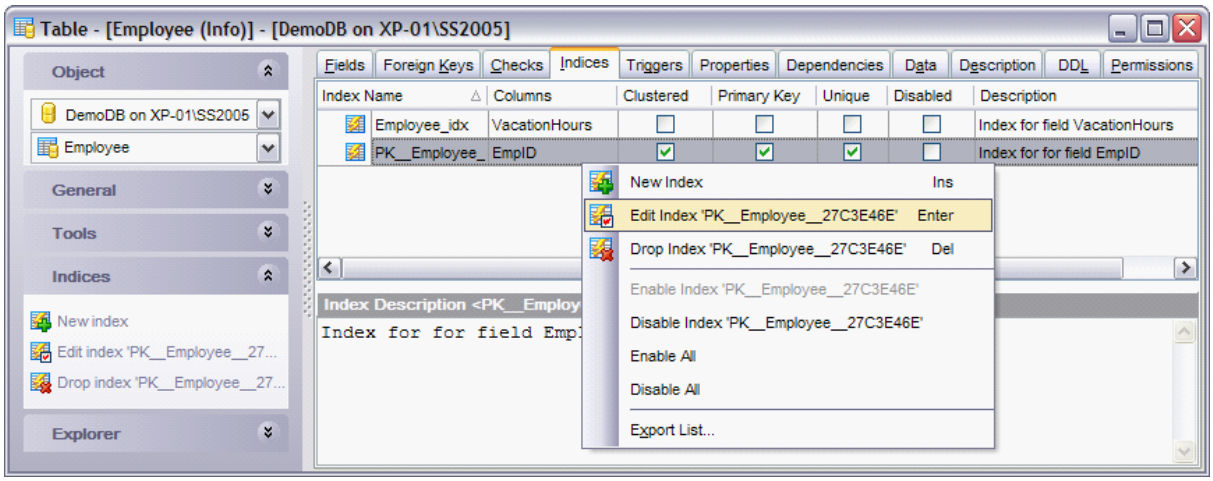
5.4.1.2.6 Managing indexes

The **Indices** tab is provided for managing table indexes. Double-click an index to open Index Editor for editing the index.

Right-click an index to display the context menu allowing you to *create* new, *edit*, *drop*, *enable*/*disable* the selected index, or *enable*/*disable* all indexes. Using the menu you can also export the list of the table indexes to any of supported formats.

Indexes management tasks are also available through the Navigation bar of **Table Editor**.

© 2008 EMS Database Management Solutions



The **Indexes** list provides the following attributes of each index of the table:
Index Name
Columns
Clustered
Primary Key
Unique
Disabled

For details see Indexes.

If necessary, you can also use the **Index Description** area to supply a *description* for each index.

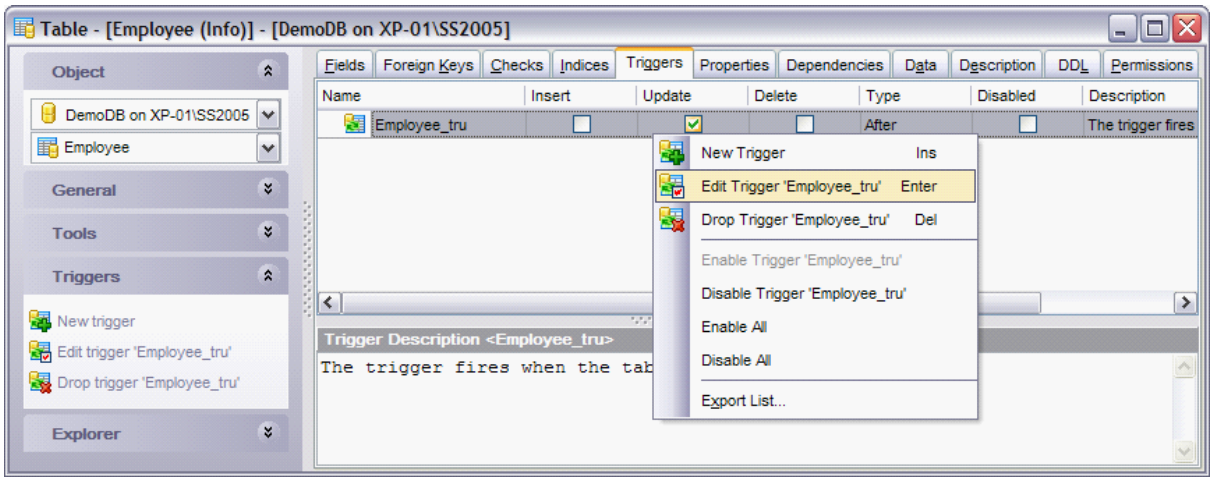
5.4.1.2.7 Managing triggers

The **Triggers** tab is provided for managing table triggers. Double-click a trigger to open Trigger Editor for editing the trigger.

Right-click the area to display the context menu allowing you to *create* new, *edit*, *drop*, *enable*/*disable* the selected trigger, or *enable*/*disable* all triggers. Using the menu you can also export the list of the table triggers to any of supported formats.

Triggers management tasks are also available through the Navigation bar of **Table Editor**.

© 2008 EMS Database Management Solutions



The **Triggers** list provides the following attributes of each trigger of the table:
Name
Insert
Update
Delete
Type
Disabled

For details see Triggers.

If necessary, you can also use the **Trigger Description** area to supply a *description* for each trigger.

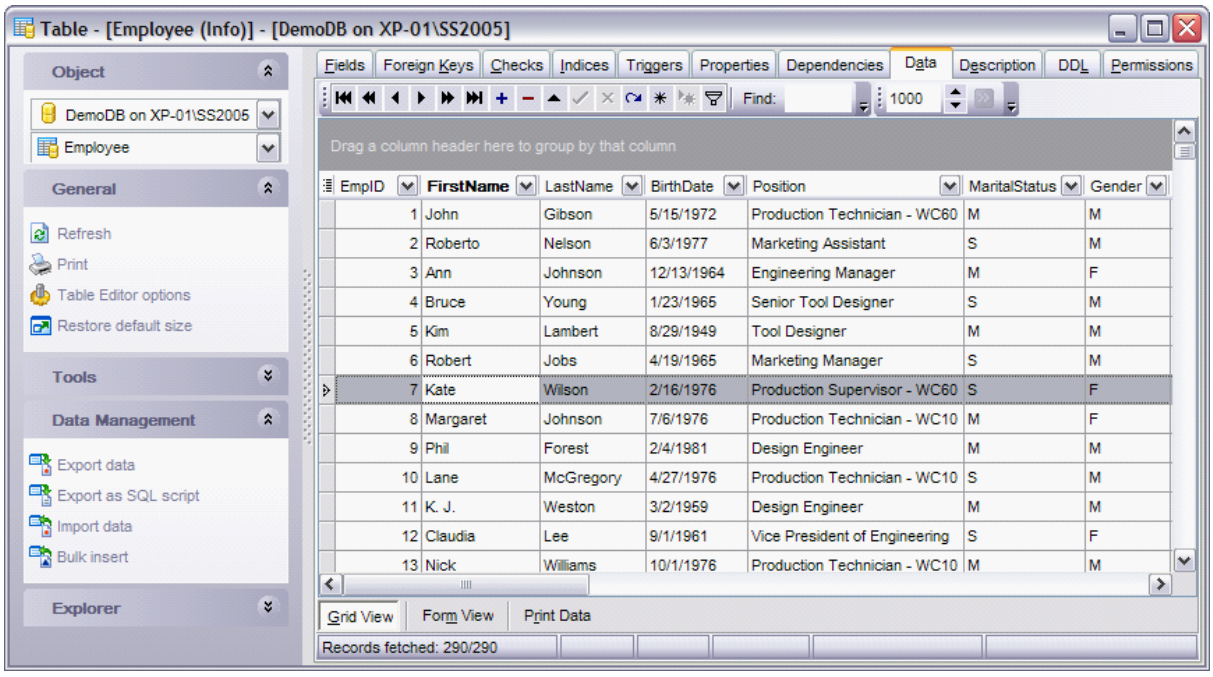
5.4.1.2.8 Working with table data

The **Data** tab displays the table data as a grid by default (see Data View for details). The context menu of this tab allows you to Export Data, Import Data, Export as SQL Script, Bulk Insert.

Data management tasks are also available through the Navigation bar of **Table Editor**.

While working with data, you are provided with a number of filtering and grouping facilities.
If necessary, you can **group the data in grid** by any of the columns. This operation is performed by dragging the column header to the gray **"Group by" box** area at the top. When grouping by a column is applied to the grid, all the rows are displayed as subnodes to the grouping row value. To reverse grouping, just drag the column header back.

© 2008 EMS Database Management Solutions



See also:

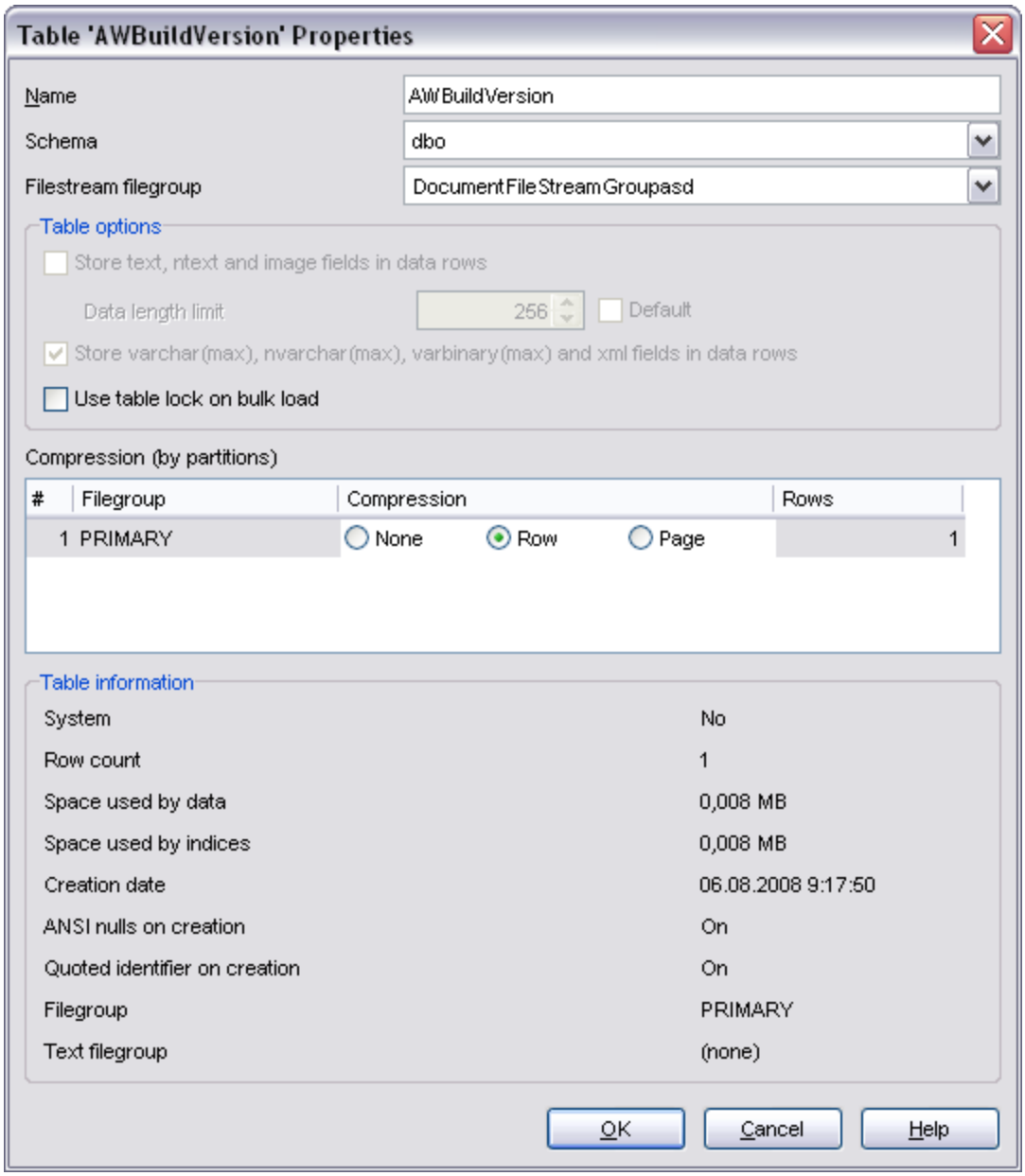
Working with view data

5.4.1.2.9 Table properties

The **Properties** tab allows you to view/edit common properties of the table: *Table name*, *Schema name*, *Table options* (*Storage attributes*, *Table lock usage*, etc.), *Table information*.

Hint: These properties are also available within a modal dialog which is called through the **Table Properties...** context menu item of the table alias in DB Explorer.

© 2008 EMS Database Management Solutions



Name
Edit the name of the table. Note that table names must comply with the rules for identifiers. The name of the table must be unique within the schema. A table name can contain a maximum of 128 characters.

Schema
Use the drop-down list to select a schema for the table.

Filestream dataspace
Select the needed Filestream filegroup from the drop-down list. This option is available only in SQL Server 2008.

© 2008 EMS Database Management Solutions

Table options
This group allows you to specify to **Store text, ntext and image fields in data rows**, set the **Data length limit**, specify to **Store varchar(max), nvarchar(max), varbinary(max) and xml fields in data rows**, and select whether to **Use table lock on bulk load** or not.

Compression (by partitions)
Use this part of the **Table Editor** to set the compression options for the table. Compression type can be different for each table partition. Set the partition compression to *Row* or *Page.* Set *None* to leave a partition uncompressed.

The **Table information** group allows you to view the table summary: *System* (*Yes*/*No*), *Row count*, *Space used by data*, *Space used by indices*, *Creation date*, *ANSI nulls on creation* (*On*/*Off*), *Quoted identifier on creation* (*On*/*Off*), *File group*, *Text file group.*

5.4.1.3 Fields

Table fields are managed within the **Fields** tab of Table Editor.

Creating Fields

To create a new table field:
- open the table in Table Editor;
- proceed to the **Fields** tab there;
- right-click the tab area and select the **New Field** context menu item, or press the *Ins* key;
- define the field properties using the Field Editor dialog.

Editing Fields

To edit an existing table field:
- open the table in Table Editor;
- proceed to the **Fields** tab there;
- right-click the field and select the **Edit Field <field_name>** context menu item, or simply double-click the field;
- edit the field properties using the Field Editor dialog.

To change the order of table fields:
- open the table in Table Editor;
- proceed to the **Fields** tab there;
- right-click a field and select the **Reorder Fields <field_name>** context menu item;
- edit fields' order using the Reorder Fields dialog.

Dropping Fields

To drop a table field:
- open the table in Table Editor;
- proceed to the **Fields** tab there;
- right-click the field and select the **Drop Field <field_name>** context menu item;
- confirm dropping in the dialog window.

© 2008 EMS Database Management Solutions

5.4.1.3.1 Field Editor

Field Editor allows you to specify field definition and set field properties. It opens automatically when you create a new field and is available on editing an existing one (see Create field and Edit field for details).

To open a field in **Field Editor**, double-click it in the DB Explorer tree, or use the **Edit Field...** item of the context menu within the Fields tab of Table Editor.

- Setting field name and type
- Setting field identity
- Setting field defaults
- Setting field description

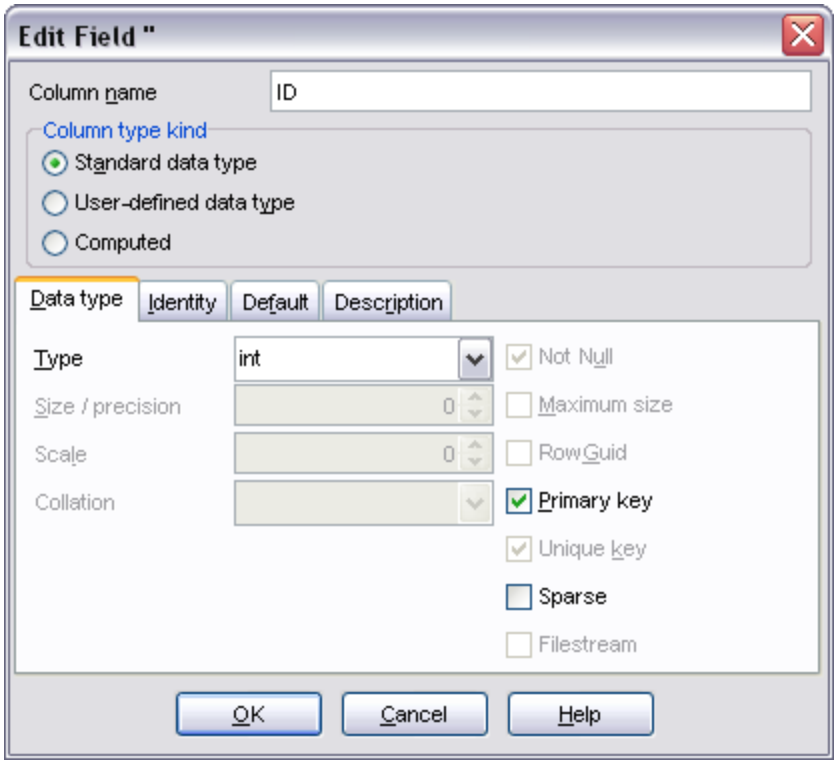
5.4.1.3.1.1 Setting field name and type

Column name
Enter a name for the new field, or modify the name of the field being edited. Note that the name of a field must be unique among all the field names in the table.

The **Data type** tab defines the type of the field data.

Column type kind
This group allows you to select whether the field being create/edited is based on a standard data type, a user-defined type, or is a computed column.

Standard data type
Specifies that the field is based upon a built-in SQL Server data type.



© 2008 EMS Database Management Solutions

Type
Here you can set the field type by selecting it from the drop-down list of the standard SQL Server data types.

Size
Specify the size value (for certain types). Check the **Maximum size** option to use the maximum values set by SQL Server.

Scale
Defines the maximum number of decimal digits that can be stored to the right of the decimal point of the value (for numeric and decimal data types).

Collation
Use the drop-down list to specify the collation for text and char columns. By default, the column is assigned the default collation of the database.

Not NULL
Check this option to specify that the values for the column should never contain a null value. *NOT NULL* affects all INSERT and UPDATE operations on a column.

RowGuid
This option specifies that the *ROWGUIDCOL* property is added to the column. *ROWGUIDCOL* indicates that the column is a row GUID column.

Primary key
Check this option to include the field into the primary key. Note that if you include a field to a primary key, you should also make it *Not Null*.
A table typically has a column or combination of columns that contain values that uniquely identify each row in the table. This column, or columns, is called the primary key (PK) of the table and enforces integrity of the table.

Unique key
Check this option to create a unique key on the field that provides entity integrity for a particular column or columns using a unique index.

Sparse
Check this option to make the column a Sparse column. Sparse columns are ordinary columns that have an optimized storage for null values. Sparse columns reduce the space requirements for null values at the cost of more overhead to retrieve nonnull values.

Filestream
Check this option to store the data from this column in the Filestream filegroup. This option is available if any filestream file group is specified for the table.

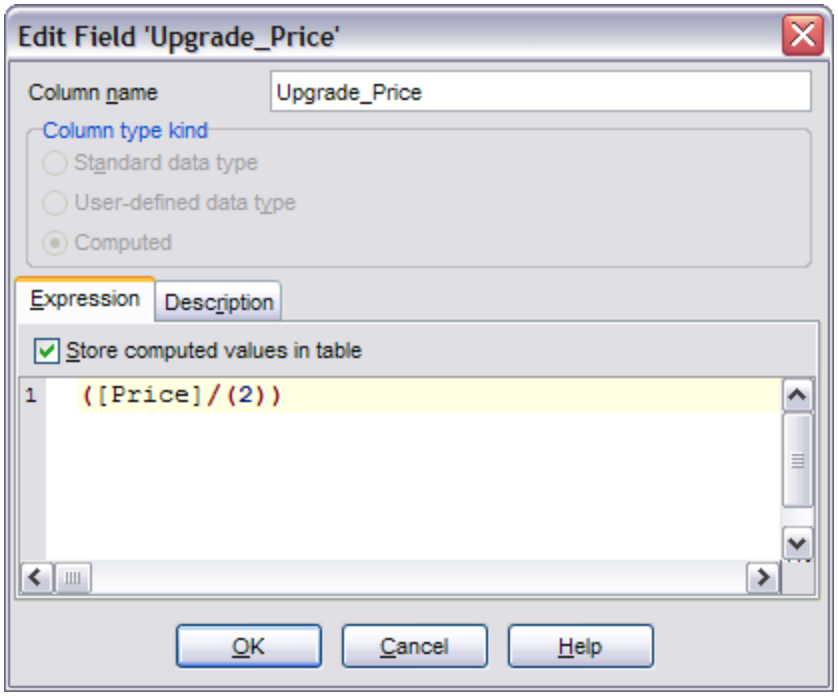
User-defined data type
Specifies that the field is based upon a user-defined data type.

Type
Use the drop-down list to select the user-defined type for the field being created/edited.

Computed

© 2008 EMS Database Management Solutions

Specifies that the field is computed from an expression that can use other columns in the same table.



Store computed values in table
This option determines whether computed values are stored in the table or the column is not physically stored in the table.

Specify an expression that defines the value of the computed column. For example, a computed column can have the following definition:
*cost AS price * qty*.
The expression can be a noncomputed column name, constant, function, variable, and any combination of these connected by one or more operators. The expression cannot be a subquery or include an alias data type.

5.4.1.3.1.2 Setting field identity

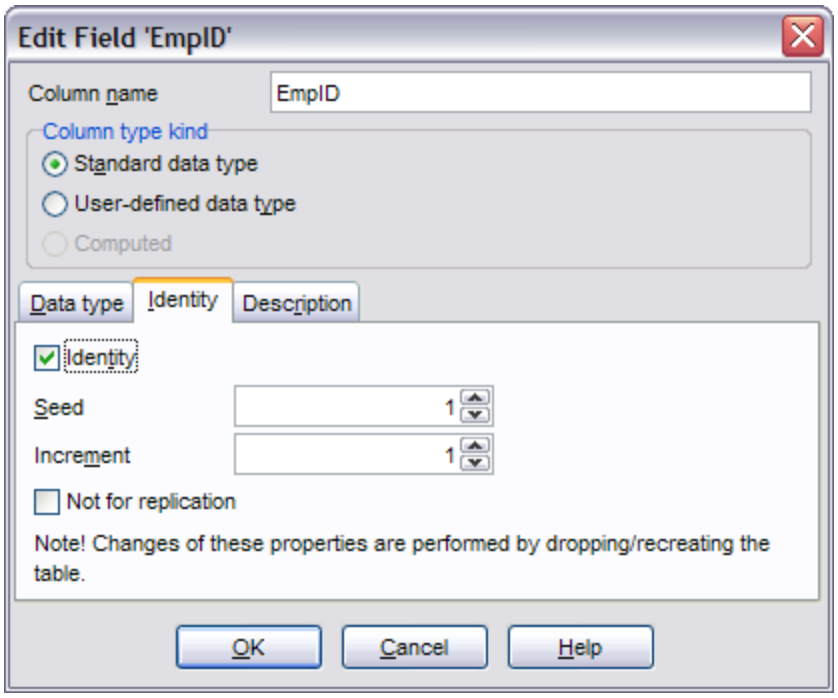
The **Identity** tab is only available for integer field types and for user-defined types based on integer types. Using this tab you can create an identity column.

Identity
This option specifies that the new column is an identity column.

Seed
Specify the value used for the first row that is loaded into the table.

Increment
Specify the value added to the identity value of the previous row that is loaded.

© 2008 EMS Database Management Solutions



Not for replication
This option allows you to specify that the Identity property should not be enabled when a replication agent modifies the table that contains the identity column.

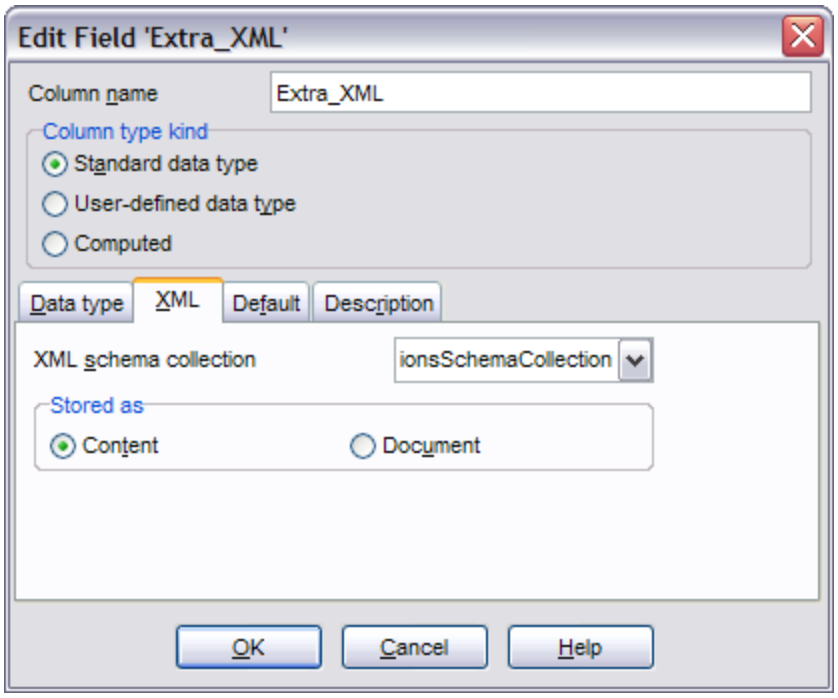
Important: This operation is performed by dropping/recreating the table.

5.4.1.3.1.3 Setting XML properties

The **Identity** tab is only available for XML field type and for user-defined types based on XML type.

The **xml** data type is a built-in data type in SQL Server and is somewhat similar to other built-in types such as *int* and *varchar*. As with other built-in types, you can use the **xml** data type as a column type when you create a table as a variable type, a parameter type, a function-return type, or in CAST and CONVERT.

© 2008 EMS Database Management Solutions



XML schema collection
Use the drop-down list to select an XML Schema Collection to be used by this field.

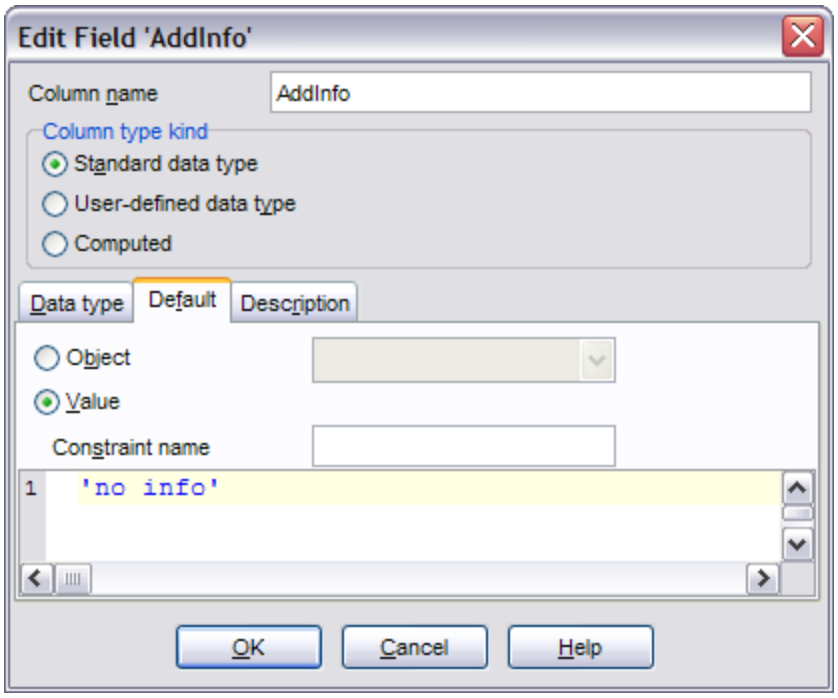
The **XML schema collection** stores the imported XML schemas.
The schemas in the collection are used to validate the XML instances. By default, instances stored in the typed xml column are stored as XML content. For typed xml columns, you can constrain the column to allow only single, top-level elements for each instance stored in it: You can do this by selecting the **Document** option.

The **Stored as** group allows you to select the way your xml data will be stored: as **Content** or as a **Document**.

5.4.1.3.1.4 Setting field defaults

The **Default** tab allows you to define the default value for the column. You can either select the default object from the list of database defaults or define a default value for the column. The second option is not available for the **Computed** field type.

© 2008 EMS Database Management Solutions

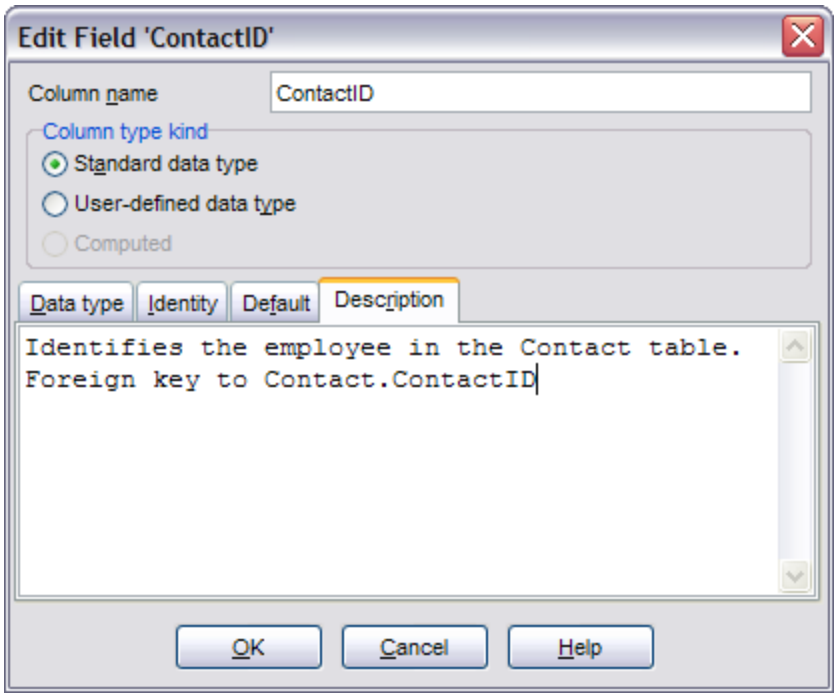


Constraint name
This box allows you to set a name for the constraint which is automatically created when you set a default value for the data field. If you leave this field empty, the constraint will be named by the server.

5.4.1.3.1.5 Setting field description

The **Description** tab allows you to enter optional text as a description for the field.

© 2008 EMS Database Management Solutions



Note: Changing object description is performed by means of *sp_updateextendedproperty 'MS_Description'*.

5.4.1.4 Foreign Keys

A **Foreign key** constraint (also called a *referential integrity constraint*) designates a column as the Foreign key and establishes a relationship between that foreign key and a specified *Primary* or *Unique* key called the *referenced key*. A composite Foreign key designates a combination of columns as the foreign key.

Table Foreign keys are managed within the **Foreign Keys** tab of Table Editor.

Creating Foreign Keys

To create a new Foreign key:
- open the table in Table Editor;
- proceed to the **Foreign Keys** tab there;
- right-click the tab area and select the **New Foreign Key** context menu item, or press the *Ins* key;
- define the Foreign key properties using the Foreign Key Editor dialog.

Editing Foreign Keys

To view/edit an existing Foreign key:
- open the table in Table Editor;
- proceed to the **Foreign Keys** tab there;
- right-click the Foreign key to edit and select the **Edit Foreign Key <foreign_key_name>** context menu item, or simply double-click the Foreign key;
- edit the Foreign key properties using the Foreign Key Editor dialog.

© 2008 EMS Database Management Solutions

Dropping Foreign Keys

To drop a Foreign key:
- open the table in Table Editor;
- proceed to the **Foreign Keys** tab there;
- right-click the Foreign key and select the **Drop Foreign Key <foreign_key_name>** context menu item;
- confirm dropping in the dialog window.

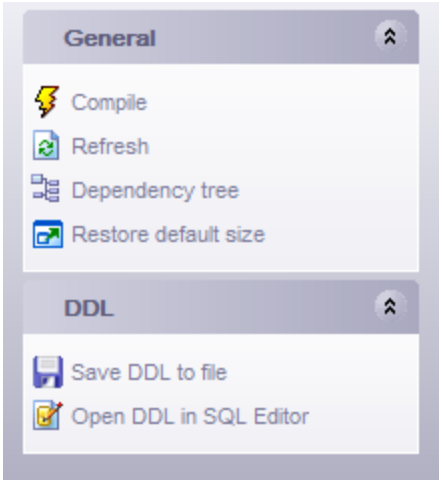
5.4.1.4.1 Foreign Key Editor

Foreign Key Editor allows you to specify foreign key definition and set foreign key properties. It opens when you create a new foreign key or edit an existing one (see Create Foreign Key and Edit Foreign Key for details).

To open a foreign key in **Foreign Key Editor**, double-click it in the DB Explorer tree, or use the **Edit Foreign Key...** item of the context menu within the Foreign Keys tab of Table Editor.
- Using Navigation bar
- Editing foreign key definition
- Browsing object dependencies
- Editing foreign key description
- Viewing DDL definition

5.4.1.4.1.1 Using Navigation bar

The **Navigation bar** of **Foreign Key Editor** allows you to:
- compile the foreign key (if it is being created/modified);
- save the object description (if it has been modified);
- refresh the content of the active tab;
- view the dependency tree for the foreign key;
- restore the default size of the editor window.



© 2008 EMS Database Management Solutions

Depending on the current tab selection, the **Navigation bar** expands to one or more additional panes with tab-specific actions that can be useful for working with the object.

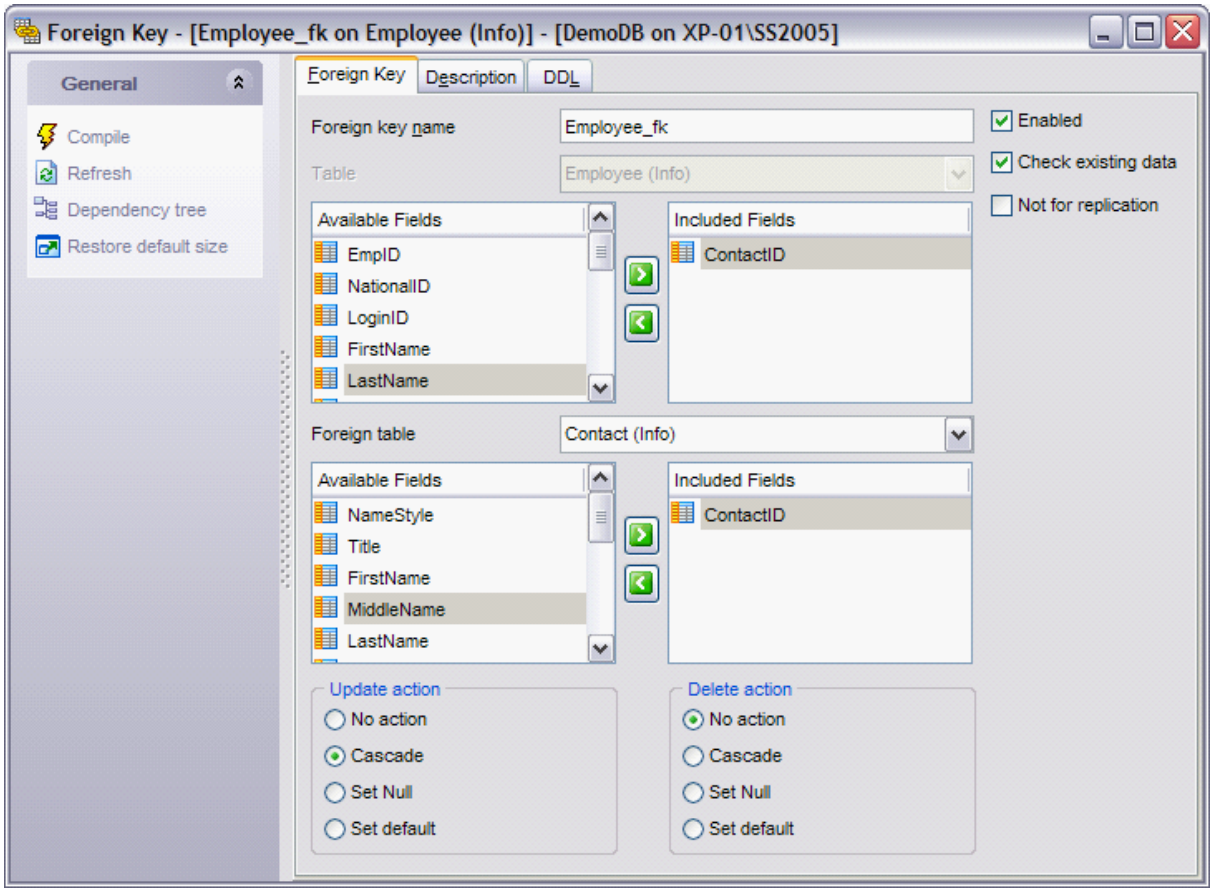
<< Back to Foreign Key Editor

5.4.1.4.1.2 Editing foreign key definition

Use the **Foreign Key** tab of **Foreign Key Editor** to create/edit a foreign key constraint and specify its properties.

Name
Enter a name for the new foreign key, or modify the name of the foreign key being edited.

Table
The drop-down list of tables allows you to select the table for which the foreign key is created.



The **Table Fields** area allows you to select Foreign key field(s).
To select a field, you need to move it from the **Available Fields** list to the **Included Fields** list. Use the buttons or drag-and-drop operations to move the fields from

© 2008 EMS Database Management Solutions

one list to another.

Foreign table
Use the drop-down list to select the foreign table.

The **Foreign Table Fields** area allows you to select the field(s) of the Foreign table. To select a field, you need to move it from the **Available Fields** list to the **Included Fields** list. Use the buttons or drag-and-drop operations to move the fields from one list to another.

If the referenced column(s) are changed frequently, it may be wise to add an index to the foreign key column so that referential actions associated with the foreign key column were performed more efficiently. See Indexes for details.

Update action / Delete action
- *No action*
  Produce an error indicating that the deletion or update would 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.

Enabled
Enables/disables the foreign key immediately after it is created.

Check existing data
This option specifies whether the data in the table is or is not validated against a newly added or re-enabled foreign key.

Not for replication
This option allows you to specify that the foreign key should not be enabled when a replication agent modifies the table that is involved in the foreign key.

5.4.1.5 Checks

A **check** specifies 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. Should 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. A check constraint specified as a column constraint should reference that column's value only, while an expression appearing in a table constraint may reference multiple columns.

Table checks are managed within the **Checks** tab of Table Editor.

© 2008 EMS Database Management Solutions

Creating Checks

To create a new check:
- open the table in Table Editor;
- proceed to the **Checks** tab there;
- right-click the tab area and select the **New Check Constraint** context menu item, or press the *Ins* key;
- define the check properties using the Check Editor dialog.

Editing Checks

To view an existing check:
- open the table in Table Editor;
- proceed to the **Checks** tab there;
- right-click the check and select the **Edit Check <check_name>** context menu item, or simply double-click the check;
- edit the check properties using the Check Editor dialog.

Dropping Checks

To drop a check:
- open the table in Table Editor;
- proceed to the **Checks** tab there;
- right-click the check and select the **Drop Check <check_name>** context menu item;
- confirm dropping in the dialog window.

5.4.1.5.1 Check Editor

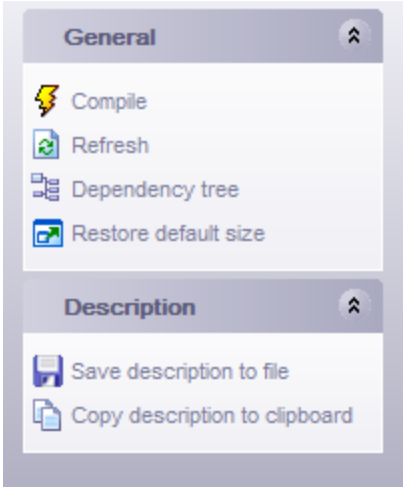
Check Editor allows you to specify check definition and set check properties. It opens automatically when you create a new check and is available on editing an existing one (see Create check and Edit check for details).

To open a check constraint in **Check Editor**, double-click it in the DB Explorer tree, or use the **Edit Check...** item of the context menu within the Checks tab of Table Editor.
- Using Navigation bar
- Editing check definition
- Browsing object dependencies
- Editing check description
- Viewing DDL definition

5.4.1.5.1.1 Using Navigation bar

The **Navigation bar** of **Check Editor** allows you to:
- compile the check (if it is being created/modified);
- save the object description (if it has been modified);
- refresh the content of the active tab;
- view the dependency tree for the check;
- restore the default size of the editor window.

© 2008 EMS Database Management Solutions



Depending on the current tab selection, the **Navigation bar** expands to one or more additional panes with tab-specific actions that can be useful for working with the object.

<< Back to Check Editor

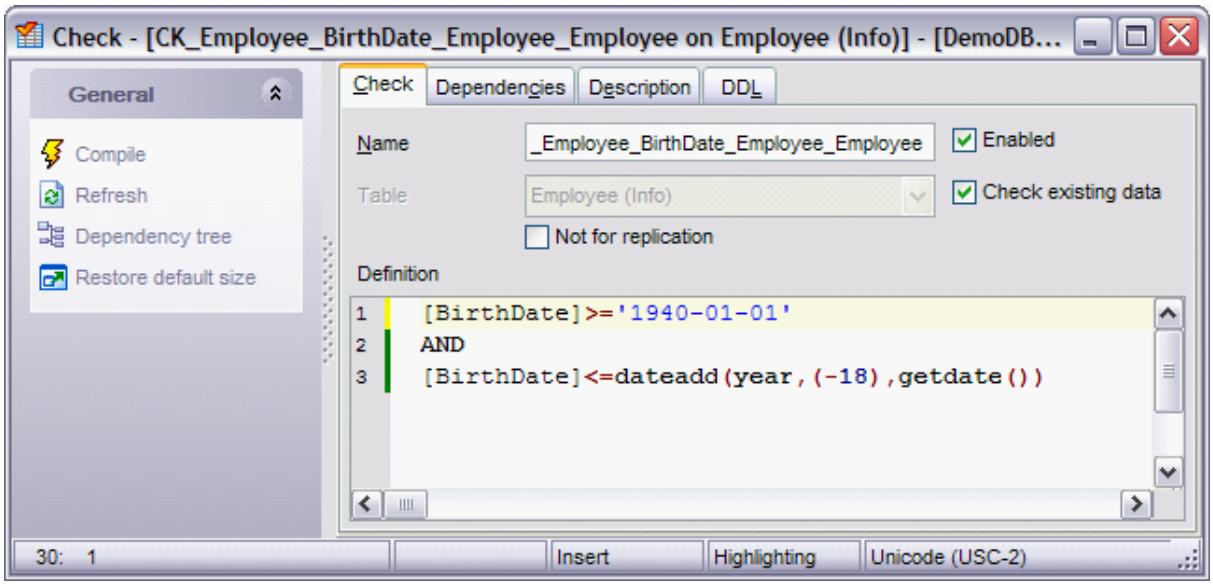
5.4.1.5.1.2 Editing check definition

Use the **Check** tab of **Check Editor** to create/edit a check constraint and specify its properties.

Name
Enter a name for the new check, or modify the name of the check being edited.

Table
The drop-down list of tables allows you to select the table for which the check is created.

© 2008 EMS Database Management Solutions



Enabled
Enables/disables the check immediately after it is created.

Check existing data
This option specifies whether the data in the table is or is not validated against a newly added or re-enabled check.

Not for replication
This option allows you to specify that the check should not be enabled when a replication agent modifies the table that is involved in the check.

Definition
This area represents the condition implied by the check constraint.
You can specify any logical (Boolean) expression that returns *TRUE* or *FALSE* based on the logical operators. For example, *salary >= 15000 AND salary <= 100000*.

For your convenience the **syntax highlight**, **code completion** and a number of other features for efficient SQL editing are implemented. For details see Working with SQL Editor area and Using the context menu.

5.4.1.6 Indexes

Table indexes are managed within the **Indices** tab of Table Editor.

Creating Indexes

To create a new table index:
- open the table in Table Editor;
- proceed to the **Indexes** tab there;
- right-click the tab area and select the **New Index** context menu item, or press the *Ins* key;

© 2008 EMS Database Management Solutions

- define the index properties using the Index Editor dialog.

Editing Indexes

To edit an existing table index:
- open the table in Table Editor;
- proceed to the **Indexes** tab there;
- right-click the index and select the **Edit Index** context menu item, or simply double-click the index;
- edit the index properties using the Index Editor dialog.

Dropping Indexes

To drop a table index:
- open the table in Table Editor;
- proceed to the **Indexes** tab there;
- right-click the index and select the **Drop Index** context menu item;
- confirm dropping in the dialog window.

5.4.1.6.1 Index Editor

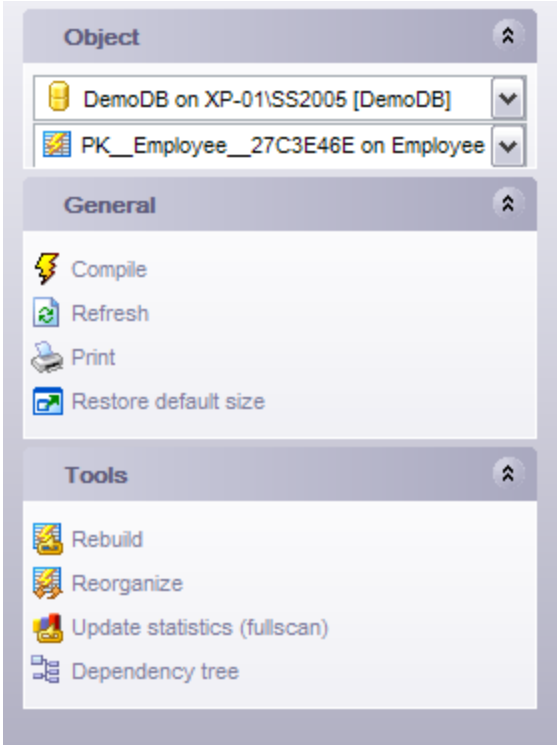
Index Editor allows you to specify index definition and set index properties. It opens automatically when you create a new index and is available on editing an existing one (see Create Index and Edit Index for details).

To open an index in **Index Editor**, double-click it in the DB Explorer tree, or use the **Edit Index...** item of the context menu within the Indices tab of Table Editor.
- Using Navigation bar
- Editing index definition
- Defining index storage
- Viewing index info
- Editing index description
- Viewing DDL definition

5.4.1.6.1.1 Using Navigation bar

The **Navigation bar** of **Index Editor** allows you to:
- select the database where the object is located;
- select the index for editing;
- compile the object (if it is being created/modified);
- save the object description (if it has been modified);
- refresh the content of the active tab;
- print metadata of the object;
- view the dependency tree for the index;
- restore the default size of the editor window;
- rebuild the index using Indices Management Wizard;
- reorganize the index using Indices Management Wizard;
- update statistics using Statistics Update Wizard;

© 2008 EMS Database Management Solutions



Depending on the current tab selection, the **Navigation bar** expands to one or more additional panes with tab-specific actions that can be useful for working with the object.

[<< Back to Index Editor]

5.4.1.6.1.2 Editing index definition

Use the **Index** tab of **Index Editor** to create/edit an index on a specified table or view, or an XML index on a specified table, and specify index properties.

Name
Enter a name for the new index, or modify the name of the index being edited.

Table or view
The drop-down list of tables and views allows you to select the table or view to be indexed.

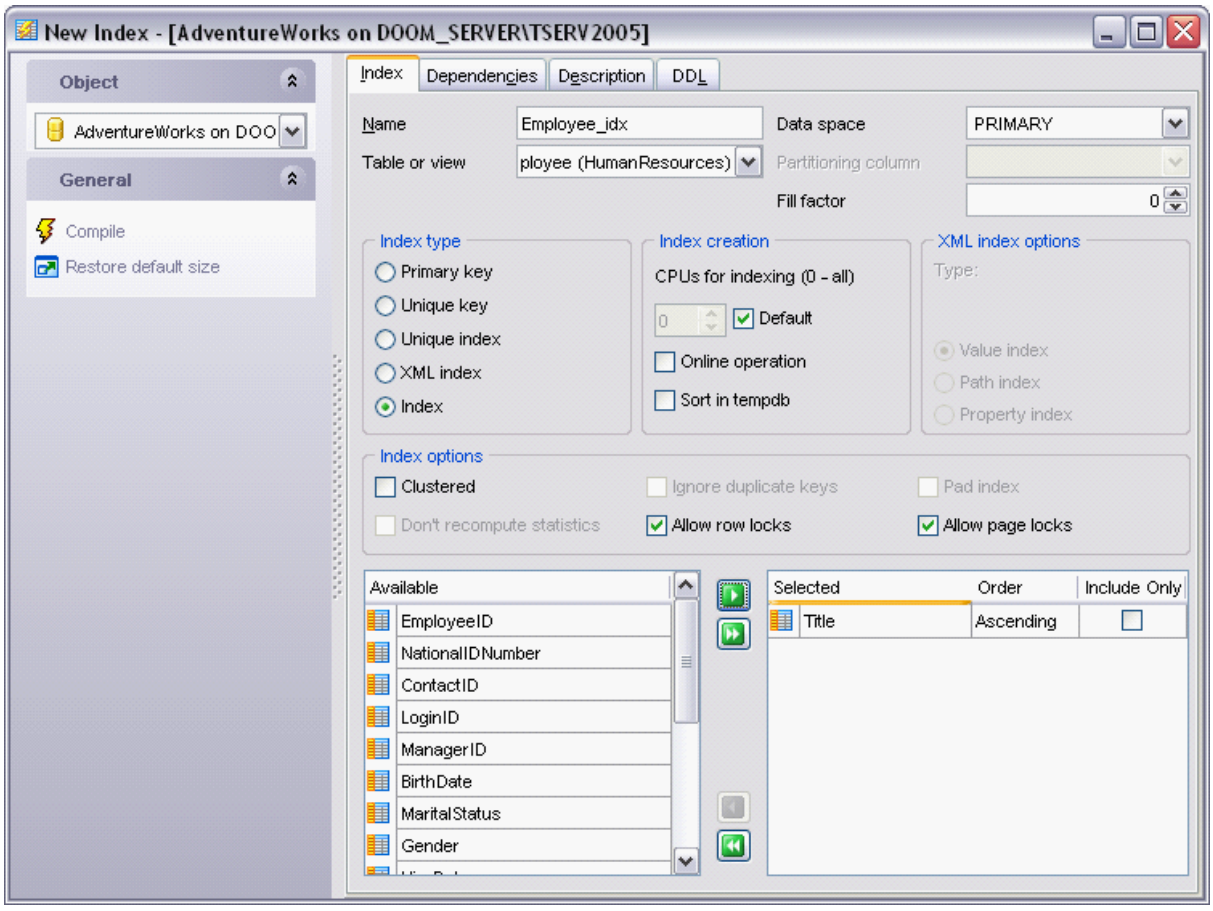
Data space
Use the drop-down list to specify the new index data storage (a file group or a partition scheme that defines the filegroups onto which the partitions of a partitioned index will be mapped).
Note: When creating an index you need not specify the data space; the data space of the table is used by default.

Partitioning column
This drop-down list is only available if a partition scheme is specified for **Data space**.

© 2008 EMS Database Management Solutions

Select a field that will be used as the partitioning column for the table.

Fill factor
Specify a percentage that indicates how full the Database Engine should make the leaf level of each index page during index creation or rebuild.



Index type

Primary Key
This selection indicates that the primary key index is created.

Unique key
This selection indicates that the unique key index is created (no two rows are permitted to have the same index value). A clustered index on a view must be unique.

Unique index
This selection indicates that the unique index is created (ensures that the index key contains no duplicate values and therefore each row in the table or view is in some way unique). Both clustered and nonclustered indexes can be unique.

© 2008 EMS Database Management Solutions

XML index
This selection indicates that the XML index on the specified XML column is created.

Index creation

CPUs for indexing
This option allows you to limit the number of processors used in a parallel plan execution. Set a value to restrict the maximum number of processors used in a parallel index operation to the specified number or fewer based on the current system workload. If *0* is specified, the actual number of processors is used.

Default
If this option is checked, the default setting for the number of processors is applied.

Online operation
This option specifies whether underlying tables and associated indexes are available for queries and data modification during the index operation.

Sort in tempdb
This option specifies whether temporary sort results should be stored in **tempdb**.

XML Index options
Select the XML index type:
Value index
This selection indicates creation of a secondary XML index on columns where key columns are (node value and path) of the primary XML index. If your workload involves querying for values from XML instances without knowing the element or attribute names that contain the values, VALUE index may be useful.
Path index
This selection indicates creation of a secondary XML index on columns built on path values and node values in the primary XML index. If your queries generally specify path expressions on xml type columns, a PATH secondary index may be able to speed up the search.
Property index
This selection indicates creation of a secondary XML index on columns (PK, path and node value) of the primary XML index where PK is the primary key of the base table. Queries that retrieve one or more values from individual XML instances may benefit from a PROPERTY index.

Index options

Clustered
Use this option to create an index in which the logical order of the key values determines the physical order of the corresponding rows in a table. The bottom, or leaf, level of the clustered index contains the actual data rows of the table. A table or view is allowed one clustered index at a time.

Ignore duplicate keys
This option is available if the Unique option is checked. This option specifies the error response to duplicate key values in a multiple-row insert operation on a unique clustered

© 2008 EMS Database Management Solutions

or unique nonclustered index.

Pad index
This option determines whether the percentage of free space that is specified by fillfactor is applied to the intermediate-level pages of the index.

Do not recompute statistics
This option specifies that out-of-date index statistics are not automatically recomputed.

In cases when access patterns are well understood and consistent, limiting the locking levels available for an index can be beneficial. Disallowing a locking level can affect the concurrency for a table or index.

Allow row locks
This option specifies whether row locks are allowed when accessing the index.

Allow page locks
This option specifies whether page locks are allowed when accessing the index.

Index filter
Use this field to set the filter for the data from the field(s) included in this index. Available only in SQL Server 2008.

Fields for index
To include field(s) in the index, you need to move it from the **Available** list to the **Selected** list. Use the buttons or drag-and-drop operations to move the fields from one list to another.

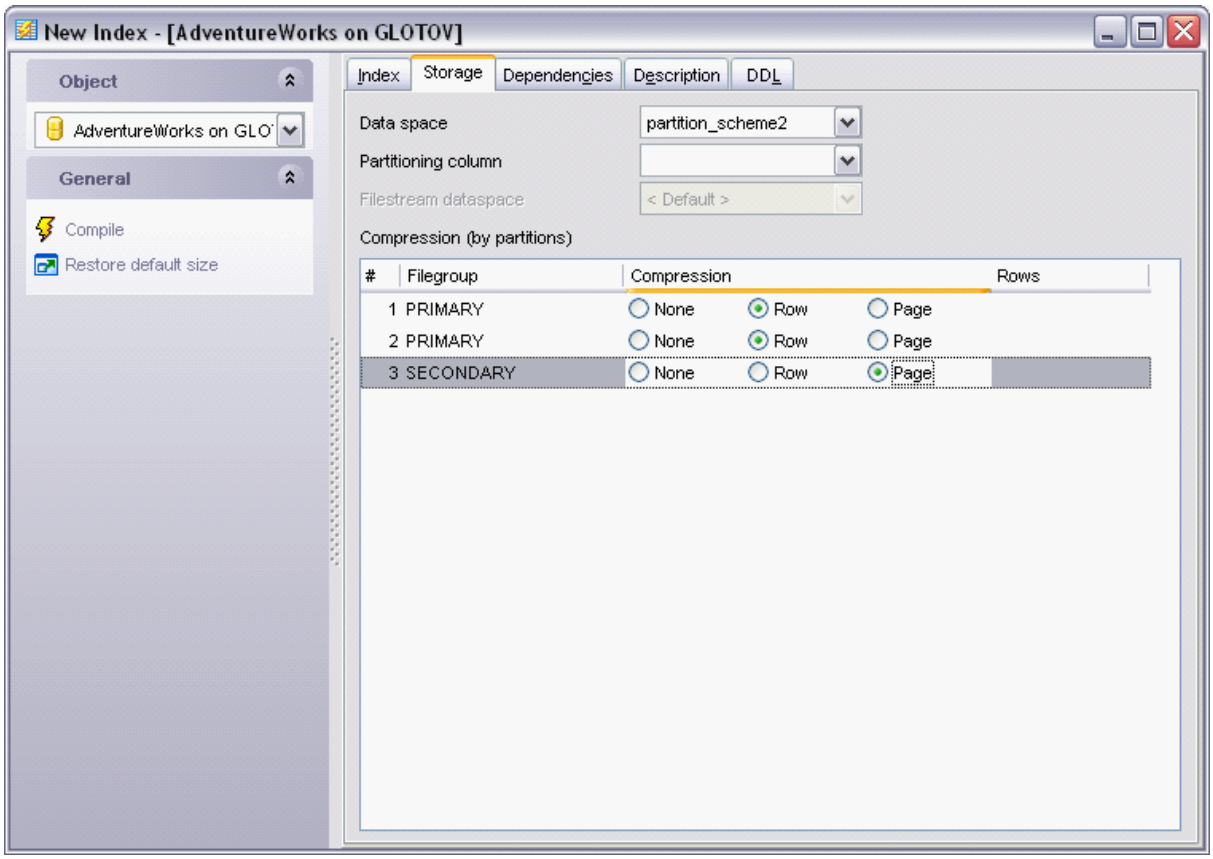
To change the sorting order for a field, select the field in the **Selected** list and change the **Order** value.

Set the **Include Only** flag to specify that the non-key column is added to the leaf level of the nonclustered index.

5.4.1.6.1.3 Defining index storage

At the **Storage** tab of the **Index Editor** you need to define the storage for the index. This tab is available only in SQL Server 2008 or higher.

© 2008 EMS Database Management Solutions



Data space
Use the drop-down list to specify the new index data storage (a file group or a partition scheme that defines the filegroups onto which the partitions of a partitioned index will be mapped).
Note: When creating an index you need not specify the data space; the data space of the table is used by default.

Partitioning column
This drop-down list is only available if a partition scheme is specified for **Data space**. Select a field that will be used as the partitioning column for the table.

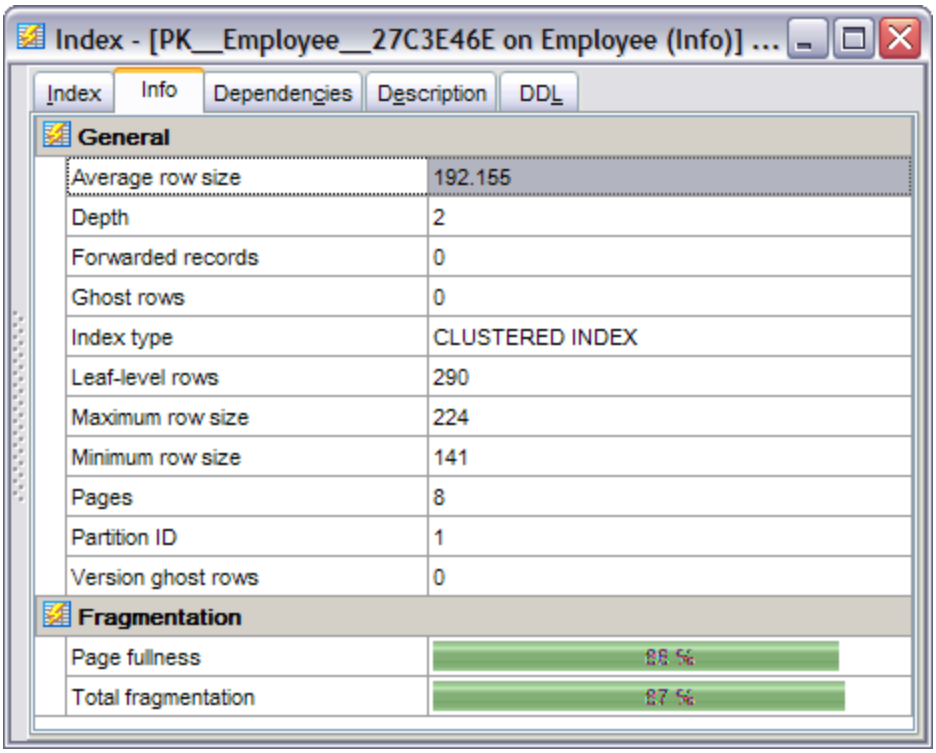
Filestream dataspace
Select the needed Filestream filegroup from the drop-down list. This option is available only in SQL Server 2008.

Compression (by partitions)
Use this section to set the compression options for the index. Compression type can be different for each partition. Set the index compression to *Row* or *Page*. Set *None* to leave a partition uncompressed.

5.4.1.6.1.4 Viewing index info

The **Info** tab of **Index Editor** allows you to view the index summary:
General: *Average row size*, *Depth*, *Forwarded records*, *Ghost rows*, *Index type*, *Leaf-level rows*, *Maximum row size*, *Minimum row size*, *Pages*, *Partition ID*, *Version ghost rows*;
Fragmentation: *Page fullness*, *Total fragmentation*.

© 2008 EMS Database Management Solutions



5.4.1.7 Triggers

A **trigger** is a special kind of stored procedure that automatically executes when an event occurs in the database server.

DML triggers execute when a user tries to modify data through a data manipulation language (DML) event. DML events are *INSERT*, *UPDATE*, or *DELETE* statements on a table or view.

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

Creating Triggers

To create a new trigger:
- open the table in Table Editor;
- proceed to the **Triggers** tab there;
- right-click the tab area and select the **New Trigger** context menu item, or press the *Ins* key;
- define the trigger properties using the Trigger Editor dialog.

Editing Triggers

To edit an existing trigger:
- open the table in Table Editor;
- proceed to the **Triggers** tab there;

© 2008 EMS Database Management Solutions

- right-click the trigger and select the **Edit Trigger <trigger_name>** context menu item, or simply double-click the trigger;
- edit the trigger properties using the Trigger Editor dialog.

Dropping Triggers

To drop a trigger:
- open the table in Table Editor;
- proceed to the **Triggers** tab there;
- right-click the trigger and select the **Drop Trigger <trigger_name>** context menu item;
- confirm dropping in the dialog window.

5.4.1.7.1 Trigger Editor

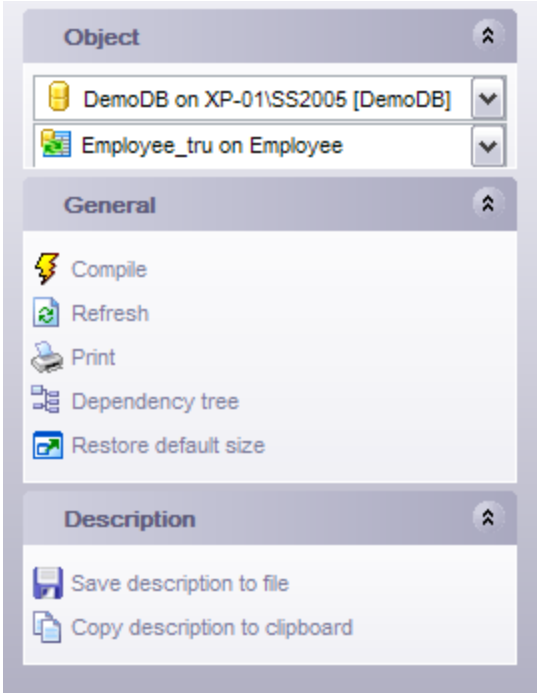
Trigger Editor allows you to specify trigger definition and set trigger properties. It opens automatically when you create a new trigger and is available on editing an existing one (see Create Trigger and Edit Trigger for details).

To open a trigger in **Trigger Editor**, double-click it in the DB Explorer tree, or use the **Edit Trigger...** item of the context menu within the Triggers tab of Table Editor.
- Using Navigation bar
- Editing trigger definition
- Browsing object dependencies
- Editing trigger description
- Viewing DDL definition

5.4.1.7.1.1 Using Navigation bar

The **Navigation bar** of **Trigger Editor** allows you to:
- select the database where the object is located;
- select the trigger for editing;
- compile the object (if it is being created/modified);
- save the object description (if it has been modified);
- refresh the content of the active tab;
- print metadata of the object;
- view the dependency tree for the trigger;
- restore the default size of the editor window.

© 2008 EMS Database Management Solutions



Depending on the current tab selection, the **Navigation bar** expands to one or more additional panes with tab-specific actions that can be useful for working with the object.

[<< Back to Trigger Editor]

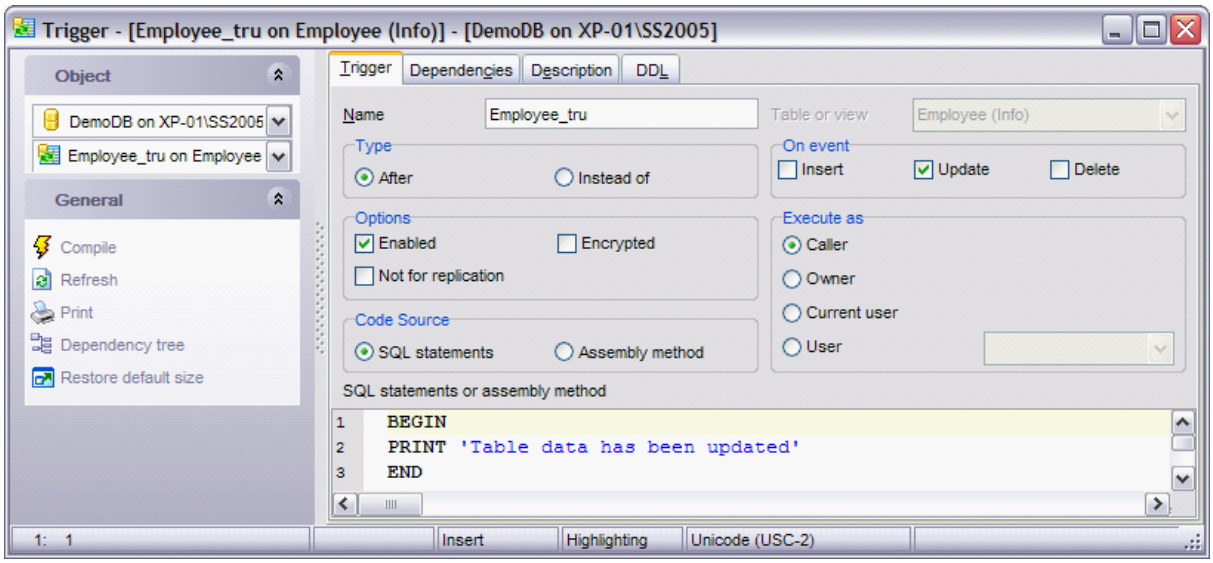
5.4.1.7.1.2 Editing trigger definition

Use the **Trigger** tab of **Trigger Editor** to create/edit a DML trigger and specify its properties.

Name
Enter a name for the new trigger, or modify the name of the trigger being edited.

Table or view
The drop-down list of tables and views allows you to select the table or view on which the trigger is executed.

© 2008 EMS Database Management Solutions



Type
Select the trigger behaviour type:
After
This selection specifies that the trigger is fired only when all operations specified in the triggering SQL statement have executed successfully.
Instead of
This selection specifies that the trigger is executed instead of the triggering SQL statement.

On event
Specify the data modification statements that activate the trigger when it is tried against this table or view: *Insert*, *Update* or *Delete*. At least one option must be specified.

Options

Enabled
Enables/disables the trigger immediately after it is created. Enabling a trigger causes it to fire when any Transact-SQL statements on which it was originally programmed are executed. A disabled trigger still exists as an object in the current database, but does not fire.

Not for replication
This option allows you to specify that the trigger should not be executed when a replication agent modifies the table that is involved in the trigger.

Encrypted
You can check this option to encrypt the text of the *CREATE TRIGGER* statement if you want to ensure that other users cannot view the trigger definition.

Execute as

© 2008 EMS Database Management Solutions

Specify the security context under which the trigger is to be executed: *Caller*, *Owner*, *Current user* or *User* (select which user account should be used to validate permissions on the objects that are referenced by the trigger).

Code Source
Select the trigger code source: it can be a direct set of SQL statements or an external procedure call from a .NET assembly or DLL.

SQL statements or assembly method
This area allows you to set the trigger conditions and actions. Trigger conditions specify additional criteria that determine whether the tried DML statements cause the trigger actions to be performed.
The trigger actions take effect when the DML operation is performed.

5.4.2 Views

A **view** is a logical table based on one or more tables or views. A view contains no data itself. The tables upon which a view is based are called *base tables*.

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).

Creating Views

To create a new view:
- select the **Database | New Object...** main menu item;
- select **View** in the Create New Object dialog;
- define view properties using the appropriate tabs of View Editor.

Hint: To create a new view, you can also right-click the **Views** node or any object within this node in the DB Explorer tree and select the **New View** item from the context menu.

Another way to create a view is to build a query in Visual Query Builder (click the **Create view** Navigation bar item after building).

To create a new view with the same properties as one of existing views has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a view in the DB Explorer tree and select the **Duplicate View <view_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new view in, and to edit the result SQL statement for creating the view.

Editing Views

To edit an existing view:
- select the view for editing in the DB Explorer tree (type the first letters of the view

© 2008 EMS Database Management Solutions

name for quick search);
- right-click the object and select the **Edit View <view_name>** context menu item, or simply double-click the view;
- edit view definition and data using the appropriate tabs of View Editor.

To change the name of a view:
- select the view to rename in the DB Explorer tree;
- right-click the view alias and select the **Rename View <view_name>...** item from the context menu;
- edit the view name using the **Rename Object...** dialog.

Dropping Views

To drop a view:
- select the view to drop in the DB Explorer tree;
- right-click the object and select the **Drop View <view_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new view;
Ctrl+O to edit the selected view;
Ctrl+R to rename the view;
Shift+Del to drop the object from the database.

5.4.2.1 View Editor

View Editor allows you to create new views and define their properties (view name and the SELECT statement it implements). It opens automatically when you create a new view and is available on editing an existing one (see Create view and Edit view for details).

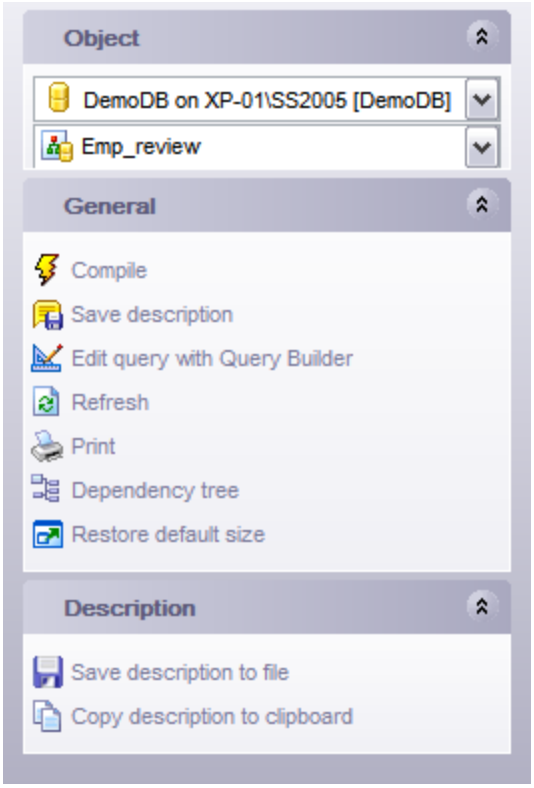
To open a view in **View Editor**, double-click it in the DB Explorer tree.
- Using Navigation bar
- Editing SQL definition
- Managing fields
- Managing indexes
- Managing triggers
- Browsing object dependencies
- Working with data
- Editing view description
- Viewing DDL definition

5.4.2.1.1 Using Navigation bar

The **Navigation bar** of **View Editor** allows you to:
- select the database where the object is located;
- select the view for editing;
- compile the object (if it is being created/modified);
- save the object description (if it has been modified);
- edit the view query using Query Builder;
- refresh the content of the active tab;

© 2008 EMS Database Management Solutions

- print metadata of the object;
- view the dependency tree for the view;
- restore the default size of the editor window.



Depending on the current tab selection, the **Navigation bar** expands to one or more additional panes with tab-specific actions that can be useful for working with the object.

<< Back to View Editor

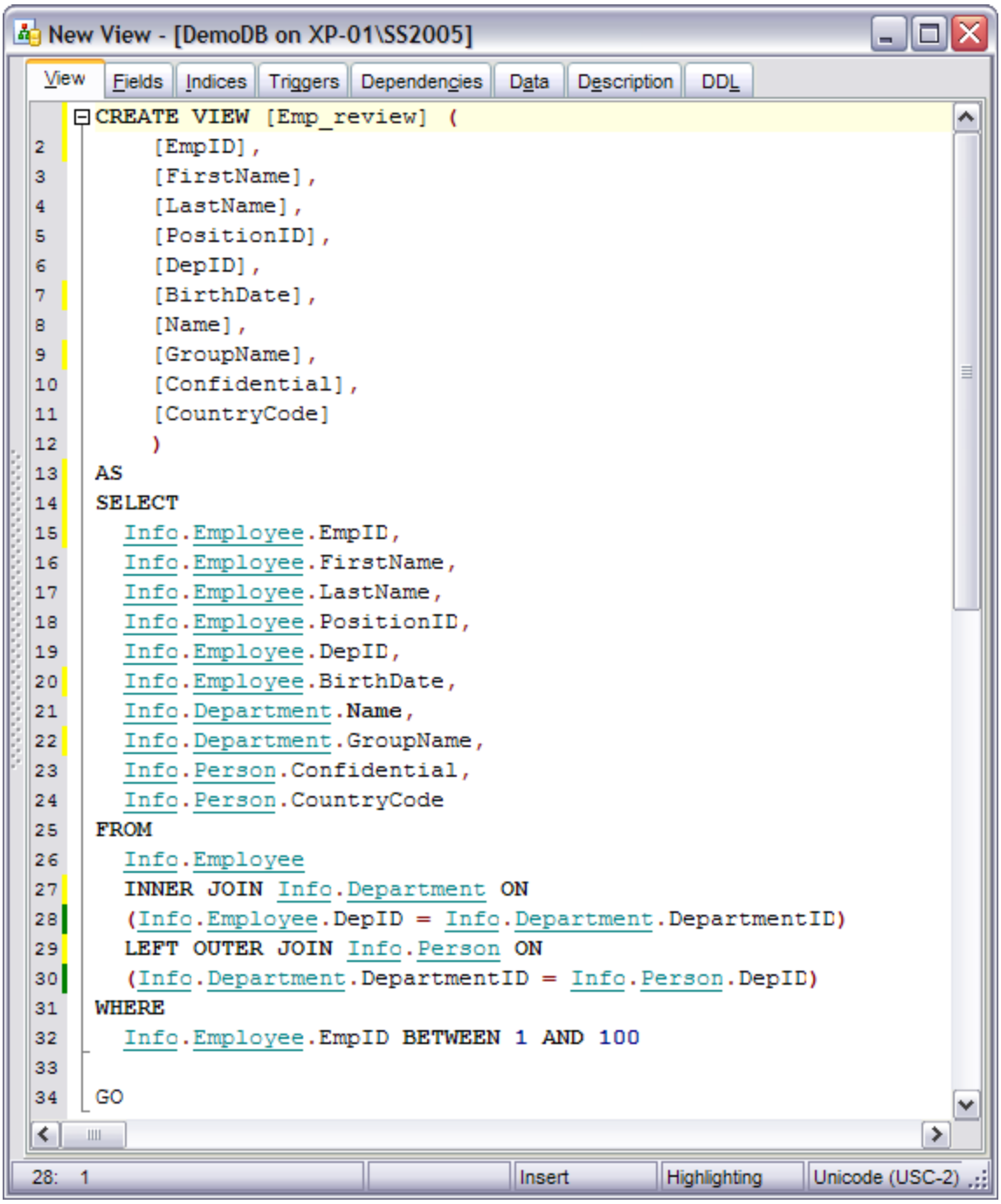
5.4.2.1.2 Editing view definition

Use the **View** tab of **View Editor** to create/edit a view and specify its definition.

This tab represents the view definition as SQL statement, hence it is enough to simply edit the whole definition using the editor area to make appropriate changes, and recompile the view.

For your convenience the **syntax highlight**, **code completion** and a number of other features for efficient SQL editing are implemented. For details see Working with SQL Editor area and Using the context menu.

© 2008 EMS Database Management Solutions



To compile a view, you can use the **Compile** item available within the Navigation bar.

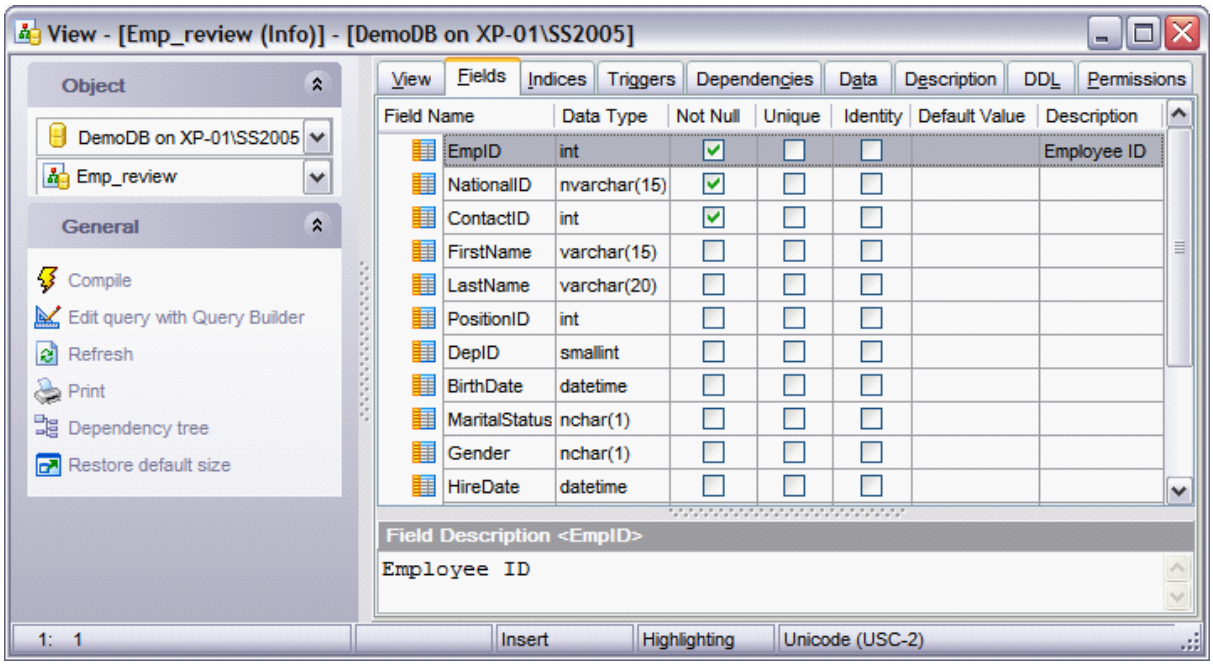
5.4.2.1.3 Managing fields

The **Fields** tab is provided for viewing fields represented in the view.

Right-click a field to display the context menu allowing you to export field name list or

© 2008 EMS Database Management Solutions

copy it to clipboard.



The **Fields** list provides the following attributes of each field of the view:
Field Name
Data Type
Not Null
Unique
Identity
Default Value
Description

For details see Fields.

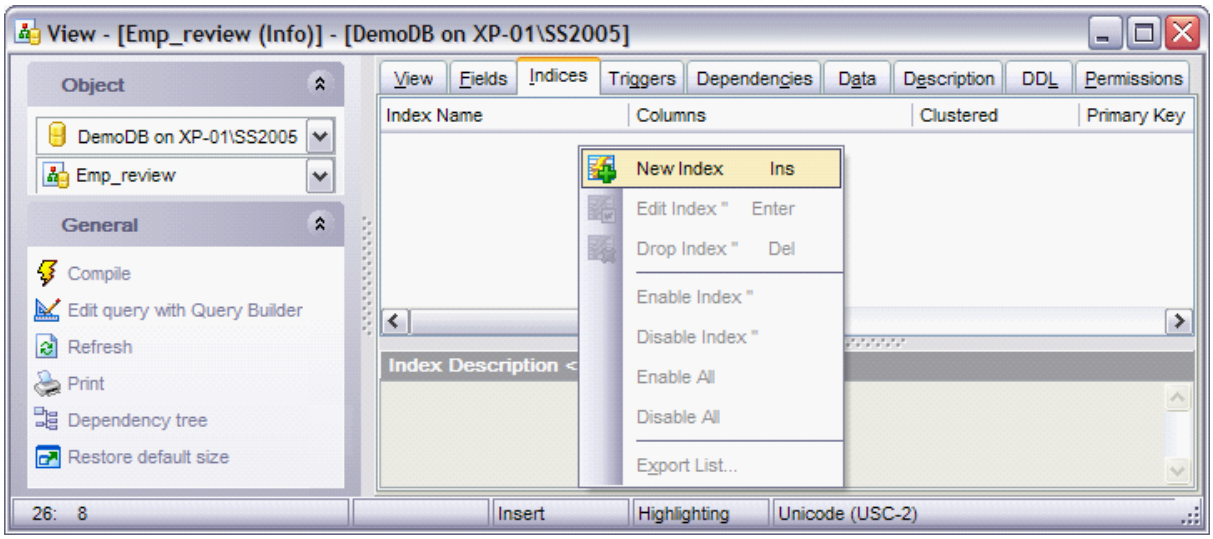
If necessary, you can also use the **Field Description** area to supply a *description* for each field.

5.4.2.1.4 Managing indexes

The **Indices** tab is provided for viewing indexes represented in the view. Double-click an index to open Index Editor for editing the index.

Right-click an index to display the context menu allowing you to *create* new, *edit*, *drop*, *enable*/*disable* the selected index, or *enable*/*disable* all indexes. Using the menu you can also export the list of the view indexes to any of supported formats.

© 2008 EMS Database Management Solutions



The **Indexes** list provides the following attributes of each index of the view:
Index Name
Columns
Clustered
Primary Key
Unique
Disabled

For details see Indexes.

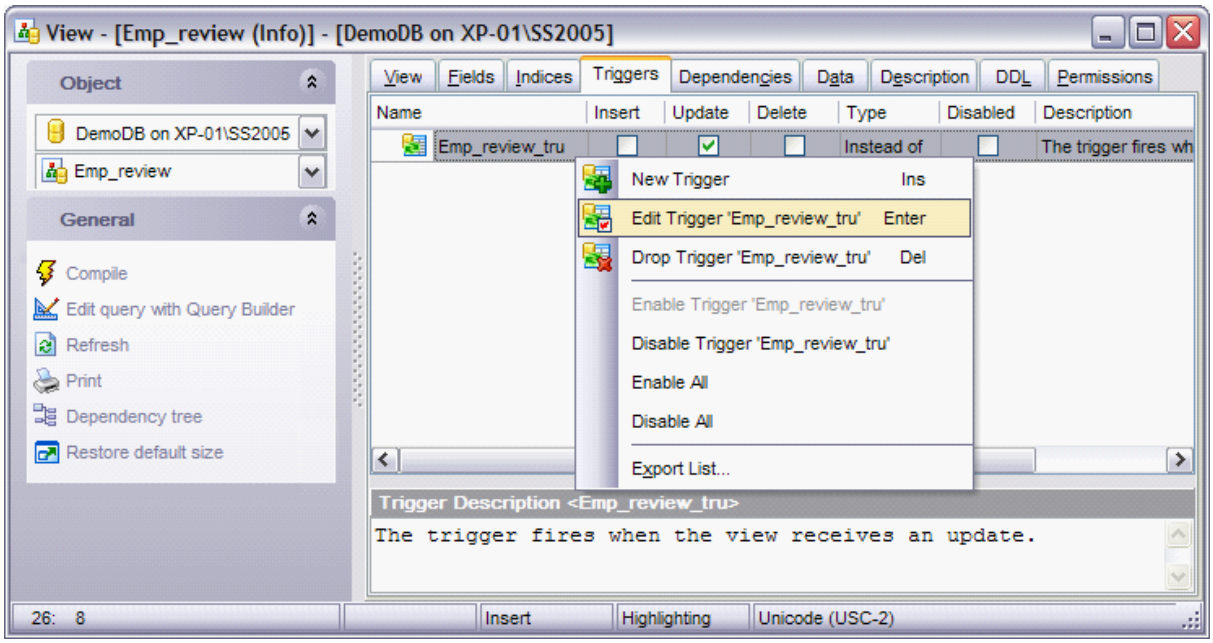
If necessary, you can also use the **Index Description** area to supply a *description* for each index.

5.4.2.1.5 Managing triggers

The **Triggers** tab is provided for managing triggers represented in the view. Double-click a trigger to open Trigger Editor for editing the trigger.

Right-click the area to display the context menu allowing you to *create* new, *edit*, *drop*, *enable*/*disable* the selected trigger, or *enable*/*disable* all triggers. Using the menu you can also export the list of the view triggers to any of supported formats.

© 2008 EMS Database Management Solutions



The **Triggers** list provides the following attributes of each trigger of the view:
Name
Insert
Update
Delete
Type
Disabled

For details see Triggers.

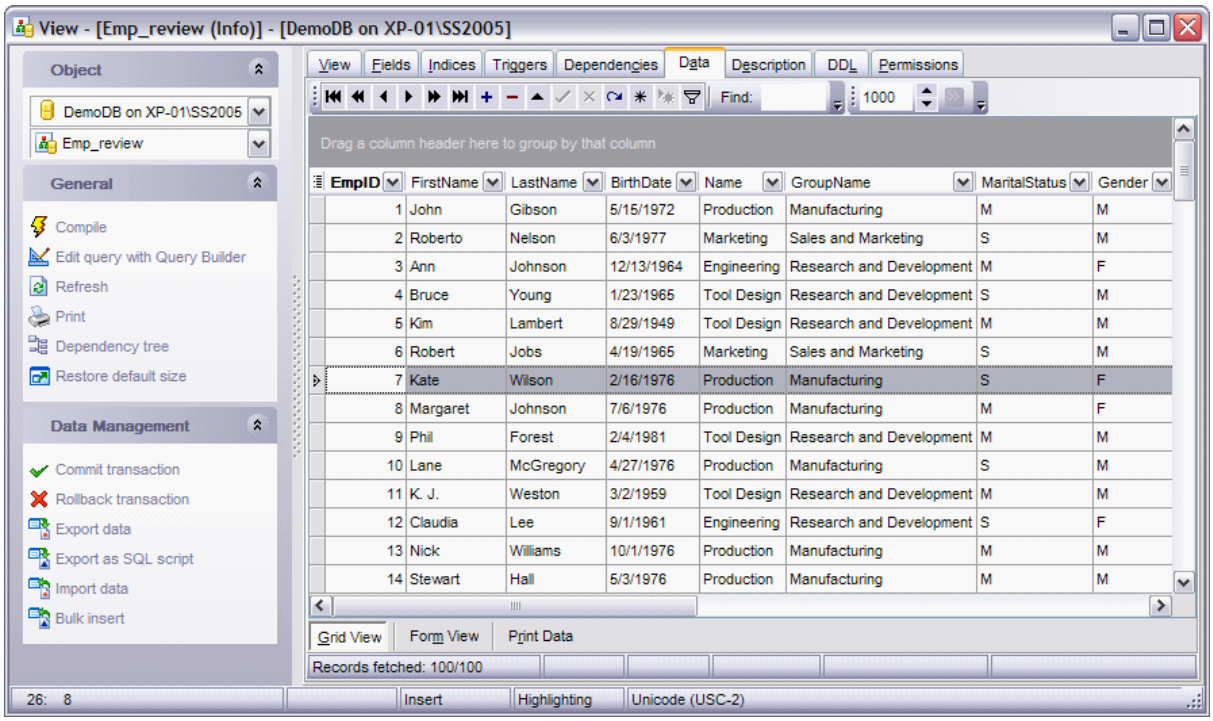
If necessary, you can also use the **Trigger Description** area to supply a *description* for each trigger.

5.4.2.1.6 Working with data

The **Data** tab displays the view data as a grid by default (see Data View for details). The context menu of this tab and the Navigation bar allow you to Export Data, Import Data, Export as SQL Script, Bulk Insert.

Data management tasks are also available through the Navigation bar of **View Editor**.

© 2008 EMS Database Management Solutions



See also:

Working with table data

5.4.3 Procedures

A **Procedure** is a set of procedural constructs and embedded SQL statements that is stored in the database and can be called by name.

When you create an application with Microsoft SQL Server 2005, the Transact-SQL programming language is the primary programming interface between your applications and the Microsoft SQL Server database. When you use Transact-SQL programs, two methods are available for storing and executing the programs. You can store the programs locally and create applications that send the commands to SQL Server and process the results, or you can store the programs as stored procedures in SQL Server and create applications that execute the stored procedures and process the results.

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.

Creating Procedures

To create a new procedure:

© 2008 EMS Database Management Solutions

- select the **Database | New Object...** main menu item;
- select **Procedure** in the Create New Object dialog;
- define procedure properties using the appropriate tabs of Procedure Editor.

Hint: To create a new procedure, you can also right-click the **Procedures** node or any object within this node in the DB Explorer tree and select the **New Procedure...** item from the context menu.

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

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

Alternatively, you can right-click a procedure in the DB Explorer tree and select the **Duplicate Procedure <procedure_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new procedure in, and to edit the result SQL statement for creating the procedure.

Editing Procedures

To edit an existing procedure:

- select the procedure for editing in the DB Explorer tree (type the first letters of the procedure name for quick search);
- right-click the object and select the **Edit Procedure <procedure_name>** context menu item, or simply double-click the procedure;
- edit procedure definition using the appropriate tabs of Procedure Editor.

To change the name of a procedure:

- select the procedure to rename in the DB Explorer tree;
- right-click the procedure alias and select the **Rename Procedure <procedure_name>...** item from the context menu;
- edit the procedure name using the **Rename Object...** dialog.

Executing Procedures

To execute a procedure:

- select the procedure to execute in the DB Explorer tree (type the first letters of the procedure name for quick search);
- right-click the object and select the **Edit Procedure <procedure_name>...** context menu item, or simply double-click the procedure;
- execute the procedure using the **Execute Procedure** Navigation bar item of Procedure Editor.

Dropping Procedures

To drop a procedure:

- select the procedure to drop in the DB Explorer tree;
- right-click the object and select the **Drop Procedure <procedure_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new procedure;
Ctrl+O to edit the selected procedure;
Ctrl+R to rename the procedure;

© 2008 EMS Database Management Solutions

Shift+Del to drop the object from the database.

5.4.3.1 Procedure Editor

Procedure Editor allows you to create new procedure, execute the existing procedure or edit its definition. It opens automatically when you create a new procedure and is available on editing an existing one (see Create procedure and Edit procedure for details).

To open a stored procedure in **Procedure Editor**, double-click it in the DB Explorer tree.

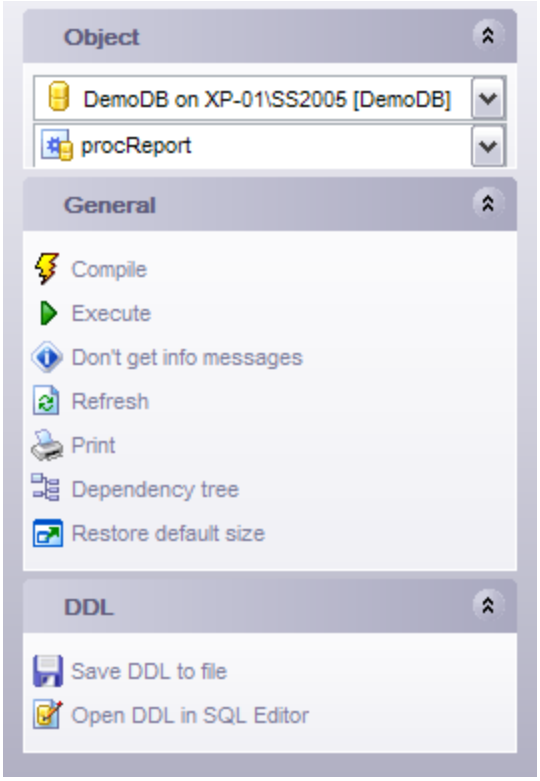
- Using Navigation bar
- Editing procedure definition
- Browsing object dependencies
- Editing procedure description
- Specifying input parameters
- Executing procedure
- Viewing DDL definition

5.4.3.1.1 Using Navigation bar

The **Navigation bar** of **Procedure Editor** allows you to:

- select the database where the object is located;
- select the procedure for editing;
- compile the object (if it is being created/modified);
- save the object description (if it has been modified);
- execute the procedure;
- enable/disable information messages;
- refresh the content of the active tab;
- print metadata of the object;
- view the dependency tree for the procedure;
- restore the default size of the editor window.

© 2008 EMS Database Management Solutions



Depending on the current tab selection, the **Navigation bar** expands to one or more additional panes with tab-specific actions that can be useful for working with the object.

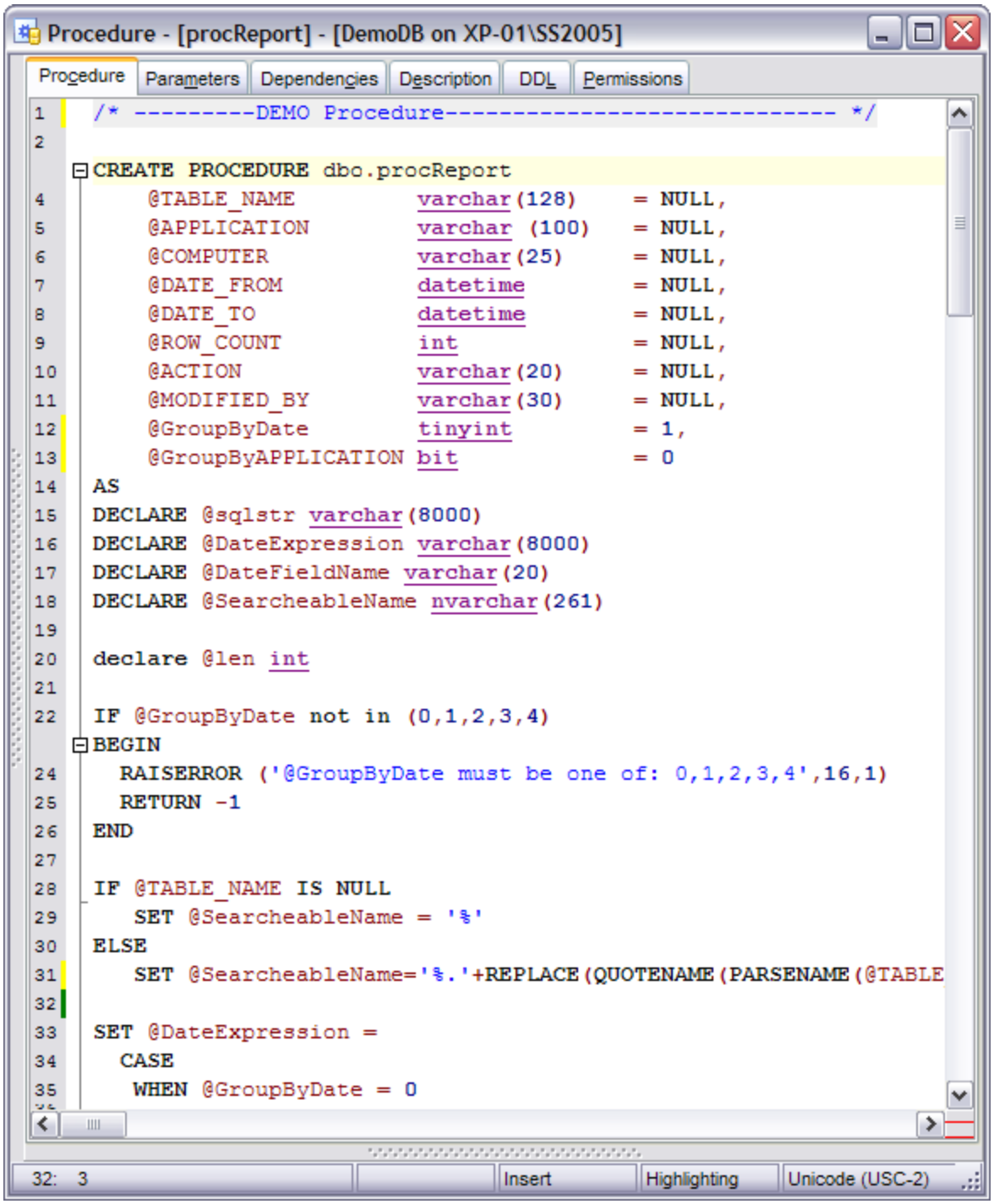
<< Back to Procedure Editor

5.4.3.1.2 Editing procedure definition

Use the **Procedure** tab of **Procedure Editor** to create/edit a stored procedure and specify its definition.

For your convenience the **syntax highlight**, **code completion** and a number of other features for efficient SQL editing are implemented. For details see Working with SQL Editor area and Using the context menu.

© 2008 EMS Database Management Solutions



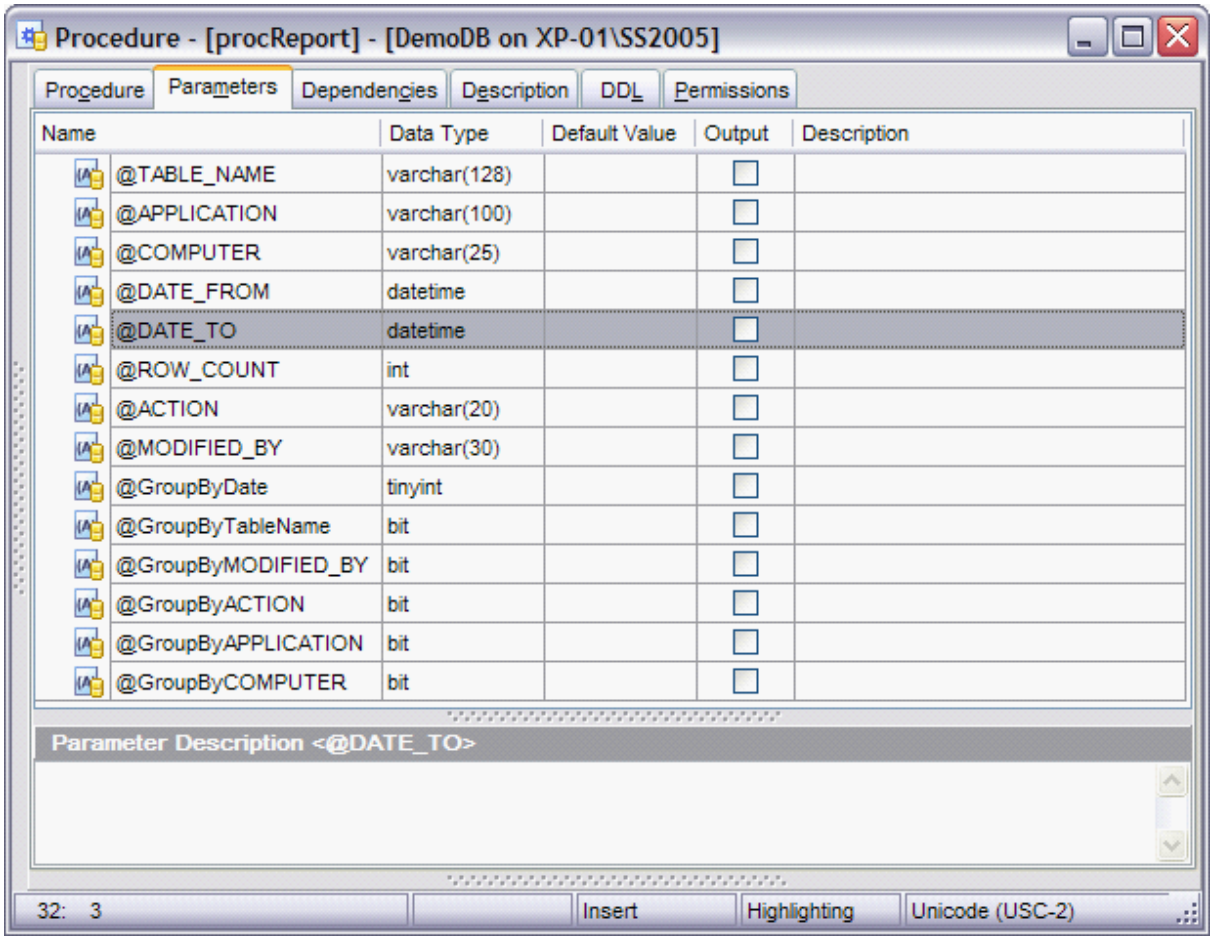
To execute a procedure, you can use the **Execute** item available within the Navigation bar.

© 2008 EMS Database Management Solutions

5.4.3.1.3 Browsing procedure parameters

The **Parameters** tab is provided for browsing the list of parameters of the stored procedure.

Right-click the list to display the popup menu allowing you to export this list to any of supported formats.



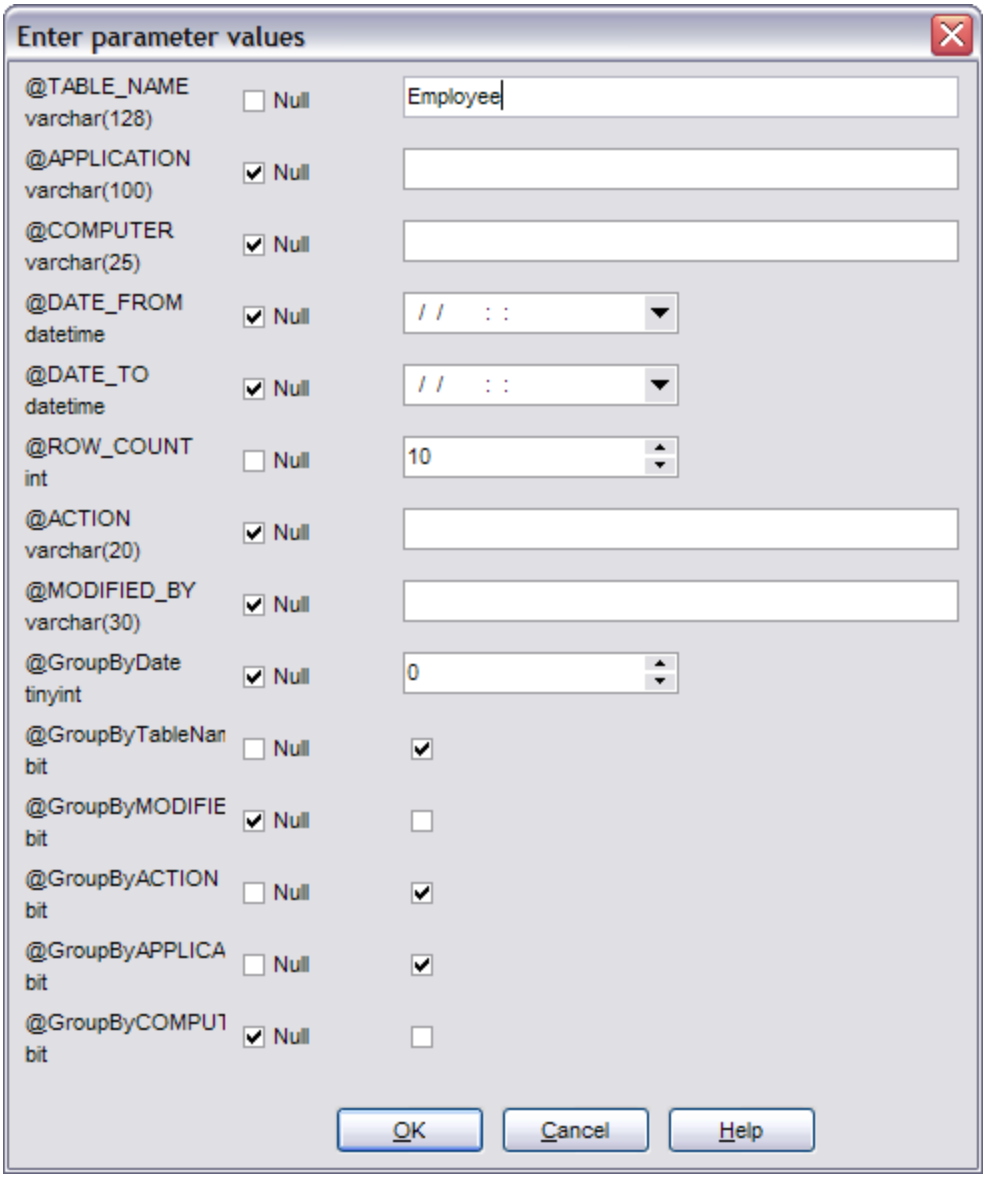
The parameter list provides the following attributes of each parameter used in the procedure:
Name (parameter name specified by using an at (@) sign as the first character)
Data Type (the parameter data type)
Default Value (a default value for the parameter)
Output (indicates that the parameter is an output parameter)

If necessary, you can also use the **Parameter Description** area to supply a *description* for each parameter.

© 2008 EMS Database Management Solutions

5.4.3.1.4 Specifying input parameters

If the stored procedure (or user-defined function) has parameters, the **Input Parameter dialog** appears before the procedure execution. It allows you to specify the values for all procedure parameters. After changes are done, click the **OK** button to execute the stored procedure, or the **Cancel** button to abort the execution.



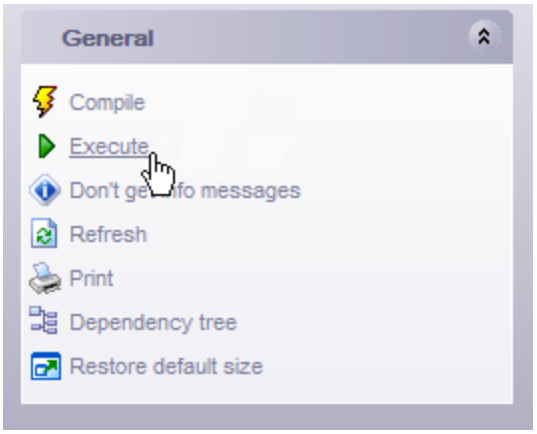
See also:

Executing procedure/function

© 2008 EMS Database Management Solutions

5.4.3.1.5 Executing procedure/function

Procedure Editor / **UDF Editor** provide an ability to execute the current procedure or user-defined function. Click the **Execute** item of the **Navigation bar** to execute the procedure/UDF. If the procedure/UDF has parameters, SQL Manager 2008 for SQL Server allows you to specify the values for these parameters in the Input Parameters dialog which appears just before execution. The result of the successfully executed procedure/UDF, as well as the error message in case of execution failure, appears in the message panel at the bottom of the **Procedure Editor** / **UDF Editor**.



See also:
Procedure Editor
UDF Editor

5.4.4 UDTs

A **User-Defined Data Type** is internally represented as an existing type, but is considered to be a separate data type. In fact, this is an alias data type in an instance of SQL Server. An alias data type's implementation is based on a SQL Server native system type. A user-defined type is implemented through a class of an assembly in the Microsoft .NET Framework common language runtime (CLR).

Creating User-Defined Types

To create a new user-defined type:
- select the **Database | New Object...** main menu item;
- select **User-Defined Data Type** in the Create New Object dialog;
- define UDT properties using the appropriate tabs of UDT Editor.

Hint: To create a new user-defined type, you can also right-click the **UDTs** node or any object within this node in the DB Explorer tree and select the **New User-Defined Type** item from the context menu.

To create a new UDT with the same properties as one of existing UDTs has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

© 2008 EMS Database Management Solutions

Alternatively, you can right-click a UDT in the DB Explorer tree and select the **Duplicate User-Defined Type <UDT_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new UDT in, and to edit the result SQL statement for creating the UDT.

Editing User-Defined Types

To edit an existing user-defined type:
- select the user-defined type for editing in the DB Explorer tree (type the first letters of the UDT name for quick search);
- right-click the object and select the **Edit User-Defined Type <UDT_name>** context menu item, or simply double-click the UDT;
- edit UDT definition using the appropriate tabs of UDT Editor.

To change the name of a user-defined type:
- select the user-defined type to rename in the DB Explorer tree;
- right-click the UDT alias and select the **Rename User-Defined Type <UDT_name>...** item from the context menu;
- edit the UDT name using the **Rename Object...** dialog.

Dropping User-Defined Types

To drop a user-defined type:
- select the user-defined type to drop in the DB Explorer tree;
- right-click the object and select the **Drop User-Defined Type <UDT_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new UDT;
Ctrl+O to edit the selected UDT;
Ctrl+R to rename the UDT;
Shift+Del to drop the object from the database.

5.4.4.1 UDT Editor

UDT Editor allows you to define user-defined type properties. It opens automatically when you create a new type and is available on editing an existing one (see Create UDT and Edit UDT for details).

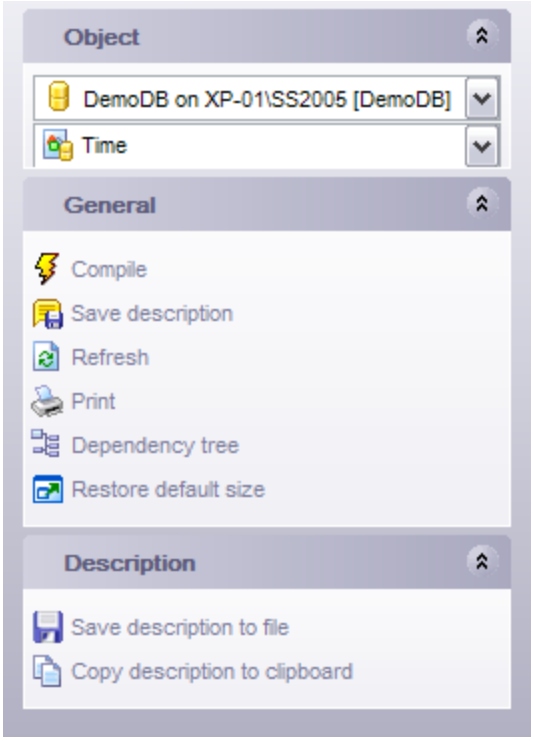
To open a user-defined type in **UDT Editor**, double-click it in the DB Explorer tree.
- Using Navigation bar
- Editing user-defined type definition
- Browsing object dependencies
- Editing user-defined type description
- Viewing DDL definition

© 2008 EMS Database Management Solutions

5.4.4.1.1 Using Navigation bar

The **Navigation bar** of **UDT Editor** allows you to:

- select the database where the object is located;
- select the user-defined type for editing;
- compile the object (if it is being created/modified);
- save the object description (if it has been modified);
- refresh the content of the active tab;
- print metadata of the object;
- view the dependency tree for the user-defined type;
- restore the default size of the editor window.



Depending on the current tab selection, the **Navigation bar** expands to one or more additional panes with tab-specific actions that can be useful for working with the object.

<< Back to UDT Editor

5.4.4.1.2 Editing UDT definition

Use the **Type** tab of **UDT Editor** to create/edit a user-defined data type and specify its properties.

Alias user-defined data type
This selection specifies that an alias data type will be created on the basis of an existing SQL Server data type.

CLR user-defined data type

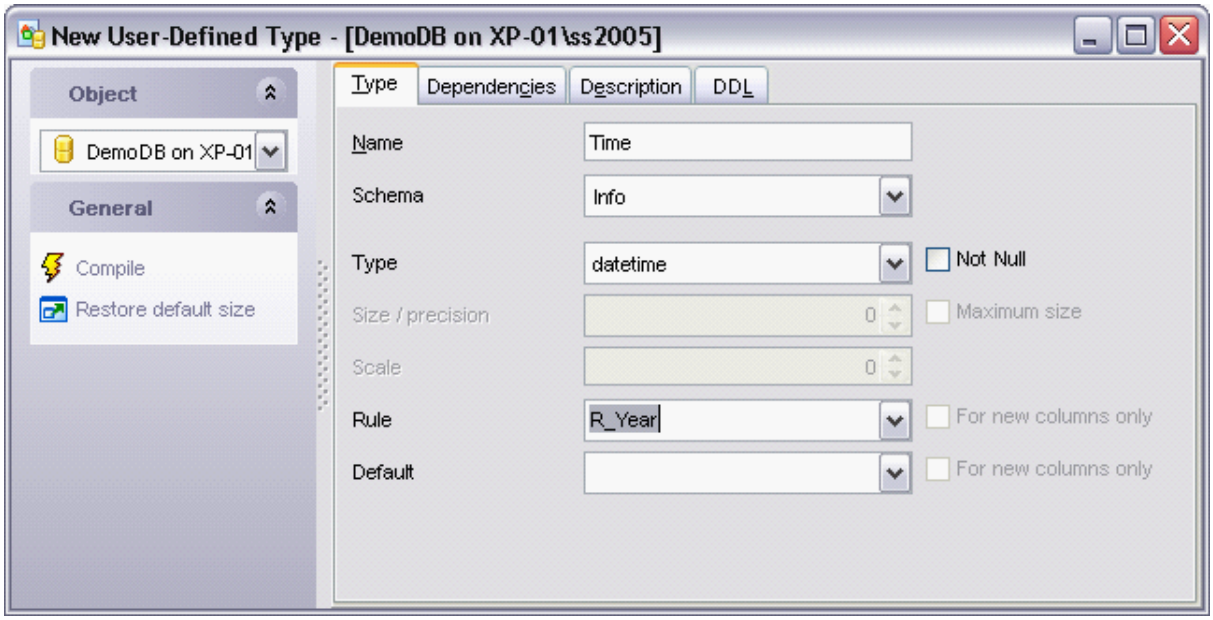
© 2008 EMS Database Management Solutions

This selection specifies that the user-defined will be taken from an assembly.

Table type
Specify to create a new user-defined table type.

Name
Enter a name for the new user-defined type, or modify the name of the user-defined type being edited.

Schema
Use the drop-down list to select the schema to which the user-defined type belongs. The default selection is the default schema for the current user. For Microsoft® SQL Server 2000 or SQL Server 7.0, specify the owner of the UDT (the default value is the current user).



Alias Type Options

Type
Use the drop-down list to select SQL Server supplied data type on which the alias data type is based (*base_type*). The base type of an existing UDT is not editable.

☑ **Not Null**
Specify whether the UDT can accept NULL values. The nullability of an existing UDT is not editable.

Size
Specify the size value (for certain types). Check the ☑ **Maximum size** option to use the maximum values set by SQL Server.

Scale
Defines the maximum number of decimal digits that can be stored to the right of the

© 2008 EMS Database Management Solutions

decimal point of the value (for numeric and decimal data types).

Rule
Use the drop-down list to select a rule to bind to the user-defined data type (optional).

Default
Use the drop-down list to select a default to bind to the user-defined data type (optional).

You can apply the **Rule** or **Default** options **For new columns only**.

CLR type options

Assembly
Use the drop-down list to select the SQL Server assembly that references the implementation of the user-defined type in the common language runtime. The assembly must already have been created in the database.

Class name
Specifies the class within the assembly that implements the user-defined type. The class name can be a namespace-qualified name if the programming language that is used to write the class uses the concept of namespaces, such as C#.

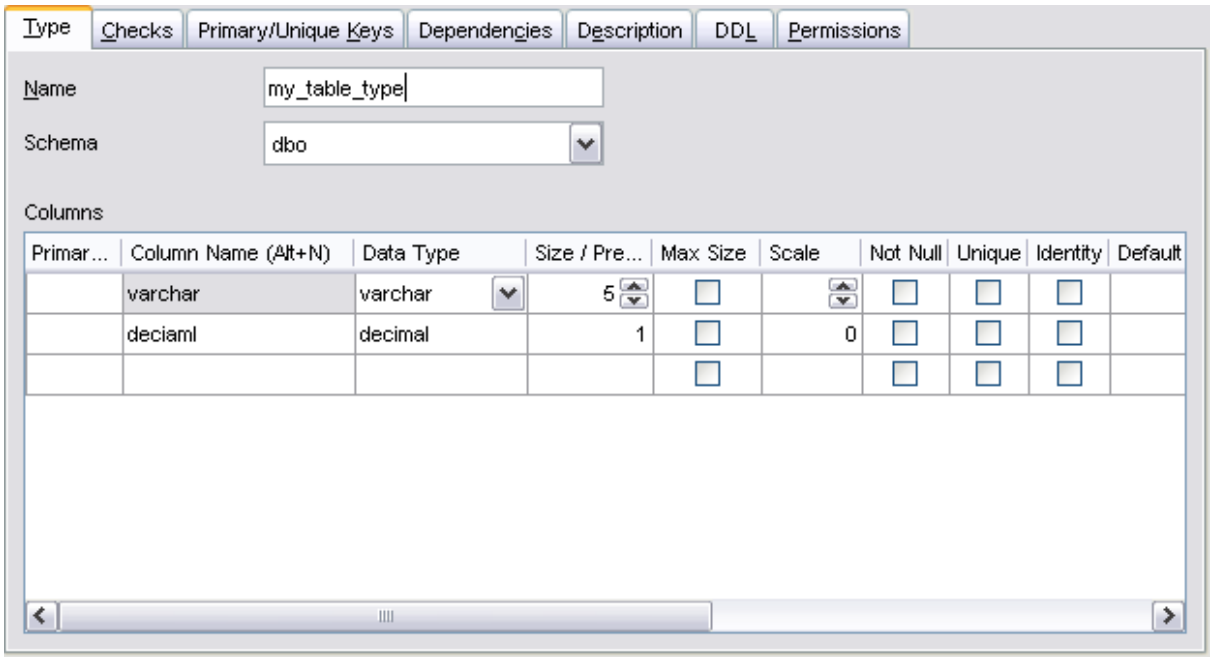
Table type options

Column Name
Defines the columns for a user-defined table type.

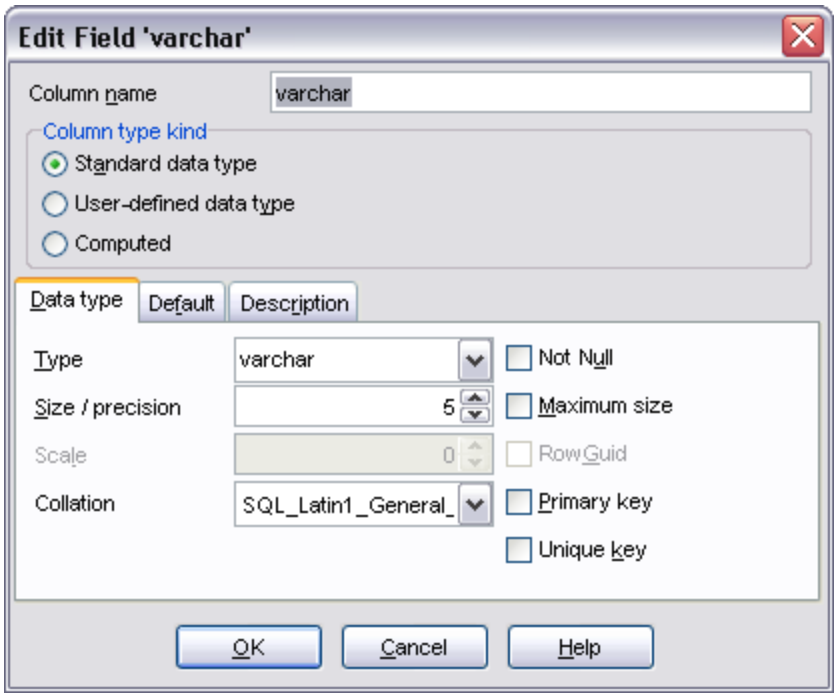
Data Type
Defines the data type in a column for a user-defined table type.

Checks, Primary/Unique keys
The tabs allow to define table constraints on a user-defined table type (primary key, unique, and checks respectively).

© 2008 EMS Database Management Solutions



With the help of context menu you can editing, adding and deleting columns. The built-in **Field Editor allows** to define all column parameters for current table type.



5.4.5 UDFs

A **Function** is a mapping embodied as a program (the function *body*) that can be invoked by using zero or more input values (*arguments*) to a single value (the *result*). A **User-Defined Function** is stored as a database object providing reusable code.

© 2008 EMS Database Management Solutions

The Transact-SQL programming language provides the following types of functions:
Rowset functions (can be used like table references in an SQL statement)
Aggregate functions (operate on a collection of values but return a single, summarizing value)
Ranking functions (return a ranking value for each row in a partition)
Scalar functions (operate on a single value and then return a single value; scalar functions can be used wherever an expression is valid)

Creating User-Defined Functions

To create a new function:
- select the **Database | New Object...** main menu item;
- select **User-Defined Function** in the Create New Object dialog;
- define function properties using the appropriate tabs of UDF Editor.

Hint: To create a new function, you can also right-click the **UDFs** node or any object within this node in the DB Explorer tree and select the **New User-Defined Function** item from the context menu.

To create a new function with the same properties as one of existing functions has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a function in the DB Explorer tree and select the **Duplicate User-Defined Function <UDF_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new function in, and to edit the result SQL statement for creating the function.

Editing User-Defined Functions

To edit an existing function:
- select the function for editing in the DB Explorer tree (type the first letters of the function name for quick search);
- right-click the object and select the **Edit User-Defined Function <UDF_name>** context menu item, or simply double-click the function;
- edit function definition using the appropriate tabs of UDF Editor.

To change the name of a UDF:
- select the UDF to rename in the DB Explorer tree;
- right-click the UDF alias and select the **Rename User-Defined Function <UDF_name>...** item from the context menu;
- edit the UDF name using the **Rename Object...** dialog.

Dropping User-Defined Functions

To drop a function:
- select the function to drop in the DB Explorer tree;
- right-click the object and select the **Drop User-Defined Function <UDF_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new UDF;

© 2008 EMS Database Management Solutions

Ctrl+O to edit the selected UDF;
Ctrl+R to rename the UDF;
Shift+Del to drop the object from the database.

5.4.5.1 UDF Editor

UDF Editor allows you to define user-defined function properties. It opens automatically when you create a new user-defined function and is available on editing an existing one (see Create UDF and Edit UDF for details).

To open a user-defined function in **UDF Editor**, double-click it in the DB Explorer tree.

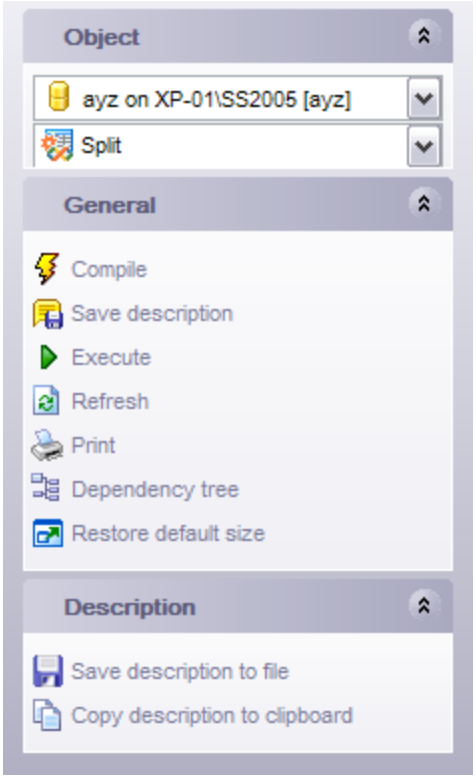
- Using Navigation bar
- Editing user-defined function definition
- Browsing user-defined function parameters
- Browsing object dependencies
- Editing user-defined function description
- Specifying input parameters
- Executing function
- Viewing DDL definition

5.4.5.1.1 Using Navigation bar

The **Navigation bar** of **UDF Editor** allows you to:

- select the database where the object is located;
- select the user-defined function for editing;
- compile the object (if it is being created/modified);
- save the object description (if it has been modified);
- execute the user-defined function;
- refresh the content of the active tab;
- print metadata of the object;
- view the dependency tree for the user-defined function;
- restore the default size of the editor window.

© 2008 EMS Database Management Solutions



Depending on the current tab selection, the **Navigation bar** expands to one or more additional panes with tab-specific actions that can be useful for working with the object.

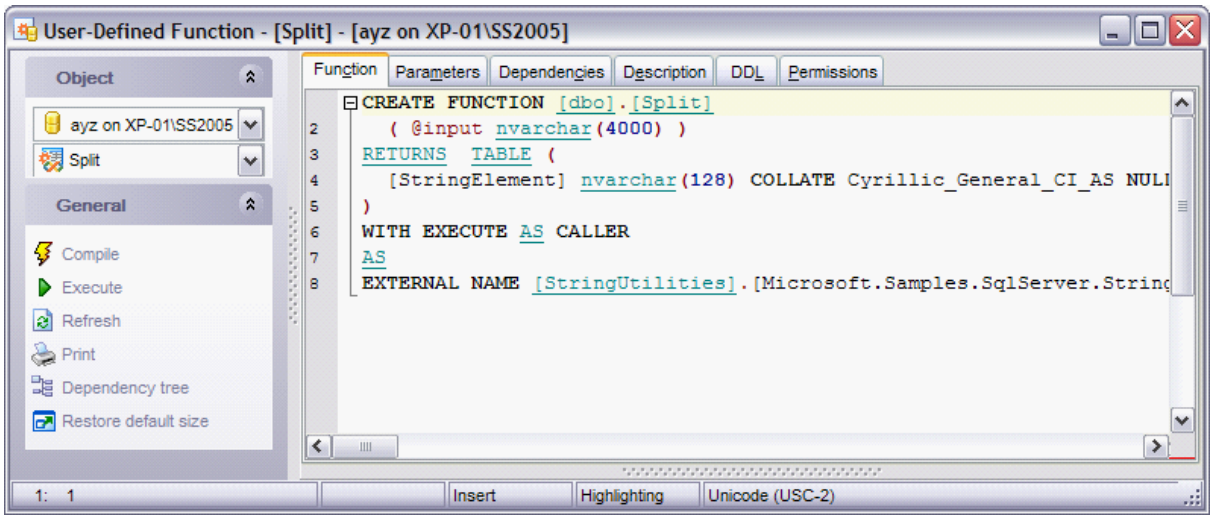
[<< Back to UDF Editor]

5.4.5.1.2 Editing UDF definition

Use the **Function** tab of **UDF Editor** to create/edit a user-defined function and specify its definition.

For your convenience the **syntax highlight**, **code completion** and a number of other features for efficient SQL editing are implemented. For details see Working with SQL Editor area and Using the context menu.

© 2008 EMS Database Management Solutions



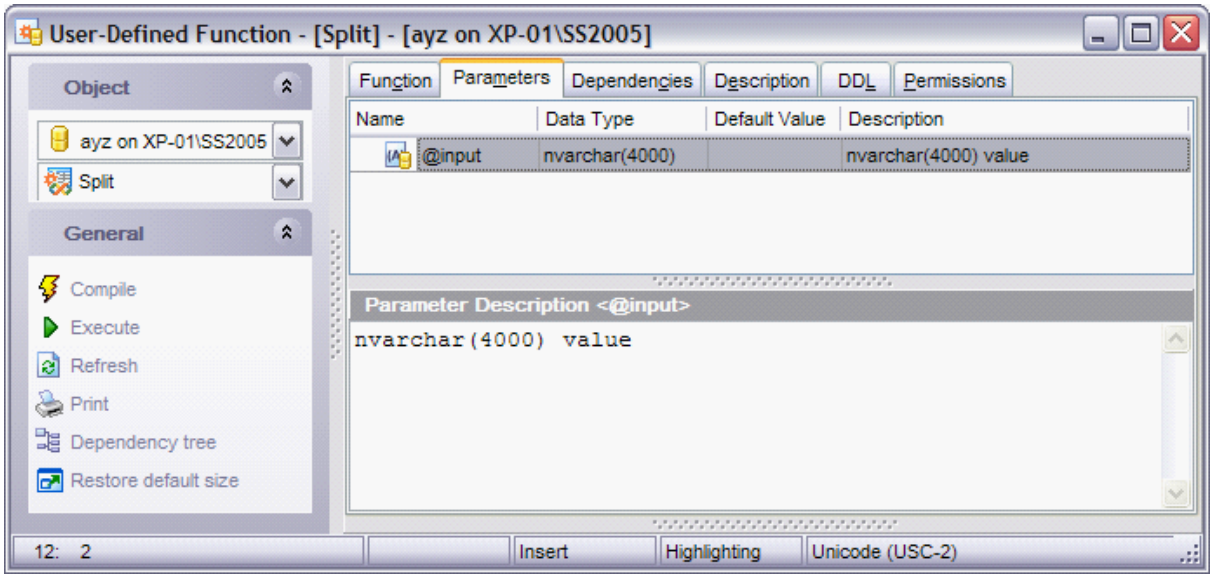
Note: An UDF is a Transact-SQL or common language runtime (CLR) routine that returns a value. User-defined functions, like system functions, can be invoked from a query. Scalar functions can be executed by using an *EXECUTE* statement like stored procedures.

To execute a UDF, you can use the **Execute** item available within the Navigation bar.

5.4.5.1.3 Browsing UDF parameters

The **Parameters** tab is provided for browsing the list of parameters of the user-defined function.

Right-click the list to display the popup menu allowing you to export this list to any of supported formats.



The parameter list provides the following attributes of each parameter used in the UDF:
Name (parameter name specified by using an at (@) sign as the first character)
Data Type (the parameter data type)
Default Value (a default value for the parameter)

If necessary, you can also use the **Parameter Description** area to supply a *description* for each parameter.

5.4.6 Synonyms

You can use **synonyms** in place of their referenced base object in a number of SQL statements and expression contexts. When working with synonyms in these contexts, the base object is affected. For example, if you have a synonym that references a base object that is a table, when you insert a row into the synonym, you are actually inserting a row into the referenced table.

Creating Synonyms

To create a new synonym:
- select the **Database | New Object...** main menu item;
- select **Synonym** in the Create New Object dialog;
- define synonym properties using the appropriate tabs of Synonym Editor.

Hint: To create a new synonym, you can also right-click the **Synonyms** node or any object within this node in the DB Explorer tree and select the **New Synonym** item from the context menu.

To create a new synonym with the same properties as one of existing synonyms has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a synonym in the DB Explorer tree and select the **Duplicate Synonym <synonym_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new synonym in, and to edit the result SQL statement for creating the synonym.

Editing Synonyms

To edit an existing synonym:
- select the synonym for editing in the DB Explorer tree (type the first letters of the synonym name for quick search);
- right-click the object and select the **Edit Synonym <synonym_name>** context menu item, or simply double-click the synonym;
- edit synonym properties using the appropriate tabs of Synonym Editor.

To change the name of a synonym:
- select the synonym to rename in the DB Explorer tree;
- right-click the synonym alias and select the **Rename Synonym <synonym_name>...** item from the context menu;
- edit the synonym name using the **Rename Object...** dialog.

© 2008 EMS Database Management Solutions

Dropping Synonyms

To drop a synonym:
- select the synonym to drop in the DB Explorer tree;
- right-click the object and select the **Drop Synonym <synonym_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new synonym;
Ctrl+O to edit the selected synonym;
Ctrl+R to rename the synonym;
Shift+Del to drop the object from the database.

5.4.6.1 Synonym Editor

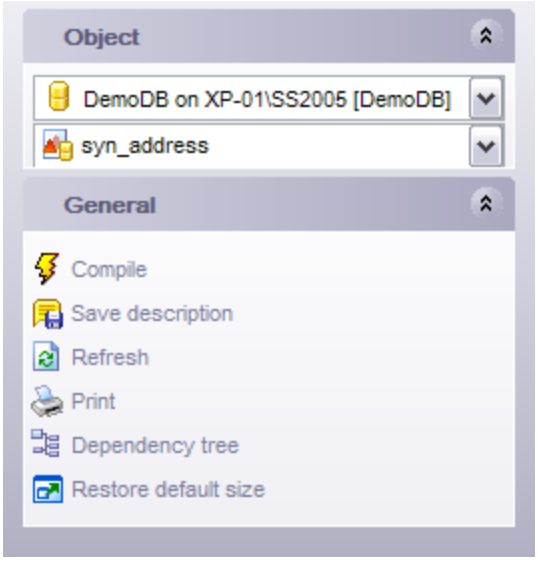
Synonym Editor allows you to define synonym properties. It opens automatically when you create a new synonym and is available on editing an existing one (see Create synonym and Edit synonym for details).

To open a synonym in **Synonym Editor**, double-click it in the DB Explorer tree.
- Using Navigation bar
- Editing synonym definition
- Browsing object dependencies
- Editing synonym description
- Viewing DDL definition

5.4.6.1.1 Using Navigation bar

The **Navigation bar** of **Synonym Editor** allows you to:
- select the database where the object is located;
- select the synonym for editing;
- compile the object (if it is being created/modified);
- save the object description (if it has been modified);
- refresh the content of the active tab;
- print metadata of the object;
- view the dependency tree for the synonym;
- restore the default size of the editor window.

© 2008 EMS Database Management Solutions



Depending on the current tab selection, the **Navigation bar** expands to one or more additional panes with tab-specific actions that can be useful for working with the object.

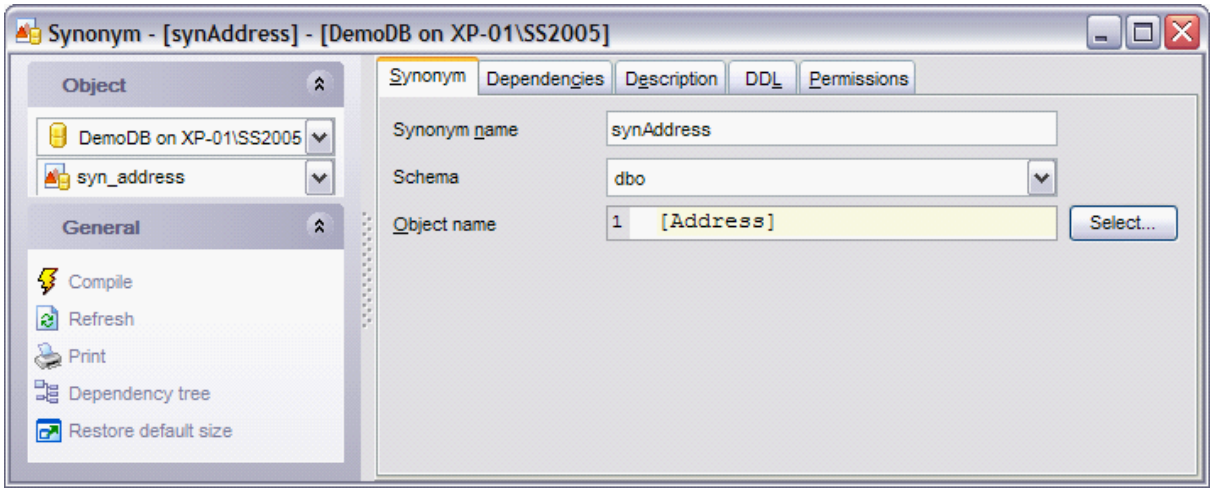
[<< Back to Synonym Editor]

5.4.6.1.2 Editing synonym definition

Use the **Synonym** tab of **Synonym Editor** to create/edit a synonym and specify its properties.

Name
Enter a name for the new synonym, or modify the name of the synonym being edited.

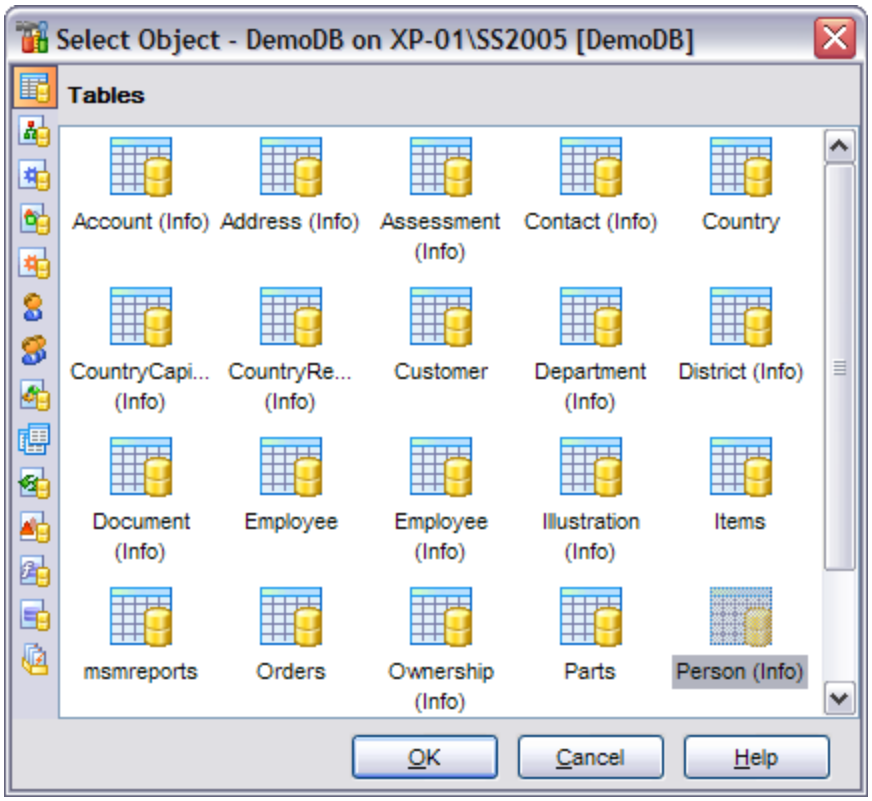
Schema
Use the drop-down list to select the schema in which the synonym is created.



© 2008 EMS Database Management Solutions

Object name
This box displays the base object that the synonym references. To change the object, click the **Select...** button.

The **Select Object** dialog allows you to specify a database object to use as the referenced object.



Hint: The context menu of the **Select Object** dialog allows you to toggle the *Icons*/*List* object representation mode.

5.4.7 Aggregates

Aggregate functions, or **aggregates**, perform a calculation on a set of values and return a single value. Traditionally Microsoft® SQL Server supported only built-in aggregate functions, such as *SUM* or *MAX*, that operate on a set of input scalar values and generate a single aggregate value from that set. SQL Server integration with the .NET Framework common language runtime (CLR) now allows developers to create custom aggregate functions in managed code and make these functions accessible to Transact-SQL or other managed code.

Creating Aggregates

To create a new aggregate:

© 2008 EMS Database Management Solutions

- select the **Database | New Object...** main menu item;
- select **Aggregate** in the Create New Object dialog;
- define aggregate properties using the appropriate tabs of Aggregate Editor.

Hint: To create a new aggregate, you can also right-click the **Aggregates** node or any object within this node in the DB Explorer tree and select the **New Aggregate** item from the context menu.

To create a new aggregate with the same properties as one of existing aggregates has:

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

Alternatively, you can right-click a aggregate in the DB Explorer tree and select the **Duplicate Aggregate <aggregate_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new aggregate in, and to edit the result SQL statement for creating the aggregate.

Editing Aggregates

To edit an existing aggregate:

- select the aggregate for editing in the DB Explorer tree (type the first letters of the aggregate name for quick search);
- right-click the object and select the **Edit Aggregate <aggregate_name>** context menu item, or simply double-click the aggregate;
- edit aggregate properties using the appropriate tabs of Aggregate Editor.

To change the name of an aggregate:

- select the aggregate to rename in the DB Explorer tree;
- right-click the aggregate alias and select the **Rename Aggregate <aggregate_name>...** item from the context menu;
- edit the aggregate name using the **Rename Object...** dialog.

Dropping Aggregates

To drop an aggregate:

- select the aggregate to drop in the DB Explorer tree;
- right-click the object and select the **Drop Aggregate <aggregate_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new aggregate;
Ctrl+O to edit the selected aggregate;
Ctrl+R to rename the aggregate;
Shift+Del to drop the object from the database.

5.4.7.1 Aggregate Editor

Aggregate Editor allows you to define aggregate properties. It opens automatically when you create a new aggregate and is available on editing an existing one (see Create aggregate and Edit aggregate for details).

To open an aggregate in **Aggregate Editor**, double-click it in the DB Explorer tree.

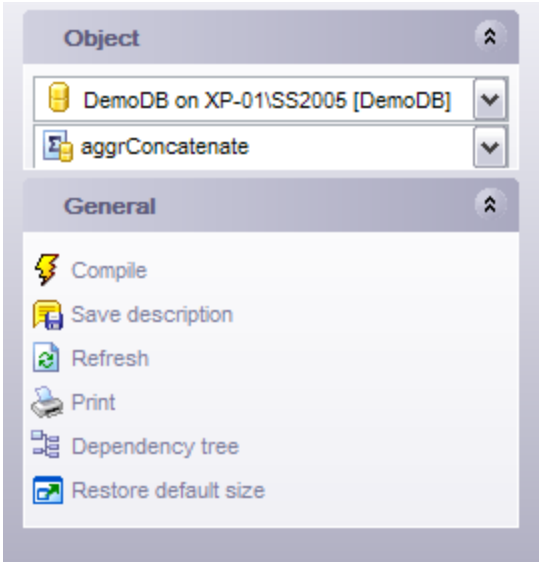
© 2008 EMS Database Management Solutions

- Using Navigation bar
- Editing aggregate definition
- Browsing object dependencies
- Editing aggregate description
- Viewing DDL definition

5.4.7.1.1 Using Navigation bar

The **Navigation bar** of **Aggregate Editor** allows you to:

- select the database where the object is located;
- select the aggregate for editing;
- compile the object (if it is being created/modified);
- save the object description (if it has been modified);
- refresh the content of the active tab;
- print metadata of the object;
- view the dependency tree for the aggregate;
- restore the default size of the editor window.



Depending on the current tab selection, the **Navigation bar** expands to one or more additional panes with tab-specific actions that can be useful for working with the object.

[<< Back to Aggregate Editor]

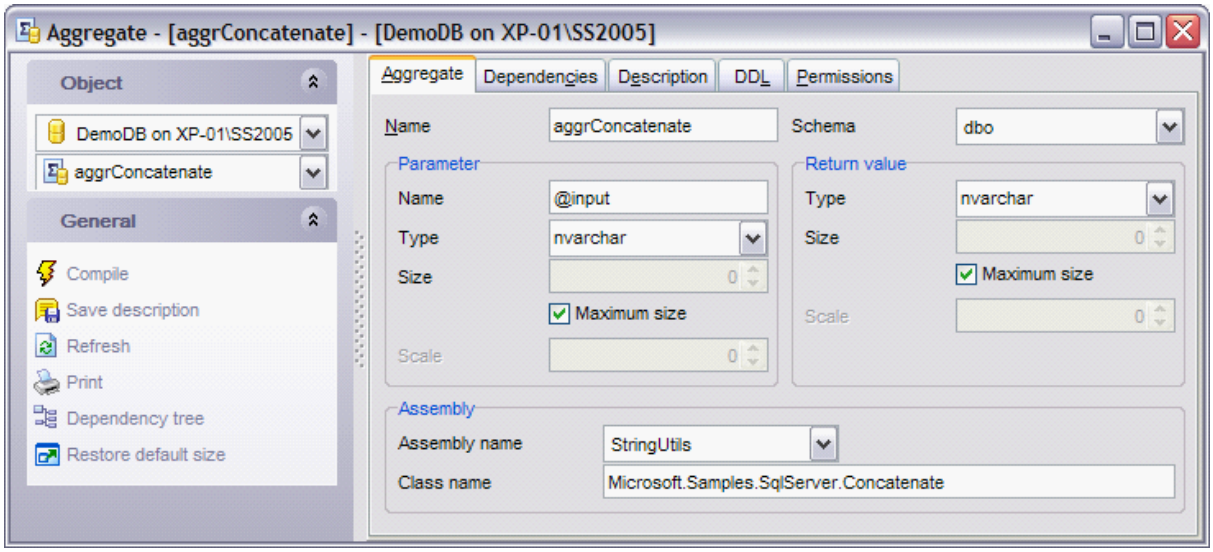
5.4.7.1.2 Editing aggregate definition

Use the **Aggregate** tab of **Aggregate Editor** to create/edit an aggregate function and specify its definition.

Name
Enter a name for the new aggregate, or modify the name of the aggregate being edited.

© 2008 EMS Database Management Solutions

Schema
Use the drop-down list to select the schema to which the user-defined aggregate function belongs.



Parameter

Name
Specify the parameter name for the user-defined aggregate. The value of the parameter is supplied by the user when the aggregate function is executed. Specify a parameter name using an at (@) sign as the first character.

Type
Use the drop-down list to select one of the SQL Server system scalar data types to hold the value of the input parameter or return value.

Size
Specify the size value (for certain types). Check the ☑ **Maximum size** option to use the maximum values set by SQL Server.

Scale
Defines the maximum number of decimal digits that can be stored to the right of the decimal point of the value (for float data types).

Return value

Type
Use the drop-down list to select the name of an existing data type to be used as the return value type.

Size
Specify the size value (for certain types). Check the ☑ **Maximum size** option to use the maximum values set by SQL Server.

© 2008 EMS Database Management Solutions

Assembly

Assembly name
Use the drop-down list to select the SQL Server assembly to bind with the user-defined aggregate function. The assembly must already have been created in the database.

Class name
Specify the name of the class in the assembly that implements the user-defined aggregate. This value must be a valid SQL Server identifier and match the name of a class that exists in the assembly. This may be a namespace-qualified name if the programming language used to write the class uses namespaces, such as C#.

5.4.8 XML Schema Collections

Microsoft® SQL Server™ provides native storage of XML data through the XML data type. 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 they are stored in the database

The XML stored in a column or variable that a schema is associated with is referred to as typed XML, because the schema provides the necessary data type information for the instance data. SQL Server uses this type information to optimize data storage.

Note that XML Schema Collection is a metadata entity like a table in the database, hence the XML Schema Collection can be created, modified and dropped.

Creating XML Schema Collections

To create a new XML Schema Collection:

- select the **Database | New Object...** main menu item;
- select **XML Schema Collection** in the Create New Object dialog;
- define XML Schema Collection properties using the appropriate tabs of XML Schema Collection Editor.

Hint: To create a new XML Schema Collection, you can also right-click the **XML Schema Collections** node or any object within this node in the DB Explorer tree and select the **New XML Schema Collection** item from the context menu.

To create a new XML Schema Collection with the same properties as one of existing XML Schema Collections has:

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

Alternatively, you can right-click a XML Schema Collection in the DB Explorer tree and select the **Duplicate XML Schema Collection <XML_Schema_Collection_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new XML Schema Collection in, and to edit the result SQL statement for creating the XML Schema Collection.

© 2008 EMS Database Management Solutions

Editing XML Schema Collections

To edit an existing XML Schema Collection:
- select the XML Schema Collection for editing in the DB Explorer tree (type the first letters of the XML Schema Collection name for quick search);
- right-click the object and select the **Edit XML Schema Collection <XML_Schema_Collection_name>** context menu item, or simply double-click the XML Schema Collection;
- edit XML Schema Collection properties using the appropriate tabs of XML Schema Collection Editor.

To change the name of a XML Schema Collection:
- select the XML Schema Collection to rename in the DB Explorer tree;
- right-click the XML Schema Collection alias and select the **Rename XML Schema Collection <XML_Schema_Collection_name>...** item from the context menu;
- edit the XML Schema Collection name using the **Rename Object...** dialog.

Dropping XML Schema Collections

To drop a XML Schema Collection:
- select the XML Schema Collection to drop in the DB Explorer tree;
- right-click the object and select the **Drop XML Schema Collection <XML_Schema_Collection_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new XML Schema Collection;
Ctrl+O to edit the selected XML Schema Collection;
Ctrl+R to rename the XML Schema Collection;
Shift+Del to drop the object from the database.

5.4.8.1 XML Schema Collection Editor

XML Schema Collection Editor allows you to define XML schema collection properties. It opens automatically when you create a new XML schema collection and is available on editing an existing one (see Create XML Schema Collection and Edit XML Schema Collection for details).

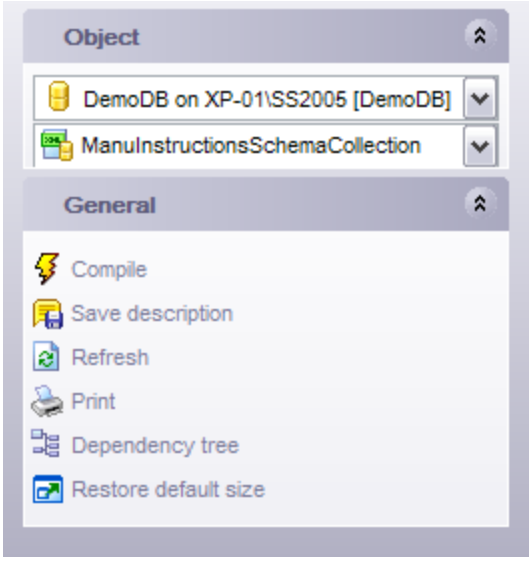
To open a XML schema collection in **XML Schema Collection Editor**, double-click it in the DB Explorer tree.
- Using Navigation bar
- Editing XML Schema Collection
- Browsing object dependencies
- Editing object description
- Viewing DDL definition

© 2008 EMS Database Management Solutions

5.4.8.1.1 Using Navigation bar

The **Navigation bar** of **XML Schema Collection Editor** allows you to:

- select the database where the object is located;
- select the XML schema collection for editing;
- compile the object (if it is being created/modified);
- save the object description (if it has been modified);
- refresh the content of the active tab;
- print metadata of the object;
- view the dependency tree for the XML schema collection;
- restore the default size of the editor window.



Depending on the current tab selection, the **Navigation bar** expands to one or more additional panes with tab-specific actions that can be useful for working with the object.

<< Back to XML Schema Collection Editor

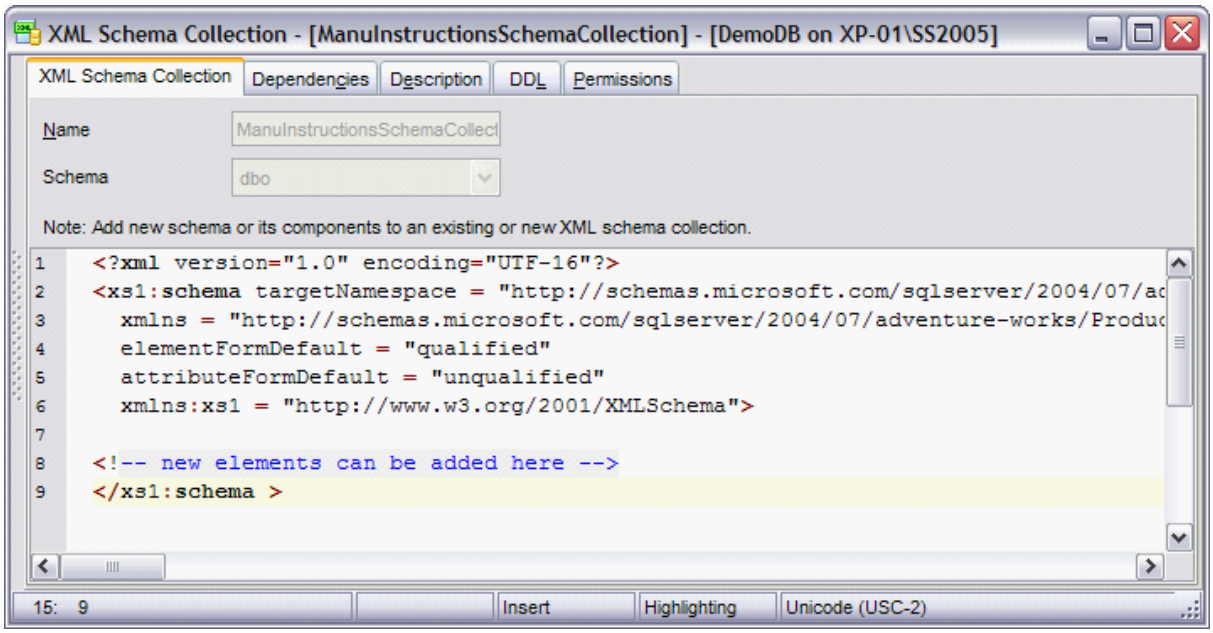
5.4.8.1.2 Editing XML Schema Collection

Use the **XML Schema Collection** tab of **XML Schema Collection Editor** to create/edit a XML Schema Collection and specify its definition.

Name
Enter a name for the new XML Schema Collection, or view the name of the XML Schema Collection being edited.

Schema
Use the drop-down list to select the schema to which the XML Schema Collection belongs.

© 2008 EMS Database Management Solutions



The lower area of the editor window allows you to specify the XML Schema Collection definition. You can add a new schema or its component(s) to **XML Schema Collection**. After successful compilation, the **DDL** tab will display the changes made to the schema. To compile the schema, use the corresponding item of the Navigation bar.

5.4.9 Defaults

When bound to a column or a user-defined data type, a **default** specifies a value to be inserted into the column to which the object is bound (or into all columns, in case of UDF) when no value is explicitly supplied during an insert operation.

Creating Defaults

To create a new default:
- select the **Database | New Object...** main menu item;
- select **Default** in the Create New Object dialog;
- define default properties using the appropriate tabs of Default Editor.

Hint: To create a new default, you can also right-click the **Defaults** node or any object within this node in the DB Explorer tree and select the **New Default** item from the context menu.

To create a new default with the same properties as one of existing defaults has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a default in the DB Explorer tree and select the **Duplicate Default <default_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new default in, and to edit the result SQL statement for creating the default.

© 2008 EMS Database Management Solutions

Editing Defaults

To edit an existing default:
- select the default for editing in the DB Explorer tree (type the first letters of the default name for quick search);
- right-click the object and select the **Edit Default <default_name>** context menu item, or simply double-click the default;
- edit default properties using the appropriate tabs of Default Editor.

To change the name of a default:
- select the default to rename in the DB Explorer tree;
- right-click the default alias and select the **Rename Default <default_name>...** item from the context menu;
- edit the default name using the **Rename Object...** dialog.

Dropping Defaults

To drop a default:
- select the default to drop in the DB Explorer tree;
- right-click the object and select the **Drop Default <default_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new default;
Ctrl+O to edit the selected default;
Ctrl+R to rename the default;
Shift+Del to drop the object from the database.

5.4.9.1 Default Editor

Default Editor allows you to define default properties. It opens automatically when you create a new default and is available on editing an existing one (see Create default and Edit default for details).

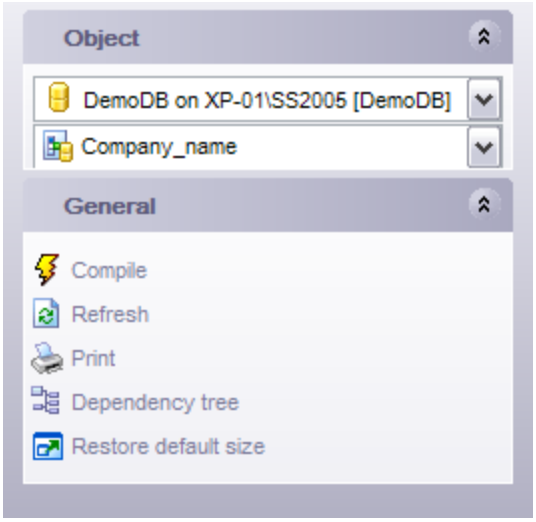
To open a default in **Default Editor**, double-click it in the DB Explorer tree.
- Using Navigation bar
- Editing default definition
- Browsing object dependencies
- Editing default description
- Viewing DDL definition

5.4.9.1.1 Using Navigation bar

The **Navigation bar** of **Default Editor** allows you to:
- select the database where the object is located;
- select the default for editing;
- compile the object (if it is being created/modified);

© 2008 EMS Database Management Solutions

- save the object description (if it has been modified);
- refresh the content of the active tab;
- print metadata of the object;
- view the dependency tree for the default;
- restore the default size of the editor window.



Depending on the current tab selection, the **Navigation bar** expands to one or more additional panes with tab-specific actions that can be useful for working with the object.

<< Back to Default Editor

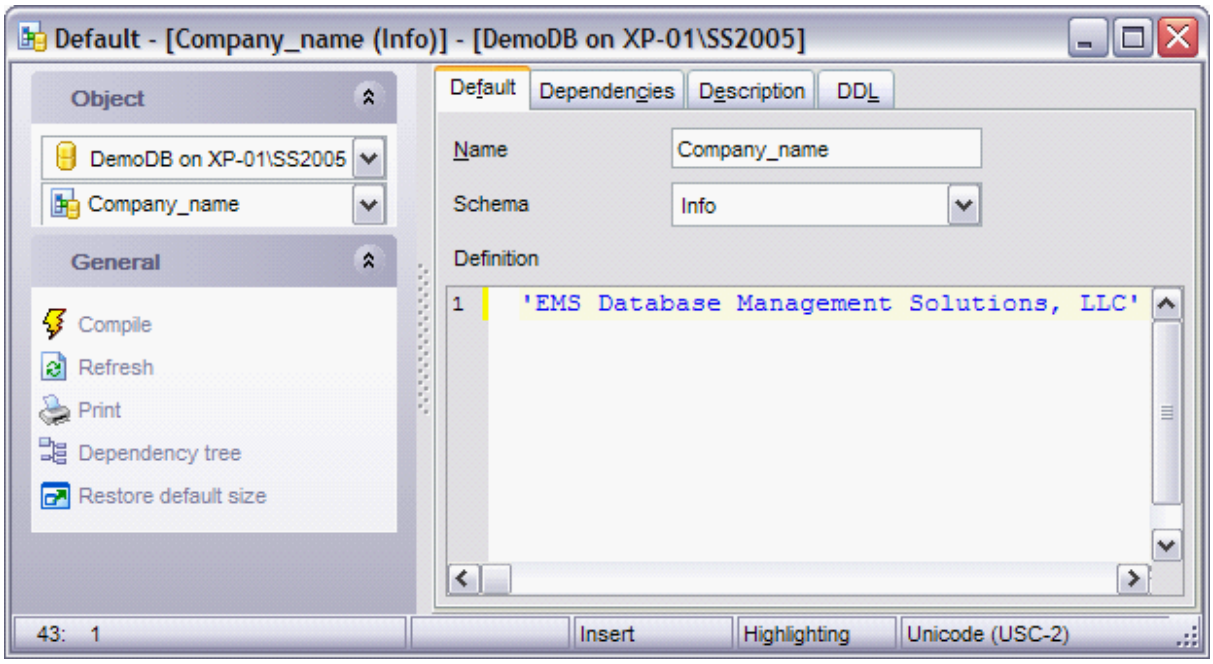
5.4.9.1.2 Editing default definition

Use the **Default** tab of **Default Editor** to create/edit a default and specify its definition.

Name
Enter a name for the new default, or modify the name of the default being edited.

Schema
Use the drop-down list to select the schema to which the default belongs.

© 2008 EMS Database Management Solutions



Definition
This area represents 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 containing alias data types. 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.

5.4.10 Rules

When bound to a column or a user-defined data type, a **rule** specifies the acceptable values that can be inserted into that column or to the fields having the specified data type.

Creating Rules

To create a new rule:
- select the **Database | New Object...** main menu item;
- select **Rule** in the Create New Object dialog;
- define rule properties using the appropriate tabs of Rule Editor.

Hint: To create a new rule, you can also right-click the **Rules** node or any object within this node in the DB Explorer tree and select the **New rule** item from the context menu.

To create a new rule with the same properties as one of existing rules has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a rule in the DB Explorer tree and select the **Duplicate Rule <rule_name>...** context menu item.

© 2008 EMS Database Management Solutions

Duplicate Object Wizard allows you to select the database to create a new rule in, and to edit the result SQL statement for creating the rule.

Editing Rules

To edit an existing rule:
- select the rule for editing in the DB Explorer tree (type the first letters of the rule name for quick search);
- right-click the object and select the **Edit Rule <rule_name>** context menu item, or simply double-click the rule;
- edit rule properties using the appropriate tabs of Rule Editor.

To change the name of a rule:
- select the rule to rename in the DB Explorer tree;
- right-click the rule alias and select the **Rename Rule <rule_name>...** item from the context menu;
- edit the rule name using the **Rename Object...** dialog.

Dropping Rules

To drop a rule:
- select the rule to drop in the DB Explorer tree;
- right-click the object and select the **Drop Rule <rule_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new rule;
Ctrl+O to edit the selected rule;
Ctrl+R to rename the rule;
Shift+Del to drop the object from the database.

5.4.10.1 Rule Editor

Rule Editor allows you to define rule properties. It opens automatically when you create a new rule and is available on editing an existing one (see Create rule and Edit rule for details).

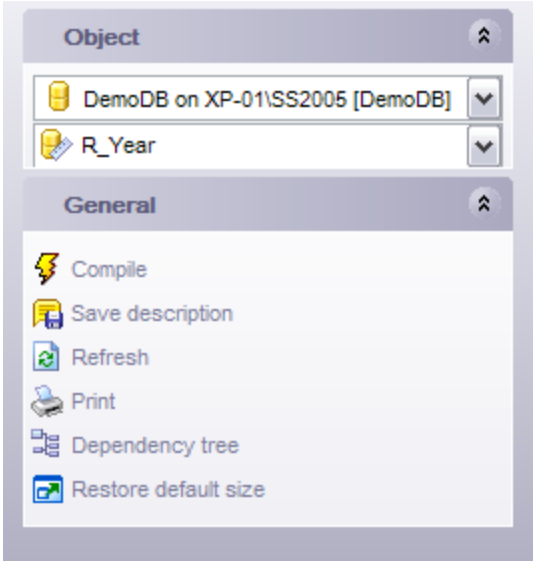
To open a rule in **Rule Editor**, double-click it in the DB Explorer tree.
- Using Navigation bar
- Editing rule definition
- Browsing object dependencies
- Editing rule description
- Viewing DDL definition

5.4.10.1.1 Using Navigation bar

The **Navigation bar** of **Rule Editor** allows you to:
- select the database where the object is located;

© 2008 EMS Database Management Solutions

- select the rule for editing;
- compile the object (if it is being created/modified);
- save the object description (if it has been modified);
- refresh the content of the active tab;
- print metadata of the object;
- view the dependency tree for the rule;
- restore the default size of the editor window.



Depending on the current tab selection, the **Navigation bar** expands to one or more additional panes with tab-specific actions that can be useful for working with the object.

<< Back to Rule Editor

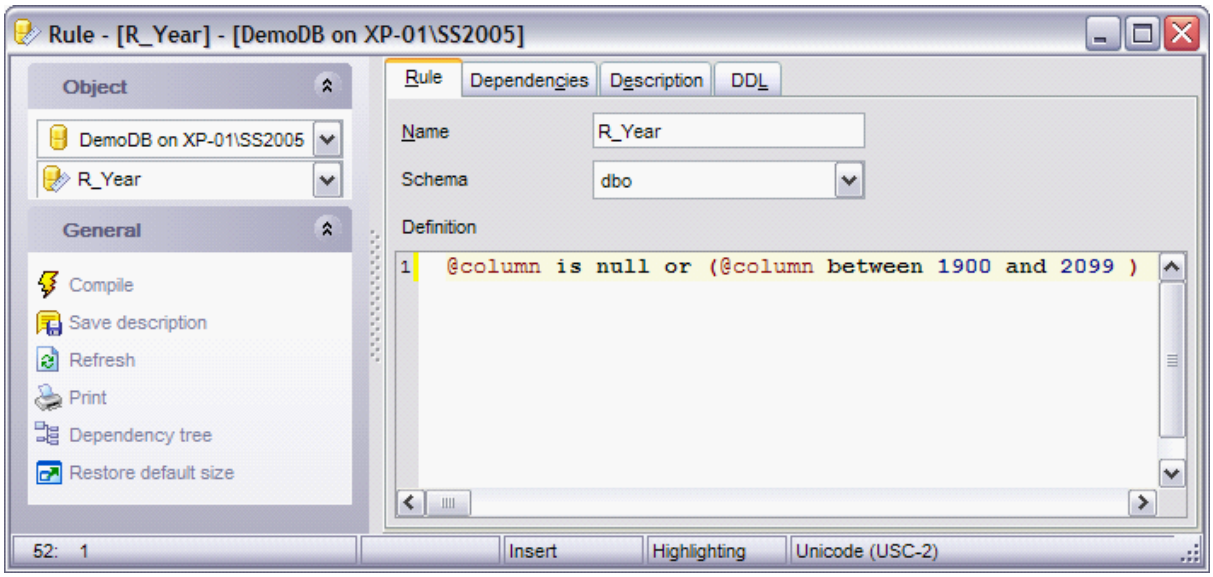
5.4.10.1.2 Editing rule definition

Use the **Rule** tab of **Rule Editor** to create/edit a rule and specify its definition.

Name
Enter a name for the new rule, or modify the name of the rule being edited.

Schema
Use the drop-down list to select the schema to which the rule belongs.

© 2008 EMS Database Management Solutions



Definition
This area represents the condition(s) that define the rule (*condition_expression*). This 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*).
Rule definition includes 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.

5.4.11 Queues

Queue is a schema object that holds incoming messages for a service. When a message arrives for a service, Service Broker puts the message on the queue associated with the service.

Creating Queues

To create a new queue:
- select the **Database | New Object...** main menu item;
- select **Queue** in the Create New Object dialog;
- define queue properties using the appropriate tabs of Queue Editor.

Hint: To create a new queue, you can also right-click the **Queues** node or any object within this node in the DB Explorer tree and select the **New queue** item from the context menu.

To create a new queue with the same properties as one of existing queues has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a queue in the DB Explorer tree and select the **Duplicate Queue <queue_name>...** context menu item.

© 2008 EMS Database Management Solutions

Duplicate Object Wizard allows you to select the database to create a new queue in, and to edit the result SQL statement for creating the queue.

Editing Queues

To edit an existing queue:
- select the queue for editing in the DB Explorer tree (type the first letters of the queue name for quick search);
- right-click the object and select the **Edit Queue <queue_name>** context menu item, or simply double-click the queue;
- edit queue properties using the appropriate tabs of Queue Editor.

To change the name of a queue:
- select the queue to rename in the DB Explorer tree;
- right-click the queue alias and select the **Rename Queue <queue_name>...** item from the context menu;
- edit the queue name using the **Rename Object...** dialog.

Dropping Queues

To drop a queue:
- select the queue to drop in the DB Explorer tree;
- right-click the object and select the **Drop Queue <queue_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new queue;
Ctrl+O to edit the selected queue;
Ctrl+R to rename the queue;
Shift+Del to drop the object from the database.

5.4.11.1 Queue Editor

Queue Editor allows you to define queue properties. It opens automatically when you create a new queue and is available on editing an existing one (see Create queue and Edit queue for details).

To open a queue in **Queue Editor**, double-click it in the DB Explorer tree.
- Using Navigation bar
- Editing queue definition
- Browsing object dependencies
- Editing rule description
- Viewing DDL definition

5.4.11.1.1 Using Navigation bar

The **Navigation bar** of **Queue Editor** allows you to:
- select the database where the object is located;
- select the queue for editing;
- compile the object (if it is being created/modified);

© 2008 EMS Database Management Solutions

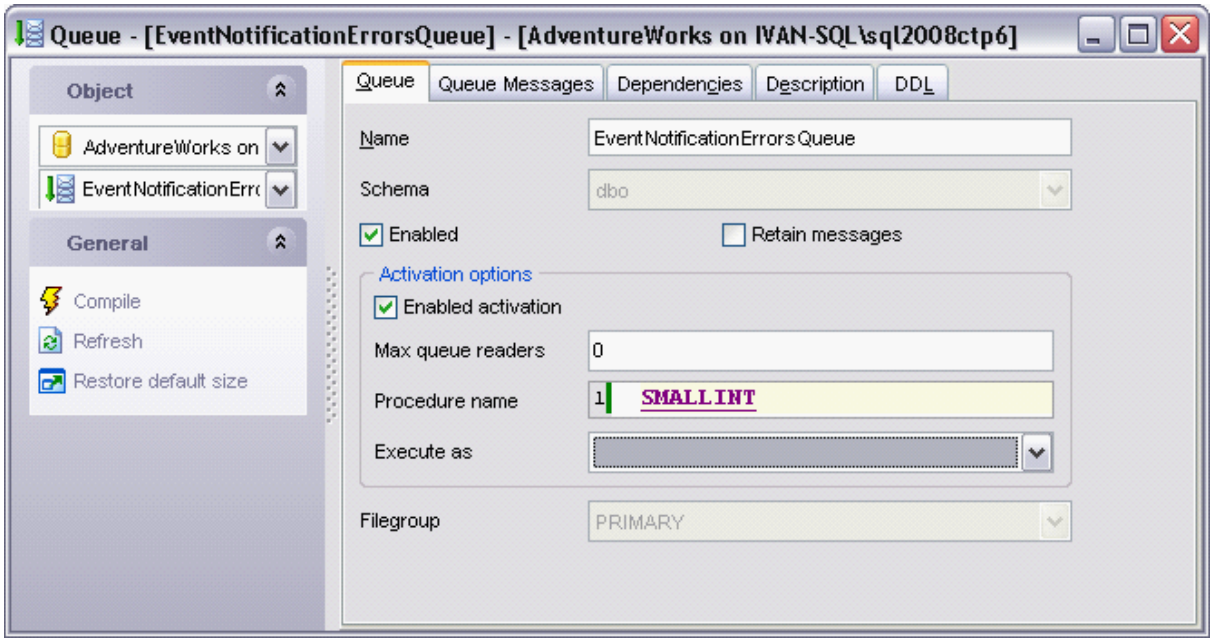
- save the object description (if it has been modified);
- refresh the content of the active tab;
- print metadata of the object;
- view the dependency tree for the queue;
- restore the default size of the editor window.

Depending on the current tab selection, the **Navigation bar** expands to one or more additional panes with tab-specific actions that can be useful for working with the object.

<< Back to Queue Editor

5.4.11.1.2 Editing queue definition

Use the **Queue** tab of **Queue Editor** to generate/edit a certificate and specify its properties.



Name
Is the name of the queue to create. This name must meet the guidelines for SQL Server identifiers.

Schema
Is the name of the schema to which the new queue belongs. The schema defaults to the default schema for the user that executes the statement.

Enabled
Specifies whether the queue is enabled. When the queue is disabled, no messages can be added to the queue or removed from the queue.

Retain messages
Specifies the retention setting for the queue. If the option is enabled, all messages sent

© 2008 EMS Database Management Solutions

or received on conversations that use this queue are retained in the queue until the conversations have ended.

Activation options

Enabled activation
Specifies information about which stored procedure you have to start to process messages in this queue.

procedure_name
Is the name of the stored procedure.

Max queue readers
Specifies the maximum number of instances of the activation stored procedure that the queue starts at the same time.

Execute as
Specifies the SQL Server database user account under which the activation stored procedure runs.

Filegroup
Specifies the SQL Server filegroup on which to create this queue.

© 2008 EMS Database Management Solutions

5.5 Non-schema Objects

Other types of objects are also stored in the database and can be created and manipulated with SQL, but are not contained in a schema:

- Assemblies
- DDL Triggers
- Partition Functions
- Partition Schemes
- Full-Text Catalogs
- Symmetric Keys
- Asymmetric Keys
- Certificates
- Message Types
- Contracts
- Services
- Routes
- Remote Service Bindings

Use the DB Explorer tree to navigate within the database(s) and the objects.

See also:

New Object dialog
Duplicate Object Wizard
Schemas
Schema objects
Database-level principals

5.5.1 Assemblies

An **assembly** is a managed application module containing class metadata and managed code as an object in SQL Server, against which CLR functions, stored procedures, triggers, user-defined aggregates, and user-defined types can be created in SQL Server.

Creating Assemblies

To create a new assembly:

- select the **Database | New Object...** main menu item;
- select **Assembly** in the Create New Object dialog;
- define assembly properties using the appropriate tabs of Assembly Editor.

Hint: To create a new assembly, you can also right-click the **Assemblies** node or any object within this node in the DB Explorer tree and select the **New Assembly** item from the context menu.

To create a new assembly with the same properties as one of existing assemblys has:

- select the **Database | Duplicate Object...** main menu item;

© 2008 EMS Database Management Solutions

- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a assembly in the DB Explorer tree and select the **Duplicate Assembly <assembly_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new assembly in, and to edit the result SQL statement for creating the assembly.

Editing Assemblies

To edit an existing assembly:

- select the assembly for editing in the DB Explorer tree (type the first letters of the assembly name for quick search);
- right-click the object and select the **Edit Assembly <assembly_name>** context menu item, or simply double-click the assembly;
- edit assembly properties using the appropriate tabs of Assembly Editor.

Dropping Assemblies

To drop a assembly:

- select the assembly to drop in the DB Explorer tree;
- right-click the object and select the **Drop Assembly <assembly_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new assembly;
Ctrl+O to edit the selected assembly;
Shift+Del to drop the object from the database.

5.5.1.1 Assembly Editor

Assembly Editor allows you to define assembly properties. It opens automatically when you create a new assembly and is available on editing an existing one (see Create assembly and Edit assembly for details).

To open an assembly in **Assembly Editor**, double-click it in the DB Explorer tree.

- Using Navigation bar
- Editing assembly definition
- Browsing object dependencies
- Editing object description
- Viewing DDL definition
- Setting object permissions

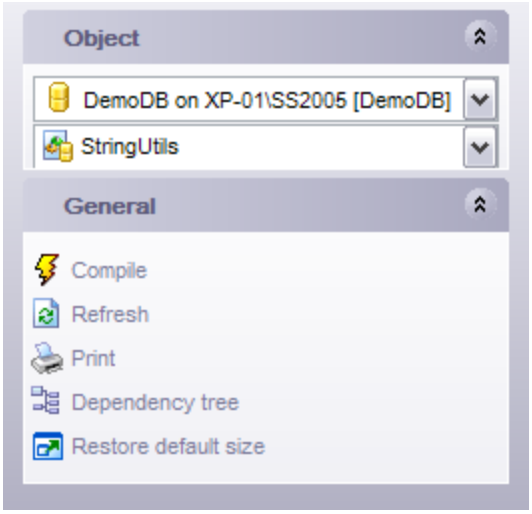
5.5.1.1.1 Using Navigation bar

The **Navigation bar** of **Assembly Editor** allows you to:

- select the database where the object is located;
- select the assembly for editing;
- compile the object (if it is being created/modified);
- save the object description (if it has been modified);

© 2008 EMS Database Management Solutions

- refresh the content of the active tab;
- print metadata of the object;
- view the dependency tree for the object;
- restore the default size of the editor window.



Depending on the current tab selection, the **Navigation bar** expands to one or more additional panes with tab-specific actions that can be useful for working with the object.

<< Back to Assembly Editor

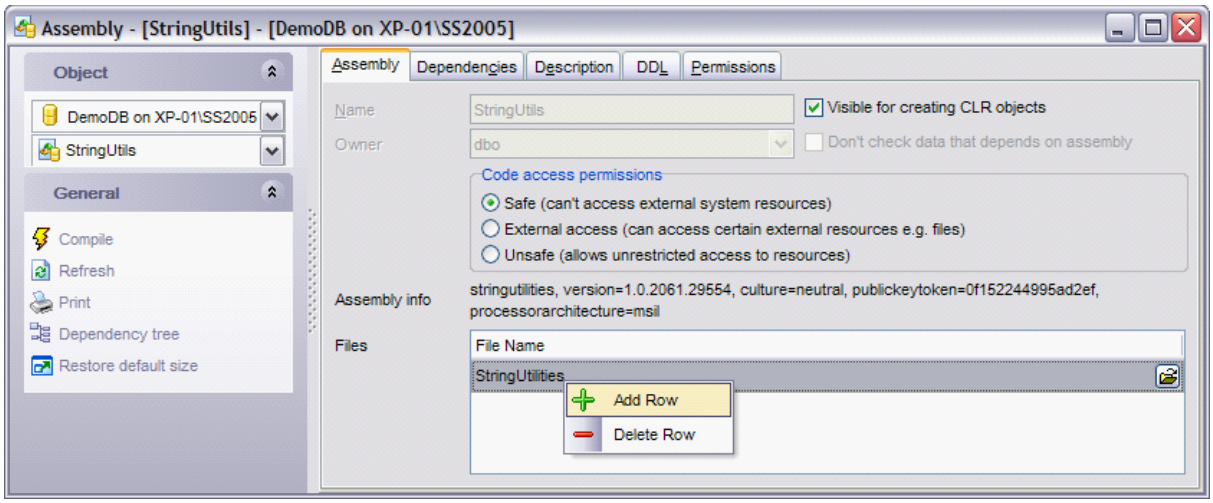
5.5.1.1.2 Editing assembly definition

Use the **Assembly** tab of **Assembly Editor** to create/edit an assembly and specify its properties.

Name
Enter a name for the new assembly, or view the name of the assembly being edited.

Owner
Use the drop-down list to select the owner of the assembly (the name of a database user or role).

© 2008 EMS Database Management Solutions



Visible for creating CLR objects
This option specifies whether the assembly is visible for creating common language runtime (CLR) functions, stored procedures, triggers, user-defined aggregates, and user-defined types against it. If disabled, the assembly is intended to be called only by other assemblies. If there are existing CLR database objects already created against the assembly, the visibility of the assembly cannot be changed.

Don't check data that depends on assembly
If this option is unchecked, SQL Server refreshes even those assemblies upon which persisted data in the database is dependent.

Code access permissions
Specifies a set of code access permissions granted to the assembly when it is accessed by SQL Server.

Safe
The most restrictive permission set. Code executed by an assembly cannot access external system resources such as files, the network, environment variables, or the registry.

External access
Allows assemblies to access certain external system resources such as files, networks, environmental variables, and the registry.

Unsafe
Allows assemblies unrestricted access to resources, both within and outside SQL Server. Code executing from within an Unsafe assembly can call unmanaged code.

The **Assembly info** panel is used to display information about the assembly or application, including a description, version information, company name, product name, and more. This information corresponds to the values of global assembly attributes.

The **Files** list allows you to manage files that correspond to the assembly, and specify the local path or network location.

© 2008 EMS Database Management Solutions

To add a file, right-click within the *Files* area and select the **Add Row** popup menu item.
To remove a file, right-click the file name within the *Files* area and select the **Delete Row** popup menu item.

5.5.2 DDL Triggers

A **trigger** is a special kind of stored procedure that automatically executes when an event occurs in the database.

DDL triggers fire in response to a variety of data definition language (DDL) events which are primarily *CREATE*, *ALTER* and *DROP* statements; logon triggers fire in response to the *LOGON* event that is raised when a user sessions is being established.

Creating DDL Triggers

To create a new DDL Trigger:
- select the **Database | New Object...** main menu item;
- select **DDL Trigger** in the Create New Object dialog;
- define DDL Trigger properties using the appropriate tabs of DDL Trigger Editor.

Hint: To create a new DDL Trigger, you can also right-click the **DDL Triggers** node or any object within this node in the DB Explorer tree and select the **New DDL Trigger** item from the context menu.

To create a new DDL Trigger with the same properties as one of existing DDL Triggers has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a DDL Trigger in the DB Explorer tree and select the **Duplicate DDL Trigger <DDL_trigger_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new DDL Trigger in, and to edit the result SQL statement for creating the DDL Trigger.

Editing DDL Triggers

To edit an existing DDL Trigger:
- select the DDL Trigger for editing in the DB Explorer tree (type the first letters of the DDL Trigger name for quick search);
- right-click the object and select the **Edit DDL Trigger <DDL_trigger_name>** context menu item, or simply double-click the DDL Trigger;
- edit DDL Trigger properties using the appropriate tabs of DDL Trigger Editor.

To change the name of a DDL Trigger:
- select the DDL Trigger to rename in the DB Explorer tree;
- right-click the DDL Trigger alias and select the **Rename DDL Trigger <DDL_trigger_name>...** item from the context menu;
- edit the DDL Trigger name using the **Rename Object...** dialog.

© 2008 EMS Database Management Solutions

Dropping DDL Triggers

To drop a DDL Trigger:
- select the DDL Trigger to drop in the DB Explorer tree;
- right-click the object and select the **Drop DDL Trigger <DDL_trigger_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new DDL Trigger;
Ctrl+O to edit the selected DDL Trigger;
Ctrl+R to rename the DDL Trigger;
Shift+Del to drop the object from the database.

5.5.2.1 DDL Trigger Editor

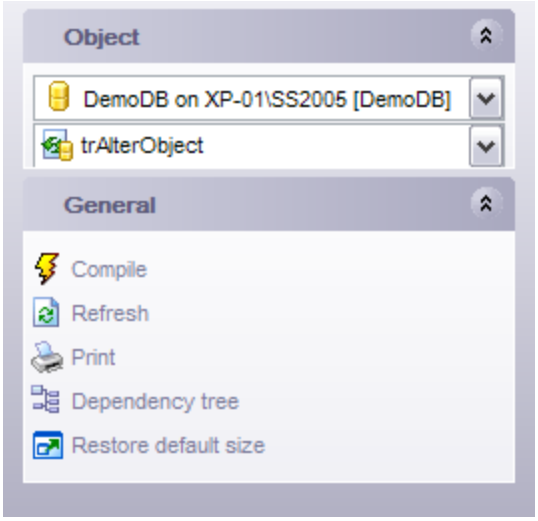
DDL Trigger Editor allows you to define DDL trigger properties (the scope of the DDL triggers is applied to the current database, i.e. the trigger fires whenever *event_type* or *event_group* happens in the current database). It opens automatically when you create a new DDL trigger and is available on editing an existing one (see Create DDL trigger and Edit DDL trigger for details).

To open a DDL trigger in **DDL Trigger Editor**, double-click it in the DB Explorer tree.
- Using Navigation bar
- Editing DDL trigger definition
- Browsing object dependencies
- Editing object description
- Viewing DDL definition

5.5.2.1.1 Using Navigation bar

The **Navigation bar** of **DDL Trigger Editor** allows you to:
- select the database where the object is located;
- select the DDL trigger for editing;
- compile the object (if it is being created/modified);
- save the object description (if it has been modified);
- refresh the content of the active tab;
- print metadata of the object;
- view the dependency tree for the object;
- restore the default size of the editor window.

© 2008 EMS Database Management Solutions



Depending on the current tab selection, the **Navigation bar** expands to one or more additional panes with tab-specific actions that can be useful for working with the object.

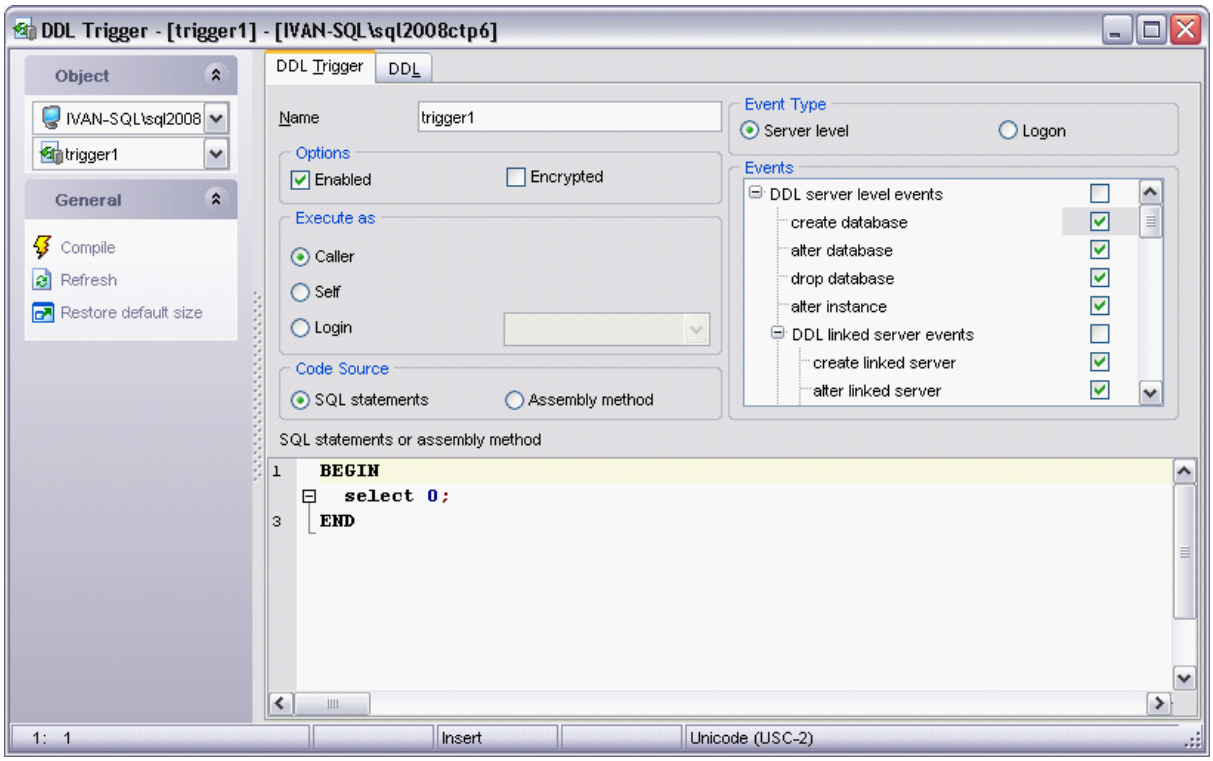
[<< Back to DDL Trigger Editor]

5.5.2.1.2 Editing DDL trigger definition

Use the **DDL Trigger** tab of **DDL Trigger Editor** to create/edit a DDL trigger and specify its properties.

Name
Enter a name for the new DDL trigger, or modify the name of the DDL trigger being edited.

© 2008 EMS Database Management Solutions



Enabled
Enables/disables the trigger immediately after it is created. A disabled trigger still exists as an object in the current database, but does not fire.

Encrypted
If this option is selected, the text of the *CREATE TRIGGER* statement is encrypted. Using this option prevents the trigger from being published as part of SQL Server replication.

Execute as
Specify the security context under which the trigger is to be executed: *Caller*, *Self* or *User* (select which user account should be used to validate permissions on any database objects that are referenced by the trigger).

Code Source
Select the trigger code source: it can be a direct set of SQL statements or an external procedure call from a .NET assembly or DLL.

Event type
Specifies whether a DDL or logon trigger will be created

Events
This area lists the DDL events as groups. The trigger will fire after execution of any data definition language event selected in this list. Set the corresponding flag next to an event or event group name. (disabled for a logon trigger)

SQL statements or assembly method
This area allows you to set the trigger conditions and actions. Trigger conditions specify

© 2008 EMS Database Management Solutions

additional criteria that determine whether the tried DDL statements cause the trigger actions to be performed.
The trigger actions take effect when the DDL operation is performed.

5.5.3 Partition Functions

A **partition function** specifies how a table or index is partitioned. The function maps the domain into a set of partitions. When creating a partition function, you specify the number of partitions, the partitioning column, and the range of partition column values for each partition. Note that when you specify the partitioning column, you can only specify one.

Creating Partition Functions

To create a new partition function:
- select the **Database | New Object...** main menu item;
- select **Partition Function** in the Create New Object dialog;
- define partition function properties using the appropriate tabs of Partition Function Editor.

Hint: To create a new partition function, you can also right-click the **Partition Functions** node or any object within this node in the DB Explorer tree and select the **New Partition Function** item from the context menu.

To create a new partition function with the same properties as one of existing partition functions has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a partition function in the DB Explorer tree and select the **Duplicate Partition Function <partition_function_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new partition function in, and to edit the result SQL statement for creating the partition function.

Editing Partition Functions

To edit an existing partition function:
- select the partition function for editing in the DB Explorer tree (type the first letters of the partition function name for quick search);
- right-click the object and select the **Edit Partition Function <partition_function_name>** context menu item, or simply double-click the partition function;
- edit partition function properties using the appropriate tabs of Partition Function Editor.

To change the name of a partition function:
- select the partition function to rename in the DB Explorer tree;
- right-click the partition function alias and select the **Rename Partition Function <partition_function_name>...** item from the context menu;
- edit the partition function name using the **Rename Object...** dialog.

© 2008 EMS Database Management Solutions

Dropping Partition Functions

To drop a partition function:
- select the partition function to drop in the DB Explorer tree;
- right-click the object and select the **Drop Partition Function <partition_function_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new partition function;
Ctrl+O to edit the selected partition function;
Ctrl+R to rename the partition function;
Shift+Del to drop the object from the database.

5.5.3.1 Partition Function Editor

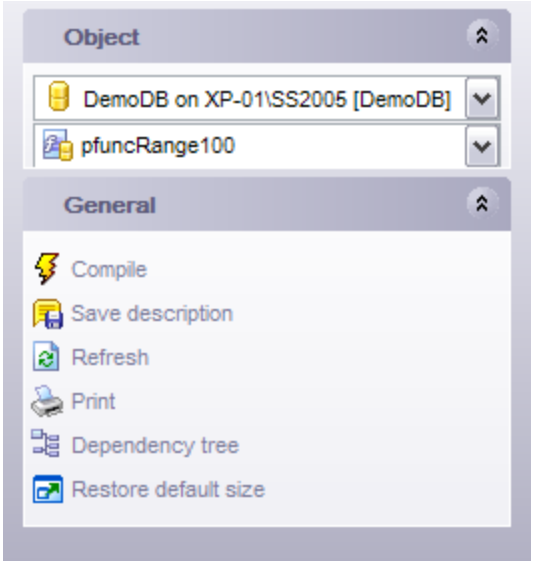
Partition Function Editor allows you to define partition function properties. It opens automatically when you create a new partition function and is available on editing an existing one (see Create partition function and Edit partition function for details).

To open a partition function in **Partition Function Editor**, double-click it in the DB Explorer tree.
- Using Navigation bar
- Editing partition function definition
- Browsing object dependencies
- Editing object description
- Viewing DDL definition

5.5.3.1.1 Using Navigation bar

The **Navigation bar** of **Partition Function Editor** allows you to:
- select the database where the object is located;
- select the partition function for editing;
- compile the object (if it is being created/modified);
- save the object description (if it has been modified);
- refresh the content of the active tab;
- print metadata of the object;
- view the dependency tree for the object;
- restore the default size of the editor window.

© 2008 EMS Database Management Solutions



Depending on the current tab selection, the **Navigation bar** expands to one or more additional panes with tab-specific actions that can be useful for working with the object.

[<< Back to Partition Function Editor]

5.5.3.1.2 Editing partition function definition

Use the **Partition Function** tab of **Partition Function Editor** to create/edit a partition function and specify its properties.

Name
Enter a name for the new partition function, or view the name of the partition function being edited.

Parameter

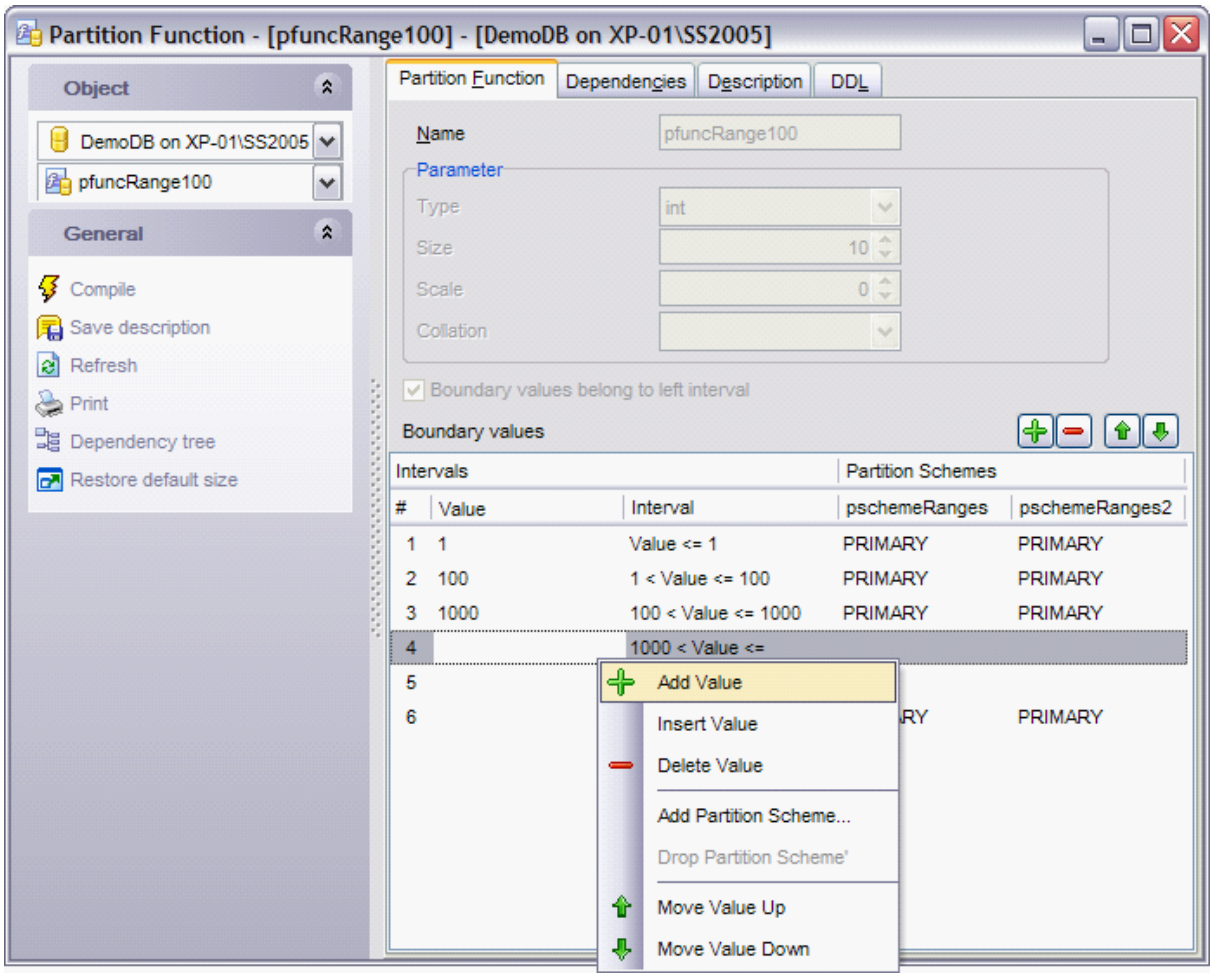
Type
Specifies the data type of the column used for partitioning.

Size
Specifies the size of the data type. This box is not editable if the size of the selected data type is fixed or not applicable.

Scale
Specifies the maximum number of decimal digits that can be stored to the right of the decimal point. This box is editable only for *numeric* and *decimal* data types only.

Collation
Defines the collation for *text*, *char*, *nchar*, *varchar*, *nvarchar* values.

© 2008 EMS Database Management Solutions



Boundary values belong to left interval
This option specifies to which side of each boundary value interval, left or right, the **boundary value** belongs, when interval values are sorted by the Database Engine in ascending order from left to right. If checked, *LEFT* is specified.

Boundary values
This area lists the boundary values specified by the partition function.

To add a value, right-click within the *Boundary values* area and select the **Add Value** popup menu item, or press the corresponding **Add value** button.

To remove a value, right-click the value within the *Boundary values* area and select the **Delete Value** popup menu item, or press the corresponding **Delete value** button.

Use the **Move up** and the **Move down** buttons or popup menu items to reorder the intervals.

To add a partition scheme which will use the current partition function, right-click within

© 2008 EMS Database Management Solutions

the *Boundary values* area and select the **Add Partition Scheme...** popup menu item. To drop the partition scheme which uses the current partition function, right-click within the *Boundary values* area and select **Drop Partition Scheme <partition_scheme_name>** popup menu item.

5.5.4 Partition Schemes

A **partition scheme** maps the partitions produced by a partition function to a set of filegroups that you define. A partition scheme can use only one partition function. However, a partition function can participate in more than one partition scheme.

Creating Partition Schemes

To create a new partition scheme:
- select the **Database | New Object...** main menu item;
- select **Partition Scheme** in the Create New Object dialog;
- define partition scheme properties using the appropriate tabs of Partition Scheme Editor.

Hint: To create a new partition scheme, you can also right-click the **Partition Schemes** node or any object within this node in the DB Explorer tree and select the **New Partition Scheme** item from the context menu.

To create a new partition scheme with the same properties as one of existing partition schemes has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a partition scheme in the DB Explorer tree and select the **Duplicate Partition Scheme <partition_scheme_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new partition scheme in, and to edit the result SQL statement for creating the partition scheme.

Editing Partition Schemes

To edit an existing partition scheme:
- select the partition scheme for editing in the DB Explorer tree (type the first letters of the partition scheme name for quick search);
- right-click the object and select the **Edit Partition Scheme <partition_scheme_name>** context menu item, or simply double-click the partition scheme;
- edit partition scheme properties using the appropriate tabs of Partition Scheme Editor.

To change the name of a partition scheme:
- select the partition scheme to rename in the DB Explorer tree;
- right-click the partition scheme alias and select the **Rename Partition Scheme <partition_scheme_name>...** item from the context menu;
- edit the partition scheme name using the **Rename Object...** dialog.

Dropping Partition Schemes

© 2008 EMS Database Management Solutions

To drop a partition scheme:
- select the partition scheme to drop in the DB Explorer tree;
- right-click the object and select the **Drop Partition Scheme <partition_scheme_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new partition scheme;
Ctrl+O to edit the selected partition scheme;
Ctrl+R to rename the partition scheme;
Shift+Del to drop the object from the database.

5.5.4.1 Partition Scheme Editor

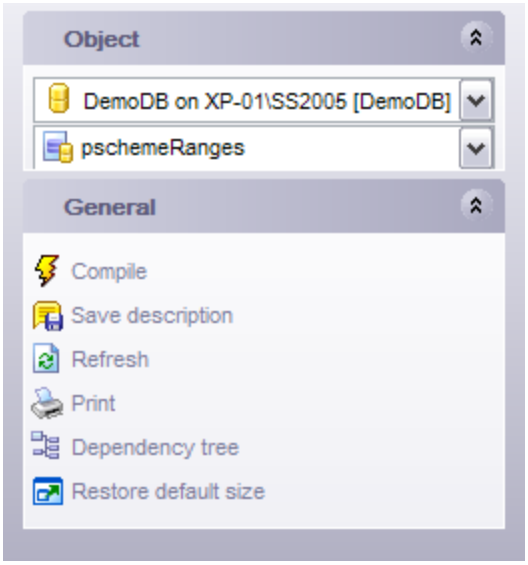
Partition Scheme Editor allows you to define partition scheme properties. It opens automatically when you create a new partition scheme and is available on editing an existing one (see Create Partition Scheme and Edit Partition Scheme for details).

To open a partition scheme in **Partition Scheme Editor**, double-click it in the DB Explorer tree.
- Using Navigation bar
- Editing partition scheme definition
- Browsing object dependencies
- Editing object description
- Viewing DDL definition

5.5.4.1.1 Using Navigation bar

The **Navigation bar** of **Partition Scheme Editor** allows you to:
- select the database where the object is located;
- select the partition scheme for editing;
- compile the object (if it is being created/modified);
- save the object description (if it has been modified);
- refresh the content of the active tab;
- print metadata of the object;
- view the dependency tree for the object;
- restore the default size of the editor window.

© 2008 EMS Database Management Solutions



Depending on the current tab selection, the **Navigation bar** expands to one or more additional panes with tab-specific actions that can be useful for working with the object.

[<< Back to Partition Scheme Editor]

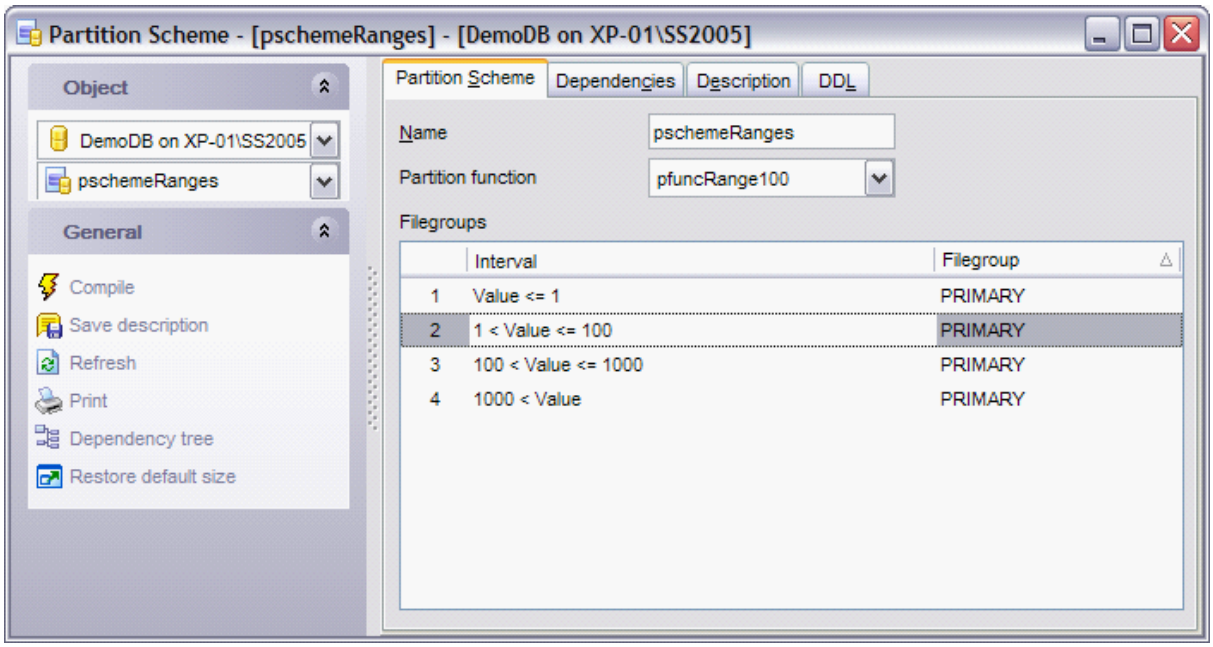
5.5.4.1.2 Editing partition scheme definition

Use the **Partition Scheme** tab of **Partition Scheme Editor** to create/edit a partition scheme and specify its properties.

Name
Enter a name for the new partition scheme, or modify the name of the partition scheme being edited.

Partition function
Use the drop-down list to select the name of the partition function that will use the partition scheme.

© 2008 EMS Database Management Solutions



Filegroups
Partitions created by the partition function are mapped to the filegroups specified in the partition scheme.
Select the filegroup name in the **Filegroup** drop-down list for each of the intervals specified by the partition function. If *PRIMARY* is selected, the partition is stored in the primary filegroup.

Hint: Filegroups can be created on the Filegroups page of the Database Properties dialog.

5.5.5 Full-Text Catalogs

Full-text catalogs and **full-text indexes** are the objects that are implemented in Microsoft® SQL Server for efficient full-text administration which is carried out at the server, database, table and column levels.

Like regular indexes, **full-text indexes** can be automatically updated as data are modified in the associated tables. Index updating is normally performed as an asynchronous process that runs in the background during periods of low database activity. Tables with the same update characteristics (e.g. small number of changes and large number of changes) should be grouped together and assigned to the same **full-text catalog**.

One **full-text catalog** can have several *full-text indexes*, but a full-text index can only be part of one full-text catalog.

Creating Full-Text Catalogs

To create a new full-text catalog:
- select the **Database | New Object...** main menu item;
- select **Full-Text Catalog** in the Create New Object dialog;

© 2008 EMS Database Management Solutions

- define full-text catalog properties using the appropriate tabs of Full-Text Catalog Editor.

Hint: To create a new full-text catalog, you can also right-click the **Full-Text Catalogs** node or any object within this node in the DB Explorer tree and select the **New Full-Text Catalog** item from the context menu.

To create a new full-text catalog with the same properties as one of existing full-text catalogs has:

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

Alternatively, you can right-click a full-text catalog in the DB Explorer tree and select the **Duplicate Full-Text Catalog <full-text_catalog_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new full-text catalog in, and to edit the result SQL statement for creating the full-text catalog.

Editing Full-Text Catalogs

To edit an existing full-text catalog:

- select the full-text catalog for editing in the DB Explorer tree (type the first letters of the full-text catalog name for quick search);
- right-click the object and select the **Edit Full-Text Catalog <full-text_catalog_name>** context menu item, or simply double-click the full-text catalog;
- edit full-text catalog properties using the appropriate tabs of Full-Text Catalog Editor.

Dropping Full-Text Catalogs

To drop a full-text catalog:

- select the full-text catalog to drop in the DB Explorer tree;
- right-click the object and select the **Drop Full-Text Catalog <full-text_catalog_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new full-text catalog;
Ctrl+O to edit the selected full-text catalog;
Shift+Del to drop the object from the database.

5.5.5.1 Full-Text Catalog Editor

Full-Text Catalog Editor allows you to define full-text catalog properties. It opens automatically when you create a new full-text catalog and is available on editing an existing one (see Create full-text catalog and Edit full-text catalog for details).

To open a full-text catalog in **Full-Text Catalog Editor**, double-click it in the DB Explorer tree.

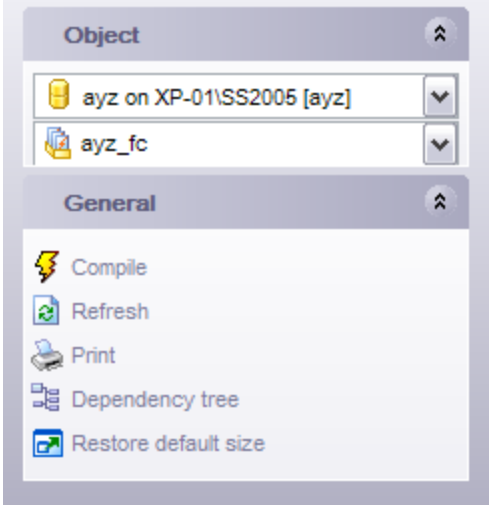
- Using Navigation bar
- Editing full-text catalog definition
- Managing full-text indexes
- Viewing DDL definition

© 2008 EMS Database Management Solutions

5.5.5.1.1 Using Navigation bar

The **Navigation bar** of **Full-Text Catalog Editor** allows you to:

- select the database where the object is located;
- select the full-text catalog for editing;
- compile the object (if it is being created/modified);
- refresh the content of the active tab;
- print metadata of the object;
- view the dependency tree for the object;
- restore the default size of the editor window.



Depending on the current tab selection, the **Navigation bar** expands to one or more additional panes with tab-specific actions that can be useful for working with the object.

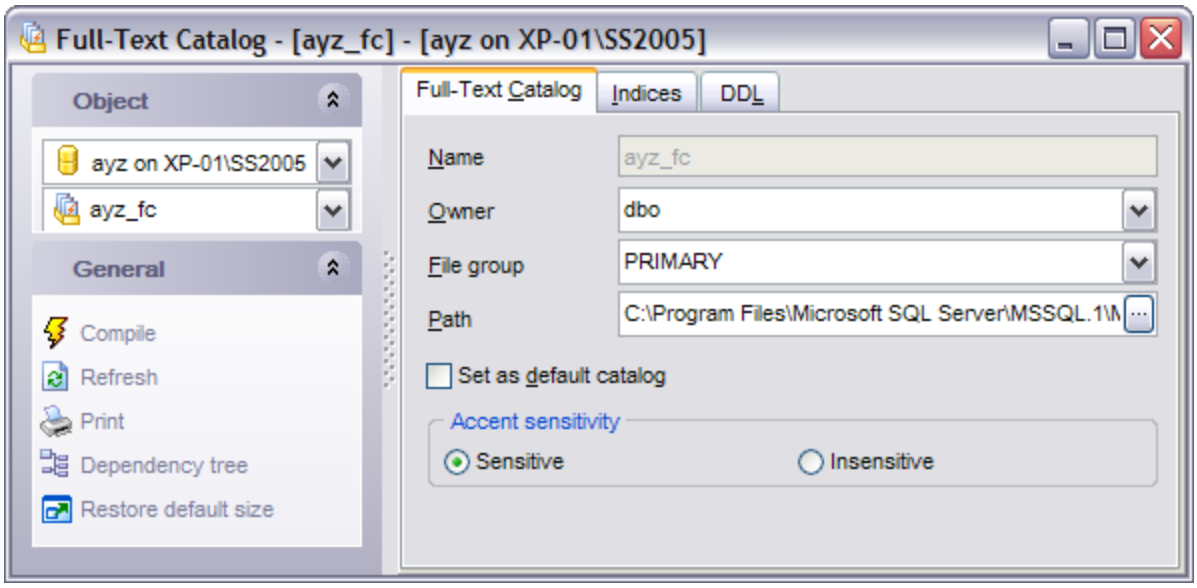
<< Back to Full-Text Catalog Editor

5.5.5.1.2 Editing full-text catalog definition

Use the **Full-Text Catalog** tab of **Full-Text Catalog Editor** to create/edit a full-text catalog and specify its properties.

Name
Enter a name for the new full-text catalog, or view the name of the full-text catalog being edited.

© 2008 EMS Database Management Solutions



Owner
Use the drop-down list to select the owner of the full-text catalog (the name of a database user or role).

File group
Use the drop-down list to select the filegroup which will include the new catalog. The default full-text filegroup is the *PRIMARY* filegroup for the database.

Path
Use the ellipsis button to specify the root directory for the catalog.
Note: Full-text catalogs must be created on a local hard drive associated with an instance of SQL Server.

Set as default catalog
This option specifies that the catalog is the default catalog. When full-text indexes are created without a full-text catalog explicitly specified, the default catalog is used.

Accent sensitivity
Specify whether the catalog is *accent sensitive* or *accent insensitive* for full-text indexing.

5.5.5.1.3 Managing full-text indexes

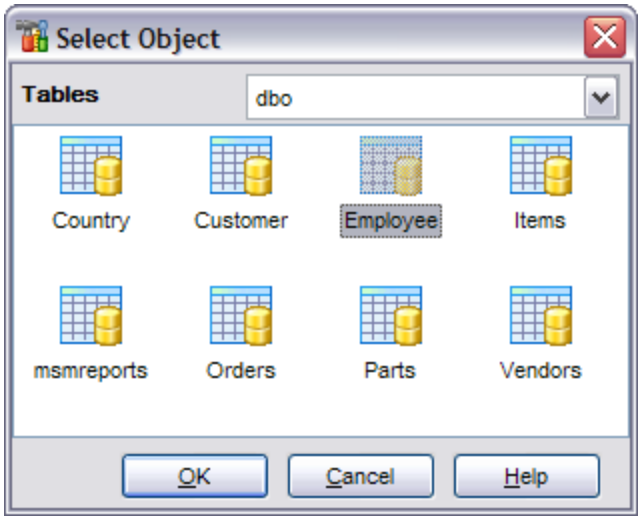
The **Indices** tab of **Full-Text Catalog Editor** allows you to manage full-text indexes and specify their properties.

Indices
This area lists the tables containing indexes that will be used by the full-text catalog.

To add a full-text index, right-click within the *Indices* area and select the **Add index** popup menu item, or use the corresponding button at the bottom.

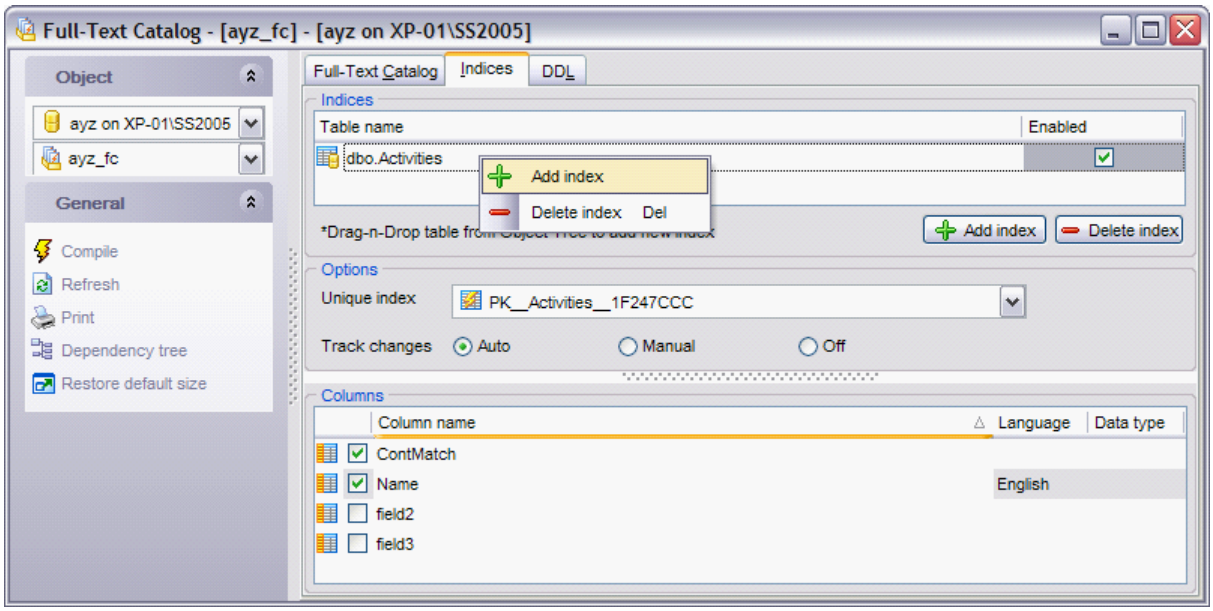
© 2008 EMS Database Management Solutions

The **Select Object** dialog allows you to specify a table to use indexes from.



Hint: The context menu of the **Select Object** dialog allows you to toggle the *Icons*/*List* object representation mode.

To remove a full-text index, right-click the index within the *Indices* area and select the **Delete index** popup menu item, or use the corresponding button at the bottom.



Options
If a table index is enabled (i.e. the *Enabled* flag is set in the **Indices** list), you can select the unique index and set the **Track changes** property:

Auto

© 2008 EMS Database Management Solutions

Specifies that SQL Server automatically updates the full-text index as the data are modified in the associated tables.

Manual
Specifies that the change-tracking log will be propagated either on a schedule using SQL Server Agent, or manually by the user.

Off
Specifies that SQL Server does not keep a list of changes to the indexed data.

Columns
This area lists the fields of the selected table. Tick off the columns that will be included in the full-text index, and set the language of the data stored in each column.

5.5.6 Symmetric Keys

In SQL Server 2005, data can be encrypted with a hierarchical encryption and key management infrastructure. Each layer encrypts the layer below it by using a combination of the following mechanisms for encryption: **symmetric keys**, asymmetric keys, certificates.

A **symmetric key** is a single key that is used for both encryption and decryption. Encryption and decryption by using a symmetric key is fast and suitable for routine use with sensitive data in the database.

Creating Symmetric Keys

To create a new symmetric key:
- select the **Database | New Object...** main menu item;
- select **Symmetric Key** in the Create New Object dialog;
- define symmetric key properties using the appropriate tabs of Symmetric Key Editor.

Hint: To create a new symmetric key, you can also right-click the **Symmetric Keys** node or any object within this node in the DB Explorer tree and select the **New Symmetric Key** item from the context menu.

To create a new symmetric key with the same properties as one of existing symmetric keys has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a symmetric key in the DB Explorer tree and select the **Duplicate Symmetric Key <symmetric_key_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new symmetric key in, and to edit the result SQL statement for creating the symmetric key.

Editing Symmetric Keys

To edit an existing symmetric key:

© 2008 EMS Database Management Solutions

- select the symmetric key for editing in the DB Explorer tree (type the first letters of the symmetric key name for quick search);
- right-click the object and select the **Edit Symmetric Key <symmetric_key_name>** context menu item, or simply double-click the symmetric key;
- edit symmetric key properties using the appropriate tabs of Symmetric Key Editor.

Dropping Symmetric Keys

To drop a symmetric key:
- select the symmetric key to drop in the DB Explorer tree;
- right-click the object and select the **Drop Symmetric Key <symmetric_key_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new symmetric key;
Ctrl+O to edit the selected symmetric key;
Shift+Del to drop the object from the database.

5.5.6.1 Symmetric Key Editor

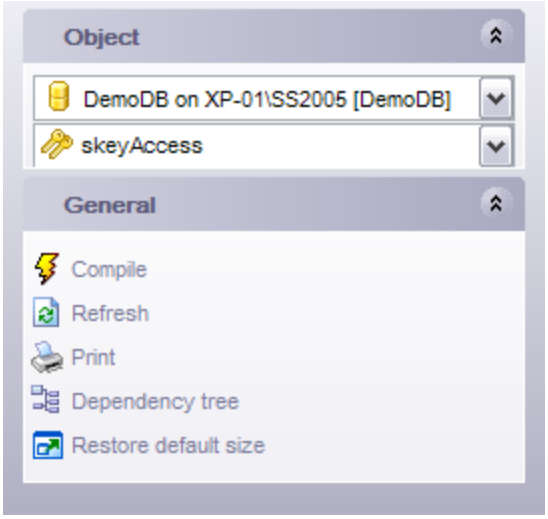
Symmetric Key Editor allows you to define symmetric key properties. It opens automatically when you create a new symmetric key and is available on editing an existing one (see Create symmetric key and Edit symmetric key for details).

To open a symmetric key in **Symmetric Key Editor**, double-click it in the DB Explorer tree.
- Using Navigation bar
- Editing symmetric key definition
- Browsing object dependencies
- Viewing DDL definition
- Setting object permissions

5.5.6.1.1 Using Navigation bar

The **Navigation bar** of **Symmetric Key Editor** allows you to:
- select the database where the object is located;
- select the symmetric key for editing;
- compile the object (if it is being created/modified);
- refresh the content of the active tab;
- print metadata of the object;
- view the dependency tree for the object;
- restore the default size of the editor window.

© 2008 EMS Database Management Solutions



Depending on the current tab selection, the **Navigation bar** expands to one or more additional panes with tab-specific actions that can be useful for working with the object.

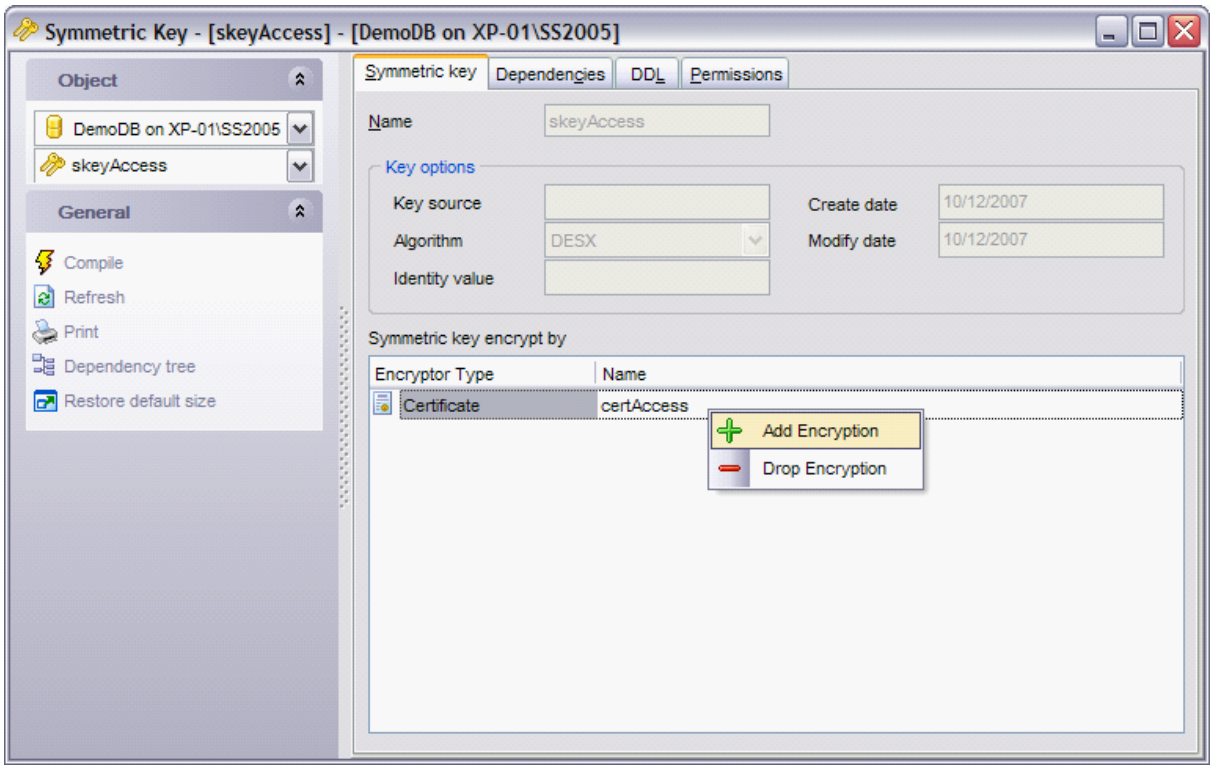
[<< Back to Symmetric Key Editor]

5.5.6.1.2 Editing symmetric key definition

Use the **Symmetric key** tab of **Symmetric Key Editor** to generate/edit a symmetric key and specify its properties.

Name
Enter a name for the new symmetric key, or view the name of the symmetric key being edited.

© 2008 EMS Database Management Solutions



Key options

Key source
Specify a pass phrase that will be used to derive the key from.

Algorithm
Use the drop-down list to specify an algorithm to be used by the symmetric key. Possible values are: *DES*, *TRIPLE_DES*, *RC2*, *RC4*, *DESX*, *AES_128*, *AES_192*, *AES_256*.

Identity value
Specify an identity phrase from which to generate a GUID for tagging data that is encrypted with a temporary key.

Create date
Stores the date when the symmetric key was created.

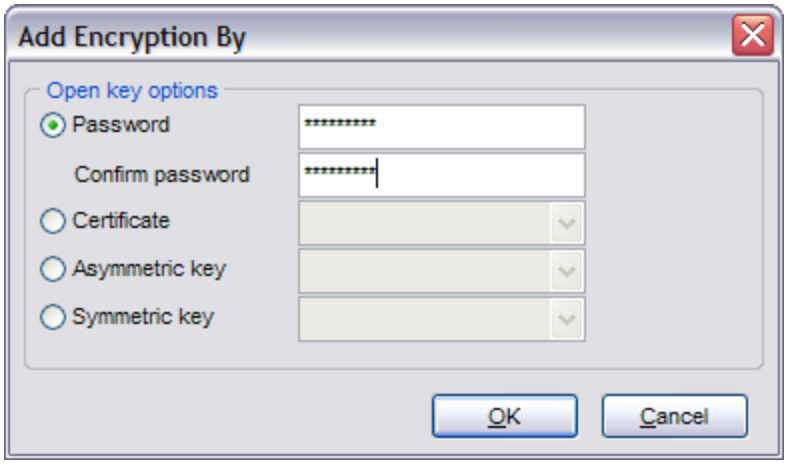
Modify date
Stores the date when the symmetric key was last modified.

Symmetric key encrypt by
This area lists the encrypting mechanisms used by the symmetric key (the *ENCRYPTION BY* clause).

To add an encrypting mechanism, right-click within the *Symmetric key encrypt by* area and select the **Add encryption** popup menu item.

© 2008 EMS Database Management Solutions

The **Add Encryption By** dialog allows you to specify a password, a certificate, an asymmetric key, or a symmetric key to be used to encrypt the key that is being created.



To remove an encrypting mechanism, right-click the encrypting mechanism within the *Symmetric key encrypt by* area and select the **Drop encryption** popup menu item.

5.5.7 Asymmetric Keys

In SQL Server 2005, data can be encrypted with a hierarchical encryption and key management infrastructure. Each layer encrypts the layer below it by using a combination of the following mechanisms for encryption: symmetric keys, **asymmetric keys**, certificates.

An **asymmetric key** is composed of a private key and the corresponding public key. Each key can decrypt data encrypted by the other. Asymmetric encryption and decryption provide a higher level of security than symmetric encryption. An asymmetric key can be used to encrypt a symmetric key for storage in a database.

Creating Asymmetric Keys

To create a new asymmetric key:

- select the **Database | New Object...** main menu item;
- select **Asymmetric Key** in the Create New Object dialog;
- define asymmetric key properties using the appropriate tabs of Asymmetric Key Editor.

Hint: To create a new asymmetric key, you can also right-click the **Asymmetric Keys** node or any object within this node in the DB Explorer tree and select the **New Asymmetric Key** item from the context menu.

To create a new asymmetric key with the same properties as one of existing asymmetric keys has:

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

Alternatively, you can right-click a asymmetric key in the DB Explorer tree and select the **Duplicate Asymmetric Key <asymmetric_key_name>...** context menu item.

© 2008 EMS Database Management Solutions

Duplicate Object Wizard allows you to select the database to create a new asymmetric key in, and to edit the result SQL statement for creating the asymmetric key.

Editing Asymmetric Keys

To edit an existing asymmetric key:
- select the asymmetric key for editing in the DB Explorer tree (type the first letters of the asymmetric key name for quick search);
- right-click the object and select the **Edit Asymmetric Key <asymmetric_key_name>** context menu item, or simply double-click the asymmetric key;
- edit asymmetric key properties using the appropriate tabs of Asymmetric Key Editor.

Dropping Asymmetric Keys

To drop a asymmetric key:
- select the asymmetric key to drop in the DB Explorer tree;
- right-click the object and select the **Drop Asymmetric Key <asymmetric_key_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new asymmetric key;
Ctrl+O to edit the selected asymmetric key;
Shift+Del to drop the object from the database.

5.5.7.1 Asymmetric Key Editor

Asymmetric Key Editor allows you to define asymmetric key properties. It opens automatically when you create a new asymmetric key and is available on editing an existing one (see Create asymmetric key and Edit asymmetric key for details).

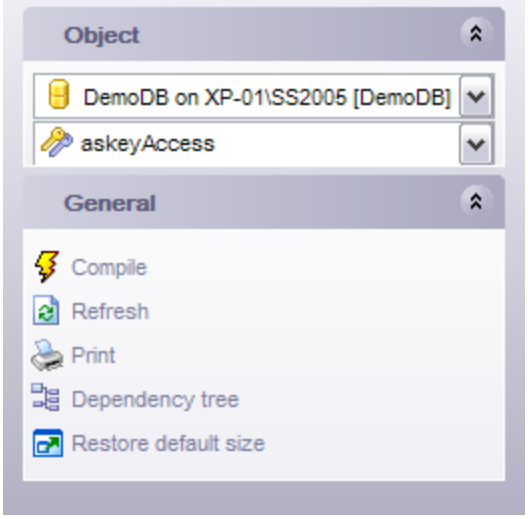
To open an asymmetric key in **Asymmetric Key Editor**, double-click it in the DB Explorer tree.
- Using Navigation bar
- Editing asymmetric key definition
- Browsing object dependencies
- Viewing DDL definition
- Setting object permissions

5.5.7.1.1 Using Navigation bar

The **Navigation bar** of **Asymmetric Key Editor** allows you to:
- select the database where the object is located;
- select the asymmetric key for editing;
- compile the object (if it is being created/modified);
- refresh the content of the active tab;
- print metadata of the object;

© 2008 EMS Database Management Solutions

- view the dependency tree for the object;
- restore the default size of the editor window.



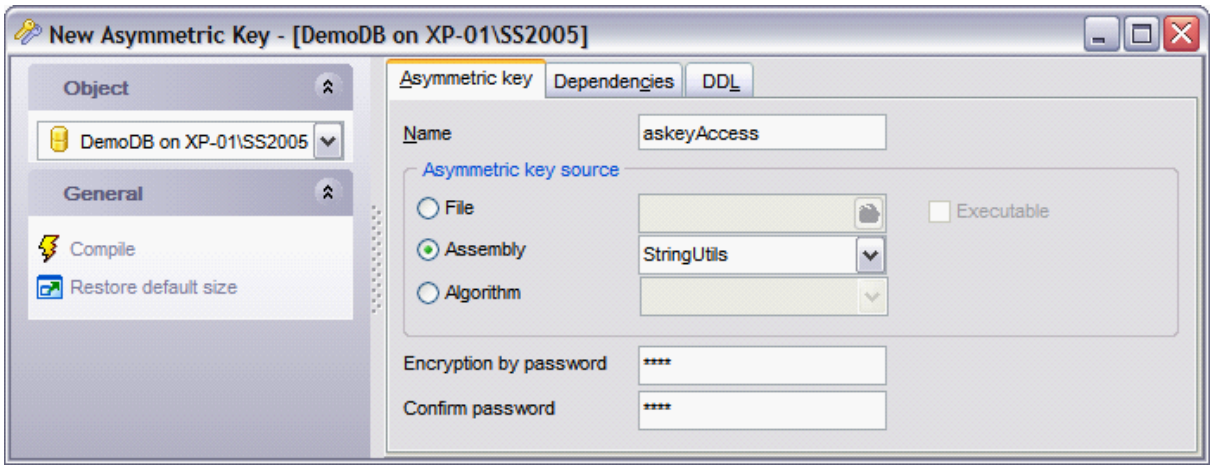
Depending on the current tab selection, the **Navigation bar** expands to one or more additional panes with tab-specific actions that can be useful for working with the object.

<< Back to Asymmetric Key Editor

5.5.7.1.2 Editing asymmetric key definition

Use the **Asymmetric key** tab of **Asymmetric Key Editor** to generate/edit an asymmetric key and specify its properties.

Name
Enter a name for the new asymmetric key, or view the name of the asymmetric key being edited.



© 2008 EMS Database Management Solutions

Asymmetric key source

File
If this option is selected, you should specify the path to a strong-name file to be used to load the key pair from. Use the button to specify the complete path and the name of the strong-name file.

Executable
If this option is selected, the file is a DLL used by the asymmetric key.

Assembly
If this option is selected, use the drop-down list to specify an assembly to be used to load the public key from.

Algorithm
If this option is selected, use the drop-down list to specify an algorithm to be used by the asymmetric key (the *WITH ALGORITHM* clause). Possible values are: *RSA_512*, *RSA_1024*, *RSA_2048*.

Encryption by password / Confirm password
Specify the password that will be used to encrypt the private key, and confirm the password in the corresponding box.

5.5.8 Certificates

In SQL Server 2005, data can be encrypted with a hierarchical encryption and key management infrastructure. Each layer encrypts the layer below it by using a combination of the following mechanisms for encryption: symmetric keys, asymmetric keys, **certificates**.

A **certificate** is a digitally-signed statement that binds the value of a public key to the identity of the person, device or service that holds the corresponding private key. Certificates are database-level securable objects that follow the X.509 standard and support X.509 V1 fields. A certificate can be loaded from a file or assembly upon its creation.

Creating Certificates

To create a new certificate:
- select the **Database | New Object...** main menu item;
- select **Certificate** in the Create New Object dialog;
- define certificate properties using the appropriate tabs of Certificate Editor.

Hint: To create a new certificate, you can also right-click the **Certificates** node or any object within this node in the DB Explorer tree and select the **New Certificate** item from the context menu.

To create a new certificate with the same properties as one of existing certificates has:

© 2008 EMS Database Management Solutions

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

Alternatively, you can right-click a certificate in the DB Explorer tree and select the **Duplicate Certificate <certificate_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new certificate in, and to edit the result SQL statement for creating the certificate.

Editing Certificates

To edit an existing certificate:

- select the certificate for editing in the DB Explorer tree (type the first letters of the certificate name for quick search);
- right-click the object and select the **Edit Certificate <certificate_name>** context menu item, or simply double-click the certificate;
- edit certificate properties using the appropriate tabs of Certificate Editor.

Dropping Certificates

To drop a certificate:

- select the certificate to drop in the DB Explorer tree;
- right-click the object and select the **Drop Certificate <certificate_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new certificate;
Ctrl+O to edit the selected certificate;
Shift+Del to drop the object from the database.

5.5.8.1 Certificate Editor

Certificate Editor allows you to define certificate properties. It opens automatically when you create a new certificate and is available on editing an existing one (see Create certificate and Edit certificate for details).

To open a certificate in **Certificate Editor**, double-click it in the DB Explorer tree.

- Using Navigation bar
- Editing certificate definition
- Browsing object dependencies
- Viewing DDL definition
- Setting object permissions

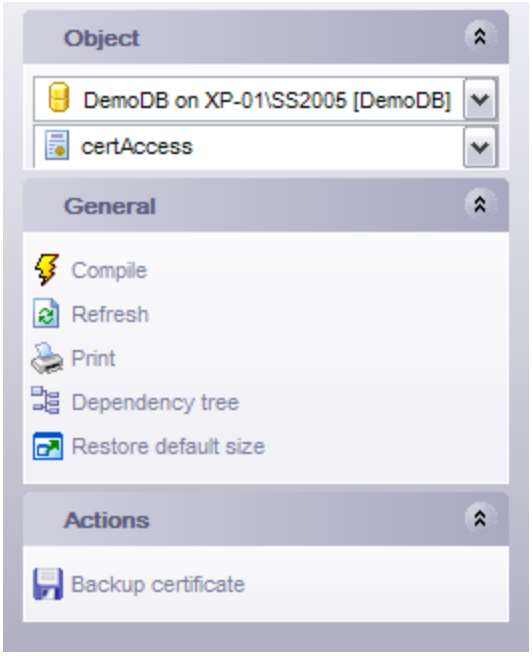
5.5.8.1.1 Using Navigation bar

The **Navigation bar** of **Certificate Editor** allows you to:

- select the database where the object is located;
- select the certificate for editing;
- compile the object (if it is being created/modified);
- refresh the content of the active tab;

© 2008 EMS Database Management Solutions

- print metadata of the object;
- view the dependency tree for the object;
- restore the default size of the editor window.



Depending on the current tab selection, the **Navigation bar** expands to one or more additional panes with tab-specific actions that can be useful for working with the object.

[<< Back to Certificate Editor]

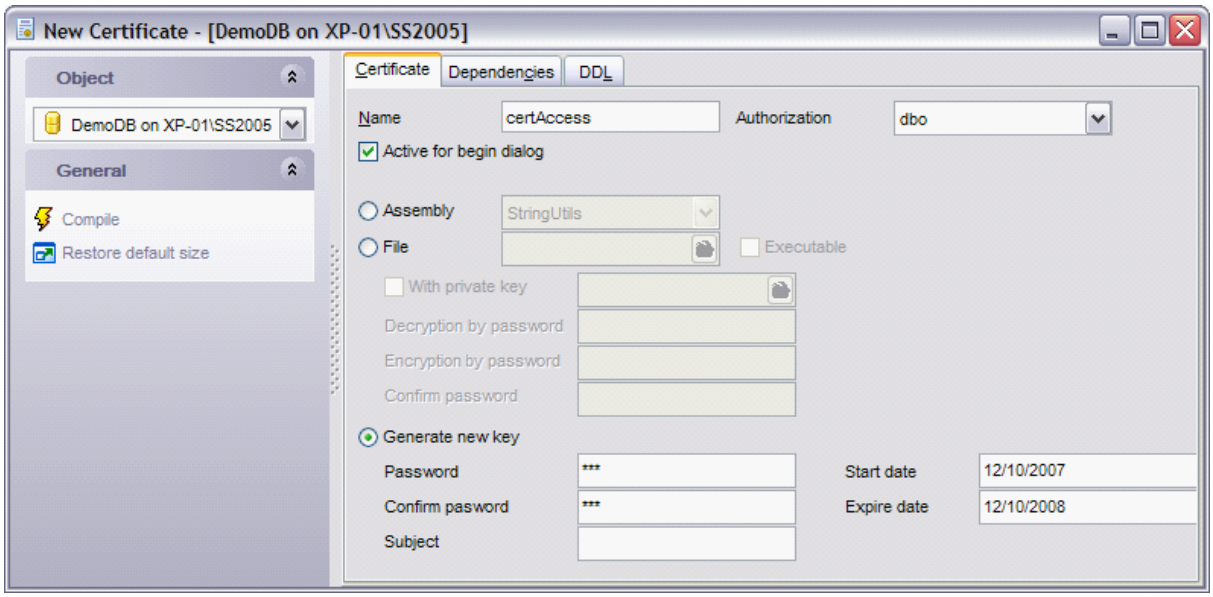
5.5.8.1.2 Editing certificate definition

Use the **Certificate** tab of **Certificate Editor** to generate/edit a certificate and specify its properties.

Name
Enter a name for the new certificate, or view the name of the certificate being edited.

Authorization
Use the drop-down list to select the name of the user that will own this certificate, or view the user selected upon the certificate creation.

© 2008 EMS Database Management Solutions



Active for begin dialog
This option makes the certificate available to the initiator of a Service Broker dialog conversation.

Assembly
If this option is selected, use the drop-down list to specify a signed assembly that has been already loaded into the database.

File
If this option is selected, you should specify the complete path and the name of a DER-encoded file that contains the certificate.

Executable
If this option is used, the file is a DLL that has been signed by the certificate.

With private key
Check this option to specify that the private key of the certificate is loaded into SQL Server. This option is only available when the certificate is being created from a file.

Use the button to specify the complete path and the name of the private key.

Decryption by password
Specify the password required to decrypt a private key that is retrieved from a file.

Encryption by password / Confirm password
Specify the password that will be used to encrypt the private key, and confirm the password in the corresponding box.

Generate new key
If this option is selected, a new key will be generated for the certificate. In this case you

© 2008 EMS Database Management Solutions

should specify the **password** for the new key, and the **subject** (a field in the metadata of the certificate as defined in the X.509 standard).

Start date
Set the date when the certificate becomes valid.

Expire date
Set the date when the certificate expires.

5.5.9 Message Types

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.

Creating Message Types

To create a new message type:
- select the **Database | New Object...** main menu item;
- select **Message Type** in the Create New Object dialog;
- define message type properties using the appropriate tabs of Message Type Editor.

Hint: To create a new message type, you can also right-click the **Message Types** node or any object within this node in the DB Explorer tree and select the **New Message Type** item from the context menu.

To create a new message type with the same properties as one of existing message types has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a message type in the DB Explorer tree and select the **Duplicate Message Type<message type_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new message type in, and to edit the result SQL statement for creating the message type.

Editing Message Types

To edit an existing message type:
- select the message type for editing in the DB Explorer tree (type the first letters of the message type name for quick search);
- right-click the object and select the **Edit Message Type <message type_name>** context menu item, or simply double-click the message type;
- edit message type properties using the appropriate tabs of Message Type Editor.

Dropping Message Types

To drop a message type:
- select the message type to drop in the DB Explorer tree;
- right-click the object and select the **Drop Message Type <message**

© 2008 EMS Database Management Solutions

type_name>... context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new message type;
Ctrl+O to edit the selected message type;
Shift+Del to drop the object from the database.

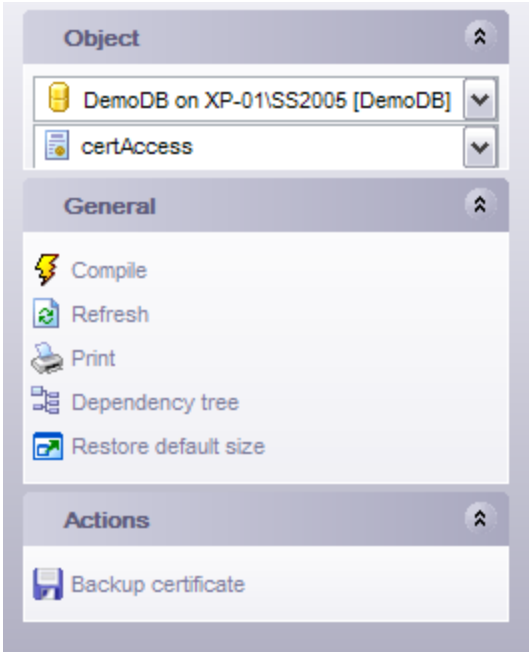
5.5.9.1 Message Type Editor

Message Type Editor allows you to define message type properties. It opens automatically when you create a new message type and is available on editing an existing one (see Create message type and Edit message type for details).

To open a message type in **Message Type Editor**, double-click it in the DB Explorer tree.
- Using Navigation bar
- Editing message type definition
- Browsing object dependencies
- Viewing DDL definition
- Setting object permissions

5.5.9.1.1 Using Navigation bar

The **Navigation bar** of **Message Type Editor** allows you to:
- select the database where the object is located;
- select the message type for editing;
- compile the object (if it is being created/modified);
- refresh the content of the active tab;
- print metadata of the object;
- view the dependency tree for the object;
- restore the default size of the editor window.

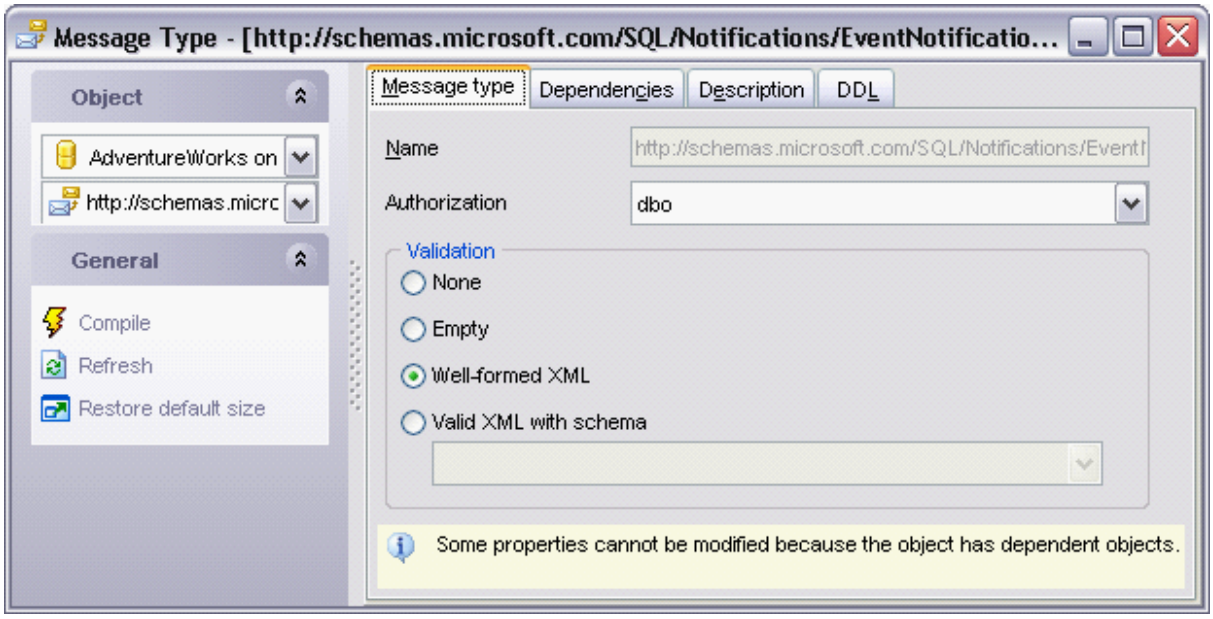


© 2008 EMS Database Management Solutions

Depending on the current tab selection, the **Navigation bar** expands to one or more additional panes with tab-specific actions that can be useful for working with the object.

5.5.9.1.2 Editing Message Type definition

Use the **Message Type** tab of **Message Type Editor** to generate/edit a certificate and specify its properties.



Name
Is the name of the message type to create. A new message type is created in the current database and owned by the principal specified in the **Authorization** field.

Authorization
Sets the owner of the message type to the specified database user or role. When the current user is dbo or sa, **Authorization** can be the name of any valid user or role. Otherwise, **Authorization** must be the name of the current user, the name of a user who the current user has IMPERSONATE permission for, or the name of a role to which the current user belongs.

Validation
Specifies how Service Broker validates the message body for messages of this type.

None
Specifies that no validation is performed. The message body can contain data, or it can be NULL.

Empty
Specifies that the message body must be NULL.

Well-formed XML
Specifies that the message body must contain well-formed XML.

Valid XML with schema

© 2008 EMS Database Management Solutions

Specifies that the message body must contain XML that complies with a schema in the specified schema collection (must be the name of an existing XML schema collection).

5.5.10 Contracts

Contracts define the name of a specific business task and list the message types used in that task. Service Broker contracts define two different service roles: the initiator and the target. The initiator of a conversation begins the conversation by sending a message to the target. The contract that the conversation uses defines which service role can send messages of a given message type.

Creating Contracts

To create a new contract:
- select the **Database | New Object...** main menu item;
- select **Contract** in the Create New Object dialog;
- define contract properties using the appropriate tabs of Contract Editor.

Hint: To create a new contract, you can also right-click the **Contracts** node or any object within this node in the DB Explorer tree and select the **New Contract** item from the context menu.

To create a new contract with the same properties as one of existing contracts has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a contract in the DB Explorer tree and select the **Duplicate Contract<contract_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new contract in, and to edit the result SQL statement for creating the contract.

Editing Contracts

To edit an existing contract:
- select the contract for editing in the DB Explorer tree (type the first letters of the contract name for quick search);
- right-click the object and select the **Edit Contract <contract_name>** context menu item, or simply double-click the contract;
- edit contract properties using the appropriate tabs of Contract Editor.

Dropping Contracts

To drop a contract:
- select the contract to drop in the DB Explorer tree;
- right-click the object and select the **Drop Contract <contract_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new contract;
Ctrl+O to edit the selected contract;
Shift+Del to drop the object from the database.

© 2008 EMS Database Management Solutions

5.5.10.1 Contract Editor

Contract Editor allows you to define contract properties. It opens automatically when you create a new contract and is available on editing an existing one (see Create contract and Edit contract for details).

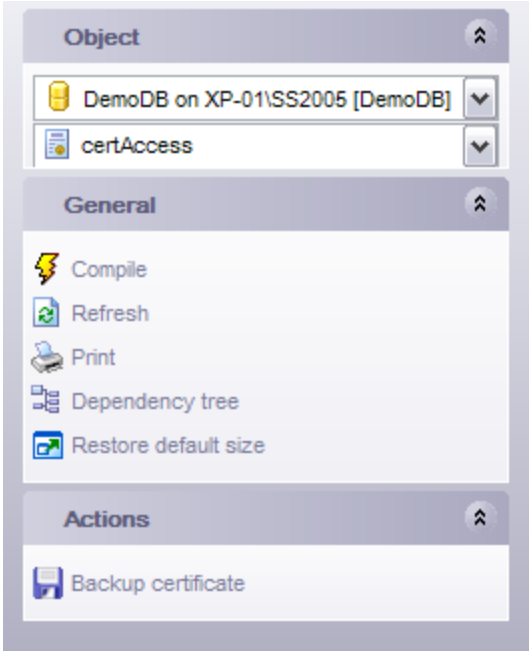
To open a contract in **Contract Editor**, double-click it in the DB Explorer tree.

- Using Navigation bar
- Editing contract definition
- Browsing object dependencies
- Viewing DDL definition
- Setting object permissions

5.5.10.1.1 Using Navigation bar

The **Navigation bar** of **Contract Editor** allows you to:

- select the database where the object is located;
- select the contract for editing;
- compile the object (if it is being created/modified);
- refresh the content of the active tab;
- print metadata of the object;
- view the dependency tree for the object;
- restore the default size of the editor window.

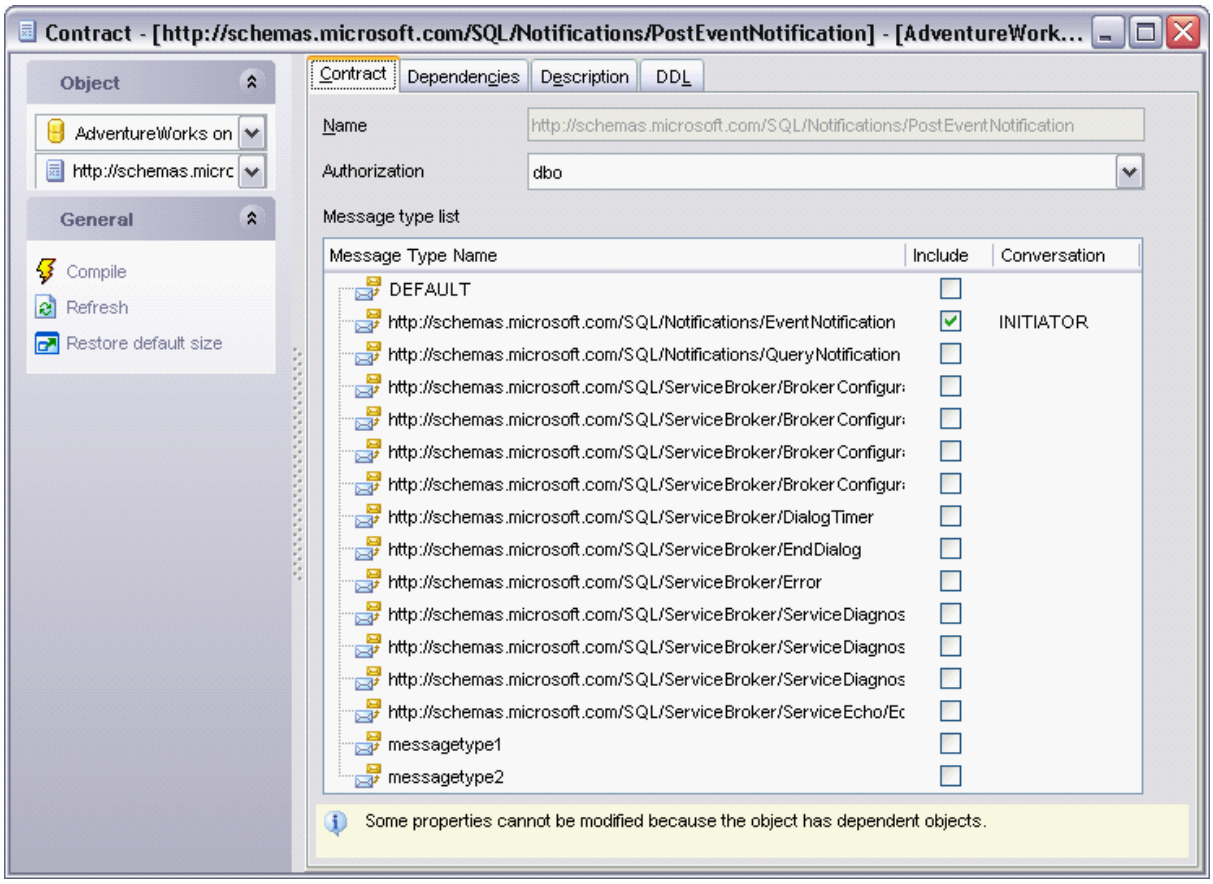


Depending on the current tab selection, the **Navigation bar** expands to one or more additional panes with tab-specific actions that can be useful for working with the object.

5.5.10.1.2 Editing Contract definition

Use the **Contract** tab of **Contract Editor** to generate/edit a certificate and specify its properties.

© 2008 EMS Database Management Solutions



Name
Is the name of the contract to create. A new contract is created in the current database and owned by the principal specified in the **Authorization** field.

Authorization
Sets the owner of the contract to the specified database user or role. When the current user is dbo or sa, **Authorization** can be the name of any valid user or role. Otherwise, **Authorization** 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.

Message type list
Is the list of a message type to be included as part of the contract.

Conversation field
Specifies which endpoint can send a message of the indicated message type. Contracts document the messages that services can use to have specific conversations. Each conversation has two endpoints: the initiator endpoint, the service that started the conversation, and the target endpoint, the service that the initiator is contacting.

INITIATOR
Indicates that only the initiator of the conversation can send messages of the specified message type. A service that starts a conversation is referred to as the initiator of the conversation.

© 2008 EMS Database Management Solutions

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.

ANY
Indicates that messages of this type can be sent by both the initiator and the target.

5.5.11 Services

A Service Broker **service** is a name for a specific task or set of tasks. Service Broker uses the name of the service to route messages, deliver messages to the correct queue within a database, and enforce the contract for a conversation.

Creating Services

To create a new service:
- select the **Database | New Object...** main menu item;
- select **Service** in the Create New Object dialog;
- define service properties using the appropriate tabs of Service Editor.

Hint: To create a new service, you can also right-click the **Services** node or any object within this node in the DB Explorer tree and select the **New Service** item from the context menu.

To create a new service with the same properties as one of existing services has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a service in the DB Explorer tree and select the **Duplicate Service<service_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new service in, and to edit the result SQL statement for creating the service.

Editing Services

To edit an existing service:
- select the service for editing in the DB Explorer tree (type the first letters of the service name for quick search);
- right-click the object and select the **Edit Service <service_name>** context menu item, or simply double-click the service;
- edit service properties using the appropriate tabs of Service Editor.

Dropping Services

To drop a service:
- select the service to drop in the DB Explorer tree;
- right-click the object and select the **Drop Service <service_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:

© 2008 EMS Database Management Solutions

Ctrl+N to create a new service;
Ctrl+O to edit the selected service;
Shift+Del to drop the object from the database.

5.5.11.1 Service Editor

Service Editor allows you to define service properties. It opens automatically when you create a new service and is available on editing an existing one (see Create service and Edit service for details).

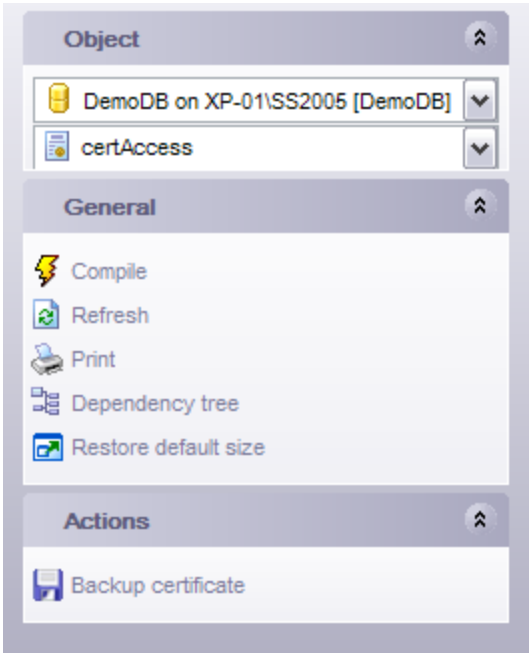
To open a service in **Service Editor**, double-click it in the DB Explorer tree.

- Using Navigation bar
- Editing service definition
- Browsing object dependencies
- Viewing DDL definition
- Setting object permissions

5.5.11.1.1 Using Navigation bar

The **Navigation bar** of **Service Editor** allows you to:

- select the database where the object is located;
- select the service for editing;
- compile the object (if it is being created/modified);
- refresh the content of the active tab;
- print metadata of the object;
- view the dependency tree for the object;
- restore the default size of the editor window.

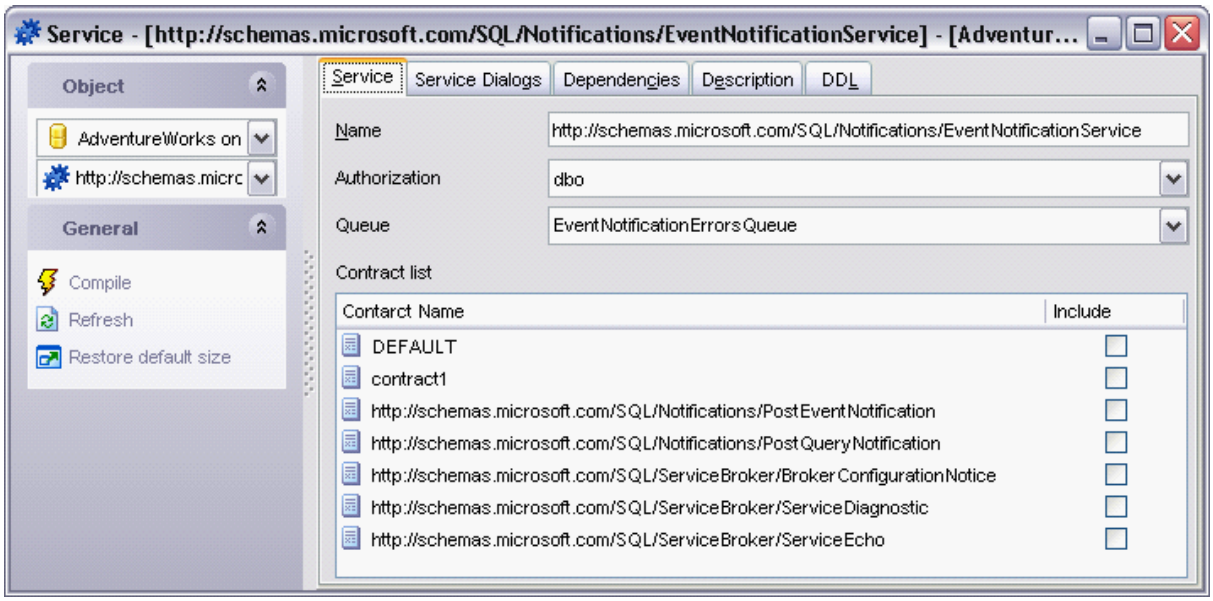


Depending on the current tab selection, the **Navigation bar** expands to one or more additional panes with tab-specific actions that can be useful for working with the object.

© 2008 EMS Database Management Solutions

5.5.11.1.2 Editing Service definition

Use the **Service** tab of **Service Editor** to generate/edit a certificate and specify its properties.



Name
Is the name of the service to create. A new contract is created in the current database and owned by the principal specified in the **Authorization** field.

Authorization
Sets the owner of the service to the specified database user or role. When the current user is dbo or sa, **Authorization** can be the name of any valid user or role. Otherwise, **Authorization** 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.

Contract list
Specifies a contract 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.

5.5.12 Routes

Rout is a non-schema object which can be added to the routing table for the current database. For outgoing messages, Service Broker determines routing by checking the routing table in the local database. For messages on conversations that originate in another instance, including messages to be forwarded, Service Broker checks the routes in msdb.

© 2008 EMS Database Management Solutions

Creating Routes

To create a new route:
- select the **Database | New Object...** main menu item;
- select **Route** in the Create New Object dialog;
- define route properties using the appropriate tabs of Route Editor.

Hint: To create a new route, you can also right-click the **Routes** node or any object within this node in the DB Explorer tree and select the **New Route** item from the context menu.

To create a new route with the same properties as one of existing routes has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a route in the DB Explorer tree and select the **Duplicate Route <route_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new route in, and to edit the result SQL statement for creating the route.

Editing Routes

To edit an existing route:
- select the route for editing in the DB Explorer tree (type the first letters of the route name for quick search);
- right-click the object and select the **Edit Route <route_name>** context menu item, or simply double-click the route;
- edit route properties using the appropriate tabs of Route Editor.

Dropping Routes

To drop a route:
- select the route to drop in the DB Explorer tree;
- right-click the object and select the **Drop Route <route_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new route;
Ctrl+O to edit the selected route;
Shift+Del to drop the object from the database.

5.5.12.1 Route Editor

Route Editor allows you to define route properties. It opens automatically when you create a new route and is available on editing an existing one (see Create route and Edit route for details).

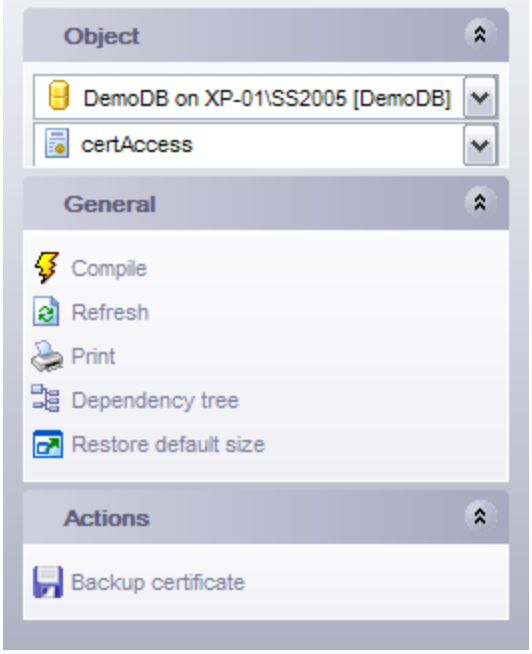
To open a route in **Route Editor**, double-click it in the DB Explorer tree.
- Using Navigation bar
- Editing route definition
- Browsing object dependencies
- Viewing DDL definition
- Setting object permissions

© 2008 EMS Database Management Solutions

5.5.12.1.1 Using Navigation bar

The **Navigation bar** of **Route Editor** allows you to:

- select the database where the object is located;
- select the route for editing;
- compile the object (if it is being created/modified);
- refresh the content of the active tab;
- print metadata of the object;
- view the dependency tree for the object;
- restore the default size of the editor window.

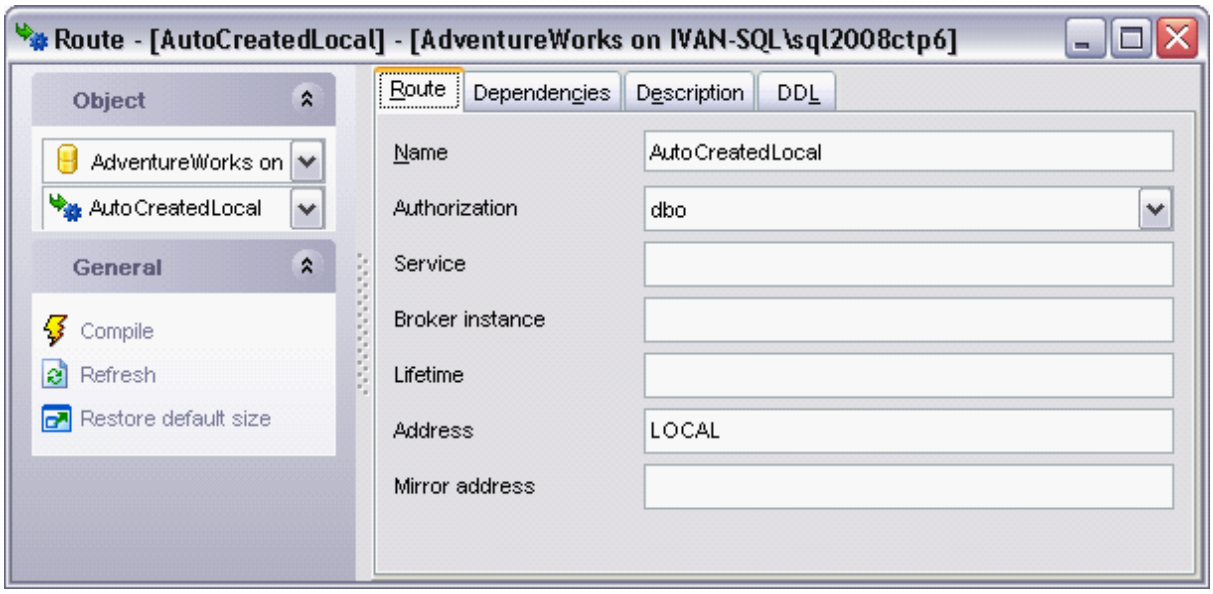


Depending on the current tab selection, the **Navigation bar** expands to one or more additional panes with tab-specific actions that can be useful for working with the object.

5.5.12.1.2 Editing Route definition

Use the **Route** tab of **Route Editor** to generate/edit a certificate and specify its properties.

© 2008 EMS Database Management Solutions



Name
Is the name of the route to create. A new route is created in the current database and owned by the principal specified in the **Authorization** field.

Authorization
Sets the owner of the route to the specified database user or role. When the current user is dbo or sa, **Authorization** can be the name of any valid user or role. Otherwise, **Authorization** 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.

Service
Specifies the name of the remote service that this route points to. It must exactly match the name the remote service uses.

Broker instance
Specifies the database that hosts the target service.

Lifetime
Specifies the time, in seconds, that SQL Server retains the route in the routing table. At the end of the lifetime, the route expires, and SQL Server no longer considers the route when choosing a route for a new conversation.

Address
Specifies the network address for this route.

Mirror address
Specifies the network address for a mirrored database.

5.5.13 Remote Service Bindings

Creating Remote Service Bindings

To create a new remote service binding:

© 2008 EMS Database Management Solutions

- select the **Database | New Object...** main menu item;
- select **Remote Service Binding** in the Create New Object dialog;
- define remote service binding properties using the appropriate tabs of Remote Service Binding Editor.

Hint: To create a new remote service binding, you can also right-click the **Remote Service Bindings** node or any object within this node in the DB Explorer tree and select the **New Remote Service Binding** item from the context menu.

To create a new remote service binding with the same properties as one of existing remote service bindings has:

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

Alternatively, you can right-click a remote service binding in the DB Explorer tree and select the **Duplicate Remote Service Binding<remote service binding_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new remote service binding in, and to edit the result SQL statement for creating the remote service binding.

Editing Remote Service Bindings

To edit an existing remote service binding:

- select the remote service binding for editing in the DB Explorer tree (type the first letters of the remote service binding name for quick search);
- right-click the object and select the **Edit Remote Service Binding <remote service binding_name>** context menu item, or simply double-click the remote service binding;
- edit remote service binding properties using the appropriate tabs of Remote Service Binding Editor.

Dropping Remote Service Bindings

To drop a remote service binding:

- select the remote service binding to drop in the DB Explorer tree;
- right-click the object and select the **Drop Remote Service Binding <remote service binding_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new remote service binding;
Ctrl+O to edit the selected remote service binding;
Shift+Del to drop the object from the database.

5.5.13.1 Remote Service Binding Editor

Remote Service Binding Editor allows you to define remote service binding properties. It opens automatically when you create a new remote service binding and is available on editing an existing one (see Create remote service binding and Edit remote service binding for details).

To open a remote service binding in **Remote Service Binding Editor**, double-click it in the DB Explorer tree.

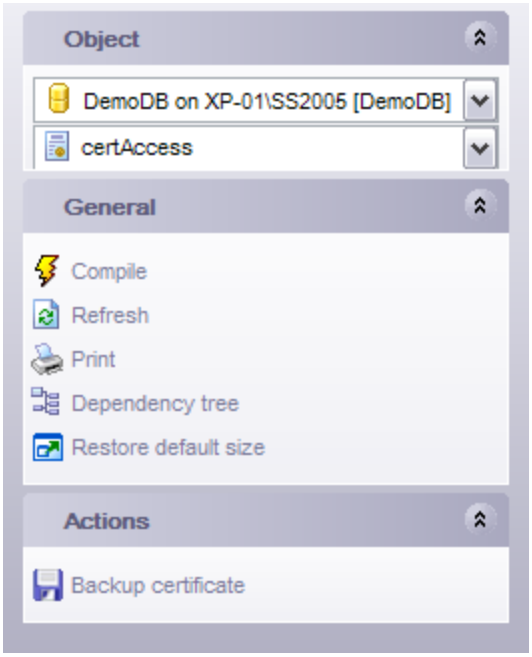
© 2008 EMS Database Management Solutions

- Using Navigation bar
- Editing remote service binding definition
- Browsing object dependencies
- Viewing DDL definition
- Setting object permissions

5.5.13.1.1 Using Navigation bar

The **Navigation bar** of **Remote Service Binding Editor** allows you to:

- select the database where the object is located;
- select the remote service binding for editing;
- compile the object (if it is being created/modified);
- refresh the content of the active tab;
- print metadata of the object;
- view the dependency tree for the object;
- restore the default size of the editor window.

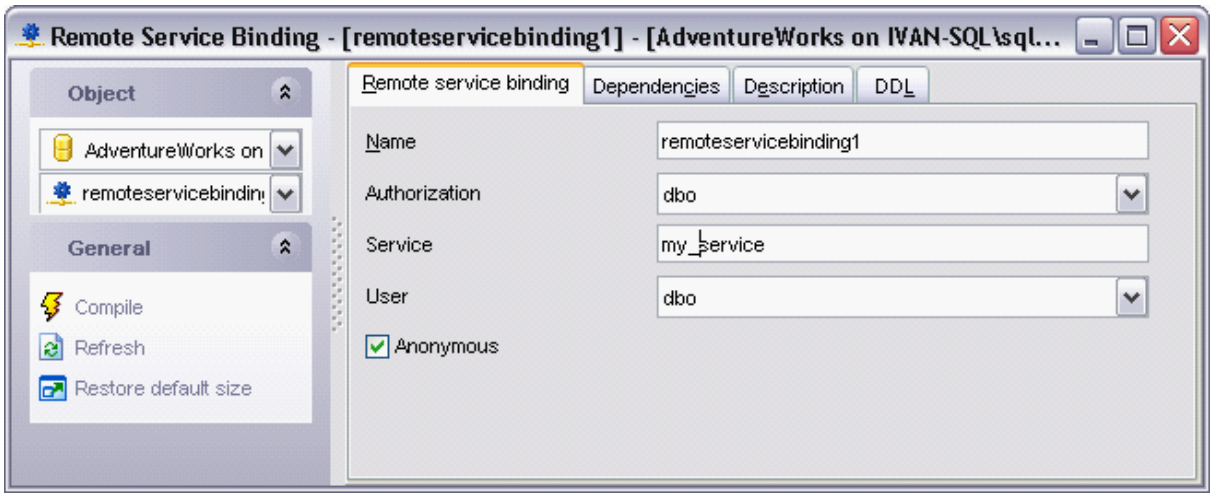


Depending on the current tab selection, the **Navigation bar** expands to one or more additional panes with tab-specific actions that can be useful for working with the object.

5.5.13.1.2 Editing Remote Service Binding definition

Use the **Remote Service Binding** tab of **Remote Service Binding Editor** to generate/edit a certificate and specify its properties.

© 2008 EMS Database Management Solutions



Name
Is the name of the remote service binding to create. A new remote service binding is created in the current database and owned by the principal specified in the **Authorization** field.

Authorization
Sets the owner of the remote service binding to the specified database user or role. When the current user is dbo or sa, **Authorization** can be the name of any valid user or role. Otherwise, **Authorization** 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.

Service
Specifies the remote service to bind to the user identified in the **User** field.

User
Specifies the database principal that owns the certificate associated with the remote service set within the **Service** field. This certificate is used for encryption and authentication of messages exchanged with the remote service.

Anonymous
Specifies whether anonymous authentication is used when communicating with the remote service.

© 2008 EMS Database Management Solutions

5.6 Database-level principals

Database principals are individuals, groups, and processes that can request database resources.

SQL Manager 2008 for SQL Server provides management tools for the following database-level security principals:

- Users

Provides tools allowing you to manage database users.

- Roles

Provides tools allowing you to manage database roles.

See also:

New Object dialog
Duplicate Object Wizard
Schemas
Schema objects
Non-schema objects
Database-level principals

5.6.1 Users

A database **user** is a database-level security principal enabling object access permission control at the finest level of granularity. A user represents a single SQL Server login within the scope of the database in which the user is defined.

Creating Users

To create a new user:

- select the **Database | New Object...** main menu item;
- select **User** in the Create New Object dialog;
- define user properties using the appropriate tabs of User Editor.

Hint: To create a new user, you can also right-click the **Users** node or any object within this node in the DB Explorer tree and select the **New User** item from the context menu.

To create a new user with the same properties as one of existing users has:

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

Alternatively, you can right-click a user in the DB Explorer tree and select the **Duplicate User <user_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new user in, and to edit the result SQL statement for creating the user.

Editing Users

To edit an existing user:

© 2008 EMS Database Management Solutions

- select the user for editing in the DB Explorer tree (type the first letters of the user name for quick search);
- right-click the object and select the **Edit User <user_name>** context menu item, or simply double-click the user;
- edit user properties using the appropriate tabs of User Editor.

Dropping Users

To drop a user:
- select the user to drop in the DB Explorer tree;
- right-click the object and select the **Drop User <user_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new user;
Ctrl+O to edit the selected user;
Shift+Del to drop the object from the database.

5.6.1.1 User Editor

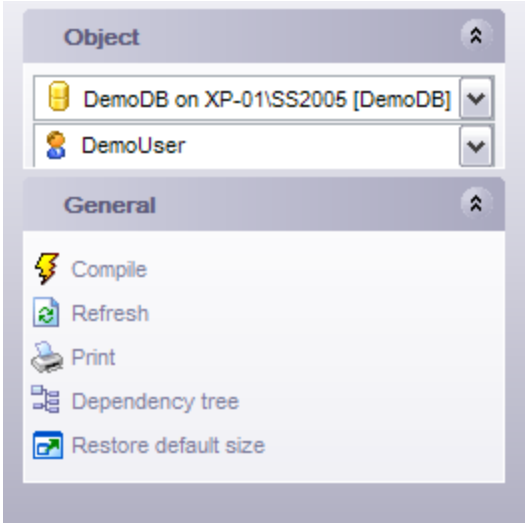
User Editor allows you to define user properties and membership. It opens automatically when you create a new user and is available on editing an existing one (see Create user and Edit user for details).

To open a user in **User Editor**, double-click it in the DB Explorer tree.
- Using Navigation bar
- Editing user definition
- Defining user membership
- Editing object description
- Viewing DDL definition

5.6.1.1.1 Using Navigation bar

The **Navigation bar** of **User Editor** allows you to:
- select the database where the object is located;
- select the user for editing;
- compile the object (if it is being created/modified);
- save the object description (if it has been modified);
- refresh the content of the active tab;
- print metadata of the object;
- view the dependency tree for the object;
- restore the default size of the editor window.

© 2008 EMS Database Management Solutions



Depending on the current tab selection, the **Navigation bar** expands to one or more additional panes with tab-specific actions that can be useful for working with the object.

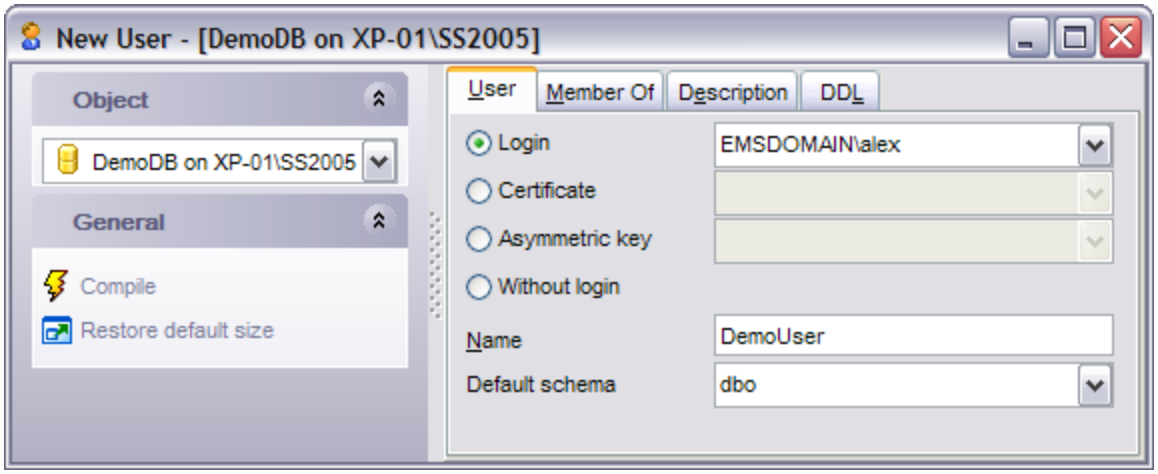
[<< Back to User Editor]

5.6.1.1.2 Editing user definition

Use the **User** tab of **User Editor** to create/edit a user and specify its properties.

Name
Specifies the name by which the user is identified inside this database.

Default schema
Use the drop-down list to specify the first schema that will be searched by the server when it resolves the names of objects for this database user.



Login

© 2008 EMS Database Management Solutions

If this option is selected, use the drop-down list to specify the SQL Server login for which the database user is being created.

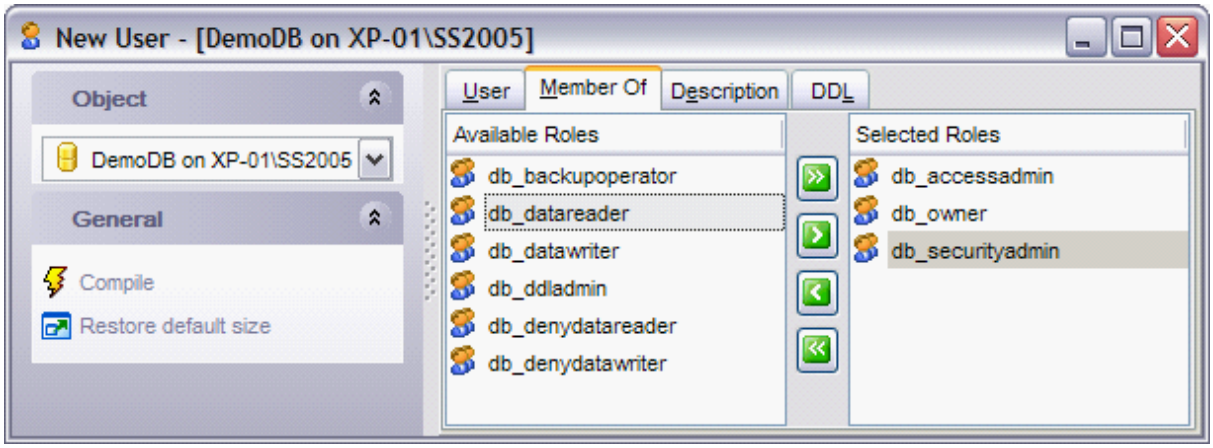
Certificate
If this option is selected, use the drop-down list to specify the certificate for which the database user is being created.

Asymmetric key
If this option is selected, use the drop-down list to specify the asymmetric key for which the database user is being created.

Without login
This option specifies that the user should not be mapped to any of existing logins.

5.6.1.1.3 Defining user membership

The **Member of** tab of **User Editor** allows you to define user membership, i.e. to select the built-in database role(s) the user will belong to. The selected roles determine the tasks that can be performed through the user.



To select a role, you need to move it from the **Available Roles** list to the **Selected Roles** list. Use the buttons or drag-and-drop operations to move the roles from one list to another.

5.6.2 Roles

All users interact with SQL Server within the context of a database-level principal - **role**. A user can belong to multiple groups and have multiple roles, and the operations that are permitted by each role determine the actions that the user can perform. It is necessary to create SQL Server database roles when a group of users needs to perform a specified set of activities in the database.

© 2008 EMS Database Management Solutions

Creating Roles

To create a new role:
- select the **Database | New Object...** main menu item;
- select **Role** in the Create New Object dialog;
- define role properties using the appropriate tabs of Role Editor.

Hint: To create a new role, you can also right-click the **Roles** node or any object within this node in the DB Explorer tree and select the **New Role** item from the context menu.

To create a new role with the same properties as one of existing roles has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a role in the DB Explorer tree and select the **Duplicate Role <role_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new role in, and to edit the result SQL statement for creating the role.

Editing Roles

To edit an existing role:
- select the role for editing in the DB Explorer tree (type the first letters of the role name for quick search);
- right-click the object and select the **Edit Role <role_name>** context menu item, or simply double-click the role;
- edit role properties using the appropriate tabs of Role Editor.

Dropping Roles

To drop a role:
- select the role to drop in the DB Explorer tree;
- right-click the object and select the **Drop Role <role_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new role;
Ctrl+O to edit the selected role;
Shift+Del to drop the object from the database.

5.6.2.1 Role Editor

Role Editor allows you to define role properties and membership. It opens automatically when you create a new role and is available on editing an existing one (see Create role and Edit role for details).

To open a role in **Role Editor**, double-click it in the DB Explorer tree.
- Using Navigation bar
- Editing role definition
- Defining role membership
- Viewing DDL definition

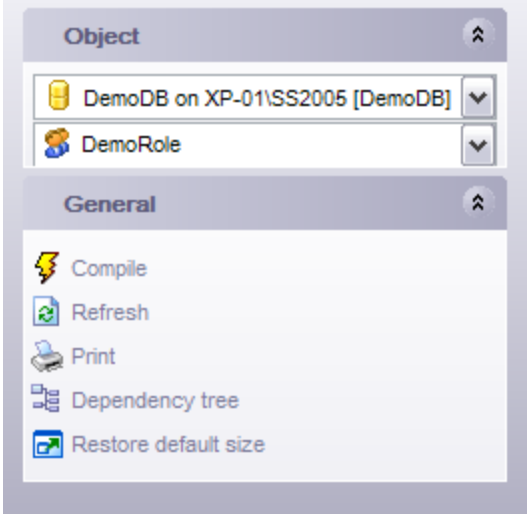
© 2008 EMS Database Management Solutions

- Setting object permissions

5.6.2.1.1 Using Navigation bar

The **Navigation bar** of **Role Editor** allows you to:

- select the database where the object is located;
- select the role for editing;
- compile the object (if it is being created/modified);
- refresh the content of the active tab;
- print metadata of the object;
- view the dependency tree for the object;
- restore the default size of the editor window.



Depending on the current tab selection, the **Navigation bar** expands to one or more additional panes with tab-specific actions that can be useful for working with the object.

<< Back to Role Editor

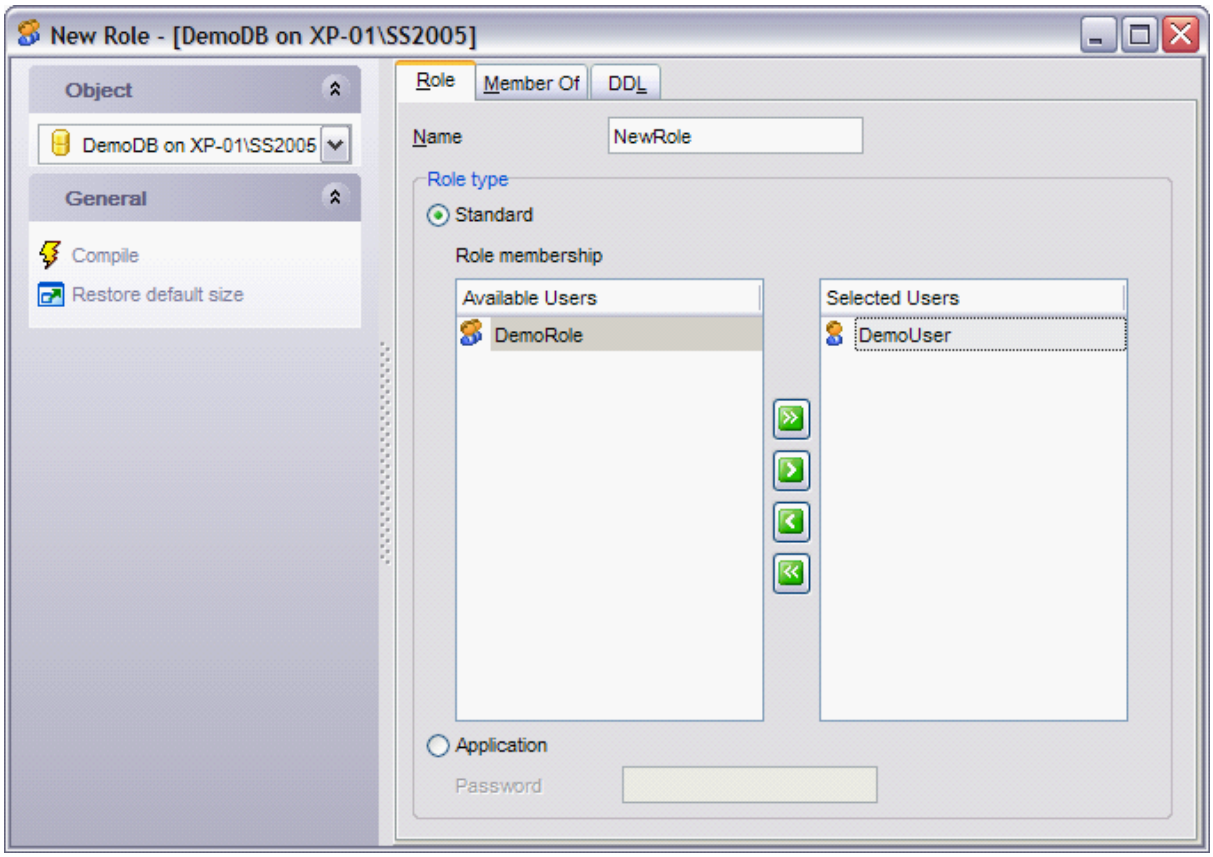
5.6.2.1.2 Editing role definition

Use the **Role** tab of **Role Editor** to create/edit a role and specify its properties and membership.

Name
Enter a name for the new role, or view the name of the role being edited.

Role type
Select whether the preferable role type:
- *Standard* (user-defined) role;
- *Application* role.

© 2008 EMS Database Management Solutions



If a **Standard** role is being created/edited, you should add members to the role, i.e. add the users (or other roles) that will inherit the permissions of the role.

To add a user, you need to move it from the **Available Users** list to the **Selected Users** list. Use the buttons or drag-and-drop operations to move the users from one list to another.

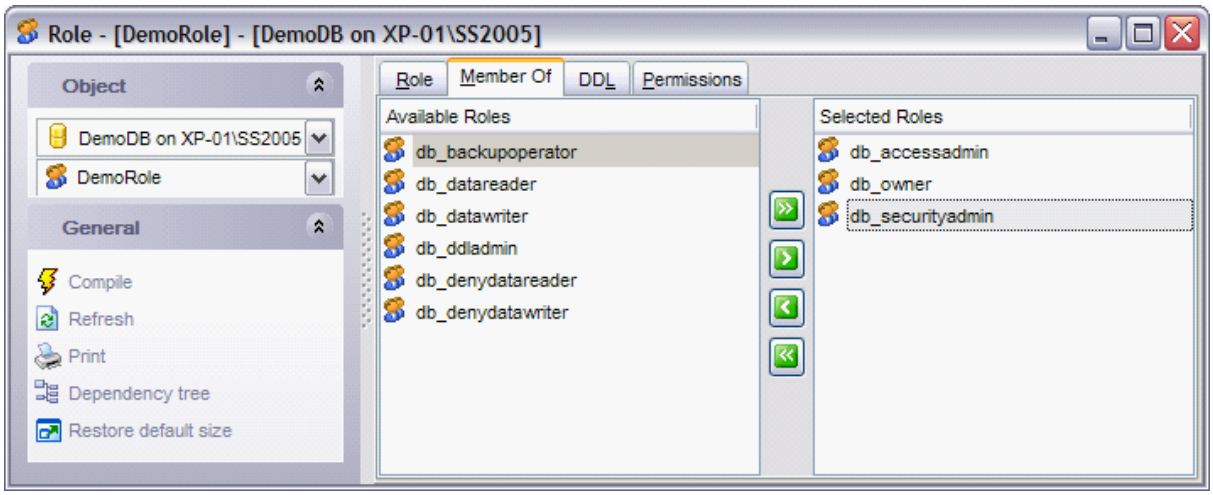
If an **Application** role is being created/edited, you should also specify a **password** for the role.

Note that in the Database explorer different icons are used for **Application role** and **User role**.

5.6.2.1.3 Defining role membership

The **Member of** tab of **Role Editor** allows you to define role membership, i.e. to select the built-in database role(s) the current role will be a member of. The selected roles determine the tasks that can be performed through the current role.

© 2008 EMS Database Management Solutions



To select a role, you need to move it from the **Available Roles** list to the **Selected Roles** list. Use the buttons or drag-and-drop operations to move the roles from one list to another.

© 2008 EMS Database Management Solutions

Part

6 Query management tools

When using **SQL Manager 2008 for SQL Server**, you are provided with two basic tools you may need to manage your SQL queries: SQL Editor for editing SQL query text directly and Visual Query Builder for building queries visually. Find the list of common SQL query management operations below.

Creating new queries

In order to create a new query in *SQL Editor*:

- select the **Tools | New SQL Editor** main menu item or use the corresponding toolbar button;
- click the **Add new query** item of the Navigation bar;
- edit the query text within the **Edit** tab of SQL Editor.

In order to create a new query in *Query Builder*:

- select the **Tools | New Query Builder** main menu item or use the corresponding toolbar button;
- build the query visually within the **Builder** tab of Visual Query Builder.

Editing queries

In order to open a query in *SQL Editor*:

- select the **Tools | Show SQL Editor** main menu item or use the corresponding toolbar button;
- use the numbered tabs at the bottom of the editor window to switch between previously edited queries. The last edited query is displayed automatically on opening the editor;
- edit the query text within the **Edit** tab of SQL Editor.

In order to open a query in *Query Builder*:

- select the **Tools | Show Query Builder** main menu item or use the corresponding toolbar button;
- the last edited query is displayed automatically on opening Query Builder;
- to load a previously saved diagram, click the **Load diagram** item of the Navigation bar ;
- to load a query from an **.sql* file, open the **Edit** tab and click the **Load SQL** button of the Navigation bar;
- edit the query visually within the **Builder** and/or the **Edit** tabs of Visual Query Builder.

In order to load a query from an **.sql* file:

- select the **Tools | New SQL Editor** main menu item or use the corresponding toolbar button;
- click the **Load from file** item of the Navigation bar;
- browse for the query file using the **Open SQL File** dialog;
- edit the query text within the **Edit** tab of SQL Editor.

Executing queries

In order to execute a query:

- *create a new query* or *open an existing one*;
- click the **Execute** item of the Navigation bar or use the *F9* hot-key to execute the

© 2008 EMS Database Management Solutions

query;
- view/edit the returned data within the **Results** tab of SQL Editor.

Saving queries

In order to save a query:
- *create a new query* or *open an existing one*;
- click the **Save to file** Navigation bar item (in *SQL Editor*) or the **Save SQL** Navigation bar item (in *Query Builder*), or use the *Ctrl+S* shortcut to save the query using the **Save as...** dialog;
- click the **Save diagram** Navigation bar item in Visual Query Builder to save the designed diagram;

or
- use the **Save all** Navigation bar item in SQL Editor if you need to save all the queries to one file.

See also:

Getting started
Database Explorer
Database management
Database objects management
Data management
Import/Export tools
Database tools
Server tools
Personalization
External tools

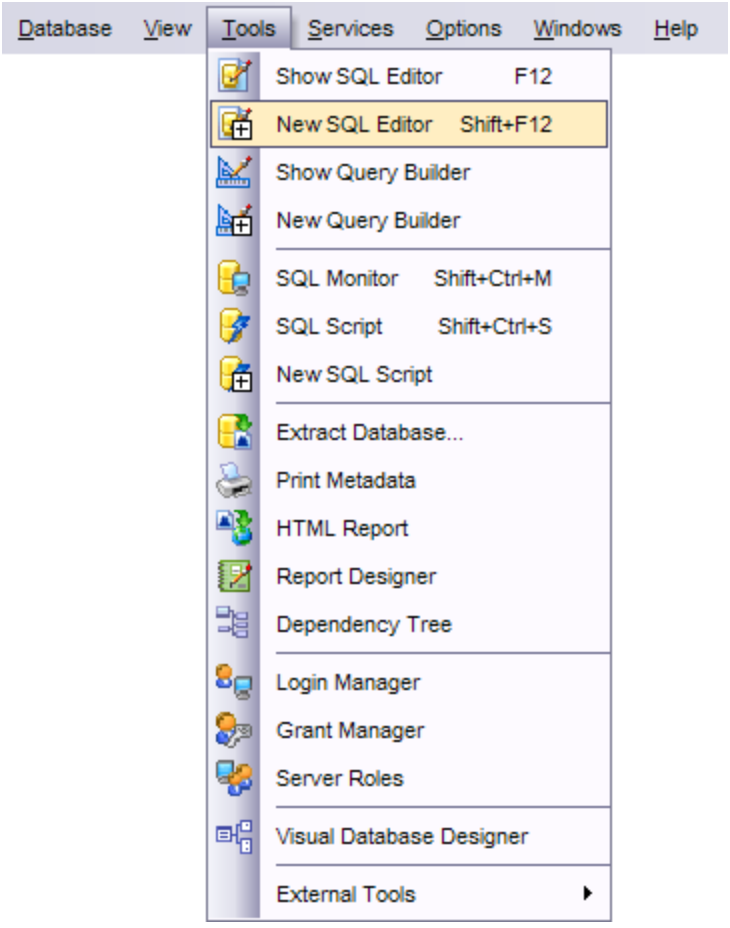
© 2008 EMS Database Management Solutions

6.1 SQL Editor

SQL Editor is the basic tool of SQL Manager 2008 for SQL Server for creating and executing queries. The tool allows you to create and edit the SQL text of a query, prepare and execute queries and view the results of query execution.

To open SQL Editor, select the **Tools | New SQL Editor** / **Tools | Show SQL Editor** main menu items or use the corresponding / toolbar buttons. You can also use the *Shift+F12* / *F12* shortcuts for the same purpose.

- Using Navigation bar
- Working with SQL Editor area
- Using the context menu
- Viewing query plan
- Using object links
- Executing queries and viewing results
- Viewing query logs
- Favorites editor
- Merging queries



Availability:

© 2008 EMS Database Management Solutions

Full version (for Windows)	**Yes**
Lite version (for Windows)	**Yes**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

Visual Query Builder

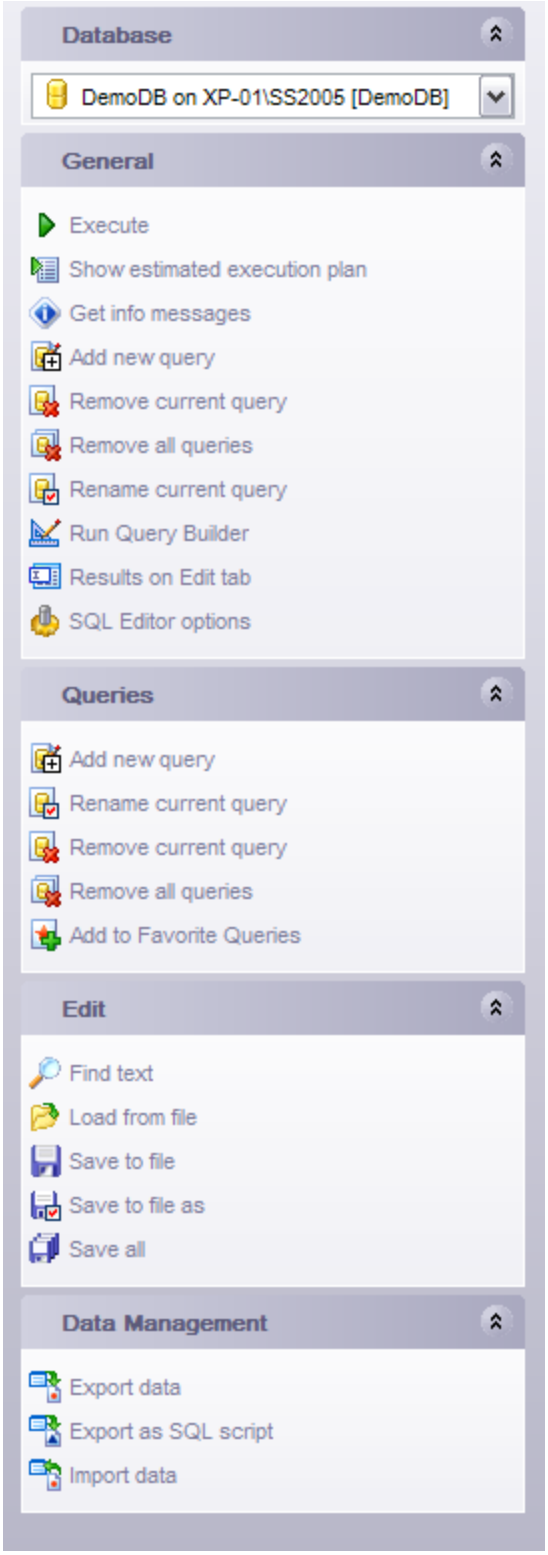
Query parameters

6.1.1 Using Navigation bar

The **Navigation bar** of **SQL Editor** allows you to:

- select a database for the query;
- execute the current query;
- view estimated query execution plan;
- show/hide information messages;
- add a new query (note that the current query text will NOT be lost);
- remove the current query;
- remove all queries from the editor;
- rename the query;
- run Visual Query Builder to design the query as a diagram;
- switch the results representation mode: *on Edit tab* or *on separate tab*;
- configure SQL Editor within the Tools | SQL Editor page of the Environment Options dialog;
- add the query to the Favorite Queries list using Favorites editor;
- activate the Find Text dialog;
- load a query from an **.sql* file using the **Open SQL File** dialog;
- save the query to an **.sql* file using the **Save as...** dialog;
- save all the queries to an **.sql* file;
- commit/rollback the transaction;
- export the returned dataset using Export Data Wizard;
- export data as SQL Script using the Export as SQL Script wizard;
- import data into the table using Import Data Wizard.

© 2008 EMS Database Management Solutions



See also:

Working with SQL Editor area

© 2008 EMS Database Management Solutions

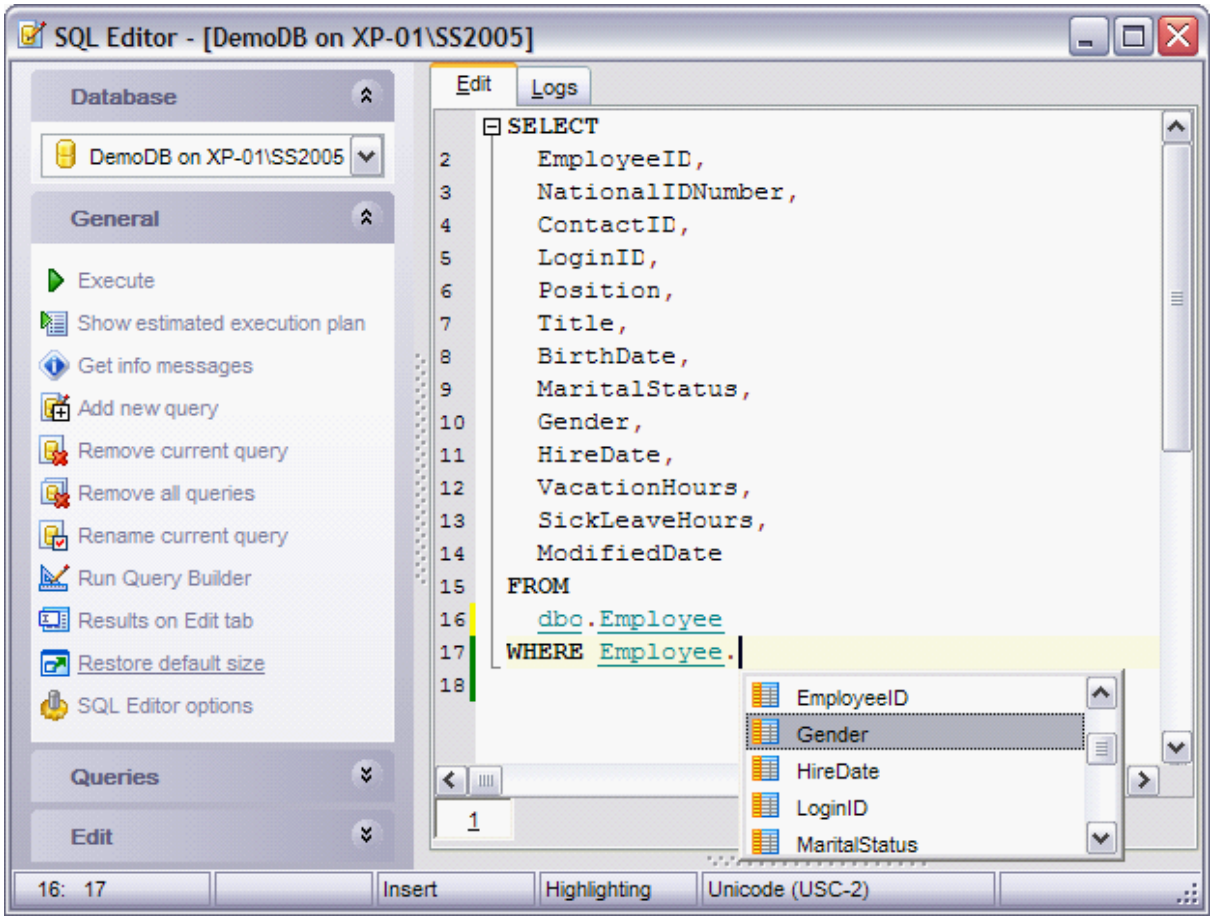
Using the context menu
Viewing query plan
Using object links
Executing queries
Viewing query logs
Favorites editor
Merging queries

6.1.2 Working with SQL Editor area

The **Editor area** of SQL Editor is available within the **Edit** tab and is provided for working with SQL queries in text mode.

For your convenience the **syntax highlight**, **code completion** and a number of other features for efficient SQL editing are implemented. If necessary, you can enable/disable or customize most of SQL Editor features using the Editor Options dialog.

The example of code completion is illustrated in the picture below. You can set the delay within the Quick code section of the Editor Options dialog or activate the completion list manually by pressing the *Ctrl+Space* shortcut.



© 2008 EMS Database Management Solutions

Hint: To paste input parameters of a procedure quickly, use the *Ctrl+Shift+P* shortcut after the procedure name.

Hint: To use a keyboard template, type the template name and press the *Ctrl+J* shortcut: the text associated with the template will be inserted automatically.

If necessary, you can **print** the SQL text of your query using the corresponding item of the context menu.

See also:
Using Navigation bar
Using the context menu
Viewing query plan
Using object links
Executing queries
Viewing query logs
Favorites editor
Merging queries

6.1.3 Using the context menu

The **context menu** of SQL Editor area contains execution commands, most of the standard text-processing functions (*Cut*, *Copy*, *Paste*, *Select All*) and functions for working with the query as a whole, e.g. you can *move the cursor to a particular line*, *change the case* of selected text, view the query *properties* or *print* the text of the query. Each of these operations can be also performed with the corresponding hot keys used.

Implementation of the Find Text and the Replace Text dialogs contributes to more efficient work with the SQL code.

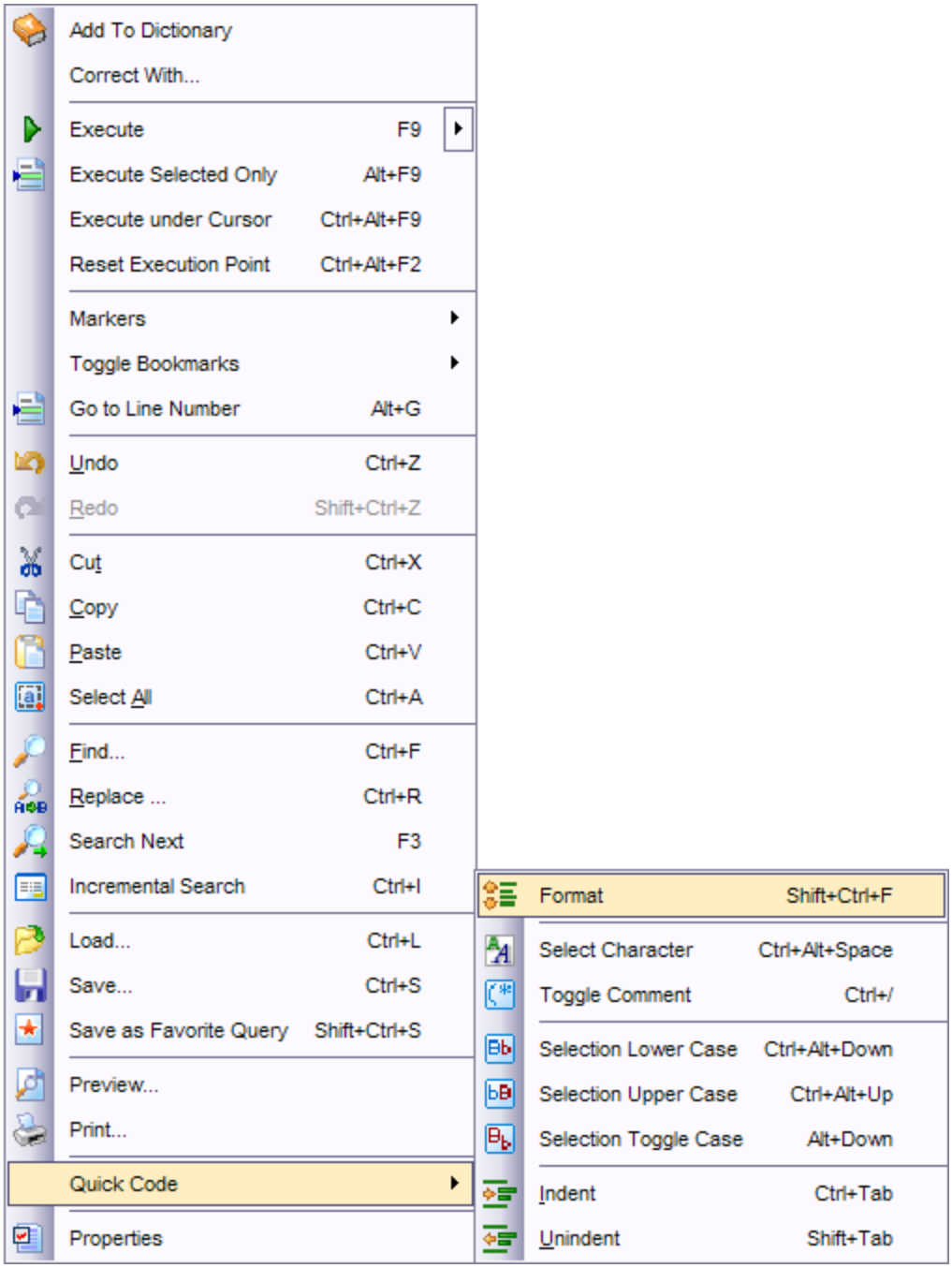
Find the complete list of **SQL Editor** context menu items below. The context menu allows you to:

- add the selected text to dictionary or correct text (see Spell checking for details);
- execute the query/selected text/text under cursor, and reset execution point (if necessary);
- manage markers: *Drop Marker*, *Collect Marker*, *Swap Marker*;
- toggle bookmarks allowing you to navigate through the query text and jump to a line with a particular number;
- perform editing operations: *Undo/Redo*, *Cut*, *Copy*, *Paste*, *Select all*;
- perform search and replace operations;
- save/load a query to/from an external **.sql* file;
- perform preview/print operations;
- use the *Quick code* group allowing you to format the selected code using *SQL*

© 2008 EMS Database Management Solutions

Formatter to make the code easier to read, toggle comments for code fragments, change case of the selected text, indent/unindent code lines;

- add the query to the Favorite Queries list;
- open the Editor Options dialog.



See also:

Using Navigation bar

Working with SQL Editor area

© 2008 EMS Database Management Solutions

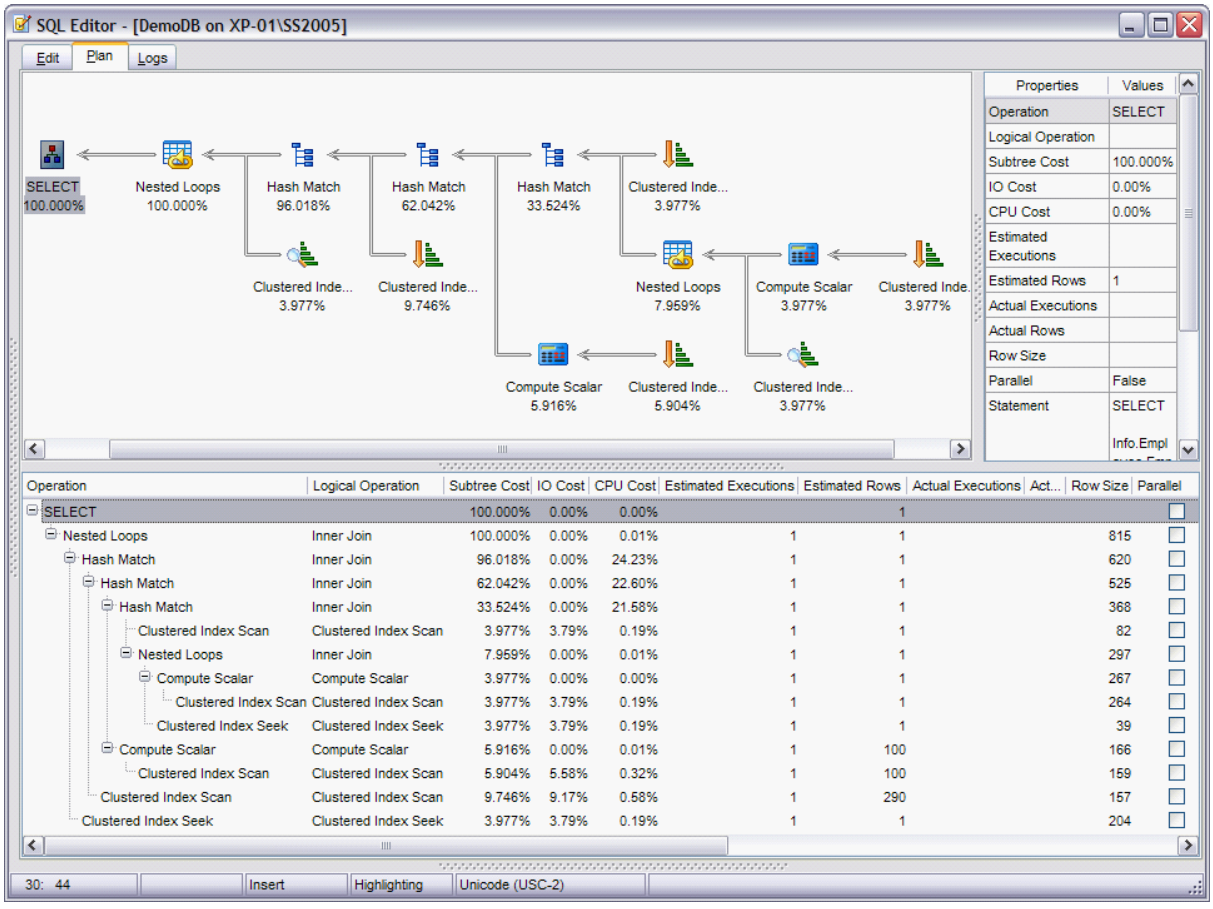
Viewing query plan
Using object links
Executing queries
Viewing query logs
Favorites editor
Merging queries

6.1.4 Viewing query plan

Using SQL Manager 2008 for SQL Server, you can view **the plan** for each of the queries created and executed in the application. The query plan is available within the corresponding **Plan** tab.

To view the **Plan** of a query, open the query in **SQL Editor** and use the **Show estimated execution plan** item of the Navigation bar.

The **Plan** tab allows you to view the sequence of actions performed by the database server in the process of the query execution, and the amount of system resources used for the query execution. Note that this tab only appears after you have executed the query.



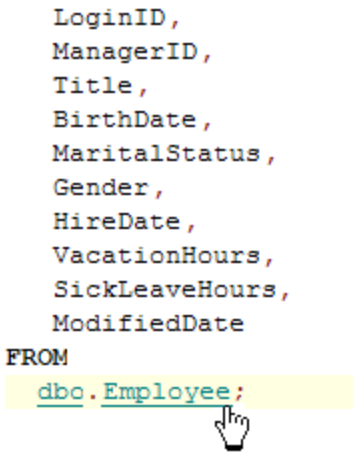
© 2008 EMS Database Management Solutions

If necessary, you can specify that the **Plan** tab appears automatically upon query execution in SQL Editor: select the **Show actual execution plan on query execution** option available within the Tools | SQL Editor section of the Environment Options dialog.

See also:
Using Navigation bar
Working with SQL Editor area
Using the context menu
Using object links
Executing queries
Viewing query logs
Favorites editor
Merging queries

6.1.5 Using object links

Objects that exist in the database are highlighted in the text as hyperlinks. You can open an object in the appropriate editor by clicking the object name in the text with the *Ctrl* key pressed.



Please note that you can change the way highlighted objects look in the editor: use the Color section of the Editor Options dialog.

See also:
Using Navigation bar
Working with SQL Editor area
Using the context menu

© 2008 EMS Database Management Solutions

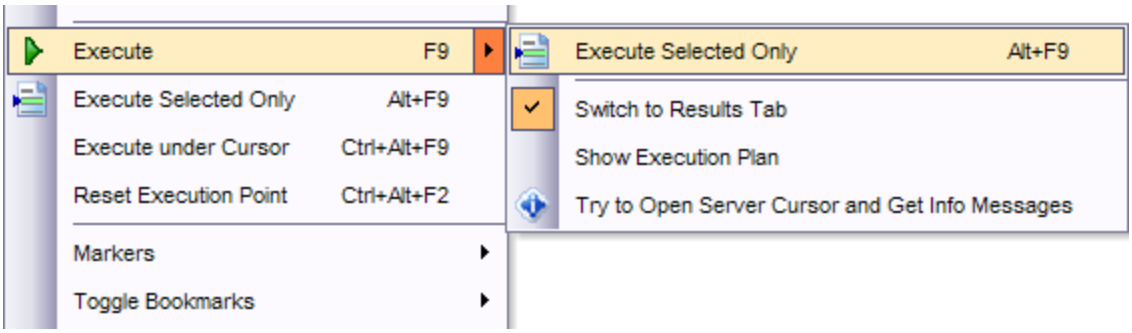
Viewing query plan
Executing queries
Viewing query logs
Favorites editor
Merging queries

6.1.6 Executing queries

When all the query parameters are set, you can immediately **execute the query** in **SQL Editor**.

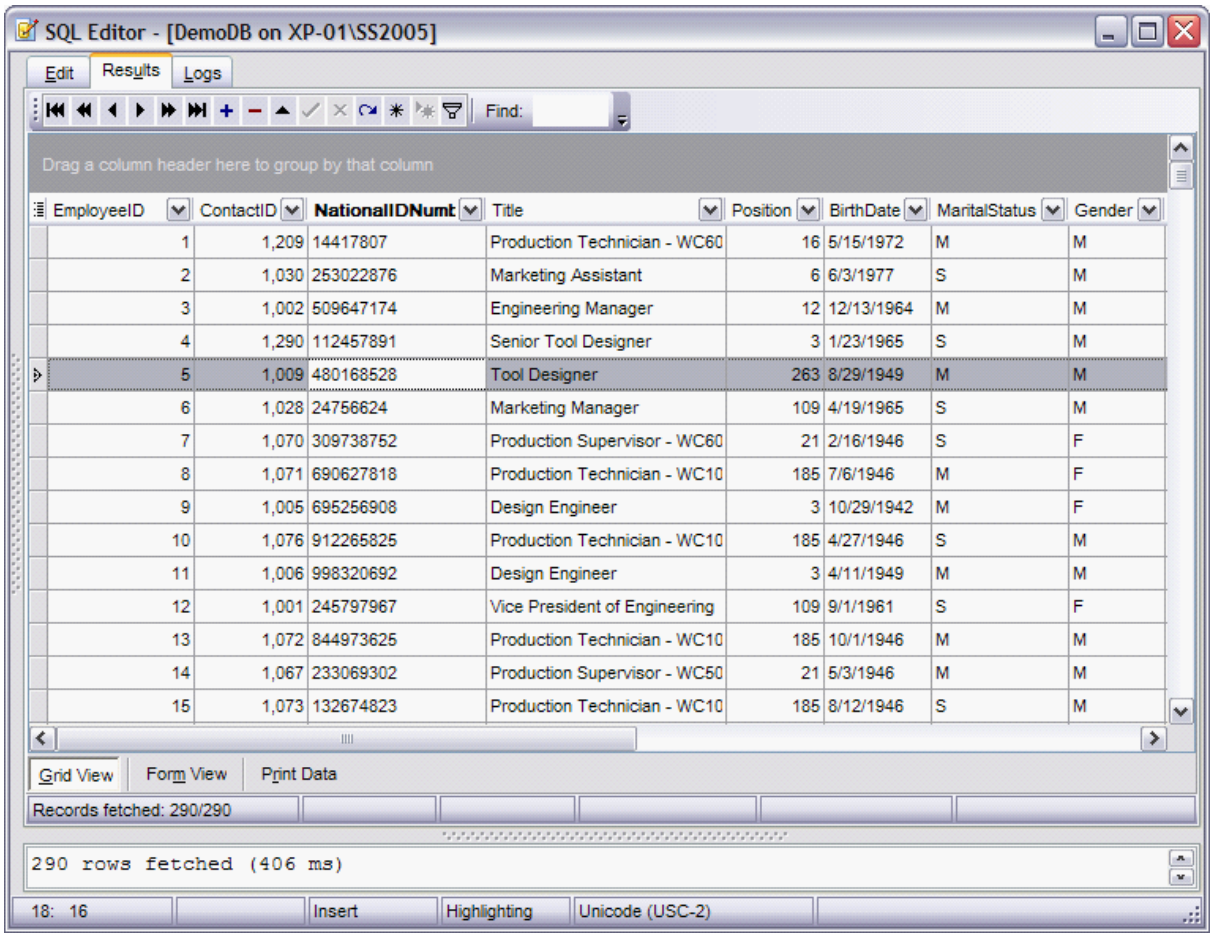
To execute a query, click the **Execute** item of the Navigation bar. You can also use the context menu or *F9* hot key for the same purpose.

If the SQL syntax is correct, the query is executed and, in case the query statement is supposed to return data (e.g. as SELECT statement), the returned dataset appears within the **Results** tab. The position of the tab depends on the **Results on Edit tab / Results on separate tab** selection in the Navigation bar.
If SQL syntax of the query contains any errors, the query execution is stopped and the corresponding error message is displayed in the status bar area at the bottom of the editor window.



By default, data returned by a query are displayed as a grid (see Data View for details). The context menu of the grid allows you to Export Data, Export as SQL Script.

© 2008 EMS Database Management Solutions



See also:

Using Navigation bar
Working with SQL Editor area
Using the context menu
Viewing query plan
Using object links
Viewing query logs
Favorites editor
Merging queries

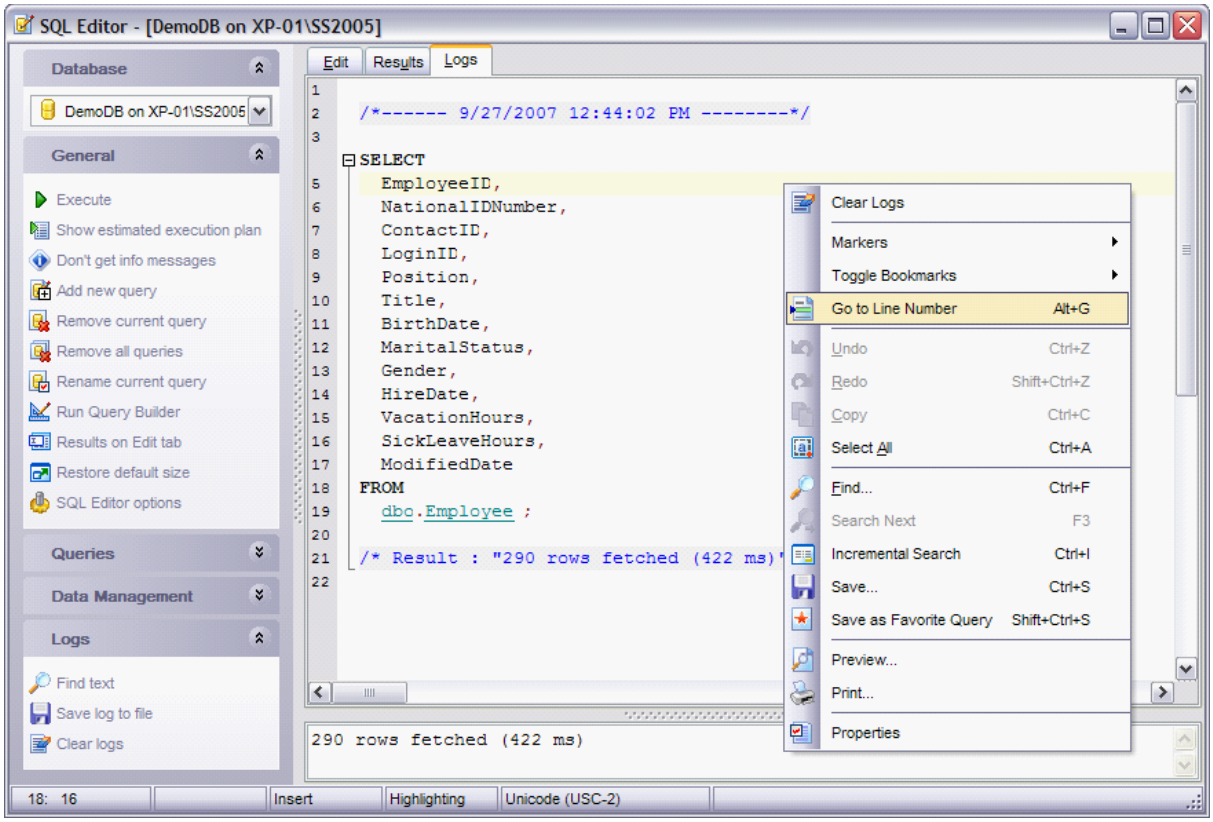
6.1.7 Viewing query logs

This tab allows you to view the query **log**. The log is available within the **Logs** tab of **SQL Editor**.

Using this tab you can view the text of each query, information on the date and time of the query execution, the number of rows fetched, etc. With the help of the context

© 2008 EMS Database Management Solutions

menu the log can be printed, saved to file or cleared. You can also use a number of SQL Editor context menu generic functions.



See also:

Using Navigation bar
Working with SQL Editor area
Using the context menu
Viewing query plan
Using object links
Executing queries
Favorites editor
Merging queries

6.1.8 Favorites editor

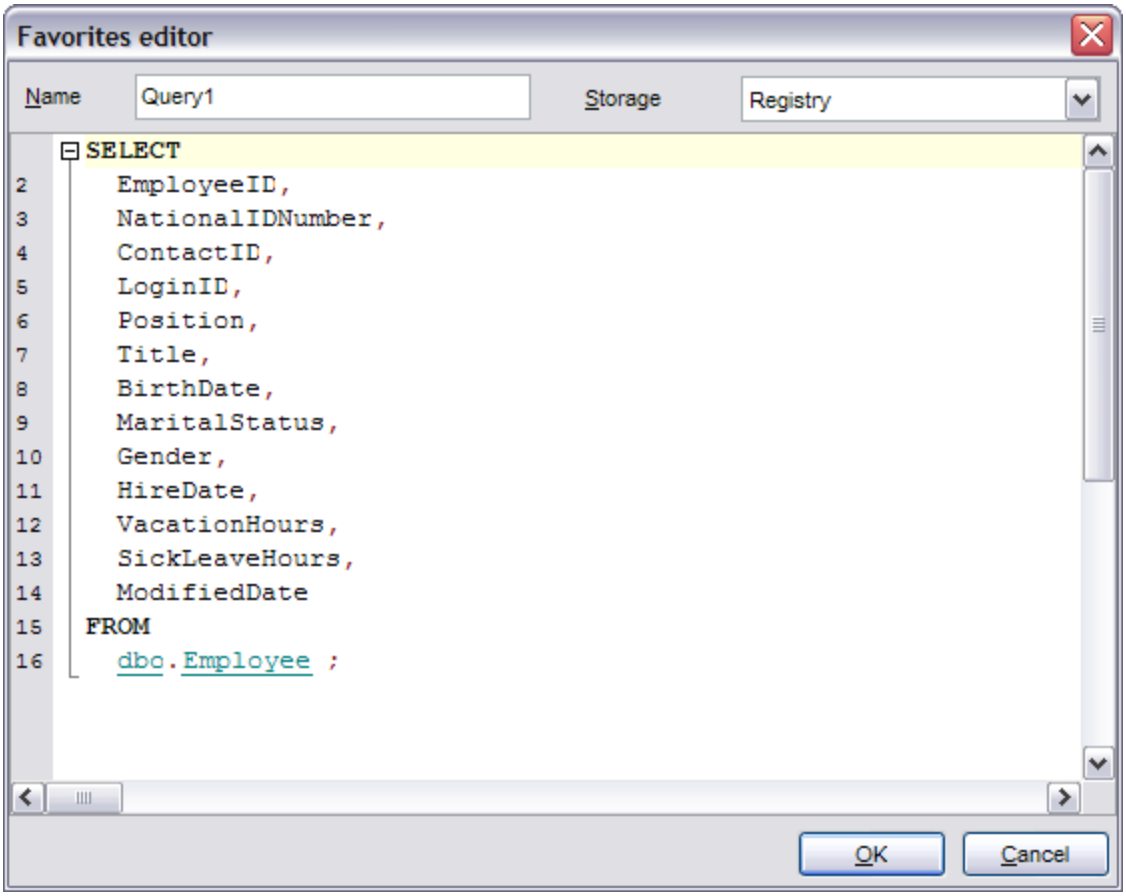
For your convenience the **Favorite Queries** list is implemented in SQL Manager 2008 for SQL Server. This list is available within the **Favorite Queries** node of Database Explorer and allows you to store the most frequently used SQL queries in one location.

To add a query to the **Favorite Queries** list, use the **Add to Favorite Queries** Navigation bar item in **SQL Editor**. The corresponding item is also available in the context

© 2008 EMS Database Management Solutions

menu of SQL Editor working area.

You can edit any of your Favorite Queries using **Favorites editor**.



Name
Set the name of the Favorite query.

Storage
Specify where the Favorite query will be stored: in *Windows Registry* or in the *Database*.

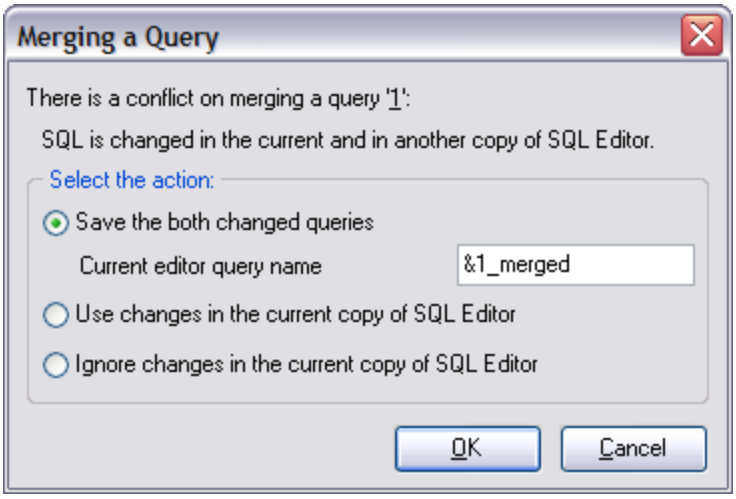
See also:

Using Navigation bar
Working with SQL Editor area
Using the context menu
Viewing query plan
Using object links
Executing queries
Viewing query logs
Merging queries

© 2008 EMS Database Management Solutions

6.1.9 Merging queries

When editing the same query in several copies of SQL Editor, on attempt to close the **SQL Editor** window the **Merging a Query** dialog will appear like the one displayed below.



The actions offered are the following:

Save the both changed queries
The query opened in the first copy of SQL editor will be saved under its original name; the query in the current copy will be saved under the name with additional postfix (by default it is *'_1'*)

Use changes in the current copy of SQL Editor
The query will be saved under the original name; the modifications of SQL made within the current copy of **SQL Editor** will be applied.

Ignore changes in the current copy of SQL Editor
The query will be saved under its original name; the modifications of SQL made within the current copy of **SQL Editor** will be ignored.

See also:
Using Navigation bar
Working with SQL Editor area
Using the context menu
Viewing query plan
Using object links
Executing queries
Viewing query logs

© 2008 EMS Database Management Solutions

Favorites editor

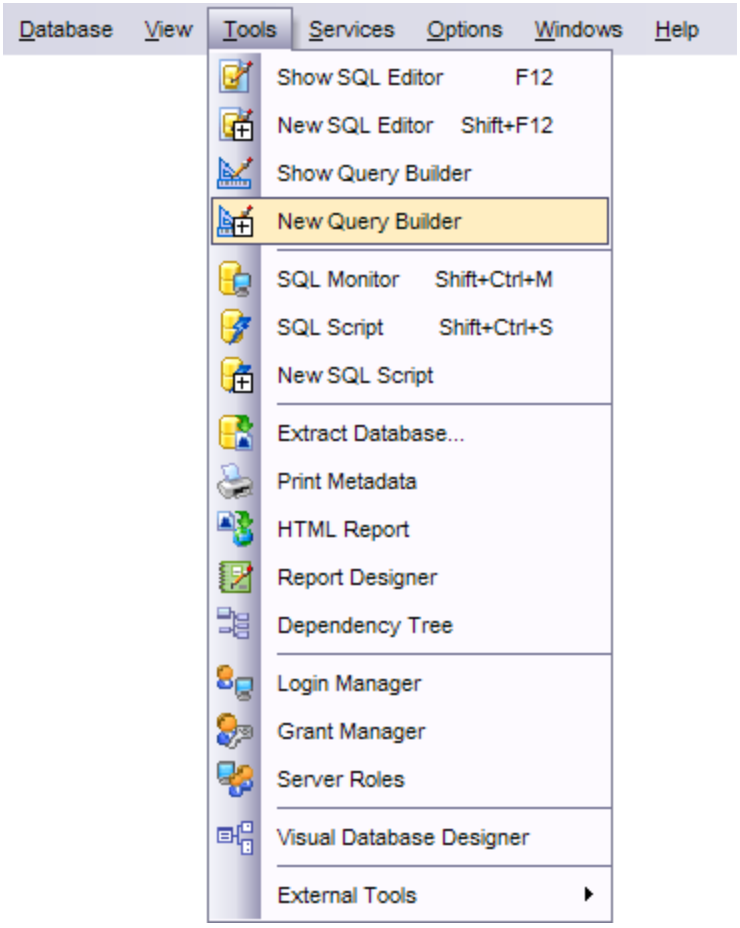
© 2008 EMS Database Management Solutions

6.2 Visual Query Builder

Visual Query Builder is implemented in SQL Manager 2008 for SQL Server for building queries visually. The tool allows you to create and edit queries without deep knowledge of SQL. You can also prepare and execute queries, and view the results of their execution.

To open Visual Query Builder, select the **Tools | New Query Builder** / **Tools | Show Query Builder** main menu items or use the corresponding / toolbar buttons.

- Using Navigation bar
- Working with diagram area
- Joining two database objects by fields
- Setting the selection criteria
- Setting output fields for selection
- Setting the grouping criteria
- Setting parameters of sorting
- Working with editor area
- Executing queries and viewing results
- Viewing query plan



Availability:

© 2008 EMS Database Management Solutions

Full version (for Windows)	**Yes**
Lite version (for Windows)	**No**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

SQL Editor

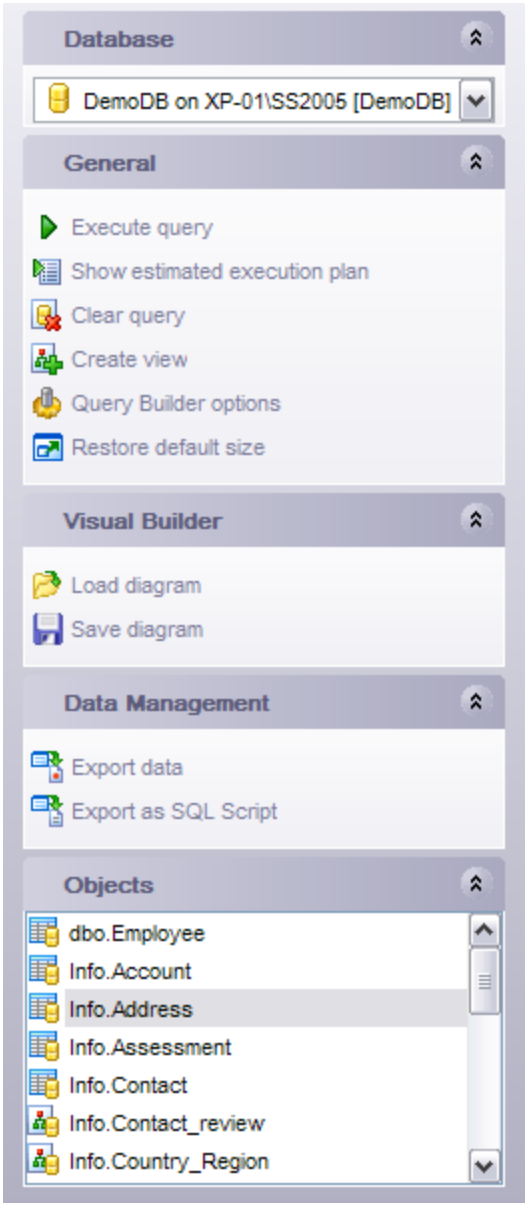
Query parameters

6.2.1 Using Navigation bar

The **Navigation bar** of **Visual Query Builder** allows you to:

- select a database for the query;
- execute the current query;
- view estimated query execution plan;
- clear the query;
- create a view;
- configure Query Builder using the Query Builder Options page;
- restore the default size of the builder window;
- load a diagram from a **.vqb* file using the **Open diagram** dialog;
- save the diagram to a **.vqb* file using the **Save diagram as...** dialog;
- export the returned dataset using Export Data Wizard;
- export data as SQL Script using the Export as SQL Script wizard;
- browse objects of the database; you can also add tables and views to the diagram using drag-and-drop operations.

© 2008 EMS Database Management Solutions



See also:

Working with diagram area
Joining two objects
Setting criteria
Setting output fields
Setting grouping criteria
Setting sorting parameters
Working with the editor area
Query execution
Viewing query plan

© 2008 EMS Database Management Solutions

6.2.2 Working with diagram area

The main working area of **Visual Query Builder** is the diagram area available within the **Builder** tab. Here you can create a query by placing the database tables and views onto the area, and edit it by selecting the required data fields and setting links between objects.

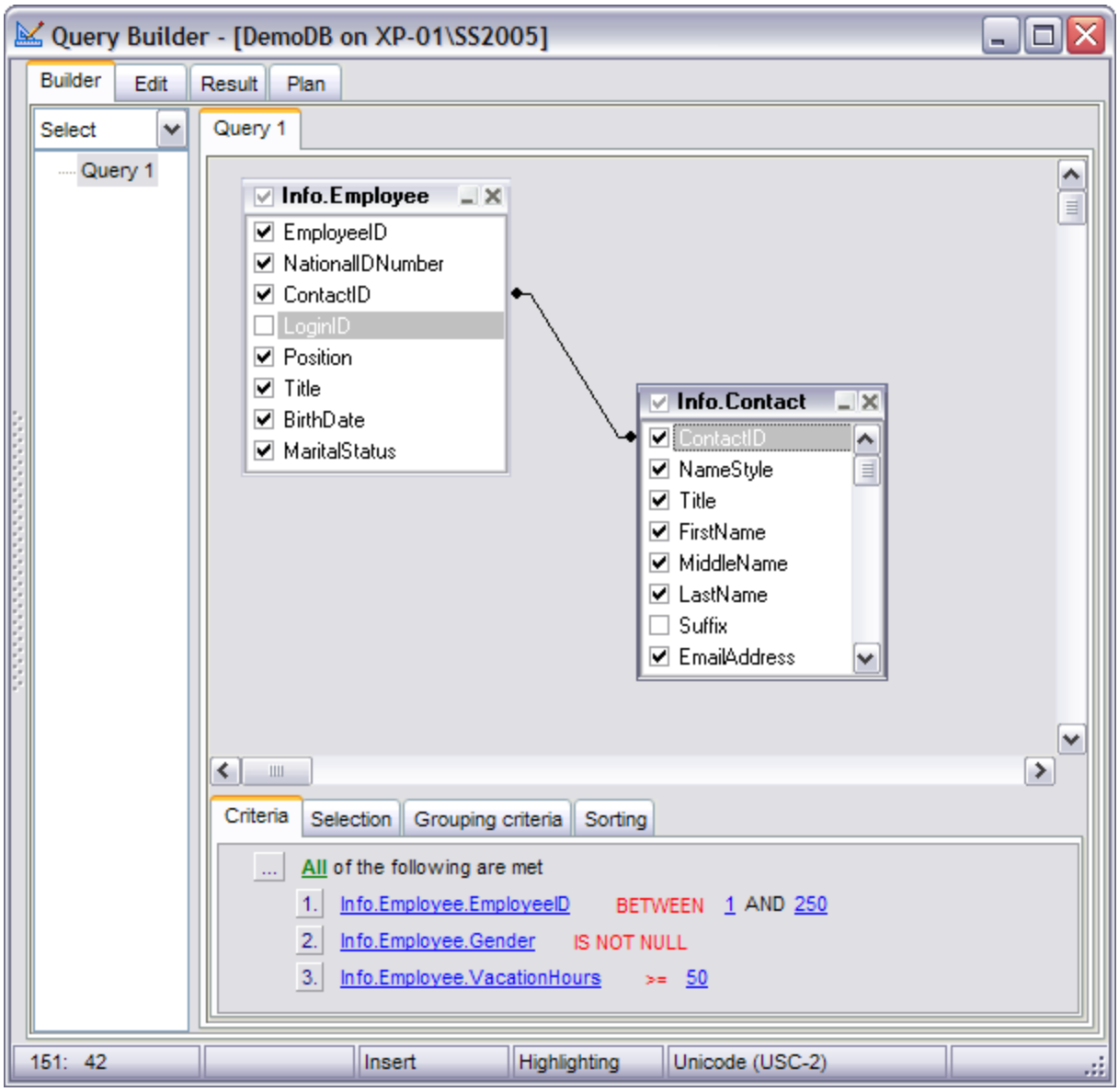
To add an object to the query, you can simply drag it from the DB Explorer tree to the diagram area.

To include a field in the query, check the corresponding box located to the left from the field name in the list, or just double-click it. To include all fields of the table/view, check the box located to the left of the table/view caption. If you do not check any fields, the SQL statement is generated as SELECT * FROM <table/view_name>, i.e. all the fields are included in the query.

To exclude a field from the query, uncheck the respective box. In order to remove the entire table/view from the query, close it by clicking the corresponding cross-button at the object caption, or select the object and press the **Del** key.

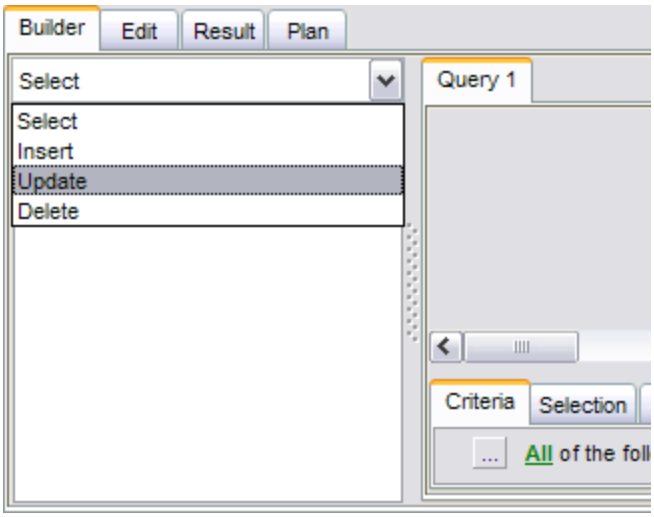
To edit the alias of a table/view, double-click the object caption and enter the new name.

© 2008 EMS Database Management Solutions



Note: Depending on which query type you need to execute, you can select one from the drop-down list at the left: *Select*, *Insert*, *Update*, or *Delete*.

© 2008 EMS Database Management Solutions



See also:
Using Navigation bar
Joining two objects
Setting criteria
Setting output fields
Setting grouping criteria
Setting sorting parameters
Working with the editor area
Query execution
Viewing query plan

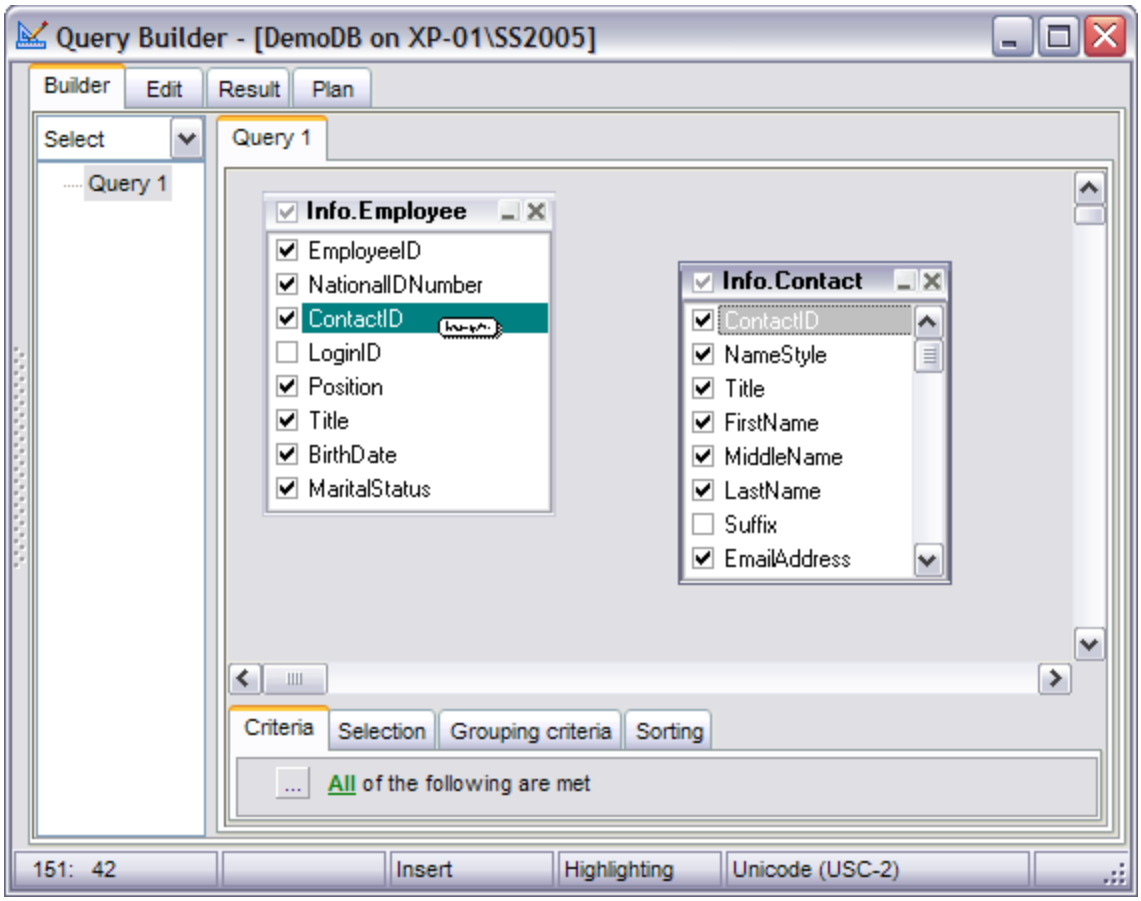
6.2.3 Joining two objects

The **diagram area** allows you to associate two objects by their fields: this operation is performed by dragging a field from one object list to another. This will set a link between these objects by the selected fields. It is indicated by a bidirectional arrow between the linked fields.

Note: Once two or more tables having foreign key connection are added to the query, the corresponding visual joining of these tables appears at the **Builder** tab and necessary *JOIN* statements appear on the **Edit** tab.

You can *view the link properties* of objects association: set the mouse cursor over the linking arrow, and a hint containing the association condition will popup after a short delay.

© 2008 EMS Database Management Solutions



To edit the link properties, double-click the linking arrow or right-click it and select the **Property** popup menu item. The **Link properties** dialog allows you to change the association condition by choosing it from the drop-down list (=, >, <, >=, <=, <>).

For your convenience the **Include all** option is available for each object of the association. Click **OK** to apply the changes you have made.

To remove a link between objects, right-click the linking arrow and select the **Delete link** popup menu item.

To add a point to the link line, right-click the linking arrow and select the **Insert point**

© 2008 EMS Database Management Solutions

popup menu item. Using the point you can move the link line easily. The point does not cause any changes to the query, it is only used for the diagram representation and makes visual building handy and more comprehensible.

See also:
Using Navigation bar
Working with diagram area
Setting criteria
Setting output fields
Setting grouping criteria
Setting sorting parameters
Working with the editor area
Query execution
Viewing query plan

6.2.4 Setting criteria

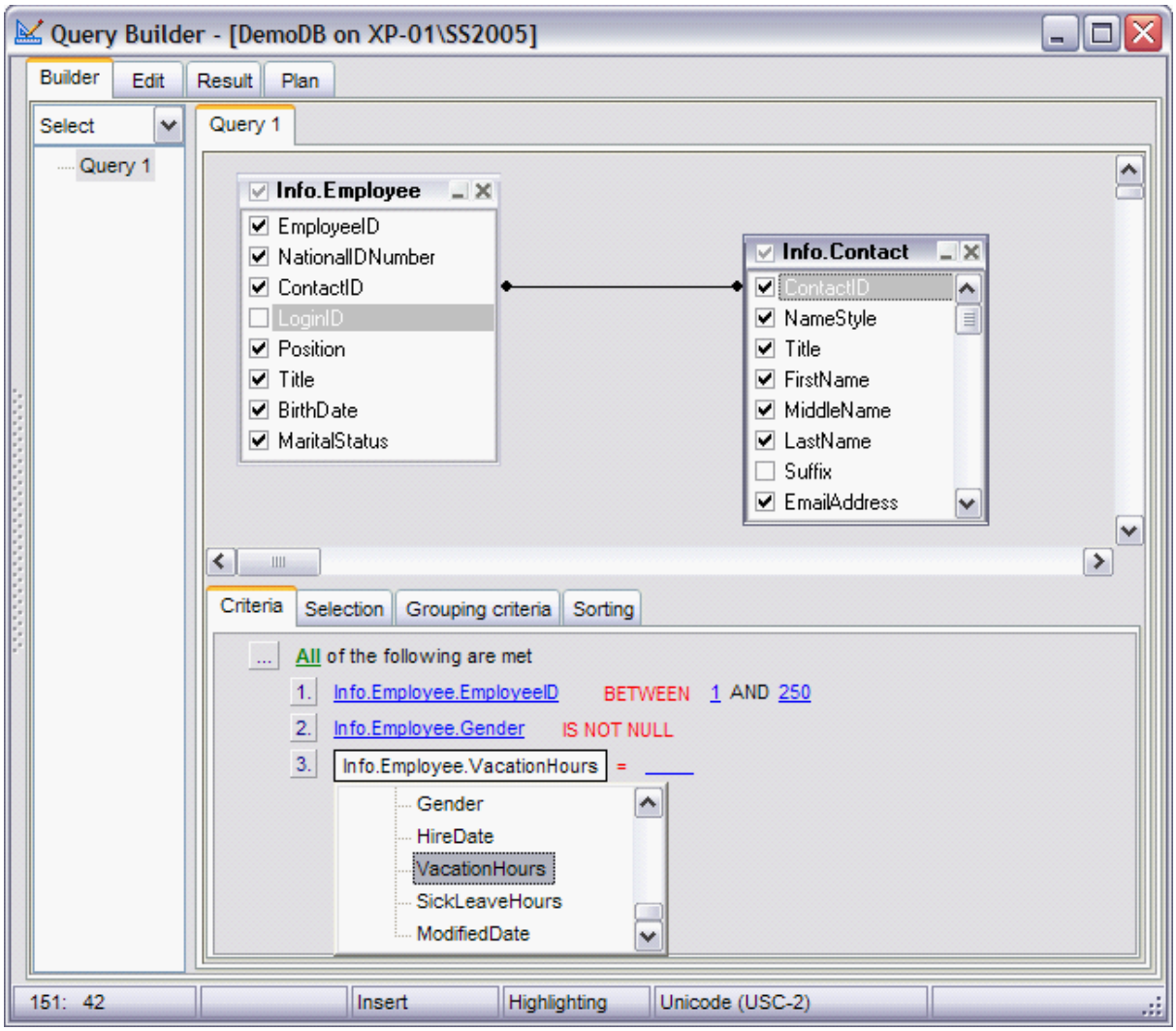
Use the **Criteria** tab to set the selection conditions.

To add a condition, click the ellipsis button on the left, and select the **Add condition** popup menu item.

Edit the condition by clicking the elements of the condition pattern and setting the necessary values. Clicking the numbered 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*, *expand* or *collapse* enclosure levels of the condition (if the condition is composite).

A simple condition pattern contains three elements: *an argument*, *a condition operator* and *a second argument* (if required for the condition).
Clicking each element field allows you to set its value. When clicking the argument field, you can edit the argument as a text string: set an object name or a certain value in this field. Right-clicking the field in the edit mode activates the popup menu with the **Insert field** function (also called by the *Shift+Enter* shortcut). This function allows you to select a field from the list of all the table fields.
Clicking the condition operator field activates the popup menu where you can specify the condition you need.

© 2008 EMS Database Management Solutions



The way the conditions are used is set in the upper string of the area (*All*, *Any*, *None* or *Not all of the following are met*). Click the green link to change it.

See also:

Using Navigation bar
Working with diagram area
Joining two objects
Setting output fields
Setting grouping criteria
Setting sorting parameters
Working with the editor area
Query execution
Viewing query plan

© 2008 EMS Database Management Solutions

6.2.5 Setting output fields

The **Selection** tab displays the output fields of the query as a grid.

The grid allows you to edit the names of the query output fields, specify their displaying order and set the aggregate functions (*SUM*, *MIN*, *MAX*, *AVG*, and *COUNT*) for each field. To remove a field from the list, right-click the field row and select the **Delete current row** popup menu item. To change the input query field, double click it and then type the field name on the keyboard or choose it from the drop-down list.

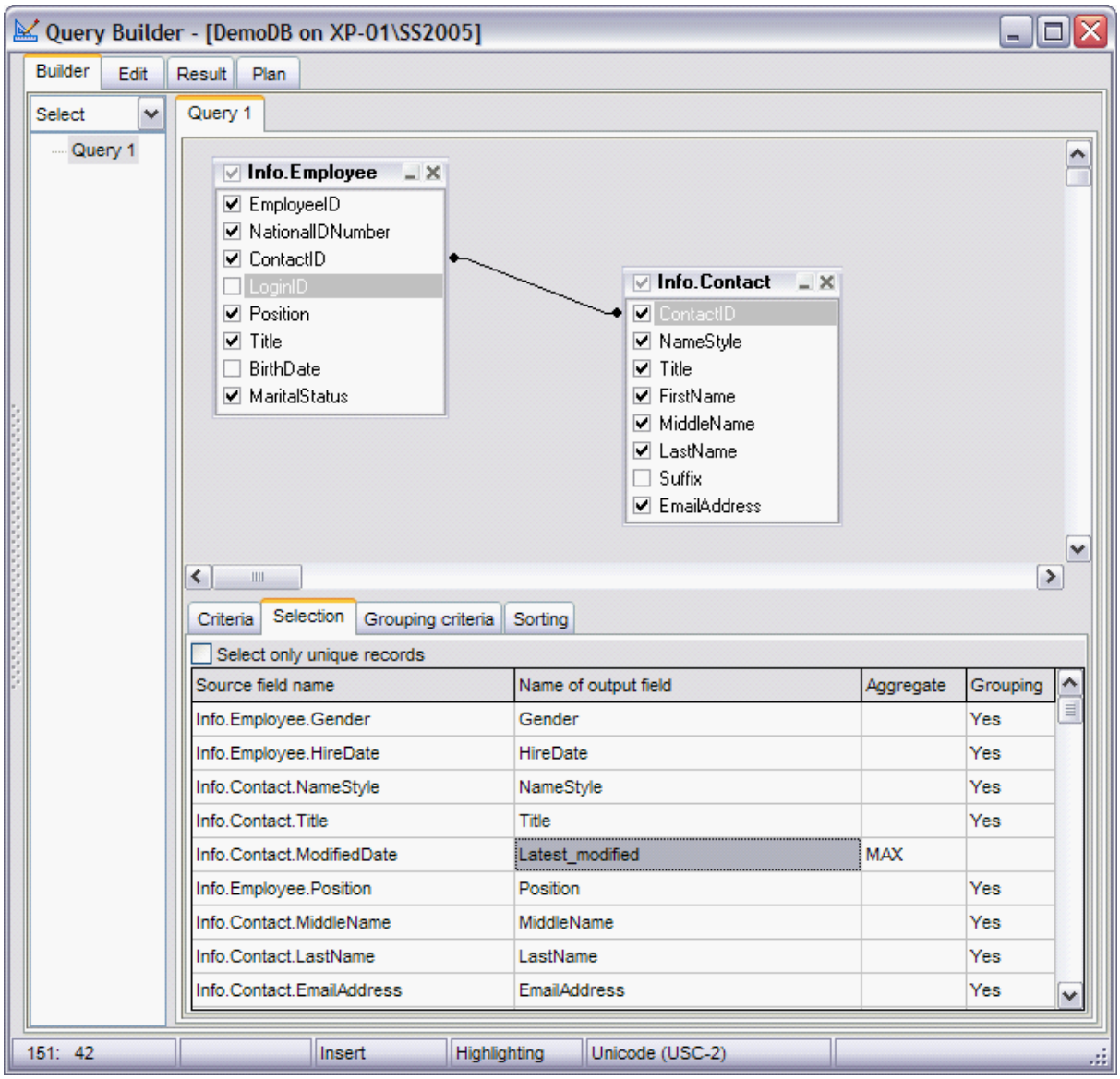
To change the output query field name, set the cursor at the corresponding column and type the required field name.

To set an aggregate function for a field, click the field row within the **Aggregate** column, and then type in the function name or select one from the drop-down list.

The **Grouping** column displays the grouping state for each of the output fields.

☑ **Select only unique records**
If you check this option, the duplicate records (if any) are not included into the query result (i.e. the DISTINCT keyword is added to the SQL query text).

© 2008 EMS Database Management Solutions



See also:

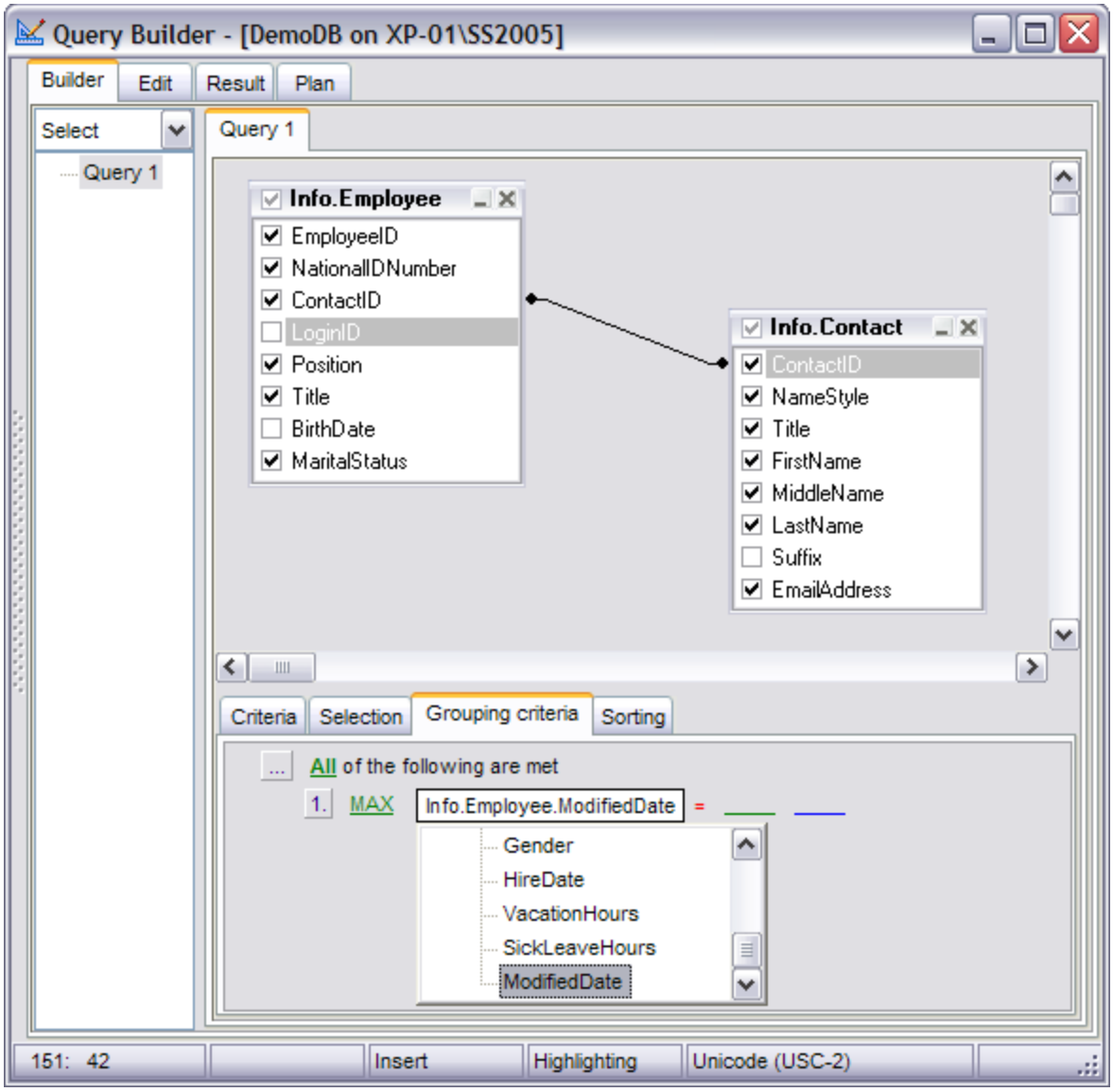
[Using Navigation bar](#)
[Working with diagram area](#)
[Joining two objects](#)
[Setting criteria](#)
[Setting grouping criteria](#)
[Setting sorting parameters](#)
[Working with the editor area](#)
[Query execution](#)
[Viewing query plan](#)

© 2008 EMS Database Management Solutions

6.2.6 Setting grouping criteria

The **Grouping criteria** tab allows you to set conditions for grouping query records.

The grouping condition pattern fields are set in the same way as those of the Criteria pattern.



These conditions will be included in the HAVING statement of the generated SQL query.

See also:

Using Navigation bar

© 2008 EMS Database Management Solutions

Working with diagram area
Joining two objects
Setting criteria
Setting output fields
Setting sorting parameters
Working with the editor area
Query execution
Viewing query plan

6.2.7 Setting sorting parameters

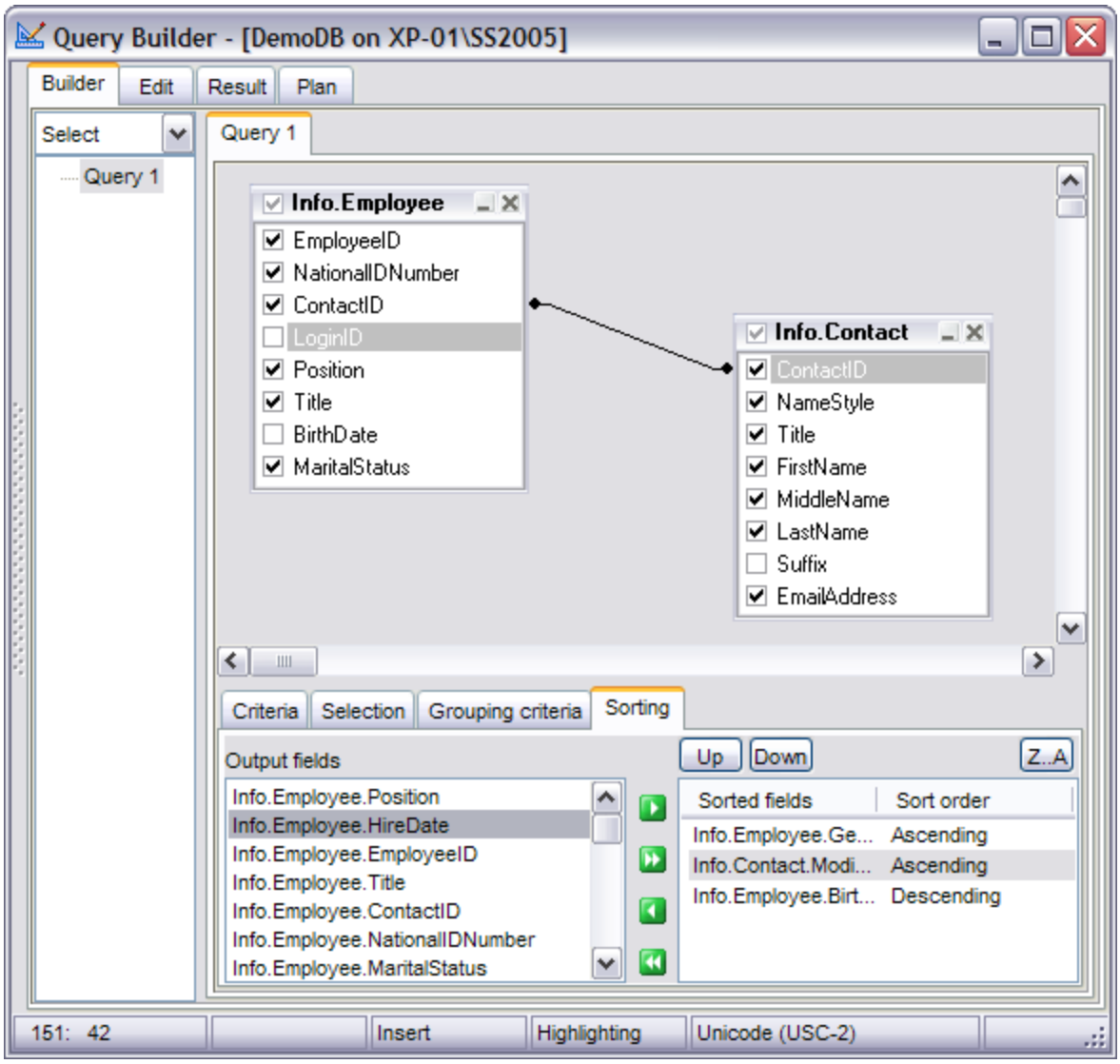
The **Sorting** tab allows you to set sorting parameters for the records returned by the query.

The working area contains the **Output fields** list (at the left) which represents all the output query fields, and the **Sorted fields** list (at the right) which contains the fields to sort records by.
To move a field from one list to another, drag the selected field or use the **Add** and **Remove** buttons.

To change the sorting order for a sorted field, select the field in the **Sorted fields** list and move it using the **Up** and **Down** buttons.

To change the sorting direction, select the field in the **Sorted fields** list and switch the direction (*Ascending*, *Descending*) using the corresponding **A..Z**/**Z..A** button.

© 2008 EMS Database Management Solutions



See also:

Using Navigation bar
Working with diagram area
Joining two objects
Setting criteria
Setting output fields
Setting grouping criteria
Working with the editor area
Query execution
Viewing query plan

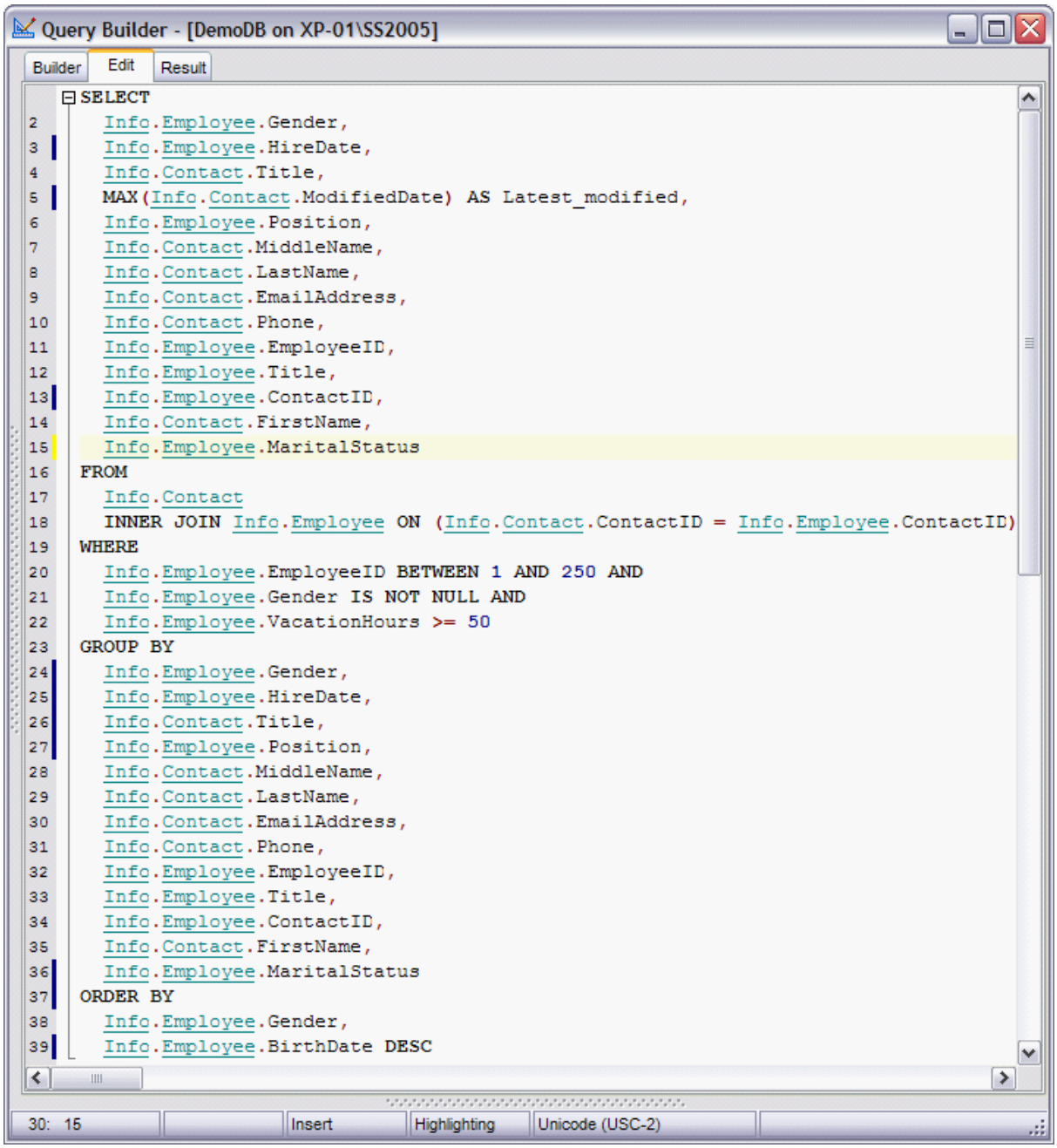
© 2008 EMS Database Management Solutions

6.2.8 Working with the editor area

The **Editor area** of **Visual Query Builder** is available within the **Edit** tab and is provided for working directly with the SQL query text which is generated automatically while you build the query visually.

You can edit this text according to the rules of SQL, and all the changes will be displayed within the **Builder** tab respectively.

To learn more about the SQL Editor features available within the **Edit** tab, see Working with SQL Editor area.



© 2008 EMS Database Management Solutions

See also:
Using Navigation bar
Working with diagram area
Joining two objects
Setting criteria
Setting output fields
Setting grouping criteria
Setting sorting parameters
Query execution
Viewing query plan

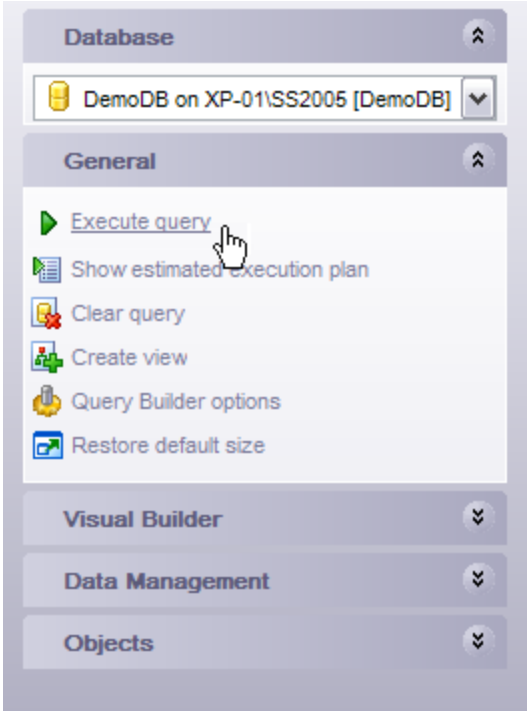
6.2.9 Query execution

When all the query parameters are set, you can immediately **execute the query** in **Visual Query Builder**.

To execute a query, click the **Execute query** item of the Navigation bar. You can also use the *F9* hot key for the same purpose.

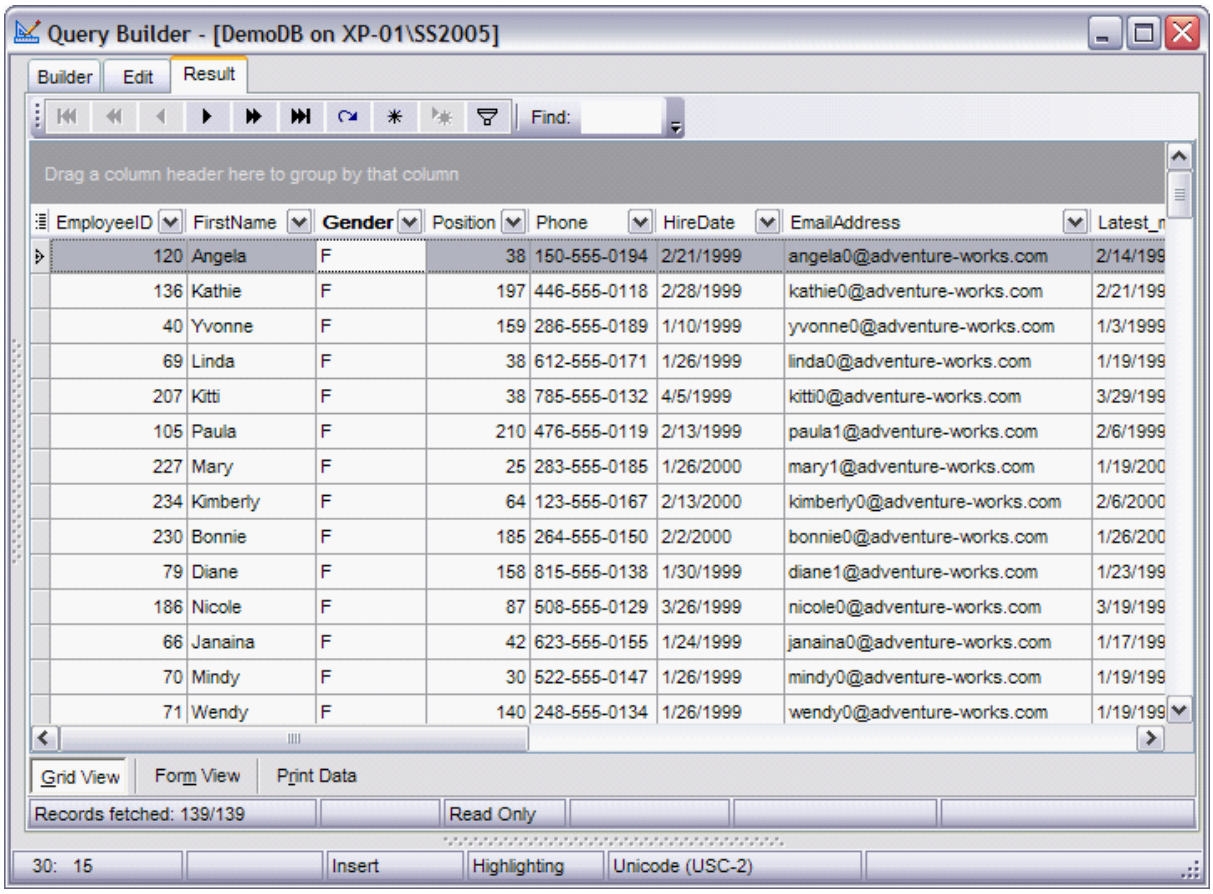
If the query parameters are specified correctly, the query is executed and, in case the query statement is supposed to return data (e.g. as SELECT statement), the returned dataset appears within the **Result** tab.
If SQL syntax of the query contains any errors, the query execution is stopped and the corresponding error message is displayed in the status bar area at the bottom of the Builder window.

© 2008 EMS Database Management Solutions



By default, data returned by a query are displayed as a grid (see Data View for details). The context menu of the grid allows you to Export Data, Export As SQL Script.

© 2008 EMS Database Management Solutions



See also:

Using Navigation bar
Working with diagram area
Joining two objects
Setting criteria
Setting output fields
Setting grouping criteria
Setting sorting parameters
Working with the editor area
Viewing query plan

6.2.10 Viewing query plan

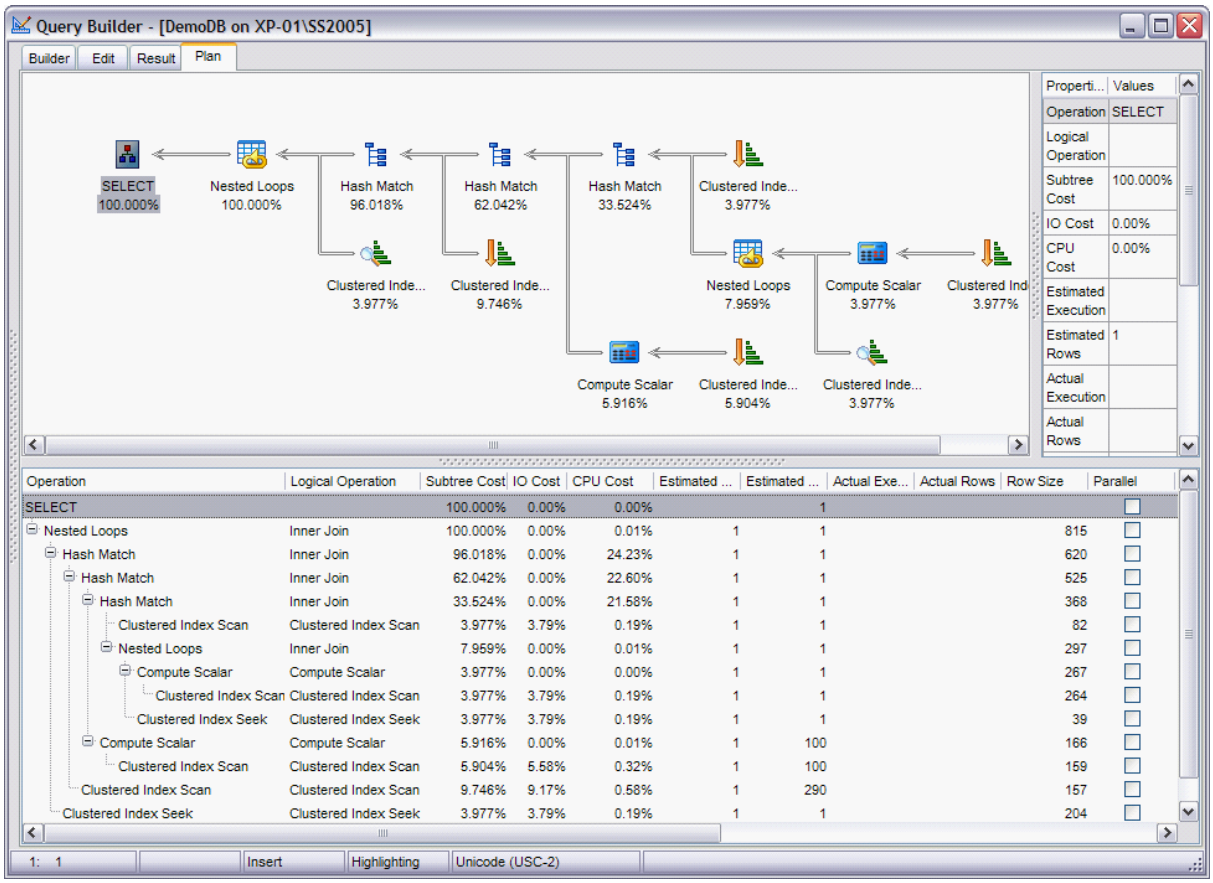
Using SQL Manager 2008 for SQL Server, you can view **the plan** for each of the queries created and executed in the application. The query plan is available within the corresponding **Plan** tab.

To view the **Plan** of a query, open **Query Builder** and use the **Show estimated**

© 2008 EMS Database Management Solutions

execution plan item of the Navigation bar.

The **Plan** tab allows you to view the sequence of actions performed by the database server in the process of the query execution, and the amount of system resources used for the query execution. Note that this tab only appears after you have executed the query.



If necessary, you can specify that the **Plan** tab appears automatically upon query execution in Query Builder: select the ☑ **Show actual execution plan on query** **execution** option available within the Tools | Query Builder section of the Environment Options dialog.

See also:

Using Navigation bar
Working with diagram area
Joining two objects
Setting criteria
Setting output fields
Setting grouping criteria
Setting sorting parameters

© 2008 EMS Database Management Solutions

Working with the editor area
Query execution

© 2008 EMS Database Management Solutions

6.3 Query parameters

Both SQL Editor and Visual Query Builder support parameters usage inside the query text. A parameter is a kind of variable for which a value can be specified just before the query execution. In the query text the parameter should appear as an identifier with a colon (':') at the beginning, e.g.

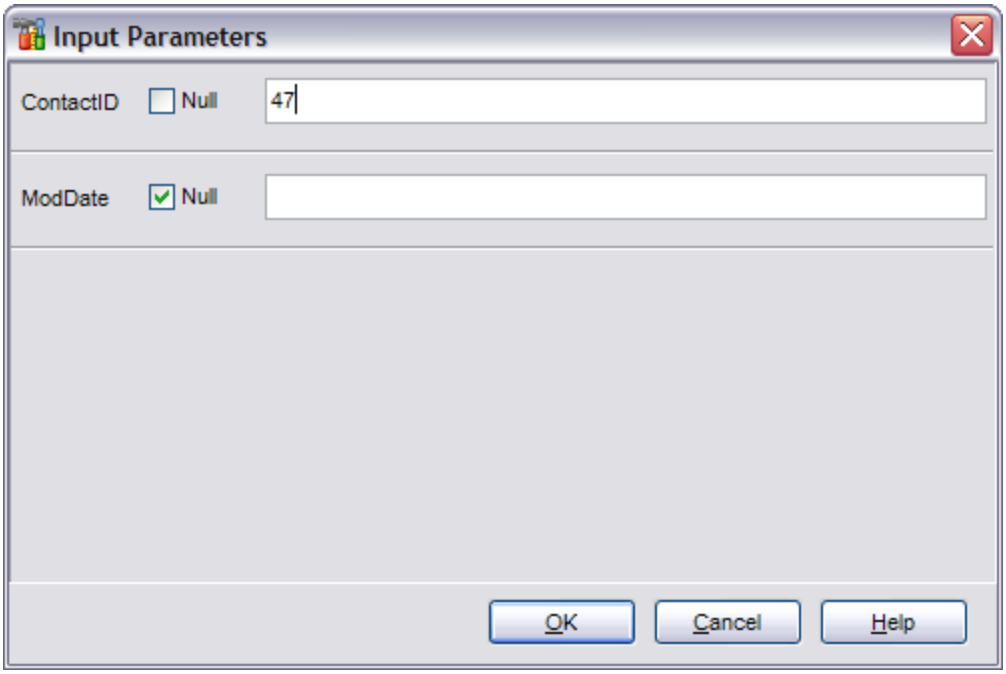
:param1.

Note: The **Allow using of parameters in query text** option should be checked on the **Options | Environment options | Tools** page for this feature to be enabled.

See also:
SQL Editor
Visual Query Builder

6.3.1 Input parameters dialog

The **Input Parameters** dialog is used to specify the query parameters as well as values of the input parameters of the query before execution.



The edit field for input parameters varies according to the field data type. For your convenience the *Calculator* and *Date editor* are implemented for *Numeric* and *Date and Time* types respectively: click the arrow-down button to call the *Calculator* / *Date editor* popup window.

Click **OK** button to apply the values and execute the query or click **Cancel** button to

© 2008 EMS Database Management Solutions

abort execution.

© 2008 EMS Database Management Solutions

Part

7 Data management

Table data and query results are displayed on the **Data** or **Results** tab of Table Editor, SQL Editor, Visual Query Builder, etc.

Data can be displayed in one of the following modes: **Grid View**, **Form View**, **Print Data**, **BLOB View**. See Data View to learn more about these modes. You are also provided with a number of filtering tools when working with your data.

- Data View
- Custom Filter
- Filter Builder dialog

See also:

Getting started
Database Explorer
Database management
Database objects management
Query management tools
Import/Export tools
Database tools
Server tools
Personalization
External tools

© 2008 EMS Database Management Solutions

7.1 Data View

SQL Manager 2008 for SQL Server provides you with powerful tools for **viewing, editing and printing data** from tables and queries:

- table / view data are available within the **Data** tab of Table Editor / View Editor correspondingly;
- upon a query execution the returned dataset appears within the **Result(s)** tab of SQL Editor / Query Builder (in SQL Editor the position of the tab depends on the **Results on Edit tab / Results on separate tab** selection in the Navigation bar).

The data can be displayed in one of four available **modes**: *Grid View*, *Form View*, *Print Data* and *BLOB View*. The **status bar** at the bottom displays the number of records in the current dataset, the time the records were fetched by the application and the status of the records (whether the data are read-only or editable).

Please see the succeeding chapters to learn how to work with your data in the simplest and most efficient way.

- Grid View
- Form View
- Print Data
- BLOB View

See also:

Custom Filter
Filter Builder Dialog

7.1.1 Grid View

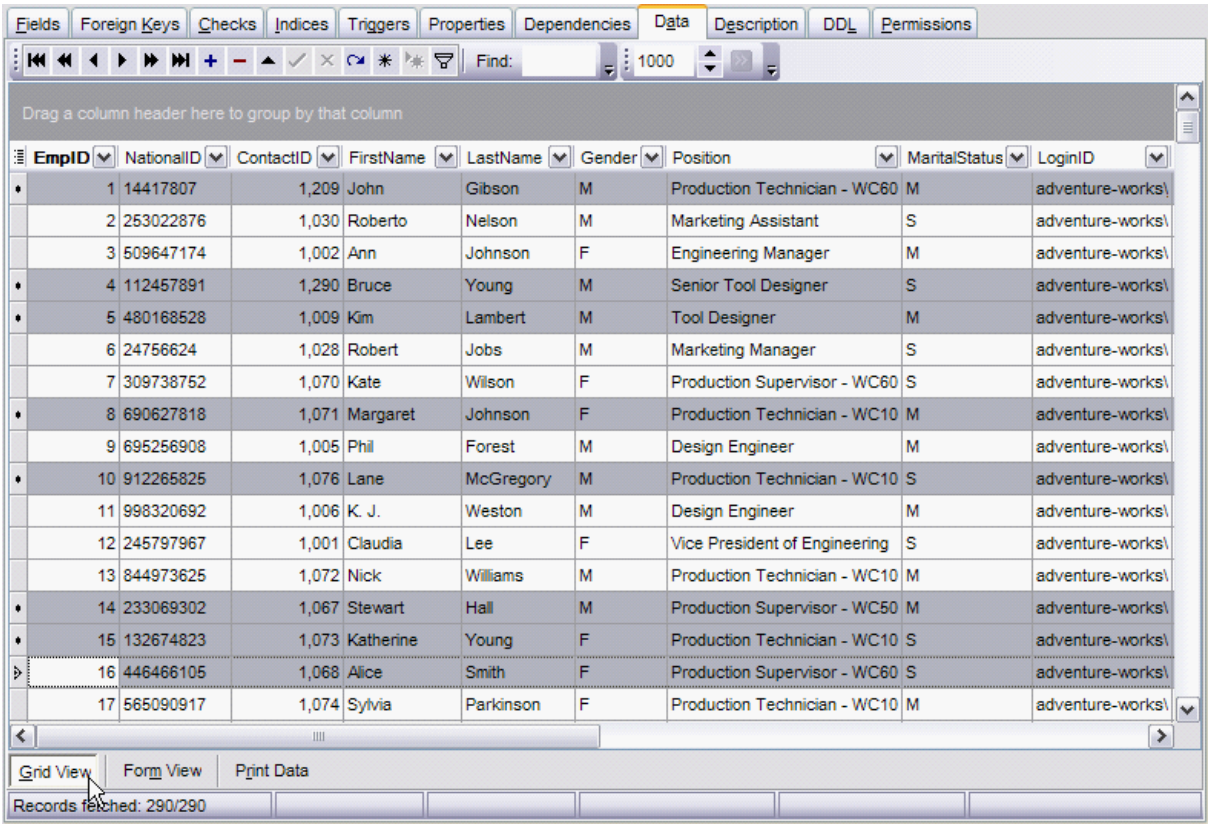
By default, data returned by a query are displayed as a grid. It is indicated by the **Grid View** tab selected on the View mode panel at the bottom of the **Results** area of the window.

When in the **Grid View** mode, the columns correspond to the fields and the rows correspond to the records.
If more convenient, you can change the order of the columns by dragging their headers horizontally. Clicking the column caption sorts data by the values of this column in the ascending or the descending mode. The navigation pane at the top of the grid allows you to browse the data quickly, to insert, update and delete records, and to set a filter for the records using the Filter Builder dialog.

The **Navigation bar** of the parent window and the context menu of the grid provide you with a number of data management functions: Export Data, Import Data, Export As SQL Script and more.

- Grouping data within the grid
- Filtering records
- Using the context menu
- Working in multi-level mode
- Viewing data in card view

© 2008 EMS Database Management Solutions



Hint: To increase the speed of opening tables and views with extremely large number of records, you can use options of the **Limit options in table and view editors** group available in the Grid | Data Options section of the Environment Options dialog.

Availability:

Full version (for Windows) **Yes**

Lite version (for Windows) **Yes**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

Form View
Print Data
BLOB View

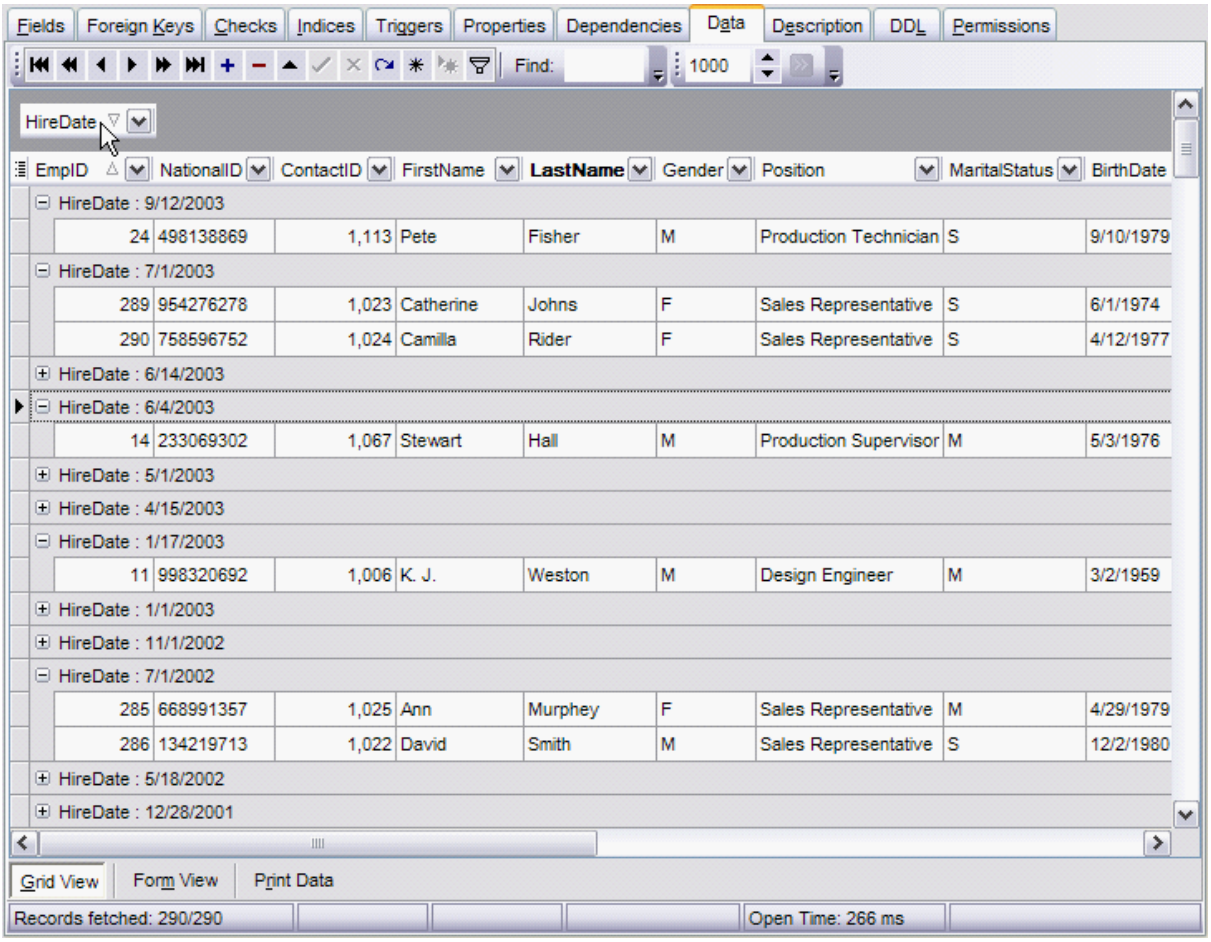
© 2008 EMS Database Management Solutions

7.1.1.1 Grouping data

If necessary, you can **group the data in grid** by any of the columns.

This operation is performed by dragging the column header to the gray **"Group by" box** area at the top. In order to display this area, select the **Show "Group by" box** option available in the Grid section of the Environment Options dialog.

When grouping by a column is applied to the grid, all the records are displayed as subnodes to the grouping row value as displayed in the screenshot below. To reverse grouping, just drag the column header back.



See also:

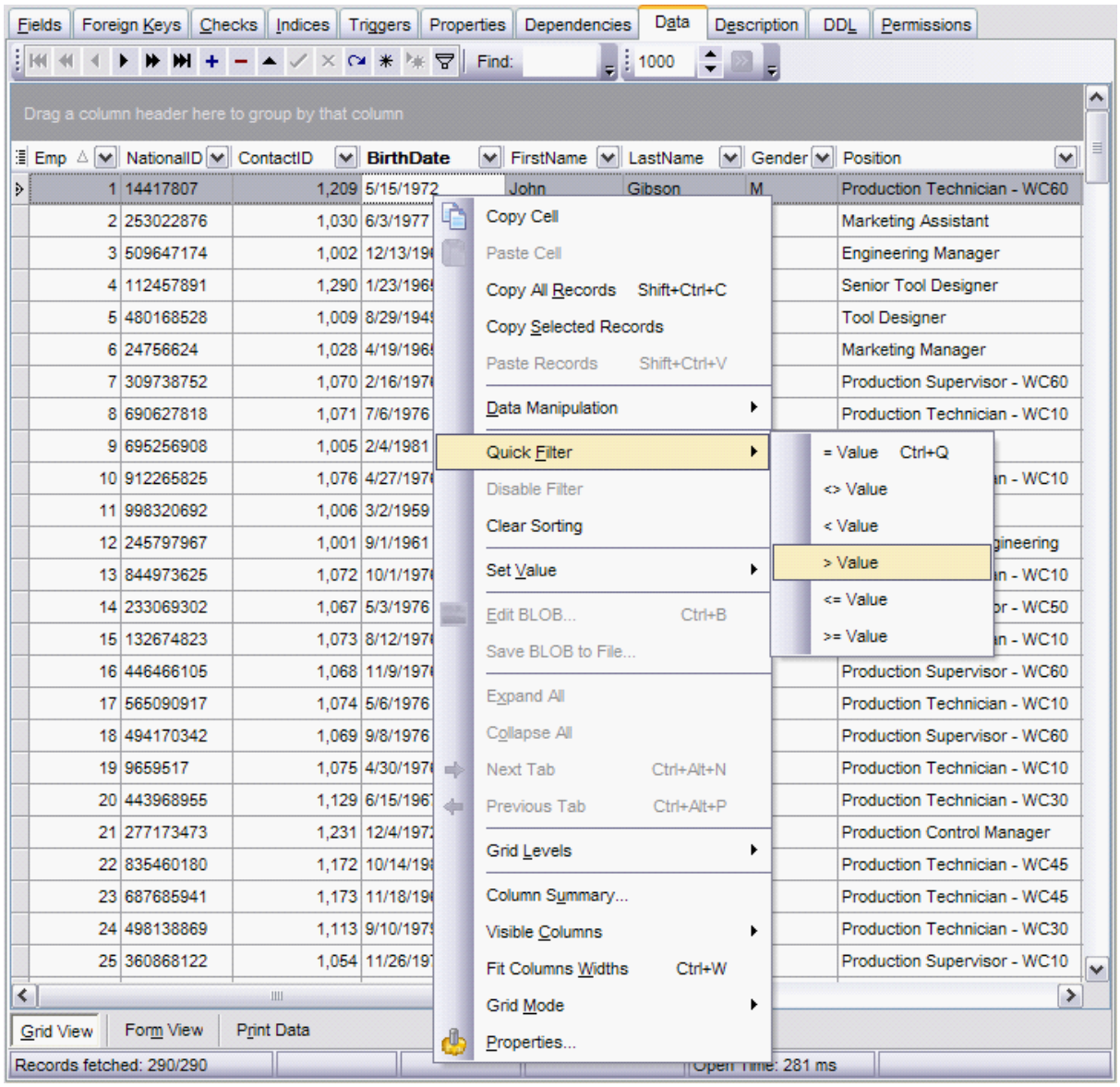
Filtering records
Using the context menu
Working in multi-level mode
Create Grid Level wizard
Working in card view mode
Column Summary

© 2008 EMS Database Management Solutions

7.1.1.2 Filtering records

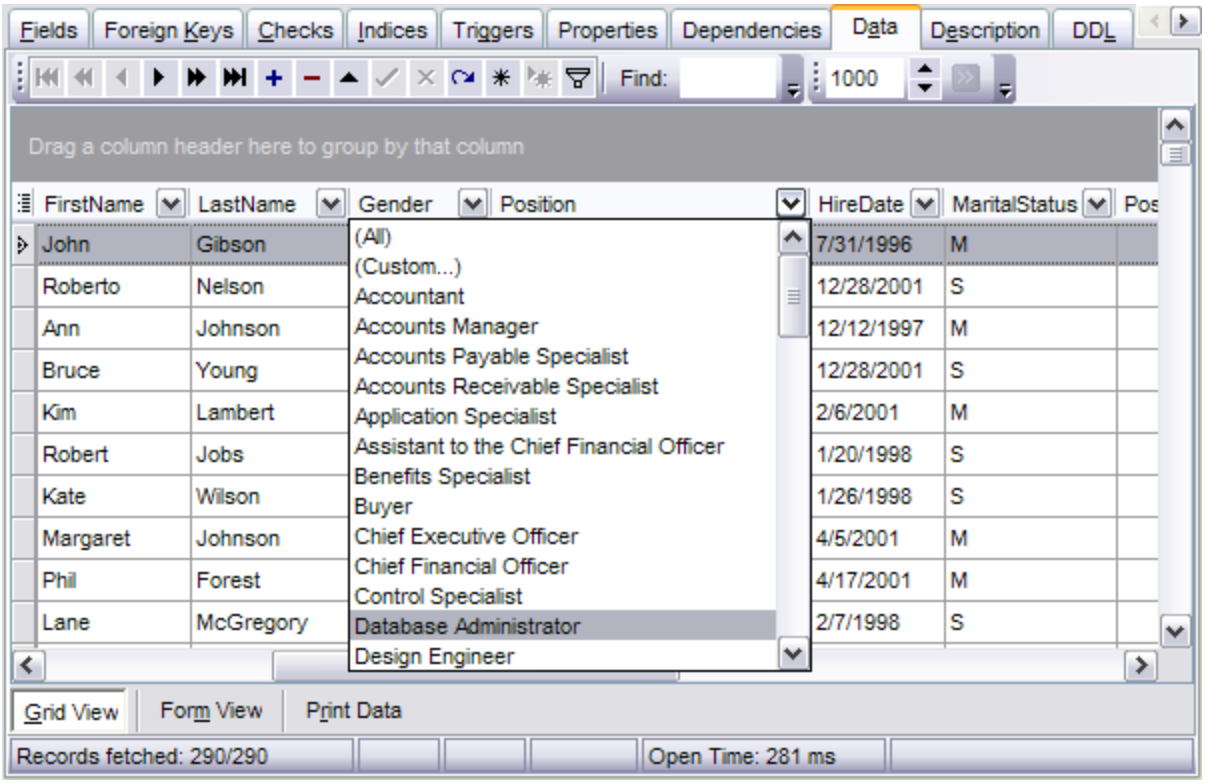
A number of **filtering** facilities are implemented in the grid for your convenience. You can filter records in the grid in either of the following ways:

- right-click a row and select the **Quick Filter** context menu item to filter records by the current value of the selected column;



- click the Arrow-Down button next to the column caption to display the drop-down list and select any of the column values to filter records by this value of the selected column;

© 2008 EMS Database Management Solutions



or

- click the Arrow-Down button next to the column caption to display the drop-down list, then select the **Custom** item and build a simple filter using the Custom Filter dialog;
- use the **Set filter** button on the navigation pane to invoke the Filter Builder dialog and create a composite filter using the dialog.

After the filter is set, the gray **filtering panel** becomes visible at the bottom of the grid. This panel allows you to see the active filtering condition and easily enable or disable it using the checkbox on the left. If necessary, you can click the **Customize...** button on the right to customize your filter and add more complicated filtering conditions within the Filter Builder dialog.

See also:

Grouping data
Using the context menu
Working in multi-level mode
Create Grid Level wizard
Working in card view mode
Column Summary

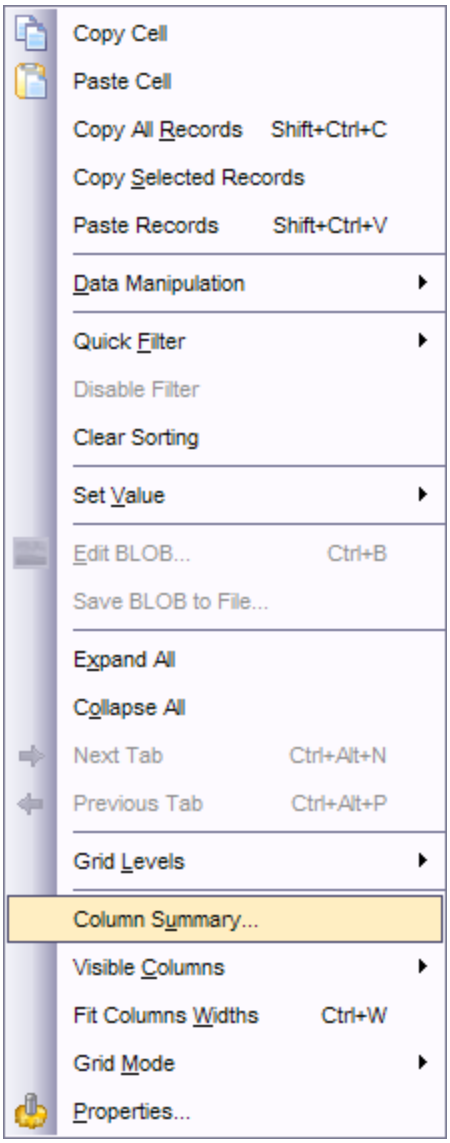
© 2008 EMS Database Management Solutions

7.1.1.3 Using the context menu

The **context menu** of the grid is aimed at facilitating your work with data: you can perform a variety of operations using the context menu items:

- copy the selected cell value to Windows clipboard;
- paste the clipboard content to the currently selected cell;
- copy/paste multiple records;
- data manipulation: Export Data from the table, Import Data to the table, Export Data as SQL Script, Bulk Insert;
- set/disable Quick Filter;
- clear sorting;
- set a value for the selected cell: *NULL*, *Empty string* (for string fields), *Now* (for TIME fields), *"Zero"* (for DATE fields);
- edit the BLOB value or save the BLOB to file using BLOB viewer/editor;
- expand/collapse grid levels and navigate within the tabs;
- manage grid levels: add a new grid level, delete the current grid level;
- switch to the Card View mode, if necessary;
- view Column Summary;
- select visible/invisible columns of the dataset;
- fit column width for better representation;
- specify the grid mode: *Load All Rows*, *Load Visible Rows*, *Default*;
- view/edit grid properties.

© 2008 EMS Database Management Solutions



See also:

Grouping data
Filtering records
Working in multi-level mode
Create Grid Level wizard
Working in card view mode
Column Summary

7.1.1.4 Working in multi-level mode

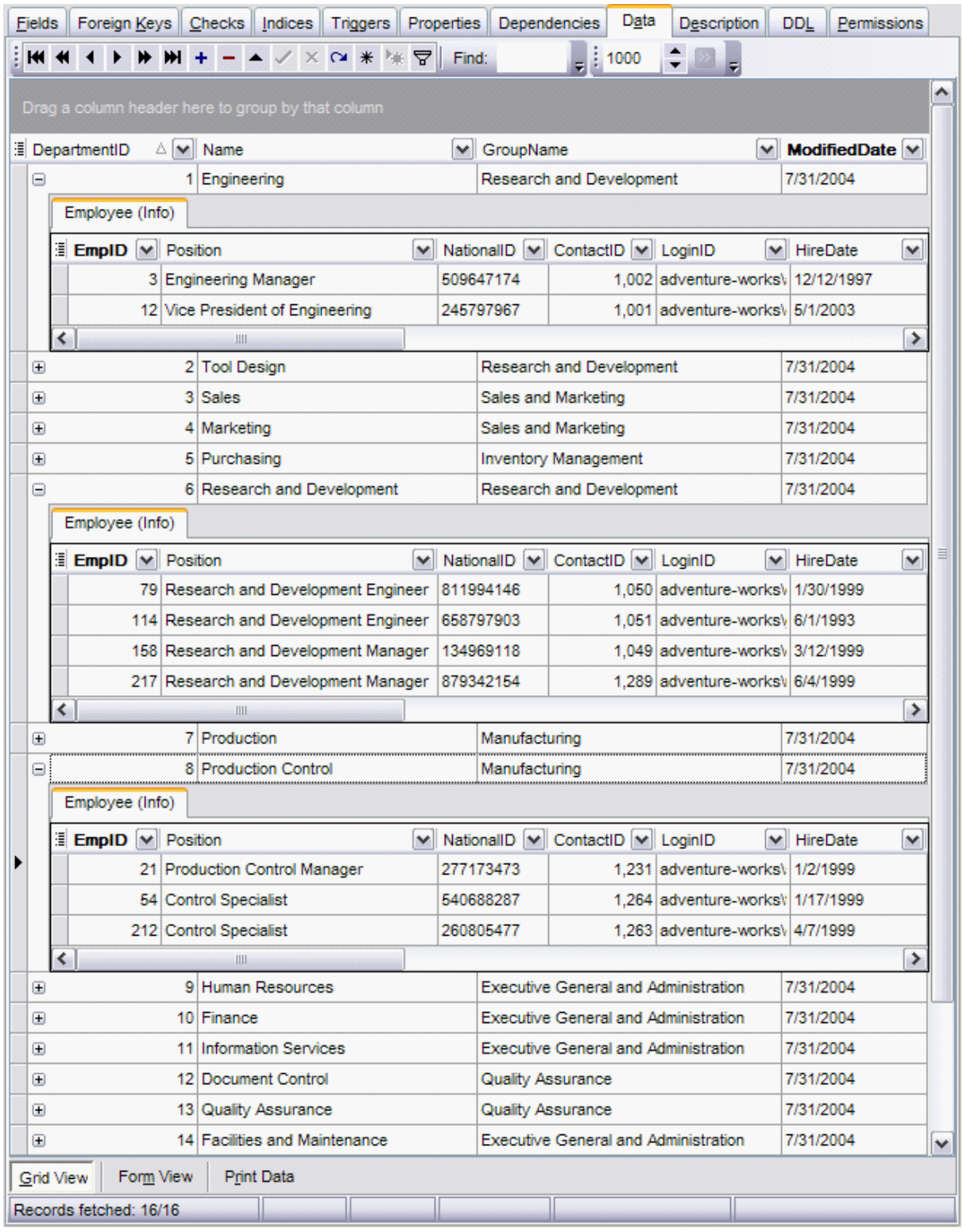
One of unique features of SQL Manager 2008 for SQL Server is the ability to work with data in multi-level mode to view and modify data in several related tables simultaneously.

© 2008 EMS Database Management Solutions

To manage grid levels, right-click the grid and select the **Grid Levels** context menu group. Items of this group allow you to:

- add a new grid level using Create Grid Level Wizard;
- delete the current grid level;
- switch between the ordinary Table View and the Card View modes.

© 2008 EMS Database Management Solutions



See also:

© 2008 EMS Database Management Solutions

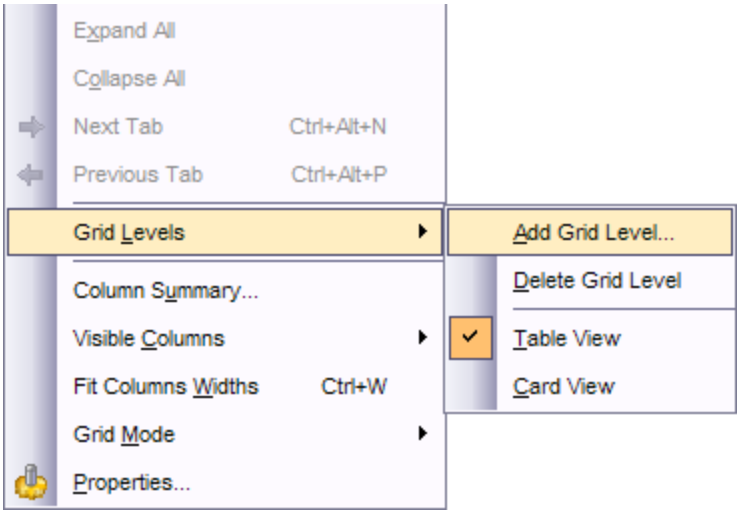
Grouping data
Filtering records
Using the context menu
Create Grid Level wizard
Working in card view mode
Column Summary

7.1.1.5 Create Grid Level wizard

Create Grid Level Wizard allows you to add a new detail level to the grid in order to get master-detail representation of your data.

To start the wizard, right-click the grid, select the **Grid Levels** context menu group and proceed to the **Add Grid Level...** item within this group.

- Specifying master level
- Selecting source table
- Binding master and detail levels
- Query parameterization
- Setting additional parameters



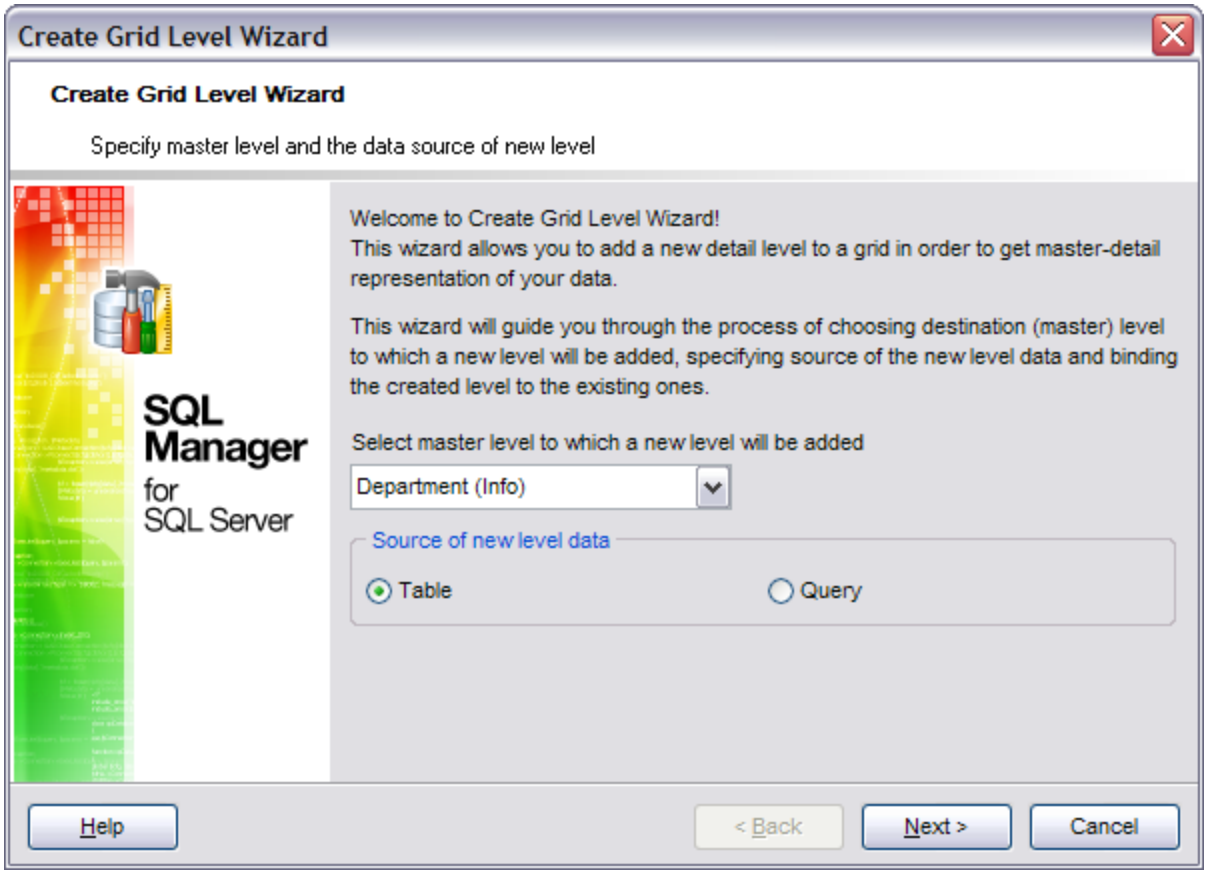
See also:
Grouping data
Filtering records
Using the context menu
Working in multi-level mode
Working in card view mode
Column Summary

© 2008 EMS Database Management Solutions

7.1.1.5.1 Specifying master level

Use the drop-down list to select the table of the **master level** to which a new level will be added.

Source of new level data
Select the source type of the new level data: *Table* or *Query*.

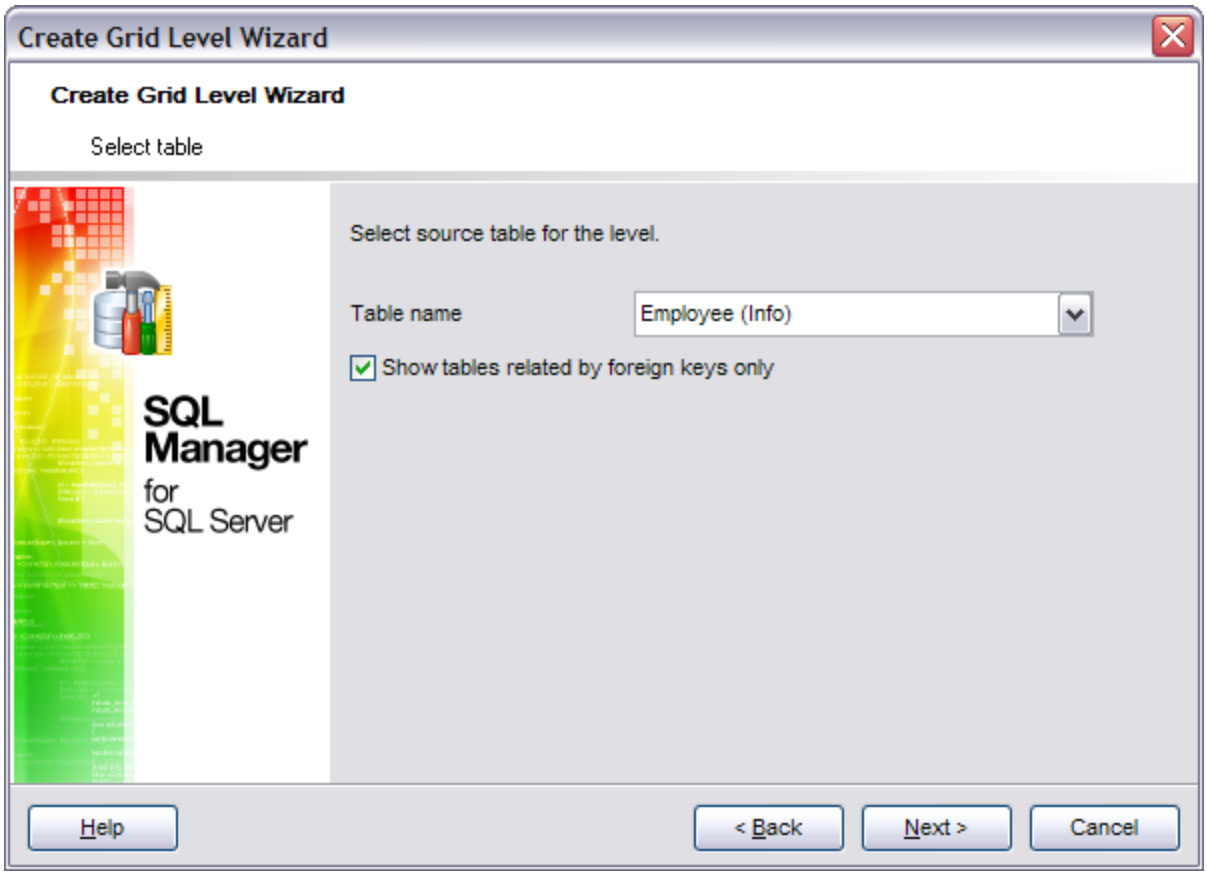


Click the **Next** button to proceed to the Defining source for detail level step to select a table for the detail level or input a query, depending on whether the **Table** or the **Query** option has been selected.

7.1.1.5.2 Defining source for detail level

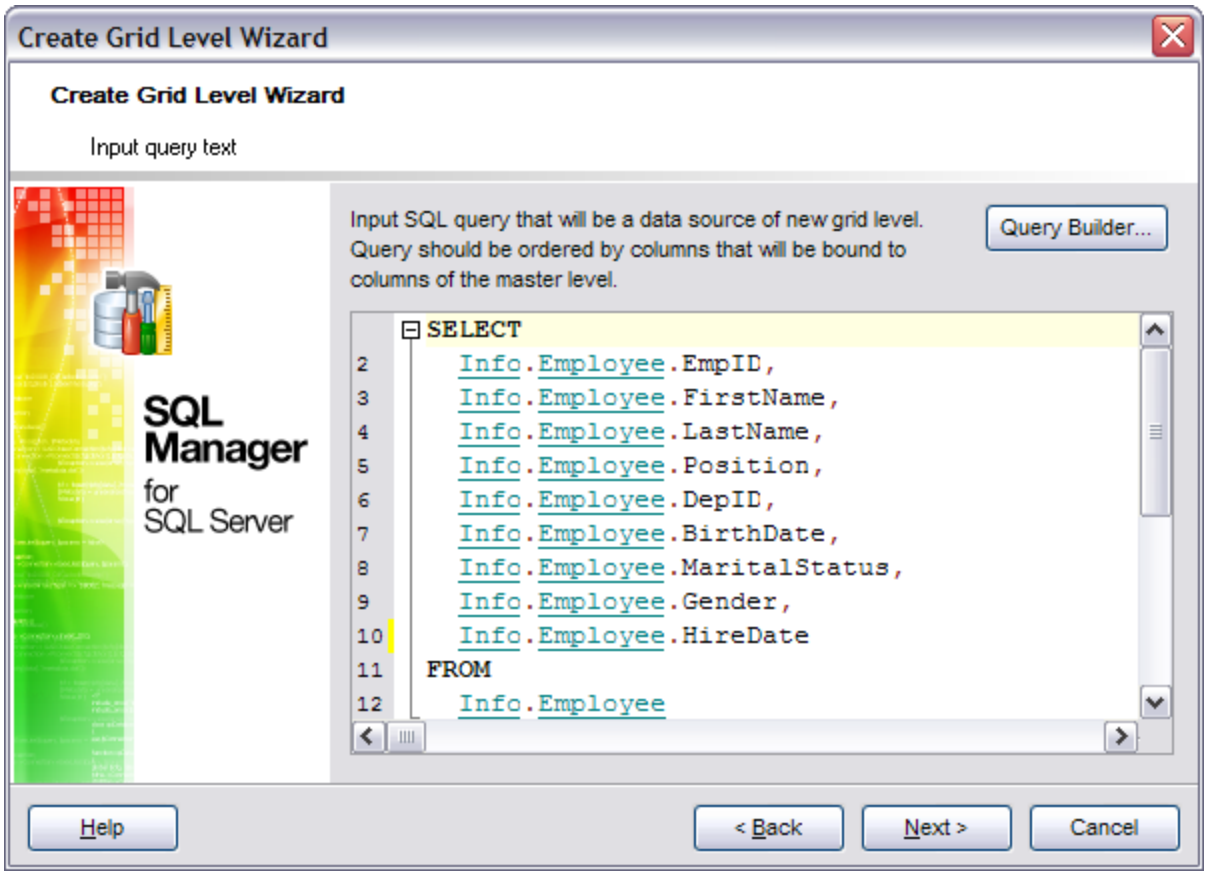
If the **Table** option has been selected at the previous step, you should now specify a table for the detail view using the **Table name** drop-down list. Set the **Show tables related by foreign keys only** option to narrow the list of tables by including only tables linked by Foreign keys.

© 2008 EMS Database Management Solutions



If the **Query** option has been selected at the previous step, you should now enter a query that will be used as the source of the new grid level. If necessary, you can use Query Builder to build the SQL query visually.

© 2008 EMS Database Management Solutions



Click the **Next** button to proceed to the Binding master and detail levels step of the wizard.

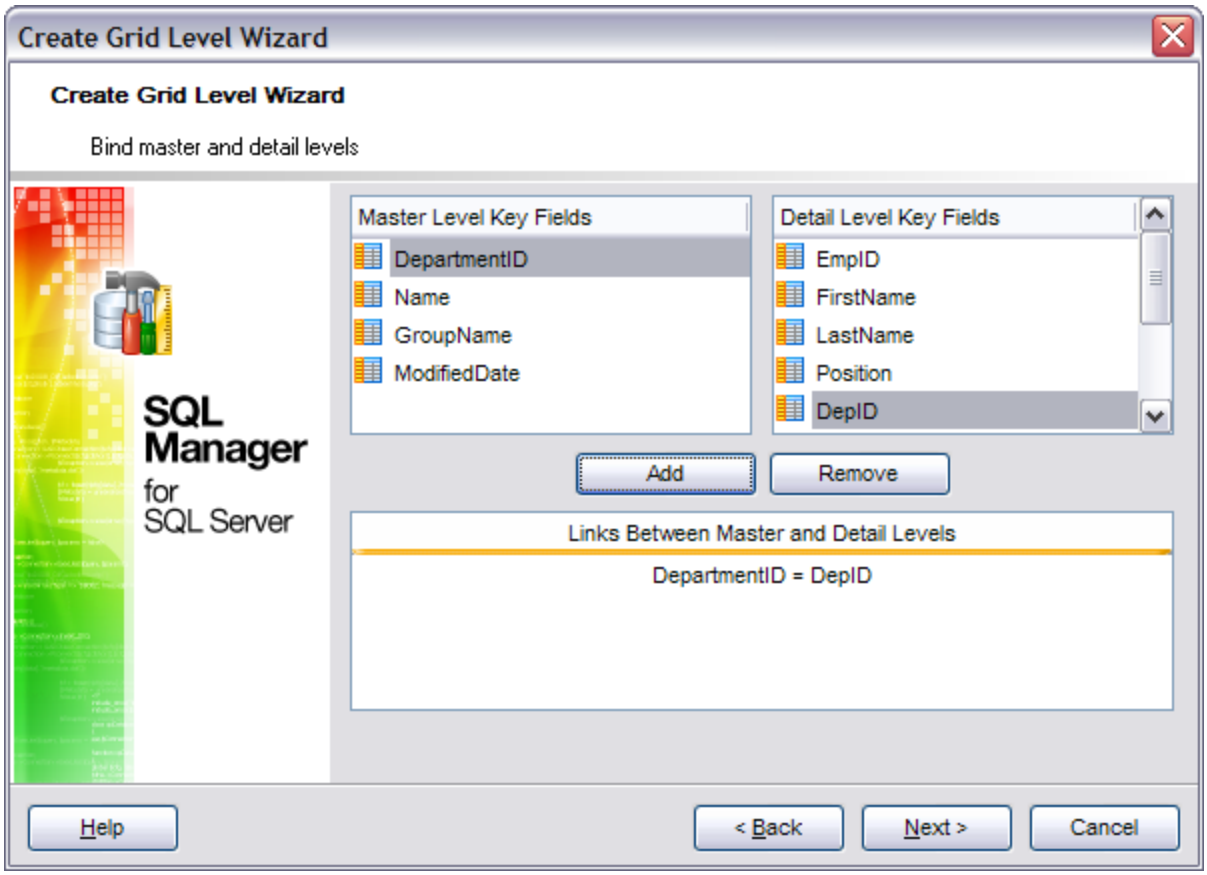
7.1.1.5.3 Binding master and detail levels

Define pairs of fields to link the Master Level and the Detail Level data sources:

- select a field in the **Master Level Key Fields** list;
- select a corresponding field in the **Detail Level Key Fields** list;
- click **Add** to set correspondence between the selected fields.

The newly created correspondences are listed in the **Links Between Master and Detail Levels** area. If necessary, you can delete any correspondence using the the **Remove** button.

© 2008 EMS Database Management Solutions

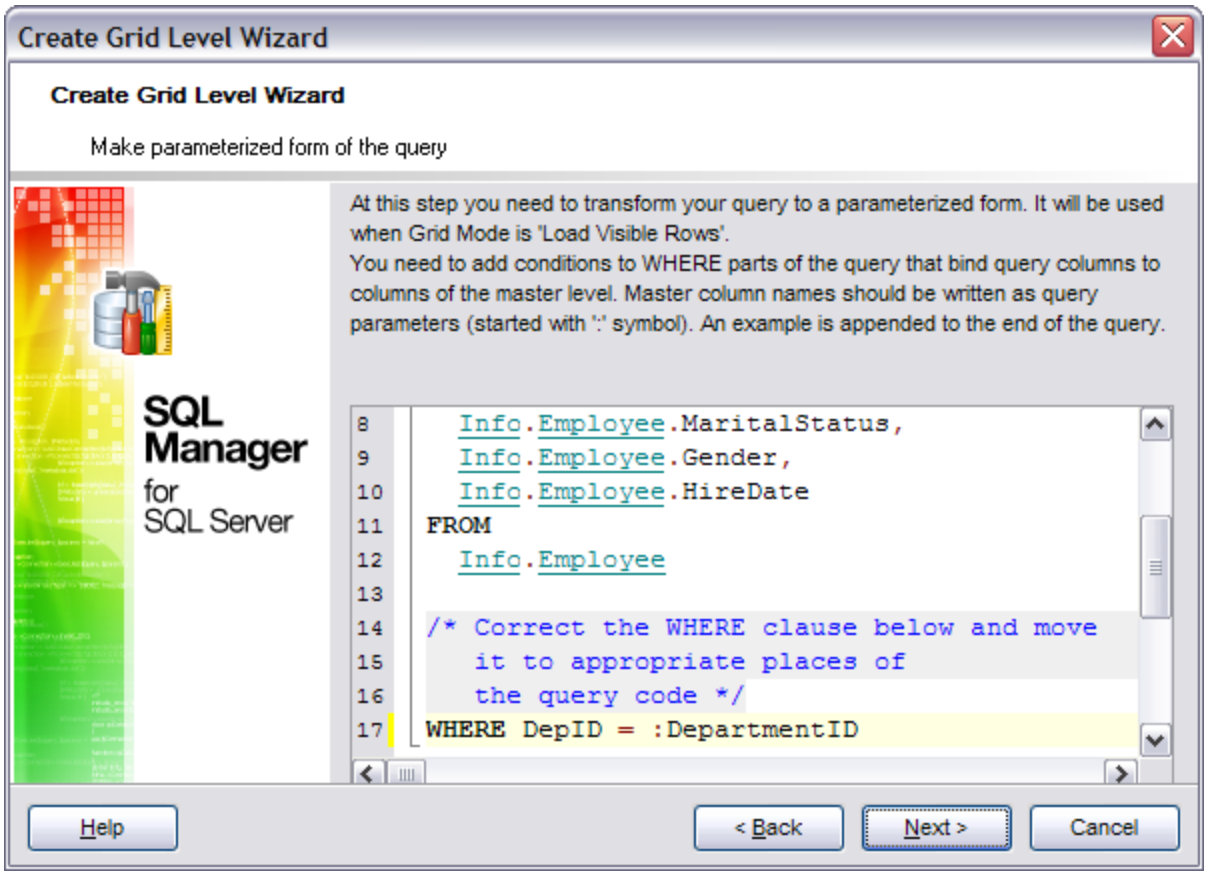


Click the **Next** button to proceed to the Setting additional parameters step or to the Query parameterization step of the wizard if **Query** was selected at the Specifying master level step of the wizard.

7.1.1.5.4 Query parameterization

If **Query** was selected at the Specifying master level step of the wizard, you should now transform the query to a parameterized form that will be used in the 'Load visible rows' Grid Mode (see the Grid | Data Options section of the Environment Options dialog to get more information about Grid modes offered by SQL Manager).

© 2008 EMS Database Management Solutions



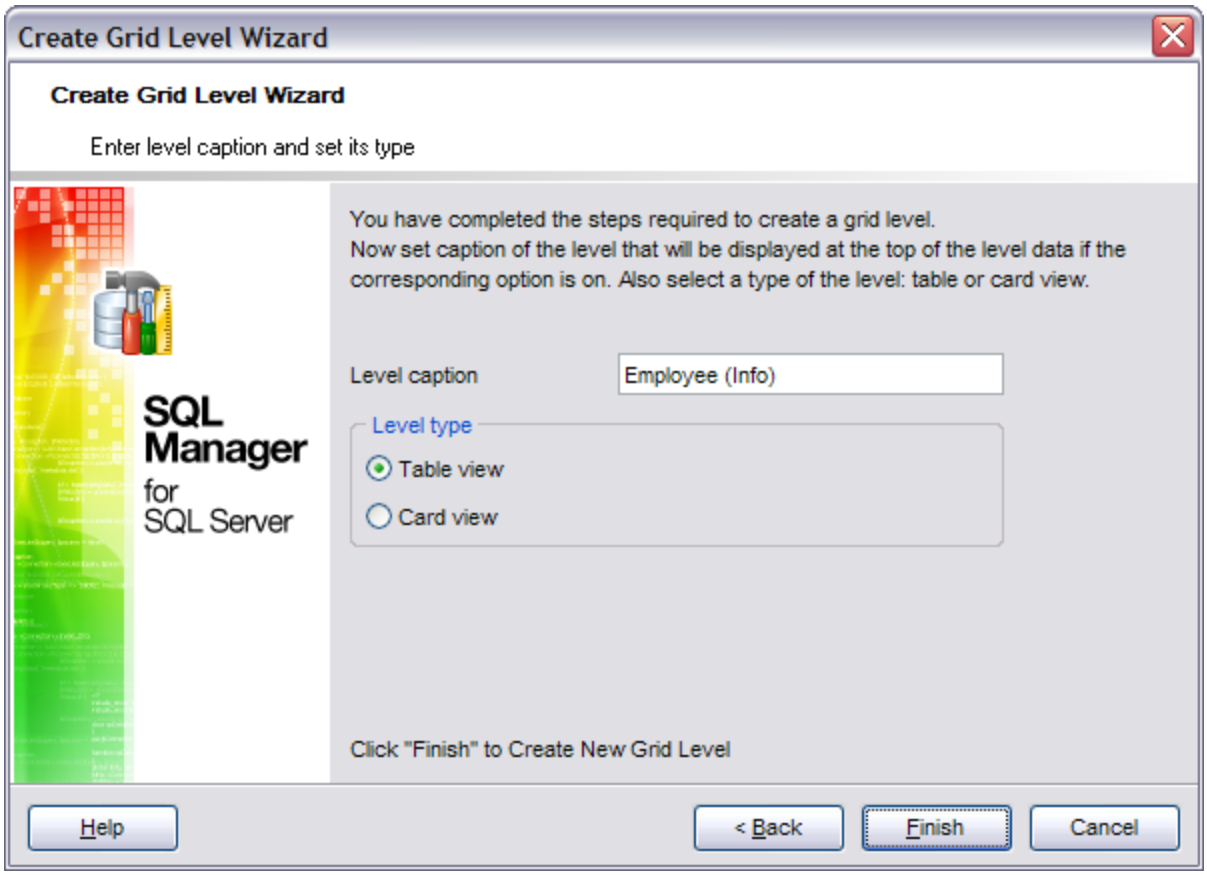
Click the **Next** button to proceed to the Setting additional parameters step of the wizard.

7.1.1.5.5 Setting additional parameters

Level caption
Set the caption to be used for the new level in the grid.

Level type
Select the type of view you wish to be applied to the grid level: *Table view* or *Card view*.

© 2008 EMS Database Management Solutions



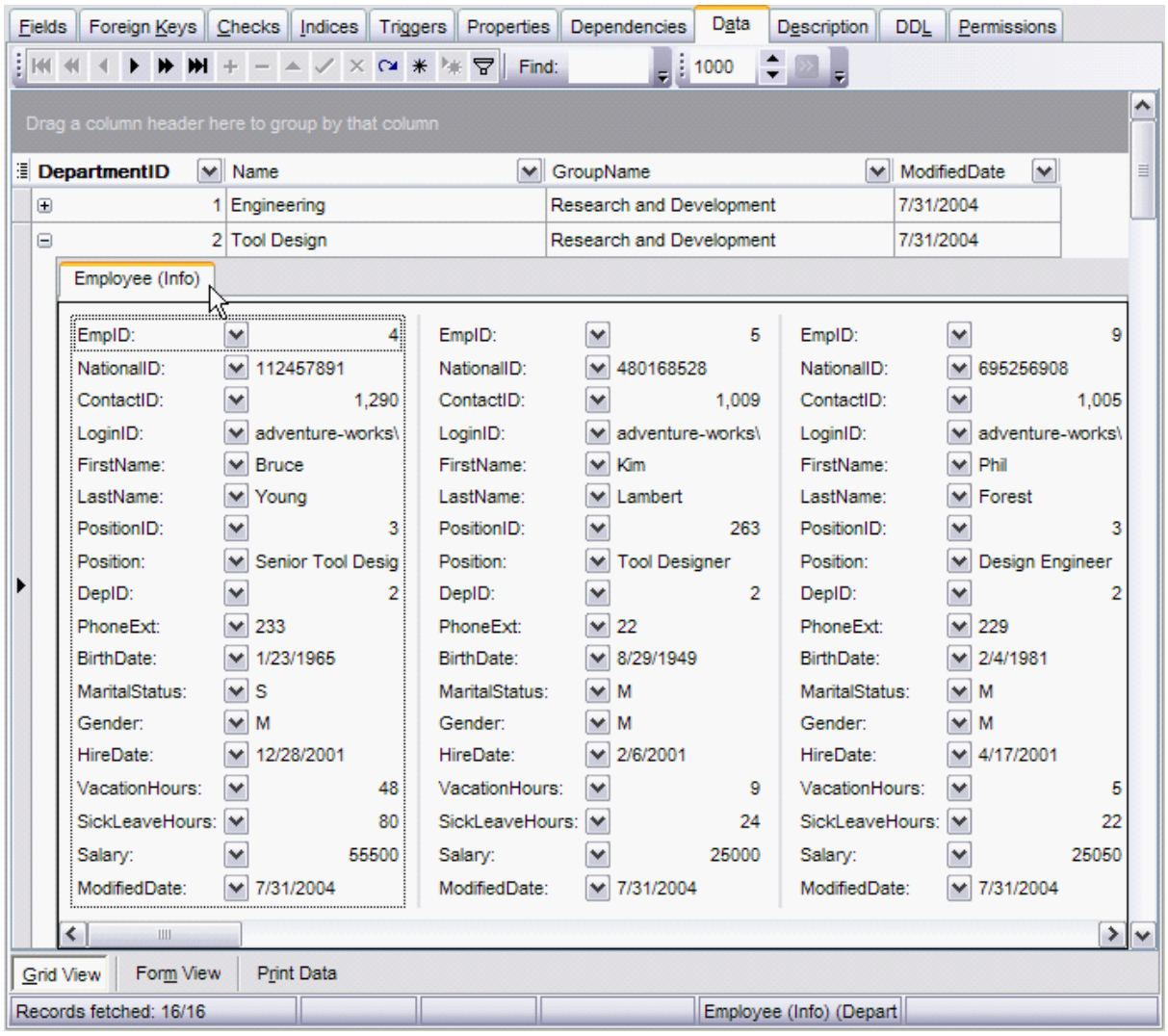
When you are done, click the **Finish** button to complete the operation.

7.1.1.6 Working in card view mode

Depending on your preferences, you can represent data in the **Table View** or in the **Card View** modes.

To switch to the **Card View** mode of data representation, right-click the grid, expand the **Grid Levels** context menu group and select the **Card View** item within this group.

© 2008 EMS Database Management Solutions



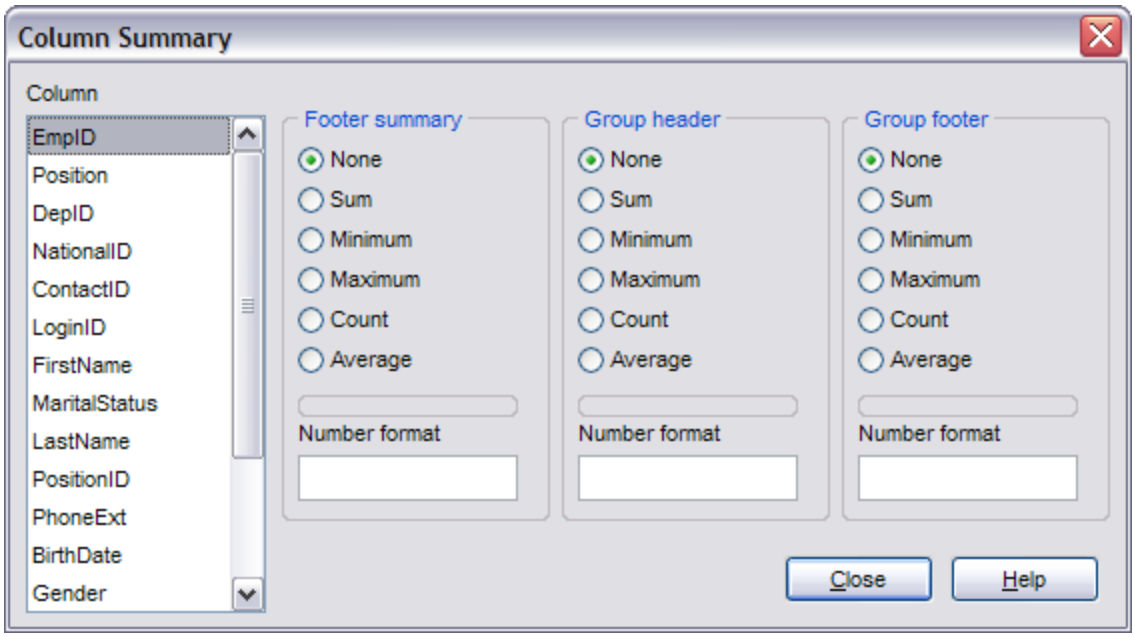
See also:

Grouping data
Filtering records
Using the context menu
Working in multi-level mode
Create Grid Level wizard
Column Summary

7.1.1.7 Column Summary

If necessary, you can select the **Column Summary...** context menu item to open the **Column Summary** dialog which allows you to view and edit the summary for each particular column.

© 2008 EMS Database Management Solutions



See also:

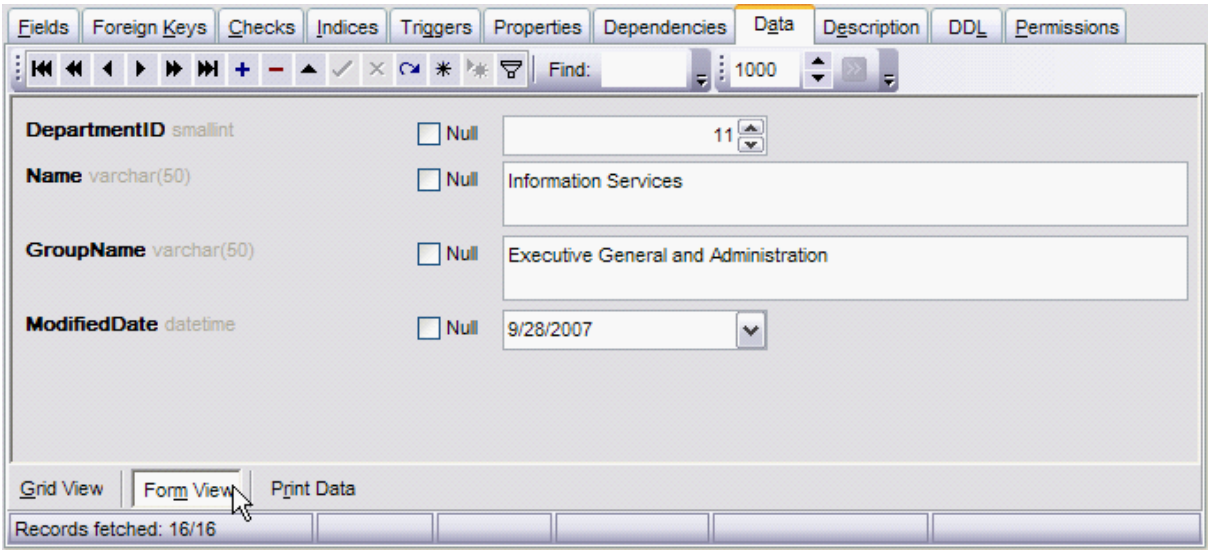
Grouping data
Filtering records
Using the context menu
Working in multi-level mode
Create Grid Level wizard
Working in card view mode

7.1.2 Form View

The **Form View** tab allows you to view data as a form. To activate this type of data view, select the **Form View** tab on the View mode panel at the bottom of the **Results** area of the window.

The form displays the current record: field names on the left and the corresponding values on the right. If the fields are available for editing, you can edit the record directly on this form. The navigation pane at the top of the form allows you to browse the data quickly, to insert, update and delete records, and to set a filter for the records using the Filter Builder dialog.

© 2008 EMS Database Management Solutions



Each field has a **Null** checkbox which allows you to clear the field value and set it to NULL (if the field is nullable).

Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**Yes**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

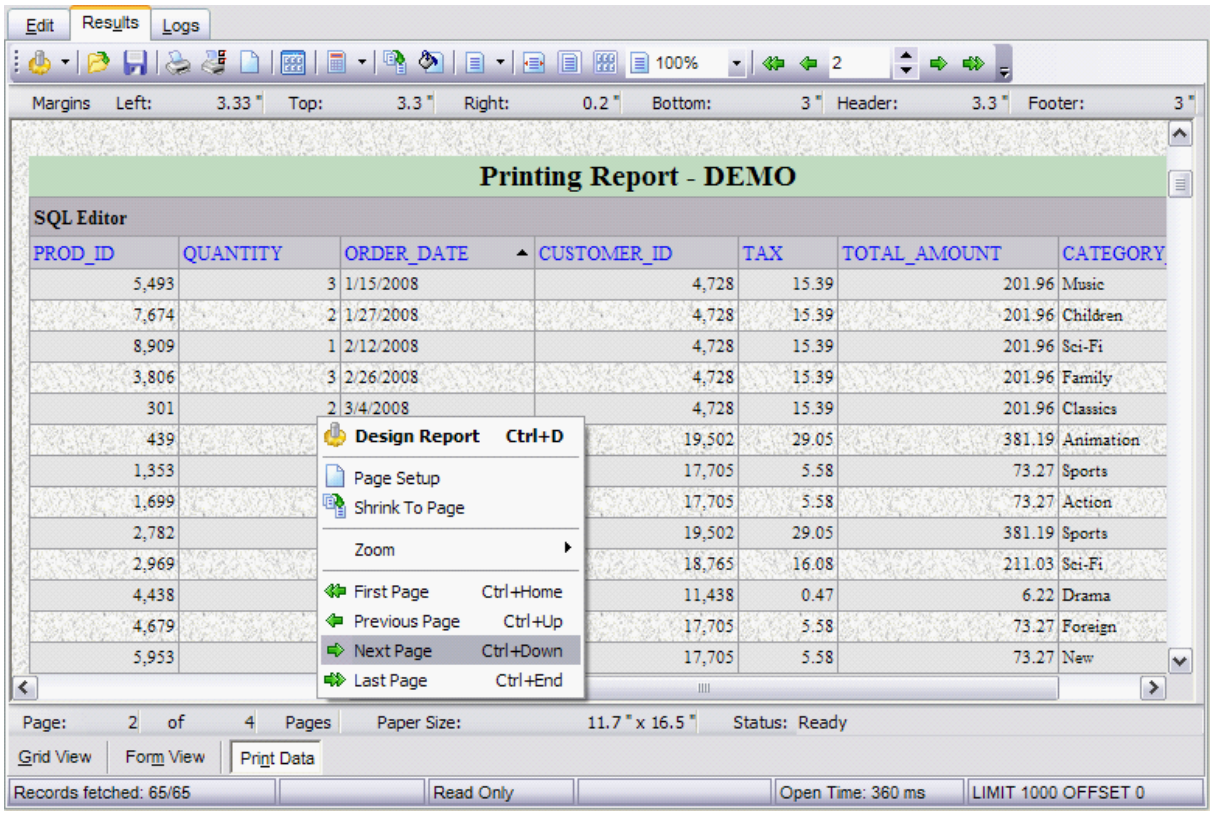
Grid View
Print Data
BLOB View

7.1.3 Print Data

Using the **Print Data** tab you can view data in the way they are printed, in WYSIWYG mode.

When in **Print Data** mode, you are provided with a powerful toolbar allowing you to design a report, change the view scope, save reports and load previously saved ones, and set a number of page printing parameters.

© 2008 EMS Database Management Solutions



Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**No**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

Grid View
Form View
BLOB View

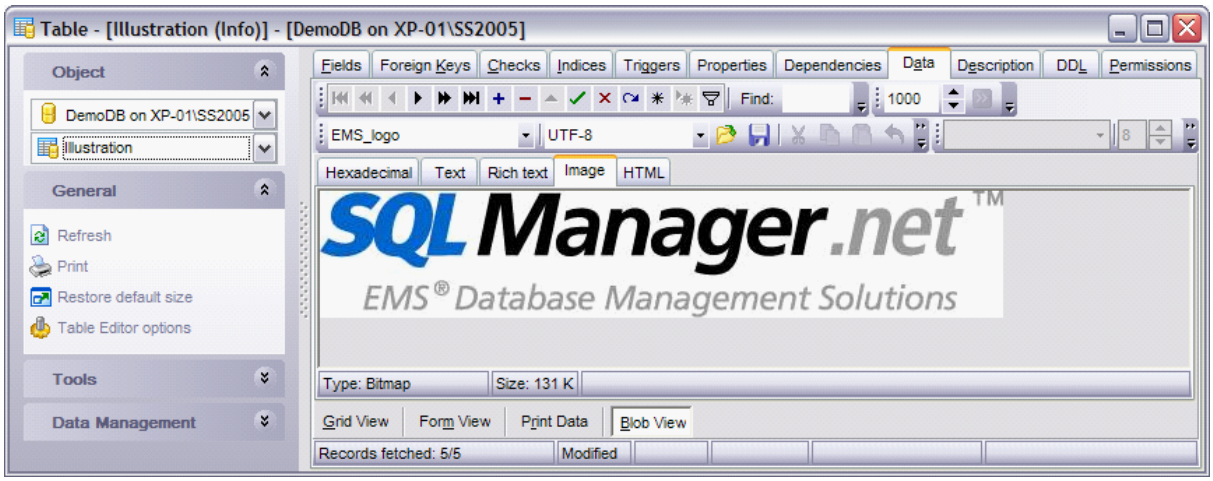
7.1.4 BLOB View

SQL Manager 2008 for SQL Server provides BLOB Viewer/Editor to view and edit BLOB fields content. The BLOB Viewer/Editor can be invoked from the data grid within Table Editor, SQL Editor, Visual Query Builder, etc.

- Navigation within the BLOB Viewer/Editor
- Viewing/Editing BLOB field as Hexadecimal dump

© 2008 EMS Database Management Solutions

- Viewing/Editing BLOB field as plain Text
- Viewing/Editing BLOB field as Rich Text (RTF)
- Viewing/Editing BLOB field as Image
- Viewing/Editing BLOB field as HTML
- Applying changes



Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**No**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

Grid View
Form View
Print Data

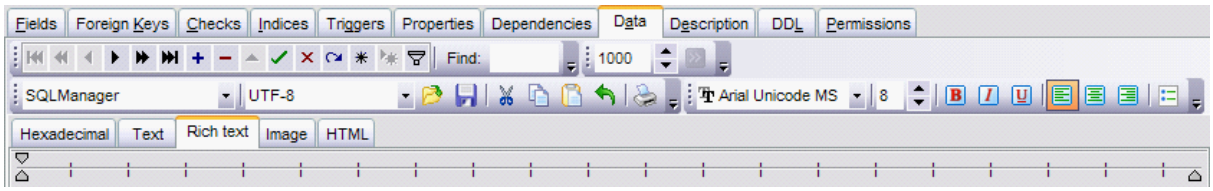
7.1.4.1 Navigation within BLOB Editor

The **BLOB Viewer/Editor** provides an ability to navigate within the records using **DB Navigation** buttons on the navigation pane at the top of the viewer window.

Using items of the navigation pane and the drop-down menu you can to browse the data quickly, to insert, update and delete records, set a filter for the records using the Filter Builder dialog, load new BLOB content and save the current content to files.

The toolbar allows you to switch the fields and perform a number of editing operations. The set of toolbar items depends on the current selection and view mode.

© 2008 EMS Database Management Solutions



See also:

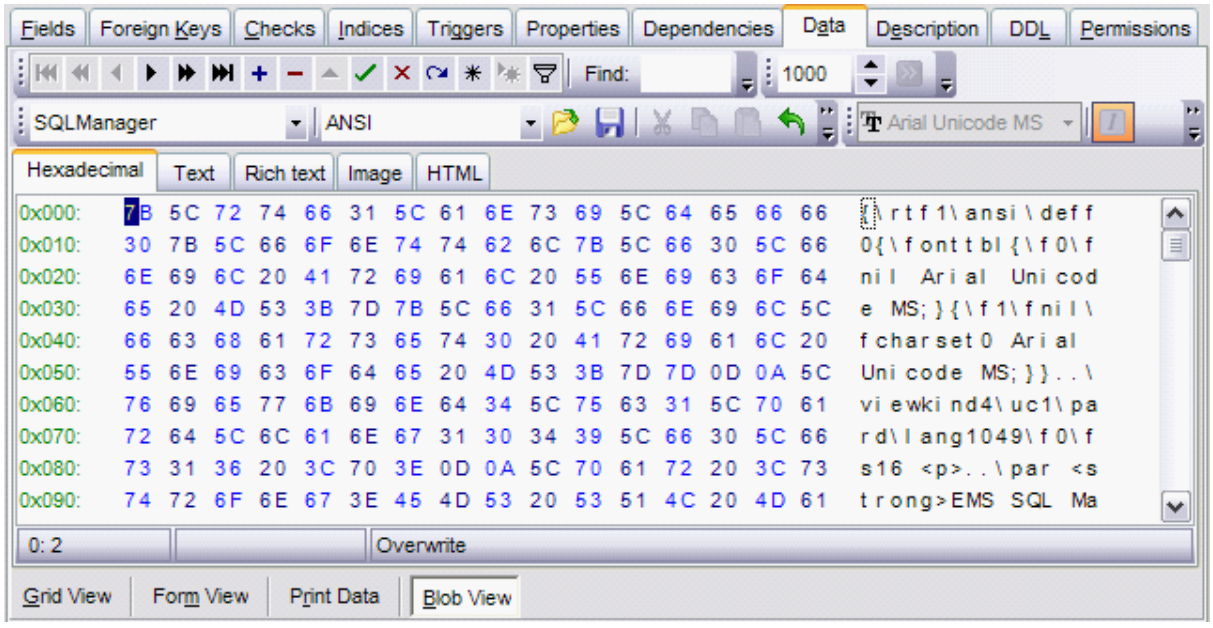
Editing as Hexadecimal
Editing as Text
Editing as Rich Text
Editing as Image
Editing as HTML
Applying changes

7.1.4.2 Editing as Hexadecimal

The **Hexadecimal** tab allows you to view/edit the BLOB data as hexadecimal.

The toolbar provides additional functionality for BLOB Viewer/Editor: use the **Save to file** and the **Load from file** toolbar buttons to save the hexadecimal data to a file, or load data from a file.

Use the *Ins* key to switch between the Insert and Overwrite modes.



© 2008 EMS Database Management Solutions

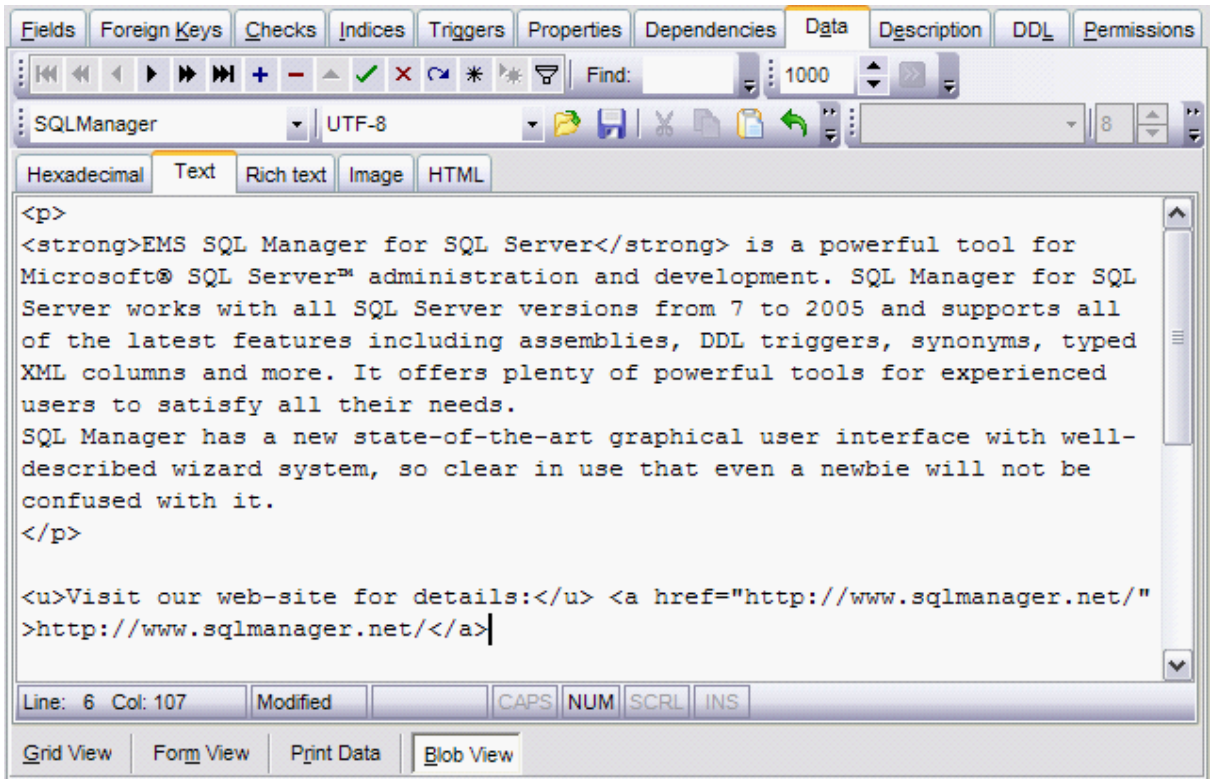
See also:

Navigation within BLOB Editor
Editing as Text
Editing as Rich Text
Editing as Image
Editing as HTML
Applying changes

7.1.4.3 Editing as Text

The **Text** tab allows you to view/edit the BLOB data as simple text.

The toolbar provides additional functionality for BLOB Viewer/Editor: use the **Save to file** and the **Load from file** toolbar buttons to save the text to a **.txt* file, or load text from a file. Additionally, you can use the *Cut*, *Copy*, *Paste*, *Select All*, *Undo*, *Word Wrap* context menu items for editing the text efficiently, and the **Print** context menu item to print the content of the **Text** tab.



See also:

Navigation within BLOB Editor
Editing as Hexadecimal

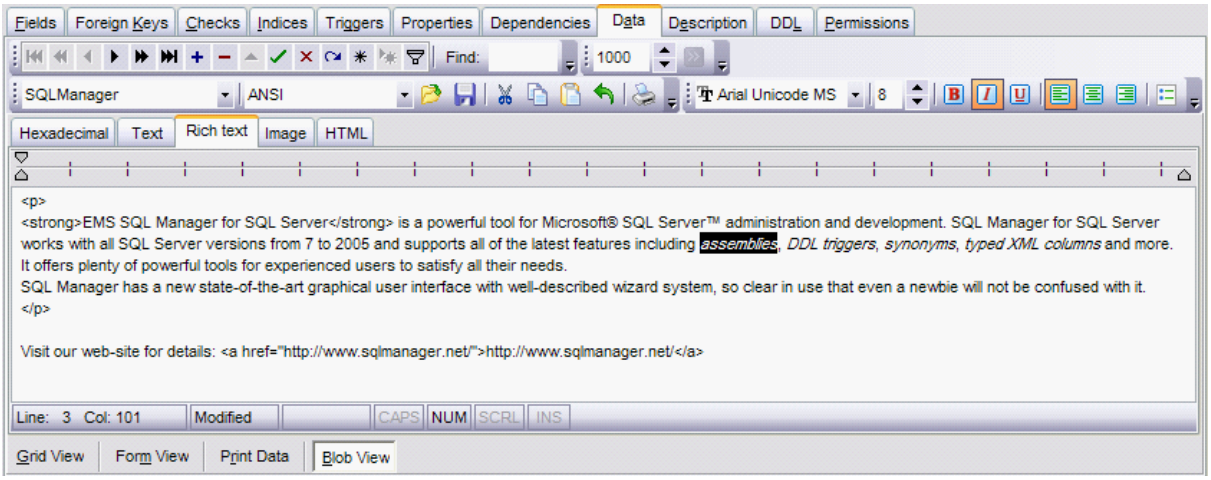
© 2008 EMS Database Management Solutions

Editing as Rich Text
Editing as Image
Editing as HTML
Applying changes

7.1.4.4 Editing as Rich Text

The **Rich Text** tab allows you to view/edit the BLOB data in Rich Text format (RTF).

The toolbar provides additional functionality for BLOB Viewer/Editor: use the **Save to file** and the **Load from file** toolbar buttons to save the Rich Text to a **.rtf* file, or load text from a file. Additionally, you can use the *Cut*, *Copy*, *Paste*, *Select All*, *Undo* context menu items for editing the text efficiently, and the **Print** context menu item to print the content of the **Rich Text** tab.



See also:

Navigation within BLOB Editor
Editing as Hexadecimal
Editing as Text
Editing as Image
Editing as HTML
Applying changes

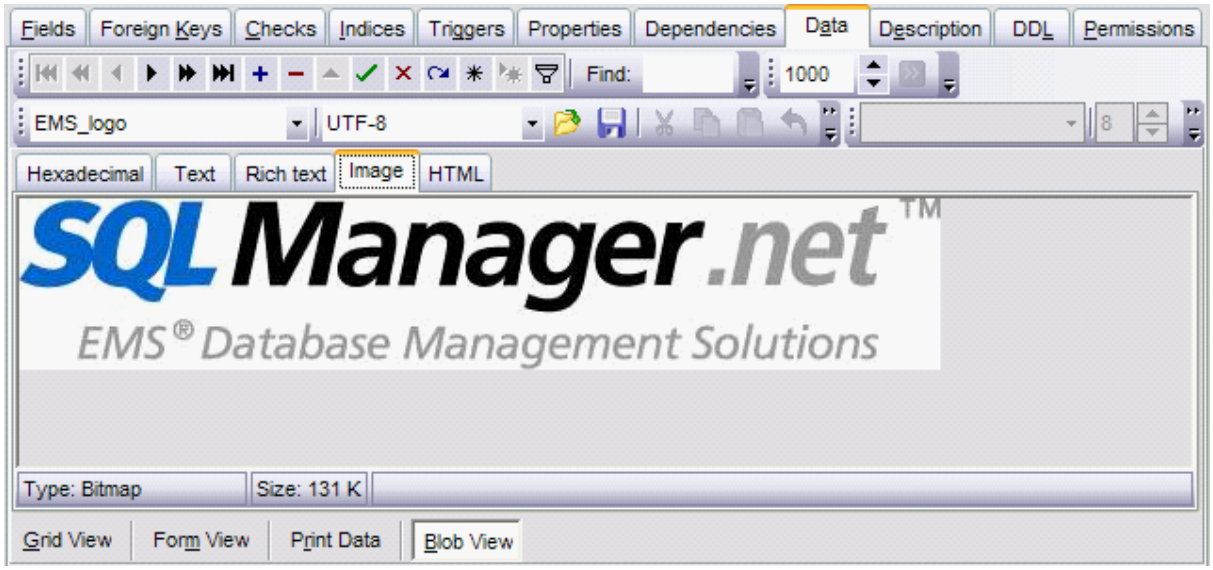
7.1.4.5 Editing as Image

The **Image** tab allows you to view the BLOB data as image.

The toolbar provides additional functionality for BLOB Viewer/Editor: use the **Save to file** and the **Load from file** toolbar buttons to save the image to a **.bmp*, **.wmf*, **.ico*

© 2008 EMS Database Management Solutions

or *.jpg file, or load an image from a file.



See also:

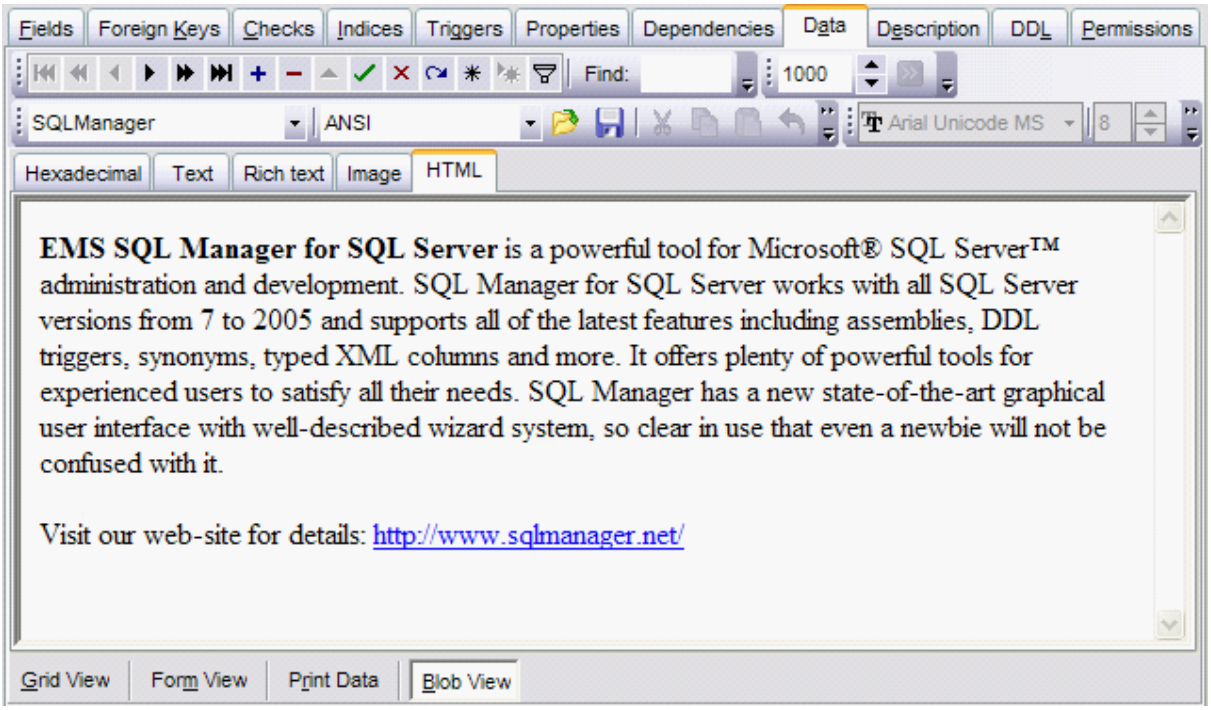
Navigation within BLOB Editor
Editing as Hexadecimal
Editing as Text
Editing as Rich Text
Editing as HTML
Applying changes

7.1.4.6 Editing as HTML

The **HTML** tab allows you to view the BLOB data as HTML (Hyper-Text Markup Language format) - in the way this data would be displayed by your Internet browser.

The toolbar provides additional functionality for BLOB Viewer/Editor: use the **Save to file** and the **Load from file** toolbar buttons to save the content as a *.html, or *.htm file, or load content from a file.

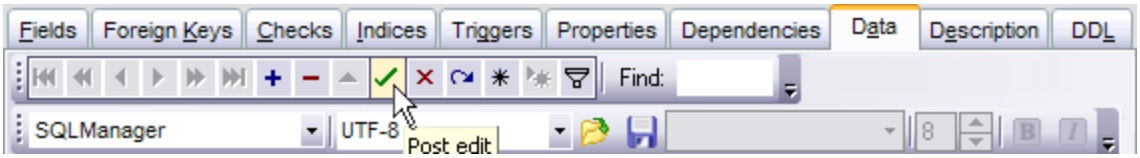
© 2008 EMS Database Management Solutions



See also:
Navigation within BLOB Editor
Editing as Hexadecimal
Editing as Text
Editing as Rich Text
Editing as Image
Applying changes

7.1.4.7 Applying changes

After changes are done, click the **Post Edit** button on the navigation pane to apply the changes or the **Cancel Edit** button to discard the changes.



See also:
Navigation within BLOB Editor
Editing as Hexadecimal

© 2008 EMS Database Management Solutions

Editing as Text
Editing as Rich Text
Editing as Image
Editing as HTML

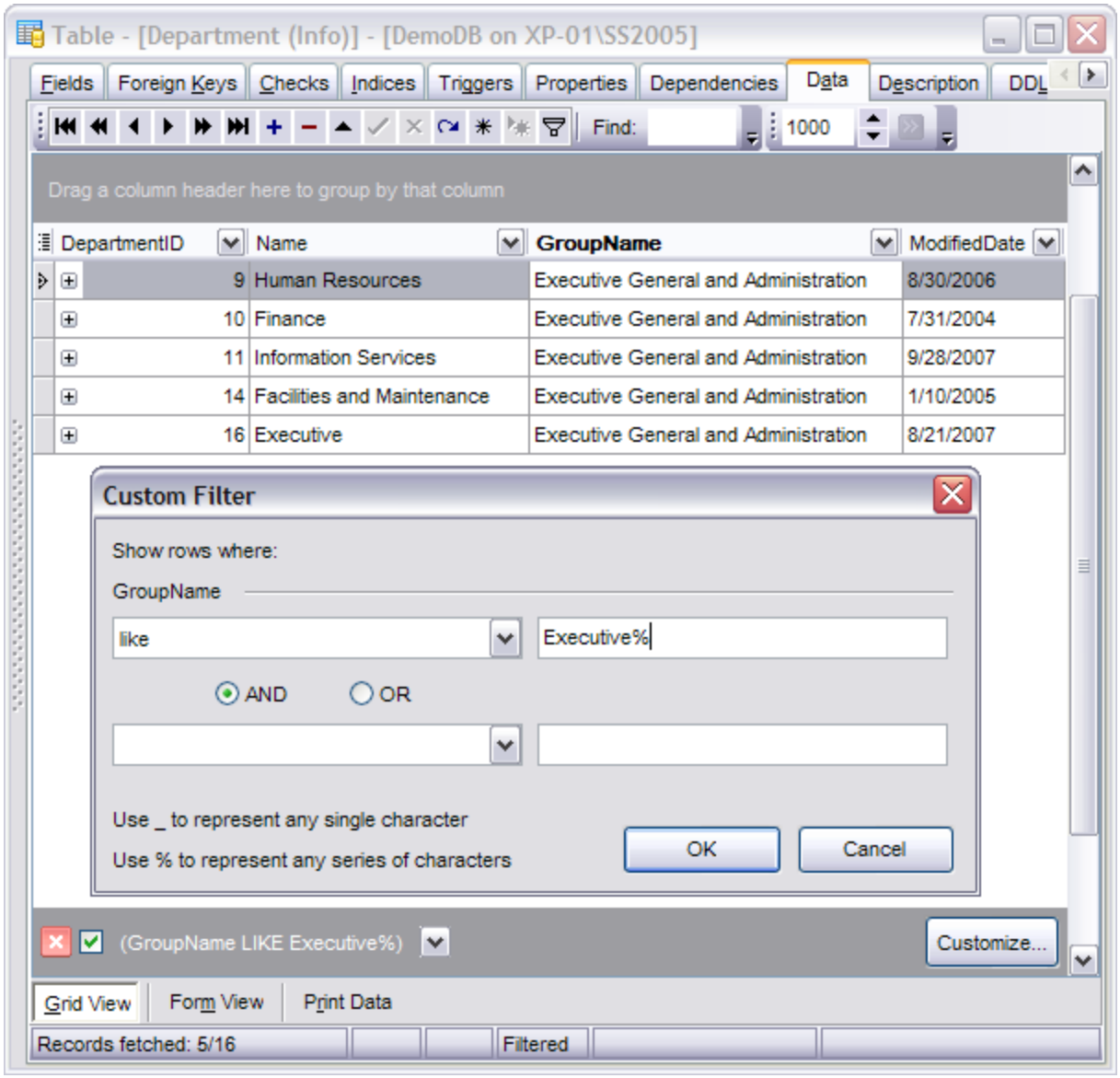
© 2008 EMS Database Management Solutions

7.2 Custom Filter

The **Custom Filter** dialog is one of the filtering facilities implemented in Data View for your convenience.

To open the dialog, click the Arrow-Down button next to the column caption, and select the **Custom** item from the drop-down list.

Select a logical operator for checking the column values (*like*, *is less than*, *is greater than* , etc.) and set a value to be checked by this operator in the corresponding box on the right.



If necessary, you can set the second condition and specify the relation between the two conditions: whether both of them should be satisfied (*AND*) or just any of them (*OR*). Use

© 2008 EMS Database Management Solutions

the '_' character to represent any single symbol, and use the '%' character to represent any series of symbols in the condition string.

See also:

Data View

Filter Builder Dialog

© 2008 EMS Database Management Solutions

7.3 Filter Builder dialog

The **Filter Builder** dialog is a powerful filtering tool implemented in Data View for your convenience.

The dialog is aimed at facilitating the procedure of creating and applying complex filter criteria for data. In addition, the tool allows you to save filter criteria to an external **.flt* file for future use.

To open the **Filter Builder** dialog, use the **Set filter** button on the navigation pane available within the Data tab of Table Editor and the **Result(s)** tabs of SQL Editor and Query Builder.

- Invoking the Filter Builder dialog
- Adding a new condition to the filter
- Setting filter criteria
- Setting filter operator
- Setting filter criteria values
- Adding a new group
- Setting group operator
- Applying filter conditions

Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**Yes**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

Data View

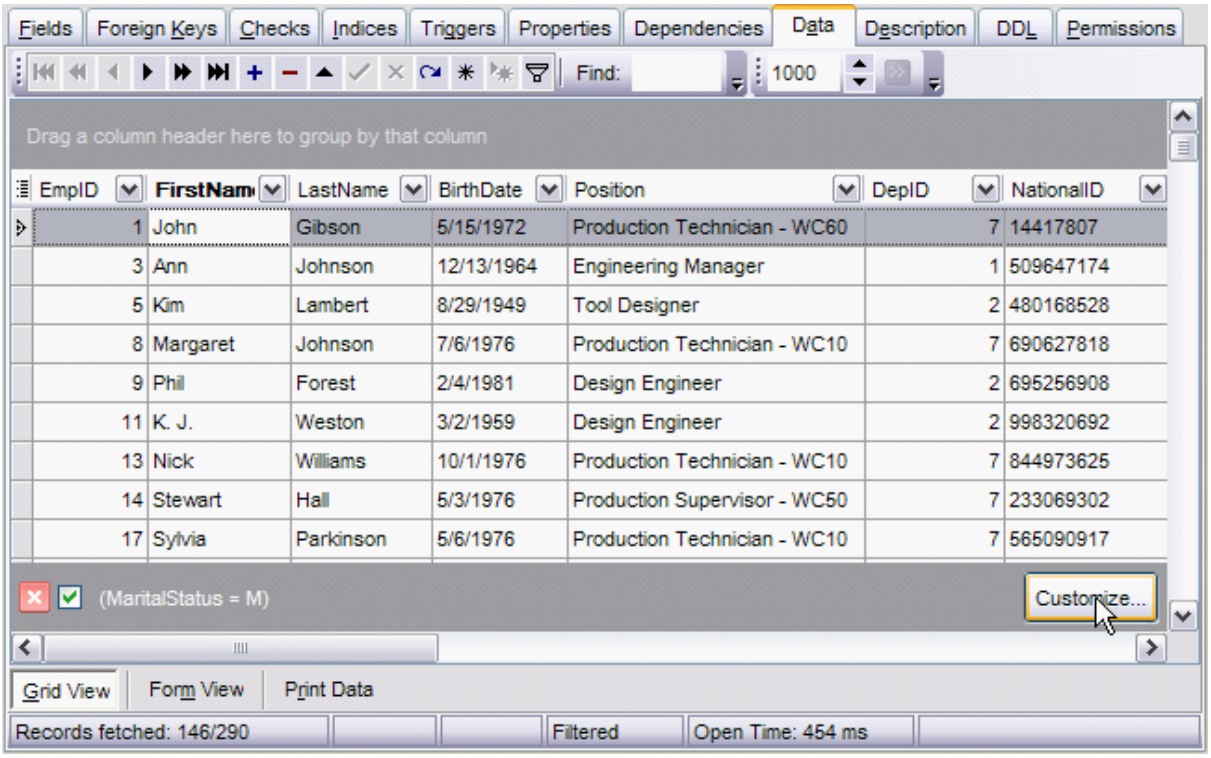
Custom Filter

7.3.1 Invoking the Filter Builder dialog

The **Filter Builder** dialog can be invoked in either of the following ways:

- if a simple filter or the Custom Filter is being used, click the **Customize...** button on the gray **filtering panel**;

© 2008 EMS Database Management Solutions



- use the **Set filter** button on the navigation pane and create a composite filter using the dialog.

The succeeding pages of this chapter are intended to illustrate usage of the **Filter Builder** dialog. Please see the instructions below to learn how to perform various operations in the easiest way.

See also:

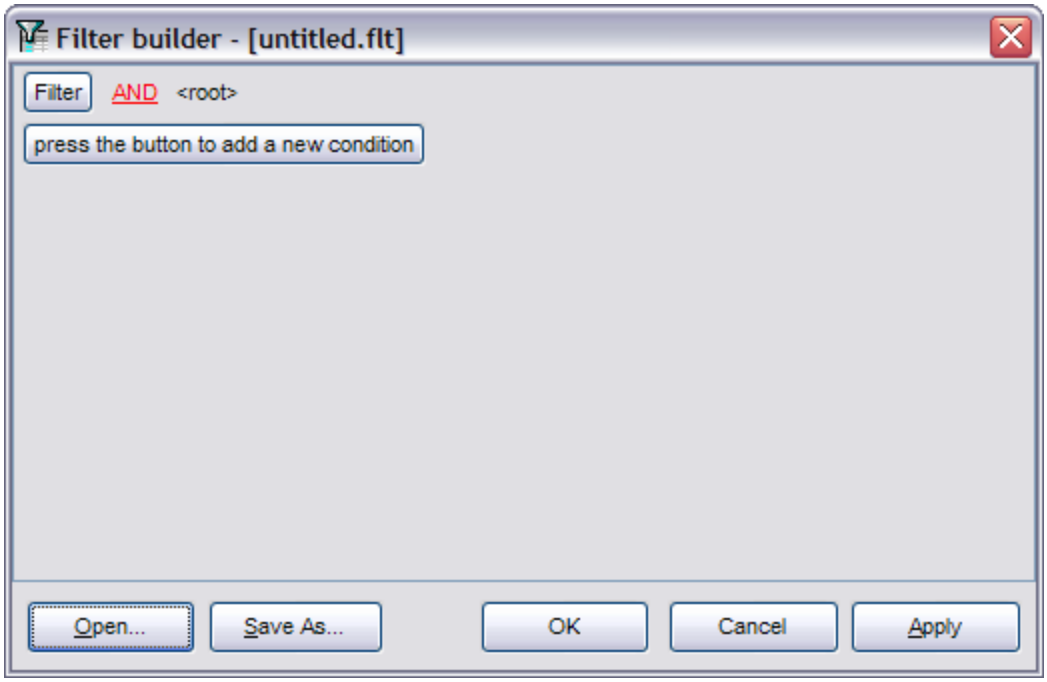
Adding a new condition
Setting filter criteria
Setting filter operator
Setting filter criteria values
Adding a new group
Setting group operator
Applying filter conditions

© 2008 EMS Database Management Solutions

7.3.2 Adding a new condition

Suppose we need to select data from the sample table *Employee* to view the list of male engineers belonging to the *Engineering* and *Tool Design* departments that were hired after *10/1/2007*. These criteria are applied to the *Gender*, *HireDate*, *Position* and the *DepID* fields.

Click **press the button to add a new condition** - this will add a new condition to the criteria. Alternatively, you can click the **Filter** button and select the **Add Condition** popup menu item.



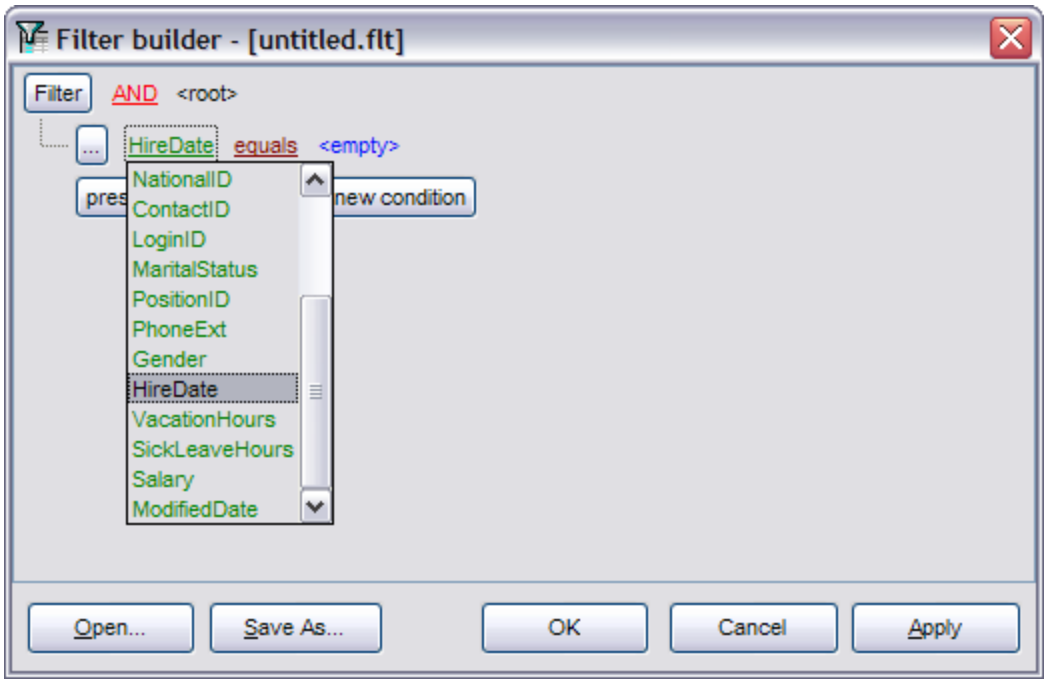
See also:

Invoking the Filter Builder dialog
Setting filter criteria
Setting filter operator
Setting filter criteria values
Adding a new group
Setting group operator
Applying filter conditions

7.3.3 Setting filter criteria

As we need to apply the filter criteria to the *HireDate* field, we click the column box (next to the ellipsis button) to open the drop-down list displaying the available column names and select the *HireDate* item.

© 2008 EMS Database Management Solutions



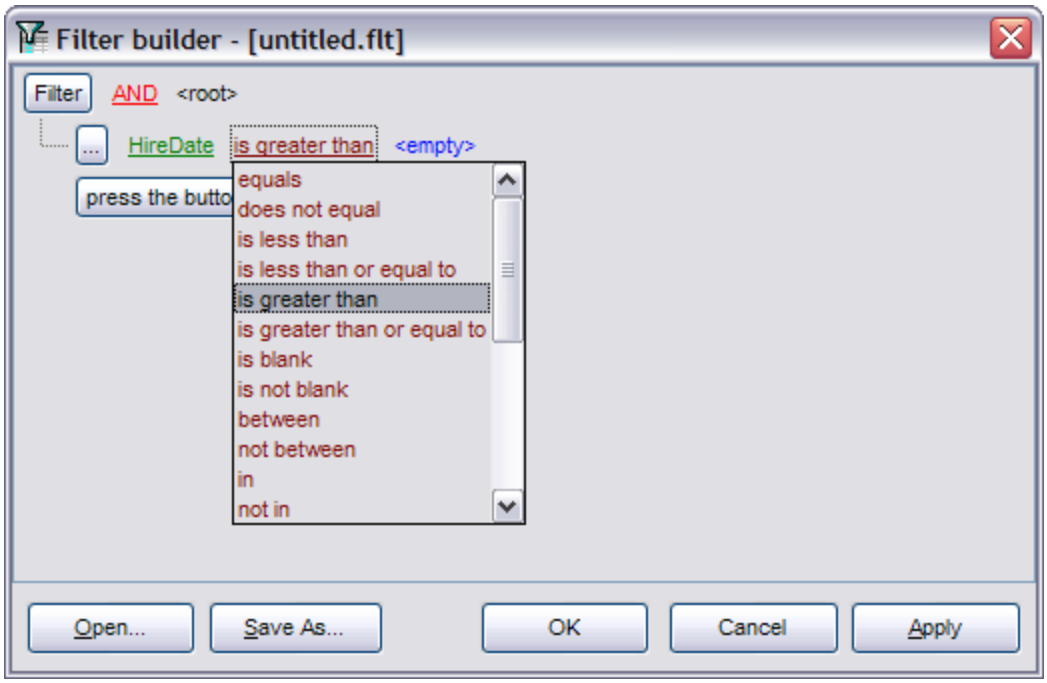
See also:

Invoking the Filter Builder dialog
Adding a new condition
Setting filter operator
Setting filter criteria values
Adding a new group
Setting group operator
Applying filter conditions

7.3.4 Setting filter operator

Since we need the list of employees hired after *10/1/2007*, we need to select the *IS GREATER THAN* operator from the corresponding drop-down list.

© 2008 EMS Database Management Solutions



See also:
Invoking the Filter Builder dialog
Adding a new condition
Setting filter criteria
Setting filter criteria values
Adding a new group
Setting group operator
Applying filter conditions

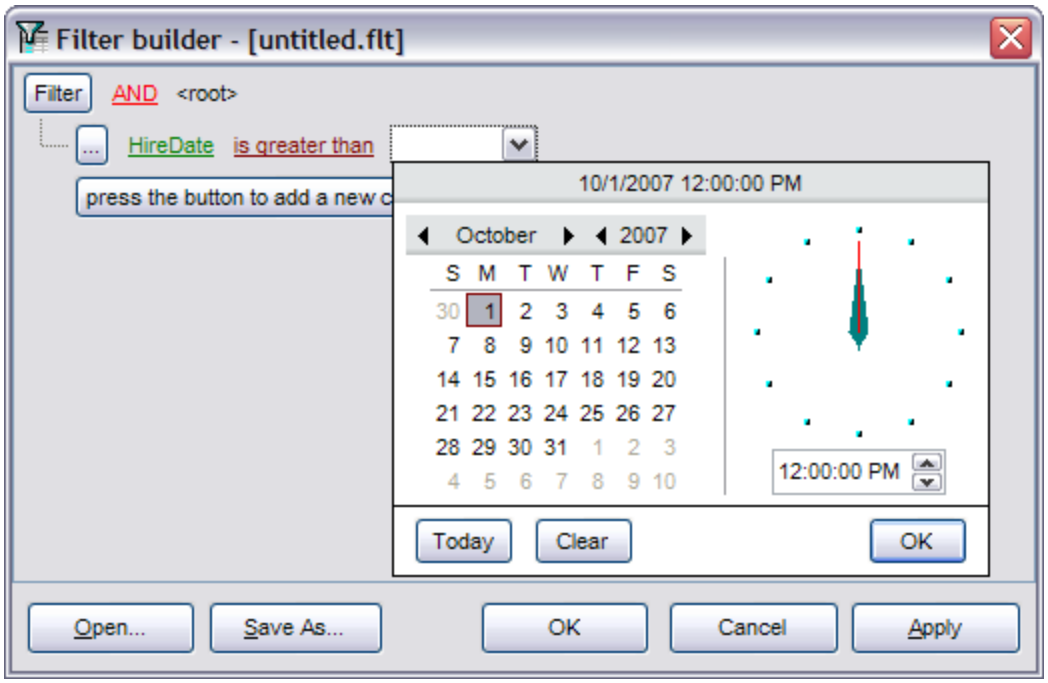
7.3.5 Setting filter criteria values

Next, we need to specify value *'10/1/2007'* for the *IS GREATER THAN* operator.

Similarly, if, for example, we need to get the list of employees hired during the *9/1/2007 - 10/1/2007* term, we set the *BETWEEN* filter operator (this will add two empty value boxes to specify the inclusive range for the *BETWEEN* condition) and specify the range for the operator, i.e. the *'9/1/2007'* and the *'10/1/2007'* values in the corresponding value boxes.

It is possible to set the date value **manually** by typing it in, or via the **date editor** which is activated when you click the value box.

© 2008 EMS Database Management Solutions



Editors used in value boxes are determined by the **data type** assigned to the corresponding columns.

See also:

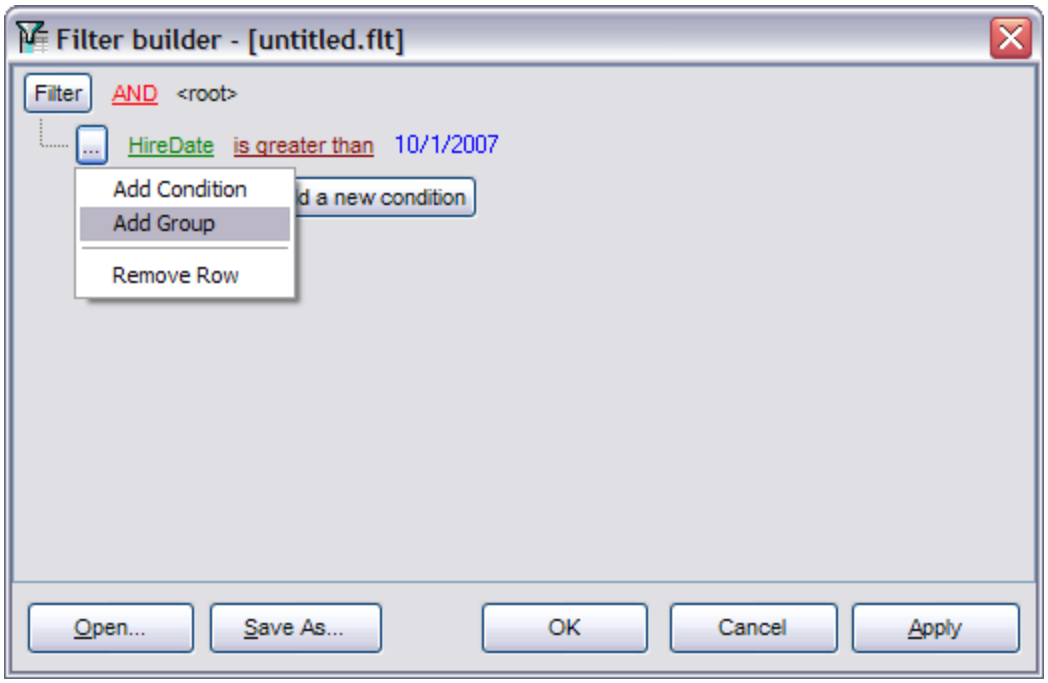
Invoking the Filter Builder dialog
Adding a new condition
Setting filter criteria
Setting filter operator
Adding a new group
Setting group operator
Applying filter conditions

7.3.6 Adding a new group

Since we also need to get the list of male specialists-engineers (i.e. those registered in the *Engineering* and *Tool Design* departments and having an engineering-oriented position), we can add a complex filter condition combining simple conditions with the *AND* operator. (However, in this particular case we can just add them on the same root level as for the existing condition).

If you need to add a group of conditions, click the ellipsis button for the *HIRE_DATE* condition and select the **Add Group** popup menu item.

© 2008 EMS Database Management Solutions



See also:

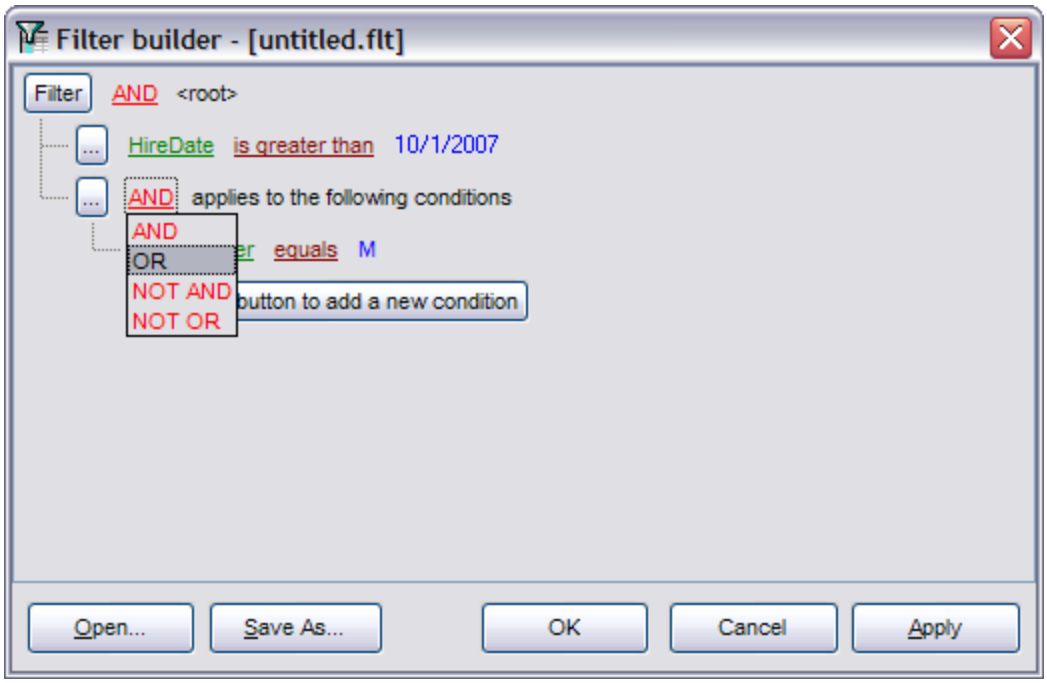
Invoking the Filter Builder dialog
Adding a new condition
Setting filter criteria
Setting filter operator
Setting filter criteria values
Setting group operator
Applying filter conditions

7.3.7 Setting group operator

Conditions of complex criteria can be combined with any of the four logical operators used: *AND*, *OR*, *NOT AND*, *NOT OR*.

In our case it is enough to click the **group operator** box and select the *AND* item from the drop-down menu.

© 2008 EMS Database Management Solutions



See also:

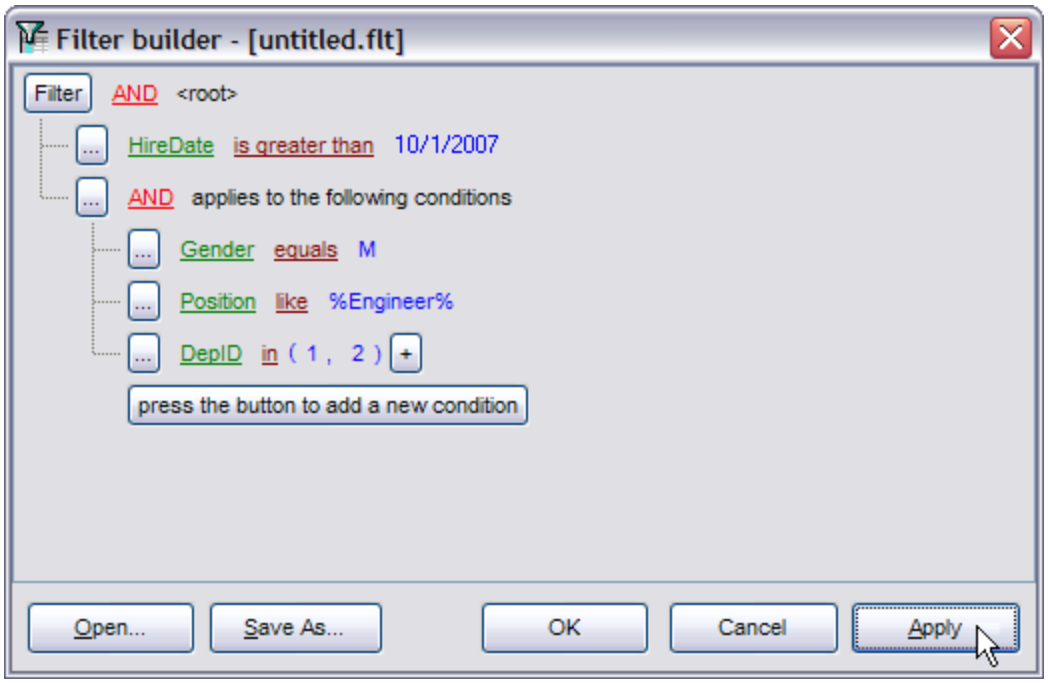
Invoking the Filter Builder dialog
Adding a new condition
Setting filter criteria
Setting filter operator
Setting filter criteria values
Adding a new group
Applying filter conditions

7.3.8 Applying filter conditions

Suppose we have created a condition within the new group. If we need, we can add more conditions at the same level and specify the required values using the value boxes. When the operation is completed, the **Filter Builder** dialog will look like in the screenshot below.

Click the **Apply** button to see the result of the filtering you have made, and click **OK** or **Cancel** to close the dialog with or without saving your filter conditions respectively.

© 2008 EMS Database Management Solutions



The **Filter Builder** dialog allows you to save filter criteria to and load them from external files. Clicking the **Save As...** or the **Open...** buttons activates the corresponding dialogs. Filter settings are stored in **.flt* files.
Please be informed that a column in the file is referenced by its position within a view, hence filter settings cannot be correctly restored if columns have been deleted from the view after saving the filter to a file.

See also:

Invoking the Filter Builder dialog
Adding a new condition
Setting filter criteria
Setting filter operator
Setting filter criteria values
Adding a new group
Setting group operator

© 2008 EMS Database Management Solutions

Part

8 Import/Export Tools

Using SQL Manager 2008 for SQL Server you are provided with powerful tools to import and export data to/from your SQL Server database.

- **Export Data Wizard**
  Exports data to various supported formats including *MS Excel*, *MS Access*, *RTF*, *HTML*, *PDF*, *CSV*, *XML*, *MS Excel 2007* and more.
- **Import Data Wizard**
  Imports data from any of supported formats: *MS Excel*, *MS Access*, *DBF*, *TXT*, *CSV*, *XML*, *MS Excel 2007*, *MS Word 2007* and more.
- **Export Data as SQL Script**
  Exports data to an SQL script as a number of INSERT statements.
- **Bulk Insert Wizard**
  Imports data with the BULK INSERT statement used.
- **Using templates**
  Facilitates using import/export wizards.

See also:

Getting started
Database Explorer
Database management
Database objects management
Query management tools
Data management
Database tools
Server tools
Personalization
External tools

© 2008 EMS Database Management Solutions

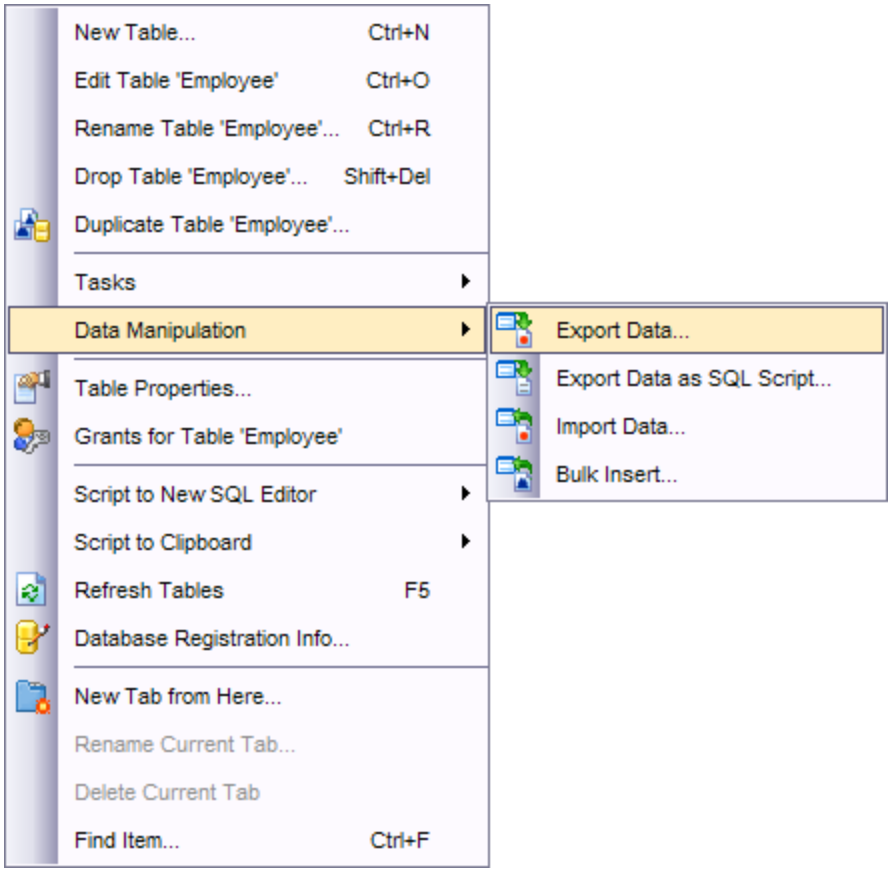
8.1 Export Data Wizard

Export Data Wizard allows you to export data from a table / view or from a query result to any of supported formats (*MS Excel*, *MS Access*, *MS Word*, *RTF*, *HTML*, *PDF*, *TXT*, *CSV*, *XML*, *DBF*, *MS Excel 2007*, *MS Word 2007*, etc.). You can save your settings as a template any time for future use.

To start the wizard, right-click the object in DB Explorer, select the **Data Manipulation** context menu group and proceed to the **Export Data...** item within this group.

Alternatively, you can open the **Data** tab of Table Editor / View Editor or the **Result(s)** tab of SQL Editor / Query Builder, right-click the grid there, then select the **Data Manipulation** context menu group and proceed to the **Export Data of <object_name>...** item within this group, or use the **Export Data** item of the Navigation bar.

- Setting name and format for the destination file
- Selecting fields for export
- Adjusting formats applied to exported data
- Setting header and footer text for the destination file
- Setting format-specific options
- Setting common export options
- Exporting data
- Using templates



© 2008 EMS Database Management Solutions

Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**No**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

Import Data Wizard
Export as SQL Script
Bulk Insert Wizard

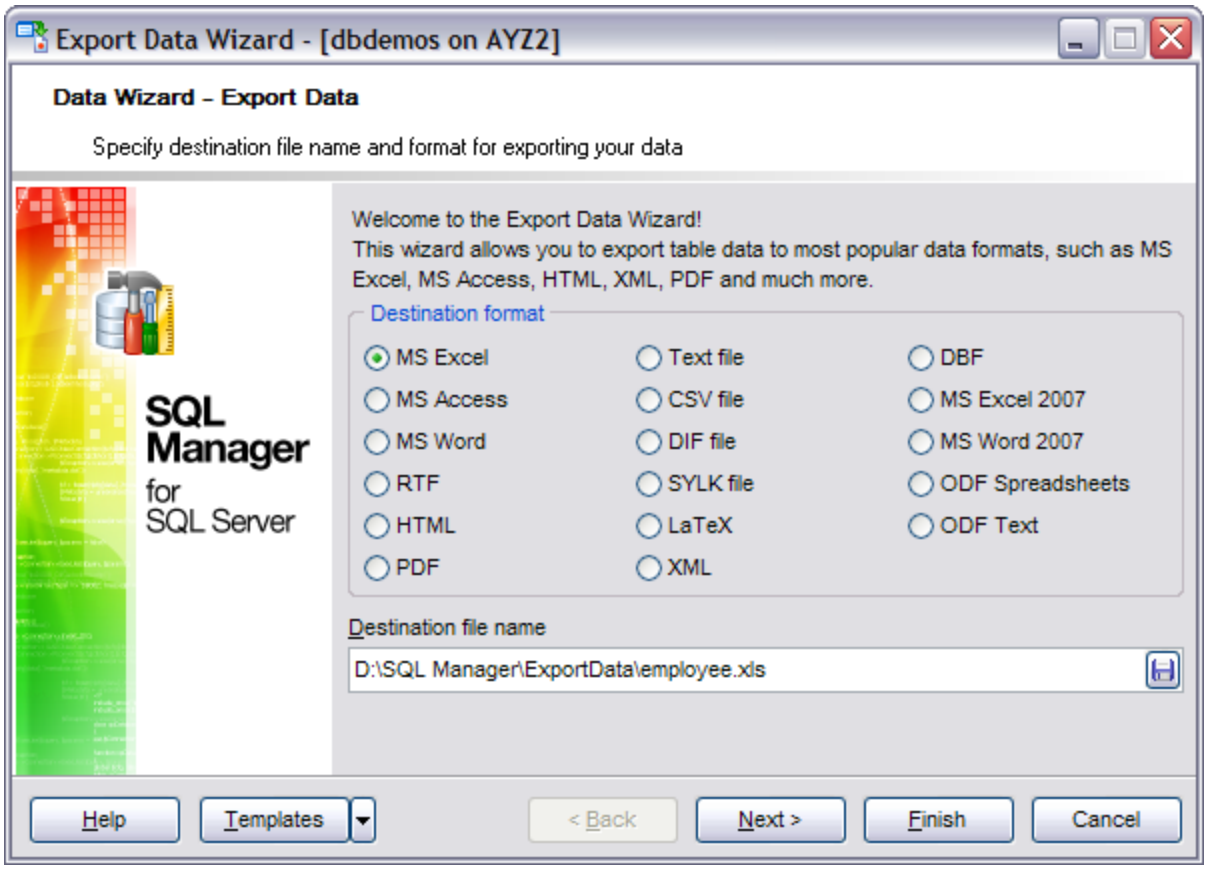
8.1.1 Setting destination file name and format

This step of the wizard allows you to select the destination file format you need to export data into.

Destination file name

Type in or use the button to specify the path to the file using the **Save as...** dialog. The file name extension changes automatically according to the selected **Destination format**.

© 2008 EMS Database Management Solutions



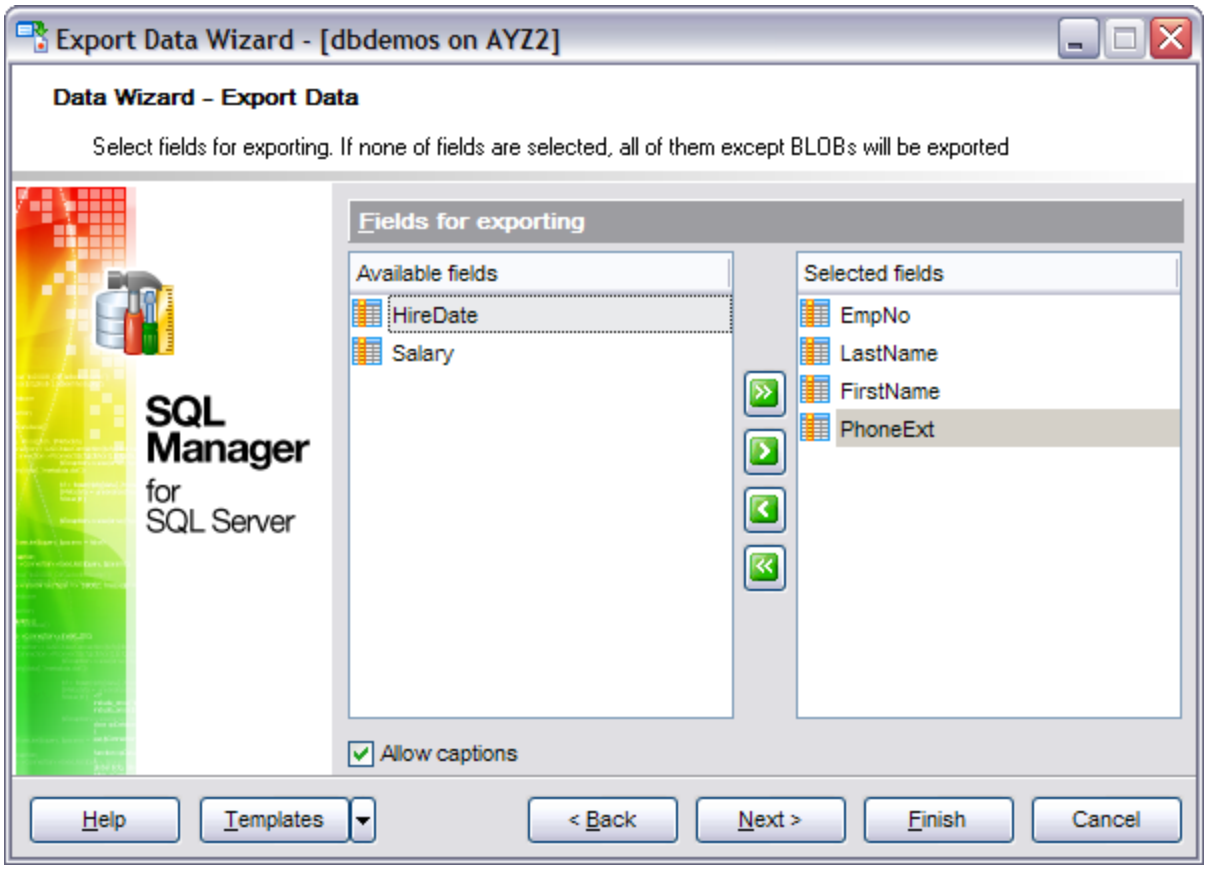
Destination format
Specify the format of the destination file. For details refer to Supported file formats.

Click the **Next** button to proceed to the Selecting fields for export step of the wizard.

8.1.2 Selecting fields for export

This step of the wizard allows you to select the table field(s) to be exported.
To select a field, you need to move it from the **Available fields** list to the **Selected fields** list. Use the buttons or drag-and-drop operations to move the fields from one list to another.

© 2008 EMS Database Management Solutions



If you leave all the fields in the **Available fields** list, all fields of the table (except BLOBs) will be exported.

Allow captions
Check this option if you need to export the field captions as well.

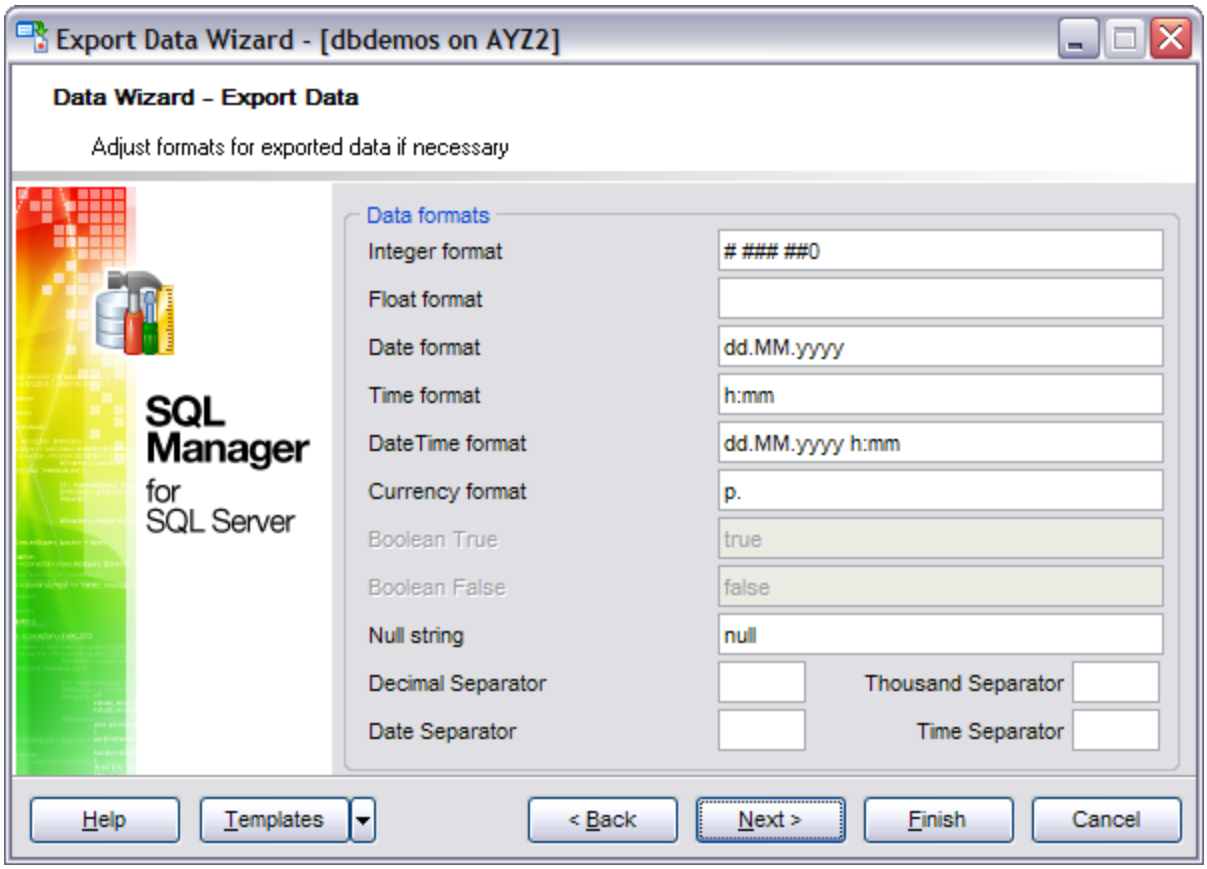
Click the **Next** button to proceed to the Adjusting data formats step of the wizard.

8.1.3 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: *Integer*, *Float*, *Date*, *Time*, *DateTime*, *Currency*, *Boolean True*, *Boolean False*, *NULL string*, *Decimal separator*, *Thousand separator*, *Date separator*, *Time separator*.

For more details refer to Format specifiers.

© 2008 EMS Database Management Solutions

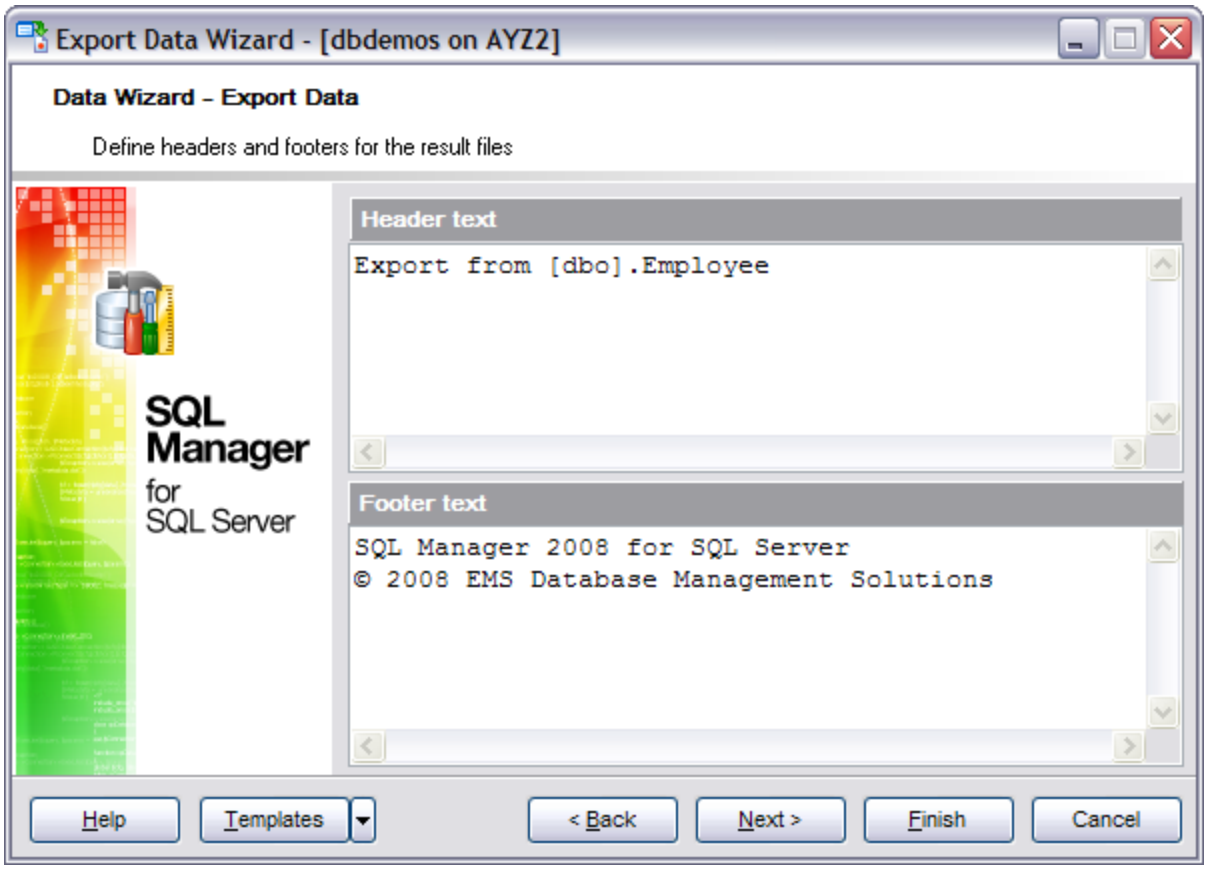


Click the **Next** button to proceed to the Setting header and footer step of the wizard.

8.1.4 Setting header and footer

Set **Header text** and **Footer text** for the result file. This text will appear at the beginning and at the end of the result file respectively.

© 2008 EMS Database Management Solutions



Click the **Next** button to proceed to Setting format-specific options.

8.1.5 Setting format-specific options

This step of the wizard allows you to customize **Format-specific options**:

- Excel options
- Access options
- Word / RTF options
- HTML options
- PDF options
- TXT options
- CSV options
- XML options
- MS Excel 2007 options
- MS Word 2007 options
- ODF options

To get more information about the file formats, see the Supported file formats page.

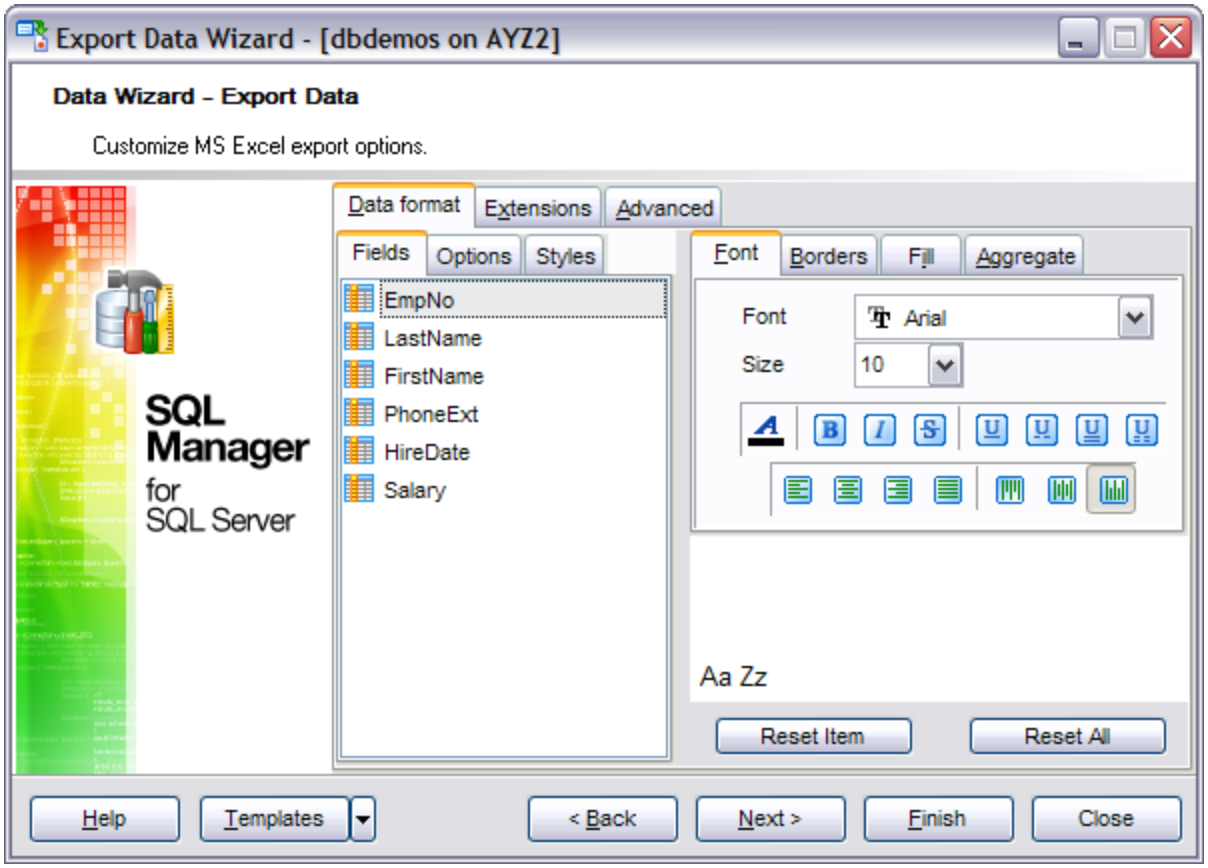
© 2008 EMS Database Management Solutions

8.1.5.1 Excel options

This step allows you to set options for the target **MS Excel** (**.xls*) file.

You can customize **Data format**, **Extensions** and set **Advanced** options available within the corresponding tabs:

- Data format
- Extensions
- Advanced



When you are done, click the **Next** button to proceed to Setting common export options.

8.1.5.1.1 Data format

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 the target Excel file striped by columns or rows.

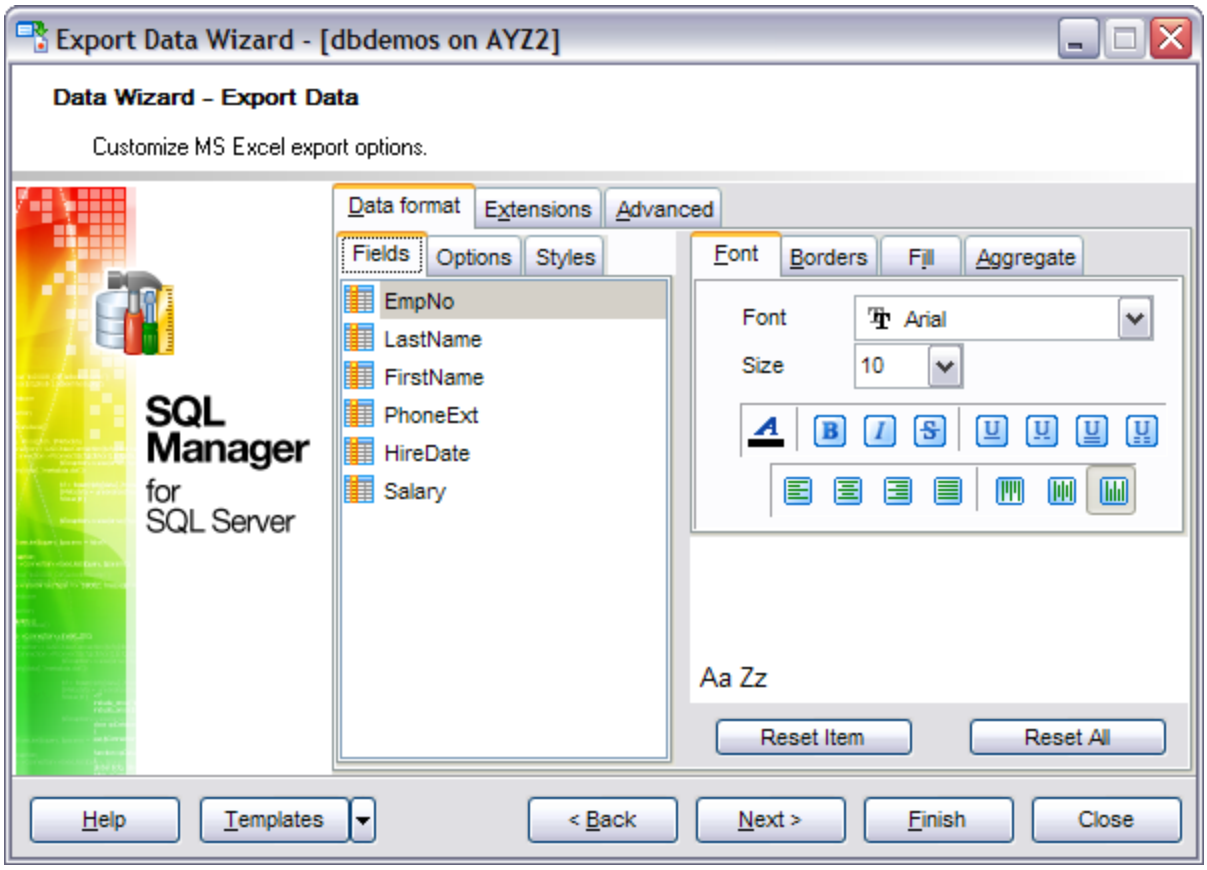
- Fields
- Options
- Styles

© 2008 EMS Database Management Solutions

For your convenience the previews illustrating the changes are displayed in the **Sample Group** area on each page of **Data Format** tab.

8.1.5.1.1.1 Fields

Using the **Fields** tab you can set *font* options, *border* and *fill* options and select and *aggregate functions* for all the **fields** you want to export.

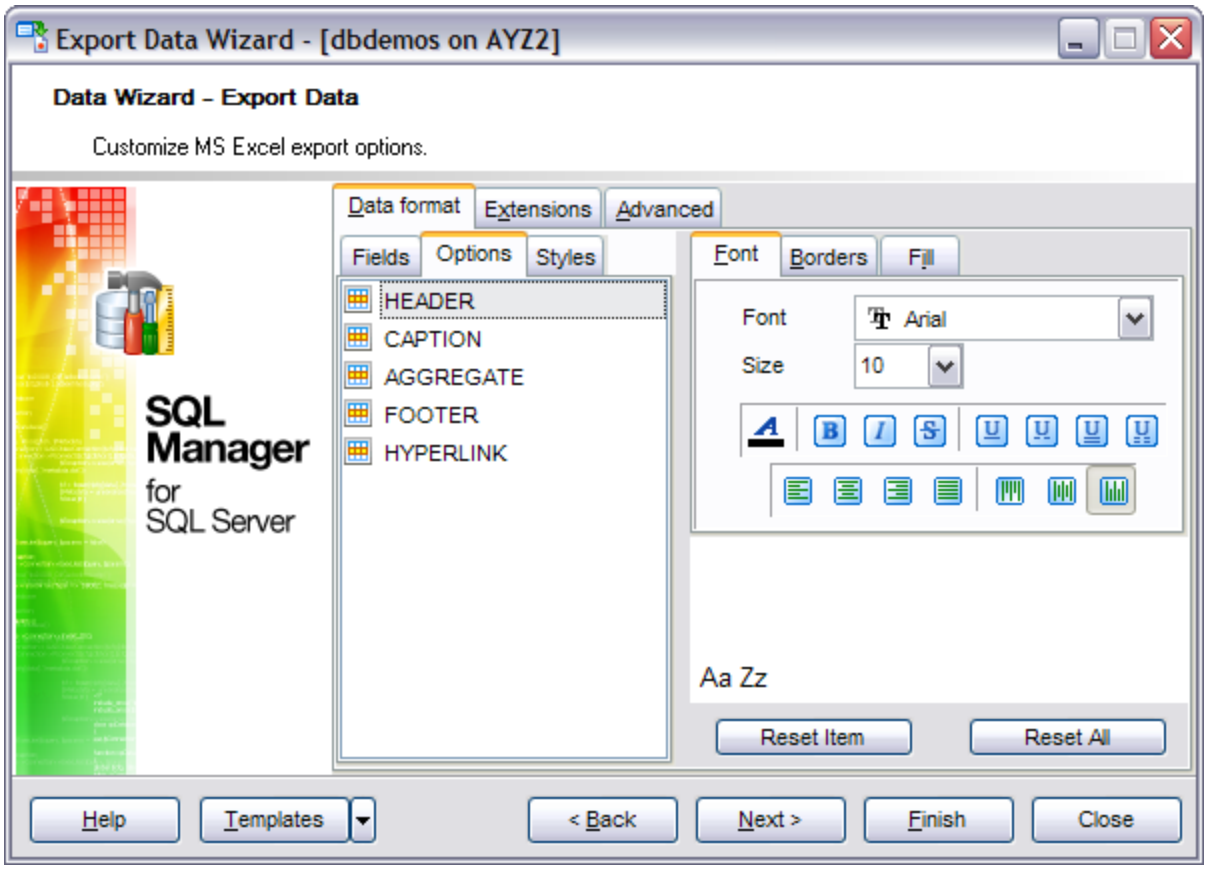


You can reset the changes any time using the **Reset Item** and the **Reset All** buttons.

8.1.5.1.1.2 Options

Using the **Options** tab you can set *font* options, *border* and *fill* options for all **elements** of the Excel sheet (*header*, *caption*, *footer*, *aggregates* and *hyperlinks*).

© 2008 EMS Database Management Solutions



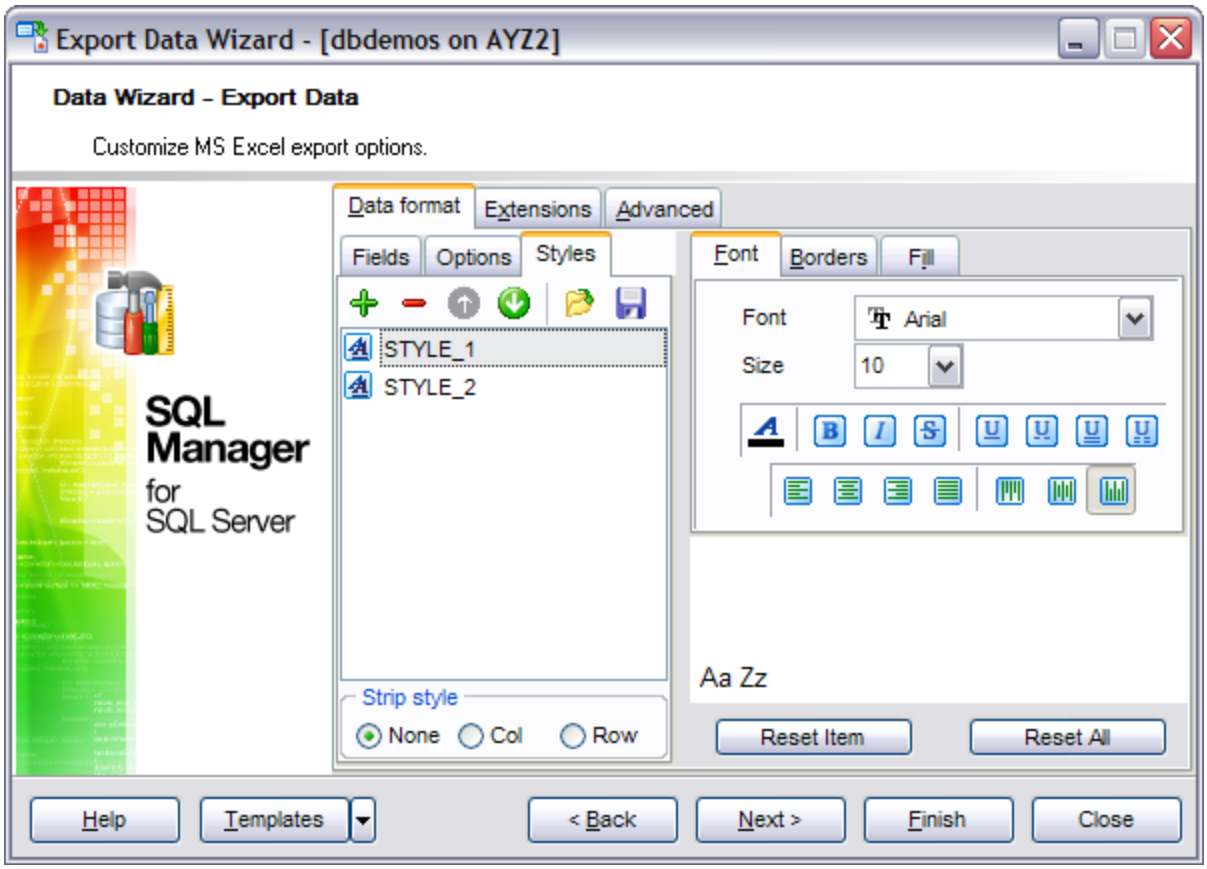
You can reset the changes any time using the **Reset Item** and the **Reset All** buttons.

8.1.5.1.1.3 Styles

Using the **Styles** tab you can make a style template: set *font* options, *border* and *fill* options and save them.

If you have created or loaded more than one style template, they can be ignored, or used *column-by-column* or *row-by-row* (it depends on the **Strip type** selection).

© 2008 EMS Database Management Solutions



You can reset the changes any time using the **Reset Item** and the **Reset All** buttons.

8.1.5.1.2 Extensions

The **Extensions** tab provides an ability to add hyperlinks and notes and to any cell of target file, to specify a value of a cell, to create a chart and to merge cells.

Click the **Plus** button to add an element;
click the **Minus** button to delete an element.

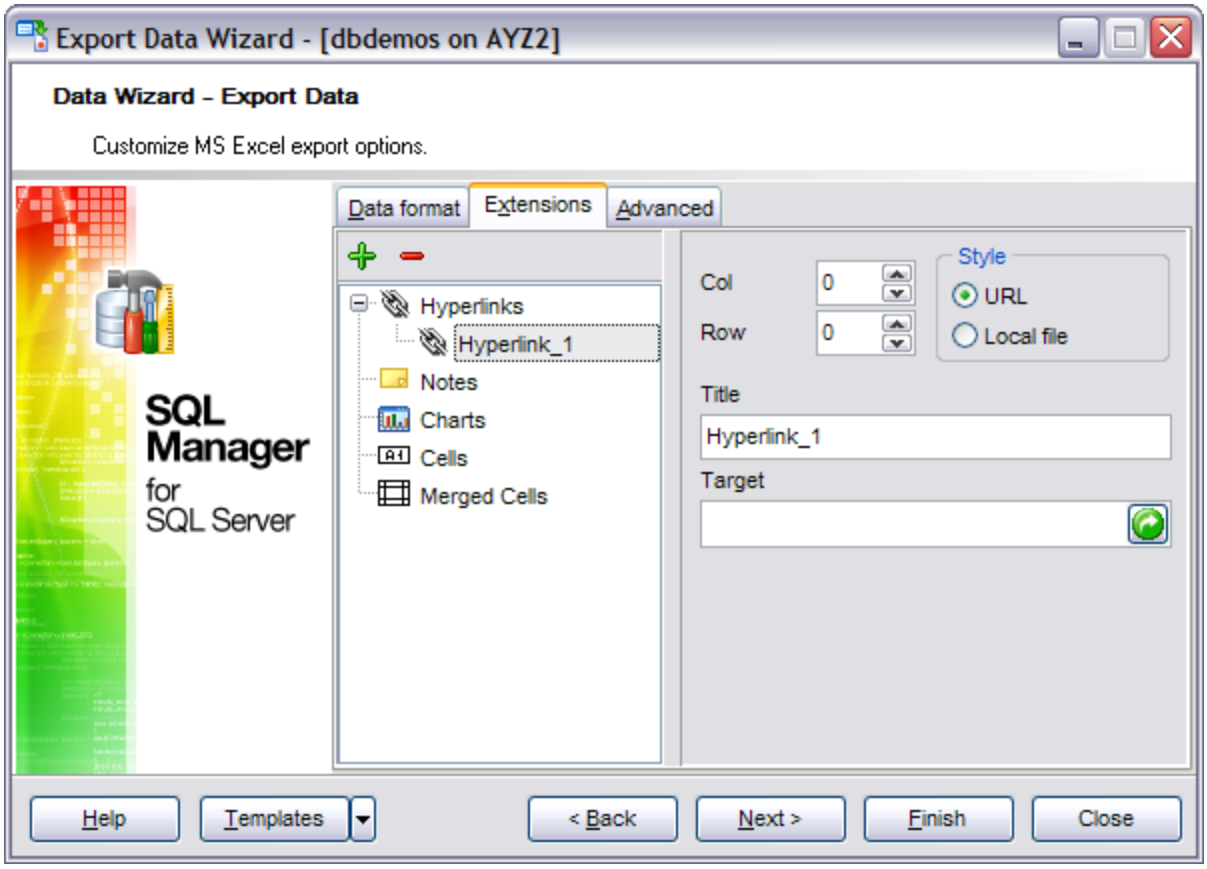
- Hyperlinks
- Notes
- Charts
- Cells
- Merged Cells

© 2008 EMS Database Management Solutions

8.1.5.1.2.1 Hyperlinks

If you need to create a **hyperlink**:

- set the cell coordinates (*Col* and *Row*);
- specify whether this is a *local* link or *URL*;
- enter the *title* of the hyperlink;
- specify the *target* file location or address.

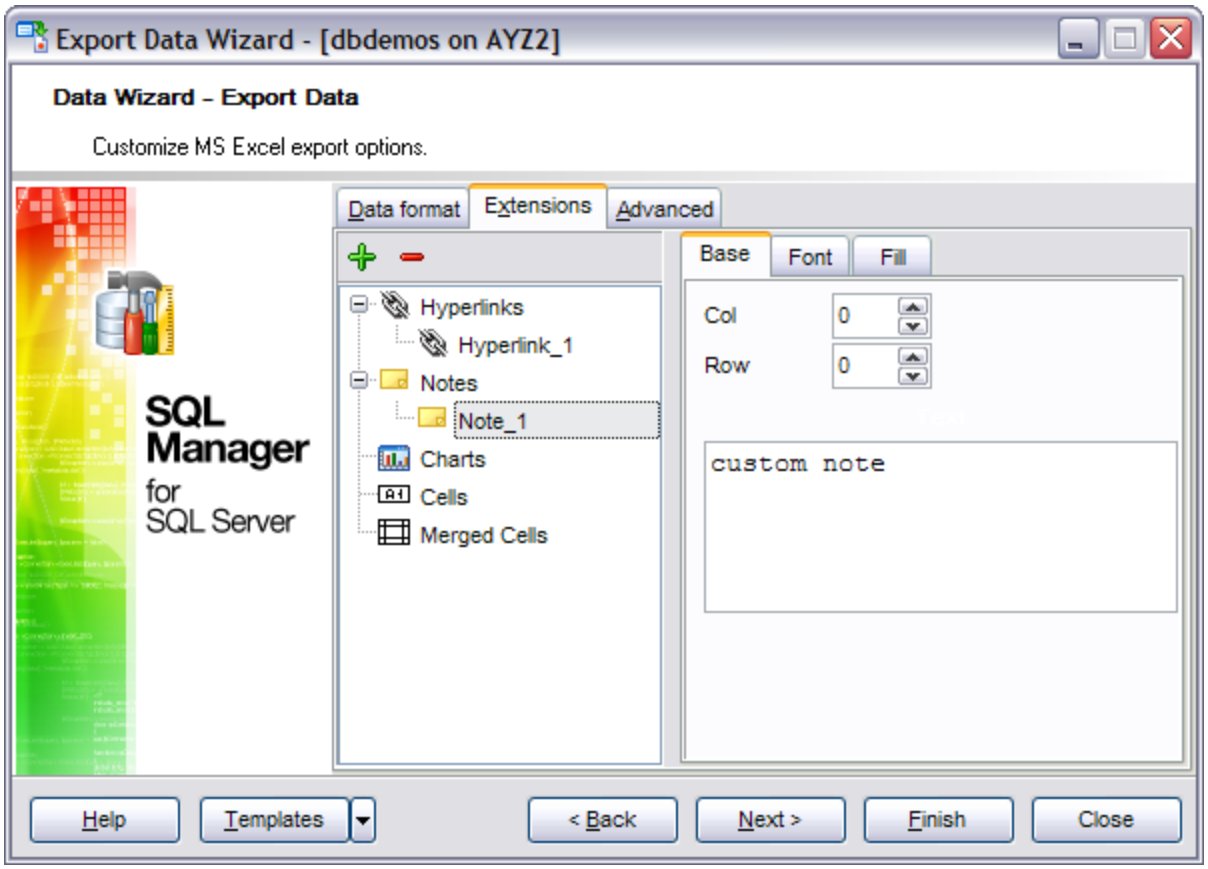


8.1.5.1.2.2 Notes

If you need to create a **note**:

- set the cell coordinates (*Col* and *Row*);
- enter *text* of a note for the cell;
- set the *font* and *fill* properties using the corresponding tabs.

© 2008 EMS Database Management Solutions

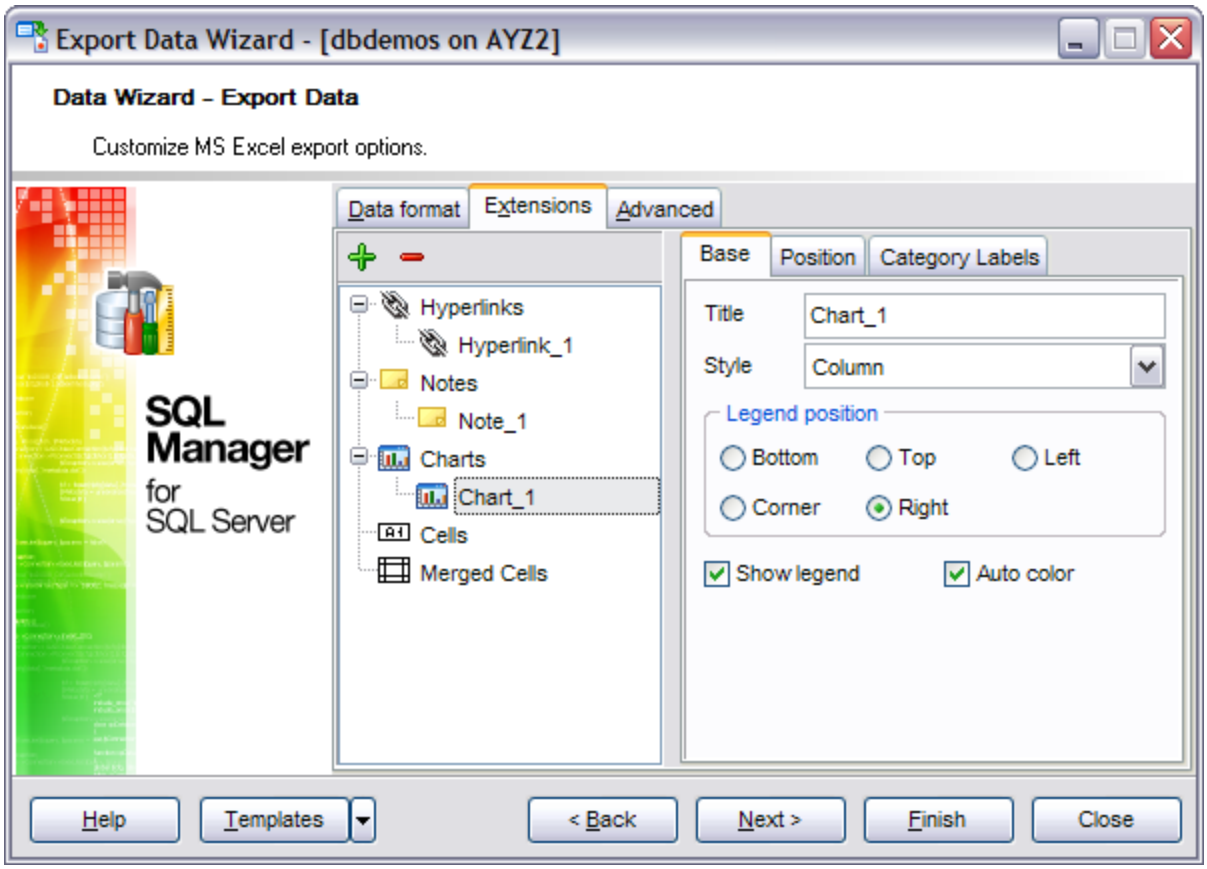


8.1.5.1.2.3 Charts

If you need to create a **chart**:

- enter the chart *title*;
- select the chart style (*Column*, *Column 3D*, *Bar*, *Bar 3D*, *Line*, *Line Mark*, *Line 3D*, etc.);
- set the legend position: *Bottom*, *Top*, *Left*, *Corner* or *Right*;
- specify if you want to show the legend;
- specify if you want to set the chart color automatically;
- define the chart *position* and *category labels* using the corresponding tabs.

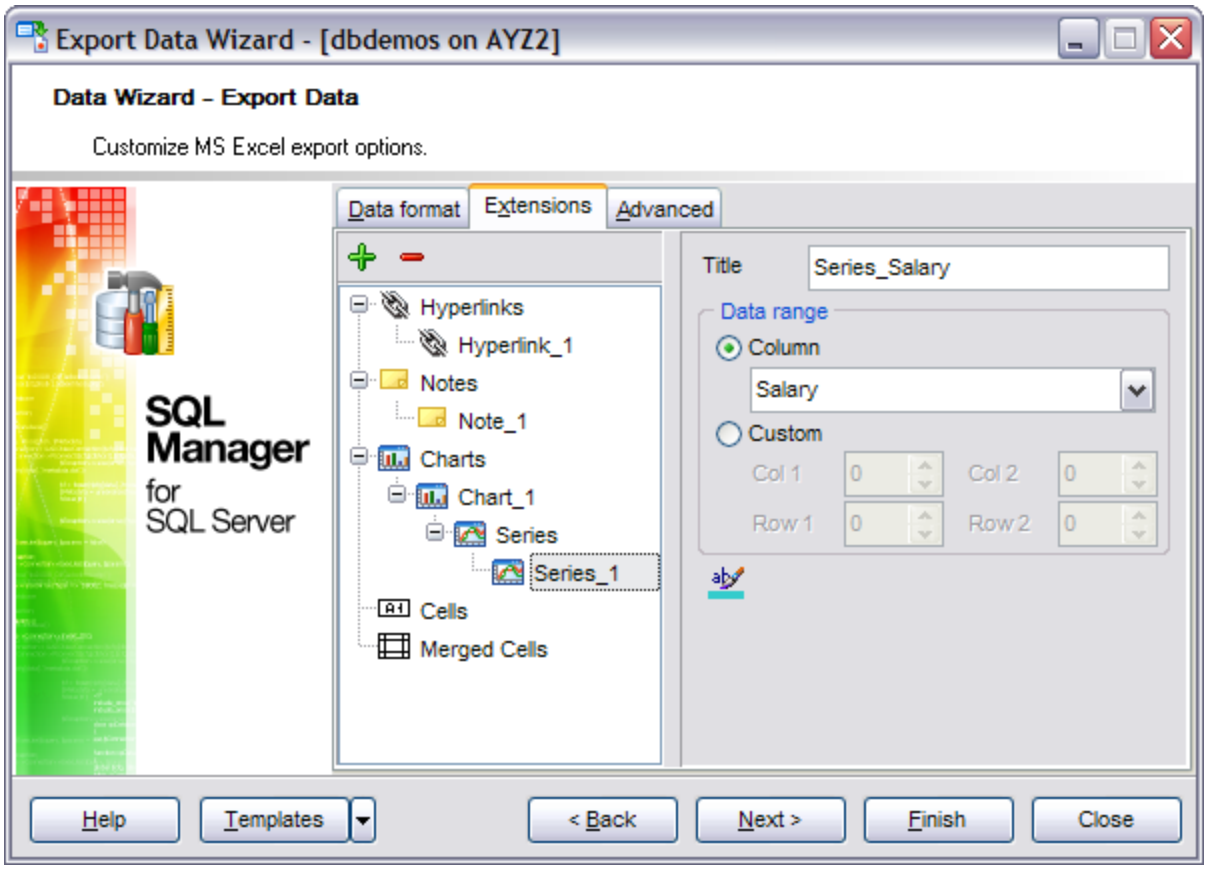
© 2008 EMS Database Management Solutions



If you need to add **series** for the chart:

- add one or more series using the + button;
- enter the *titles*;
- set data ranges (select a column from the drop-down list or set the custom range);
- define colors for all the graphs.

© 2008 EMS Database Management Solutions

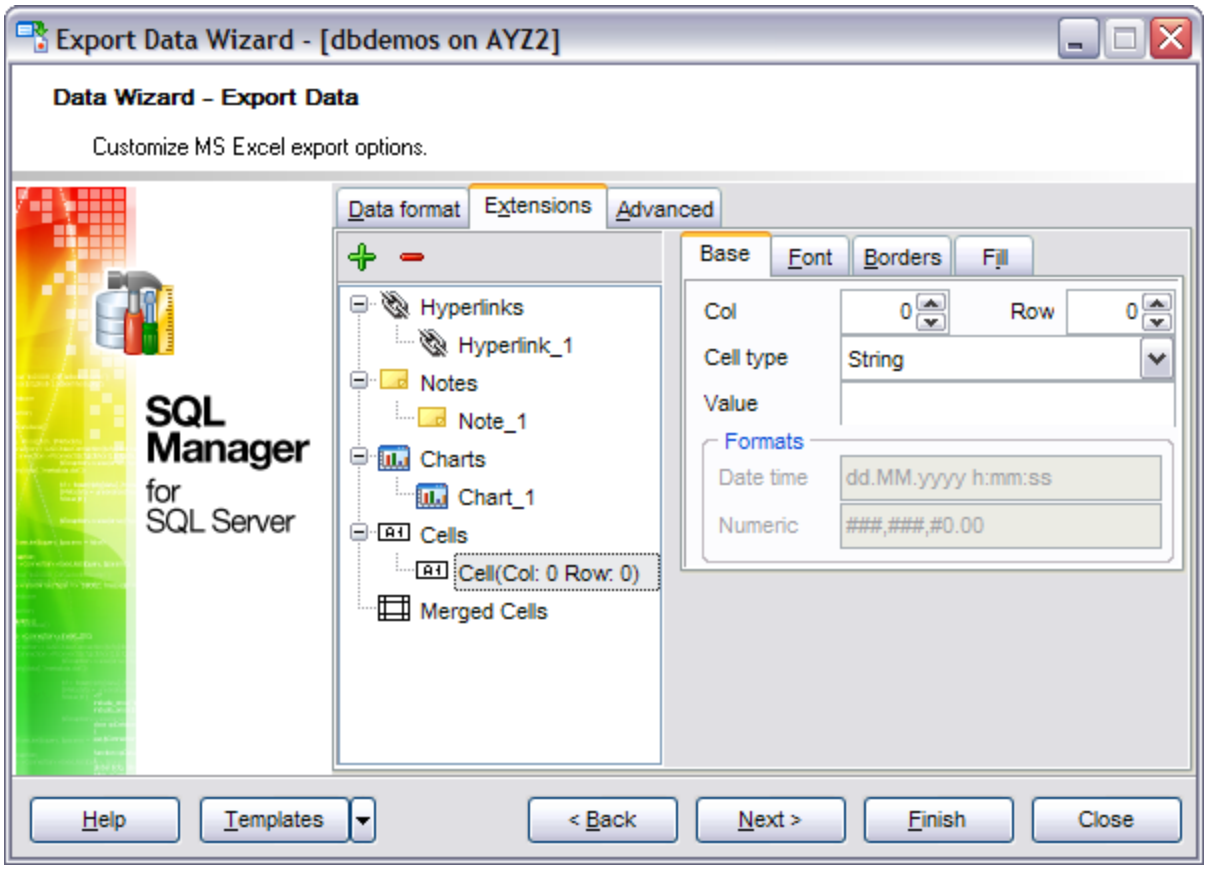


8.1.5.1.2.4 Cells

If you need to add a value in a specific cell:

- set the cell coordinates (*Col* and *Row*);
- select the cell type: *Boolean*, *DateTime*, *Numeric* or *String*;
- enter a *value*;
- if you are adding a numeric or a date/time value, you can set the cell *format;*
- set the *font*, *borders* and *fill* properties using the corresponding tabs.

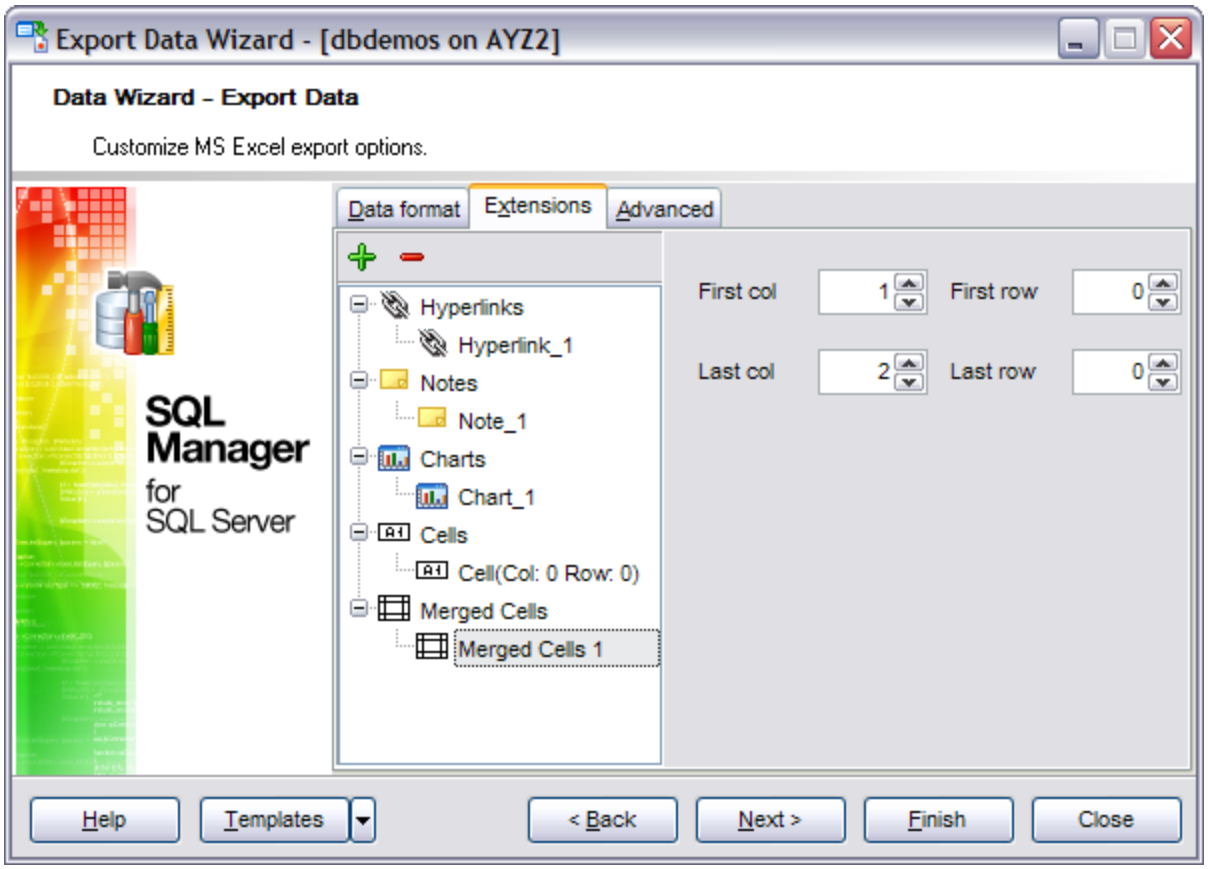
© 2008 EMS Database Management Solutions



8.1.5.1.2.5 Merged Cells

If you want to merge two or more cells, set the range of cell coordinates: *First col*, *Last col*, *First row*, *Last row*.

© 2008 EMS Database Management Solutions



8.1.5.1.3 Advanced

The **Advanced** tab allows you to set a number of advanced options to be applied to the result MS Excel file.

Page header
If necessary, enter some text for the page header.

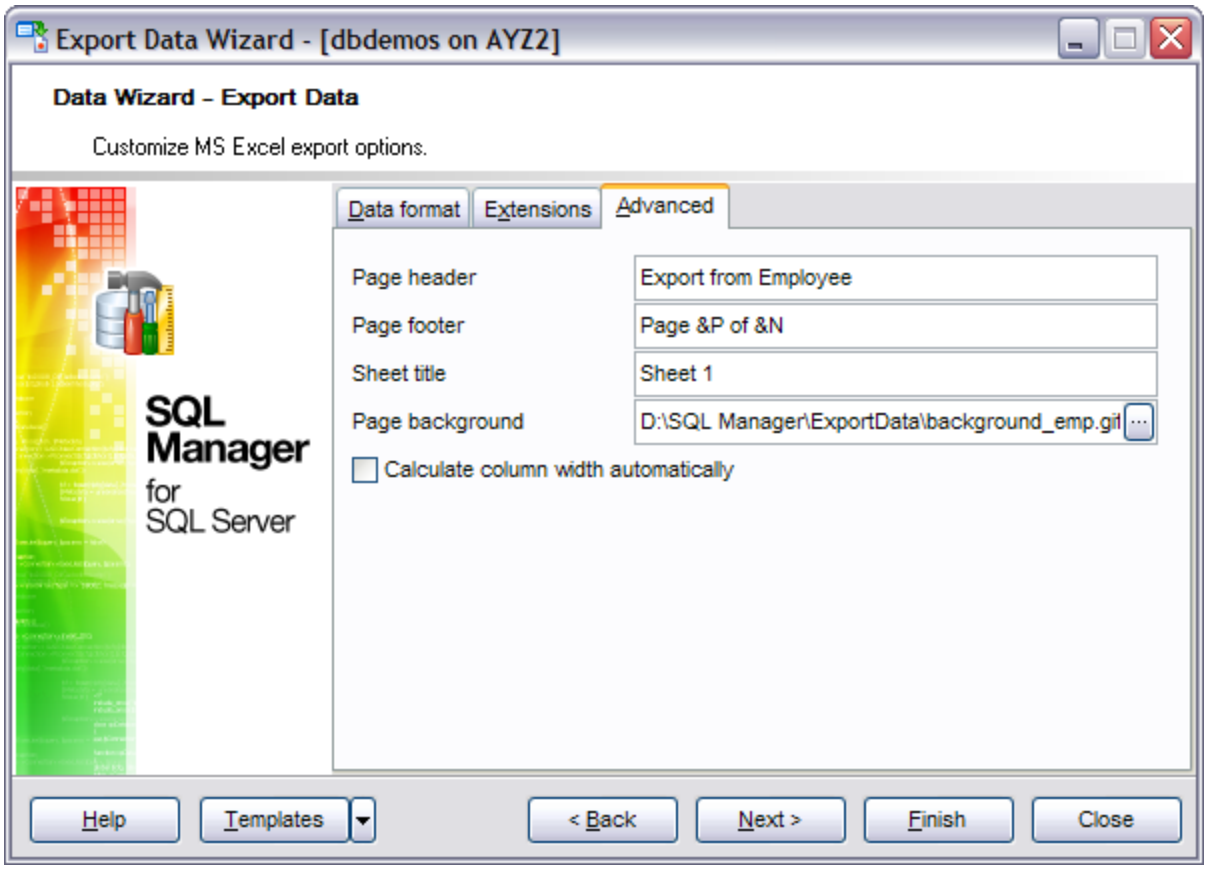
Page footer
If necessary, enter some text for the page footer.

Hint: It is also possible to set macros in the **Page header** and **Page footer** fields:
&N stands for the quantity of pages;
&P - the number of the current page.

Sheet title
Specify the sheet title for the target file.

Page background
If necessary, use the **Ellipsis** button to browse for a graphical file to be applied as the page background.

© 2008 EMS Database Management Solutions



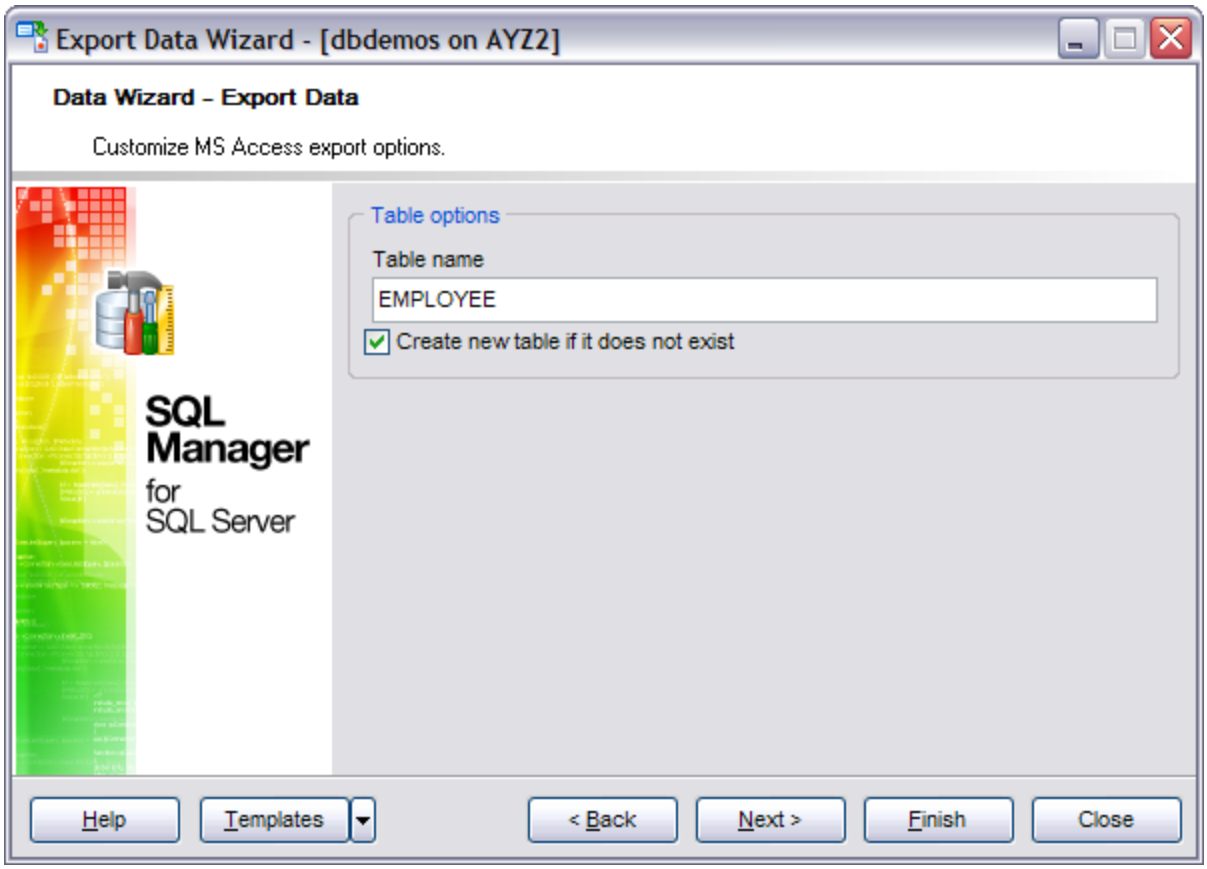
Calculate column width automatically
This option allows the wizard to determine column width in the target file automatically.

8.1.5.2 Access options

This step allows you to set options for the target **MS Access** (**.mdb*) file.

Set the name for the target table and specify whether the wizard should create a new table in the MS Access database if it does not exist yet, or use the existing table to export data into.

© 2008 EMS Database Management Solutions



When you are done, click the **Next** button to proceed to Setting common export options.

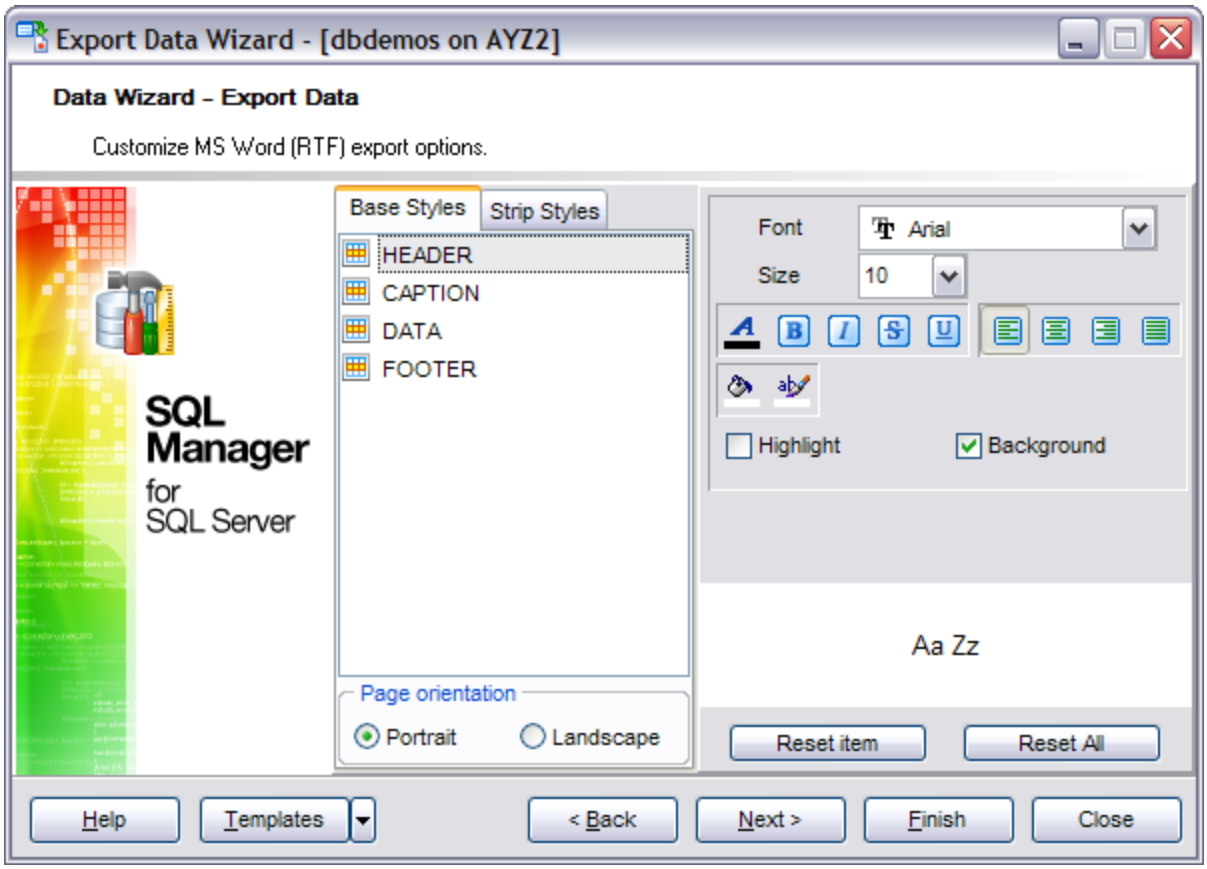
8.1.5.3 Word / RTF options

This step allows you to set options for the target **MS Word** (*.*doc*) and **Rich Text Format** (*.*rtf*) files.

- Base Styles
- Strip Styles

For your convenience the previews illustrating the changes are displayed in the **Sample Group** area within the *Base Styles* and the *Strip Styles* tabs.

© 2008 EMS Database Management Solutions

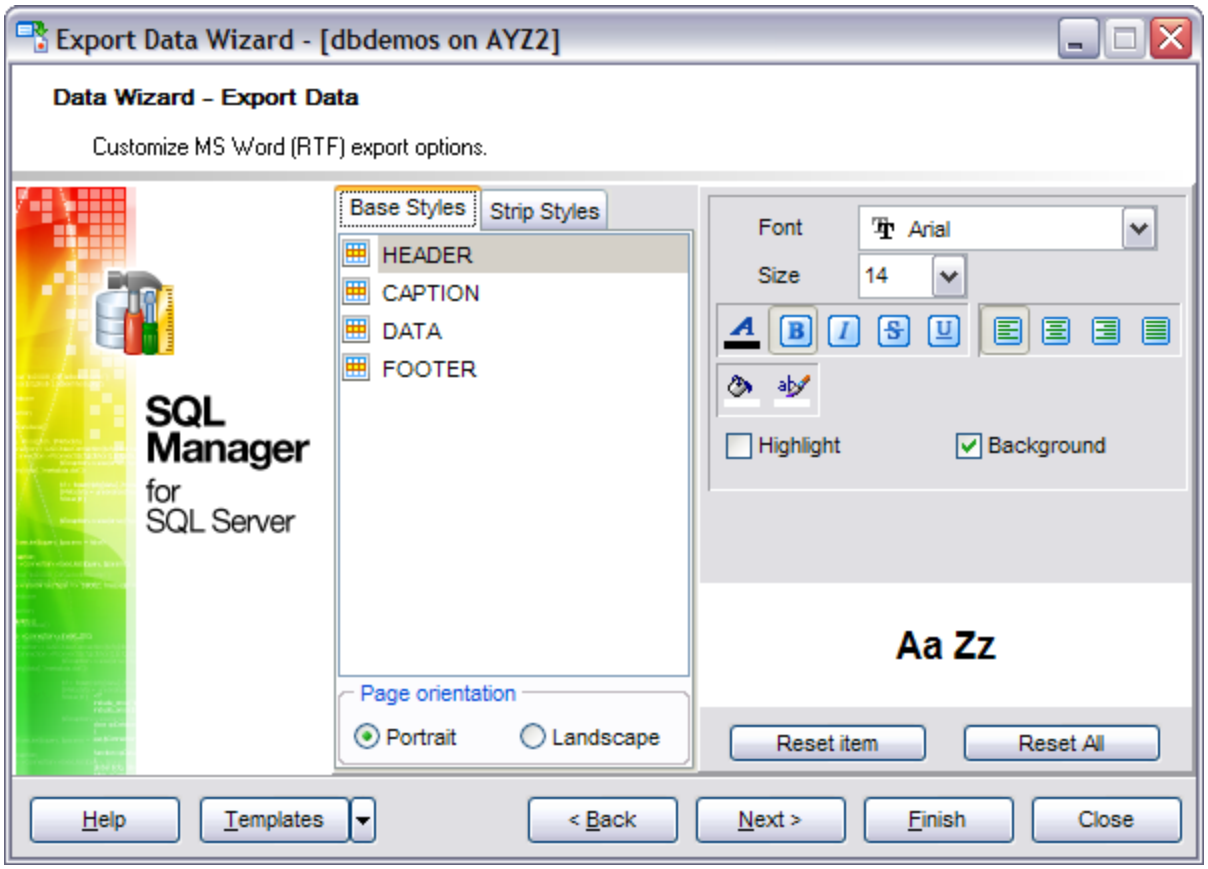


When you are done, click the **Next** button to proceed to Setting common export options.

8.1.5.3.1 Base Styles

The **Base Styles** tab contains the list of target file entities: *HEADER*, *CAPTION*, *DATA*, *FOOTER*. You can customize style options, such as *font* and *size*, *background* and *foreground colors*, *text alignment*, etc. for each of them by clicking the corresponding item in the list and setting the options in the right-side panel. You can also switch **page orientation** for the target Word/RTF file using this tab.

© 2008 EMS Database Management Solutions



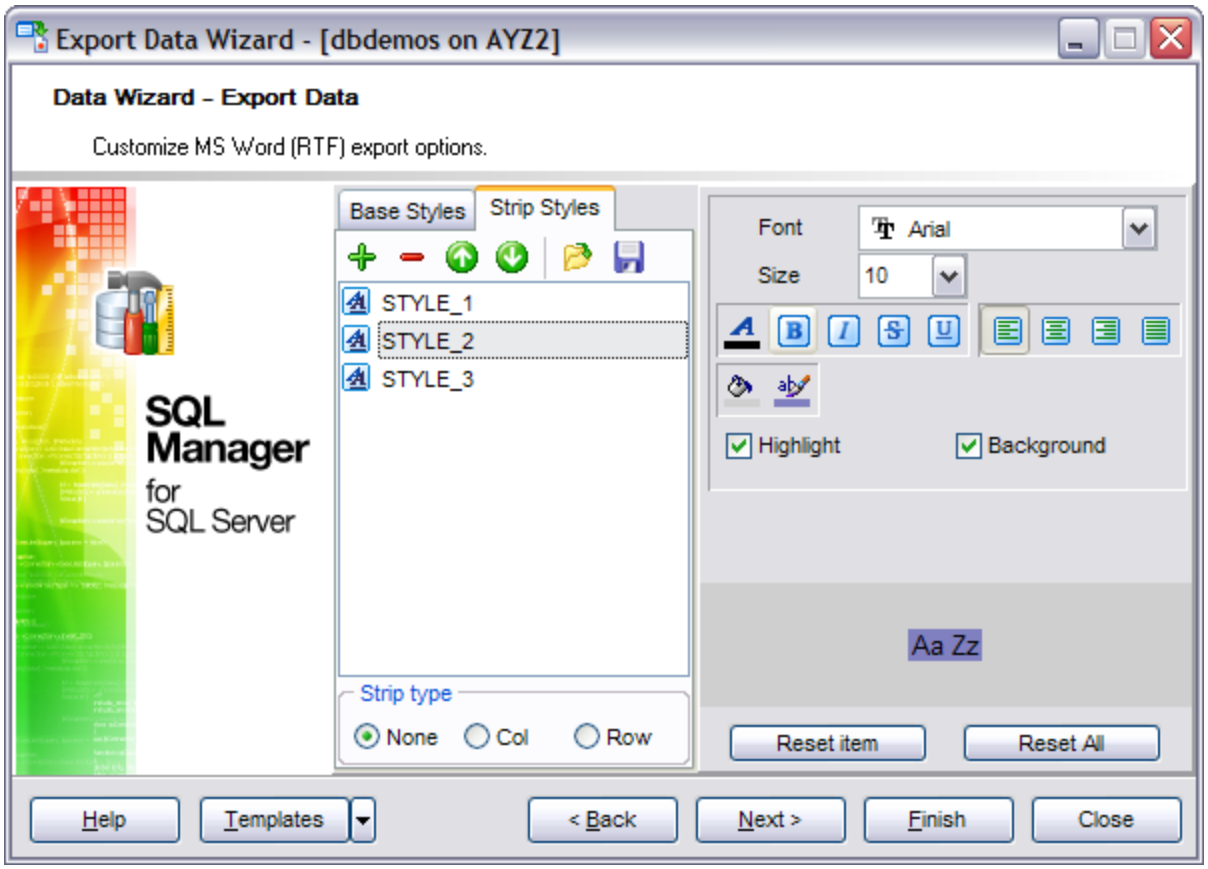
You can reset the changes any time using the **Reset Item** and the **Reset All** buttons.

8.1.5.3.2 Strip Styles

Using the **Strip Styles** tab you can create a style template: set *font*, *size*, *background* and *foreground colors*, *text alignment*, *highlight* and save them.

If you have created or loaded more than one style template, they can be ignored, or used *column-by-column* or *row-by-row* (it depends on the **Strip type** selection).

© 2008 EMS Database Management Solutions



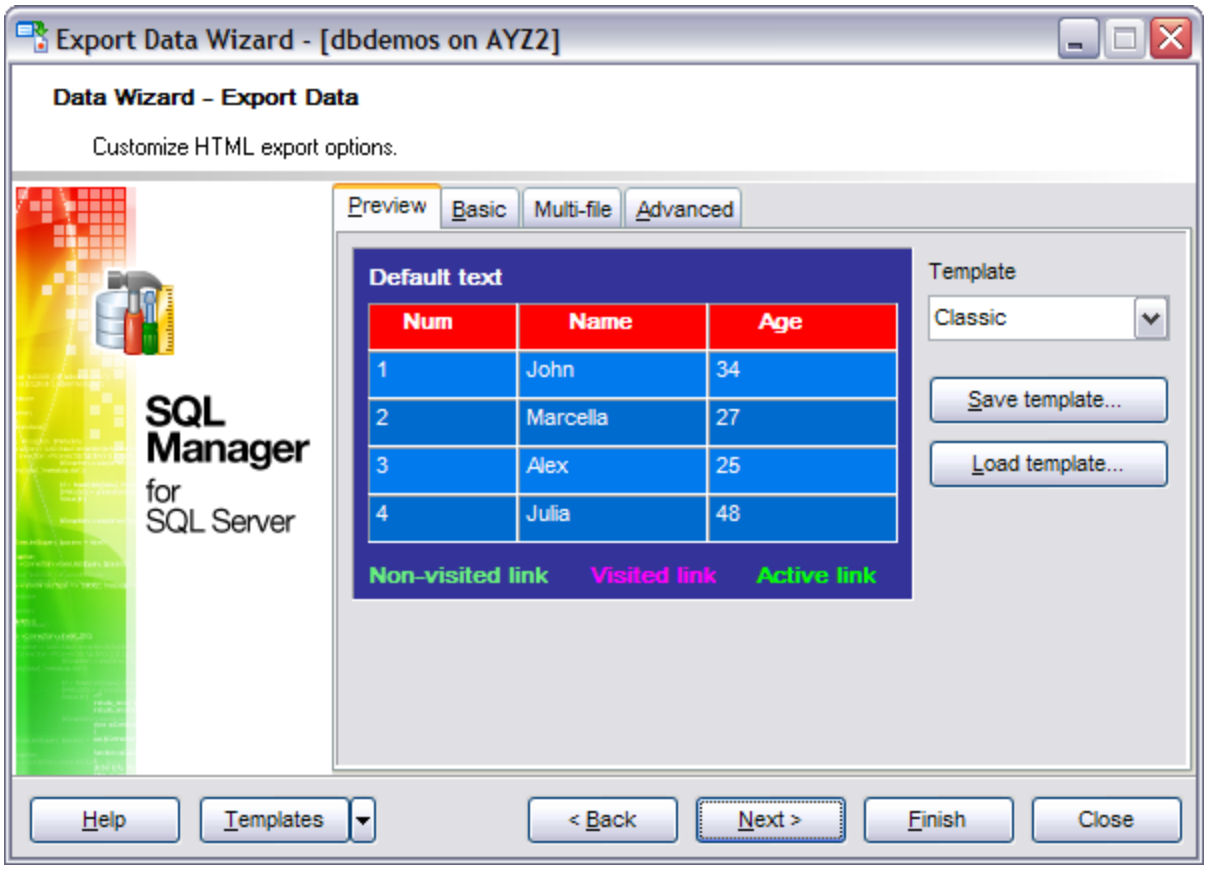
You can reset the changes any time using the **Reset Item** and the **Reset All** buttons.

8.1.5.4 HTML options

This step allows you to set options for the target **HTML** (*.*html*) file.

- Preview
- Basic
- Multi-file
- Advanced

© 2008 EMS Database Management Solutions

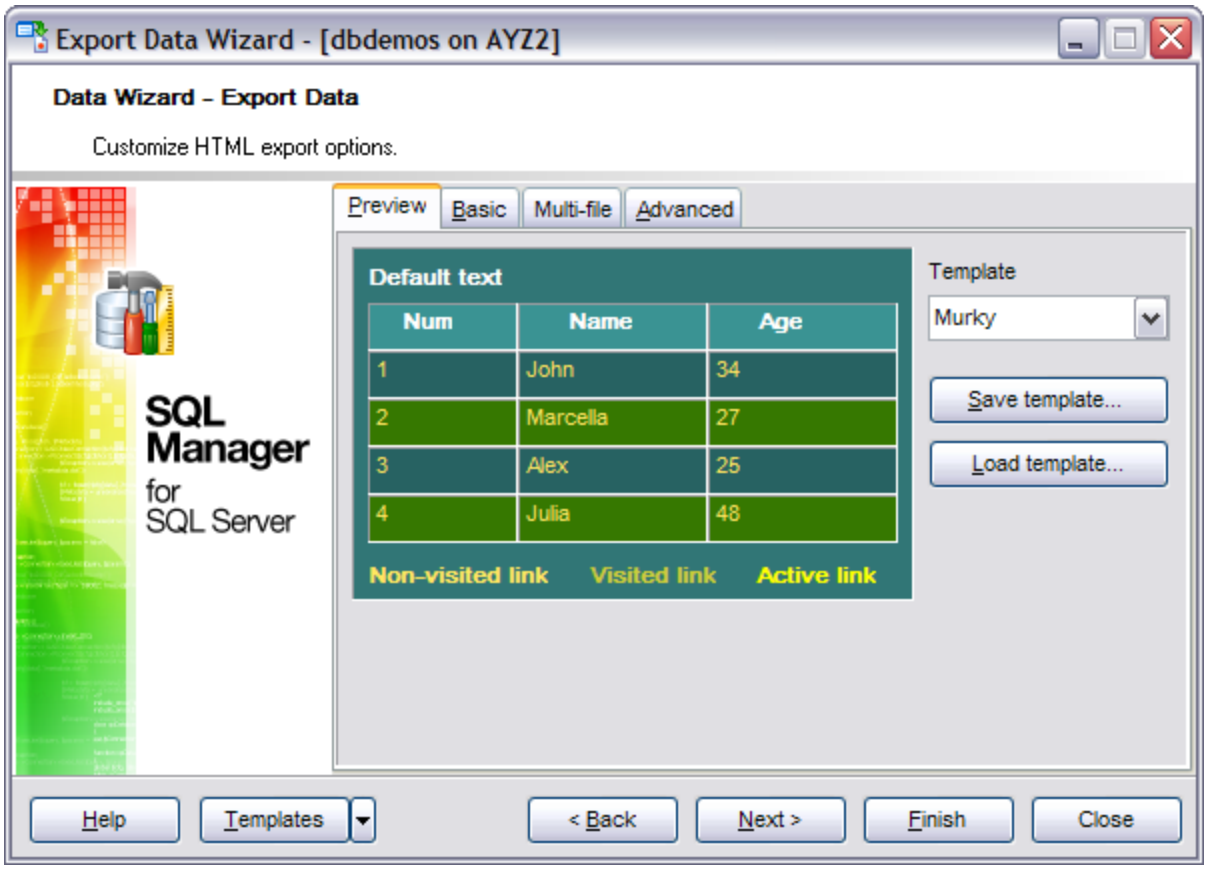


When you are done, click the **Next** button to proceed to Setting common export options.

8.1.5.4.1 Preview

The **Preview** tab allows you to customize the style that will be applied to the target HTML file using a number of built-in templates provided in the **Templates** combo-box. You can select any of these templates and customize it by clicking objects in the preview panel, and save the settings as a custom template using the **Save template...** button. Use the **Load template...** button to load a previously saved custom template from your hard disk. Click on an element of the table to select a color.

© 2008 EMS Database Management Solutions

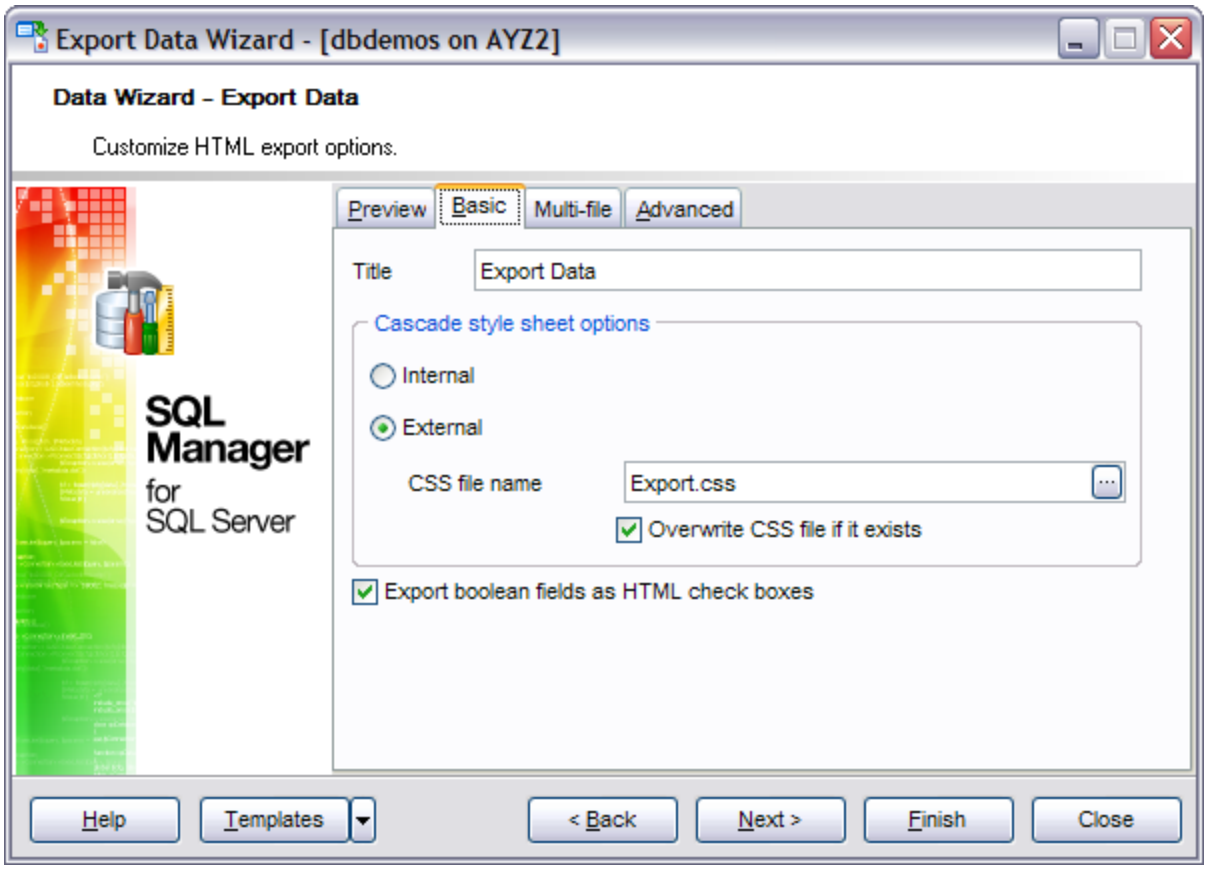


8.1.5.4.2 Basic

The **Basic** tab allows you to specify the basic parameters of target HTML file:

- specify the title of the result file;
- select whether the cascade style sheet (CSS) should be internal or external (the **Ellipsis** button to browse for a **.css* file);
- determine whether boolean fields of the table should be exported as HTML check boxes.

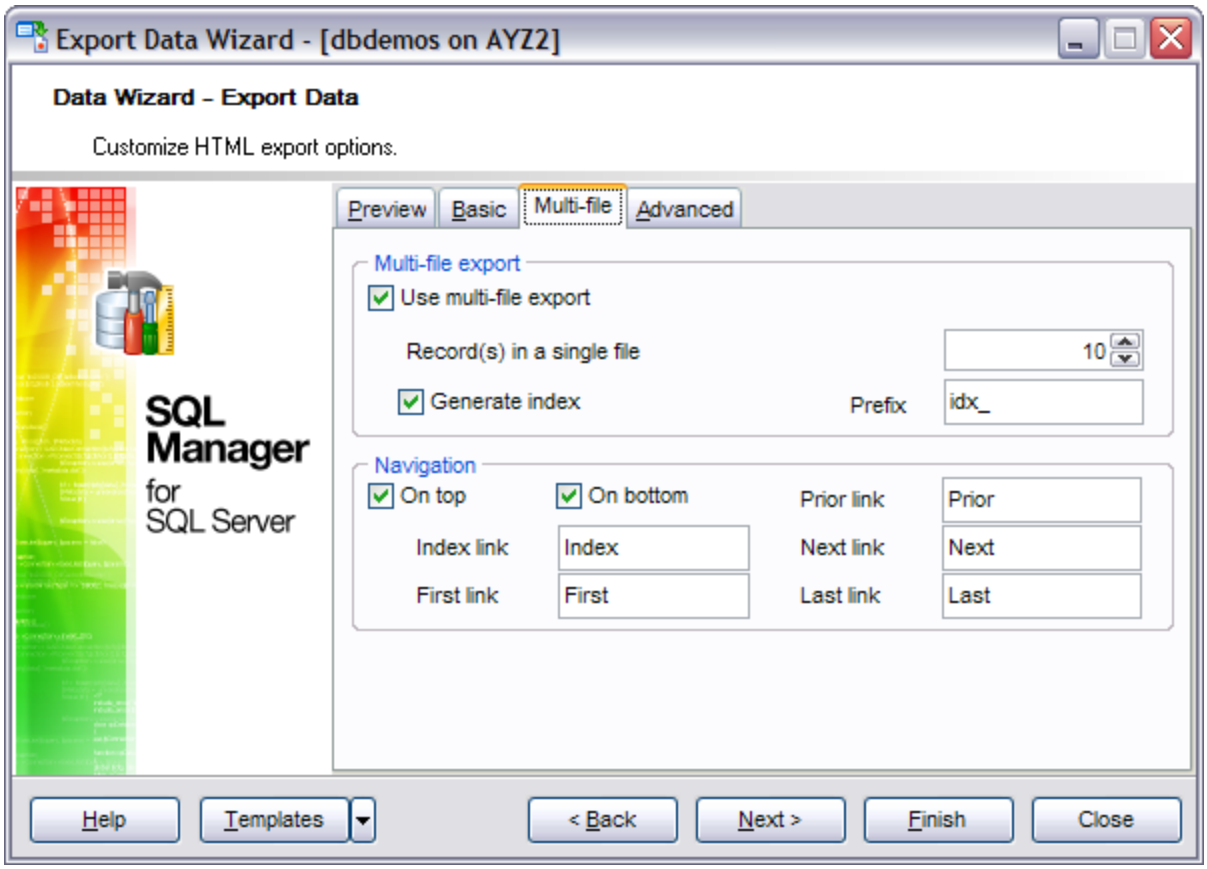
© 2008 EMS Database Management Solutions



8.1.5.4.3 Multi-file

The **Multi-file** tab provides you with an ability to split the 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 to navigate between the exported files.

© 2008 EMS Database Management Solutions



8.1.5.4.4 Advanced

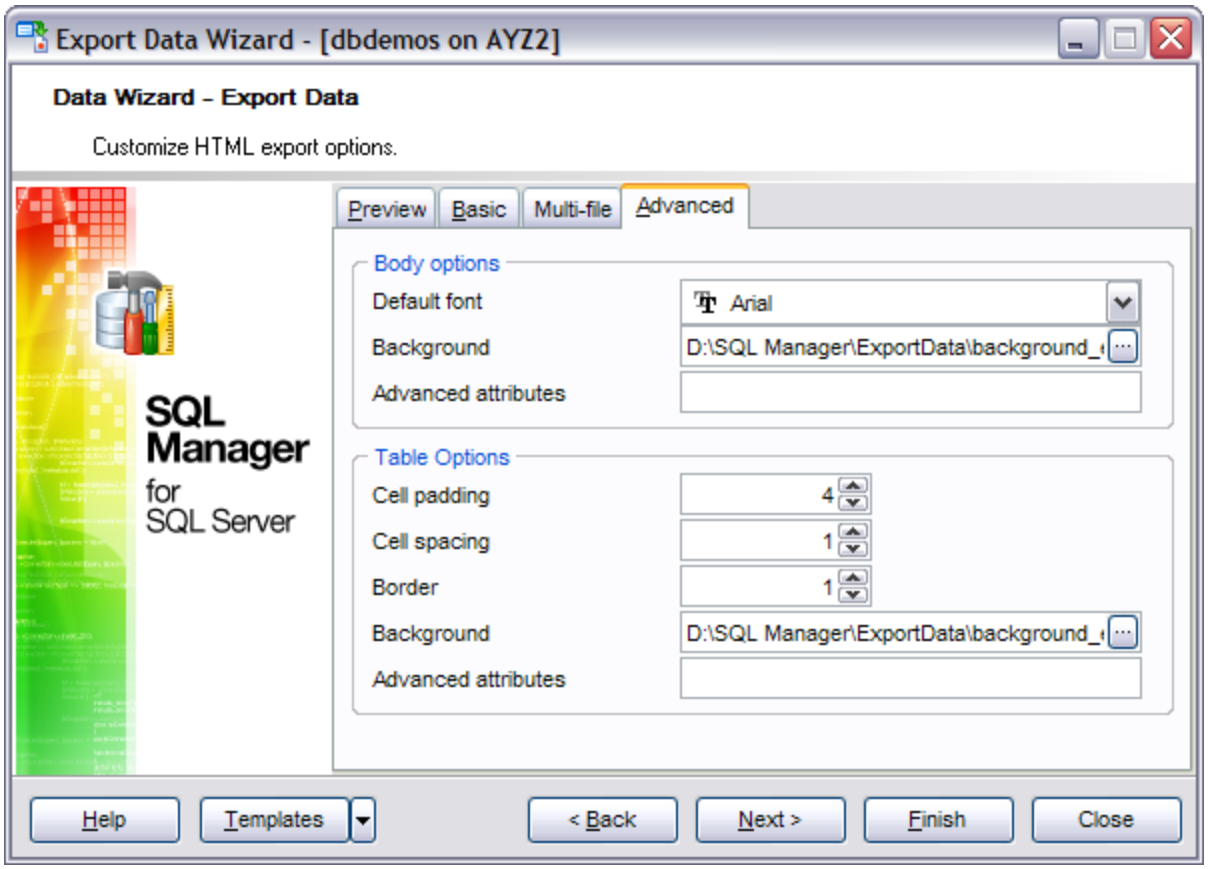
The **Advanced** tab allows you to set a number of advanced options to be applied to the result HTML file.

Default font
Use the drop-down menu to select the font that will be used in the result file by default.

Background
If necessary, use the **Ellipsis** button to browse for a graphical file to be applied as the page background.

You can also set a number of common **Table options**: *cell padding*, *cell spacing*, *border*, *background*.

© 2008 EMS Database Management Solutions



It is also possible to define **advanced attributes** for both the HTML body and table.

8.1.5.5 PDF options

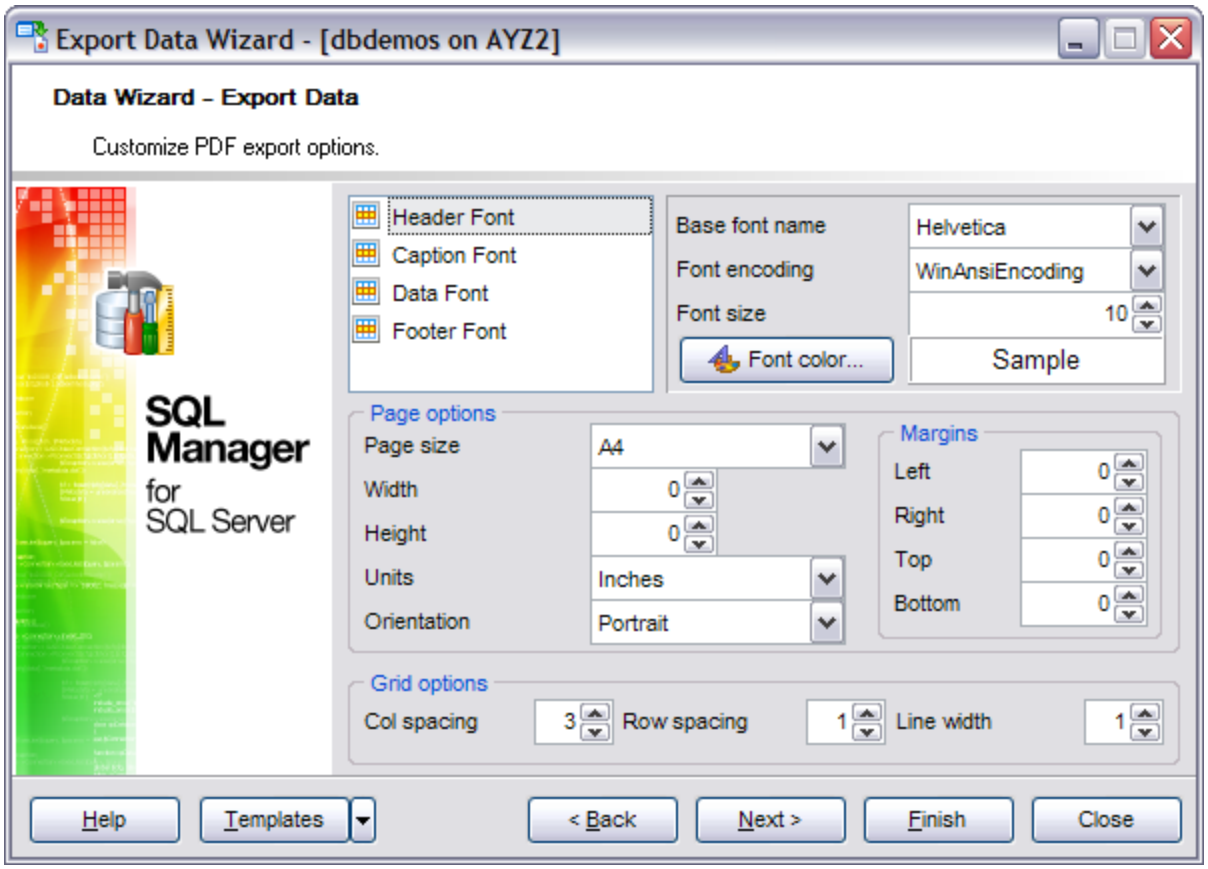
This step allows you to set options for the target **PDF** (**.pdf*) file.

Fonts
This group of options allows you to customize fonts for the *header*, *caption*, *data*, *footer* of the result file.

For your convenience the previews illustrating the changes are displayed in the **Sample Group** area.

Additionally you can customize the **page options** (page size, width, height, etc.), **page margins** and **Grid options** of the result PDF file.

© 2008 EMS Database Management Solutions



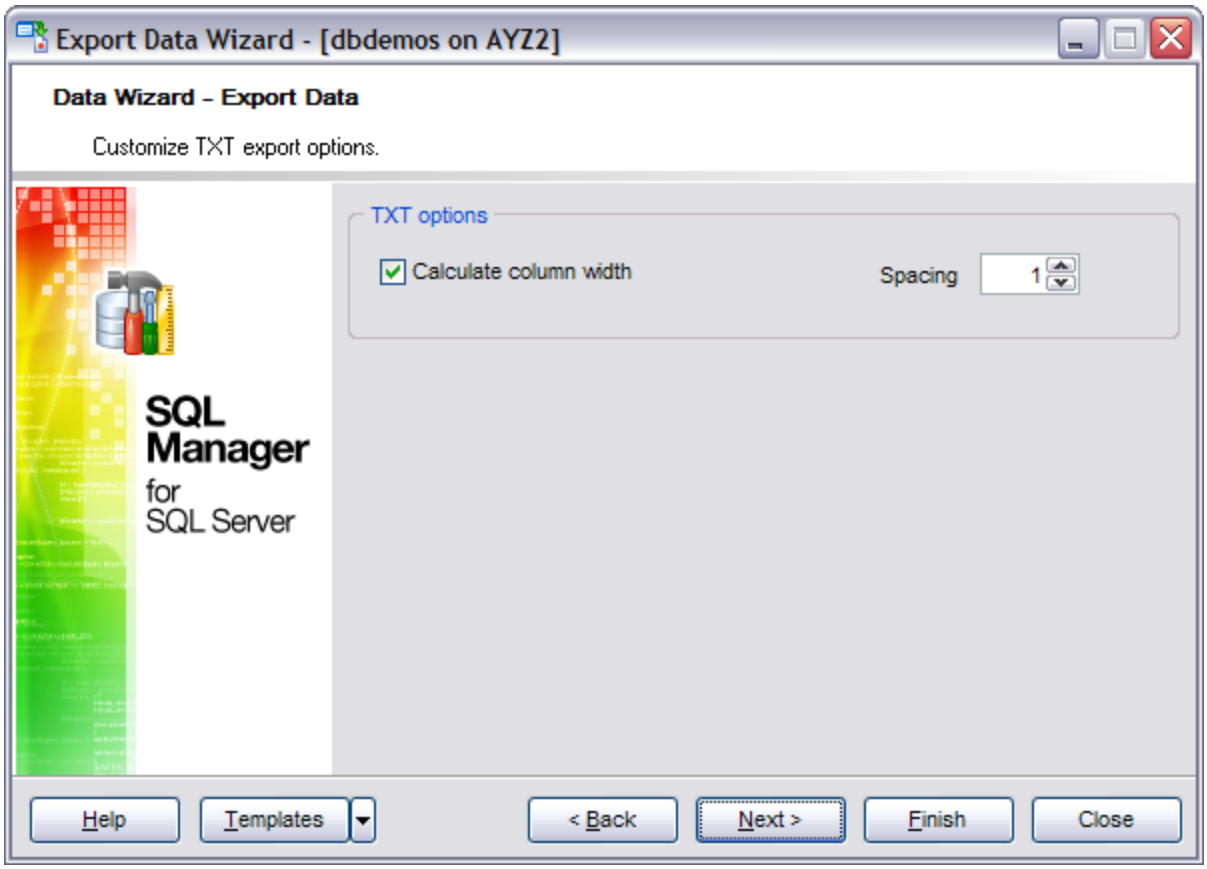
When you are done, click the **Next** button to proceed to Setting common export options.

8.1.5.6 TXT options

This step allows you to set options for the target **text** (*.*txt*) file.

Set the **Calculate column width** option on if you want each column of the 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.

© 2008 EMS Database Management Solutions



When you are done, click the **Next** button to proceed to Setting common export options.

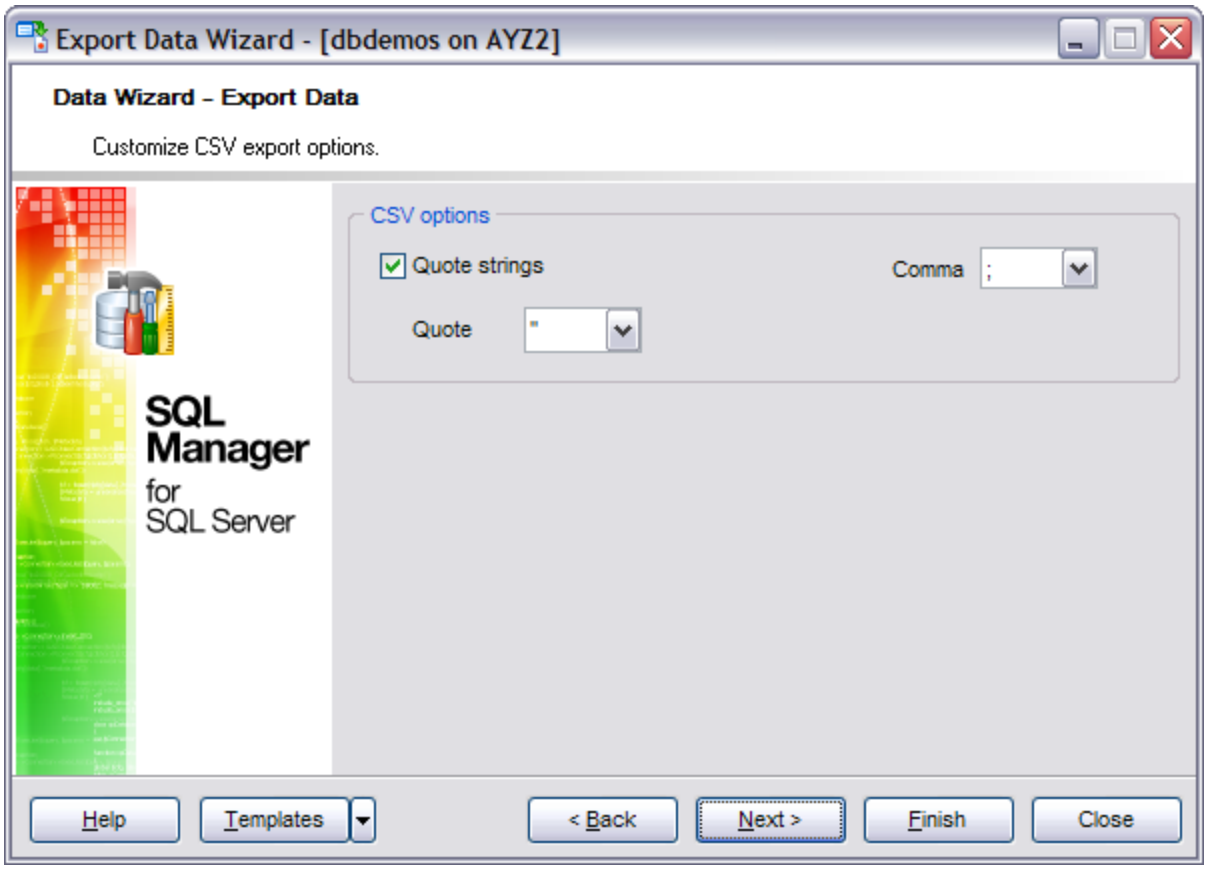
8.1.5.7 CSV options

This step allows you to set options for the target **CSV** (**.csv*) file.

Quote strings
Check this option to apply quoting for string values in the target file.

Specify the column separator using the **Comma** drop-down list and the preferable quote character using the **Quote** drop-down list.

© 2008 EMS Database Management Solutions



When you are done, click the **Next** button to proceed to Setting common export options.

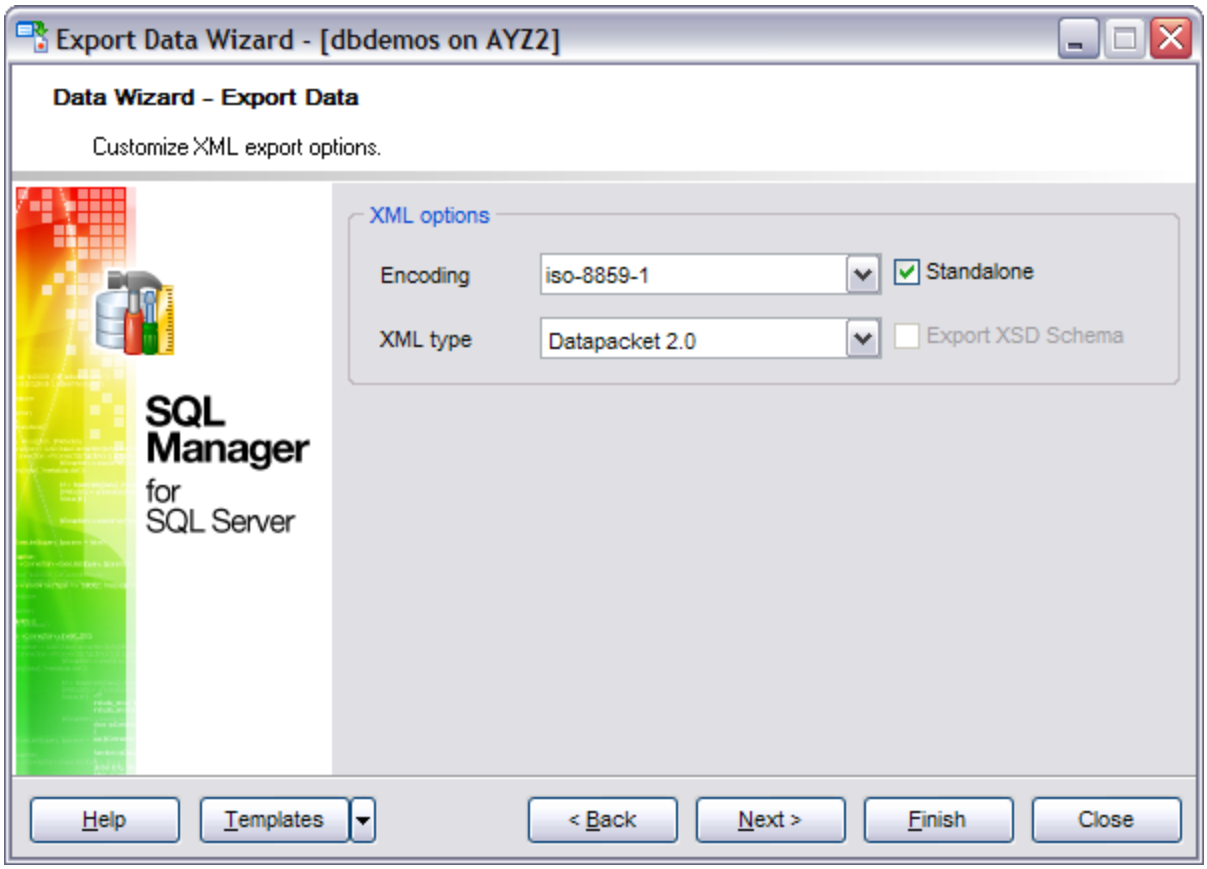
8.1.5.8 XML options

This step allows you to set options for the target **XML** (**.xml*) file.

Specify XML document encoding in the **Encoding** edit box and set the **Standalone** option on if you intend to create a standalone document.

XML type
Select the type of the result XML document: *Datapacket 2.0* or *Access*.
Conversion between generic XML documents and documents of the *XML-Datapacket (CDS)* format can be performed with the help of XML Mapper by Borland®.

© 2008 EMS Database Management Solutions



When you are done, click the **Next** button to proceed to Setting common export options.

8.1.5.9 MS Excel 2007 options

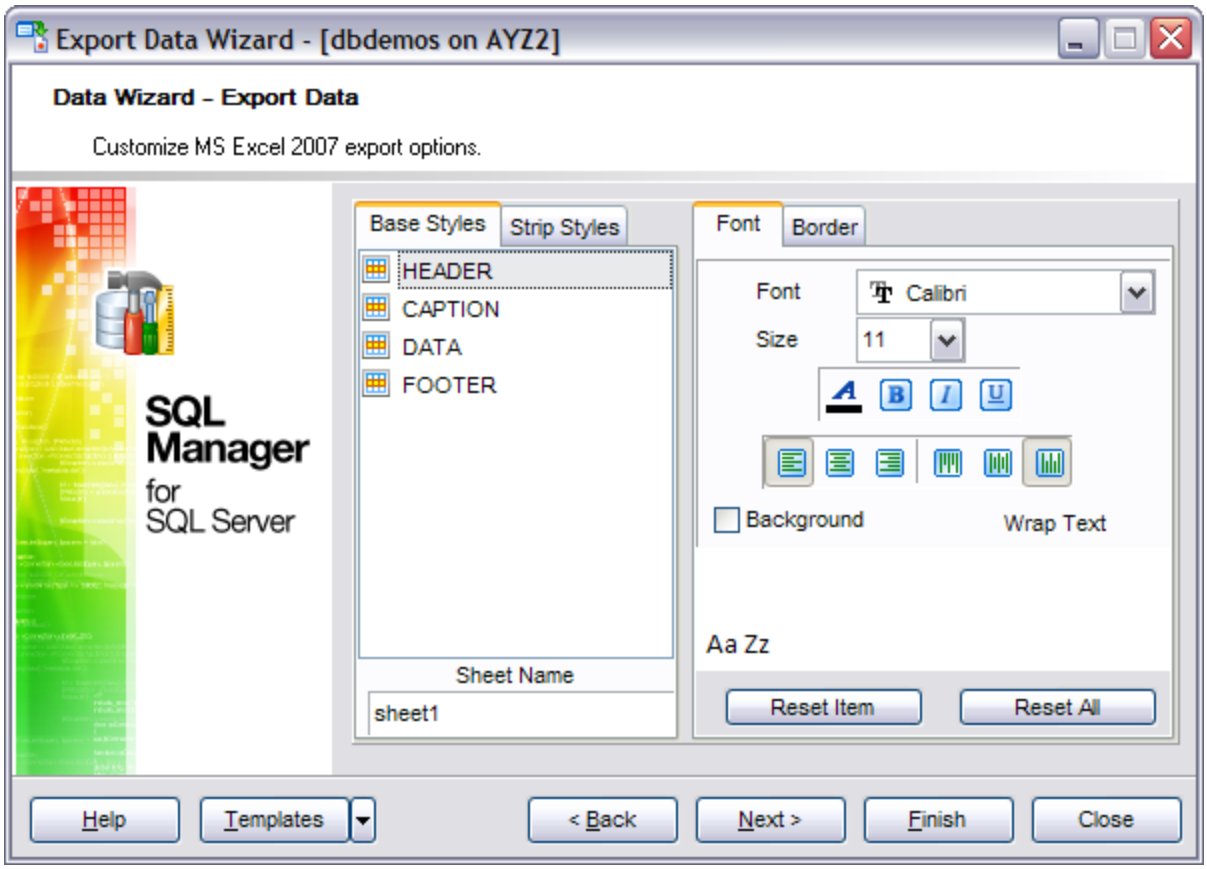
This step allows you to set options for the target **MS Excel 2007** (**.xlsx*) file.

Using the **Base Styles** tab you can set *font* and *border* options for all **elements** of the Excel 2007 sheet (*HEADER*, *CAPTION*, *DATA*, *FOOTER*). You can customize style options, such as *font* and *size*, *background* and *foreground colors*, *text alignment*, etc. for each of them by clicking the corresponding item in the list and setting the options in the right-side panel.

For your convenience the previews illustrating the changes are displayed in the **Sample Group** area within the *Base Styles* and the *Strip Styles* tabs.

If necessary, you can also specify the **sheet name** for the target Excel 2007 file.

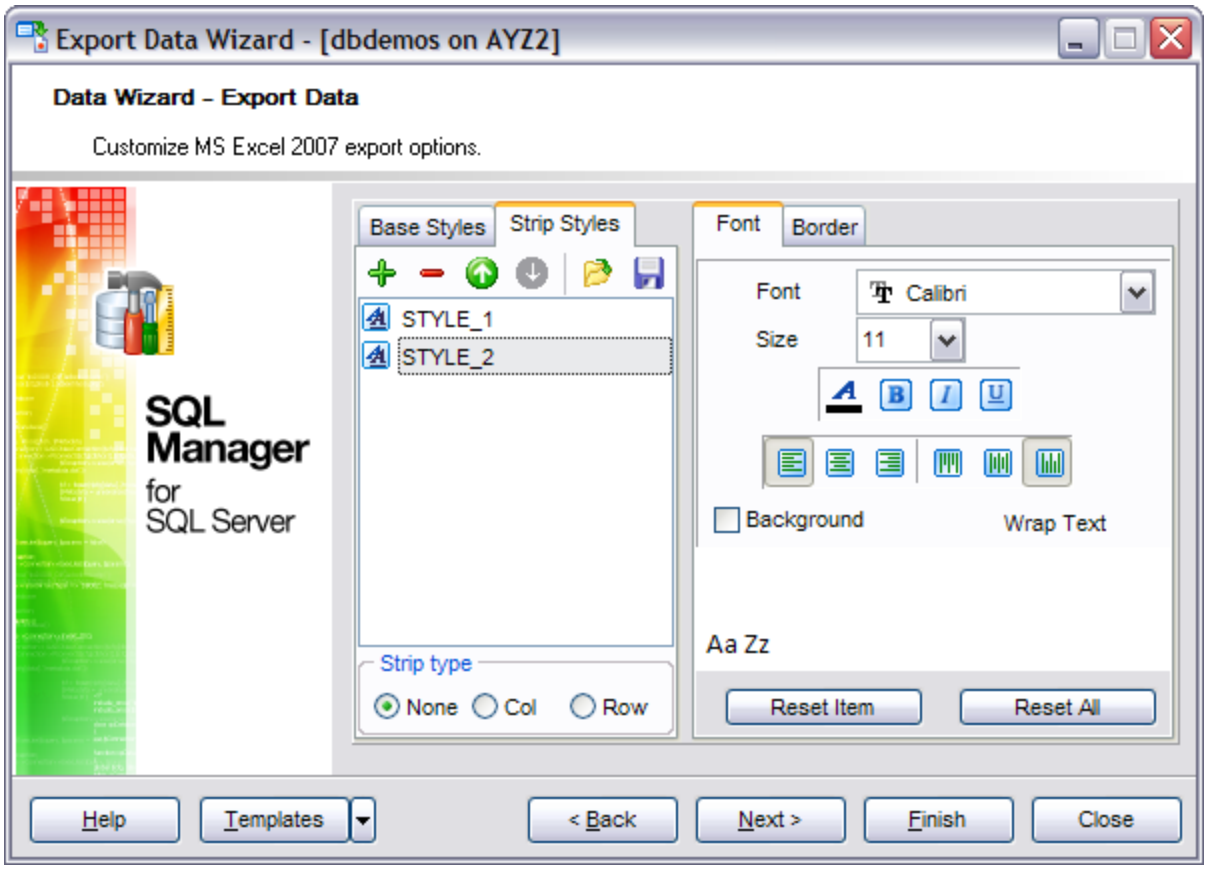
© 2008 EMS Database Management Solutions



Using the **Strip Styles** tab you can create a style template: set *font*, *size*, *background color*, *text alignment*, *wrap text* options and save them.

If you have created or loaded more than one style template, they can be ignored, or used *column-by-column* or *row-by-row* (it depends on the **Strip type** selection).

© 2008 EMS Database Management Solutions



You can reset the changes any time using the **Reset Item** and the **Reset All** buttons.

When you are done, click the **Next** button to proceed to Setting common export options.

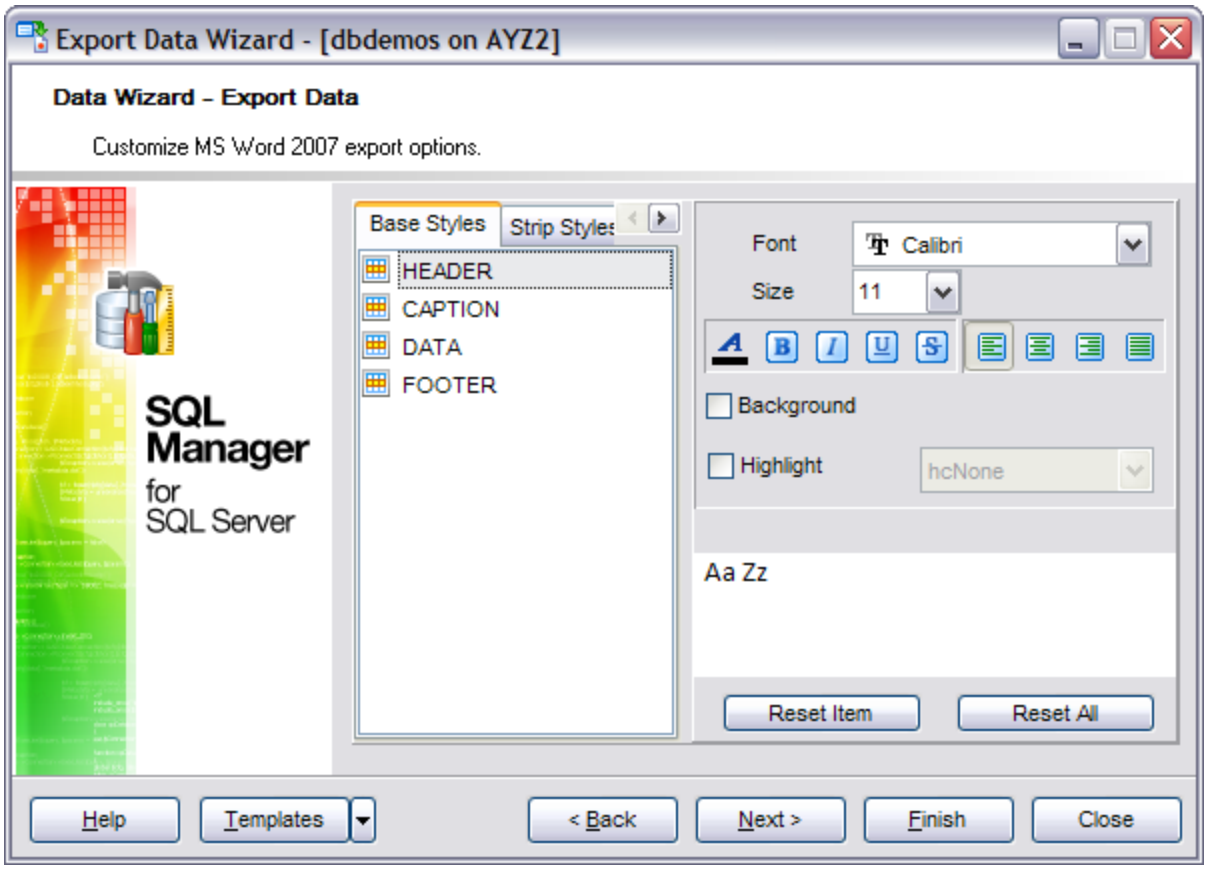
8.1.5.10 MS Word 2007 options

This step allows you to set options for the target **MS Word 2007** (*.*docx*) file.

Using the **Base Styles** tab you can set *font* options for all **elements** of the Word 2007 document (*HEADER*, *CAPTION*, *DATA*, *FOOTER*). You can customize style options, such as *font* and *size*, *background* and *foreground colors*, *text alignment*, text *highlight*, etc. for each of them by clicking the corresponding item in the list and setting the options in the right-side panel.

For your convenience the previews illustrating the changes are displayed in the **Sample Group** area within the *Base Styles* and the *Strip Styles* tabs.

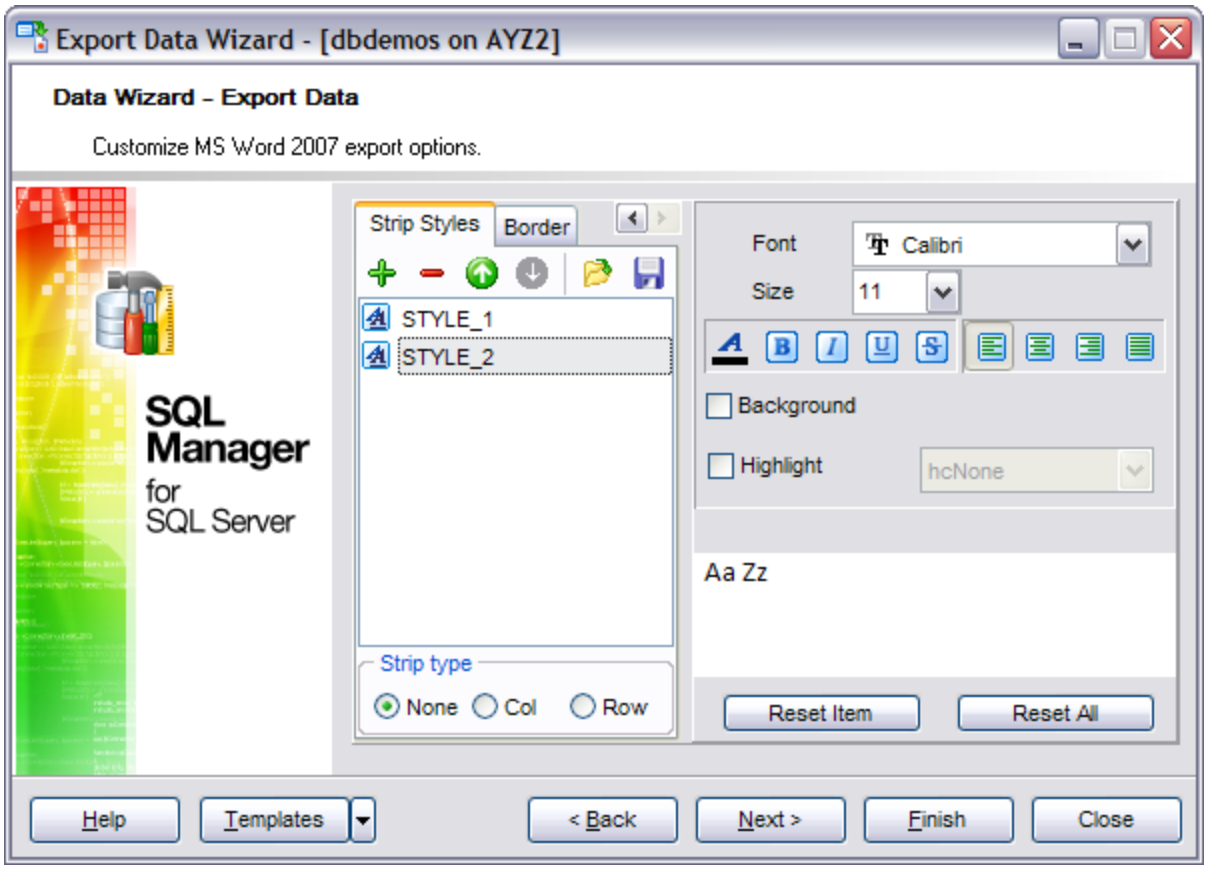
© 2008 EMS Database Management Solutions



Using the **Strip Styles** tab you can create a style template: set *font*, *size*, *background color*, *text alignment*, *highlight* options and save them.

If you have created or loaded more than one style template, they can be ignored, or used *column-by-column* or *row-by-row* (it depends on the **Strip type** selection).

© 2008 EMS Database Management Solutions

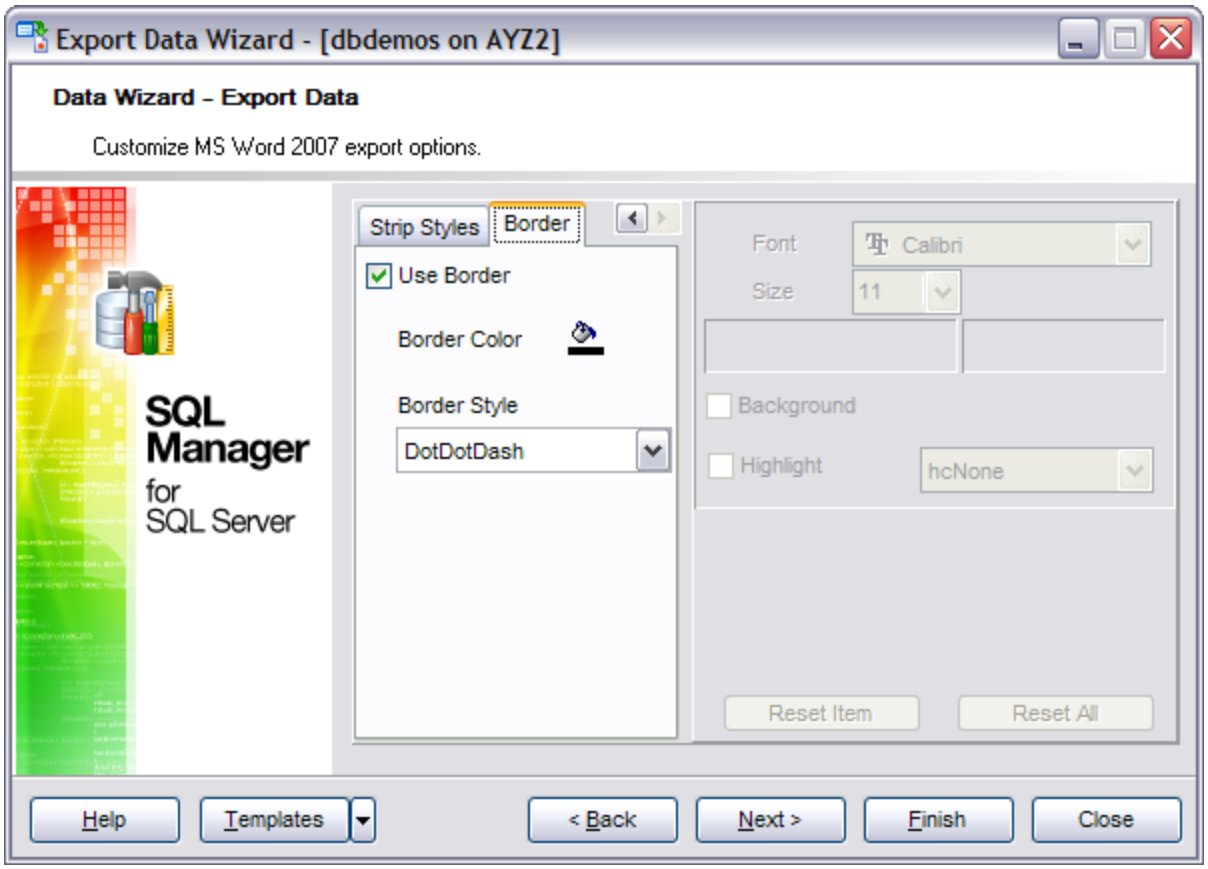


Using the **Border** tab you can enable borders in the result Word 2007 document and customize them.

Click the **Border Color** icon to select a color using the **Color** dialog where you can specify the required color from the palette.

Border Style
Use the drop-down list to select the preferable style to applied to the border: *single*, *thick*, *double*, *hairline*, *dot*, etc.

© 2008 EMS Database Management Solutions



You can reset the changes any time using the **Reset Item** and the **Reset All** buttons.

When you are done, click the **Next** button to proceed to Setting common export options.

8.1.5.11 ODF options

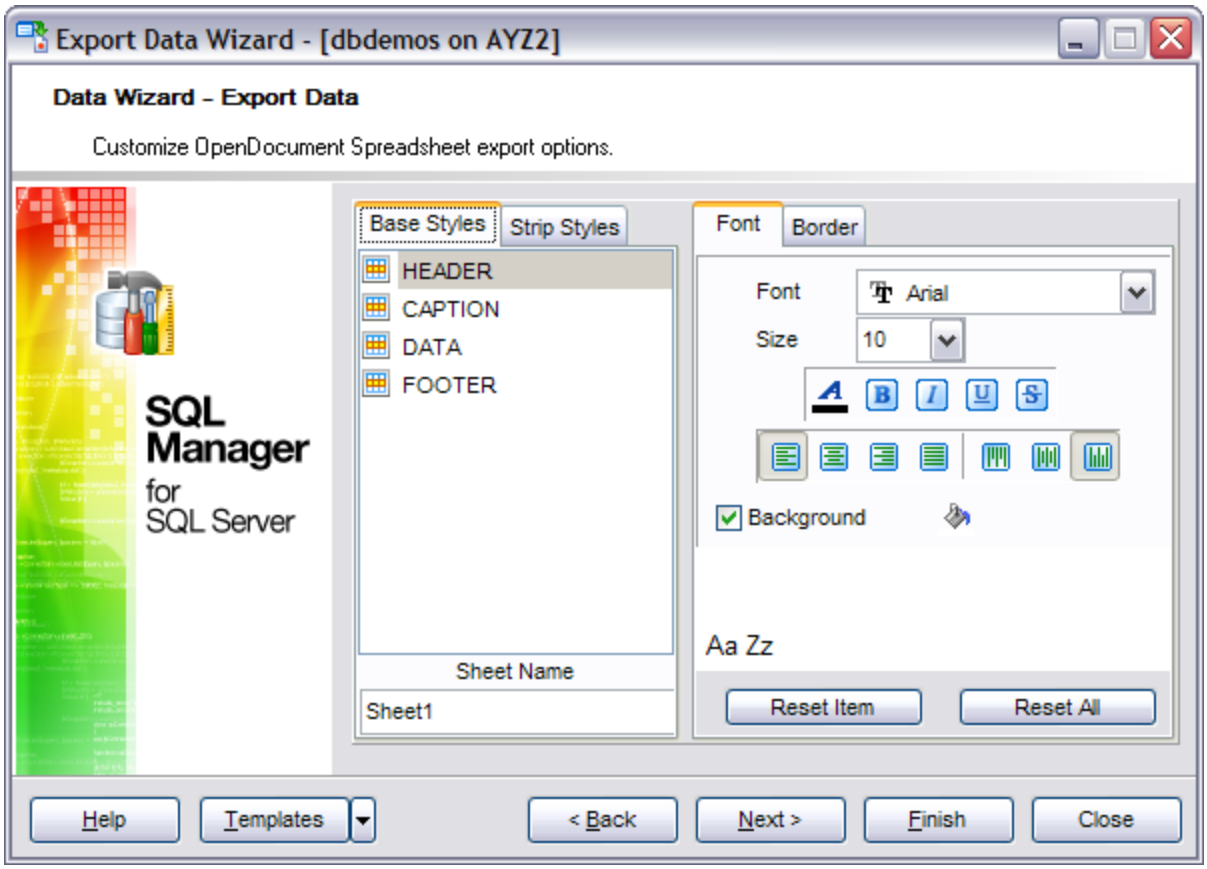
This step allows you to set options for the target **ODF Spreadsheets** (**.ods*) and **ODF text** (**.odt*) files.

Using the **Base Styles** tab you can set *font* options for all **elements** of the ODF document (*HEADER*, *CAPTION*, *DATA*, *FOOTER*). You can customize style options, such as *font* and *size*, *background* and *foreground colors*, *text alignment*, *highlight*, etc. for each of them by clicking the corresponding item in the list and setting the options in the right-side panel.

For your convenience the previews illustrating the changes are displayed in the **Sample Group** area within the *Base Styles* and the *Strip Styles* tabs.

If necessary, you can also specify the **table name** for the target ODF file.

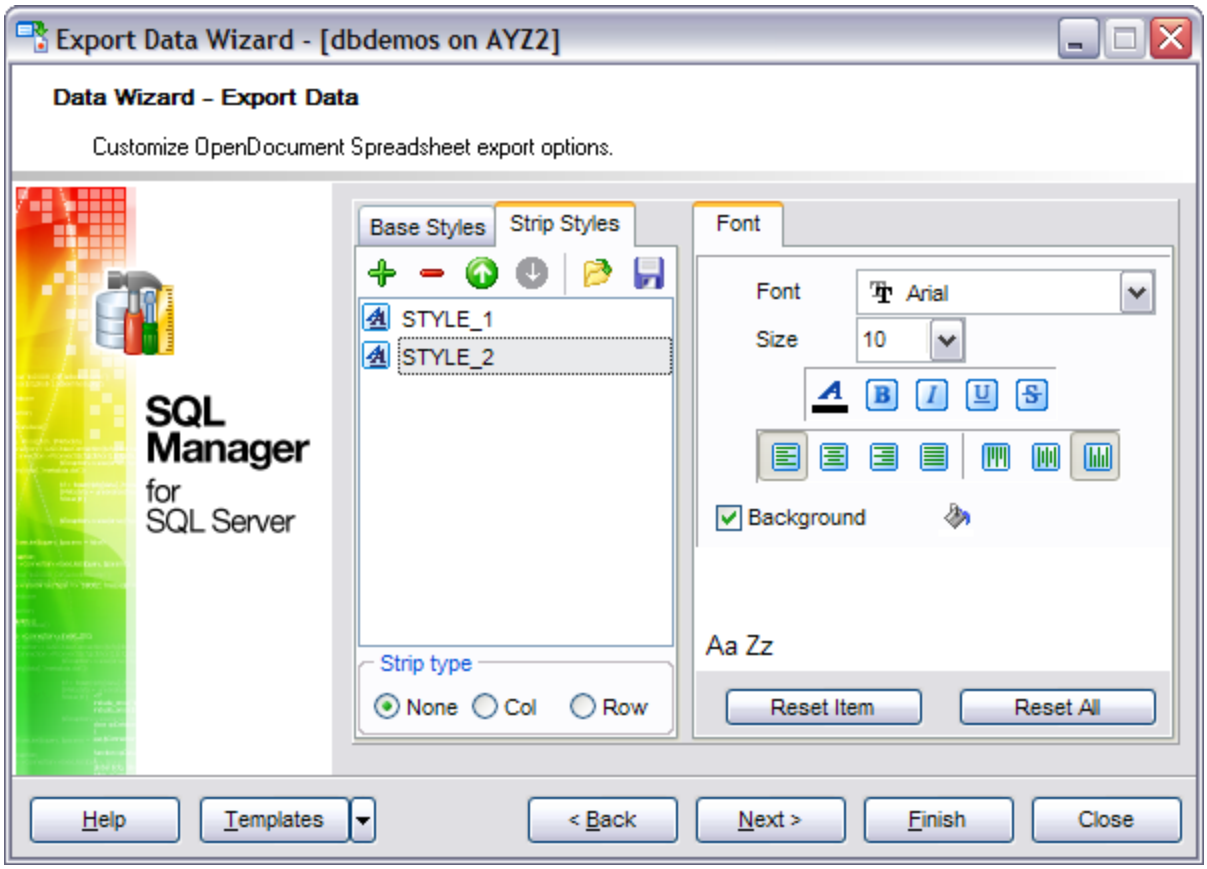
© 2008 EMS Database Management Solutions



Using the **Strip Styles** tab you can create a style template: set *font*, *size*, *background color*, *text alignment*, *highlight* options and save them.

If you have created or loaded more than one style template, they can be ignored, or used *column-by-column* or *row-by-row* (it depends on the **Strip type** selection).

© 2008 EMS Database Management Solutions

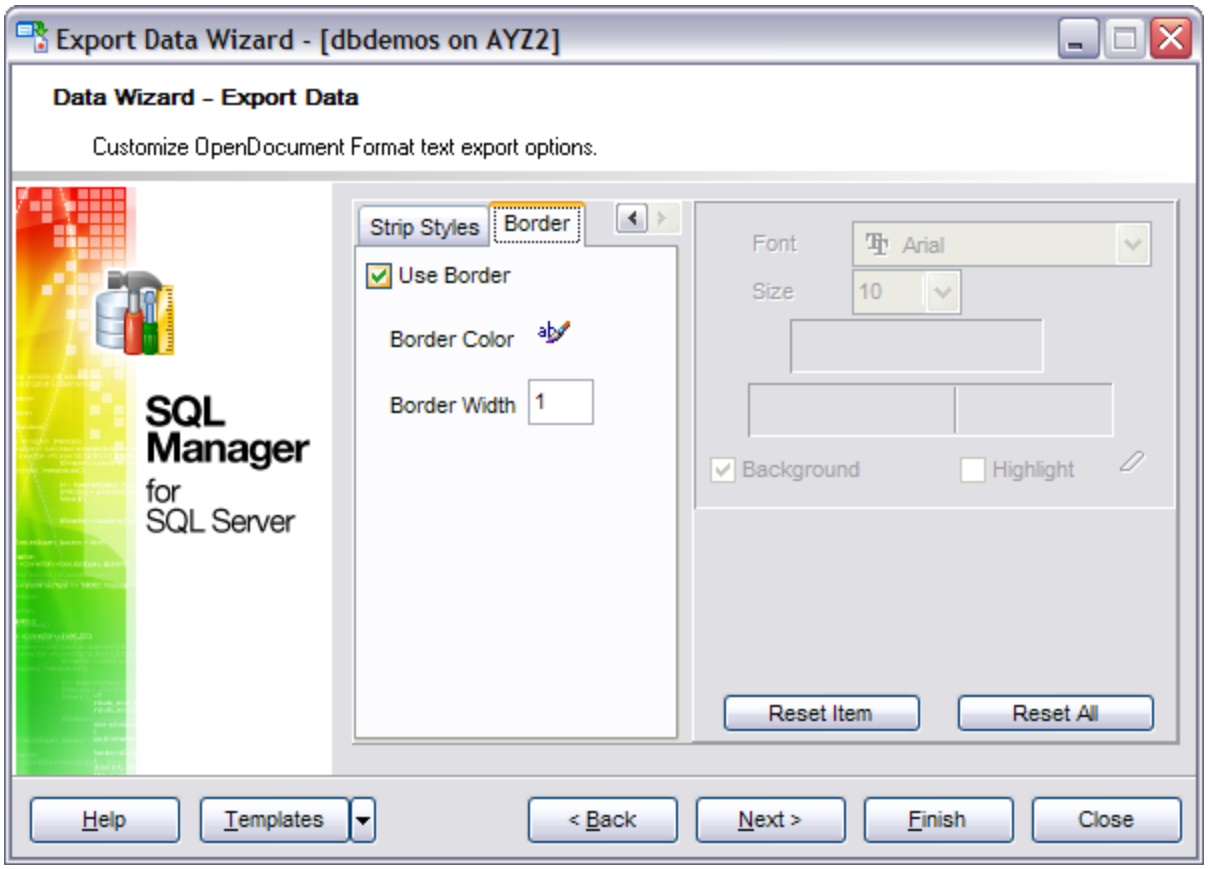


Using the **Border** tab you can enable borders in the result ODF document and customize them.

Click the **Border Color** icon to select a color using the **Color** dialog where you can specify the required color from the palette.

Border Width
Set a numeric value to indicate the width of the border.

© 2008 EMS Database Management Solutions



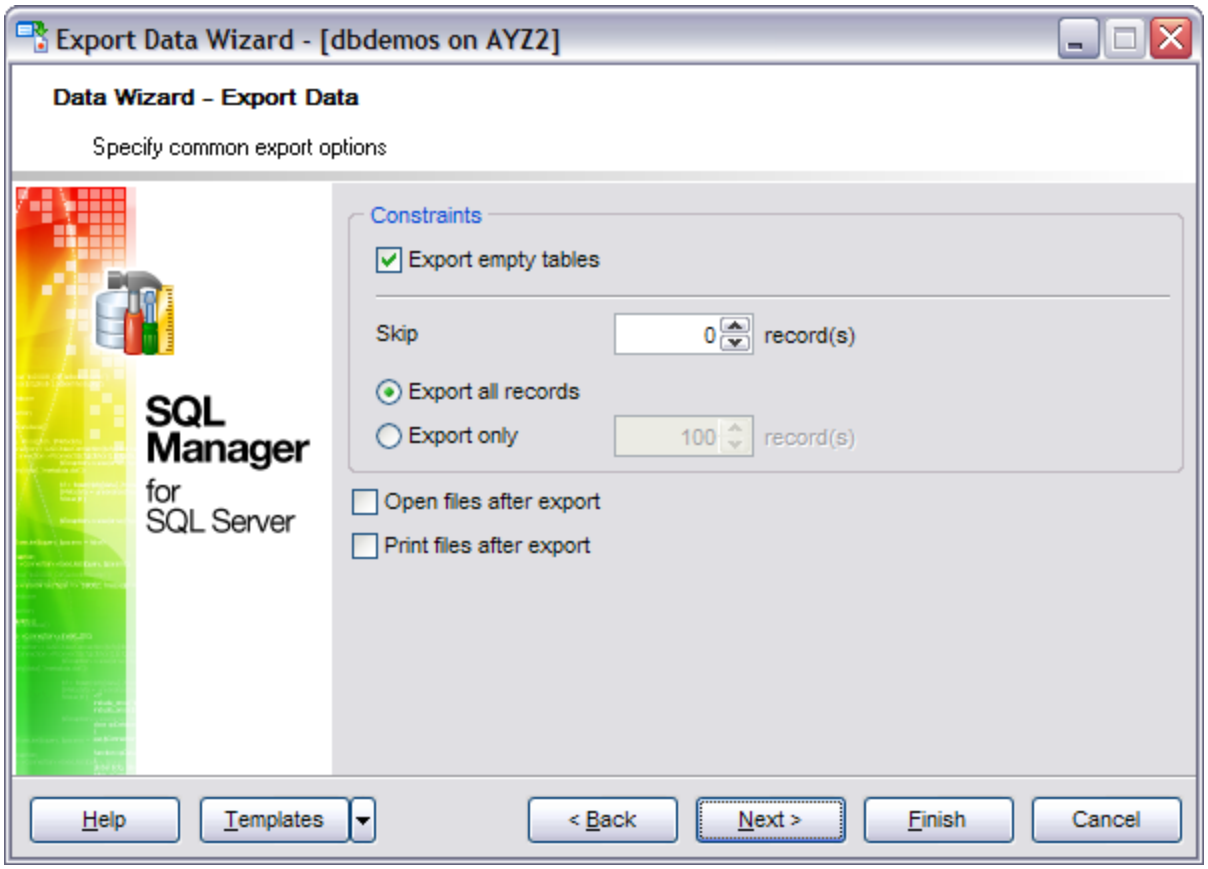
You can reset the changes any time using the **Reset Item** and the **Reset All** buttons.

When you are done, click the **Next** button to proceed to Setting common export options.

8.1.6 Setting common export options

Use this step of the wizard to set common export options. The detailed description of these options is given below.

© 2008 EMS Database Management Solutions



Export empty tables
If checked, you can export the table even if it does not contain any data.

Skip ... record(s)
Specifies the number of records to be skipped before export starts.

Export all records
Specifies that all records of the table will be exported.

Export only ... record(s)
Specifies the number of records to be exported.

Open files after export
If this option is checked, the result file will be opened with the currently associated program after the export operation is completed.

Print files after export
If this option is checked, the result file will be sent to the default printer after the export operation is completed.

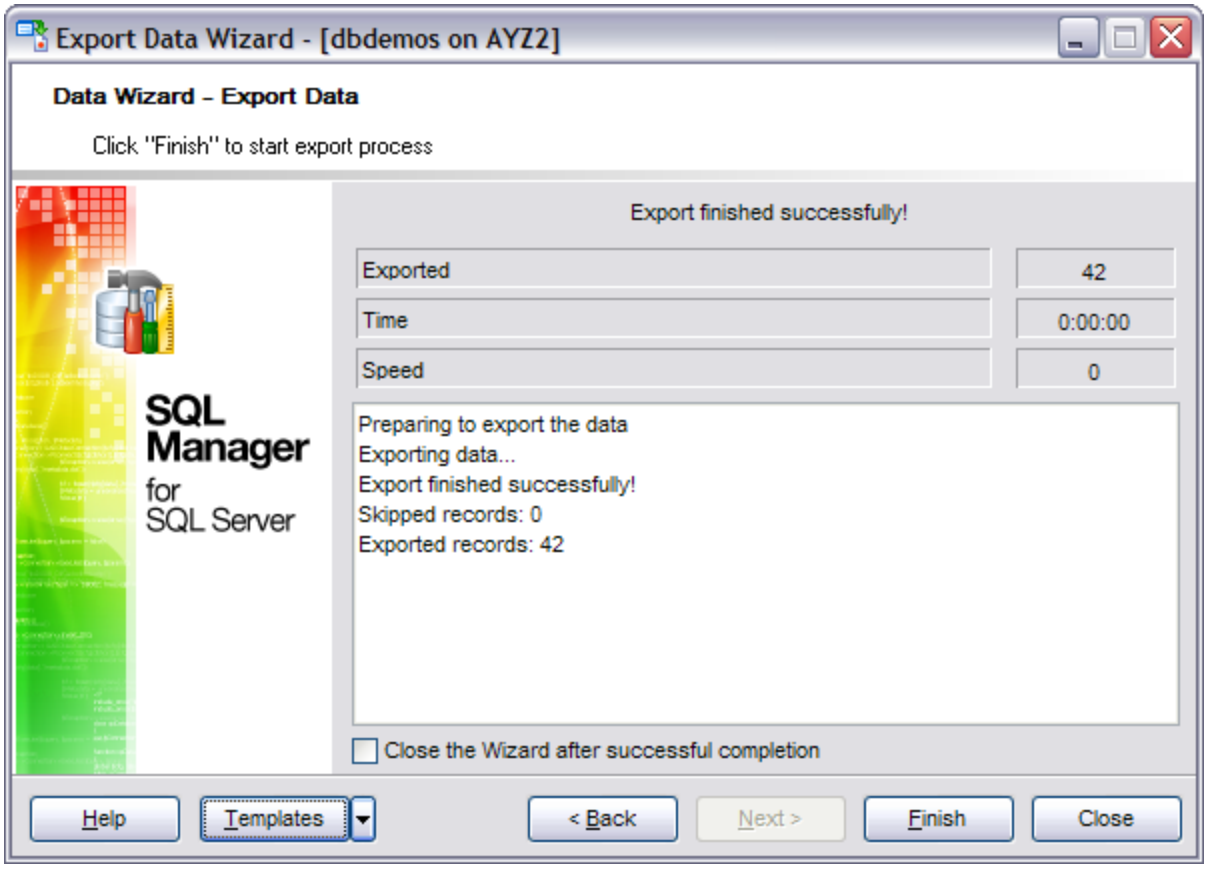
When you are done, click the **Next** button to proceed to the last step of the wizard.

© 2008 EMS Database Management Solutions

8.1.7 Exporting data

This step of the wizard is intended to inform you that all export options have been set, and you can start the export process.

Close the Wizard after successful completion
If this option is selected, the wizard is closed automatically when the export process is completed.



If necessary, you can save a template for future use.

Click the **Finish** button to run the export process.

© 2008 EMS Database Management Solutions

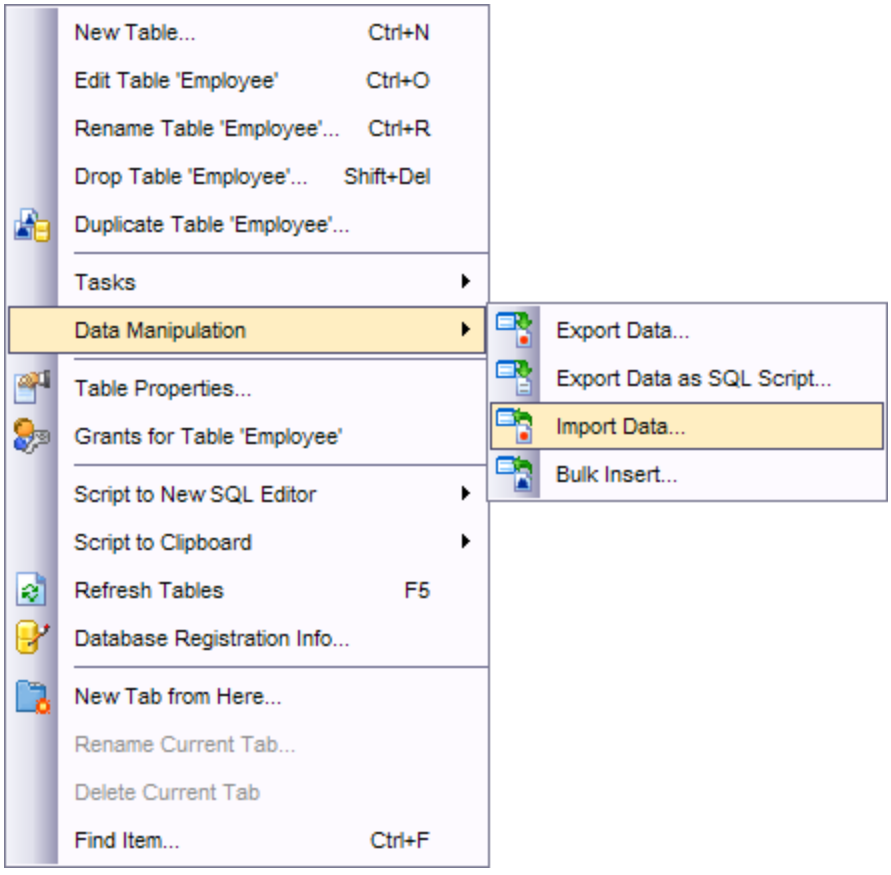
8.2 Import Data Wizard

Import Data Wizard allows you to import data to a table / view from any of supported formats (*MS Excel*, *MS Access*, *DBF*, *XML*, *TXT*, *CSV*, *HTML*, *MS Excel 2007*, *MS Word 2007*, *ODF*). You can save your settings as a template any time for future use.

To start the wizard, right-click the table/view in DB Explorer, select the **Data Manipulation** context menu group and proceed to the **Import Data...** item within this group.

Alternatively, you can open the **Data** tab of Table Editor / View Editor, right-click the grid there, then select the **Data Manipulation** context menu group and proceed to the **Import Data to <object_name>...** item within this group, or use the **Import Data** item of the Navigation bar.

- Setting source file name and format
- Selecting the source to import data from
- Setting correspondence between the source and target fields
- Adjusting common data formats
- Setting advanced field formats
- Setting import mode and data write type
- Customizing common import options
- Importing data
- Using templates



© 2008 EMS Database Management Solutions

Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**No**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

Export Data Wizard
Export as SQL Script
Bulk Insert Wizard

8.2.1 Selecting source file name and format

This step of the wizard allows you to select the source file format you need to import data from.

Source file name

Type in or use the button to specify the path to the file using the **Open file...** dialog. The file name extension changes automatically according to the selected **Import Type**.

© 2008 EMS Database Management Solutions

Import Type
Specify the format of the source file. For details refer to Supported file formats.

CSV format parameters
For CSV import you should define **Delimiter** and **Quote** settings using the corresponding drop-down lists.

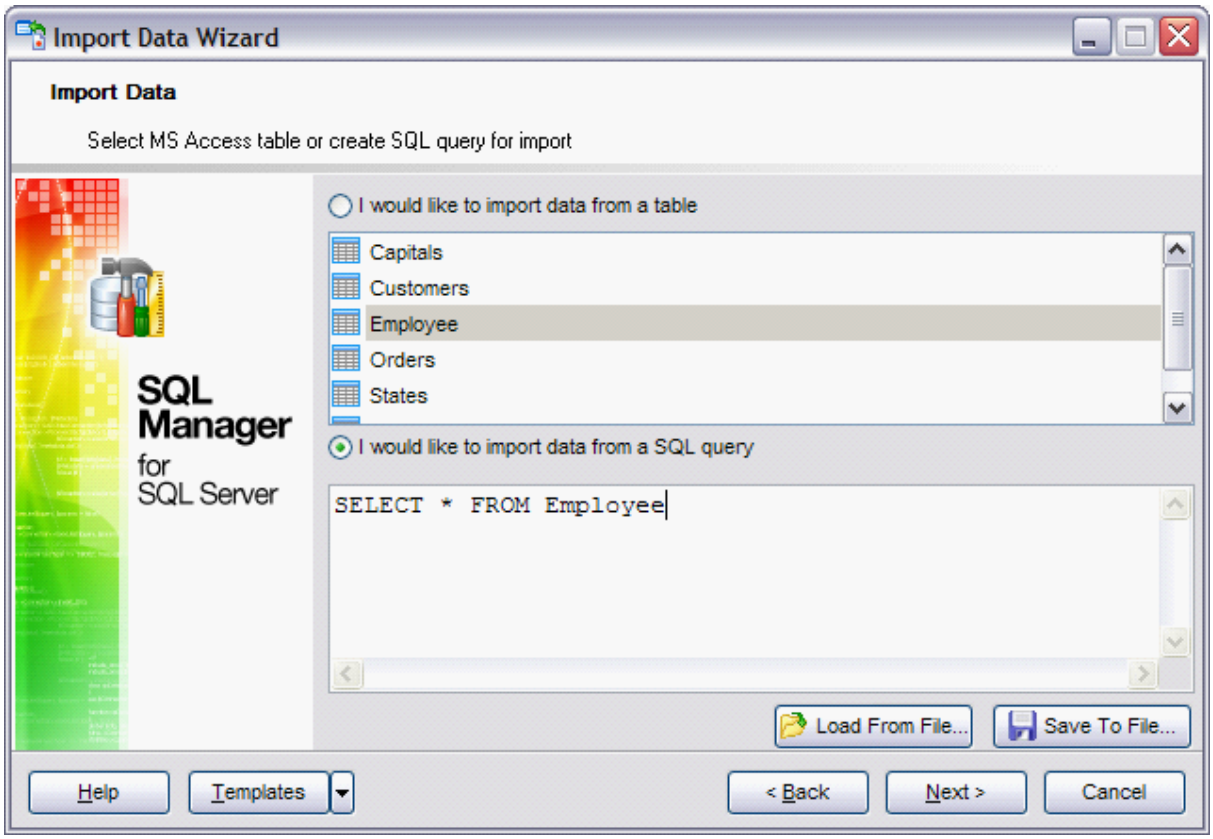
Click the **Next** button to proceed to the Setting fields correspondence step or to the Selecting data source step of the wizard if you have selected **MS Access** as the source file format.

8.2.2 Selecting data source

This step of the wizard is only available when you are importing data from *MS Access*. Select a **table** from the table list or input a **query** in the corresponding text boxes to specify the data source.

If you choose a query as the data source, you also can load a SQL query from a **.sql* file or save the current query text to a file using the **Load from File...** and the **Save to File...** buttons correspondingly.

© 2008 EMS Database Management Solutions



Click the **Next** button to proceed to the Setting fields correspondence step of the wizard.

8.2.3 Setting fields correspondence

This step of the wizard allows you **to set correspondence** between columns of the source file and fields of the target SQL Server table.

- MS Excel
- MS Access
- DBF
- XML Datapacket
- TXT
- CSV
- HTML
- XML Generic
- MS Excel/Word 2007, ODF

To get more information about the file formats, see the Supported file formats page.

© 2008 EMS Database Management Solutions

8.2.3.1 Excel

Specify ranges in the grid for the target and source fields:

- select a field of the target SQL Server table in the **Fields** list;
- proceed to the **Sheet** grid: click a column caption to select the whole column or click the row number to select the whole row;
- the selected column/row of the source file gets green highlight, and a new range indicating the source and target fields correspondence appears in the **Ranges** list;
- repeat the operation for all the fields you need to be included in the import process.

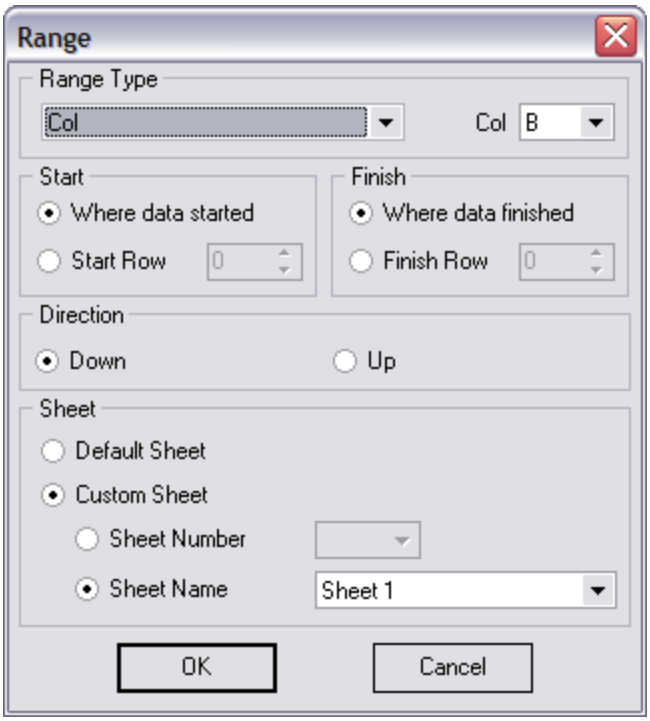
If the source Excel file and the destination SQL Server table have the same order of columns or rows, you can use the **Auto Fill Cols** or the **Auto Fill Rows** buttons to set correspondence between them automatically.

If necessary, you can choose to **skip** a defined number of the source file columns and/or rows using the **Col(s)** and **Row(s)** spin-edits.

To clear ranges for a field, select the field in the **Fields** list and press the **Clear Ranges** button.
To clear all ranges specified for the target table fields, press the **Clear All** button.

Right-click a range in the **Ranges** list to call its popup menu. Using the popup menu you can add or edit ranges manually, as well as remove them or change their order.

© 2008 EMS Database Management Solutions



Click the **Next** button to proceed to the Adjusting data formats step of the wizard.

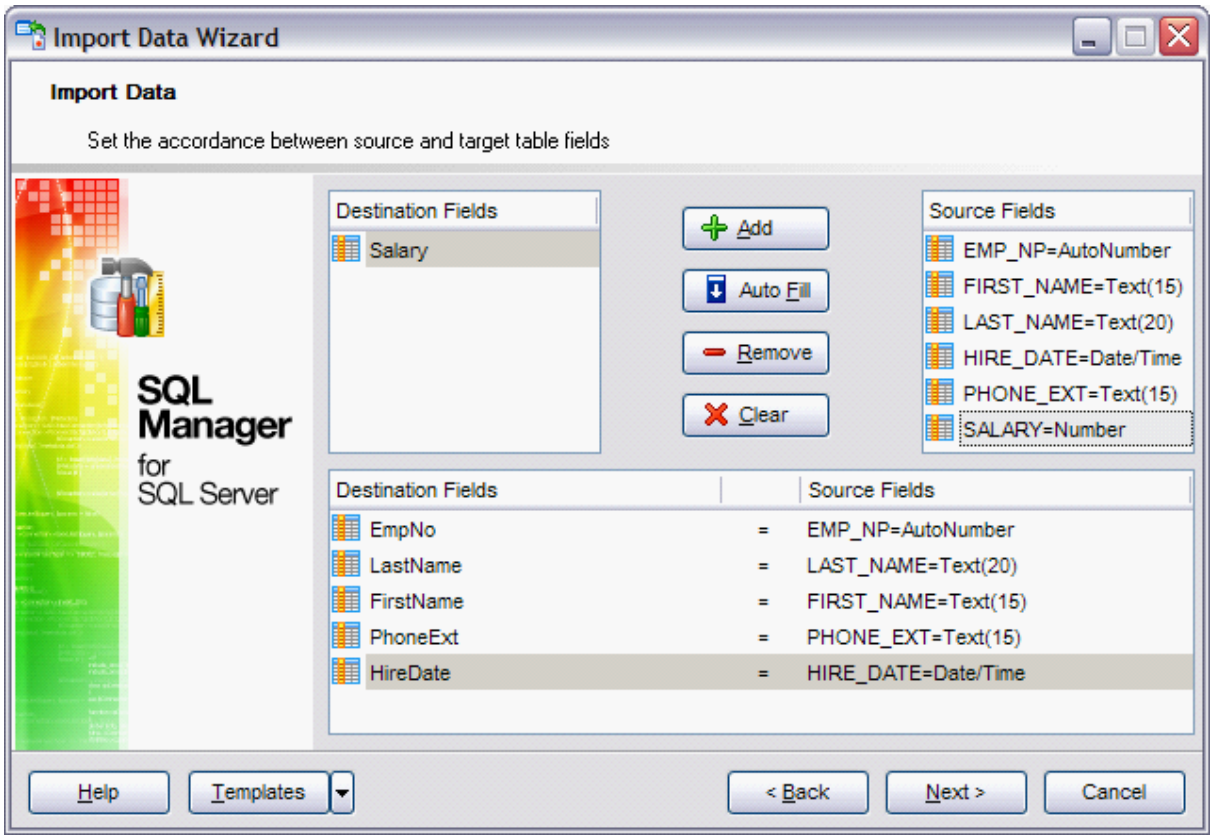
8.2.3.2 Access

Set correspondence between the source MS Access fields and the target SQL Server table fields:

- select a field of the target SQL Server table in the **Destination Fields** list;
- select the corresponding field of the source MS Access table in the **Source Fields** list;
- click the **Add** button to set correspondence between the selected fields;
- the pair of fields appears in the list below;
- repeat the operation for all the fields you need to be included in the import process.

Use the **Auto Fill** button to set correspondence between the source and target fields automatically on the basis of their order.

© 2008 EMS Database Management Solutions



To remove a correspondence, select the pair of fields in the list below and press the **Remove** button.
To remove all correspondences, press the **Clear** button.

Click the **Next** button to proceed to the Adjusting data formats step of the wizard.

8.2.3.3 DBF

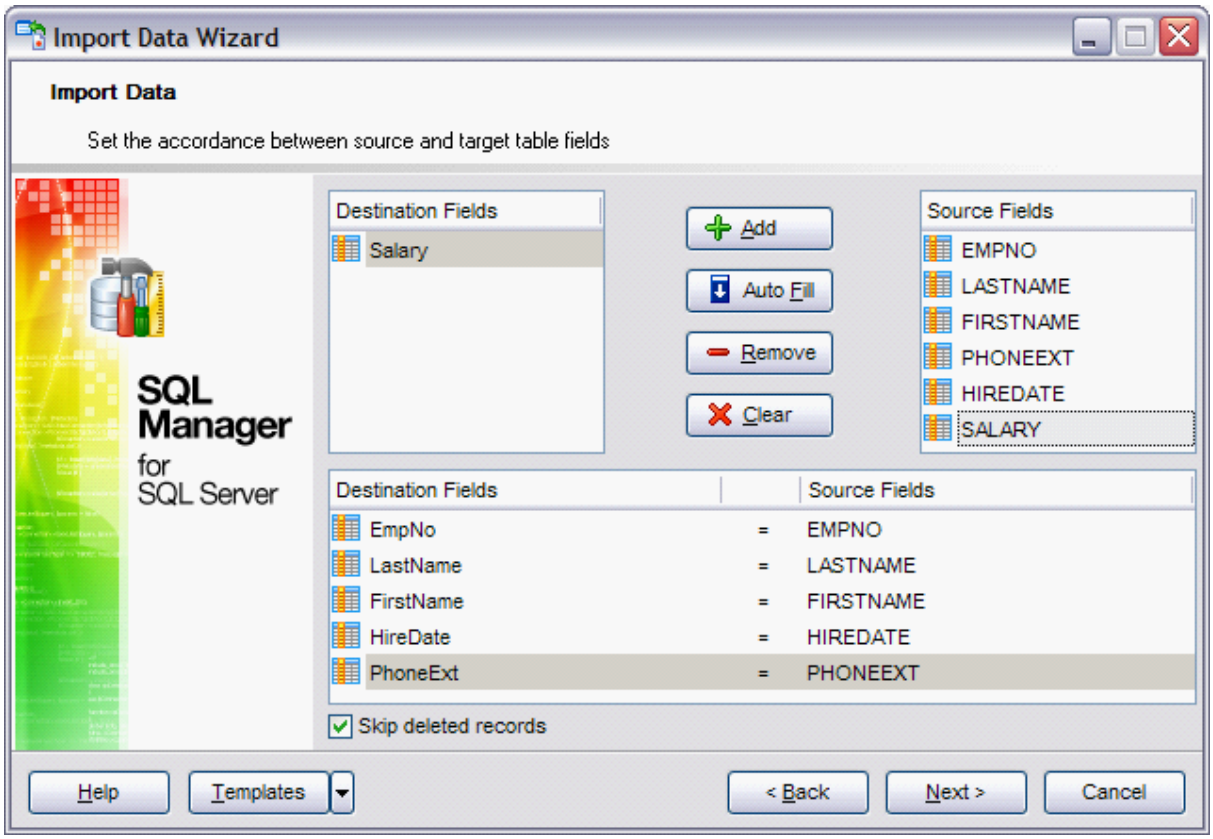
Set correspondence between the source DBF columns and the target SQL Server table fields:

- select a field of the target SQL Server table in the **Destination Fields** list;
- select the corresponding column of the source DBF table in the **Source Fields** list;
- click the **Add** button to set correspondence between the selected fields;
- the pair of fields appears in the list below;
- repeat the operation for all the fields you need to be included in the import process.

Use the **Auto Fill** button to set correspondence between the source and target fields automatically on the basis of their order.

Skip deleted records
Use the option to exclude records marked as deleted.

© 2008 EMS Database Management Solutions



To remove a correspondence, select the pair of fields in the list below and press the **Remove** button.
To remove all correspondences, press the **Clear** button.

Click the **Next** button to proceed to the Adjusting data formats step of the wizard.

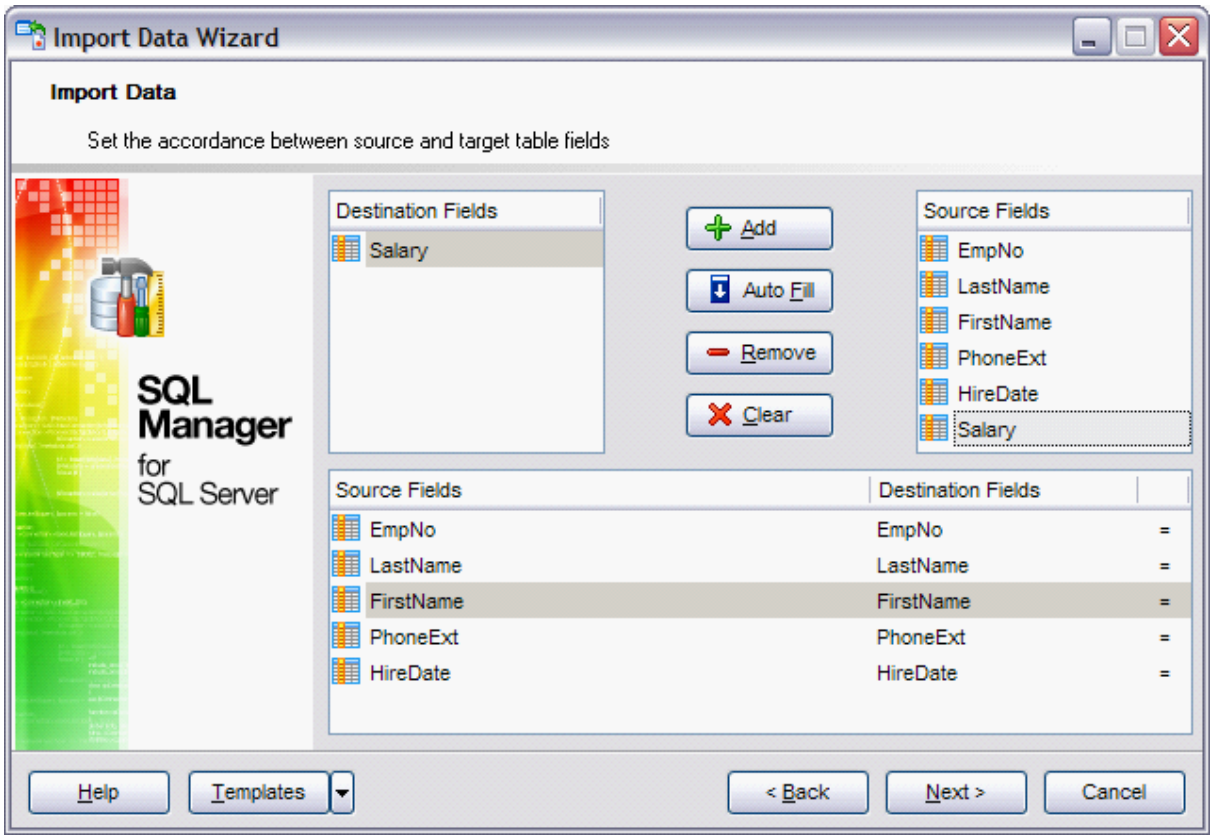
8.2.3.4 XML Datapacket

Set correspondence between the source XML columns and the target SQL Server table fields:

- select a field of the target SQL Server table in the **Destination Fields** list;
- select the corresponding column of the source XML table in the **Source Fields** list;
- click the **Add** button to set correspondence between the selected fields;
- the pair of fields appears in the list below;
- repeat the operation for all the fields you need to be included in the import process.

Use the **Auto Fill** button to set correspondence between the source and target fields automatically on the basis of their order.

© 2008 EMS Database Management Solutions



To remove a correspondence, select the pair of fields in the list below and press the **Remove** button.
To remove all correspondences, press the **Clear** button.

Click the **Next** button to proceed to the Adjusting data formats step of the wizard.

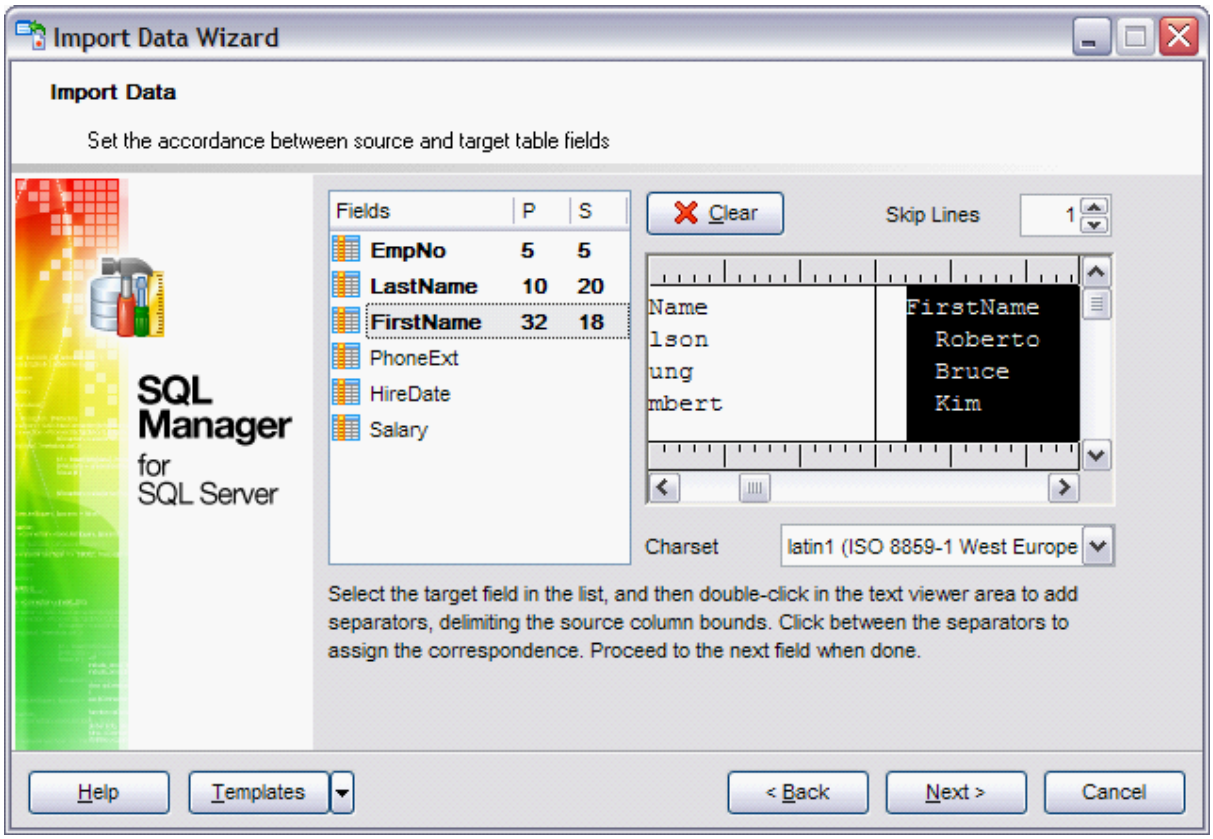
8.2.3.5 TXT

Set correspondence between the source text file columns and the target SQL Server table fields:

- select a field of the target SQL Server table in the **Fields** list;
- double-click in the text viewer area to add vertical separators delimiting the source column bounds;
- click the area between the separators to assign the column to the selected target table field - the selected source column gets black highlight;
- repeat the operation for all the fields you need to be included in the import process.

If necessary, you can choose to **skip** a defined number of the source file lines using the **Skip Lines** spin-edit.

© 2008 EMS Database Management Solutions



To clear all correspondences, press the **Clear** button.

Note: if you cannot see the content of the source text file properly, you should select the appropriate **Charset** to be used for processing data.

Click the **Next** button to proceed to the Adjusting data formats step of the wizard.

8.2.3.6 CSV

Set correspondence between the target table fields and the source CSV file columns:

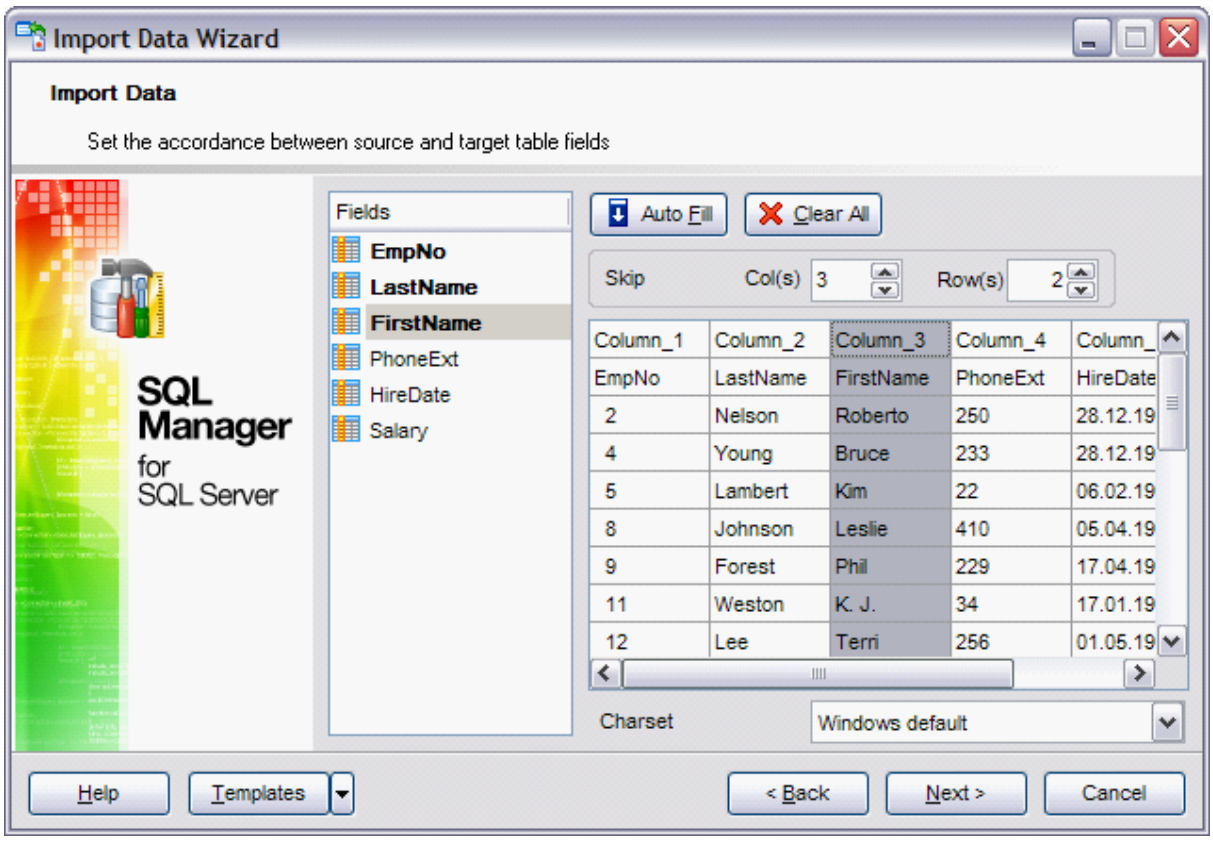
- select a field of the target SQL Server table in the **Fields** list;
- proceed to the source grid viewer area: click a caption to assign the column to the selected target table field;
- the selected column of the source file gets gray highlight;
- repeat the operation for all the fields you need to be included in the import process.

If the source CSV file and the destination SQL Server table have the same order of columns, you can use the **Auto Fill** button to set correspondence between them automatically.

Note that the CSV delimiter is specified at the Selecting source file name and format step of the wizard.

© 2008 EMS Database Management Solutions

If necessary, you can choose to **skip** a defined number of the source file rows using the **Skip** spin-edits.



To remove a correspondence, select the field in the **Fields** list and press the **Clear** button.

Click the **Next** button to proceed to the Adjusting data formats step of the wizard.

8.2.3.7 HTML

Set correspondence between the target table fields and the source HTML file columns:

- select a field of the target SQL Server table in the **Fields** list;
- proceed to the source grid viewer area: click a column to assign the column to the selected target table field;
- the selected column of the source file gets gray highlight;
- repeat the operation for all the fields you need to be included in the import process.

If the source HTML file and the destination SQL Server table have the same order of columns, you can use the **Auto Fill** button to set correspondence between them automatically.

If necessary, you can choose to **skip** a defined number of the source file rows using the **Skip** spin-edit.

© 2008 EMS Database Management Solutions

To remove a correspondence, select the field in the **Fields** list and press the **Clear** button.
To remove all correspondences, press the **Clear All** button.

Click the **Next** button to proceed to the Adjusting data formats step of the wizard.

8.2.3.8 XML Generic

In order to set mapping of a Generic XML document, you should enter the relative **XPath** (the path must be specified in the XPath format). Press the **Fill Grid** button to get the grid filled with text and attribute values of the selected node.

Note: if the source XML document contains huge amount of data, building the tree may take a long time.

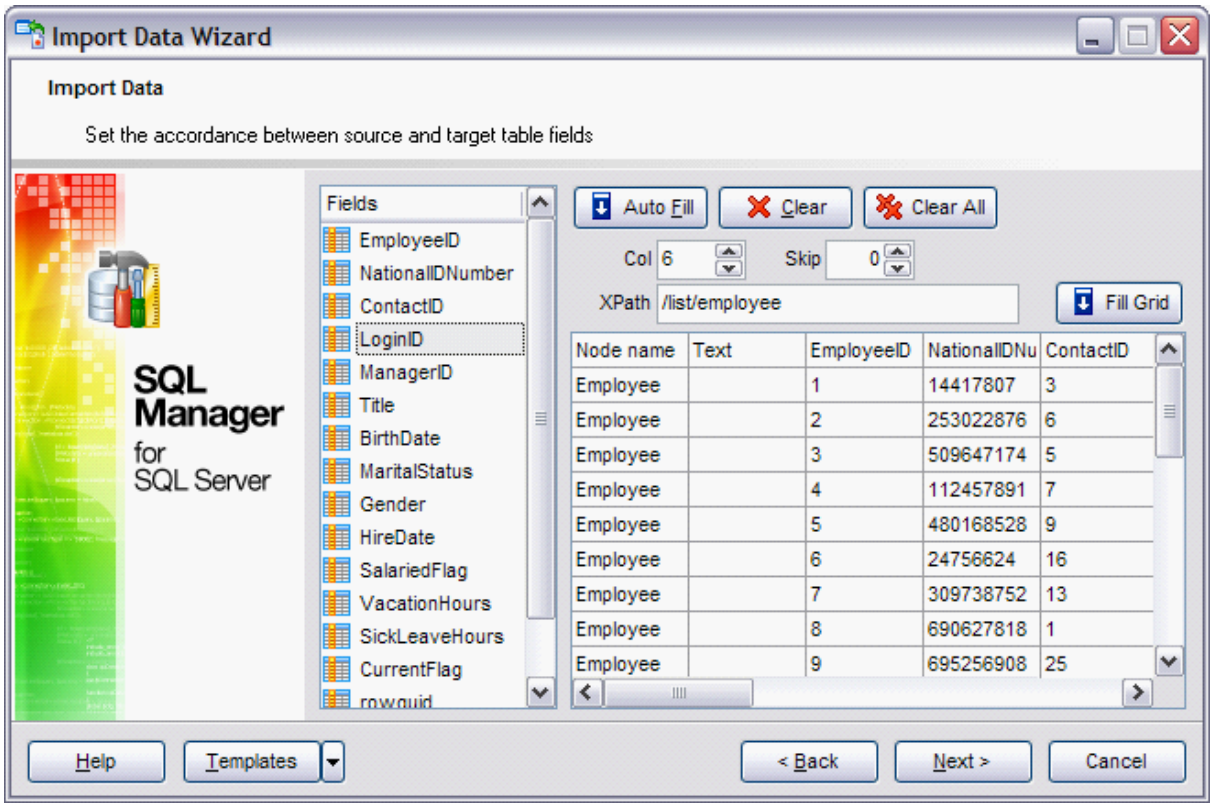
Set correspondence between the source XML file columns and the target SQL Server table fields:

- select a field of the target SQL Server table in the **Fields** list;
- proceed to the source grid viewer area: click a column to assign the column to the selected target table field;
- the selected column of the source file gets gray highlight;
- repeat the operation for all the fields you need to be included in the import process.

© 2008 EMS Database Management Solutions

You can use the **Auto Fill** button to set correspondence between the source and target fields automatically according to their order (mapping is started from the first attribute value in this case).

If necessary, you can choose to **skip** a defined number of the source file lines using the **Skip** spin-edit.



To remove a correspondence, select the field in the **Fields** list and press the **Clear** button.
To remove all correspondences, press the **Clear All** button.

Click the **Next** button to proceed to the Adjusting data formats step of the wizard.

8.2.3.9 MS Excel/Word 2007, ODF

Specify ranges in the grid for the target and source fields:

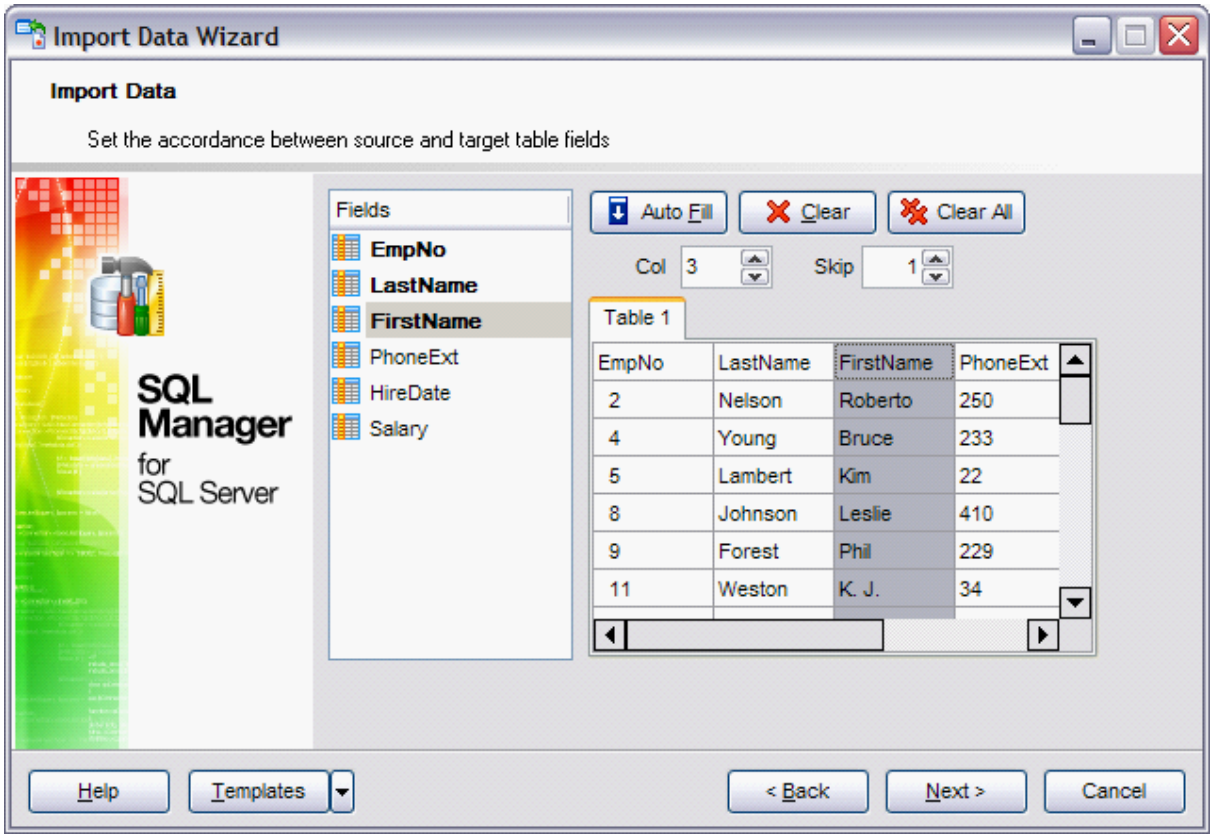
- select a field of the target SQL Server table in the **Fields** list;
- proceed to the **Sheet** grid: click a column to assign the column to the selected target table field;
- the selected column of the source file gets gray highlight;
- repeat the operation for all the fields you need to be included in the import process.

If the source file and the destination SQL Server table have the same order of columns,

© 2008 EMS Database Management Solutions

you can use the **Auto Fill** button to set correspondence between them automatically.

If necessary, you can choose to **skip** a defined number of the source file rows using the **Skip** spin-edit.



To remove a correspondence, select the field in the **Fields** list and press the **Clear** button.
To remove all correspondences, press the **Clear All** button.

Click the **Next** button to proceed to the Adjusting data formats step of the wizard.

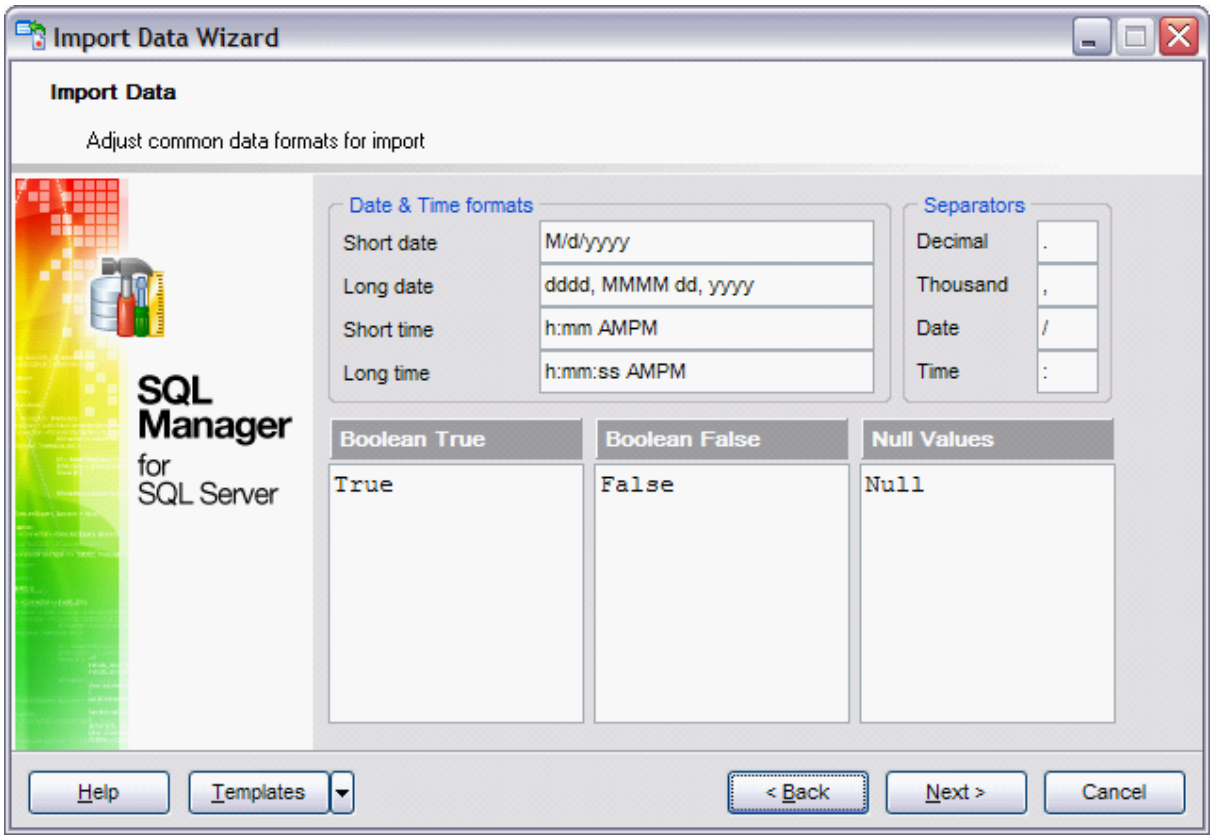
8.2.4 Adjusting data formats

This step of the wizard provides a number of options for setting common formats for all imported data:

Date & Time formats: *Short date*, *Long date*, *Short time*, *Long time*;
Separators: *Decimal*, *Thousand*, *Date*, *Time*;
Boolean True;
Boolean False;
NULL values.

© 2008 EMS Database Management Solutions

For more information refer to the Format specifiers page.



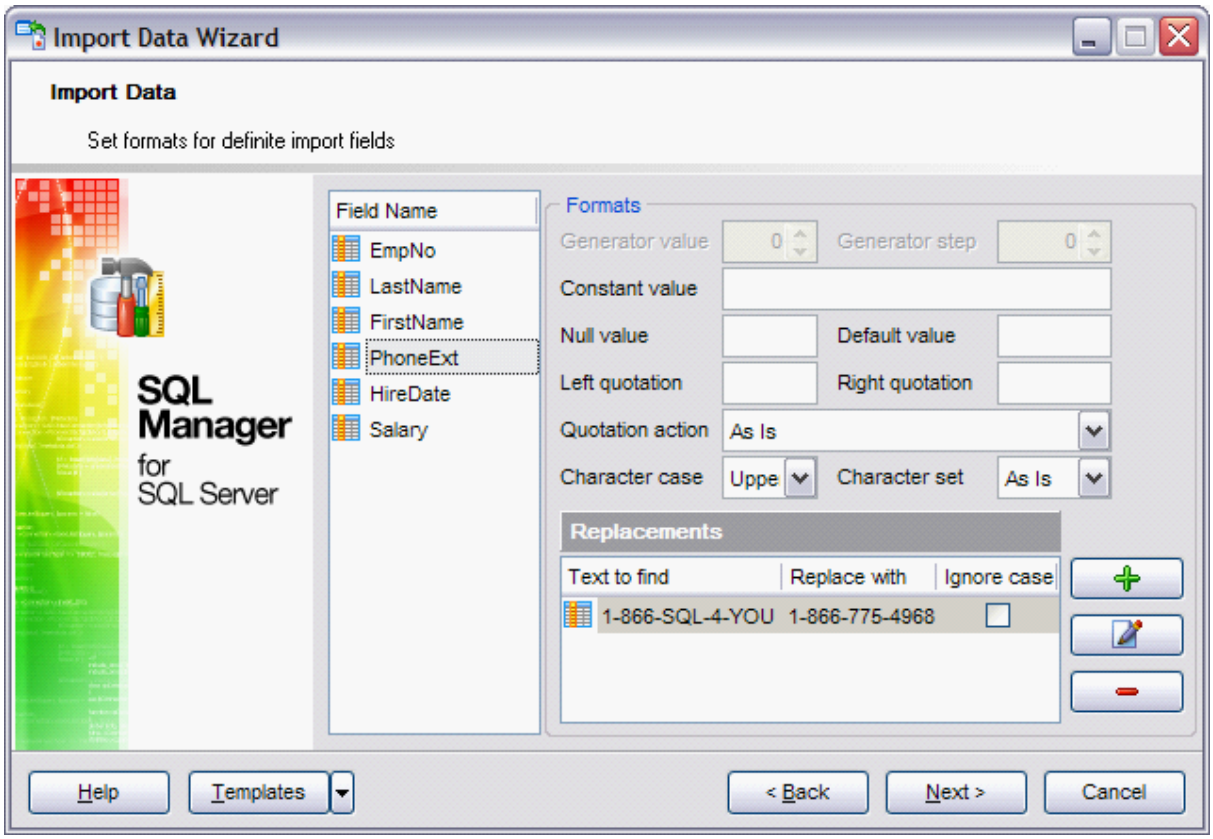
Click the **Next** button to proceed to the Setting advanced field formats step of the wizard.

8.2.5 Setting advanced field formats

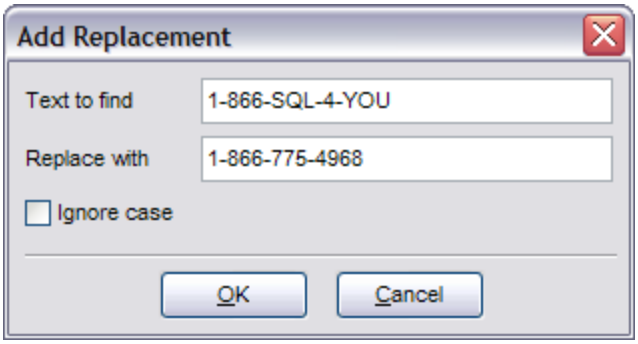
This step of the wizard allows you to set **formats** each field separately.

Select a field in the list and adjust **format options** that will be applied to this field only: *generator value*, *generator step*, *constant value*, *NULL value*, *default value*, *left/right quotation*, *character case*, *character set*.

© 2008 EMS Database Management Solutions



The **Replacements** area allows you to set the text you need to be replaced during data import into the selected field. Press the **Plus** button to specify a new replacement options using the **Add Replacement** dialog.



When you are done, click the **Next** button to proceed to the Setting import mode step of the wizard.

8.2.6 Setting import mode

This step of the wizard allows you to define the records processing mode as *Insert All*, *Insert New*, *Update*, *Update or Insert*, *Delete*, *Delete or Insert* mode:

© 2008 EMS Database Management Solutions

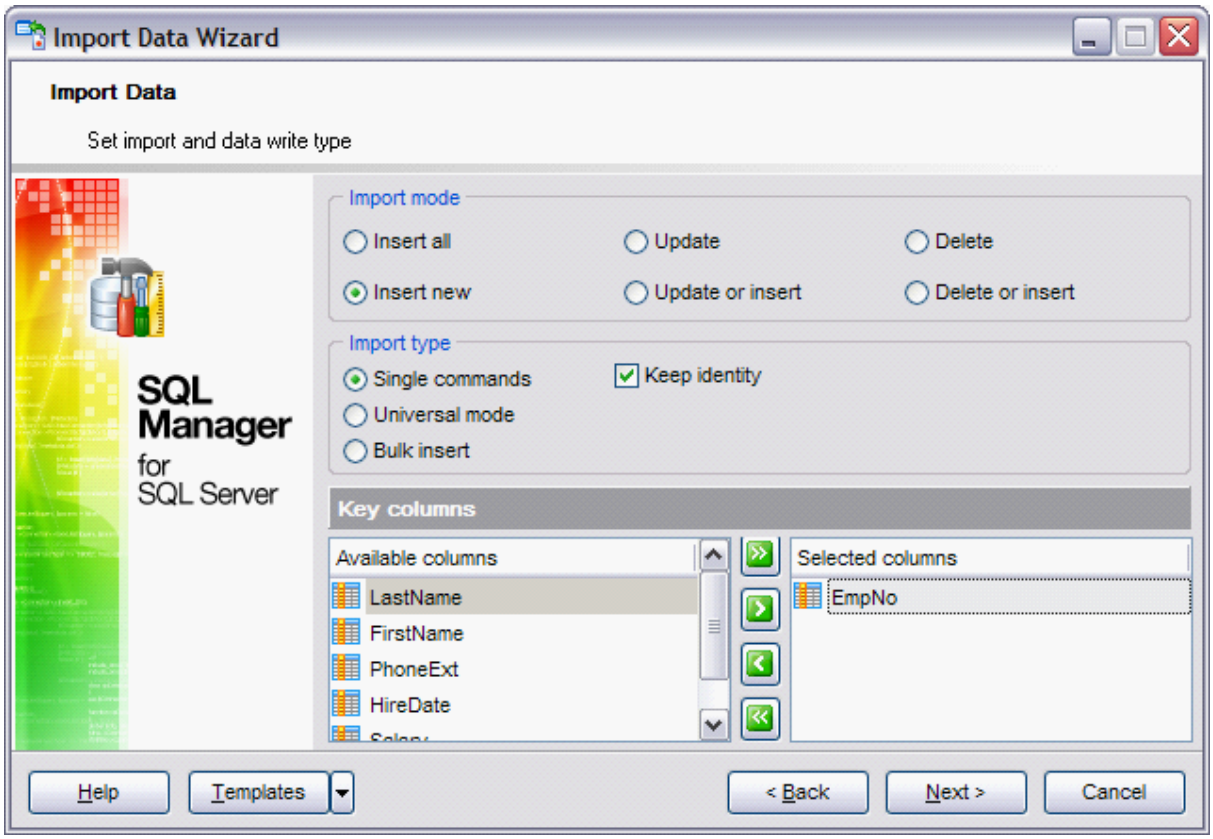
- **Insert all**: all records from the source file are inserted into the tables irrespective of whether any records exist in the destination table or not
- **Insert new**: already existing records are skipped, and new records are inserted into the destination table
- **Update**: all existing records are updated from the source file
- **Update or insert**: already existing records are updated and new records are inserted into the destination table
- **Delete**: already existing records are deleted
- **Delete or insert**: existing records are deleted and new records are inserted into the destination table

Here is an **example** of some import modes offered by Import Data Wizard:

All import modes (except for the **Insert All** mode) are based on primary key values information. In order to perform import operations with these modes used you need to have matches between the source file primary key column(s) and the destination table primary key column(s).
For example, your source file contains three rows with the primary key values *1*, *2*, *3*, but your destination table contains only two rows with the primary key values *1*, *2*.
If you use the **Update** import mode, then the rows with key values *1*, *2* will be updated.
If you use the **Update or insert** import mode, then rows *1*, *2* will be updated and the row with key value *3* will be inserted.
If you use the **Insert new** import mode, in this case only the row with key value *3* will be inserted into the destination table.
It is applied to all other import modes, except for the **Insert all** mode. For all these modes (except for the **Insert all** mode) it is obligatory to select the primary key fields. This field (or fields) is used as key field to identify specific data in the target database.

The key columns for these operations are defined in the **Key columns** area.

© 2008 EMS Database Management Solutions



Single commands / **Universal mode** / **Bulk insert** type
The *Single commands* import mode is performed with the Single Commands method used and serves to generate and execute single SQL commands on the server, whereas the *Bulk insert* mode uses native commands for a particular server: *BULK INSERT* for Microsoft® SQL Server™. With the help of the *Single commands* import mode your data can be imported considerably faster as compared to the *Universal* mode which is used for backward compatibility.

Keep identity
This option specifies that the values for an identity column are present in the file being imported.

Use **Import mode** to select whether to insert all records, or to update/delete existing ones. Note that for updating/deleting existing records in the target table you should move its key columns from the **Available columns** list to the **Selected columns** list.

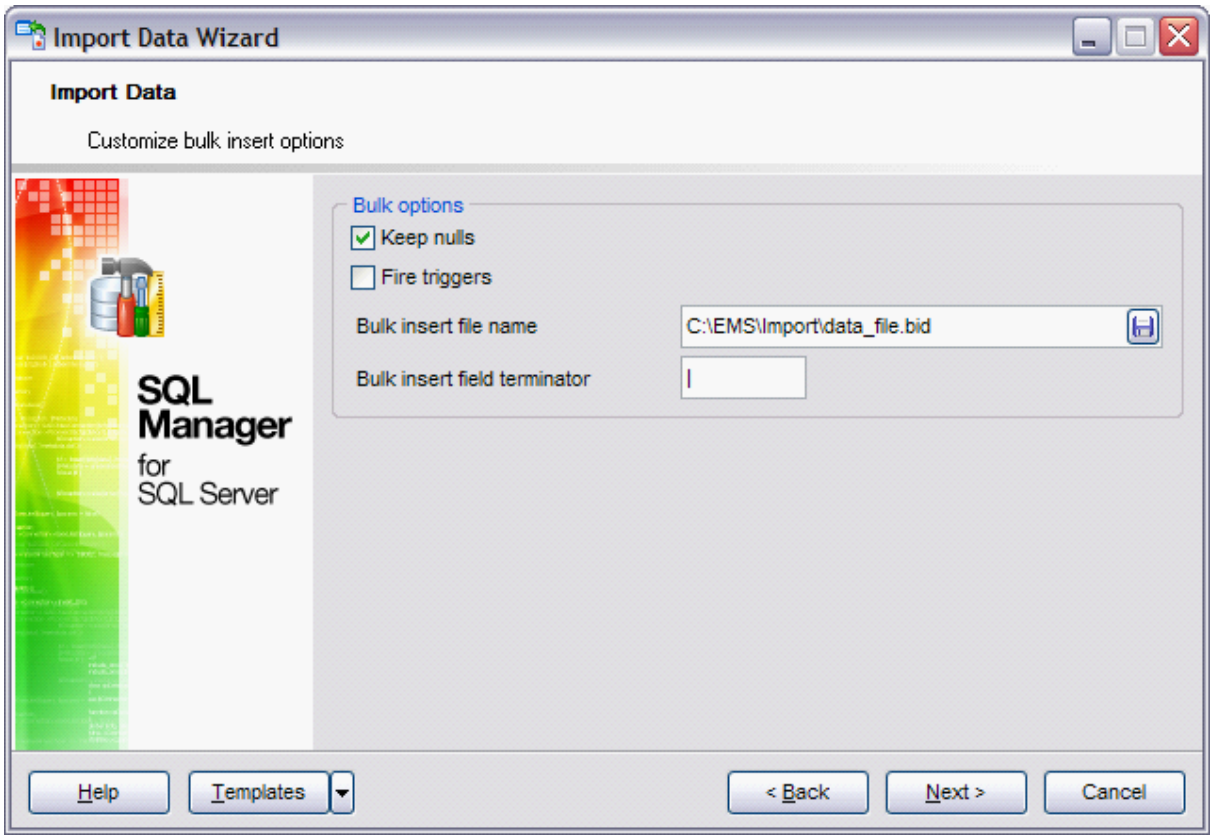
The **Key columns** area allows you to select the fields of the table to be used as the key fields for the import process.
To select a field, you need to move it from the **Available columns** list to the **Selected columns** list. Use the buttons or drag-and-drop operations to move the fields from one list to another.

© 2008 EMS Database Management Solutions

Click the **Next** button to proceed to the Customizing common options step or to the Customizing bulk insert options step of the wizard if you have selected **Bulk insert** as the import type.

8.2.7 Customizing bulk insert options

Use this step of the wizard to set bulk insert options. The detailed description of these options is given below.



Keep nulls
This option specifies that NULL columns should retain a null value during the bulk copy operation, rather than have any default values for the columns being inserted.

Fire triggers
This option specifies that any insert triggers defined on the destination table will execute during the bulk copy operation.

Bulk insert file name
Type in or use the button to specify the name and path to the bulk insert file name using the **Save As...** dialog. If *data_file* is a remotely stored file, specify the Universal Naming Convention (UNC) name.

Bulk insert field terminator

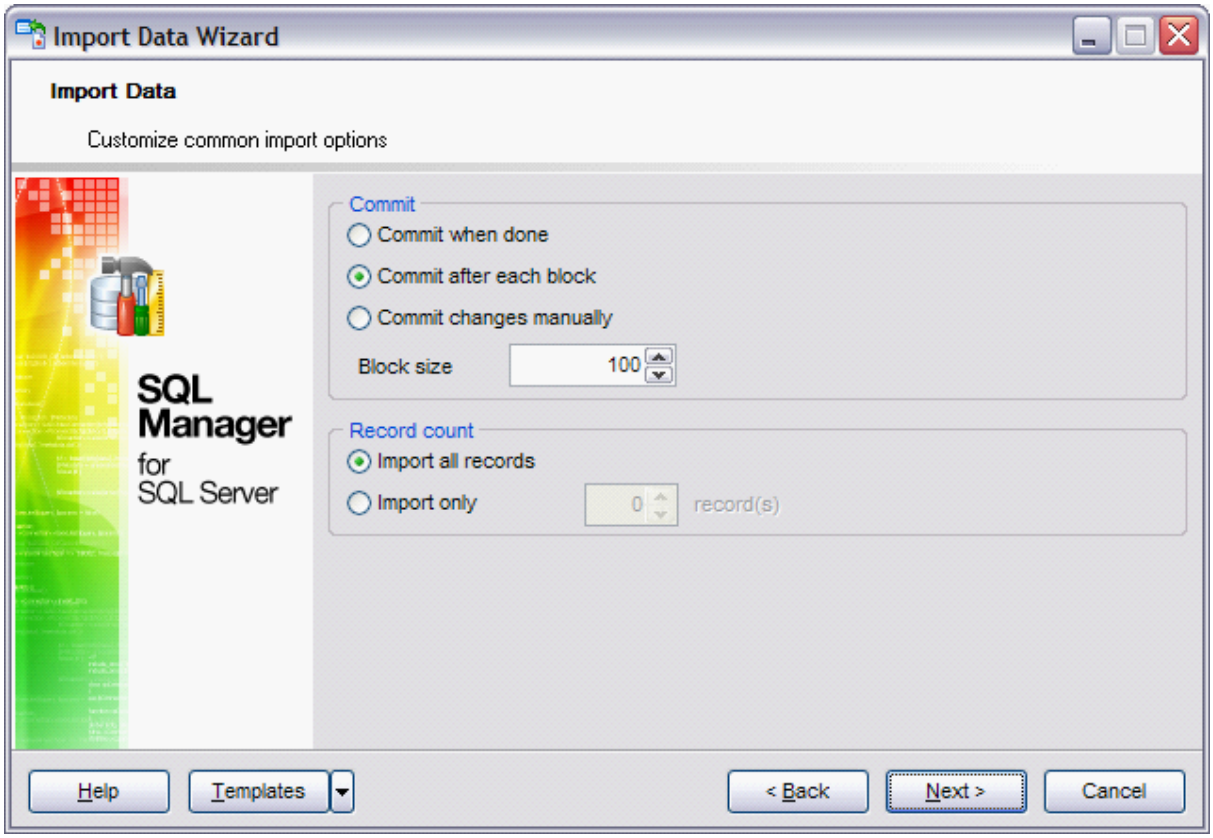
© 2008 EMS Database Management Solutions

Specify the field terminator to be used for *char* and *varchar* data types.

When you are done, click the **Next** button to proceed to the Customizing common options step of the wizard.

8.2.8 Customizing common options

Use this step of the wizard to set common import options. The detailed description of these options is given below.



Commit

Commit when done
Commits the transaction when all records are imported.

Commit after each block
Inserts the *COMMIT* statement after a defined number of records.

Commit changes manually
Select this option if you intend to commit the transaction manually.

Block size

© 2008 EMS Database Management Solutions

Use the spin-edit box to define the number of records in each committed block.

Record count

Import all records
Specifies that all records of the source file will be imported.

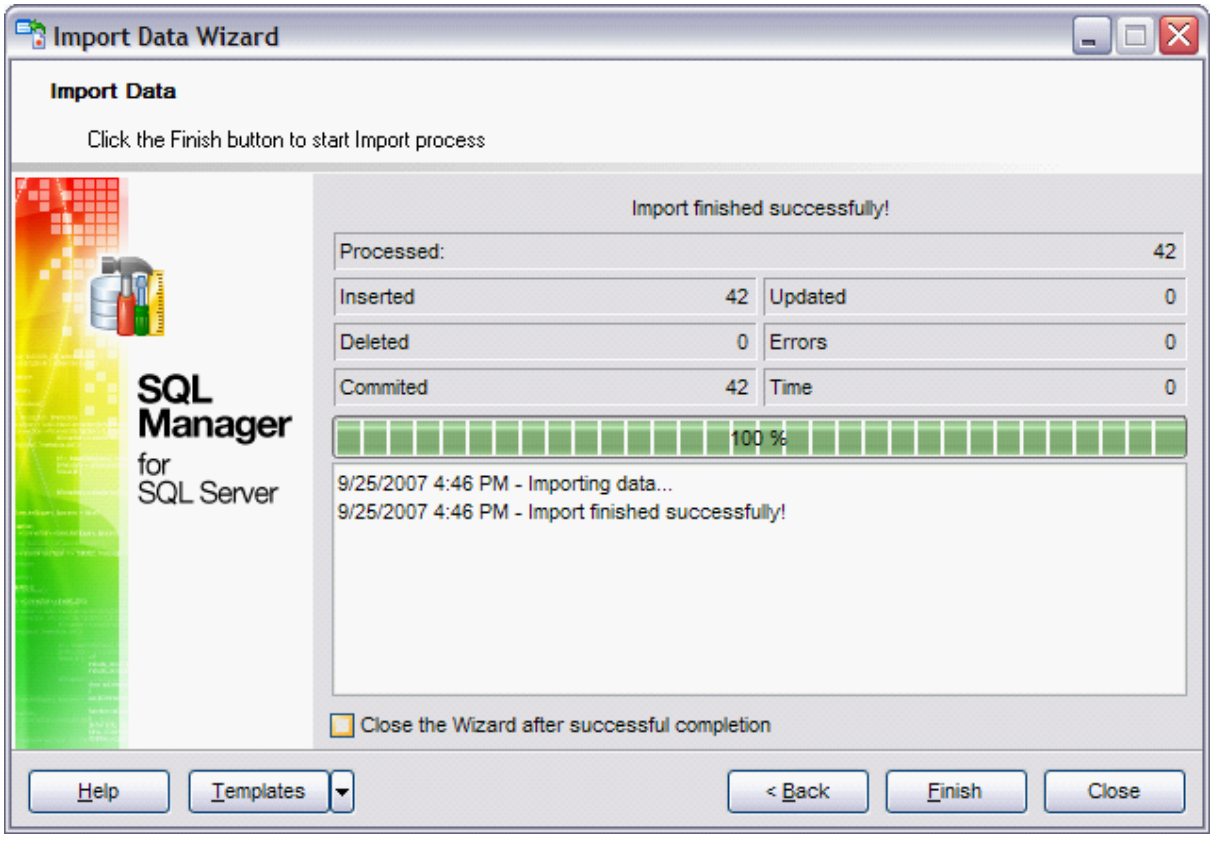
Import only ... record(s)
Specifies the number of records to be imported.

When you are done, click the **Next** button to proceed to the last step of the wizard.

8.2.9 Importing data

This step of the wizard is intended to inform you that all import options have been set, and you can start the import process.

Close the Wizard after successful completion
If this option is selected, the wizard is closed automatically when the import process is completed.



If necessary, you can save a template for future use.

© 2008 EMS Database Management Solutions

Click the **Finish** button to run the import process.

© 2008 EMS Database Management Solutions

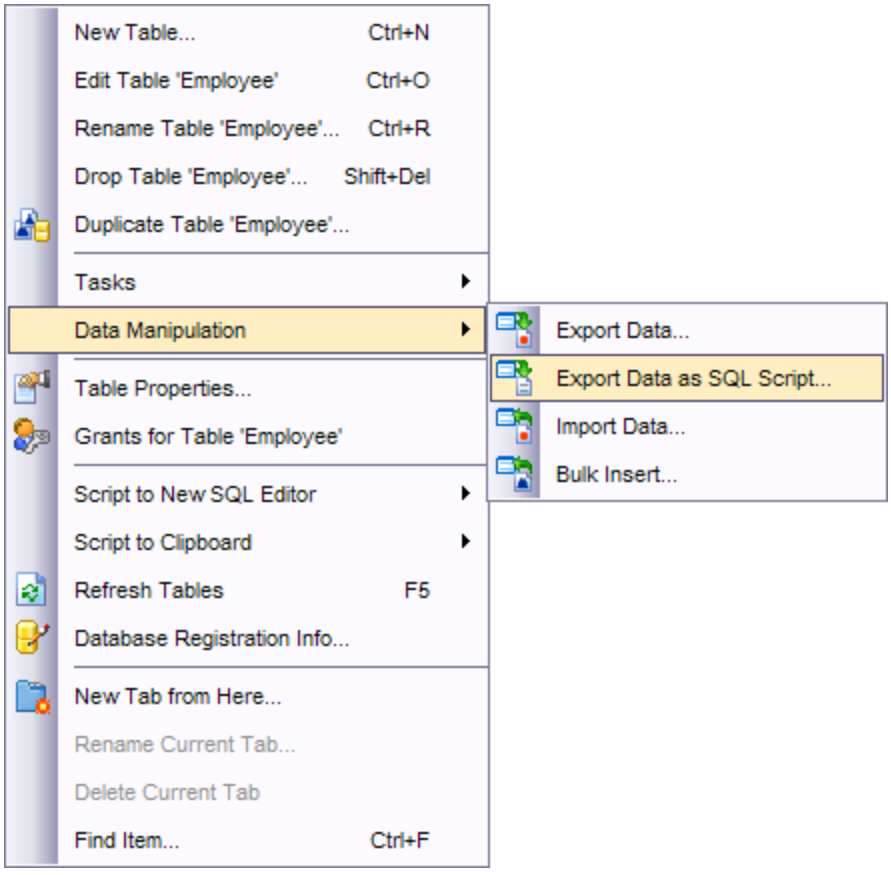
8.3 Export as SQL Script

Export as SQL Script Wizard allows you to export data from a table / view or from a query result to SQL script as a number of INSERT statements. You can save your settings as a template any time for future use.

To start the wizard, right-click the object in DB Explorer, select the **Data Manipulation** context menu group and proceed to the **Export Data as SQL Script...** item within this group.

Alternatively, you can open the **Data** tab of Table Editor / View Editor or the **Result(s)** tab of SQL Editor / Query Builder, right-click the grid there, then select the **Data Manipulation** context menu group and proceed to the **Export <object_name> as SQL Script...** item within this group, or use the **Export as SQL Script** item of the Navigation bar.

- Defining the type of destination server
- Setting destination file name
- Setting BLOB options
- Selecting field to export
- Editing the result table definition
- Setting export options
- Exporting as SQL Script
- Using templates



© 2008 EMS Database Management Solutions

Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**No**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

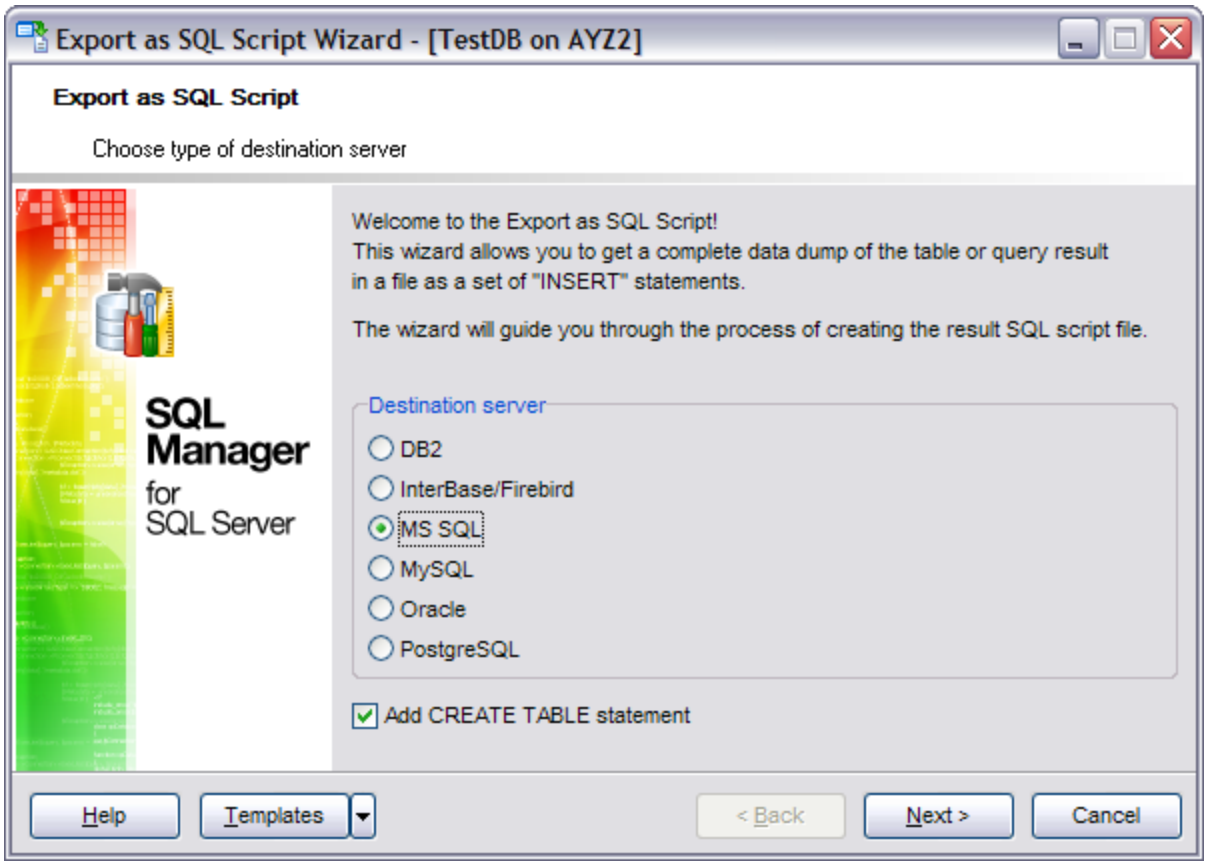
Export Data Wizard
Import Data Wizard
Bulk Insert Wizard

8.3.1 Defining the type of destination server

This step of the wizard allows you to define the type of **destination server** you need to export data for. The result script will be generated in compliance with the specifications of the selected server type:

- *DB2*
- *InterBase/Firebird*
- *Microsoft® SQL Server™*
- *MySQL*
- *Oracle*
- *PostgreSQL*

© 2008 EMS Database Management Solutions



Add CREATE TABLE statement
Check this option to add the *CREATE TABLE* statement to the result script.

Click the **Next** button to proceed to the Setting destination file name step of the wizard.

8.3.2 Setting destination file name

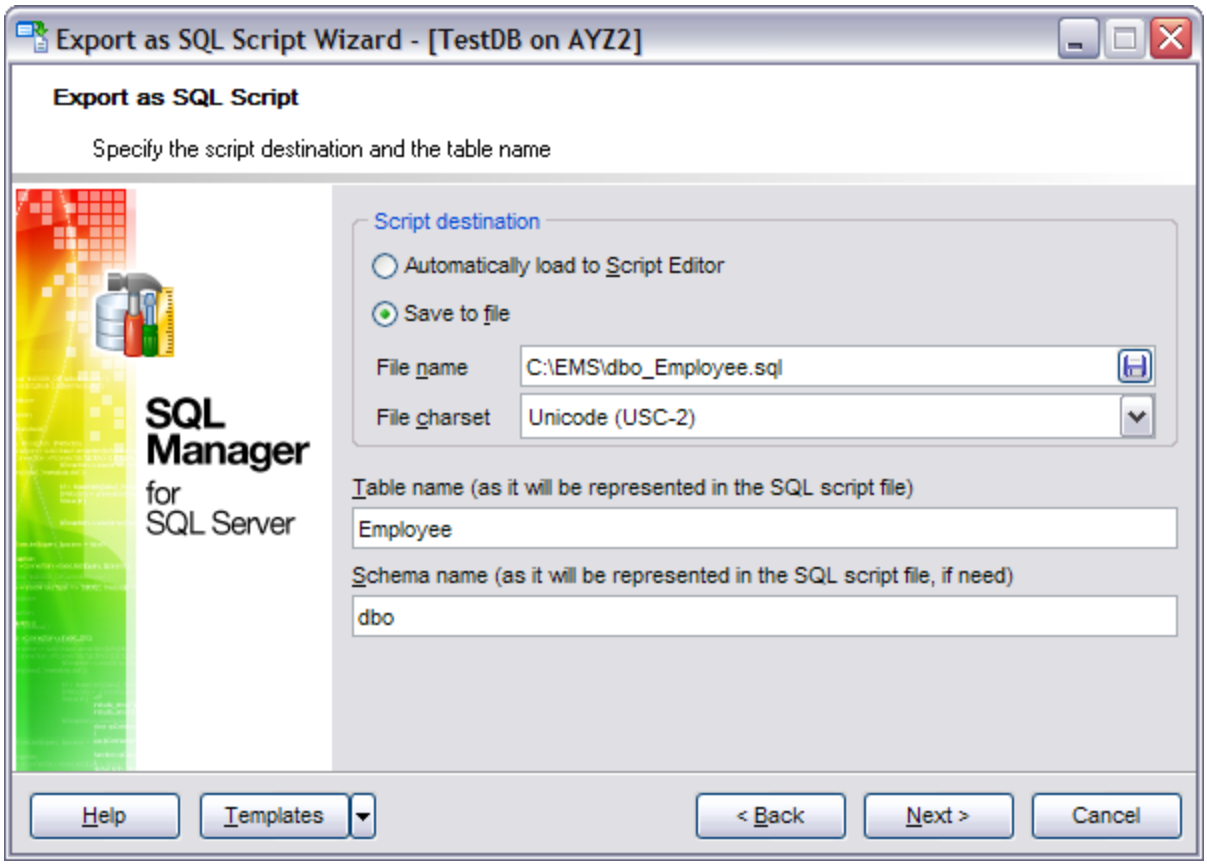
Specify whether the result script will be loaded to SQL Script Editor or saved to a file.

File name
Type in or use the button to specify the path to the file and the file name.

If necessary, select the **File charset** using the corresponding drop-down list.

Enter the **Table name** and the **Schema name** to be included in the result SQL script.

© 2008 EMS Database Management Solutions



Click the **Next** button to proceed to the Setting BLOB options step of the wizard.

8.3.3 Setting BLOB options

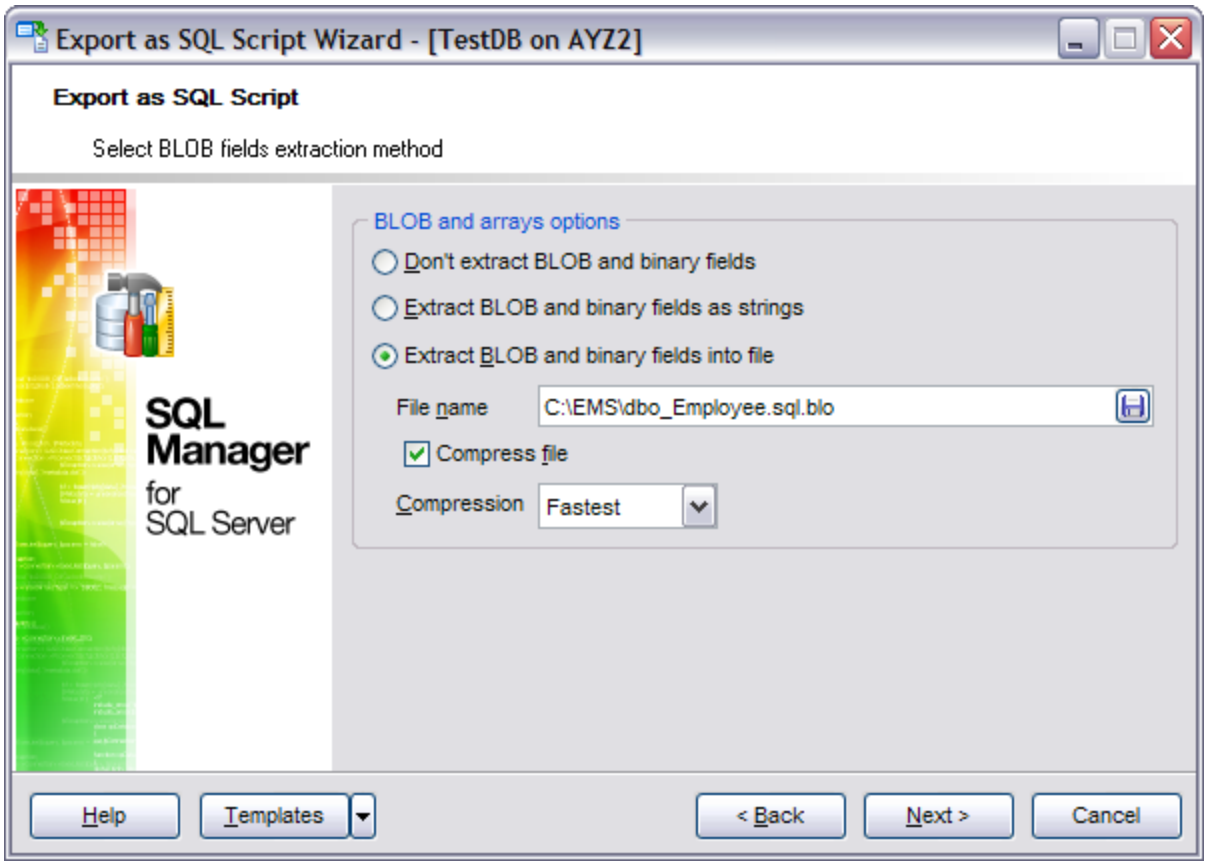
BLOB and arrays options

In this group of options you can determine whether BLOB fields are not to be extracted, extracted as strings, or extracted into a separate file. If the latter is selected, you also need to specify the **File name** (the **.blo* file where the BLOB data will be stored) and the location of the file on your local machine using the button.

Compress file
Check this option if you wish to compress the file containing BLOB data.

Compression
Define the desired compression level to be applied for the file: *None*, *Fastest*, *Default*, *Best*.

© 2008 EMS Database Management Solutions

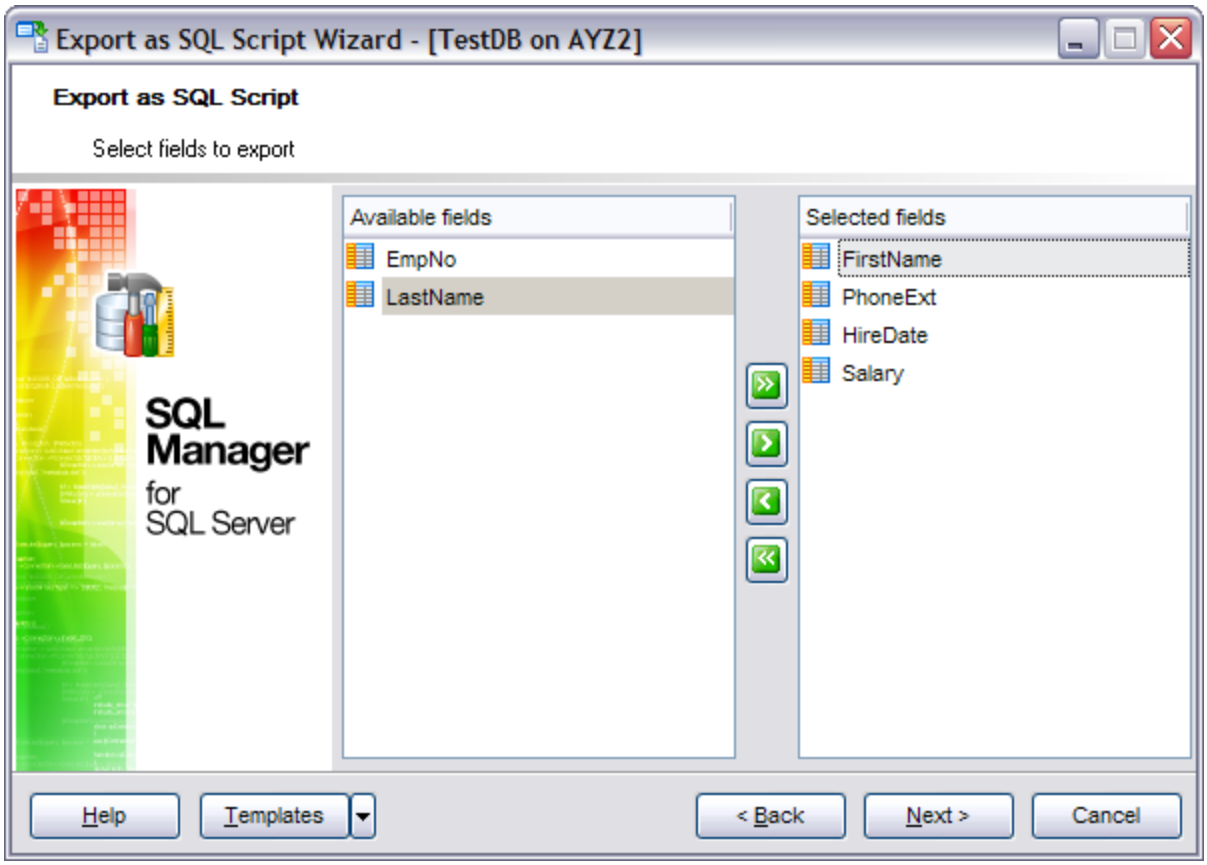


Click the **Next** button to proceed to the Selecting fields to export step of the wizard.

8.3.4 Selecting fields to export

This step of the wizard allows you to select the table field(s) to be exported to SQL script.
To select a field, you need to move it from the **Available fields** list to the **Selected fields** list. Use the buttons or drag-and-drop operations to move the fields from one list to another.

© 2008 EMS Database Management Solutions

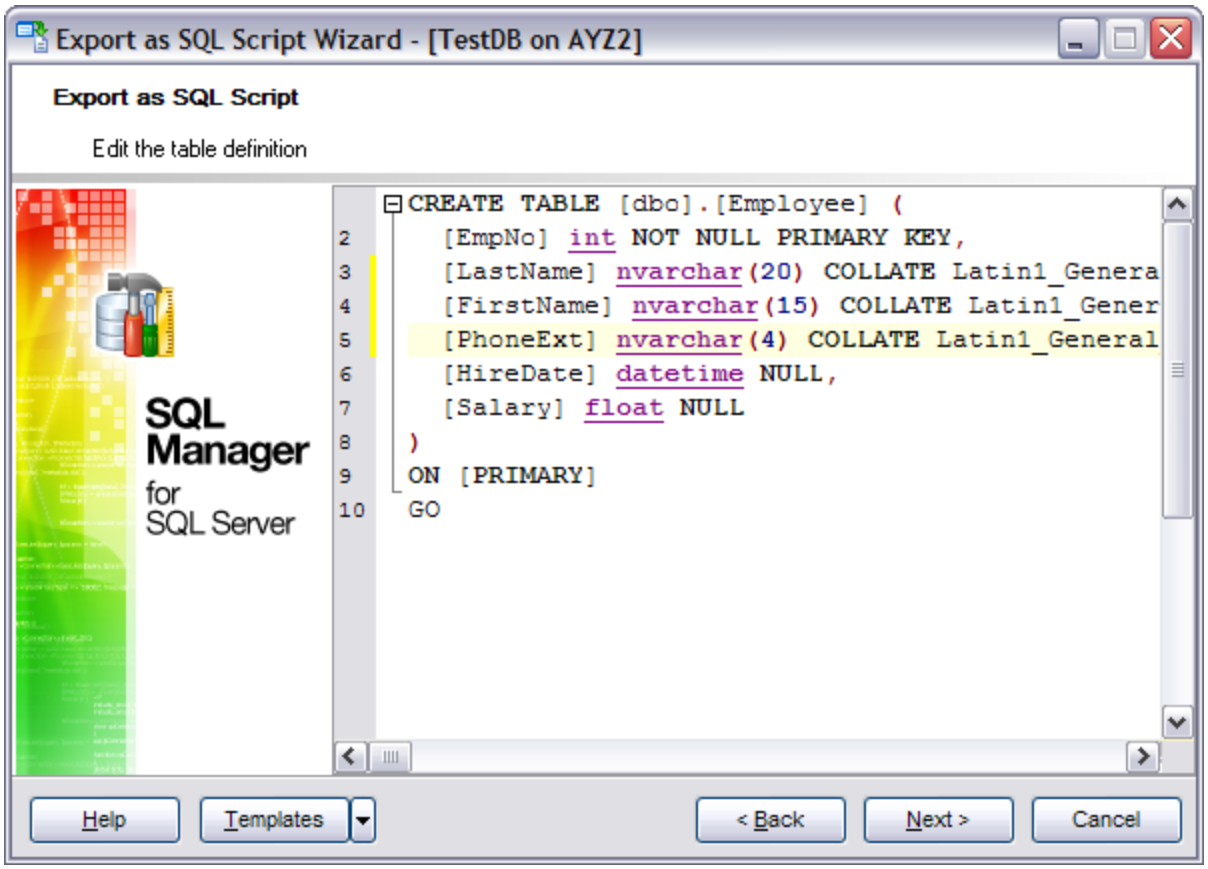


Click the **Next** button to proceed to the Editing table definition step of the wizard.

8.3.5 Editing table definition

This step is available only if the **Add CREATE TABLE statement** option was checked on the Defining the type of destination server step of the wizard. It allows you to edit the SQL script for creating the table.

© 2008 EMS Database Management Solutions



Click the **Next** button to proceed to the Setting export options step of the wizard.

8.3.6 Setting export options

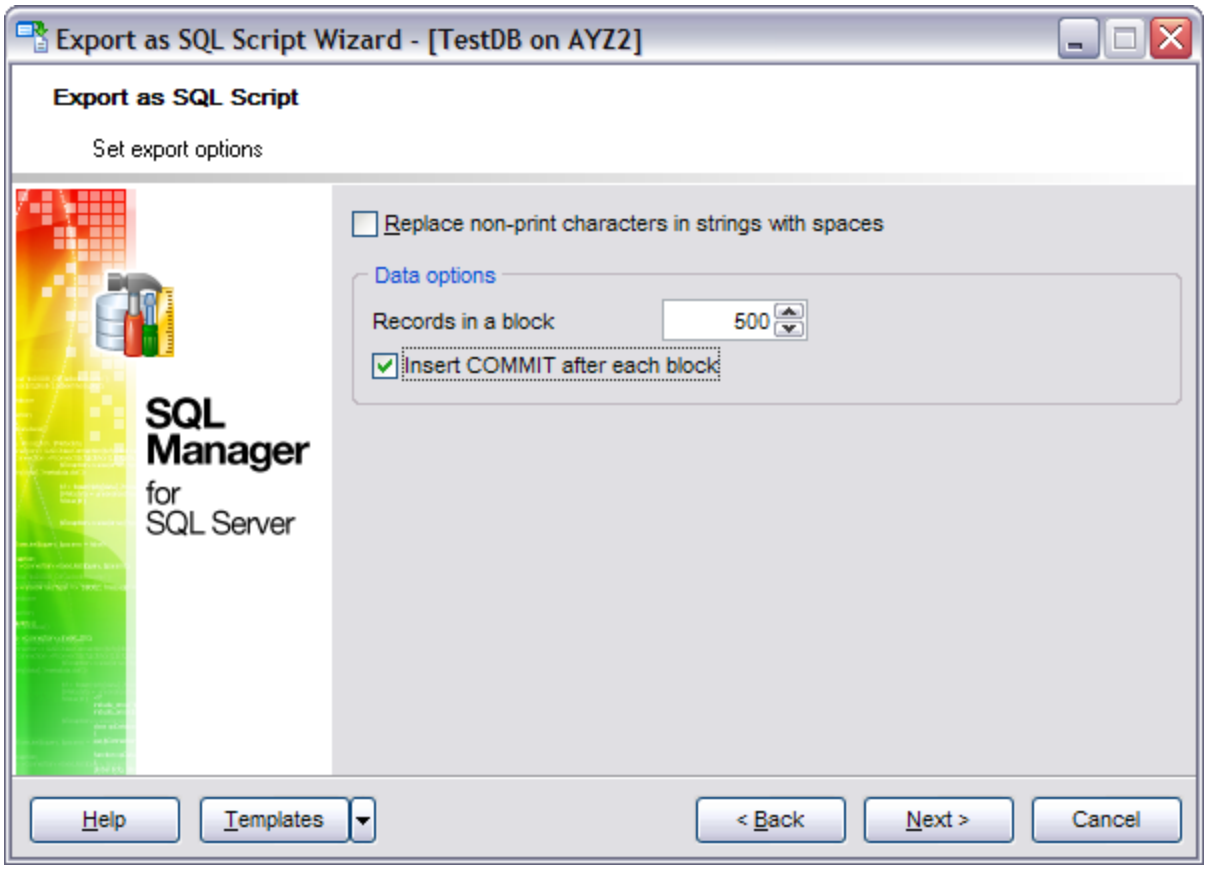
Specify common export options according to your needs.

If necessary, you can choose **to replace non-print characters in strings with spaces**.

Records in block
Use the spin-edit box to define the number of records in each committed block.

☑ **Insert COMMIT after each block**
Check this option to add the *COMMIT* statement after a defined number of records.

© 2008 EMS Database Management Solutions



Click the **Next** button to proceed to Exporting as SQL Script.

8.3.7 Exporting as SQL Script

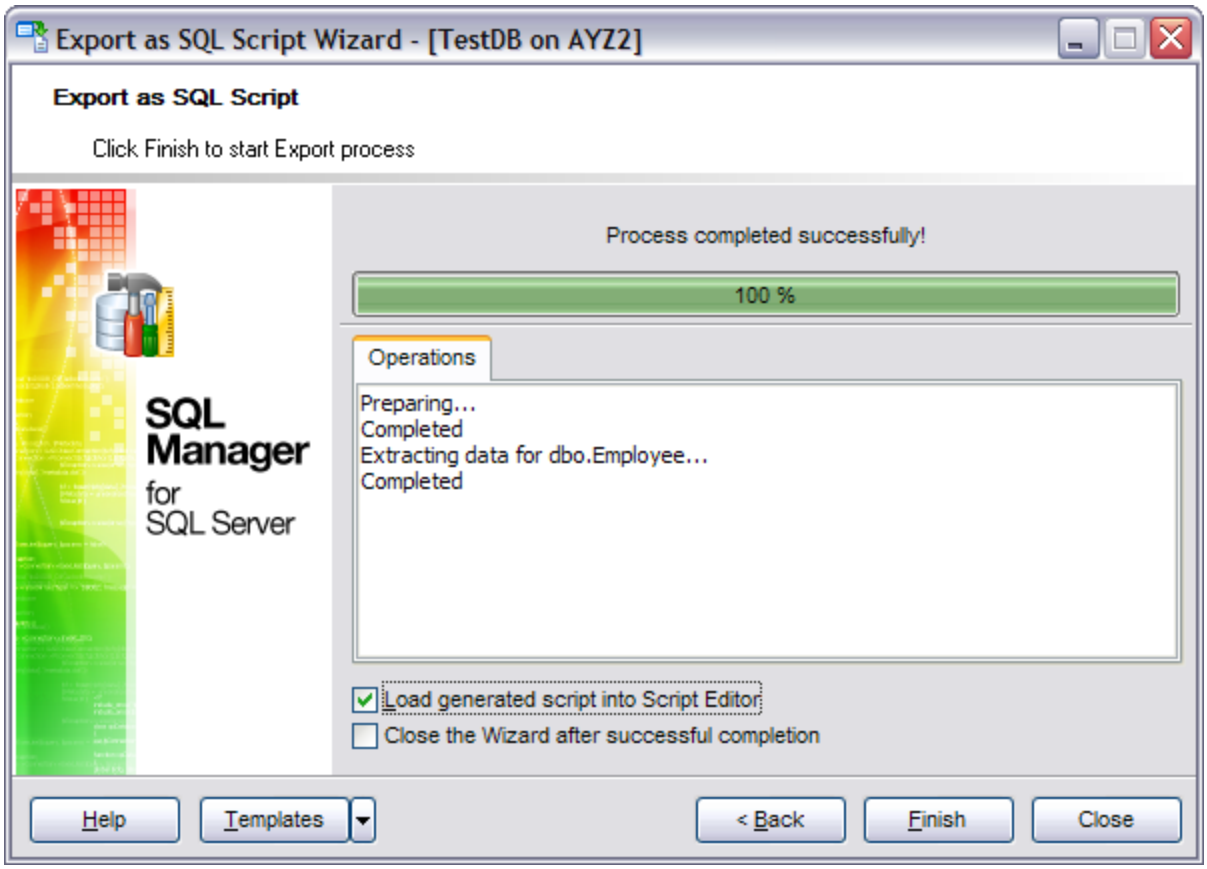
This step of the wizard is intended to inform you that all export options have been set, and you can start the export as SQL script process.

Load generated script into Script Editor
Check this option to load the result script to SQL Script Editor.

Close the Wizard after successful completion
If this option is selected, the wizard is closed automatically when the export process is completed.

The **Operations** tab allows you to view the log of operations and errors (if any).

© 2008 EMS Database Management Solutions



If necessary, you can save a template for future use.

Click the **Finish** button to run the export as SQL script process.

© 2008 EMS Database Management Solutions

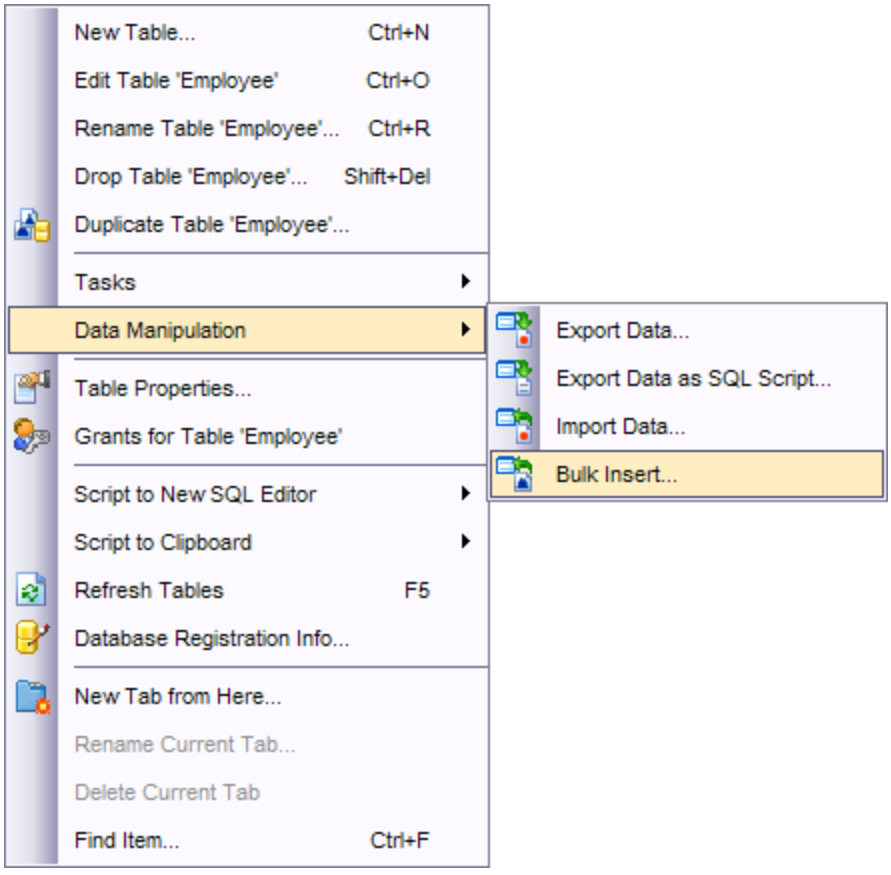
8.4 Bulk Insert Wizard

Bulk Insert Wizard allows you to use the Transact-SQL *BULK INSERT* statement to load a data file into a database table / view in a user-specified format. This statement copies data from a file to a table, appending the records to already existing data in the table. The file must be accessible to the server and the filename must be specified from the point of view of the server. You can save your settings as a template any time for future use.

To start the wizard, right-click the table/view in DB Explorer, select the **Data Manipulation** context menu group and proceed to the **Bulk Insert...** item within this group.

Alternatively, you can open the **Data** tab of Table Editor / View Editor, right-click the grid there, then select the **Data Manipulation** context menu group and proceed to the **Bulk Insert to <object_name>...** item within this group, or use the **Bulk Insert** item of the Navigation bar.

- Selecting a table or view and the data file
- Setting bulk insert options
- Customizing data file format
- Viewing the result SQL statement
- Running bulk insert
- Using templates



© 2008 EMS Database Management Solutions

Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**No**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

Export Data Wizard
Import Data Wizard
Export as SQL Script

8.4.1 Selecting a table or view and the data file

This step of the wizard allows you to select the target table (or view) and specify the data file that contains data to load into the specified table or view.

Table or view
Use the drop-down list to select the target table or view.

Data file

Select whether the data file is located *on the server* or *in the network*. Note that it is obligatory to specify a valid path from the server on which SQL Server is running. If the data file is a remote file, you should specify the Universal Naming Convention (UNC) name.

File name
Type in or use the button to specify the full path to the data file for bulk insert using the **Open file...** dialog.

© 2008 EMS Database Management Solutions

Click the **Next** button to proceed to the Setting bulk insert options step of the wizard.

8.4.2 Setting bulk insert options

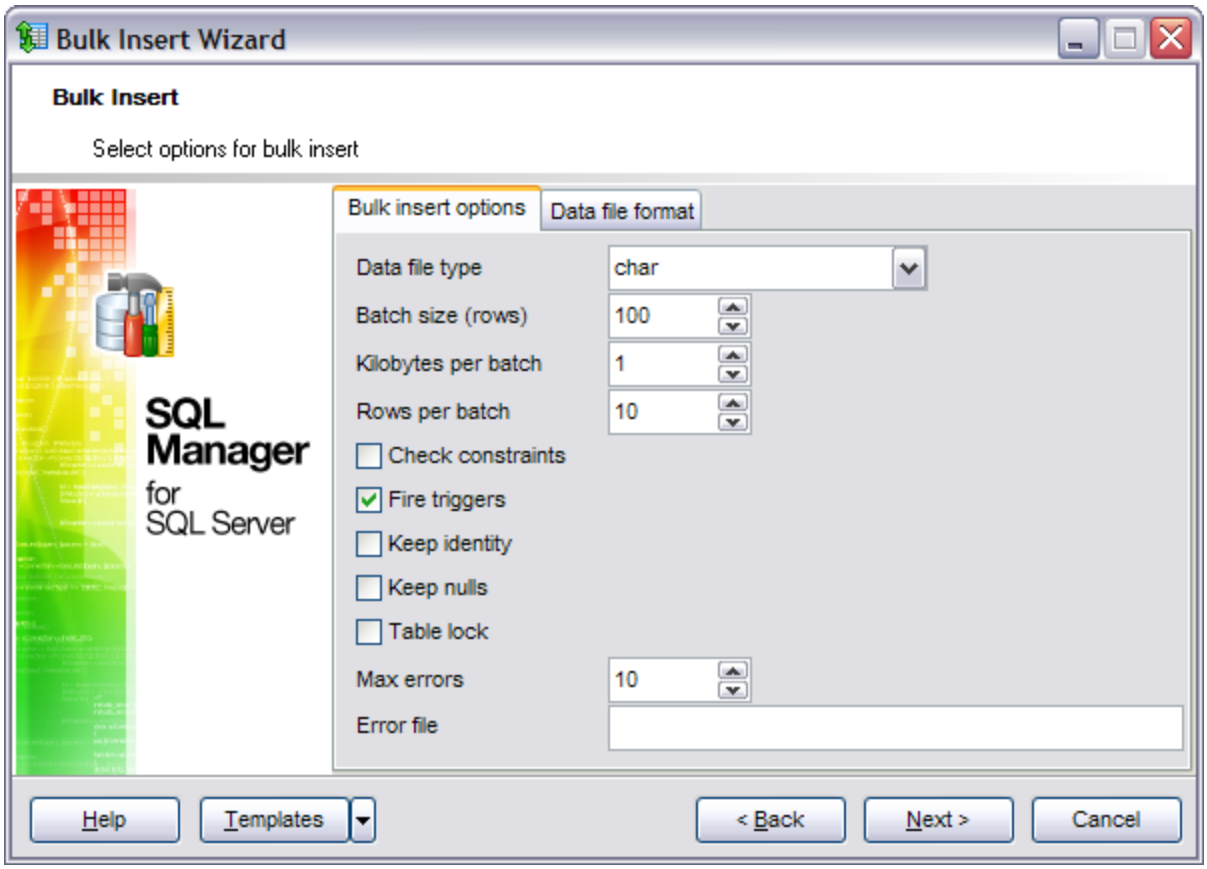
Data file type
Use this drop-down list to select the data file type value to be used to perform the load operation. Possible data file type values are: *char*, *native*, *widechar* and *widenative*.

Batch size (rows)
Specify the number of rows in a batch. Each batch is copied to the server as one transaction. If this operation fails, SQL Server commits or rolls back the transaction for every batch.

Kilobytes per batch
Specify the approximate number of kilobytes (KB) of data in each batch.

Rows per batch
This value indicates the number of rows in each batch.

© 2008 EMS Database Management Solutions



Check constraints
Enable this option to specify that all constraints on the target table or view must be checked during the bulk insert operation. If disabled, any CHECK constraints are ignored, and after the operation the constraint on the table is marked as not-trusted. This might be necessary if the input data contains rows that violate constraints.
Note: *UNIQUE*, *PRIMARY KEY*, *FOREIGN KEY* and *NOT NULL* constraints are always enforced.

Fire triggers
Use this option to specify that any insert triggers defined on the destination table execute during the bulk load operation. If triggers are defined for *INSERT* operations on the target table, they are fired for every completed batch.

Keep identity
Use this option to specify that identity value or values in the imported data file are to be used for the identity column. If this option is not checked, the identity values for this column are verified but not imported, and SQL Server automatically assigns unique values based on the seed and increment values specified during table creation.

Keep nulls
Use this option to specify that empty columns should retain a null value during the bulk load operation, instead of having any default values for the columns inserted.

Table lock

© 2008 EMS Database Management Solutions

Use this option to specify that a table-level lock is acquired for the duration of the bulk load operation. Holding a lock for the duration of the bulk load operation reduces lock contention on the table and may significantly improve performance.

Max errors
Specifies the maximum number of syntax errors allowed in the data before the bulk insert operation is canceled. Each row that cannot be imported by the bulk load operation is ignored and counted as one error.
Note: This option is not applied to constraint checks and to converting *money* and *bigint* data types.

Error file
Specify the file used to collect rows that have formatting errors and cannot be converted to an OLEDB rowset. These rows are copied into this error file from the data file "as is". The error file is created when the command is executed.

Click the **Next** button to proceed to the Customizing data file format step of the wizard.

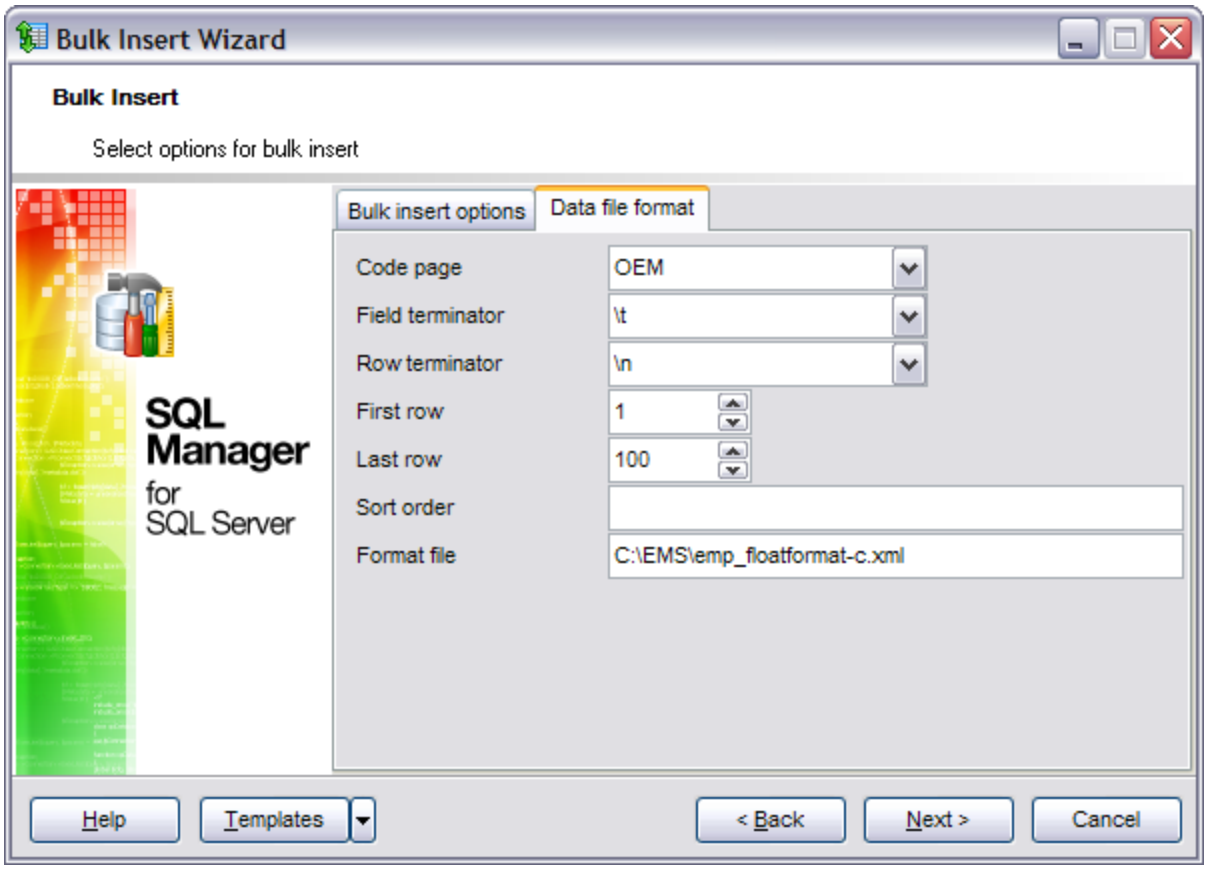
8.4.3 Customizing data file format

Code page
Use this drop-down list to select the code page of the data in the data file: *ACP*, *OEM* or *RAW*. This option is relevant only if the data contains *char*, *varchar* or *text* columns with character values greater than *127* or less than *32*.
- **ACP**: columns are converted from the ANSI/Microsoft Windows code page (ISO 1252) to the SQL Server code page
- **OEM**: columns are converted from the system OEM code page to the SQL Server code page
- **RAW**: no conversion is performed

Field terminator
Use the drop-down list to specify the field terminator to be used for *char* and *widechar* data files: *tab character (\t)*, *comma (,)* or *semicolon (;)*.

Row terminator
Use the drop-down list to select the row terminator to be used for *char* and *widechar* data files: *newline character (\n)* or *return character (\r)*.

© 2008 EMS Database Management Solutions



First row
Specify the number of the first row to load.

Last row
Specify the number of the last row to load. If this parameter is set to 0, the last row to load is the last row in the specified data file.

Sort order
{ column_name [ASC | DESC] } [,... n]
This string specifies the way the data in the data file are sorted. Bulk insert operation performance is improved in case the data are sorted according to the clustered index on the table. The column names supplied must be valid columns in the destination table. By default, the bulk insert operation assumes the data file is unordered.

Format file
Specify the full path of a format file. A format file describes the data file. The format file should be used in the following cases:

- The data file contains greater or fewer columns than the table or view
- The columns are in a different order
- The column delimiters vary
- There are other changes in the data format

Click the **Next** button to proceed to the Viewing the result SQL statement step of the

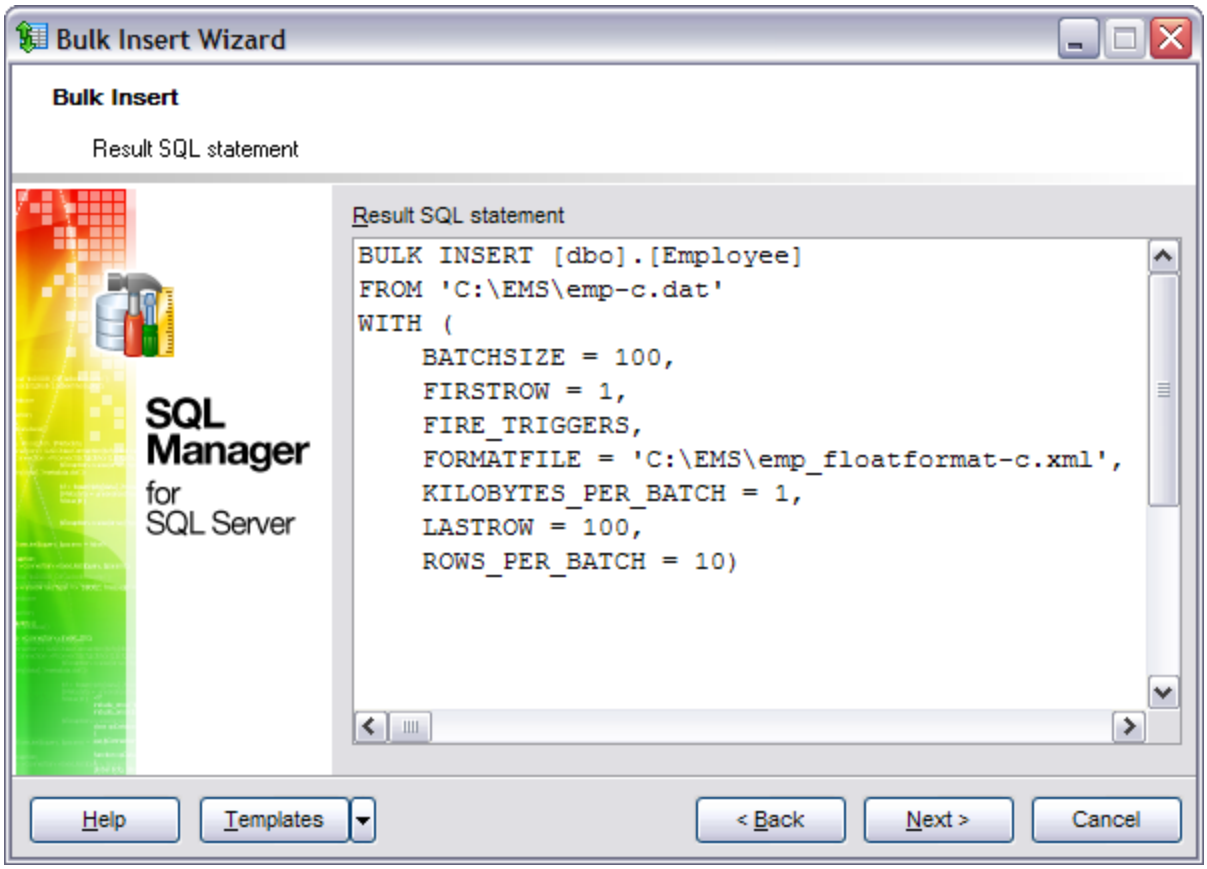
© 2008 EMS Database Management Solutions

wizard.

8.4.4 Viewing the result SQL statement

This step of the wizard allows you to browse **the result SQL statement**.

If necessary, you can edit the definition of the new object.



When you are done, click the **Next** button to proceed to the last step of the wizard.

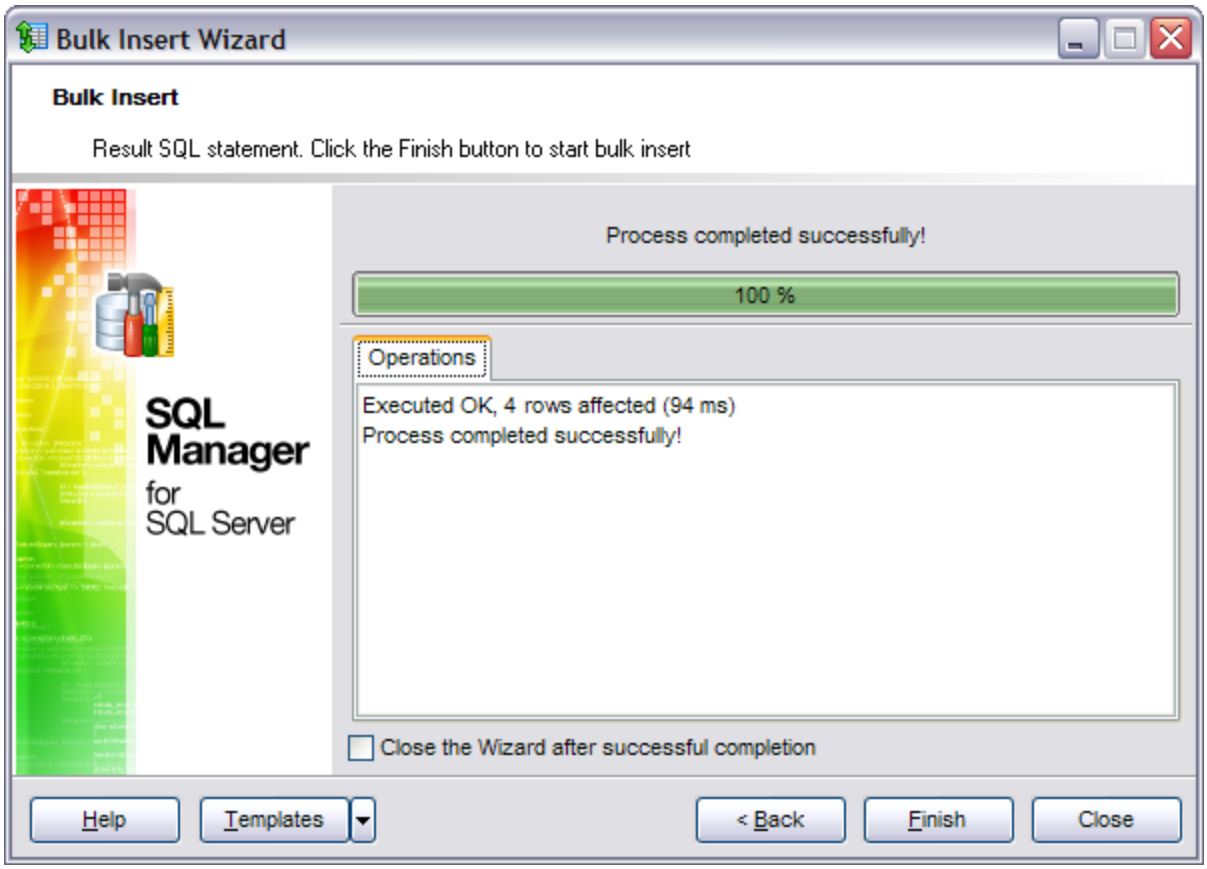
8.4.5 Running bulk insert

This step of the wizard is intended to inform you that all bulk insert options have been set, and you can start the process.

☑ **Close the Wizard after successful completion**
If this option is selected, the wizard is closed automatically when the bulk insert process is completed.

The **Operations** tab allows you to view the log of operations and errors (if any).

© 2008 EMS Database Management Solutions



If necessary, you can save a template for future use.

Click the **Finish** button to run the bulk insert process.

© 2008 EMS Database Management Solutions

Part

9 Database tools

SQL Manager 2008 for SQL Server provides a number of powerful tools that allow you to perform various operations over your databases.

- Common DB Tools
- Specific DB Tools

See also:

Getting started
Database Explorer
Database management
Database objects management
Query management tools
Data management
Import/Export tools
Server tools
Personalization
External tools

© 2008 EMS Database Management Solutions

9.1 Common DB Tools

The following *common database tools* are available in SQL Manager 2008 for SQL Server:

- **Dependency Tree**
  Allows you to view all the object dependencies in one diagram.
- **Visual Database Designer**
  Allows you to lay out your database schema visually.
- **SQL Script Editor**
  Executes SQL scripts in the database.
- **Extract Database Wizard**
  Extracts the table metadata and/or data to an SQL script which can be executed later on another machine to restore the database structure and/or data.
- **Print Metadata**
  Creates powerful metadata reports in the WYSIWYG mode ready for printing.
- **HTML Report**
  Creates powerful metadata reports in the HTML format.
- **Report Designer**
  Allows you to build and print database reports.
- **SQL Monitor**
  Displays all the SQL statements executed while working in SQL Manager 2008 for SQL Server.
- **Using templates**
  Facilitates using SQL Manager wizards.

See also:

Specific database tools

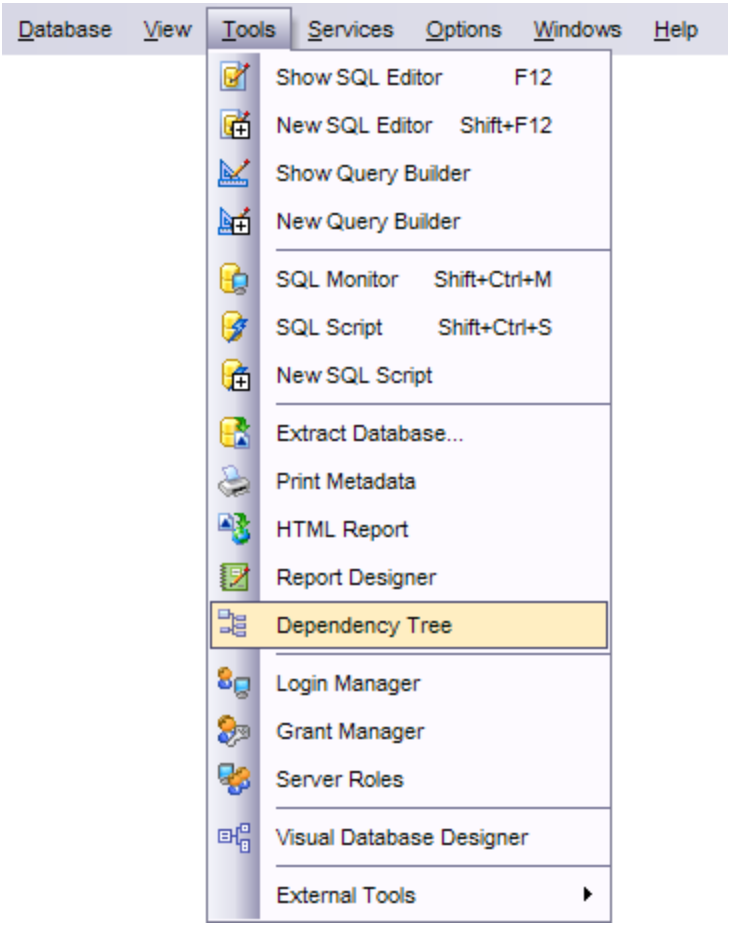
9.1.1 Dependency Tree

The **Dependency Tree** tool allows you to view all the object dependencies in one diagram.

To call the **Dependency Tree** window, select the **Tools | Dependency Tree** menu item, or use the **Dependency Tree** button on the main toolbar.

- Using Navigation bar
- Viewing dependency tree

© 2008 EMS Database Management Solutions



Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**No**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

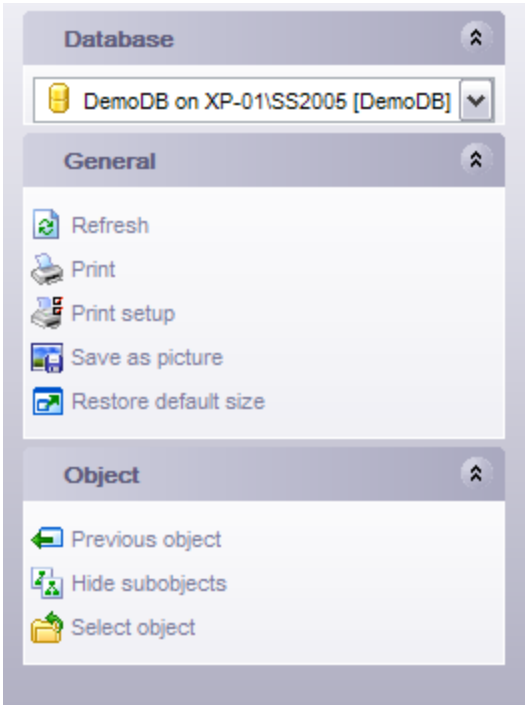
Visual Database Designer
SQL Script Editor
Extract Database Wizard
Print Metadata
HTML Report Wizard
Reports management
SQL Monitor

© 2008 EMS Database Management Solutions

9.1.1.1 Using Navigation bar

The **Navigation bar** of the **Dependency Tree** window allows you to:

- select the database;
- refresh the currently displayed dependency tree;
- print the diagram;
- set printing options using the **Print Setup** dialog;
- save the current diagram as a picture;
- restore the default size of the window;
- navigate within selected objects;
- show/hide subobjects;
- select an object.



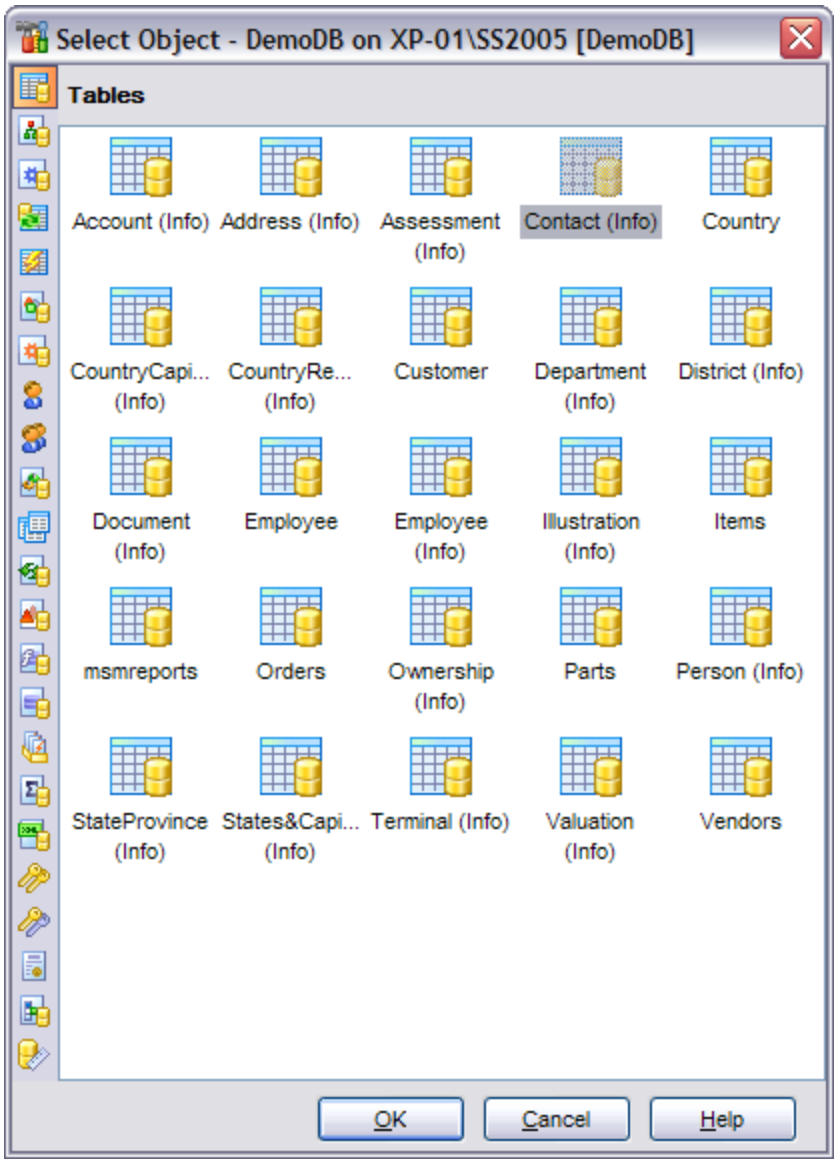
See also:

Viewing dependency tree

9.1.1.2 Viewing dependency tree

To view dependencies of an object, click the **Select object** Navigation bar item. Then select the required object in the **Select Object** dialog window. The dependency tree will appear in the main area of the window.

© 2008 EMS Database Management Solutions



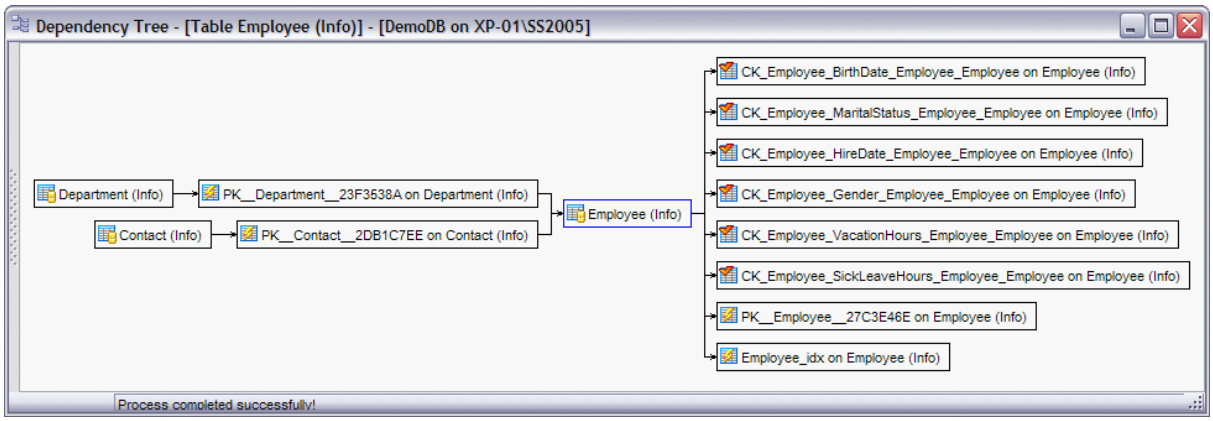
The *root* object is marked out with a blue frame.

The objects that the root object depends on are situated to the left of the root object. The objects that depend on the root object are situated to the right of the root object.

You can switch between objects by selecting them in the diagram. The selected object becomes the root object.

To show/hide subobjects click the **Show Subobjects / Hide Subobjects** link on the Navigation bar.

© 2008 EMS Database Management Solutions



The history of selected root objects is available: you can move back and forward through this history using the **Previous object** and the **Next object** links on the Navigation bar.

See also:

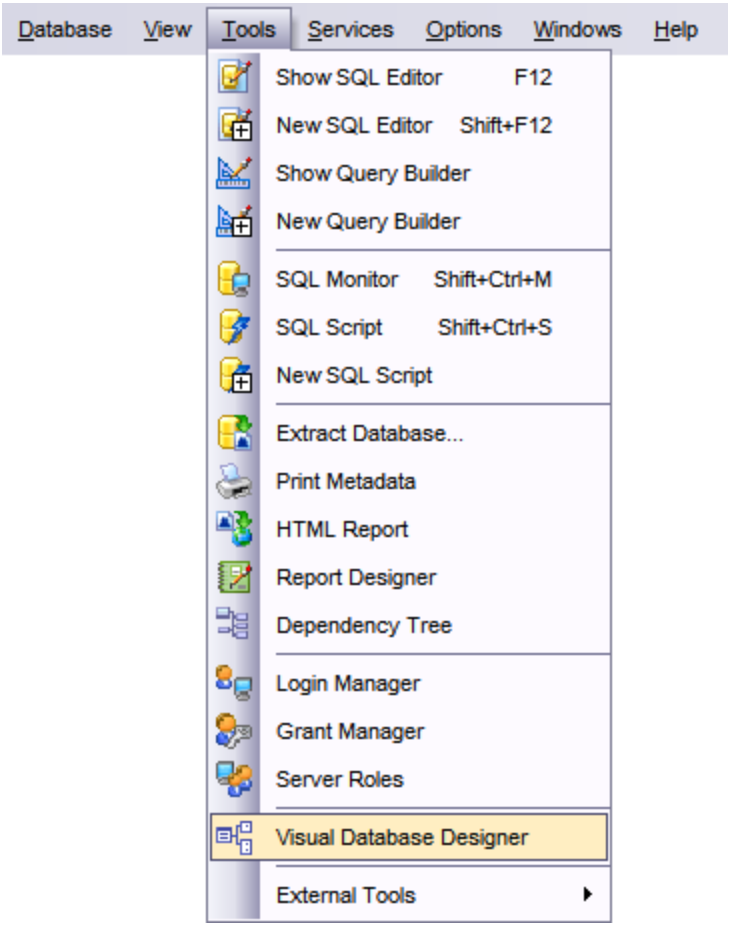
Using Navigation bar

9.1.2 Visual Database Designer

Visual Database Designer is provided for designing your database visually. It allows you to create, edit and drop tables and table fields, set relations between tables and perform other operations you may need to achieve your purpose.

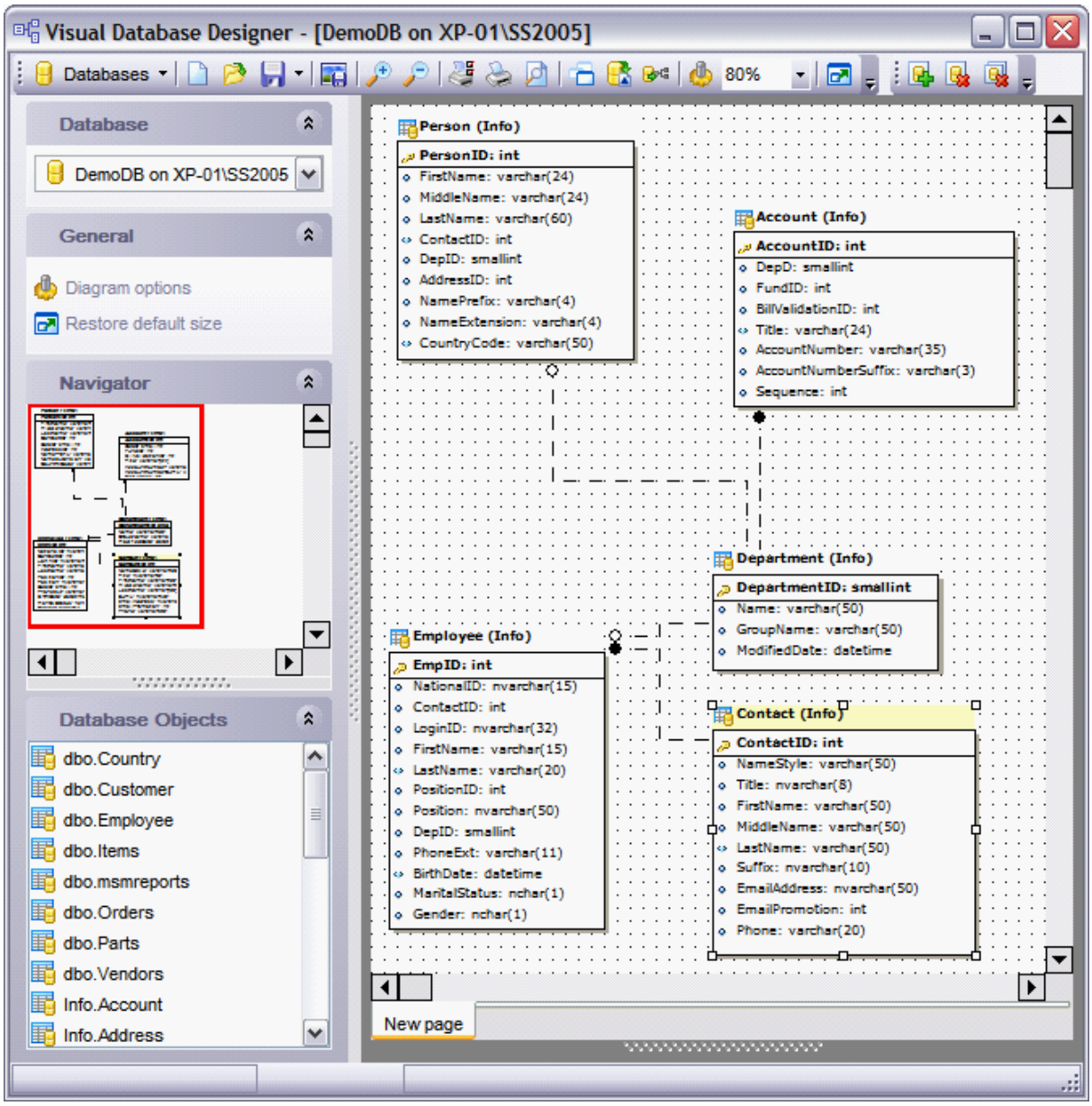
To open the designer, select the **Tools | Visual Database Designer** main menu item, or use the **VDBD** button on the main toolbar.

© 2008 EMS Database Management Solutions



- Using Navigation bar
- Diagram navigator
- Adding objects to diagram
- Removing objects from diagram
- Creating new table
- Creating relationships
- Working with multiple pages
- Reverse engineering
- Printing diagram
- Saving/Loading diagram
- Setting diagram options

© 2008 EMS Database Management Solutions



Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**No**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

Dependency Tree

© 2008 EMS Database Management Solutions

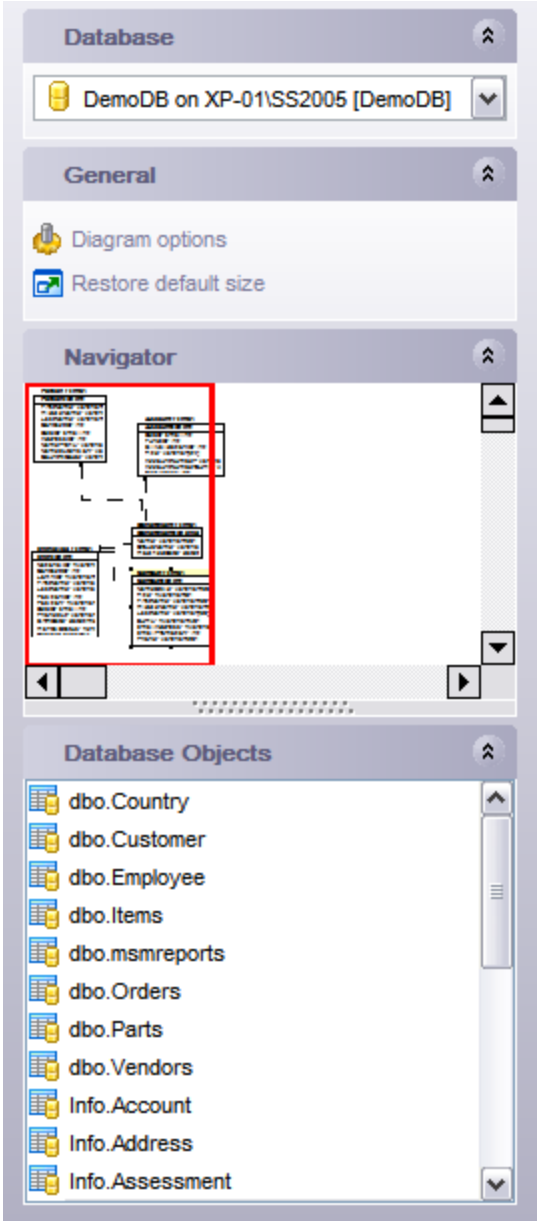
SQL Script Editor
Extract Database Wizard
Print Metadata
HTML Report Wizard
Reports management
SQL Monitor

9.1.2.1 Using Navigation bar and context menu

The **Navigation bar** of **Visual Database Designer** allows you to:

- select the database for the diagram;
- edit diagram options;
- restore the default size of the window;
- use Diagram Navigator;
- add objects to the diagram using the **Database Objects** group.

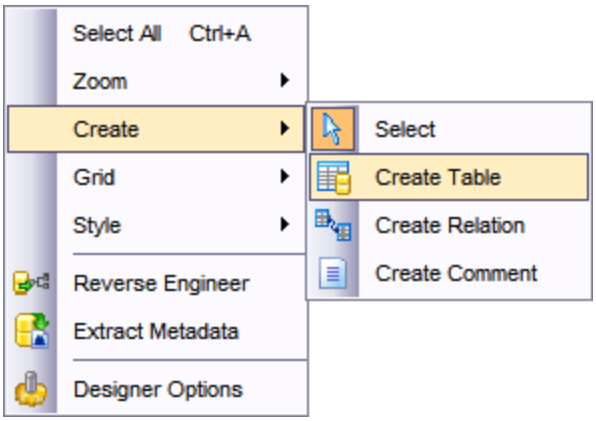
© 2008 EMS Database Management Solutions



The **context menu** of **Visual Database Designer** allows you to:

- select all objects in the diagram area;
- adjust diagram zoom for optimal representation;
- create a new table;
- create a new relation;
- create a comment;
- configure the grid;
- adjust the diagram style;
- perform Reverse Engineering;
- extract metadata of the diagram objects to SQL Script Editor;
- view/edit diagram options.

© 2008 EMS Database Management Solutions



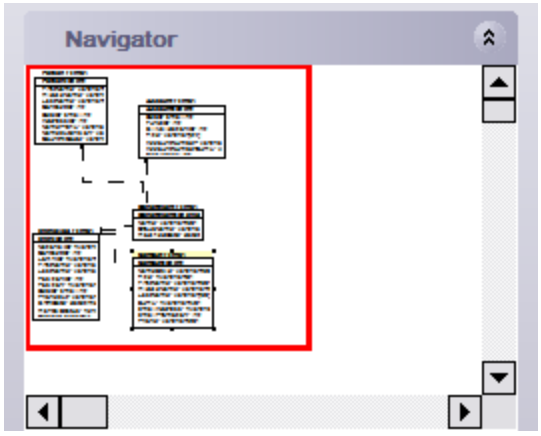
See also:

Diagram Navigator
Adding objects to diagram
Removing objects from diagram
Creating new table
Creating relationships
Working with multiple pages
Reverse engineering
Printing diagram
Saving/Loading diagram
Setting diagram options

9.1.2.2 Diagram Navigator

To navigate within the large diagram, use the **Navigator** tool on the Navigation bar. It allows you to see the whole diagram in a reduced size and even to move the objects in the diagram.

The area currently visible in the main diagram area is outlined in red color.



© 2008 EMS Database Management Solutions

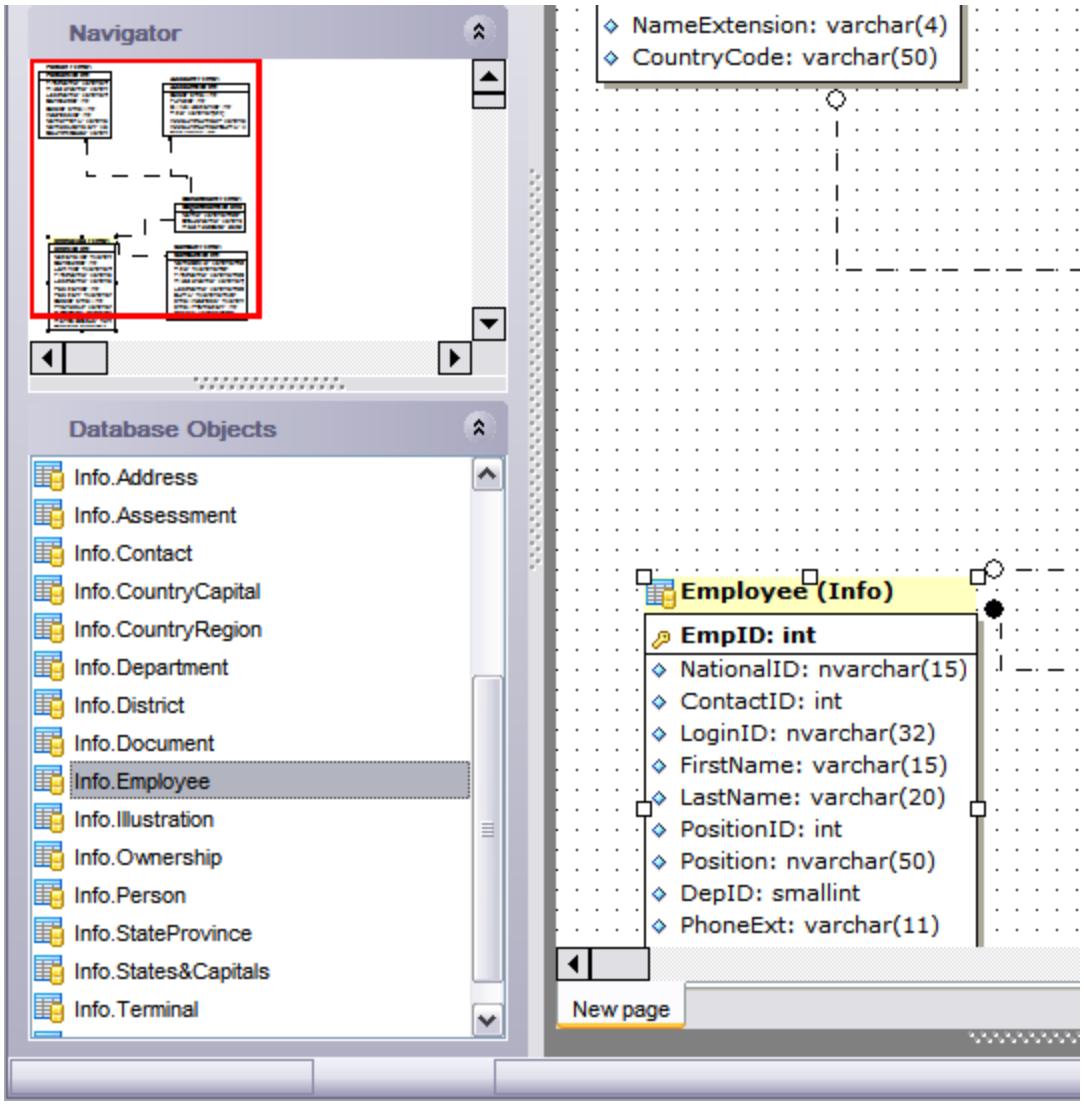
See also:

Using Navigation bar
Adding objects to diagram
Removing objects from diagram
Creating new table
Creating relationships
Working with multiple pages
Reverse engineering
Printing diagram
Saving/Loading diagram
Setting diagram options

9.1.2.3 Adding objects to diagram

To add an object to the diagram, drag it from the **Database Objects** group on the Navigation bar into the desired place on the diagram or simply double-click this object.

© 2008 EMS Database Management Solutions



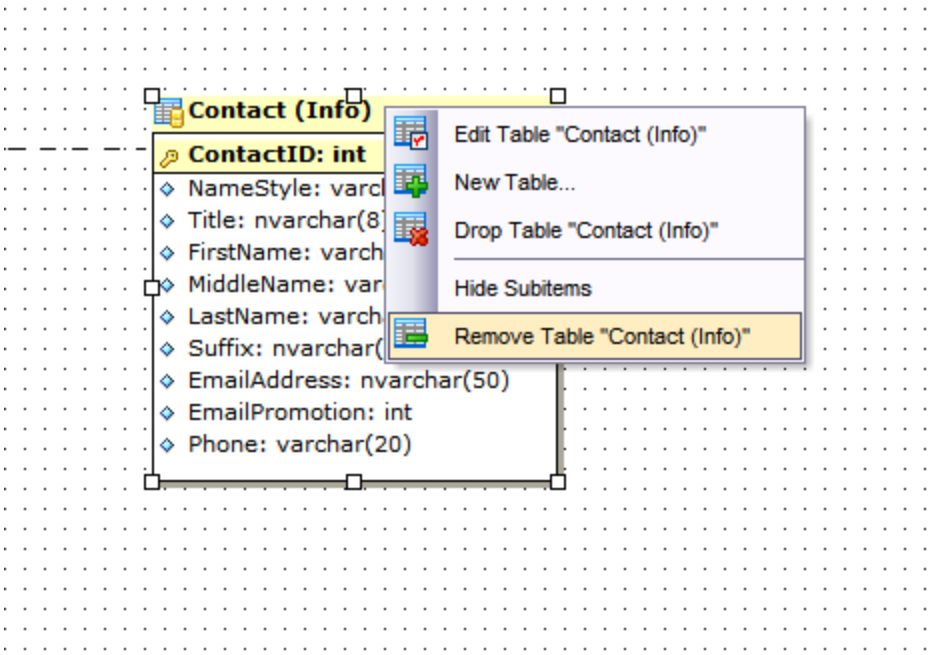
See also:

Using Navigation bar
Diagram Navigator
Removing objects from diagram
Creating new table
Creating relationships
Working with multiple pages
Reverse engineering
Printing diagram
Saving/Loading diagram
Setting diagram options

© 2008 EMS Database Management Solutions

9.1.2.4 Removing objects from diagram

To remove an object(s) from the diagram, select it, then right-click its title and choose the **Remove <object_name>** menu item or simply press the **Del** key.



See also:

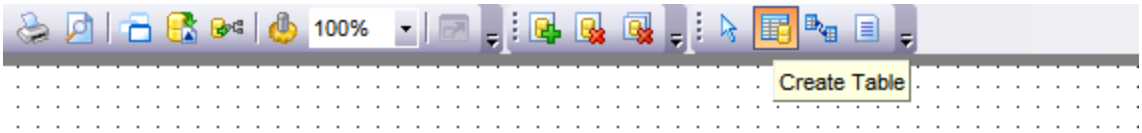
Using Navigation bar
Diagram Navigator
Adding objects to diagram
Creating new table
Creating relationships
Working with multiple pages
Reverse engineering
Printing diagram
Saving/Loading diagram
Setting diagram options

9.1.2.5 Creating new table

To create a new table using Visual Database Designer:

- press the **Create table** button on the toolbar;
- click the desired point on the diagram to place the new table at;
- create a new table using the Table Editor.

© 2008 EMS Database Management Solutions



Hint: To create a new table, you can also use the corresponding item of the context menu.

See also:
Using Navigation bar
Diagram Navigator
Adding objects to diagram
Removing objects from diagram
Creating relationships
Working with multiple pages
Reverse engineering
Printing diagram
Saving/Loading diagram
Setting diagram options

9.1.2.6 Creating relationships

To establish a new relationship (which is the foreign key in terms of database):
- press the **Create relation** button on the toolbar;
- click the table where the referential constraint should be created;
- click the referred table;
- create a new foreign key using Foreign key Editor.

Hint: To create a relation, you can also use the corresponding item of the context menu.

See also:
Using Navigation bar
Diagram Navigator
Adding objects to diagram
Removing objects from diagram
Creating new table

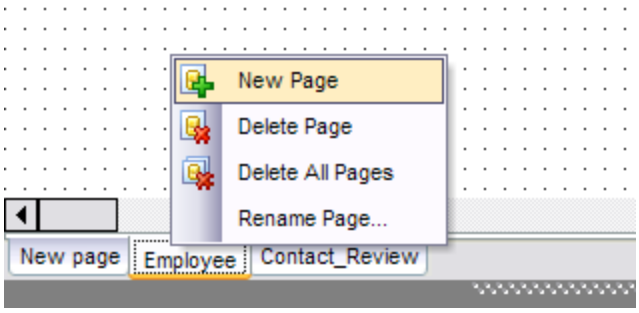
© 2008 EMS Database Management Solutions

Working with multiple pages
Reverse engineering
Printing diagram
Saving/Loading diagram
Setting diagram options

9.1.2.7 Working with multiple pages

You can create several pages in one diagram to separate certain subject areas from the database, e.g. for better comprehension.

To manage diagram pages, right-click the tabs at the bottom of the diagram area and select the appropriate popup menu items for *adding*, *deleting* or *renaming* pages.



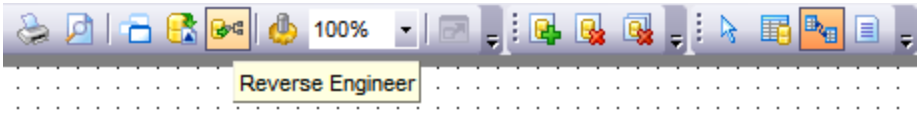
See also:
Using Navigation bar
Diagram Navigator
Adding objects to diagram
Removing objects from diagram
Creating new table
Creating relationships
Reverse engineering
Printing diagram
Saving/Loading diagram
Setting diagram options

9.1.2.8 Reverse engineering

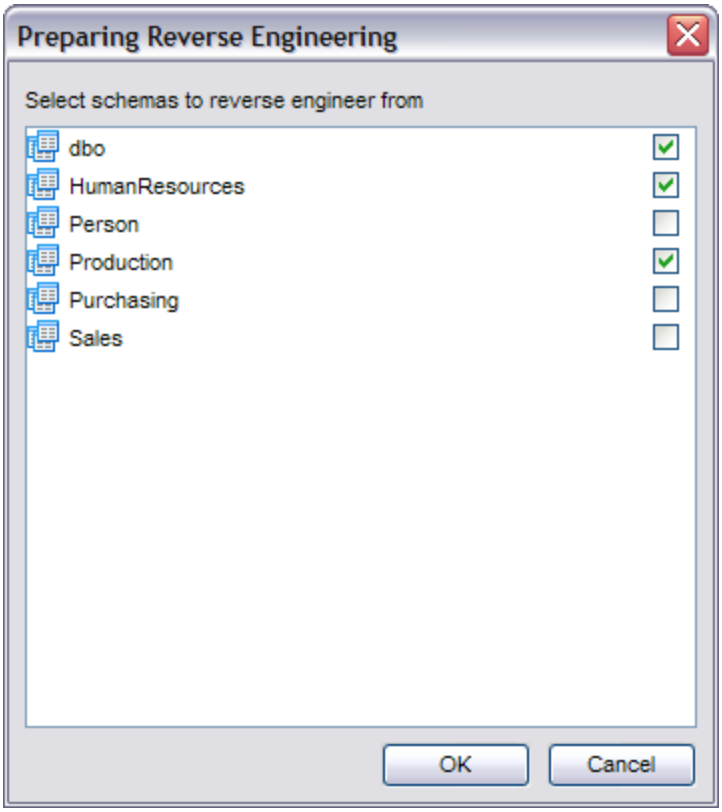
The **reverse engineering** process builds relationship diagram on the basis of the current database's structure. The objects are arranged automatically within the diagram.

To start the reverse engineering process, press the **Reverse Engineer** button on the toolbar, or use the corresponding item of the context menu.

© 2008 EMS Database Management Solutions



The **Preparing Reverse Engineering** dialog allows you to select schemas to reverse engineer.



Note: For SQL Server versions earlier than 2005, you are supposed to select database principals associated with the objects.

See also:

Using Navigation bar
Diagram Navigator
Adding objects to diagram
Removing objects from diagram
Creating new table
Creating relationships
Working with multiple pages
Printing diagram

© 2008 EMS Database Management Solutions

Saving/Loading diagram
Setting diagram options

9.1.2.9 Printing diagram

Visual Database Designer allows you to print and preview the diagram.

To preview the diagram:

- press the **Print Preview** button on the toolbar;
- preview the diagram using the Print Preview dialog.

To setup print options:

- press the **Print Setup** button on the toolbar, or use the corresponding link on the Navigation bar;
- set printing options using the Print Setup dialog and press **OK**.

To print the diagram:

- press the **Print** button on the toolbar;
- set printing options using the Print Setup dialog and press the **Print** button.

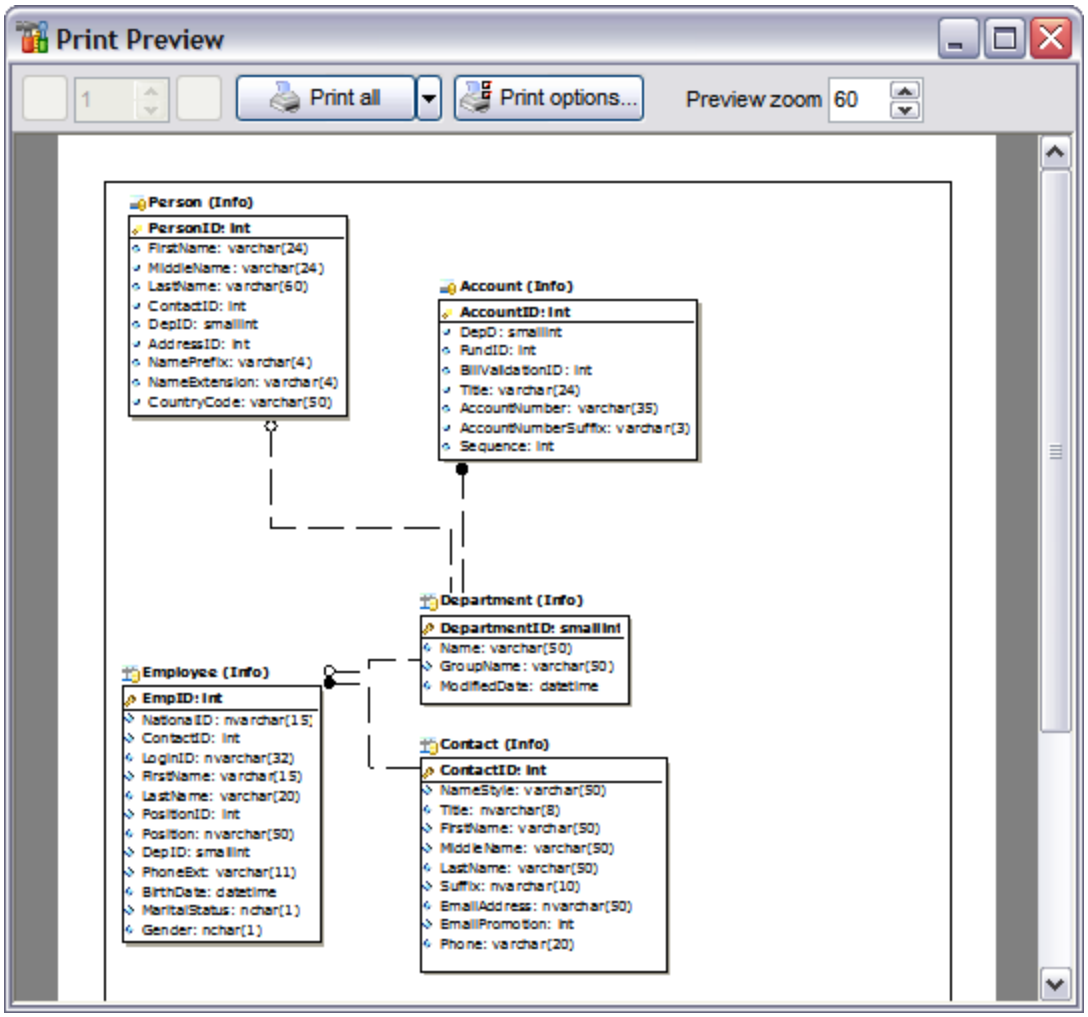
See also:

Using Navigation bar
Diagram Navigator
Adding objects to diagram
Removing objects from diagram
Creating new table
Creating relationships
Working with multiple pages
Reverse engineering
Saving/Loading diagram
Setting diagram options

9.1.2.9.1 Print Preview dialog

The **Print Preview** dialog allows you to see the diagram layout in WYCIWYG mode before it will be printed.

© 2008 EMS Database Management Solutions



See also:

Print Setup dialog

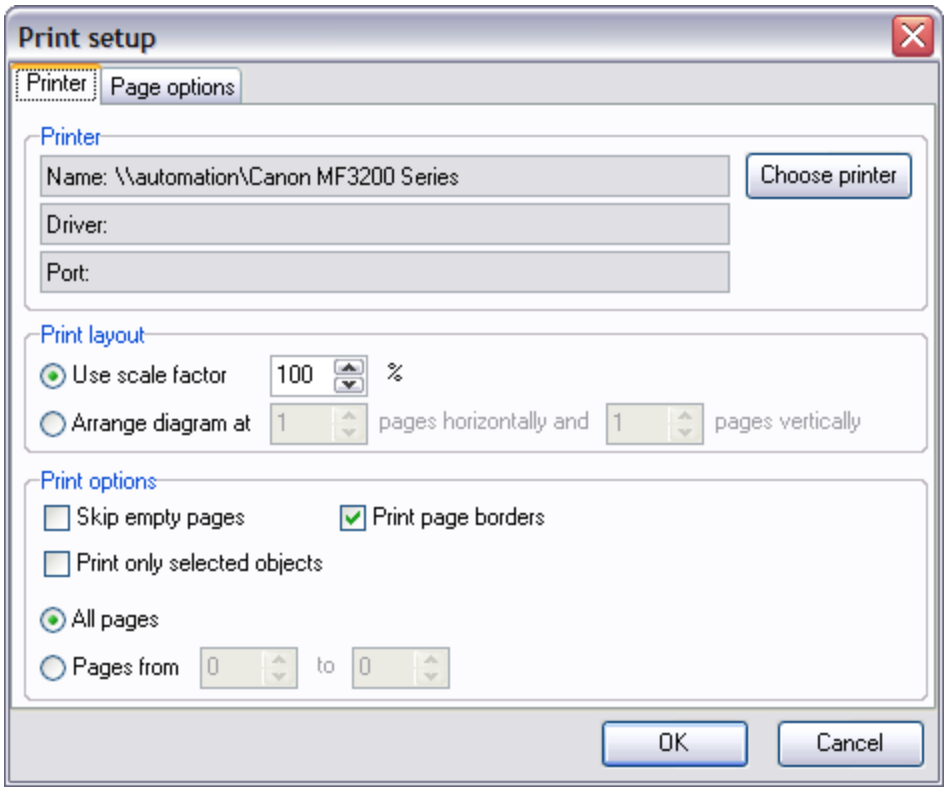
9.1.2.9.2 Print Setup dialog

The **Print Setup** dialog of **Visual Database Designer** provides two tabs for setting printing options: **Printer** and **Page options**.

The **Printer** tab of the **Print Setup** dialog allows you to:

- specify the printer (use the **Choose printer** button to select a printer which is not set by default on your system; the *name*, *driver*, *port* fields display the selected printer details);
- specify print layout: print using a defined *scale factor* or arrange diagram at a defined number of pages horizontally and vertically;
- set other print options.

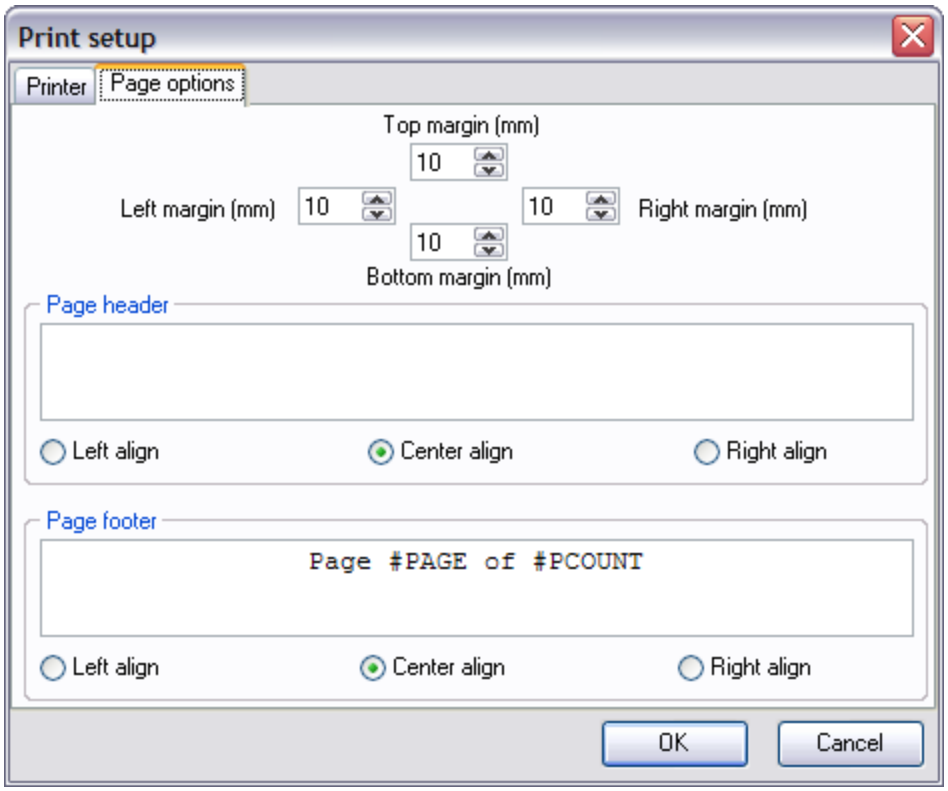
© 2008 EMS Database Management Solutions



The **Page options** tab of the **Print Setup** dialog allows you to:

- specify page margins (in millimeters): *Top margin*, *Bottom margin*, *Left margin*, *Right margin*;
- specify **Page header** and **Page footer**: enter header/footer *text*, set left/center/right *align*.

© 2008 EMS Database Management Solutions

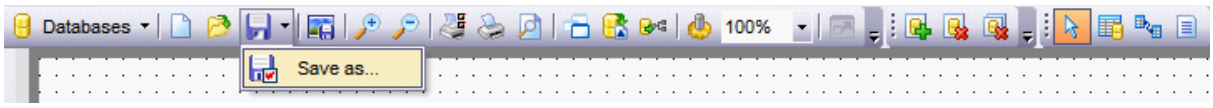


Hint: It is also possible to set macros in the **Page header** and **Page footer** fields:
#PCOUNT stands for the quantity of pages;
#PAGE - the number of the current page;
#DATE denotes the current date;
#TIME denotes the current time.

See also:
Print Preview dialog

9.1.2.10 Saving/Loading diagram

Use the **Save Diagram** and the **Open Diagram** buttons on the toolbar to save the diagram for future use or to load the previously saved diagram.



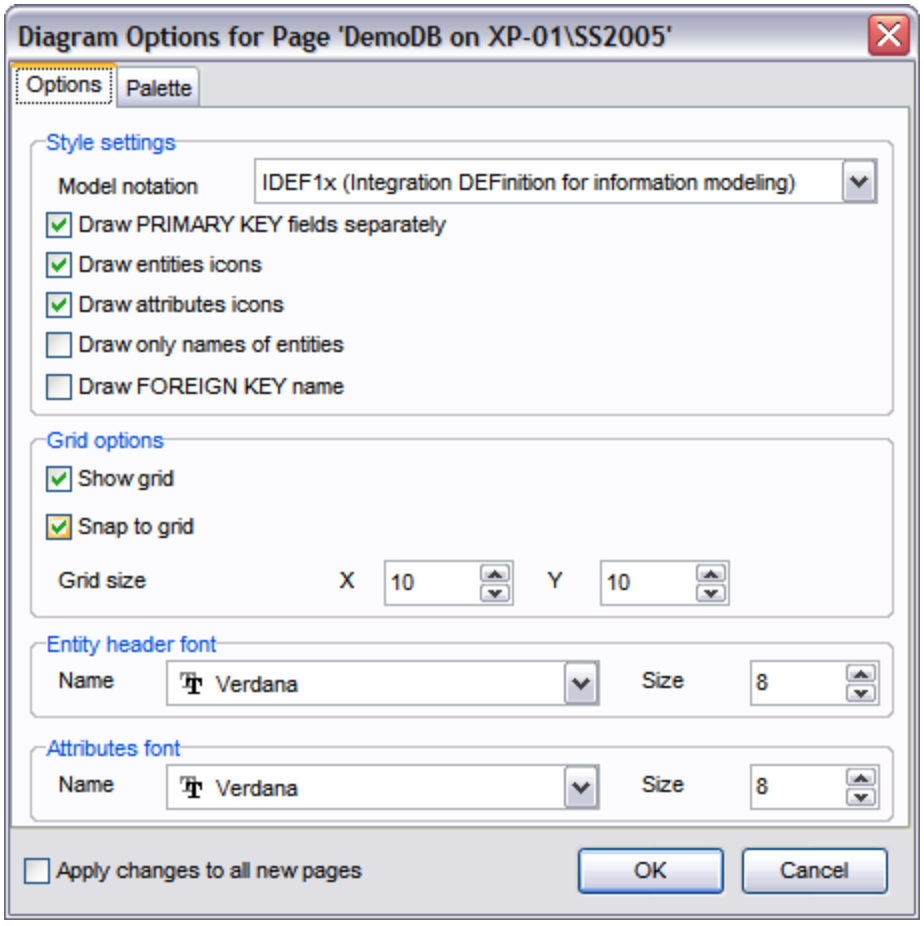
See also:
Using Navigation bar

© 2008 EMS Database Management Solutions

Diagram Navigator
Adding objects to diagram
Removing objects from diagram
Creating new table
Creating relationships
Working with multiple pages
Reverse engineering
Printing diagram
Setting diagram options

9.1.2.11 Setting diagram options

Using this dialog you can setup the behavior and look of each diagram page. See detailed description of each option on the Visual Database Designer page of the Environment Options dialog.



To call this dialog, you can also use the corresponding item of the context menu.

© 2008 EMS Database Management Solutions

See also:
Using Navigation bar
Diagram Navigator
Adding objects to diagram
Removing objects from diagram
Creating new table
Creating relationships
Working with multiple pages
Reverse engineering
Printing diagram
Saving/Loading diagram

9.1.3 SQL Script Editor

Using **SQL Script Editor** you can view, edit and execute SQL scripts.

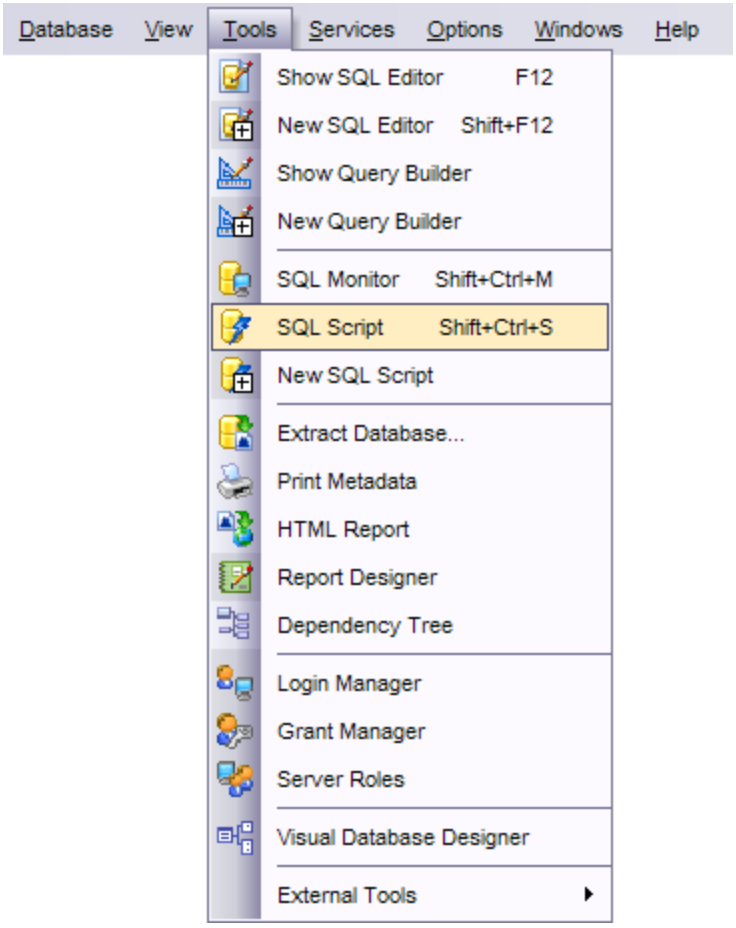
To open SQL Script Editor, select the **Tools | New SQL Script** / **Tools | SQL Script** main menu items or use the corresponding / toolbar buttons. You can also use the *Shift+Ctrl+S* shortcut for the same purpose.

In the script area you can view and edit the SQL script text. For your convenience syntax highlight and code completion features are implemented.

- Using Navigation bar
- Using the context menu
- Using Script Explorer
- Script execution

Note: **SQL Script Editor** does not show results returned upon SELECT queries execution. Please use SQL Editor for that purpose instead.

© 2008 EMS Database Management Solutions



Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**Yes**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

Dependency Tree
Visual Database Designer
Extract Database Wizard
Print Metadata
HTML Report Wizard
Reports management
SQL Monitor

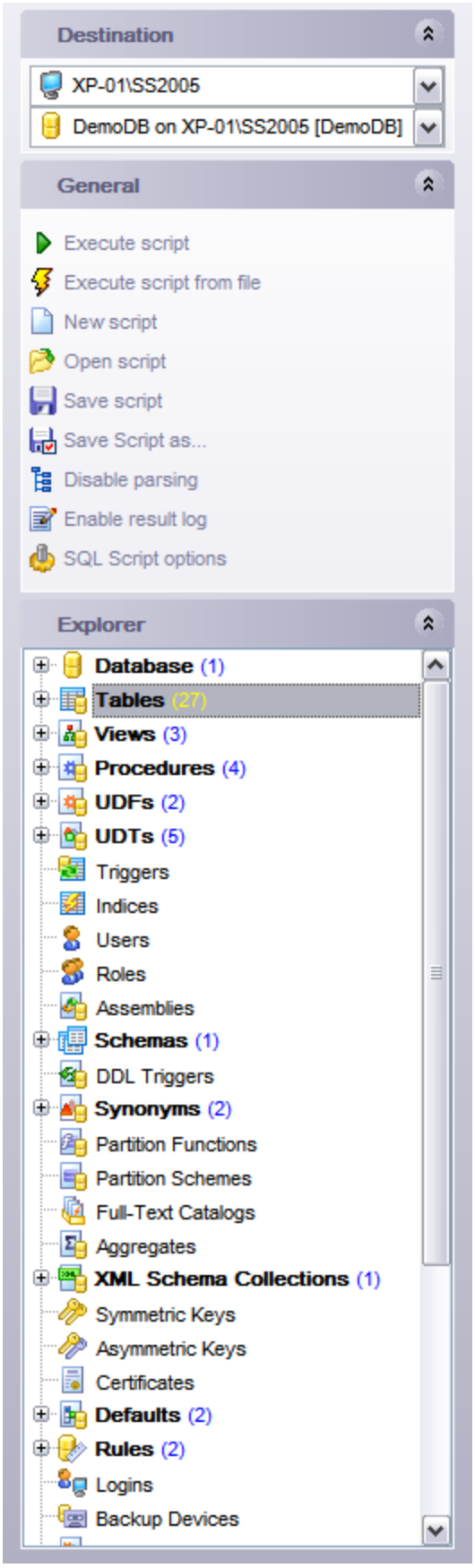
© 2008 EMS Database Management Solutions

9.1.3.1 Using Navigation bar

The **Navigation bar** of **SQL Script Editor** allows you to:

- select the destination host and database for the script;
- execute the current script;
- execute a script from file;
- create a new script;
- load a script from an **.sql* file using the **Open SQL Script** dialog;
- save the current script;
- save the script to an **.sql* file using the **Save as...** dialog;
- disable parsing of SQL code;
- enable result log;
- configure SQL Script Editor within the Script Options section of the Environment Options dialog;
- browse the objects used in the script within Script Explorer.

© 2008 EMS Database Management Solutions



© 2008 EMS Database Management Solutions

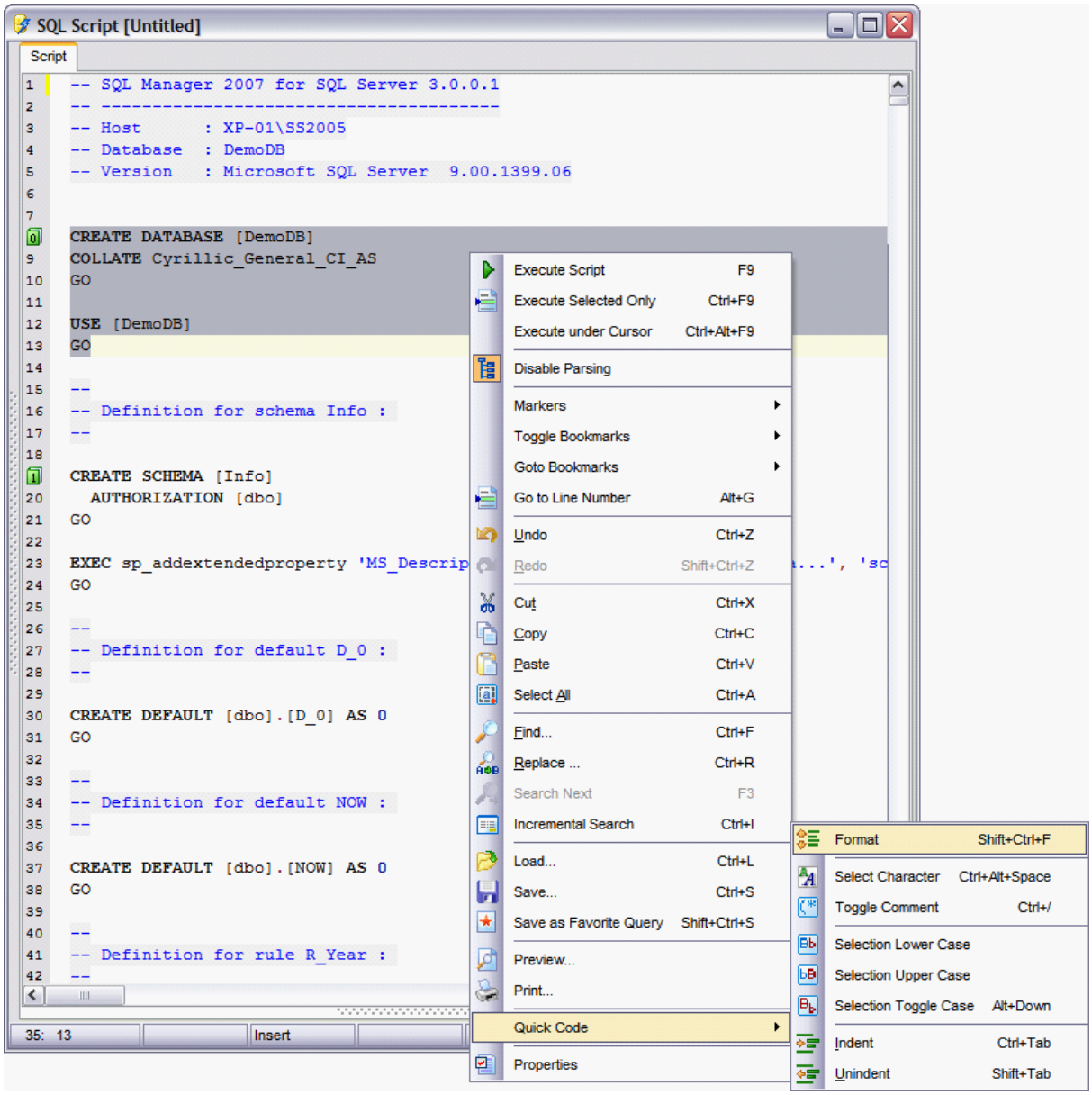
See also:
Using the context menu
Using Script Explorer
Script execution

9.1.3.2 Using the context menu

The **context menu** of SQL Script Editor area contains execution commands, most of the standard text-processing functions (*Cut*, *Copy*, *Paste*, *Select All*) and functions for working with the script as a whole, e.g. you can *move the cursor to a particular line*, *change the case* of selected text, view the script properties or *print* the text of the script. Each of these operations can be also performed with the corresponding hot keys used.

Implementation of the Find Text and the Replace Text dialogs contributes to more efficient work with the SQL code.

© 2008 EMS Database Management Solutions



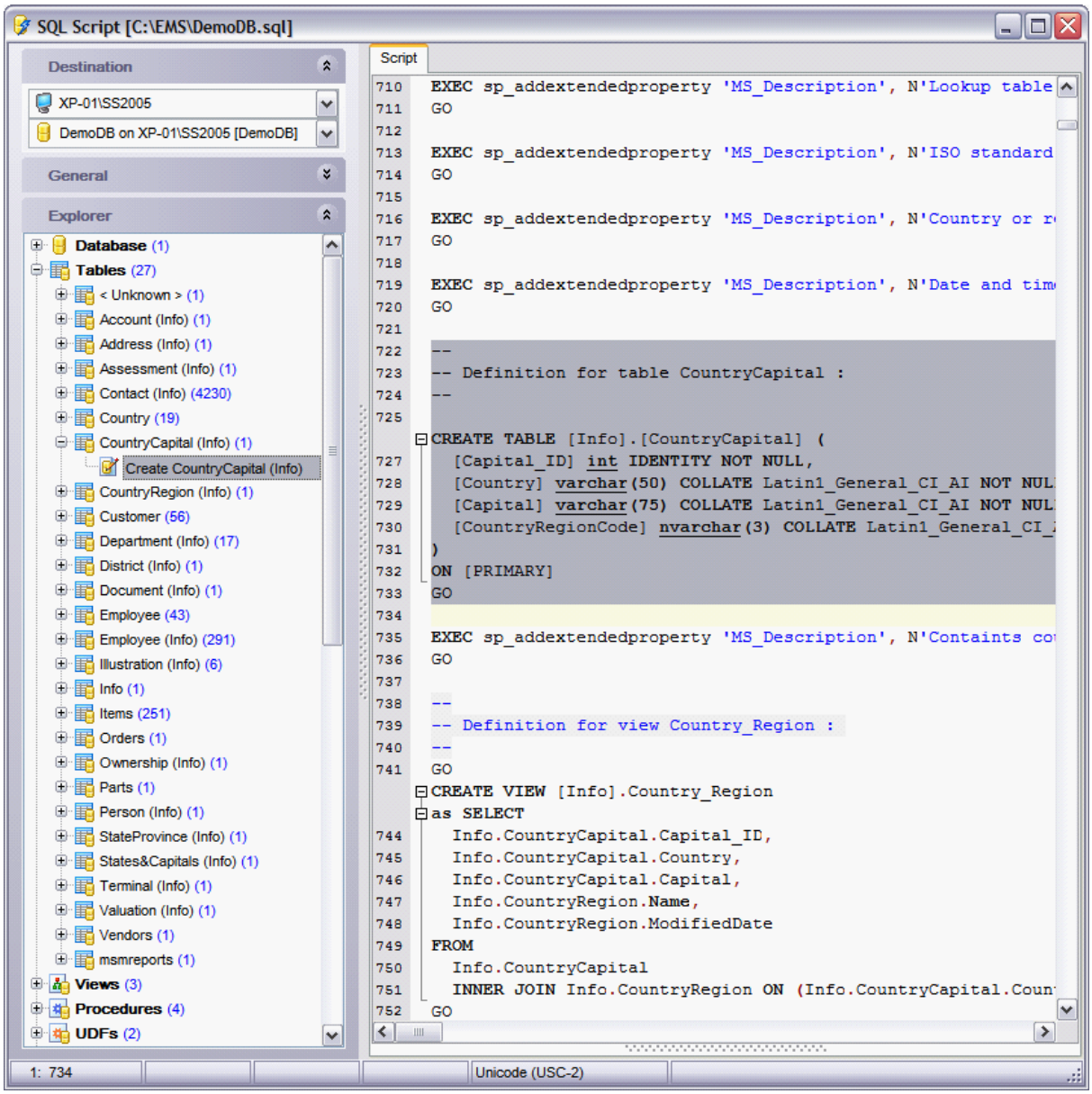
See also:

Using Navigation bar
Using Script Explorer
Script execution

9.1.3.3 Using Script Explorer

The **Explorer** group on the Navigation bar displays the tree of objects, used in the current script and allows you to get to the required script fragment quickly by clicking the object in the tree.

© 2008 EMS Database Management Solutions



Hint: When you click a node in the **Script Explorer** tree, the corresponding SQL statement is highlighted in the editor area. If you double-click a node, the corresponding SQL statement is highlighted, and the current focus is switched to the editor area (the cursor appears after the highlighted statement).

See also:

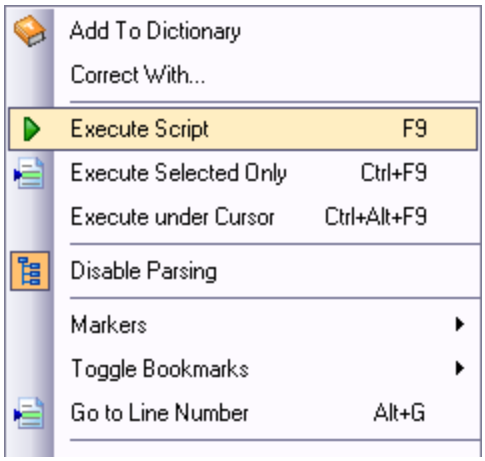
Using Navigation bar
Using the context menu
Script execution

© 2008 EMS Database Management Solutions

9.1.3.4 Script execution

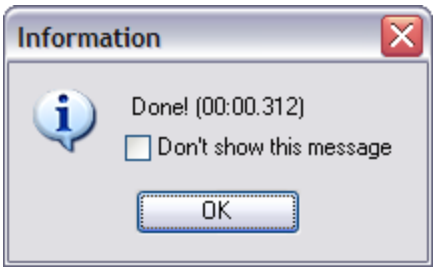
When all the script parameters are set, you can immediately **execute the script** in **SQL Script Editor**.

To execute a script, click the **Execute script** item of the Navigation bar. You can also use the context menu or *F9* hot key for the same purpose.



Note: If the **Execute selected text separately** option (see the Tools | SQL Script section of the Environment Options dialog) is enabled (by default) and a text fragment is currently selected, only this fragment is executed when you click *Execute script* on the Navigation bar or press *F9*. If this option is disabled, the whole script is executed, but you can still execute the selected fragment using the corresponding *Execute Selected Only* item of the context menu or by pressing *Ctrl+F9*.

If the SQL syntax is correct, the script is executed and the *'Done!'* information message appears.



If the syntax contains errors or script cannot be executed, the corresponding error message is displayed in the status bar area at the bottom of the editor window.

Hint: When you select an item from the error list (in the status bar area), the corresponding SQL statement is highlighted in the editor area. If you double-click an item, the corresponding SQL statement is highlighted, and the current focus is switched to the

© 2008 EMS Database Management Solutions

editor area (the cursor appears after the highlighted statement).

Note: **SQL Script Editor** does not show results returned upon SELECT queries execution. Please execute such queries in SQL Editor to see the result dataset.

See also:

Using Navigation bar
Using the context menu
Using Script Explorer

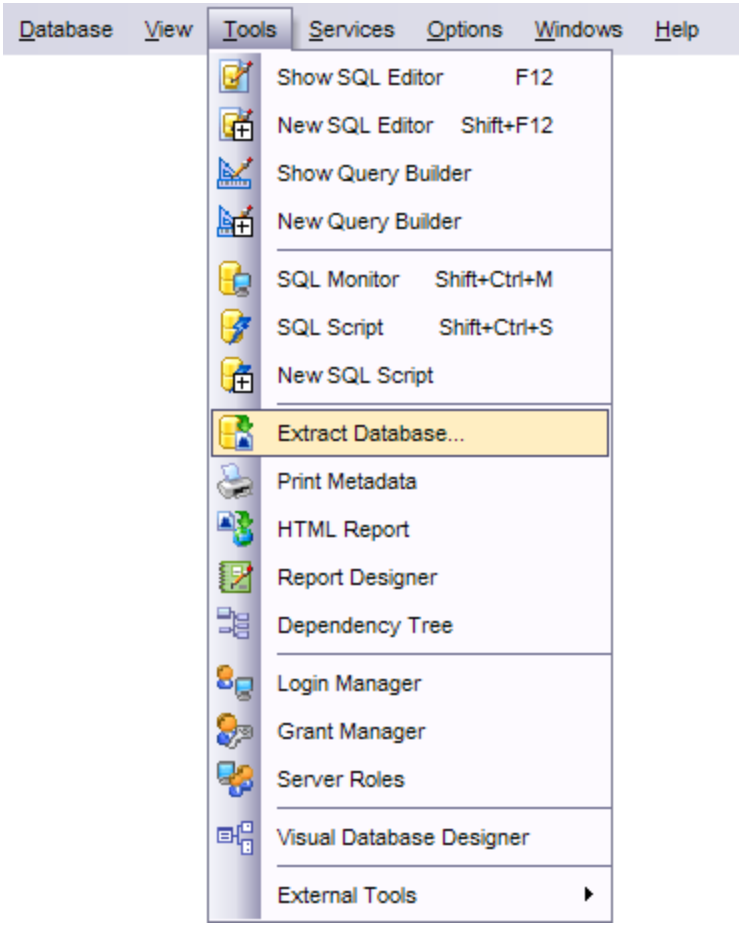
9.1.4 Extract Database Wizard

Extract Database Wizard allows you to extract database objects and/or data to an SQL script, e.g. for backup purposes.

To start the wizard, select the **Tools | Extract Database...** main menu item.

- Selecting a database for extraction
- Specifying destination file name
- Setting extraction mode
- Setting BLOB options
- Selecting objects for metadata extraction
- Selecting objects for data extraction
- Customizing script options
- Start of extraction process
- Using templates

© 2008 EMS Database Management Solutions



Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**Yes**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

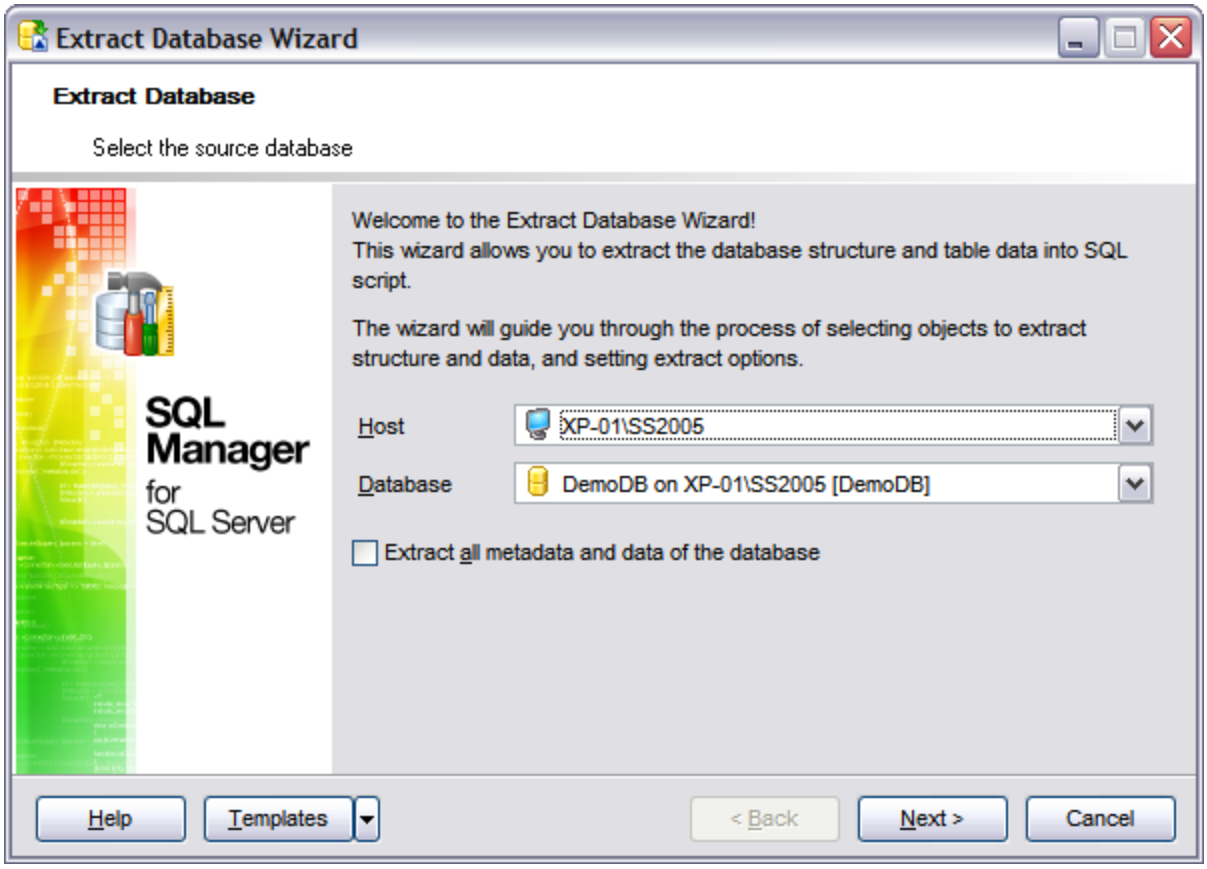
Dependency Tree
Visual Database Designer
SQL Script Editor
Print Metadata
HTML Report Wizard
Reports management
SQL Monitor

© 2008 EMS Database Management Solutions

9.1.4.1 Selecting source database

This step of the wizard allows you to select the **source database** from which metadata and/or data are to be extracted.

If necessary, check the **Extract all metadata and data of the database** option to simplify the wizard.



Click the **Next** button to proceed to the Specifying destination file name step of the wizard.

9.1.4.2 Specifying destination file name

Script destination

This group of options allows you to specify whether the result SQL script will be automatically loaded to SQL Script Editor or saved into a file.

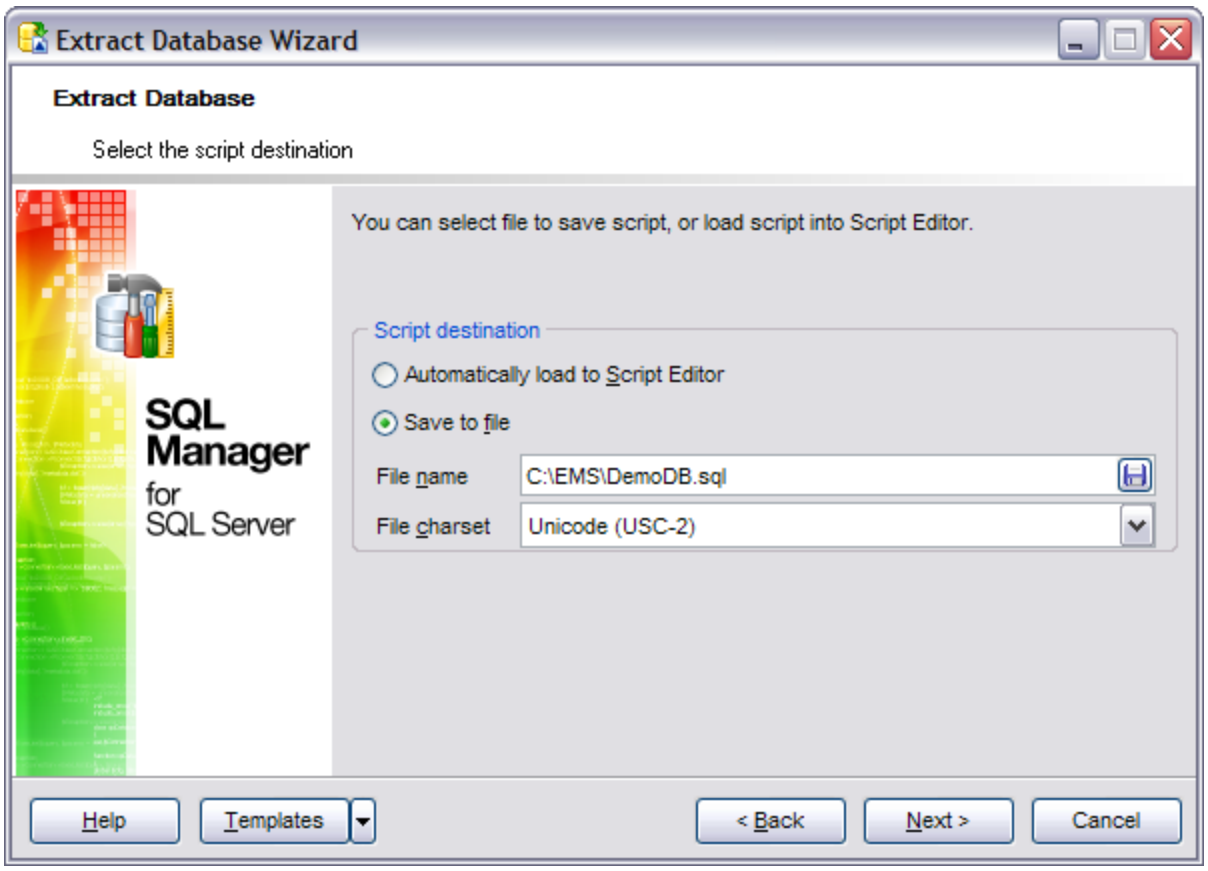
File name

Set a name for the result **.sql* file and type in or use the **Save as...** button to specify the path to this file on your local machine or on a machine in the LAN.

File charset

© 2008 EMS Database Management Solutions

If necessary, use the drop-down list to select the character set to be applied to the output file.

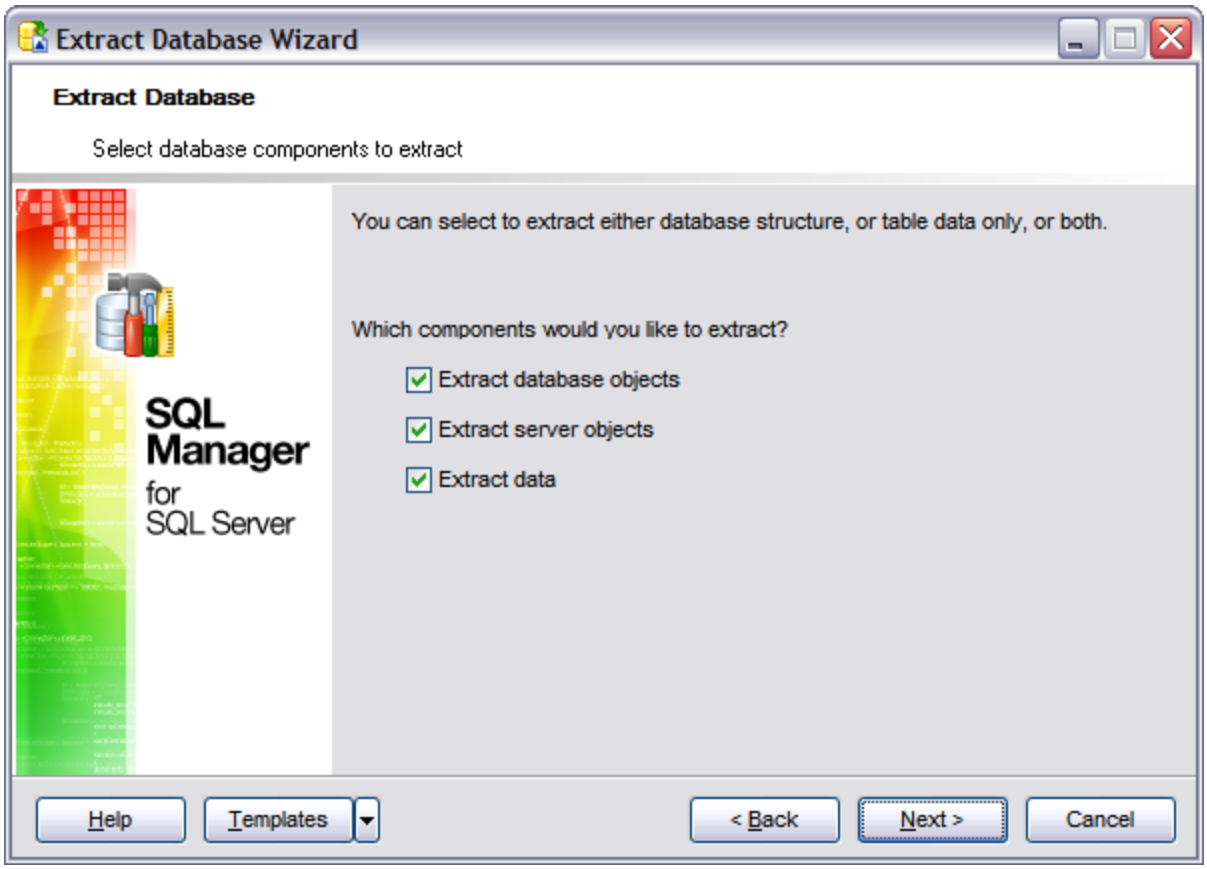


Depending on whether you have checked the ☑ **Extract all metadata and data of the database** option at the Selecting source database step, upon pressing the **Next** button you will either proceed to the next step of the wizard, or you will be immediately forwarded to the Setting BLOB options step, and then to the Customizing script options step of the wizard.

9.1.4.3 Setting extraction mode

This step allows you to specify the **extraction mode**: choose whether *database objects*, *server objects* or *data* only are to be extracted.

© 2008 EMS Database Management Solutions



Click the **Next** button to proceed to the Setting BLOB options step of the wizard.

9.1.4.4 Setting BLOB options

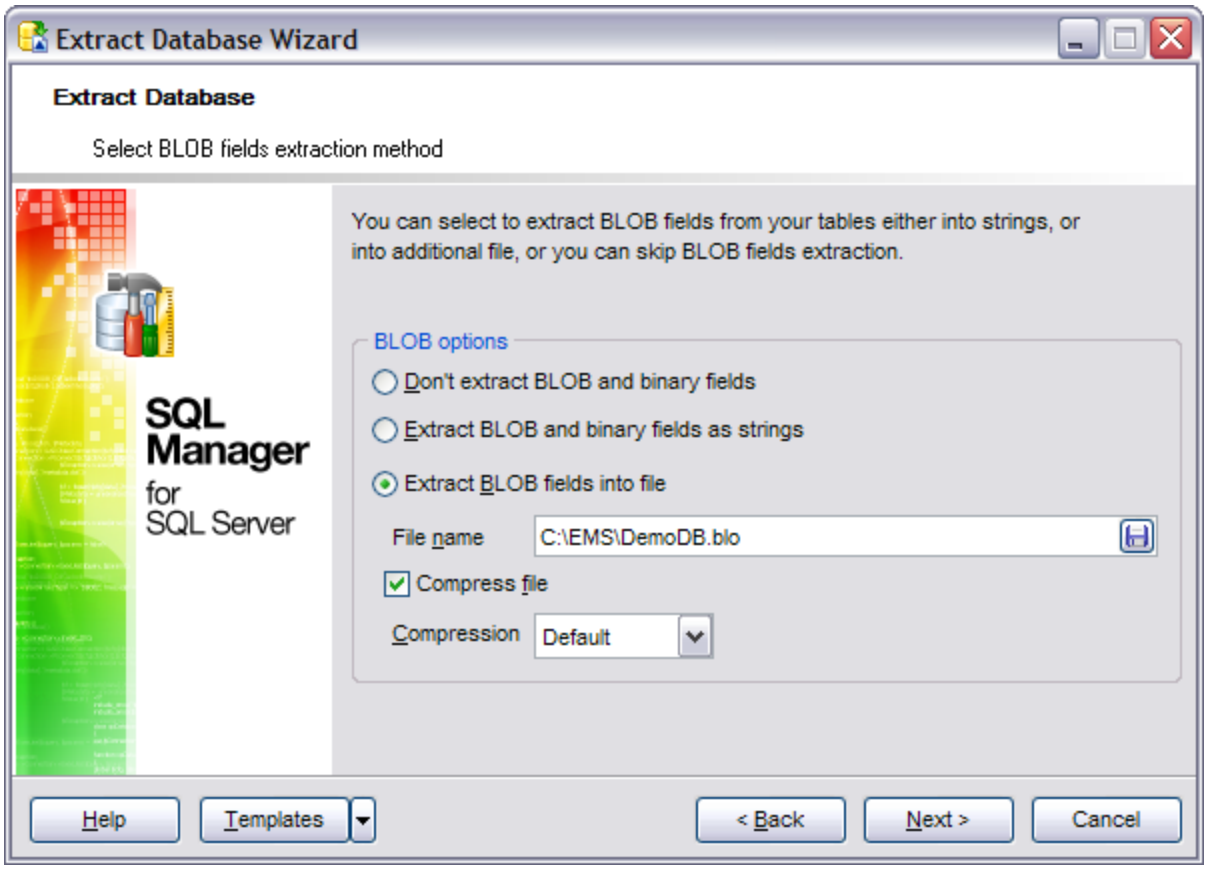
BLOB options

In this group of options you can determine whether BLOB fields are *not to be extracted*, *extracted as strings*, or *extracted into a separate file*. If the latter is selected, you also need to specify the **File name** (the **.blo* file where the BLOB data are to be stored) and the location of the file on your local machine using the **Save as...** button.

Compress file
Check this option if you wish to compress the file containing BLOB data.

Compression
Define the desired compression level to be applied for the file: *None*, *Fastest*, *Default*, *Best*.

© 2008 EMS Database Management Solutions



Click the **Next** button to proceed to Selecting objects for structure extraction.

9.1.4.5 Selecting objects for structure extraction

This step of the wizard allows you to **select objects for metadata extraction**.

Note that this step is only available if the **Extract all metadata and data of the database** option was unchecked when selecting the source database.

Extract all objects
Adds all objects of the database to structure extraction process.

Extract all objects of schema...
Adds all objects of a schema to structure extraction process.

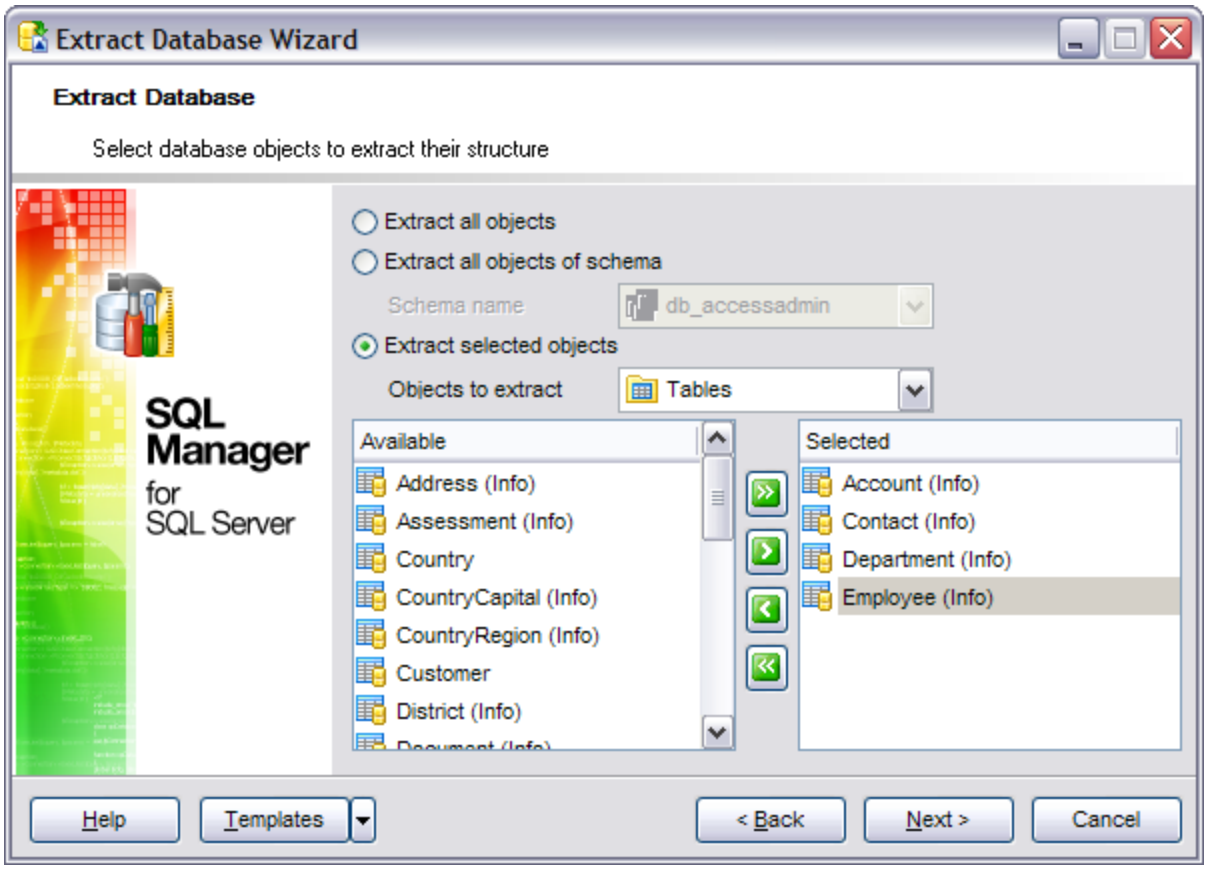
Schema name
Use the drop-down list to select the schema to extract all objects from.

Extract selected objects
Adds only selected objects to structure extraction process.

Objects to extract

© 2008 EMS Database Management Solutions

Use the drop-down list to select the type of objects to be extracted.
To select an object, you need to move it from the **Available** list to the **Selected** list. Use the buttons or drag-and-drop operations to move the objects from one list to another.



Click the **Next** button to proceed to Selecting objects for data extraction.

9.1.4.6 Selecting server objects for extraction

This step of the wizard allows you to **select objects for metadata extraction**.

Note that this step is only available if the **Extract all metadata and data of the database** option was unchecked when selecting the source database.

Extract all objects
Adds all objects of the database to structure extraction process.

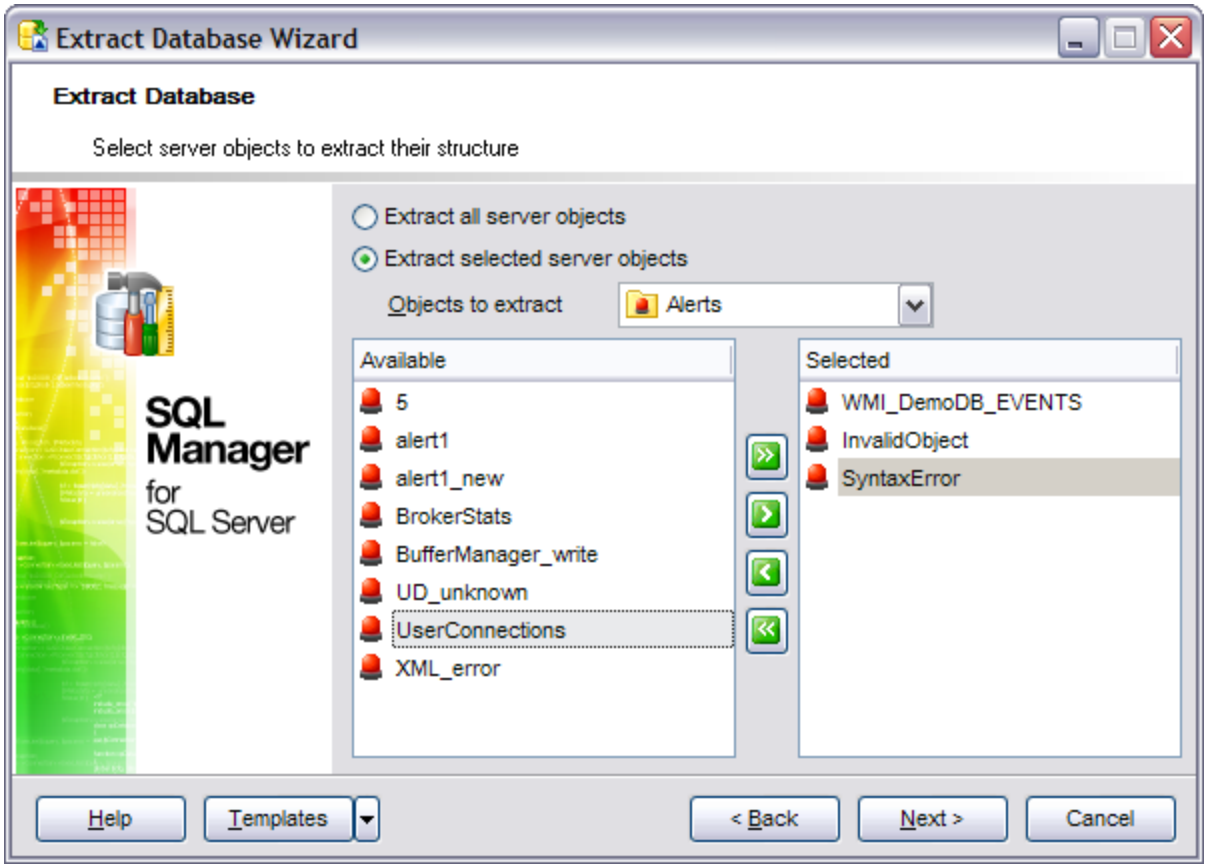
Extract all objects of schema...
Adds all objects of a schema to structure extraction process.

Schema name
Use the drop-down list to select the schema to extract all objects from.

© 2008 EMS Database Management Solutions

Extract selected objects
Adds only selected objects to structure extraction process.

Objects to extract
Use the drop-down list to select the type of objects to be extracted.
To select an object, you need to move it from the **Available** list to the **Selected** list. Use the buttons or drag-and-drop operations to move the objects from one list to another.



Click the **Next** button to proceed to Selecting objects for data extraction.

9.1.4.7 Selecting objects for data extraction

This step of the wizard allows you to **select tables for data extraction**.

Note that this step is only available if the **Extract all metadata and data of the database** option was unchecked when selecting the source database.

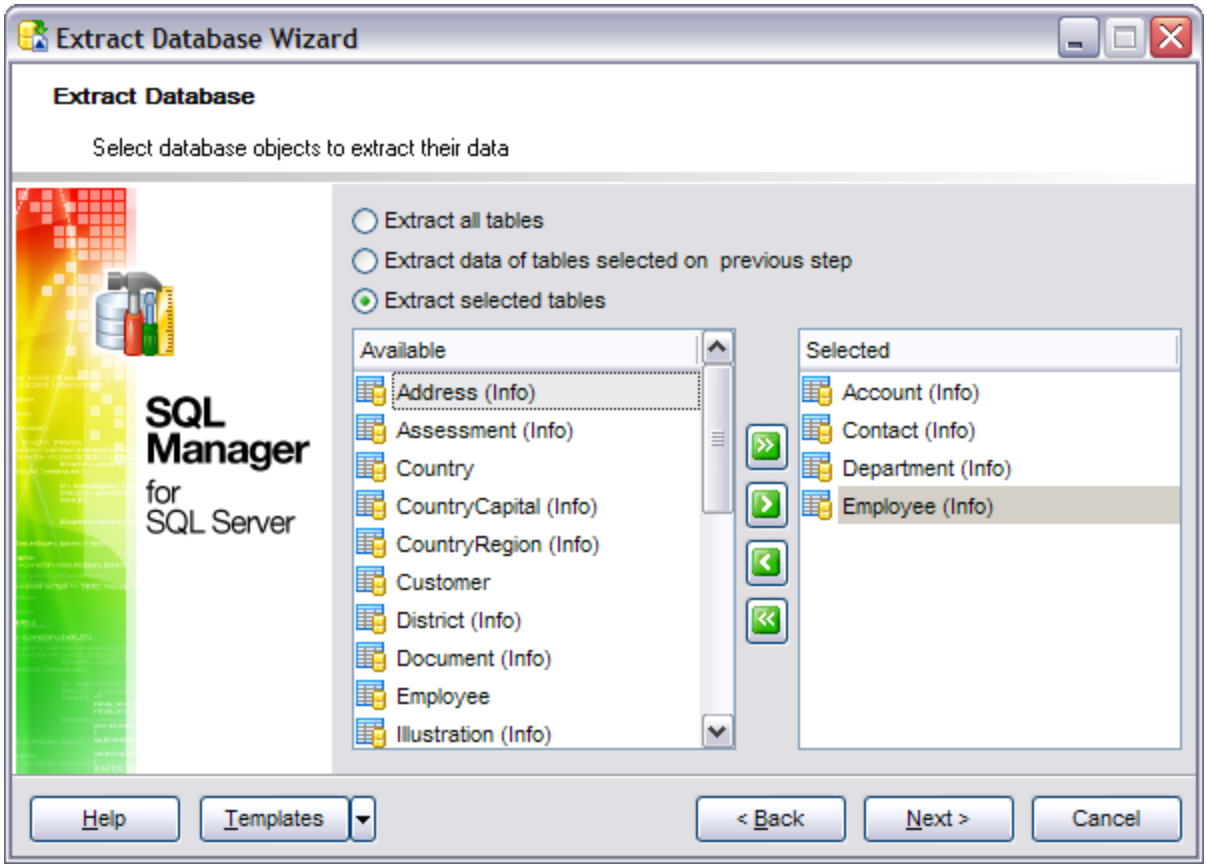
Extract data of the selected tables
Adds only selected tables to data extraction process.
To select a table, you need to move it from the **Available** list to the **Selected** list. Use

© 2008 EMS Database Management Solutions

the buttons or drag-and-drop operations to move the tables from one list to another.

Extract data of all tables
Adds all tables of the database to data extraction process.

Extract data of tables selected on the previous step
Adds only the tables selected for metadata extraction.



Click the **Next** button to proceed to the Customizing script options step of the wizard.

9.1.4.8 Customizing script options

This step allows you to customize common **script options** and **data options** for the extraction process.

Script options

Generate "CREATE DATABASE" statement
Check this option to add the *CREATE DATABASE* statement to the result script.

Generate DROP statements

© 2008 EMS Database Management Solutions

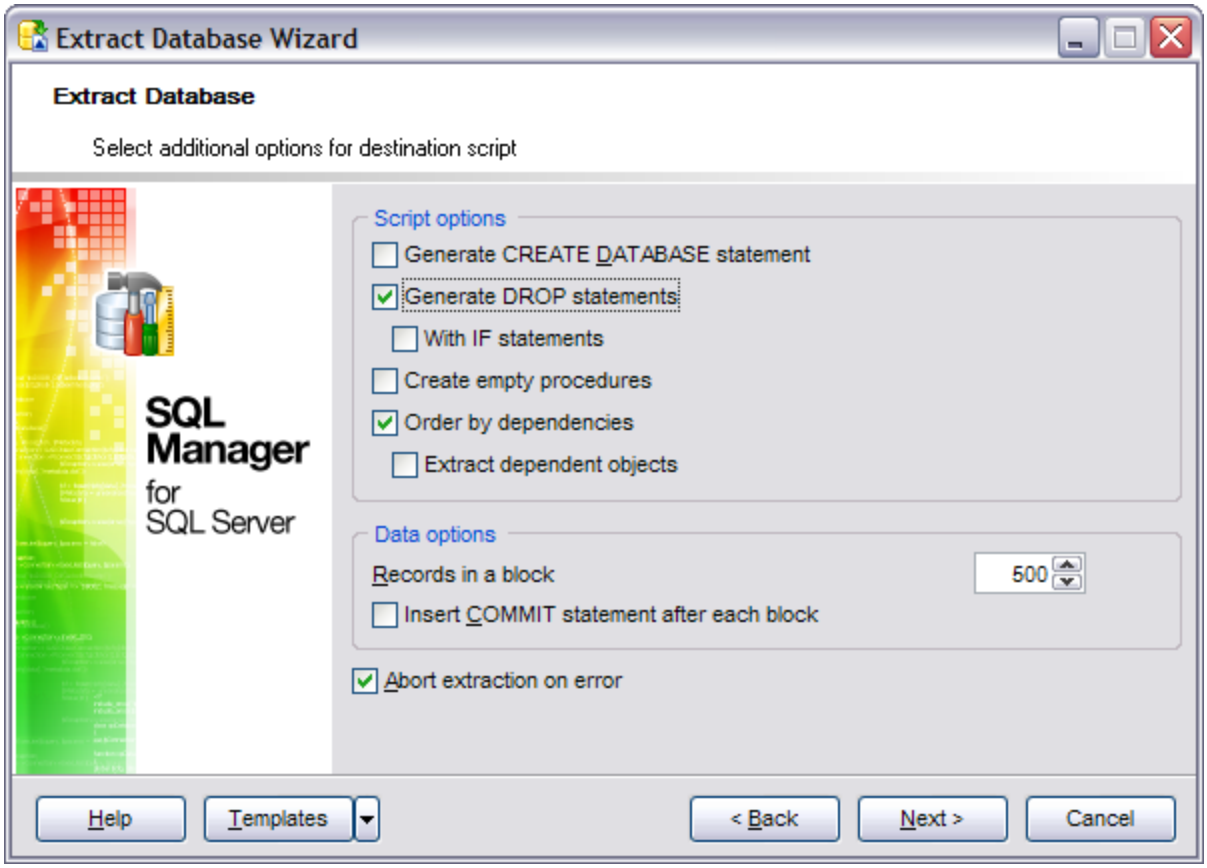
Check the option to add the *DROP* statements for the extracted objects in the result script.

With IF statements
Check this option to add the *IF* keyword to the *DROP* statements in the result script.

Create empty procedures
This option specifies whether procedures with empty bodies are included in the result script or not.

Order by dependencies
This option specifies whether objects should be ordered by their dependencies in the result script or not.

Extract dependent objects
This option determines objects' dependencies usage in the extraction process. Check the option to extract all objects that the selected objects depend on.



Data options

Records in a block / Insert "COMMIT" statement after each block
These controls allow you to define whether the *COMMIT* statement is added to the script or not, and to specify the number of records in each block to be supplemented with this statement.

© 2008 EMS Database Management Solutions

Abort extraction on error
This option determines whether the extraction process should be stopped or forced to continue if an error occurs.

Click the **Next** button to proceed to the last step of the wizard.

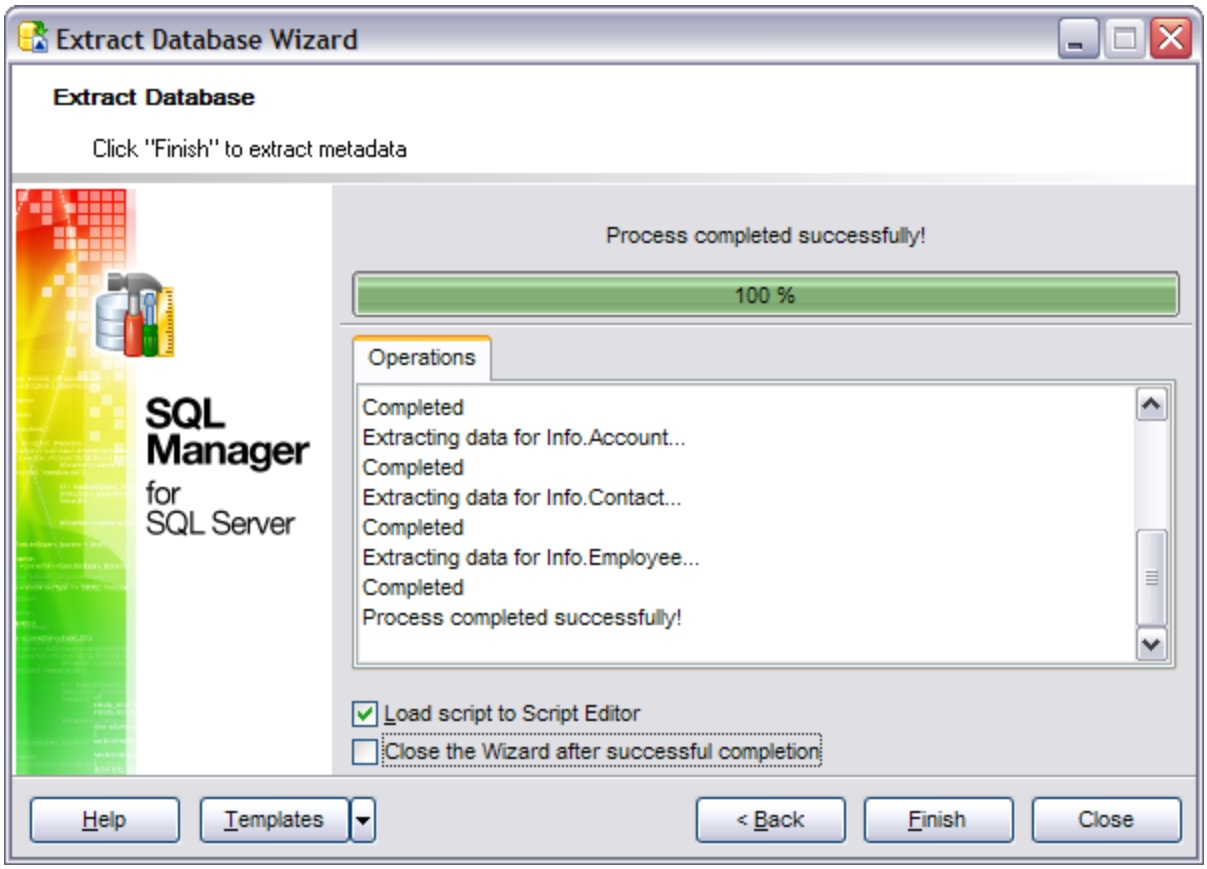
9.1.4.9 Start of extraction process

This step of the wizard is intended to inform you that all extraction options have been set, and you can start the extraction process.

Load script to Script Editor
Check this option to load the result script to SQL Script Editor.

Close the wizard after successful completion
If this option is selected, the wizard is closed automatically when the extraction process is completed.

The **Operations** tab allows you to view the log of operations and errors (if any).



If necessary, you can save a template for future use.

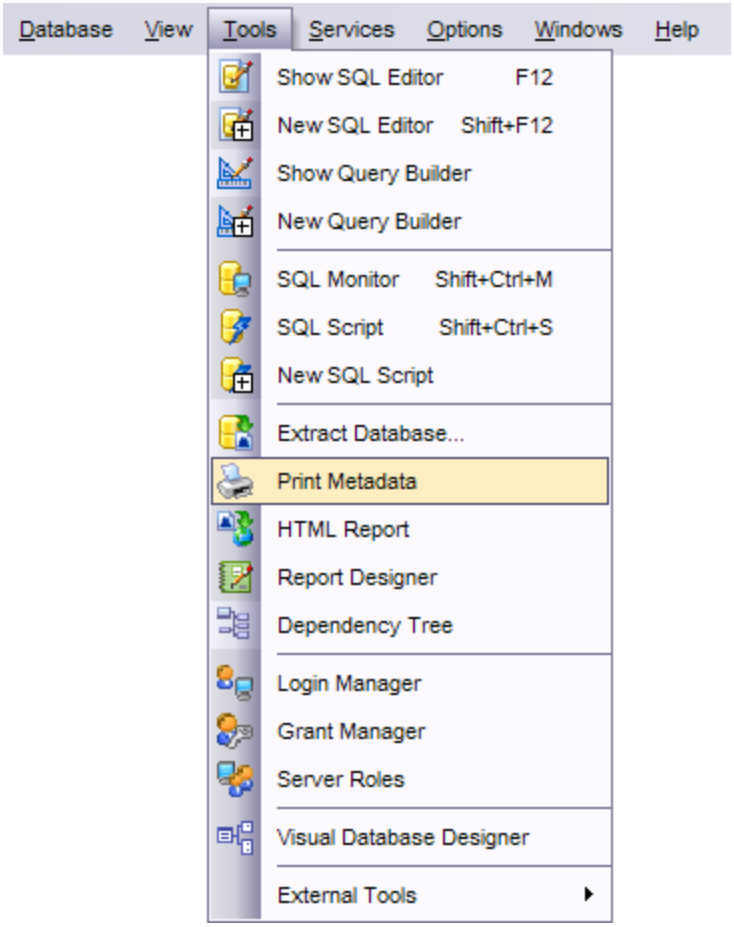
© 2008 EMS Database Management Solutions

Click the **Finish** button to run the extraction process.

9.1.5 Print Metadata

Print Metadata allows you to generate and print metadata reports of any database/server object(s).

To open the window, select the **Tools | Print Metadata** main menu item, or use the **Print Metadata** button on the main toolbar.



Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**No**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

© 2008 EMS Database Management Solutions

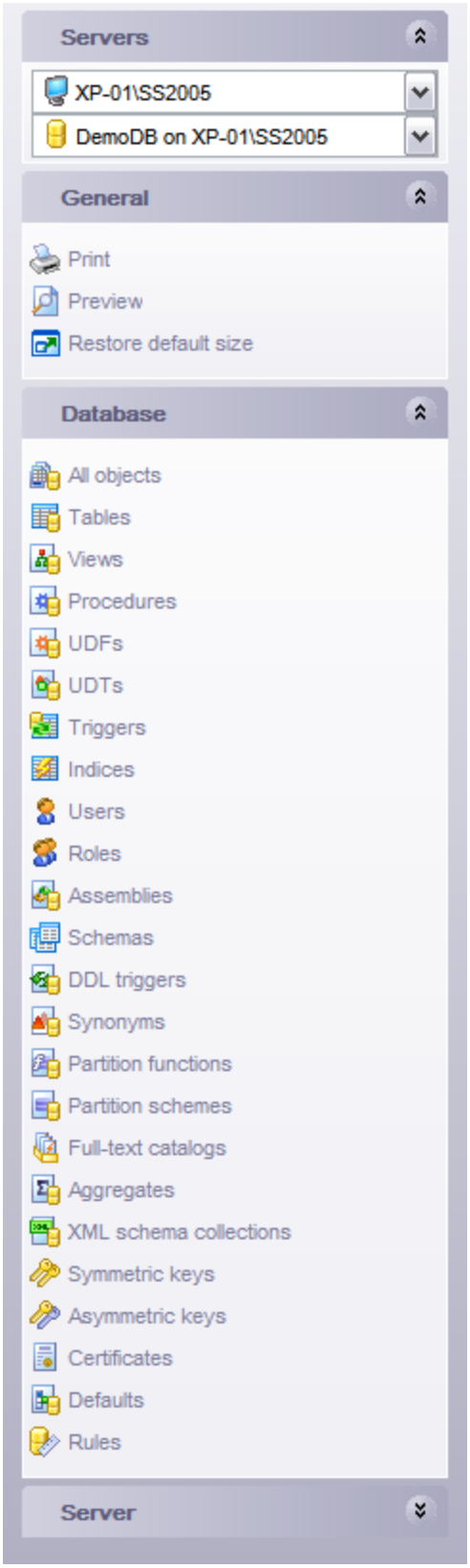
See also:
Dependency Tree
Visual Database Designer
SQL Script Editor
Extract Database Wizard
HTML Report Wizard
Reports management
SQL Monitor

9.1.5.1 Using Navigation bar

The **Navigation bar** of the **Print Metadata** window allows you to:

- select the host and database for the printing report;
- print metadata of the selected object(s);
- preview the printing report;
- cancel report building (if it is in progress);
- restore the default size of the window;
- filter database objects by type;
- filter server objects by type.

© 2008 EMS Database Management Solutions



© 2008 EMS Database Management Solutions

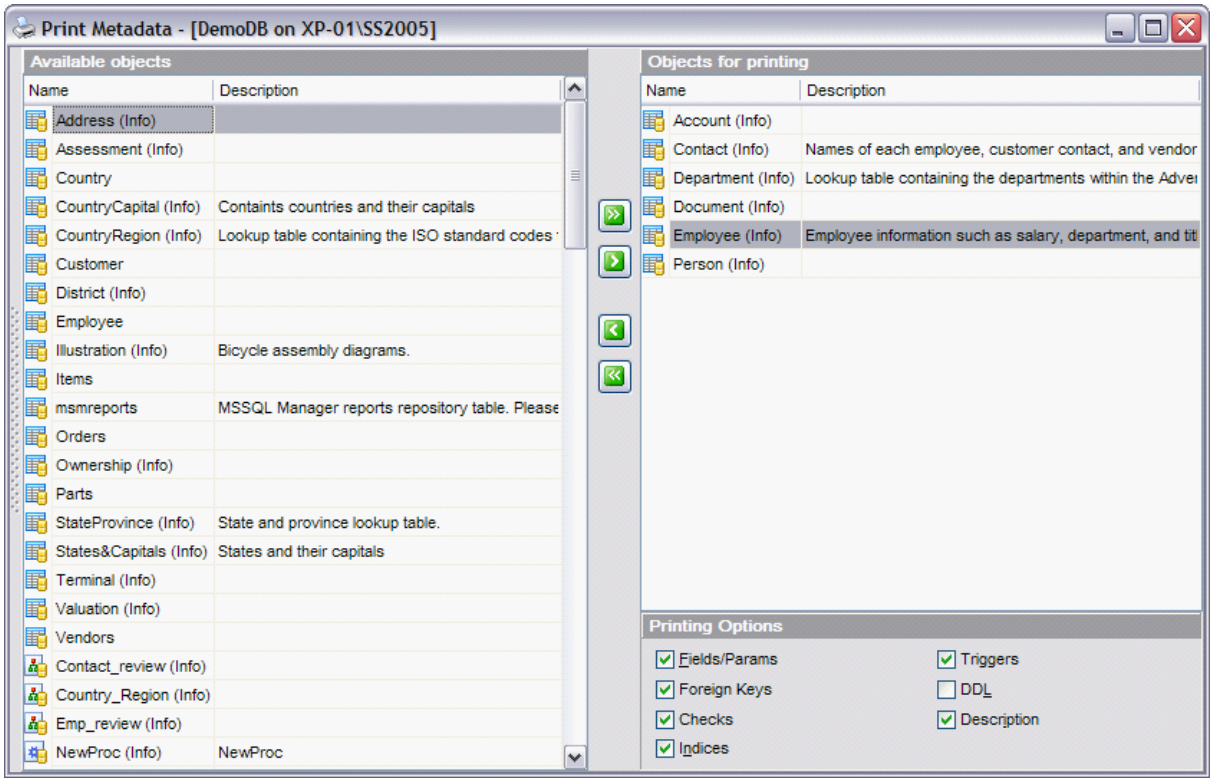
See also:
Printing options

9.1.5.2 Printing options

The **Print Metadata** window allows you to select the database objects for printing metadata.
To select an object, you need to move it from the **Available objects** list to the **Objects for printing** list. Use the buttons or drag-and-drop operations to move the objects from one list to another.

After you select one or several objects, the **Printing Options** pane appears at the bottom.

Printing Options
Select an object in the **Objects for printing** list and specify items to be included into the printing report: *DDL*, *Description* (for all database objects), *Fields/Params*, *Foreign Keys*, *Checks*, *Indexes*, *Triggers* (for tables).

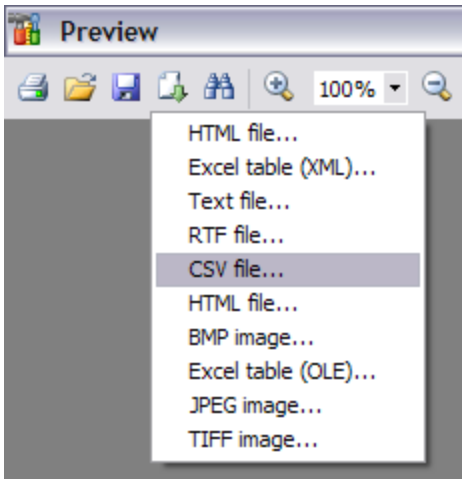


Note that you can *make a preview* of the printing report and *print metadata* for objects of the selected type using the corresponding items of the Navigation bar.

Use the **Export** toolbar button to export the preview content to any of the available formats: *HTML file*, *Excel file*, *Text file*, *RTF file*, *CSV file*, *HTML file*, *BMP image*, *Excel*

© 2008 EMS Database Management Solutions

table (OLE), *JPEG image*, *TIFF image*.



See also:
Using Navigation bar

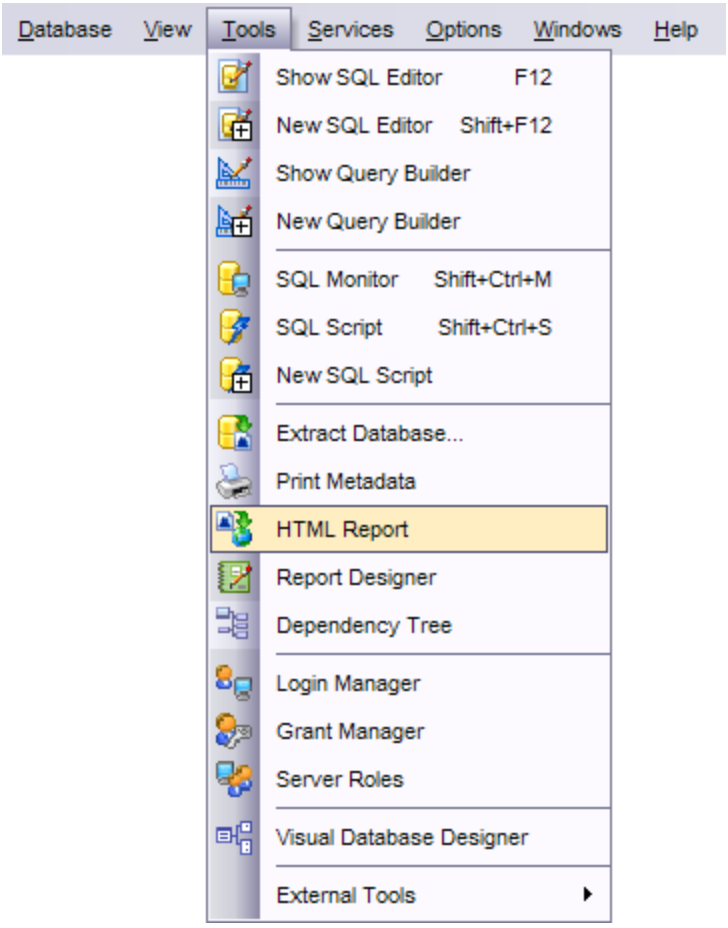
9.1.6 HTML Report Wizard

HTML Report wizard allows you to generate a detailed HTML report of the selected database objects.

To start **HTML Report Wizard**, select the **Tools | HTML Report** main menu item, or use the **HTML Report** button on the main toolbar.

- Selecting database and directory
- Selecting object types
- Specifying CSS for HTML report
- Setting additional report options
- Creating HTML report
- Using templates

© 2008 EMS Database Management Solutions



Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**No**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

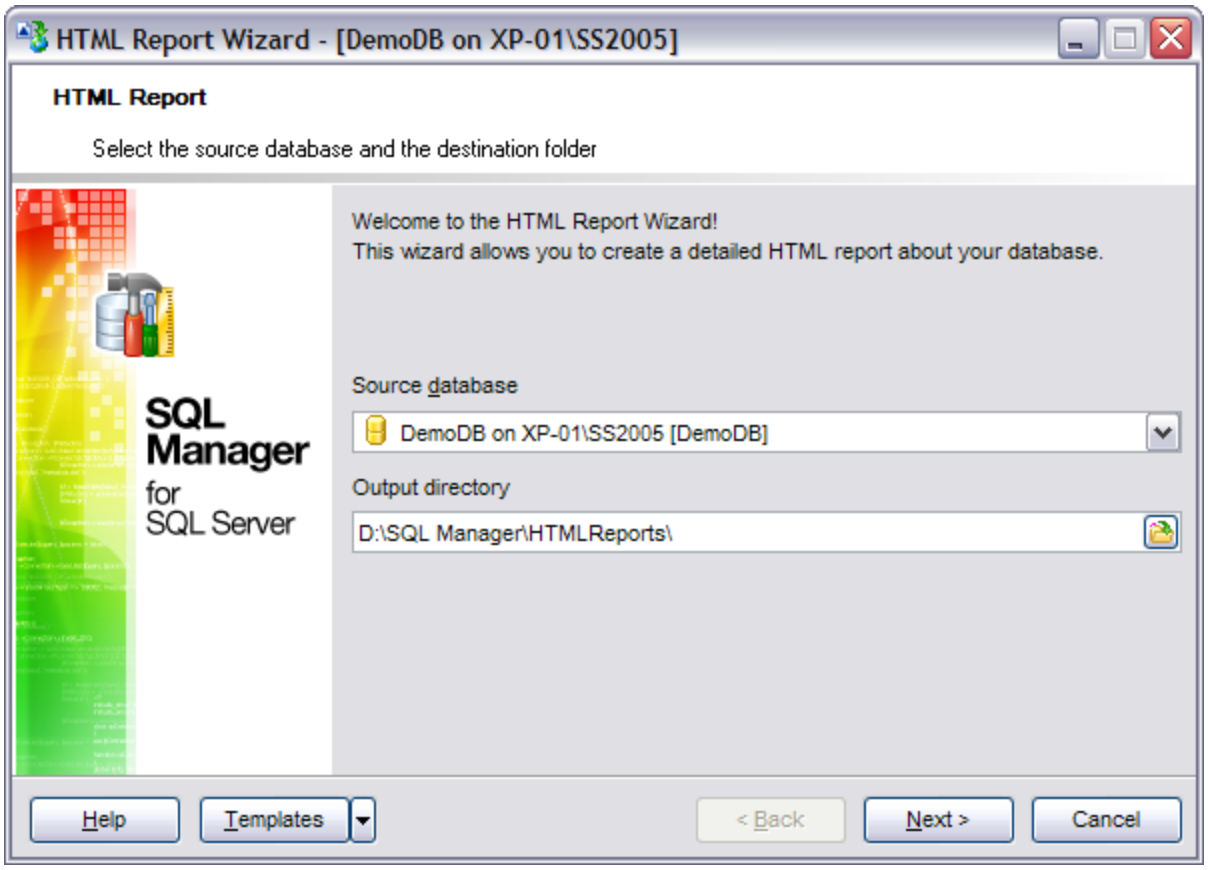
Dependency Tree
Visual Database Designer
SQL Script Editor
Extract Database Wizard
Print Metadata
Reports management
SQL Monitor

© 2008 EMS Database Management Solutions

9.1.6.1 Selecting database and directory

At this step of the wizard you should select the **source database** for the report.

Output directory
Type in or use the button to specify the path to the output directory for the result HTML files using the **Browse for Folder** dialog.

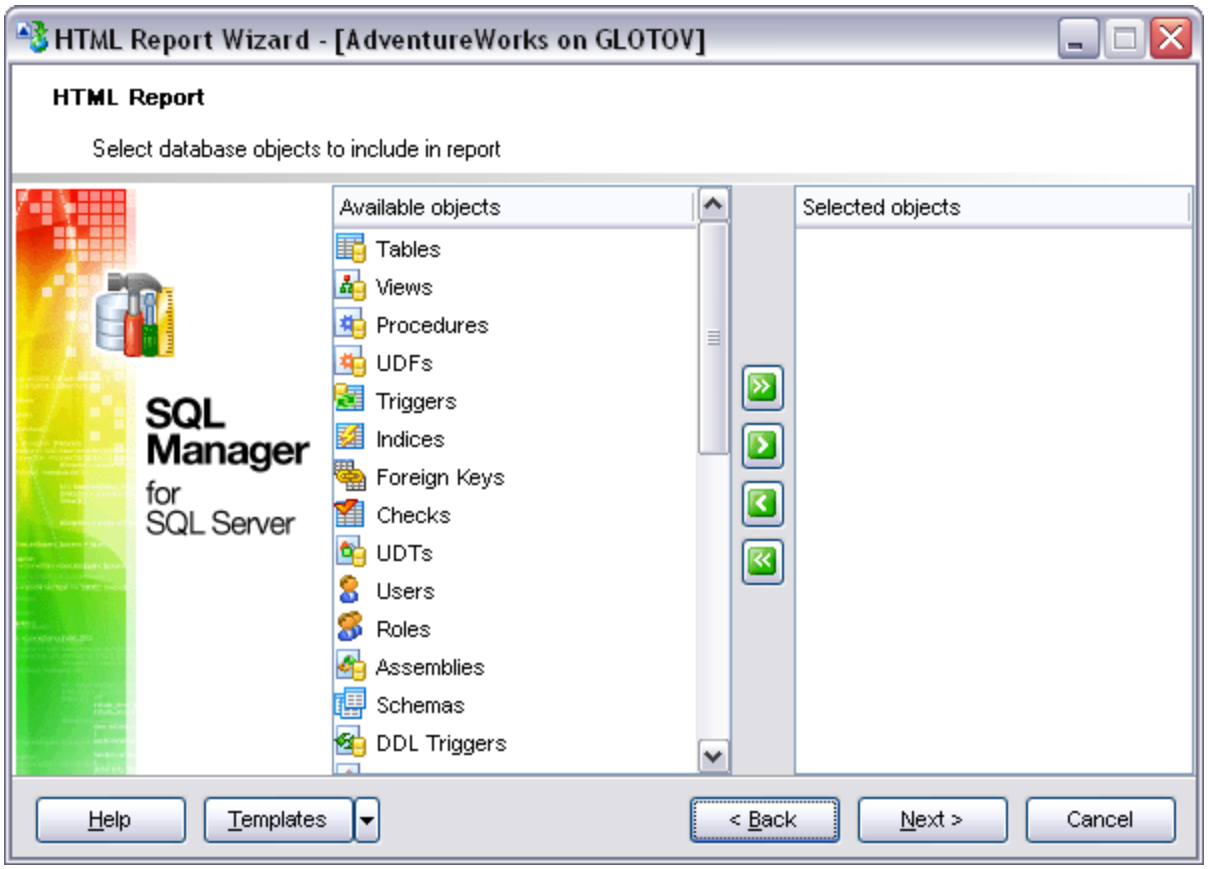


Click the **Next** button to proceed to the Selecting object types step of the wizard.

9.1.6.2 Selecting object types

At this and the next steps you need to

© 2008 EMS Database Management Solutions



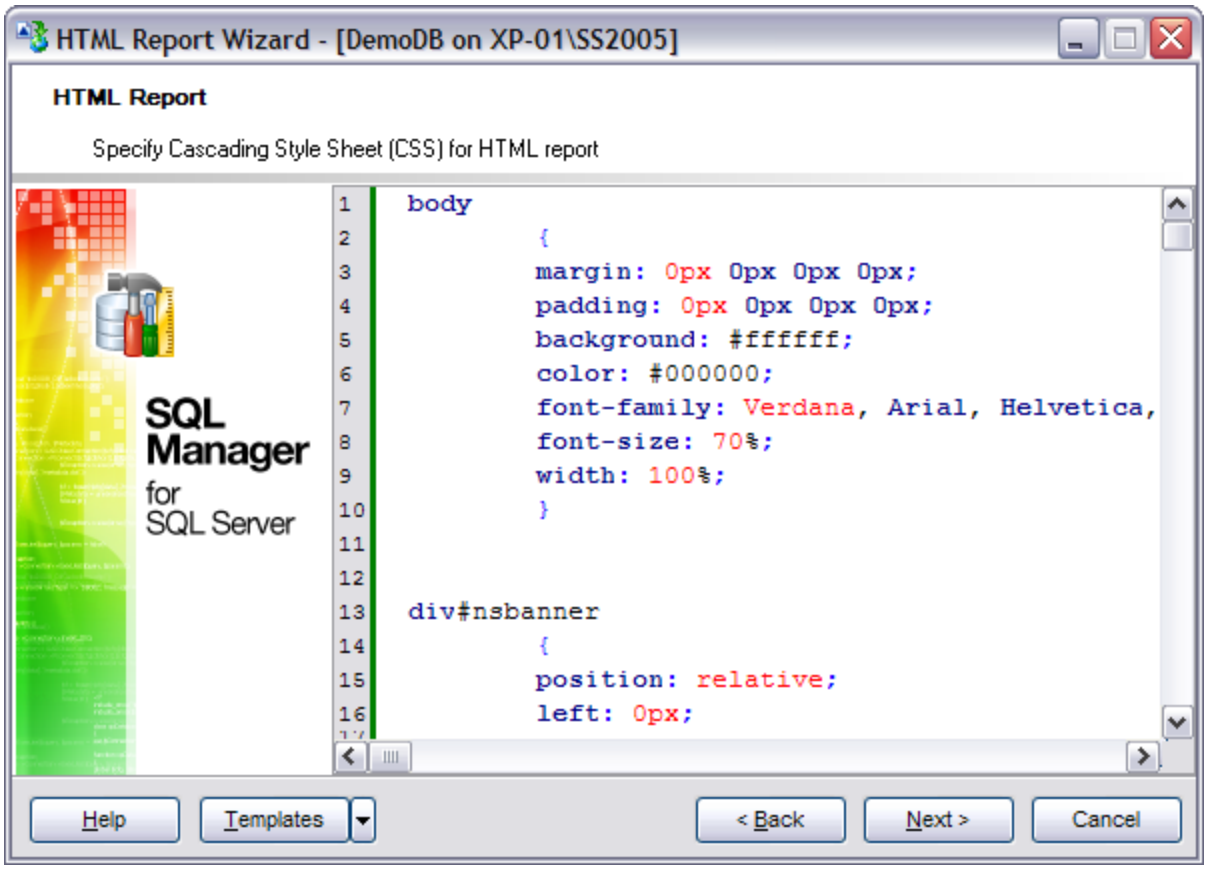
For your convenience the *Check All* and *Unckeck All* buttons are implemented at the bottom of the objects list area.

Click the **Next** button to proceed to the Specifying CSS step of the wizard.

9.1.6.3 Specifying CSS

This step of the wizard allows you *to edit the CSS (Cascading Style Sheet) file* that will be used by the result HTML report.

© 2008 EMS Database Management Solutions



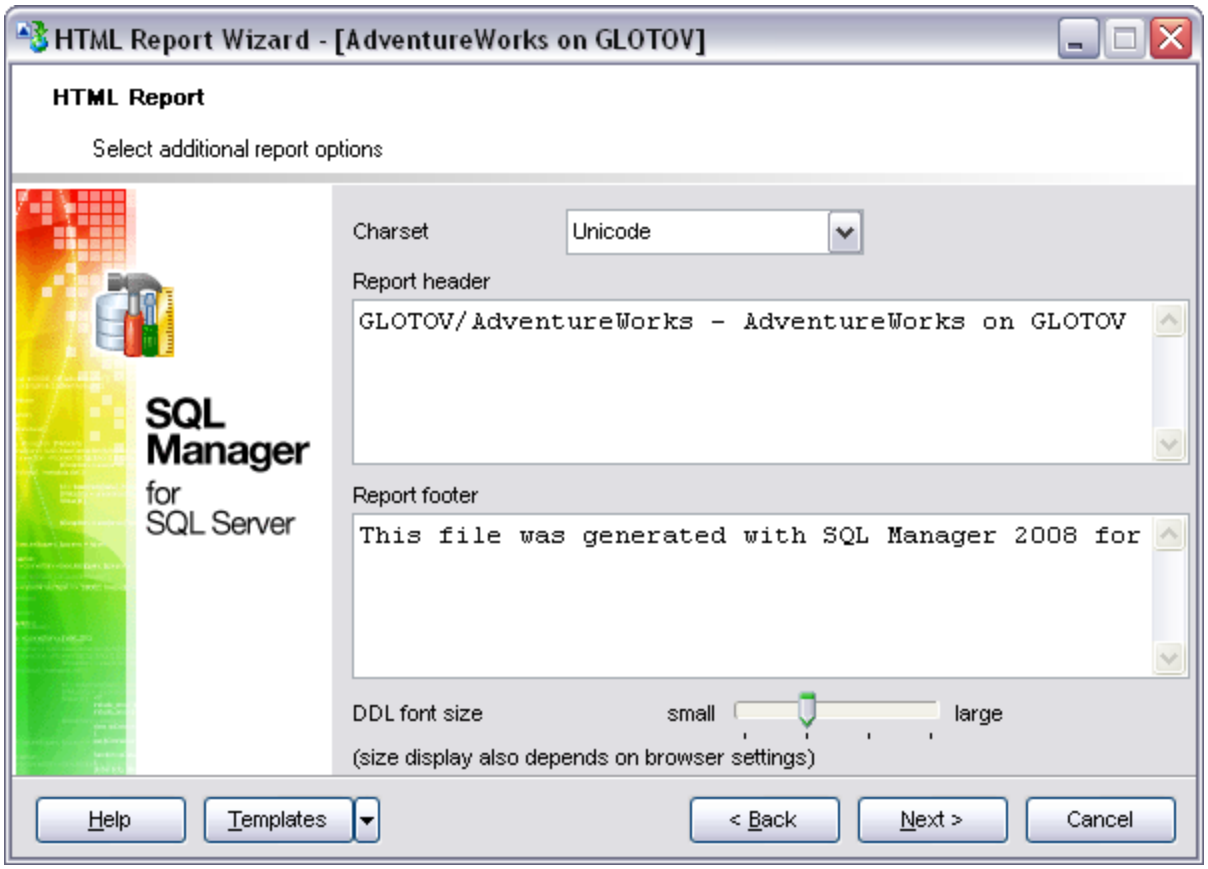
Click the **Next** button to proceed to the Setting additional report options step of the wizard.

9.1.6.4 Setting additional report options

Use this step of the wizard to set additional HTML report options.

Use the **Charset** drop-down list to select the preferable character set to be applied to the result HTML report.

© 2008 EMS Database Management Solutions



If necessary, you can set optional text to **Report header** and **Report footer** of the result HTML report. For your convenience the default header and footer text is already available. If necessary, you can edit this text according to your needs. You can also change the **DDL font size** if needed.

Click the **Next** button to proceed to Creating HTML report.

9.1.6.5 Creating HTML report

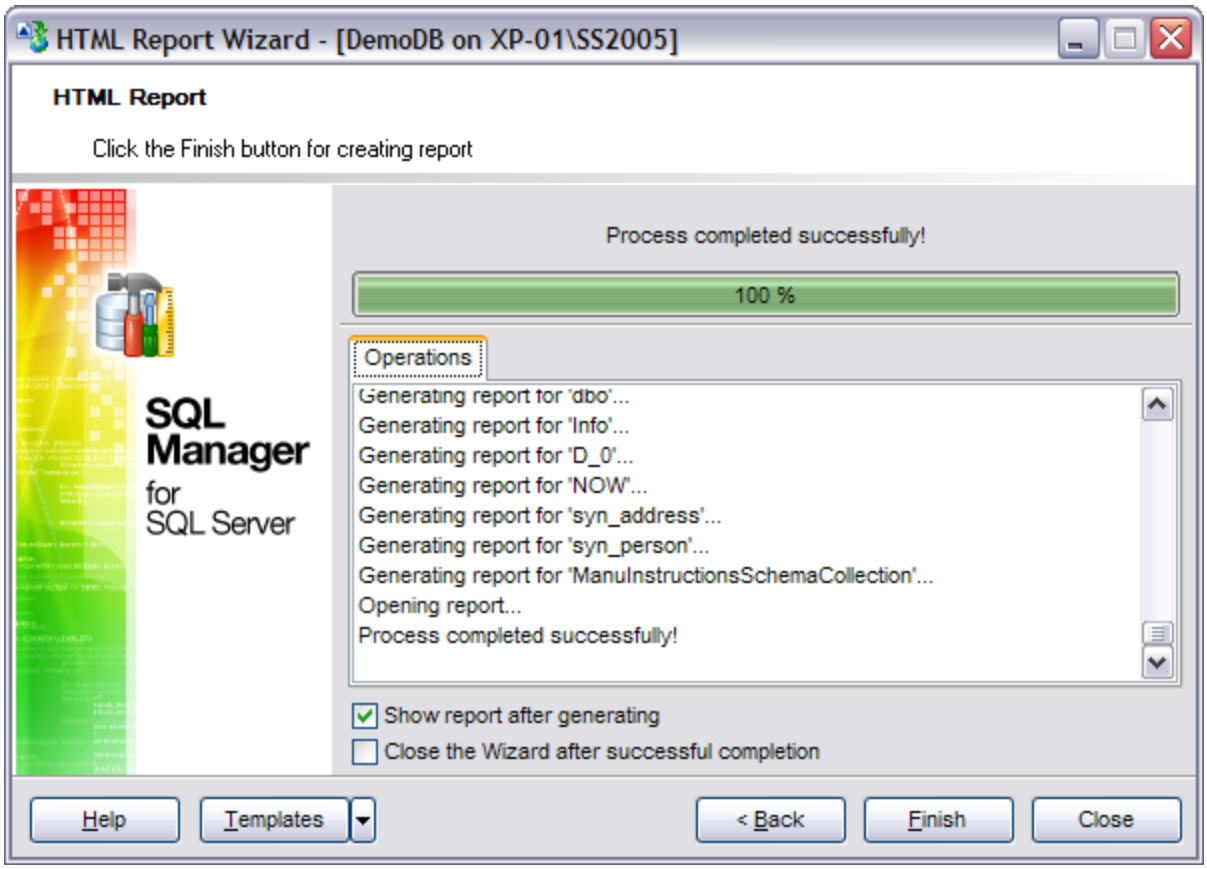
This step of the wizard is intended to inform you that all necessary options have been set, and you can start the process.

☑ **Show report after generating**
This option opens the result report in your default browser after generating.

☑ **Close the wizard after successful completion**
If this option is selected, the wizard is closed automatically when the creating HTML report process is completed.

The **Operations** tab allows you to view the log of operations and errors (if any).

© 2008 EMS Database Management Solutions



If necessary, you can save a template for future use.

Click the **Finish** button to run the process.

9.1.7 Reports management

SQL Manager 2008 for SQL Server provides several tools for efficient *reports management*:

- **Create Report Wizard**
  This tool is used to simplify the process of creating reports.
- **Report Designer**
  It is a basic tool for creating powerful reports.
- **Report Viewer**
  Allows you to view created reports.

Reports can be stored either in the database (table *xxxreports* will be created to store them) or in a folder on your hard drive specified on the Directories page of the Database Registration Info dialog.

See also:

© 2008 EMS Database Management Solutions

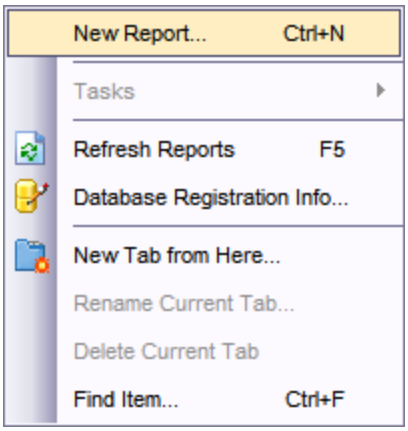
Dependency Tree
Visual Database Designer
SQL Script Editor
Extract Database Wizard
Print Metadata
HTML Report Wizard
SQL Monitor

9.1.7.1 Create Report Wizard

Using **Create Report Wizard** you can create a report containing required datasets, bands and fields on them, with a definite report style applied.

To start the wizard, select the **Database | New Object...** main menu item, then select **Report** in the Create New Object dialog. Alternatively, you can right-click the **Reports** node of the DB Explorer tree and select the **New Report...** item from the context menu.

- Specifying report name and options
- Selecting report bands
- Selecting report style
- Specifying paper settings
- Specifying margins
- Specifying other page settings



Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**No**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

© 2008 EMS Database Management Solutions

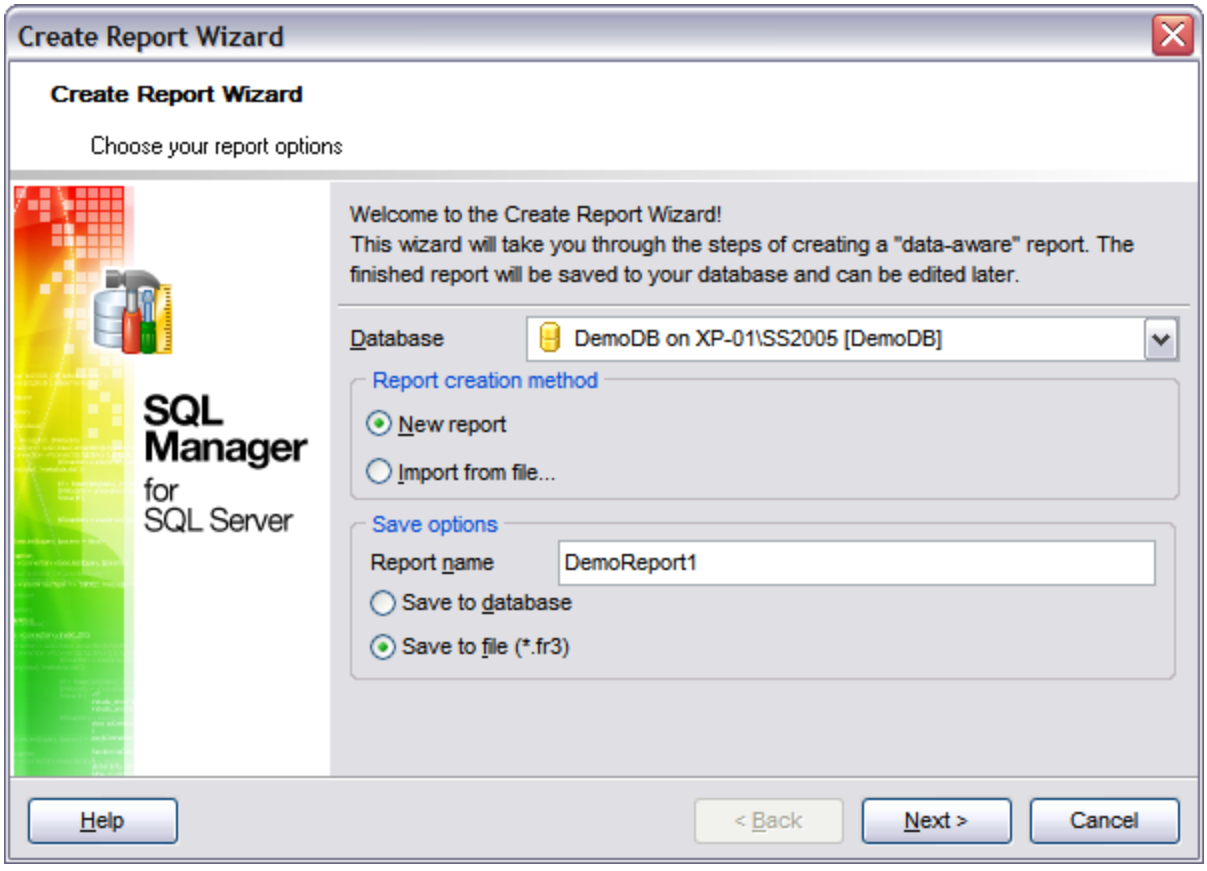
See also:
Report Designer
Report Viewer

9.1.7.1.1 Specifying database name and report options

Select the source **database** for adding a report and choose the action you need to perform: either *create a new report* or *import an existing report from file*.

Set the **name** for the new report and specify the save options for it:

Save to database
The report will be created on the server inside the database.
Save to file
If this option is selected, the report will be saved as a **.fr3* file to the directory specified on the Directories page of the Database Registration Info dialog.



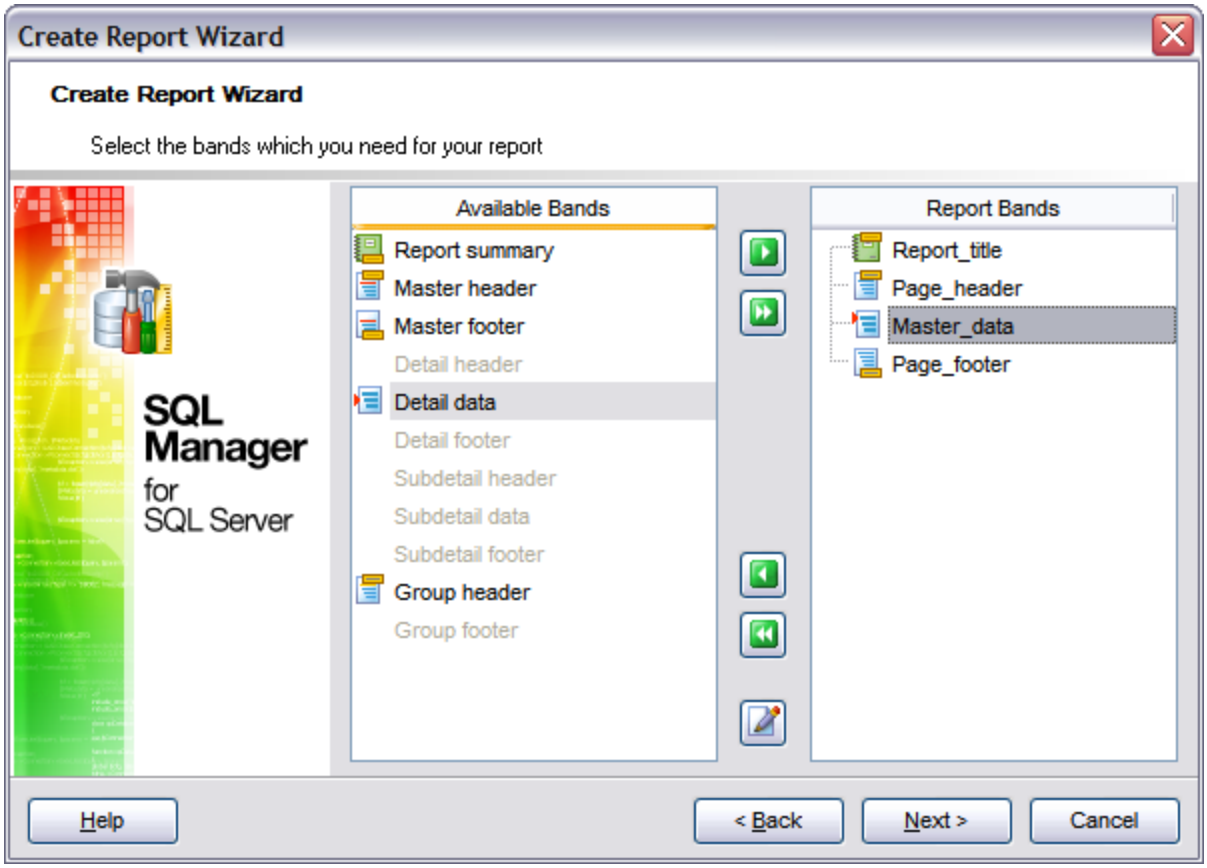
Click the **Next** button to proceed to the Selecting report bands step of the wizard.

© 2008 EMS Database Management Solutions

9.1.7.1.2 Selecting report bands

This step of the wizard allows you to select the bands to be included in the report. To select a band, you need to move it from the **Available Bands** list to the **Report Bands** list. Use the buttons or drag-and-drop operations to move the fields from one list to another.

Use the **Edit** button to create datasets for 'data' bands using Query Builder.



Brief information about bands functionality is listed below. See **FastReport Help** for more information.

Name	**Functionality**
Report title	*Prints once at the beginning of report*
Report summary	*Prints once at the end of report*
Page header	*Prints at the top of each page*
Page footer	*Prints at the bottom of each page*
Master header	*Prints at the beginning of master list*
Master data	*Data rows of master list*
Master footer	*Prints at the end of master list*
Detail header	*Prints at the beginning of detail list*

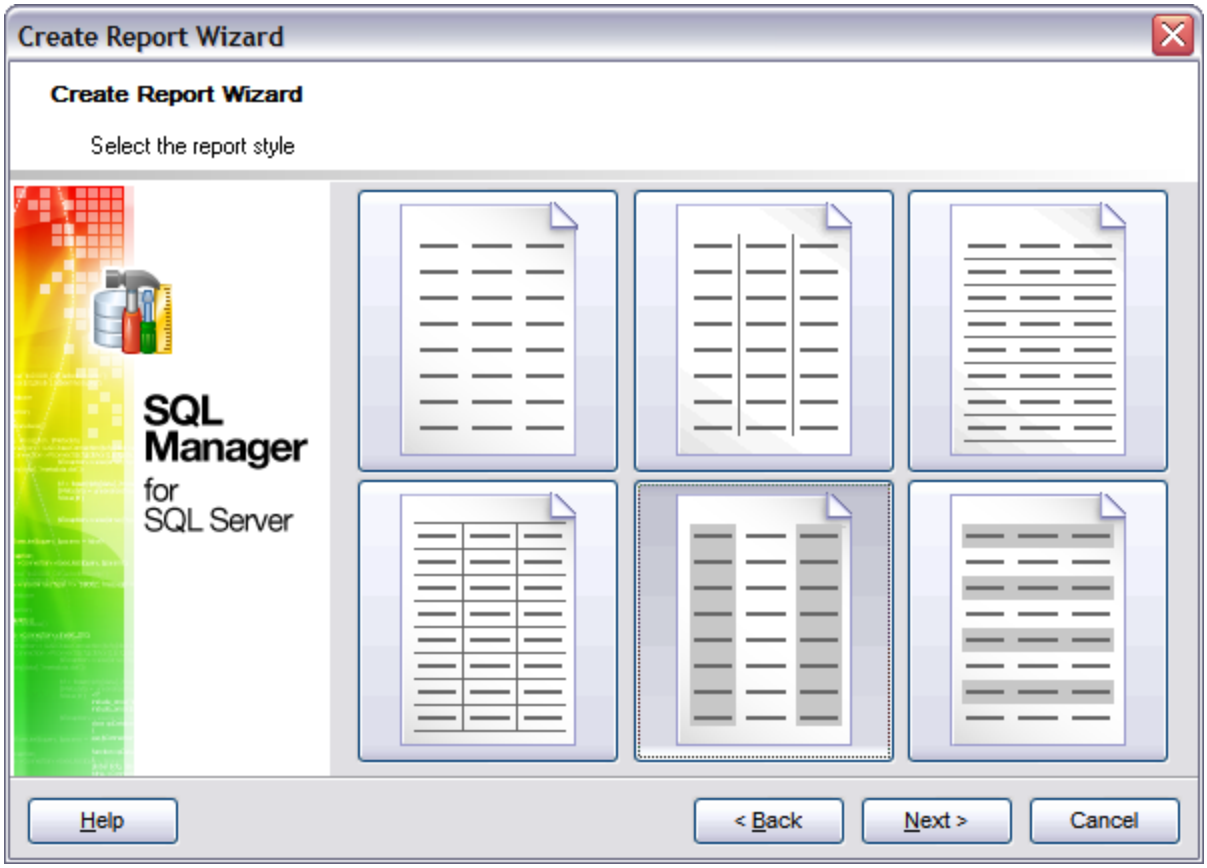
© 2008 EMS Database Management Solutions

Detail data	*Data rows of detail list*
Detail footer	*Prints at the end of detail list*
Subdetail header	*Prints at the beginning of subdetail list*
Subdetail data	*Data rows of subdetail list*
Subdetail footer	*Prints at the end of subdetail list*
Group header	*Prints at the beginning of each group*
Group footer	*Prints at the end of each group*

Click the **Next** button to proceed to the Selecting report style step of the wizard.

9.1.7.1.3 Selecting report style

Select the report style by clicking one of the images illustrating the styled available for the report.



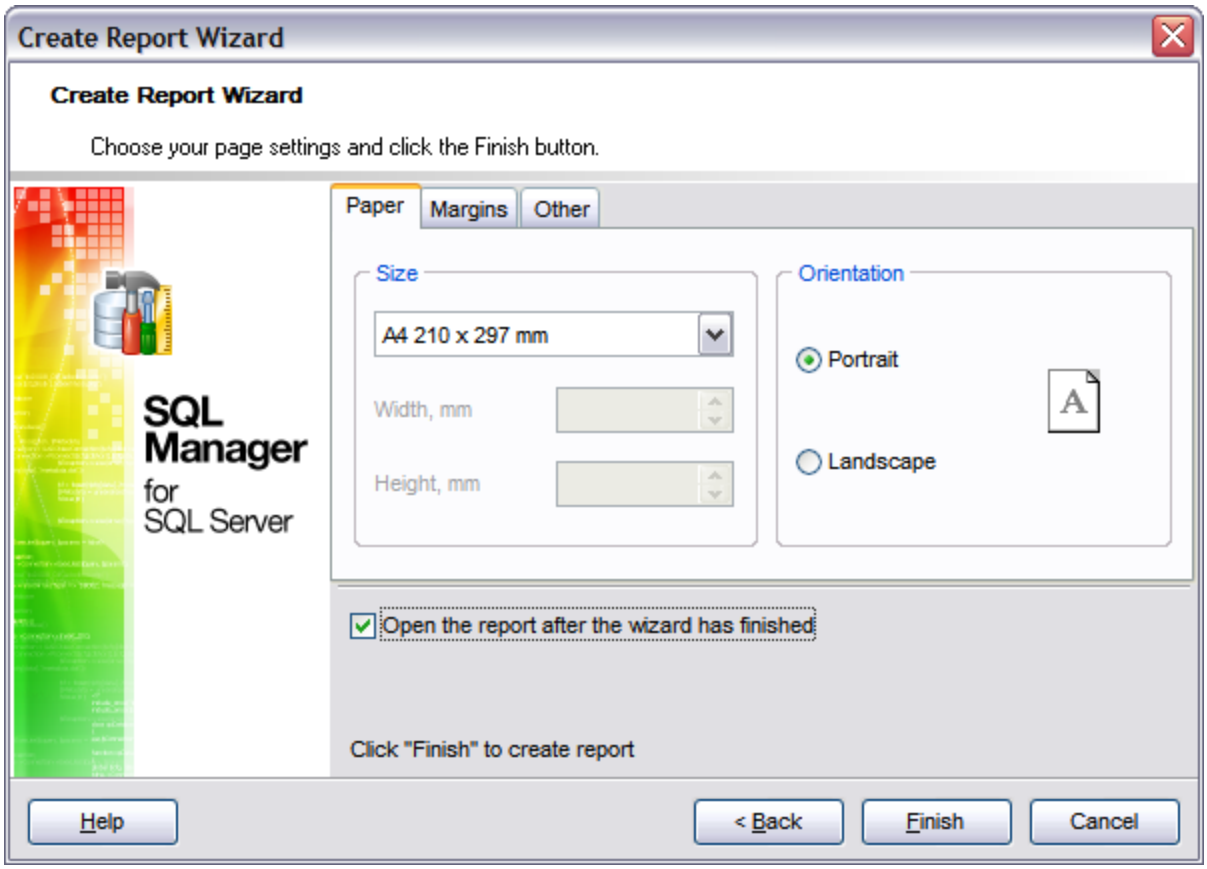
Click the **Next** button to proceed to the Specifying paper settings step of the wizard.

© 2008 EMS Database Management Solutions

9.1.7.1.4 Specifying paper settings

Specify report options: paper *size* and *orientation*, page margins, other settings.

Open the report after the wizard has finished
Check this option to edit the generated report with Report Designer.



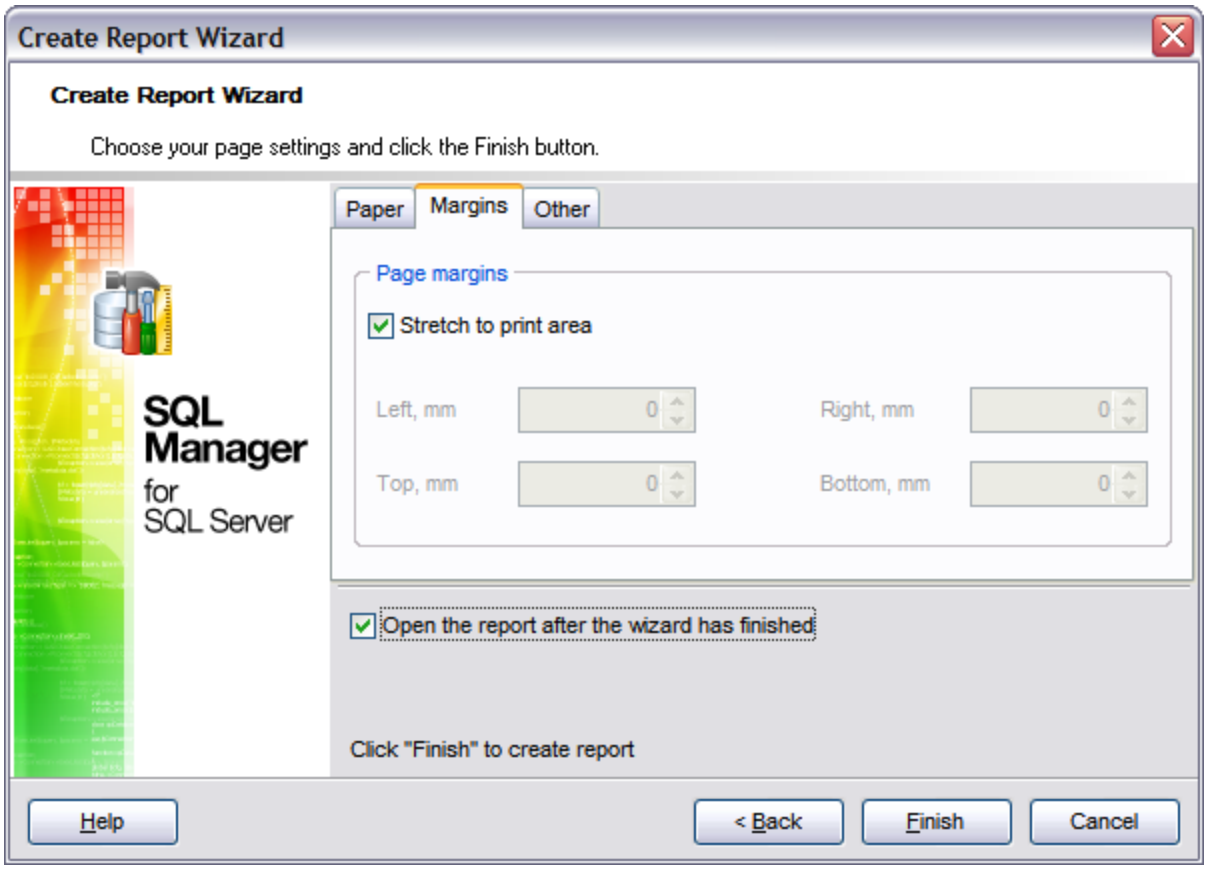
Use the **Margins** tab to specify margins for the result report.

9.1.7.1.5 Specifying margins

Stretch to print area
If this option is checked, the size of report is adjusted to the print area. If this option is unchecked, you can specify the left, right, top and bottom margins (in millimeters).

Open the report after the wizard has finished
If this option is checked, the report will be opened after generating.

© 2008 EMS Database Management Solutions



Use the **Other** tab to specify other page settings for the result report.

9.1.7.1.6 Specifying other page settings

Options

Print to previous page
This option allows to use white space on a previous page. This option can be used in case when a report template consists of several pages or when printing batch (composite) reports.

Two-pass report
If this option is selected, report's formation will be performed in two steps. During the first pass, a report is formed, and is divided into pages, but the result is not saved anywhere. During the second pass a standard report formation with saving a result in the stream is performed.

Page numbering
This option allows to print a page numbers.

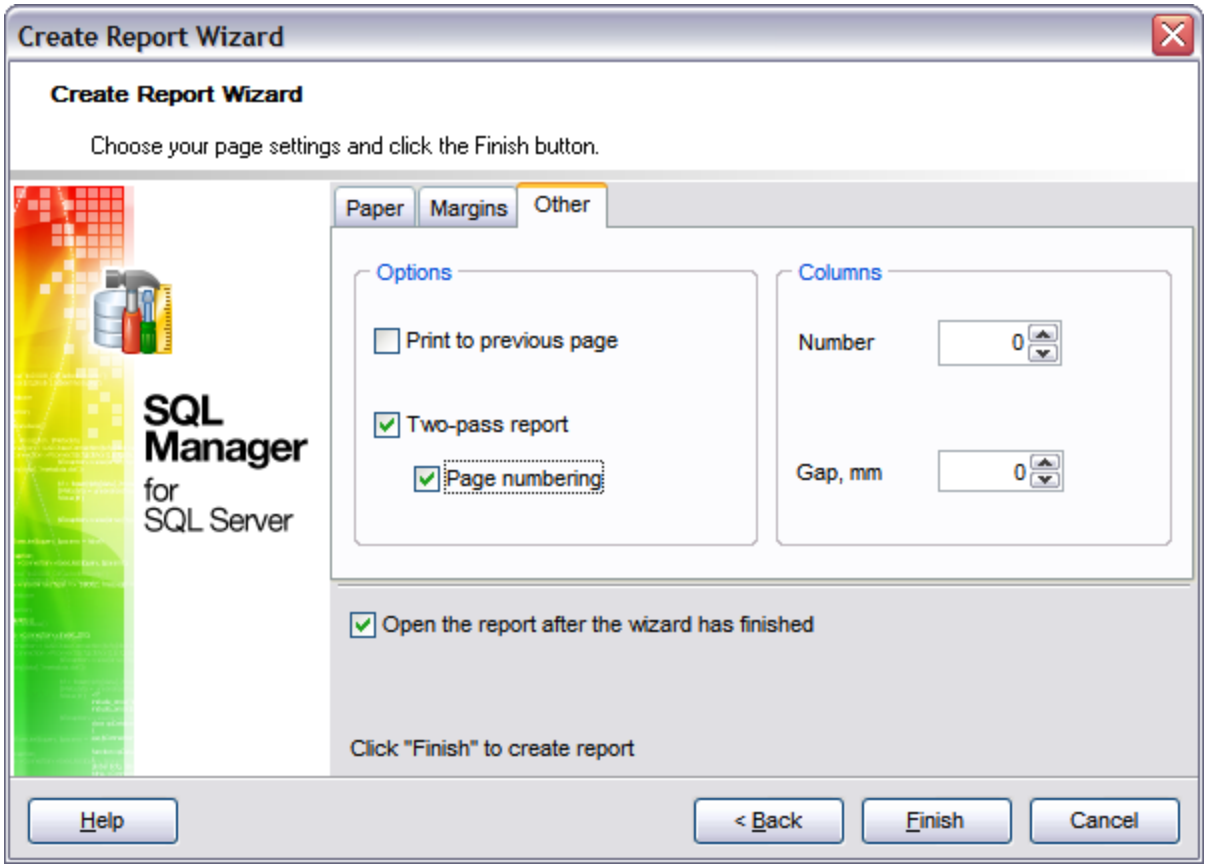
Columns

Number

© 2008 EMS Database Management Solutions

This parameter specifies the number of columns for multi-column reports' printing.

Gap, mm
This parameter specifies the width of the gap between columns.



When you are done, click the **Finish** button to run the report generation process.

9.1.7.2 Report Designer

Report Designer allows you to create and edit reports. This tool can be opened after completion of Create Report Wizard to design a new report.
To edit an already existing project, use the appropriate Navigation bar item of Report Viewer.

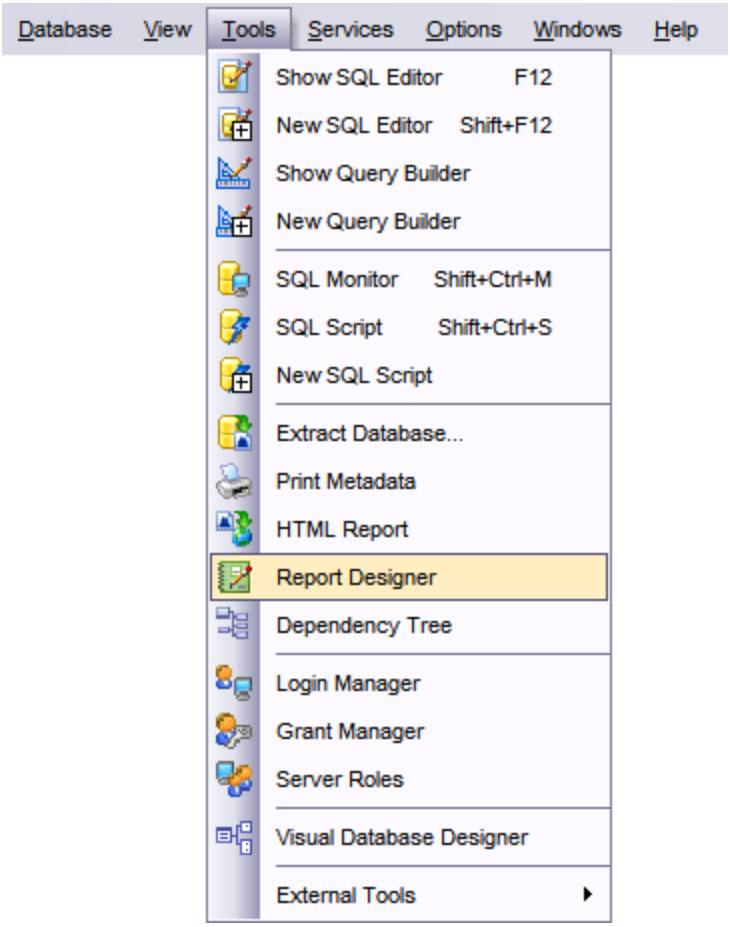
This module is provided by FastReport (http://www.fast-report.com) and has its own help system. Press **F1** key in the **Report Designer** to call the **FastReport** help.

Please find the instructions on how to create a simple report in the **Report Designer** below:

- Adding dialog form
- Adding database and query components
- Adding report data
- Viewing the report

© 2008 EMS Database Management Solutions

- Saving the report



Note: The **Object Inspector** that allows you to edit report object properties, can be opened by pressing the **F11** key.

Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**No**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

Create Report Wizard

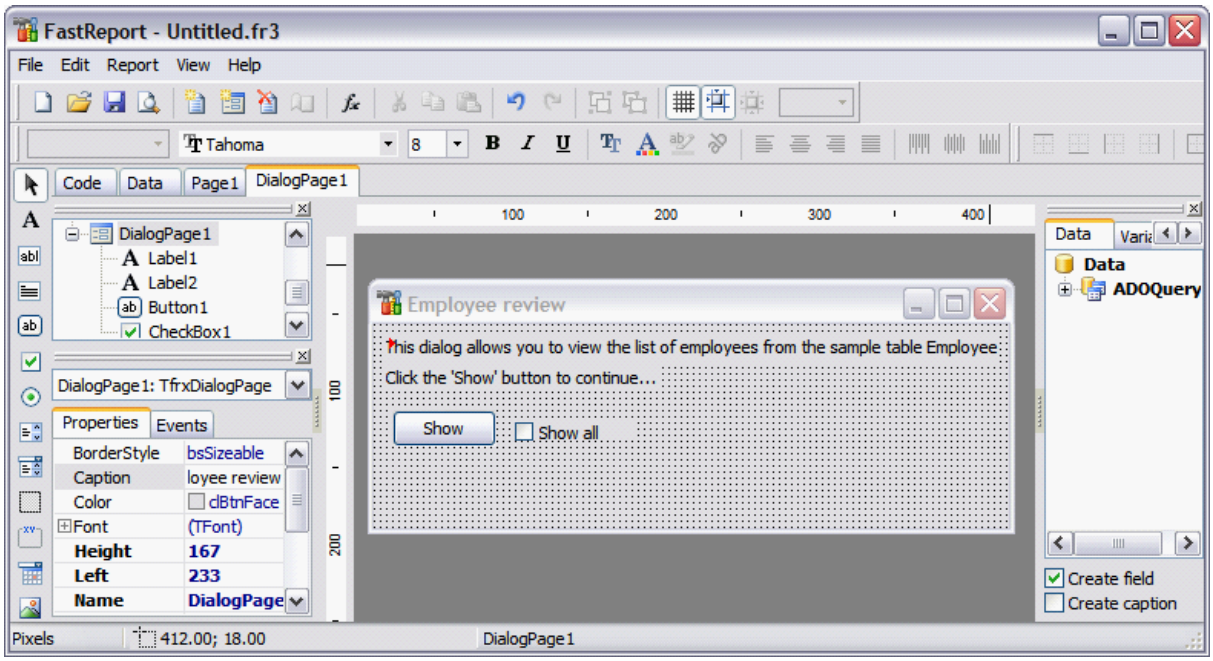
Report Viewer

© 2008 EMS Database Management Solutions

9.1.7.2.1 Adding dialog form

To add a dialog form, select the **File | New Dialog** main menu item in **Report Designer**.

The new dialog appears within the *DialogPage1* tab of the designer. Use the available RAD tools to add necessary interface elements to the dialog.



To call the dialog, proceed to the **Code** tab and supply the corresponding statement (*PascalScript*), e.g.

```
begin
DialogPage1.ShowModal;
end.
```

Using the **Language** drop-down list you can select the script language to be used for the event handler: *PascalScript* (by default), *C++Script*, *BasicScript*, *JScript*.

For instance, the following C++ Script code can be used as the handler for the *OnClick* event of the 'Show' button to open ADOQuery:

```
{
 ADOQuery1.Active = true;
}
```

See also:

Adding database and query components
Adding report data
Viewing the report
Saving the report

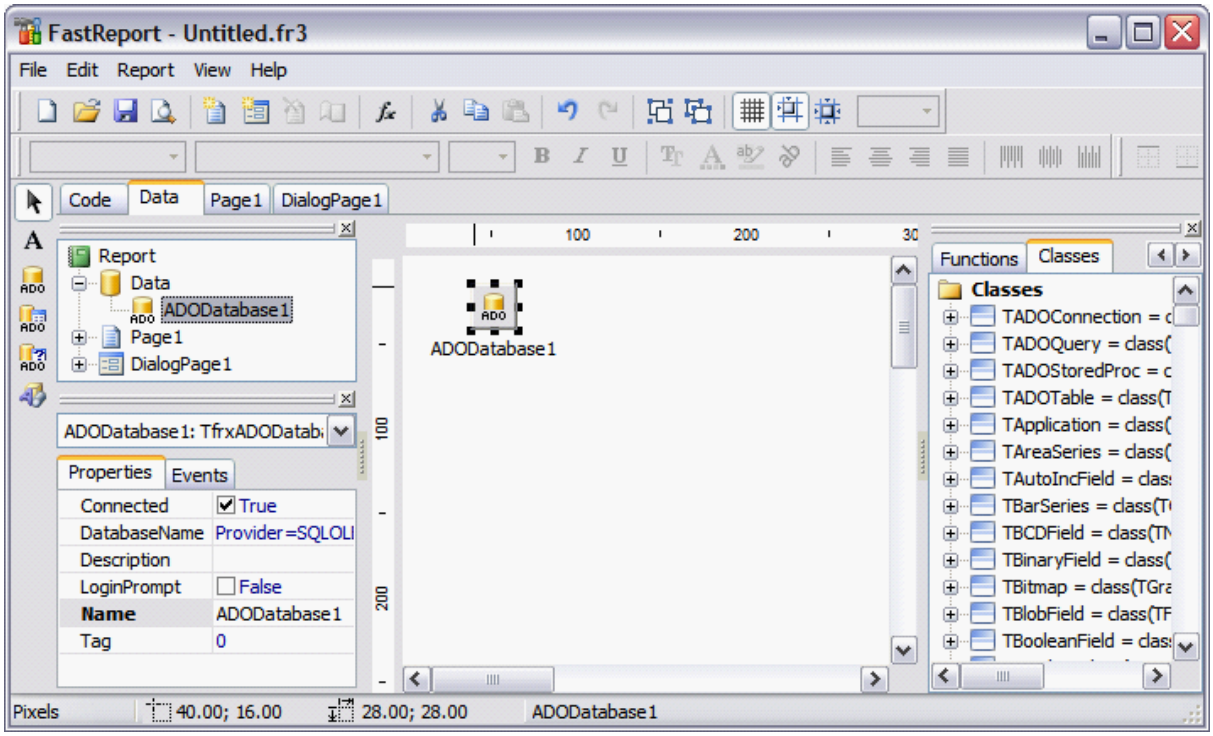
© 2008 EMS Database Management Solutions

9.1.7.2.2 Adding database and query components

Adding database component

In order to add the *Database* component:

- proceed to the **Data** tab of **Report Designer**;
- pick the **ADO Database** component on the toolbar (on the left);
- click within the working area - the corresponding *ADODatabase1* icon appears in the area;
- set the database connection and authorization parameters using the **Properties Inspector**.

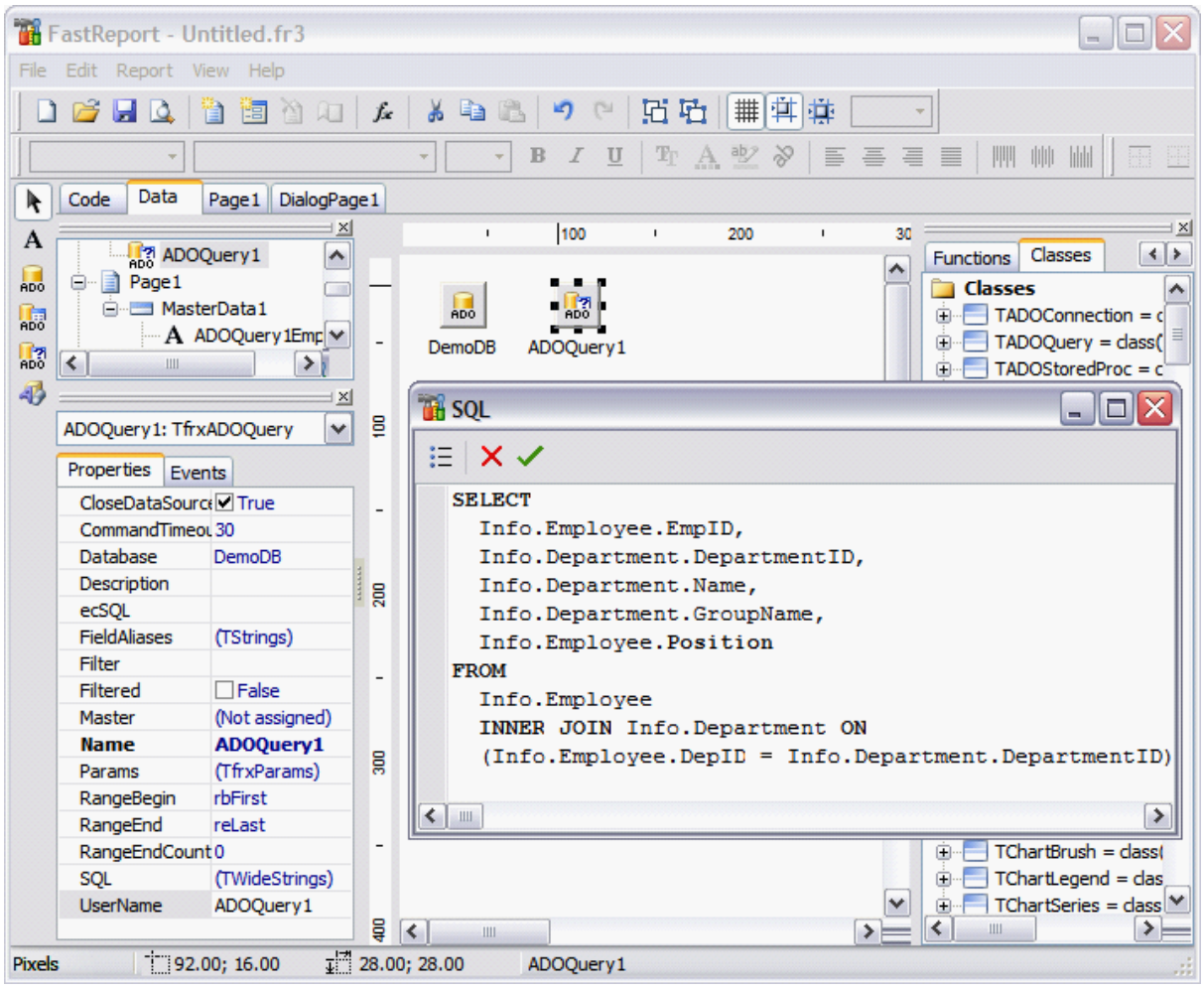


Adding query component

In order to add the *Query* component:

- proceed to the **Data** tab of **Report Designer**;
- pick the **ADO Query** component on the toolbar (on the left);
- click within the working area - the corresponding *ADOQuery1* icon appears in the area;
- set the database name and authorization parameters within the **Properties Inspector**;
- double-click the *ADOQuery1* icon to open the **SQL** window;
- input the SQL query that returns the required dataset and click the button;
- repeat the operation if you wish to add other query components to the report.

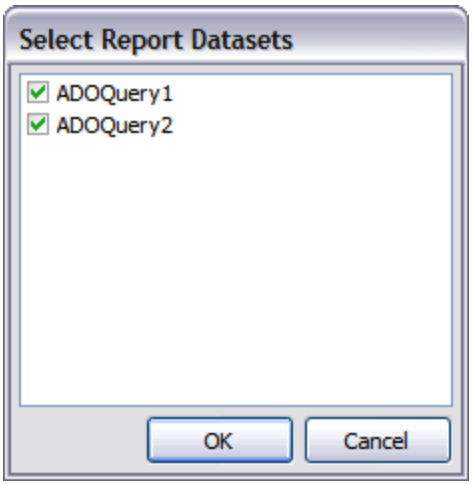
© 2008 EMS Database Management Solutions



Note: The **Properties Inspector** panel that allows you to edit report object properties can be shown/hidden by pressing the **F11** key.

Using the above given steps you can create as many queries as you need. In order to select a dataset returned by a query, select the **Report | Data...** main menu item of **Report Designer** to call the **Select Report Datasets** dialog. Pick the required query within the dialog and press **OK**.

© 2008 EMS Database Management Solutions



See also:
Adding dialog form
Adding report data
Viewing the report
Saving the report

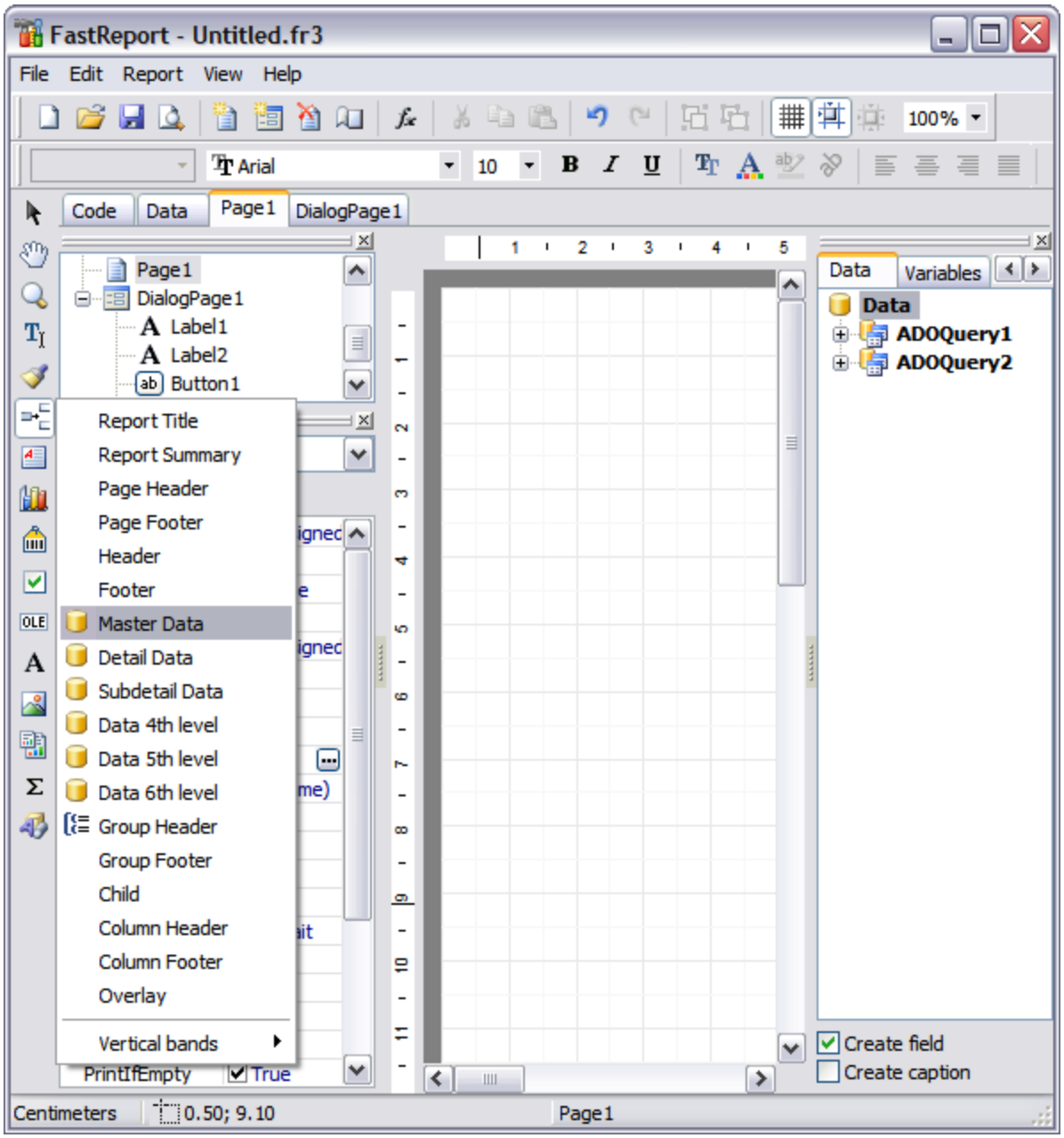
9.1.7.2.3 Adding report data

Adding bands

In order to add a band to the report:
- proceed to the **Page1** tab of **Report Designer**;
- pick the **Insert Band** component on the toolbar (on the left);
- select the band to be added to the report;
- click within the working area - the corresponding element appears in the area;
- set element properties within the **Properties Inspector**.

Note: The **Properties Inspector** panel that allows you to edit report object properties can be shown/hidden by pressing the **F11** key.

© 2008 EMS Database Management Solutions

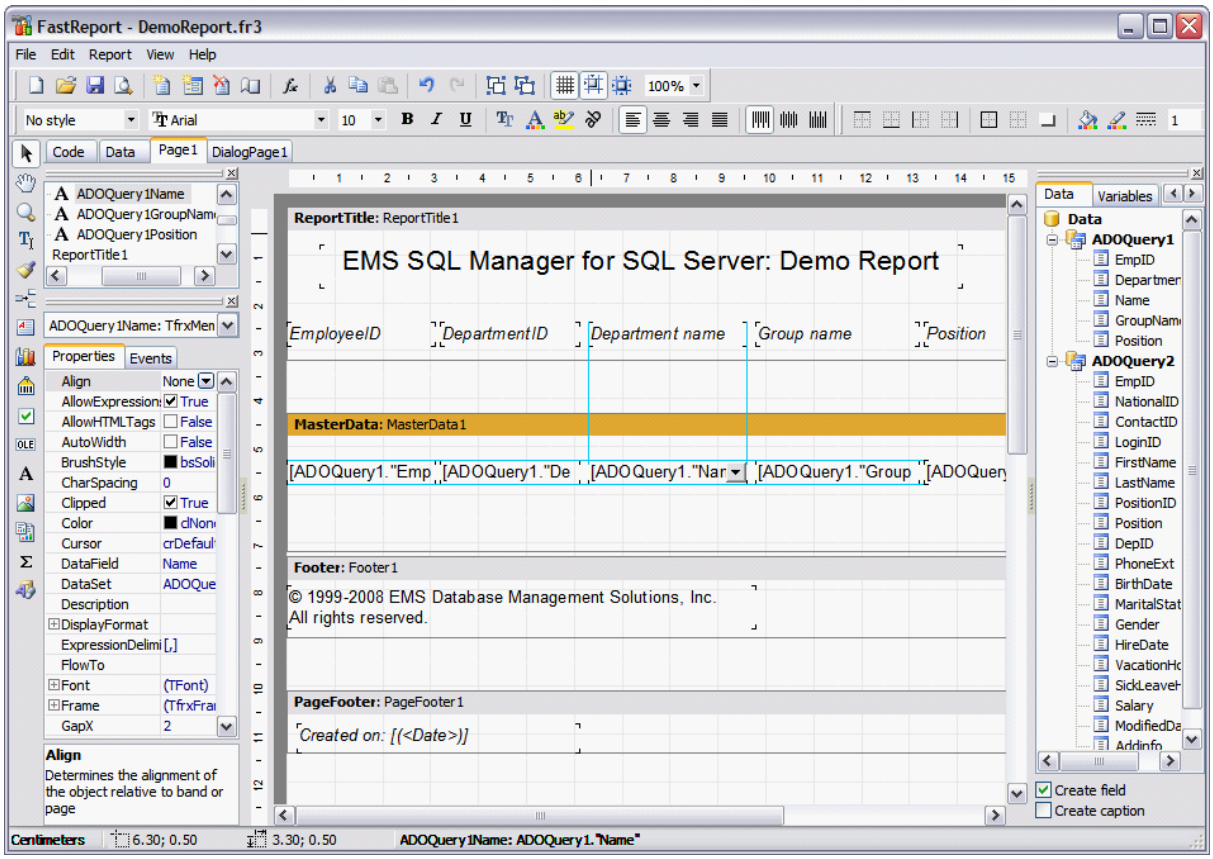


Adding report data

In order to add data to the report:

- proceed to the **Data** tab within the panel on the right side of the window;
- pick a field within the **Data** tree and drag it to the working area;
- add all necessary elements one by one using drag-and-drop operation for each of them.

© 2008 EMS Database Management Solutions



See also:

Adding dialog form
Adding database and query components
Viewing the report
Saving the report

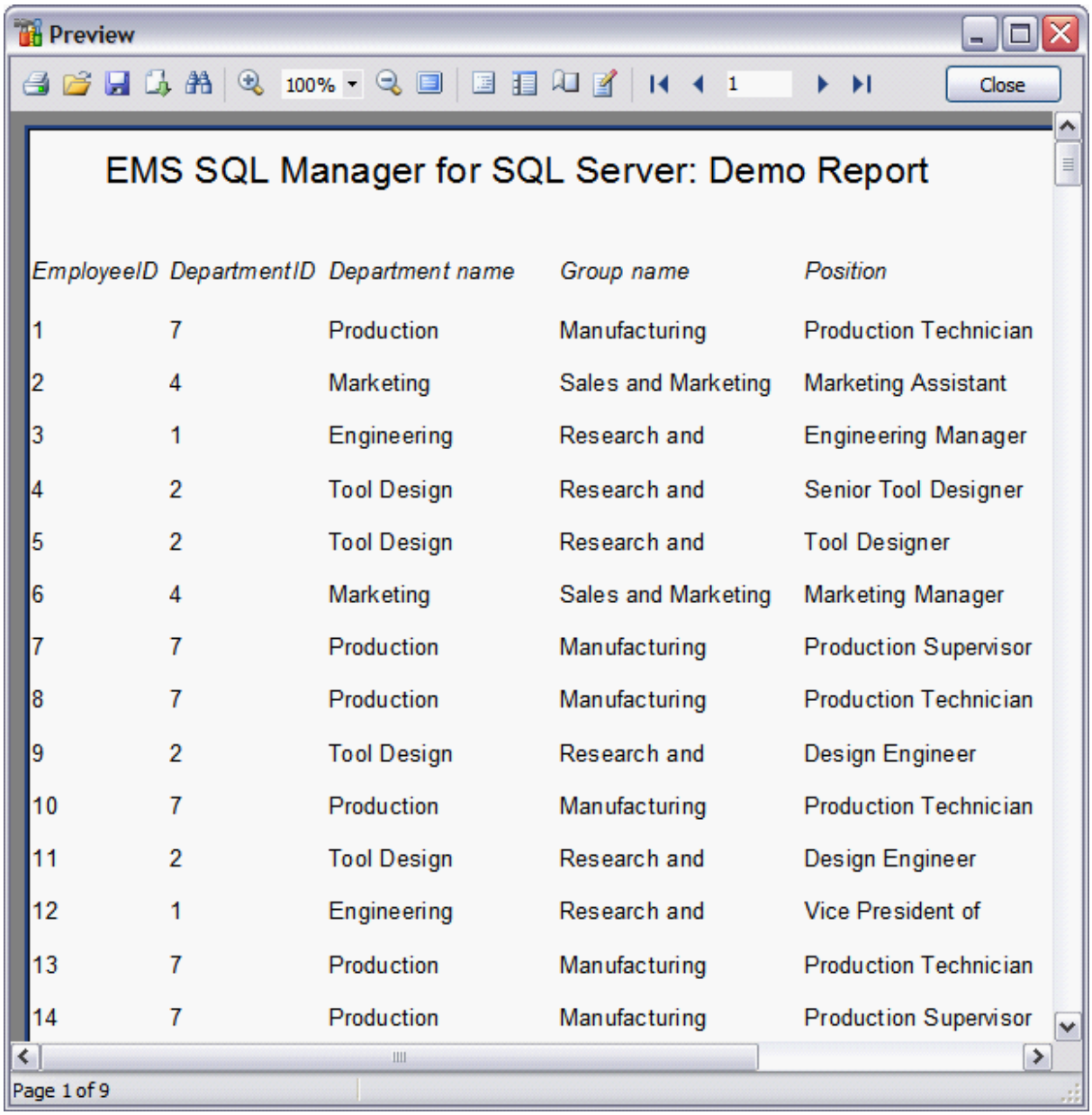
9.1.7.2.4 Viewing the report

Viewing the report

To preview the newly created report, select the **File | Preview** main menu item or use the corresponding **Preview** toolbar button. You can also use the *Ctrl+P* shortcut for the same purpose. This mode allows you to view, edit and print the result report.

To print the report, use the **Print** toolbar button or the corresponding context menu item.

© 2008 EMS Database Management Solutions



It is also possible to preview/print the report using Report Viewer.

See also:

Adding dialog form
Adding database and query components
Adding report data
Saving the report

© 2008 EMS Database Management Solutions

9.1.7.2.5 Saving the report

When all report parameters are set, you can save the report to an external *.fr3 file on your local machine or in a machine in the LAN.

To save the report, select the **File | Save** main menu item or use the corresponding **Save Report** toolbar button. You can also use the *Ctrl+S* shortcut for the same purpose.

If necessary, you can add the report to the database using Create Report Wizard and perform preview/print operations using Report Viewer.

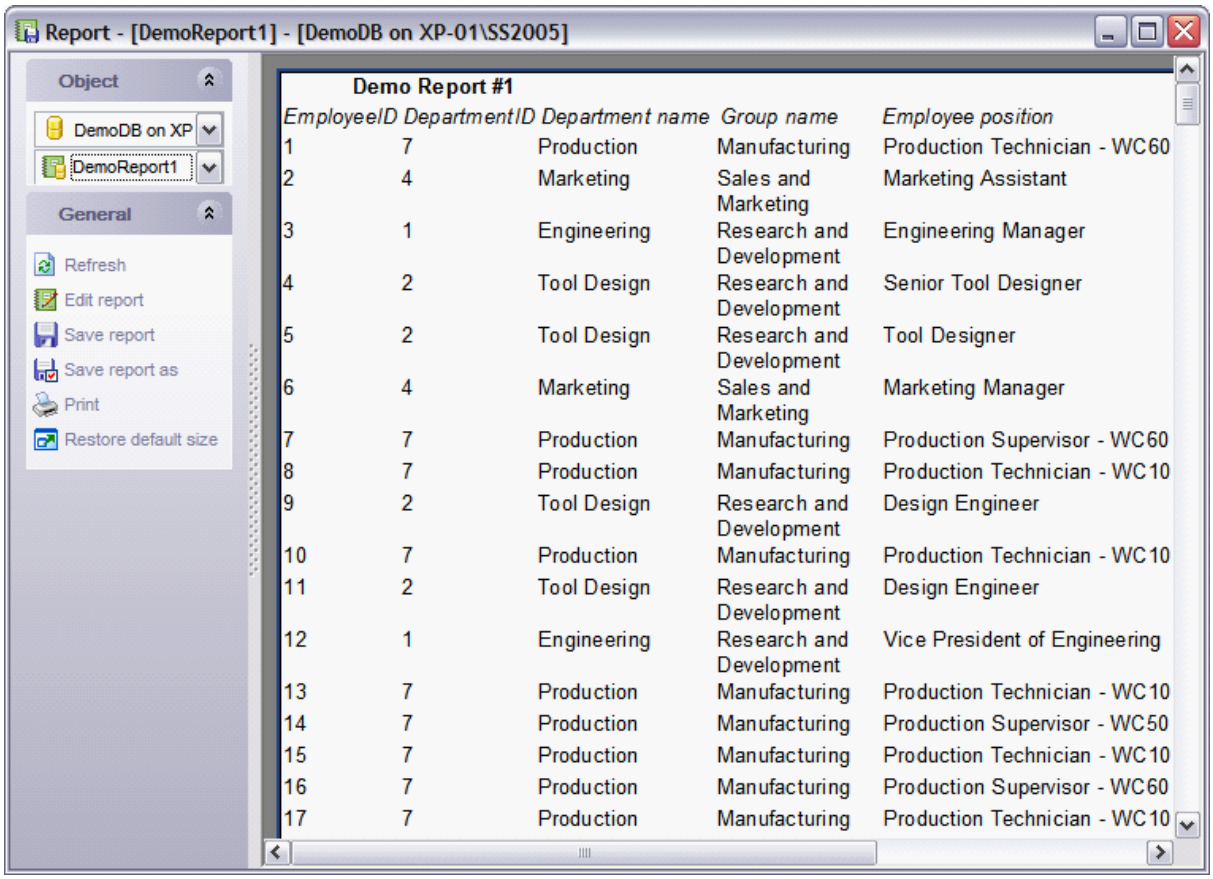
See also:
Adding dialog form
Adding database and query components
Adding report data
Viewing the report

9.1.7.3 Report Viewer

Using **Report Viewer** you can view, edit, save and print reports.

The main report operations can be found at the Using Navigation bar page.

© 2008 EMS Database Management Solutions



Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**No**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

Create Report Wizard

Report Designer

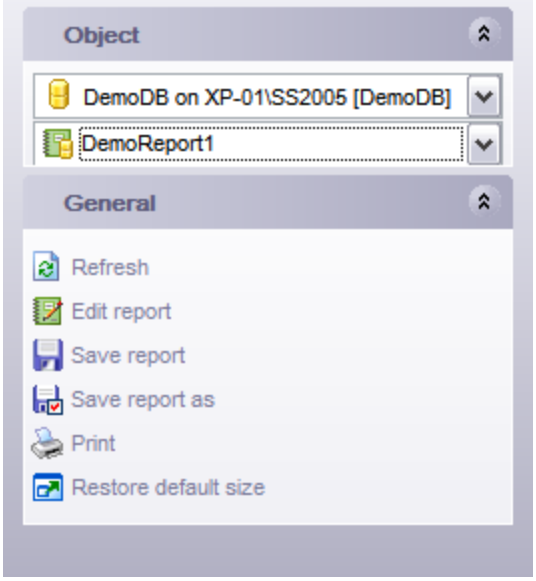
9.1.7.3.1 Using Navigation bar

The **Navigation bar** of **Report Viewer** allows you to:

- refresh the content of the window;
- edit report using Report Designer;
- save the current report;
- save the report to a **.fr3* file using the **Save as...** dialog;

© 2008 EMS Database Management Solutions

- print the report;
- restore the default size of the window.



See also:

Report Viewer

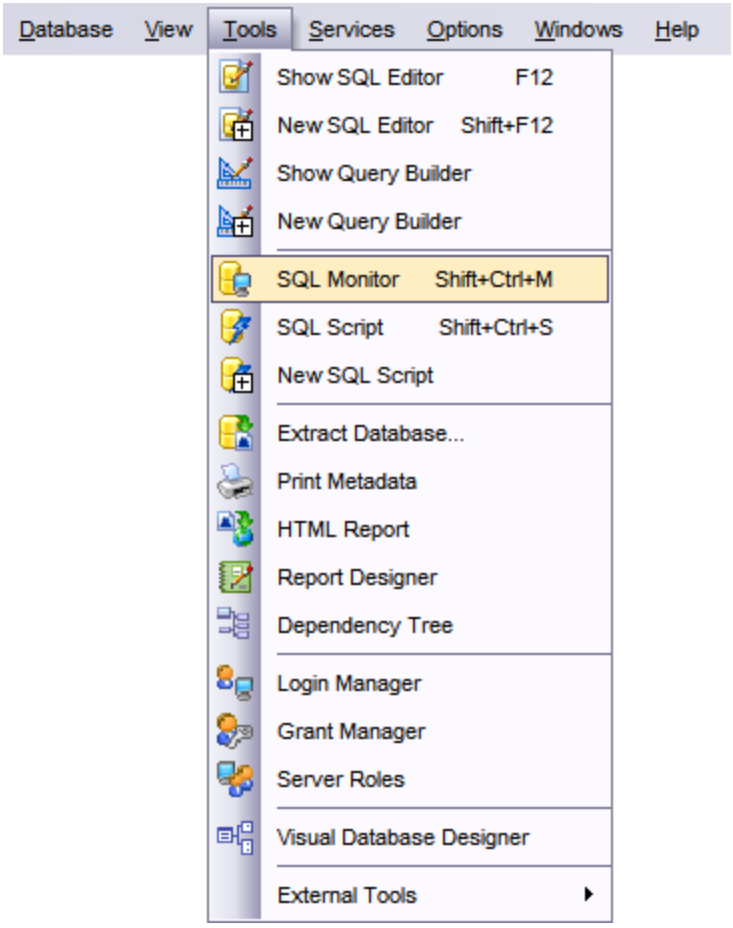
9.1.8 SQL Monitor

SQL Monitor allows you to view the log of all operations performed over databases and database objects in SQL Manager 2008 for SQL Server. The content of the window is read-only.

To open the **SQL Monitor** window, select the **Tools | SQL Monitor** main menu item, or use the *Shift+Ctrl+M* shortcut.

- Using Navigation bar
- Working with SQL Monitor

© 2008 EMS Database Management Solutions



Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**Yes**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

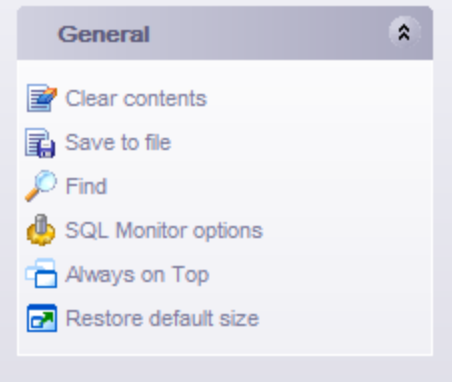
Dependency Tree
Visual Database Designer
SQL Script Editor
Extract Database Wizard
Print Metadata
HTML Report Wizard
Reports management

© 2008 EMS Database Management Solutions

9.1.8.1 Using Navigation bar

The **Navigation bar** of **SQL Monitor** allows you to:

- clear the content of the window;
- save the content to a **.txt* file using the **Save as...** dialog;
- search for a string using the Find Text dialog;
- configure SQL Monitor using the SQL Monitor section of the Environment Options dialog;
- always show the window on top;
- restore the default size of the window.



See also:

Working with SQL Monitor

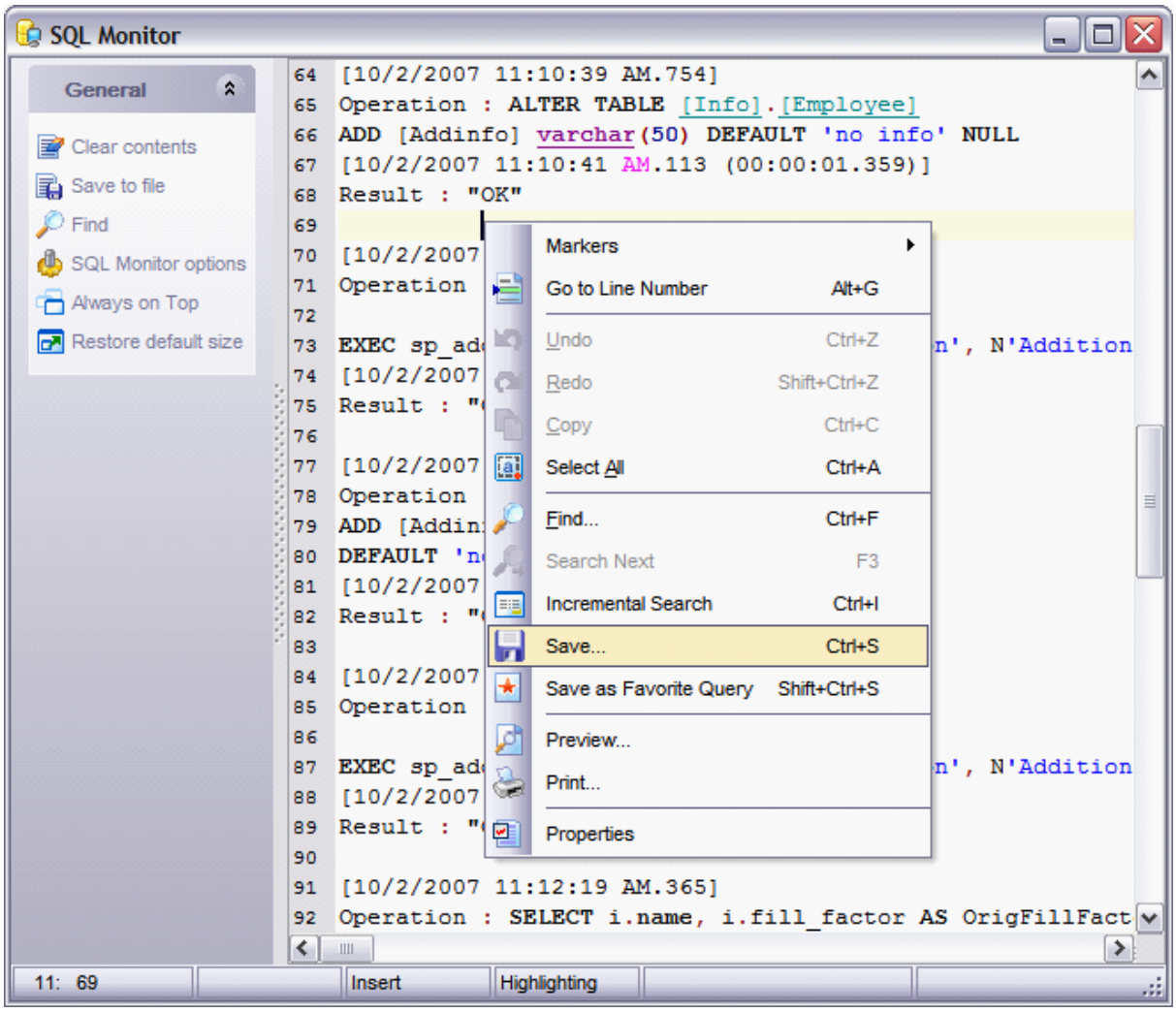
9.1.8.2 Working with SQL Monitor

The working area of **SQL Monitor** lists the log of database operations and SQL queries as items, each consisting of 3 parts: *Executed* (the date and time of the operation), *Operation* (SQL statement), *Result* (the result of the operation).

Items of the **context menu** of SQL Monitor area provide access to various functions for working with the window content. The context menu contains standard text-processing functions (*Copy*, *Select All*, *Undo*, *Redo*) and functions for working with the content as a whole, e.g. you can use *bookmarks* for quick navigation within the text, *move the cursor to a particular line*, configure the tool using the *properties* item or *print* the content. Most of these operations can be also performed with the corresponding hot keys used.

Implementation of the Find Text dialog contributes to more efficient work with the content of SQL Monitor.

© 2008 EMS Database Management Solutions



See also:

Using Navigation bar

© 2008 EMS Database Management Solutions

9.2 Specific DB Tools

SQL Manager 2008 for SQL Server provides graphical interface for all database maintenance operations. The following *specific database tools* are available in SQL Manager:

- **Attach Database Wizard**
  Allows you to attach an existing database to the instance of Microsoft® SQL Server™.
- **Detach Database Wizard**
  Allows you to detach a database from the instance of Microsoft® SQL Server™.
- **Backup Database Wizard**
  Creates backup copies of entire databases, transaction logs and files/filegroups.
- **Restore Database Wizard**
  Restores entire databases, transaction logs and files/filegroups from previously created backups.
- **Shrink Database Wizard**
  Reduces data and transaction log files.
- **Indices Management Wizard**
  Allows you to reorganize, rebuild, and disable indexes.
- **Statistics Update Wizard**
  Updates database statistics.
- **Check Database Wizard**
  Checks the allocation, structural, and logical integrity of all database objects.
- **Grant Manager**
  Allows you to grant/revoke privileges on database objects.
- **Using templates**
  Facilitates using SQL Manager wizards.

To obtain detailed information concerning specific SQL Server database maintenance services, refer to the official Microsoft® SQL Server™ documentation.

See also:
Common database tools

9.2.1 Attach Database

Attach Database Wizard allows you to attach an existing database that is made up of one or more files to the instance of Microsoft® SQL Server™ (with SQL Server routines used).

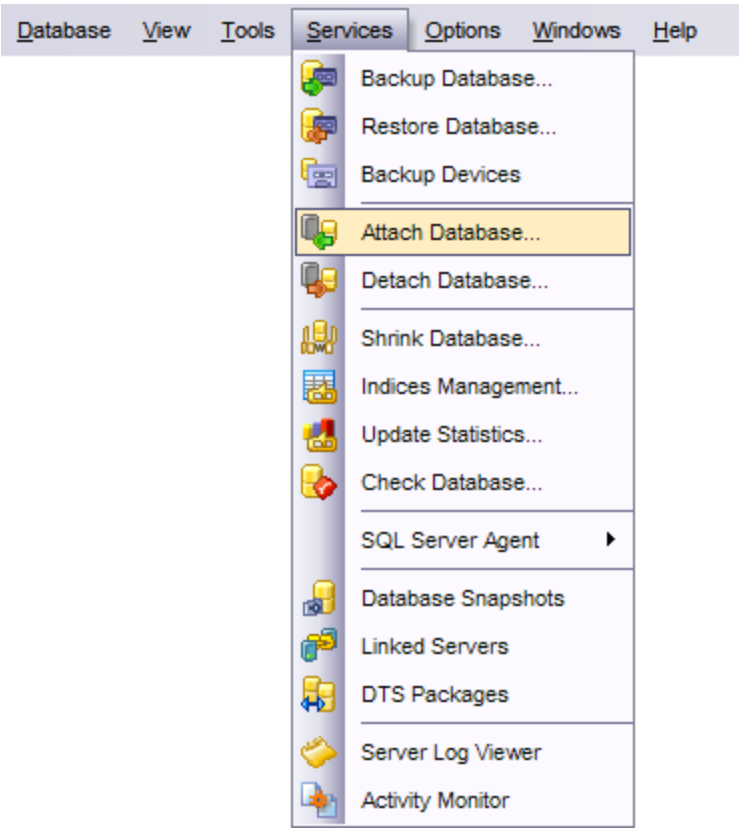
This operation is used to attach a database with the specified name and using the specified data files.

To run the wizard, select the **Services | Attach Database...** main menu item, or right-click the **Databases** node in the DB Explorer tree and select the **Tasks | Attach Database...** context menu item.

- Setting connection properties
- Defining database files and general options

© 2008 EMS Database Management Solutions

- Attaching database
- Using templates



Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**Yes**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

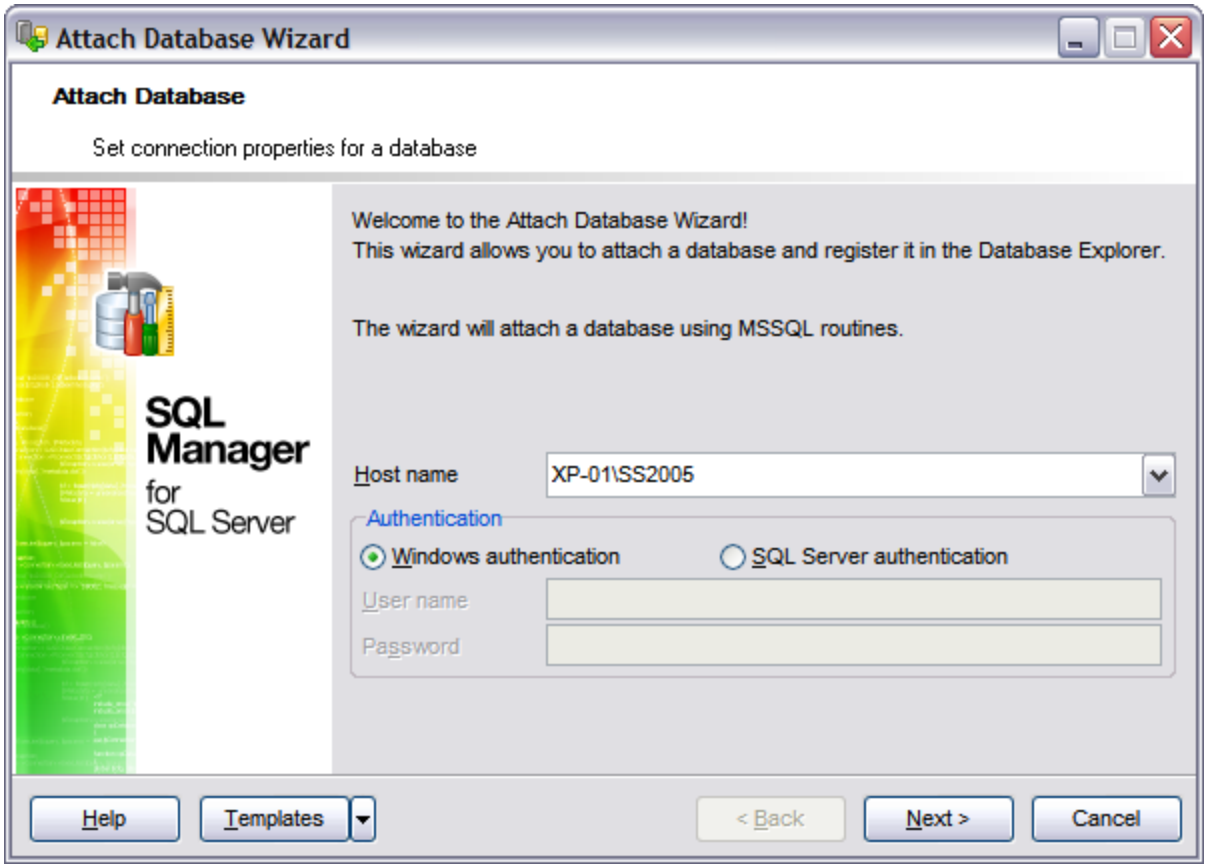
See also:

Detach Database
Backup Database
Restore Database
Shrink Database
Indices Management
Update Statistics
Check Database
Grant Manager

© 2008 EMS Database Management Solutions

9.2.1.1 Setting connection properties

This step of the wizard allows you to specify the **host/instance** to attach database to, and to provide **authentication** parameters.



Host
Type in or use the drop-down list to select the host/instance name to attach database to.

Authentication
Specify the type of Microsoft® SQL Server™ authentication to be used for the connection: the built-in **Windows authentication** or **SQL Server authentication**.

If 'SQL Server' has been selected as the *authentication type*, you should also provide *authorization* settings: **User name** and **Password**.

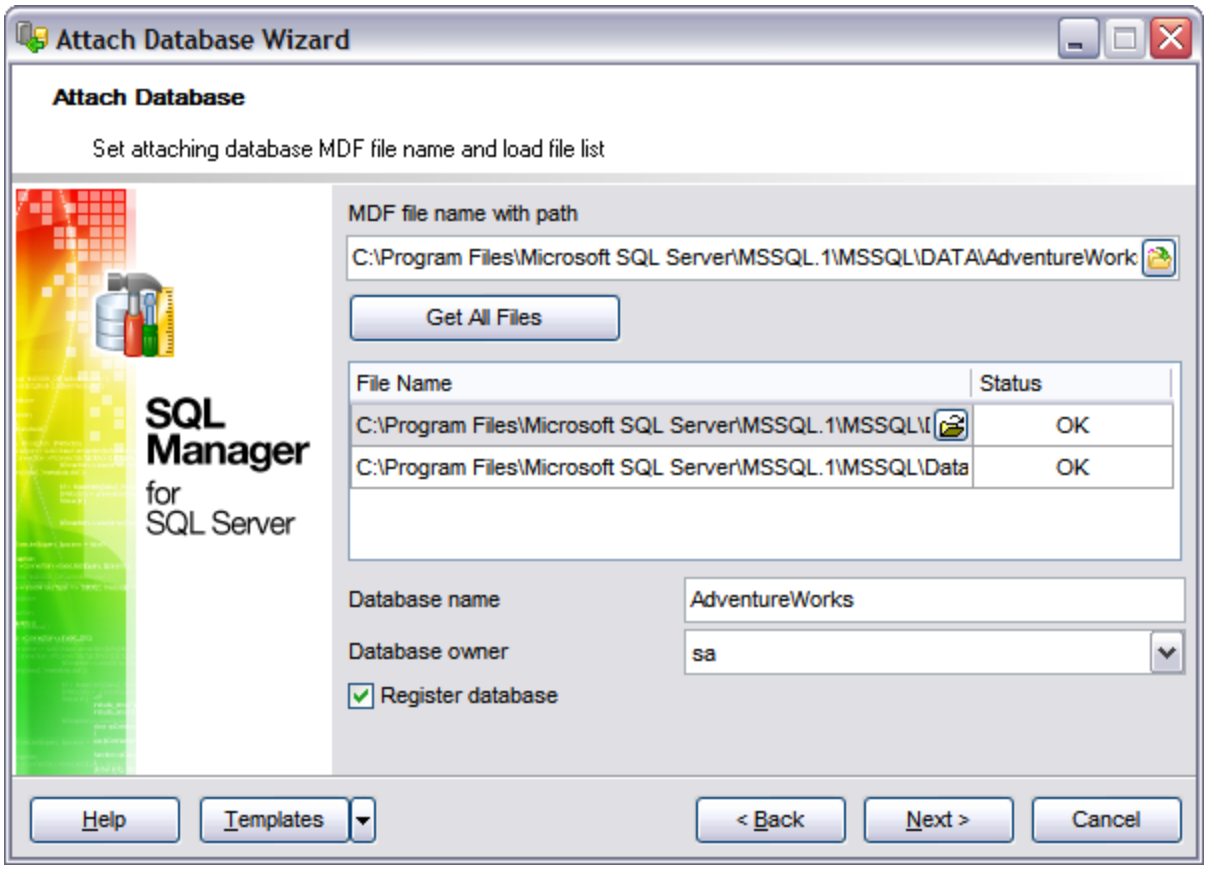
Click the **Next** button to proceed to the Defining database files and general options step of the wizard.

© 2008 EMS Database Management Solutions

9.2.1.2 Defining database files and general options

Use this step of the wizard to **select the files** to attach to the server as a database.

MDF file name with path
Use the button to specify the path and name of the file to attach to the server. Then press the **Get All Files** button to add the database files to the file list.



Database name
Enter the name of the database to be attached.

Database owner
Use the drop-down list to select the database owner.

Register Database
This option indicates that the Database Registration Info dialog for the new database will appear immediately after attaching the database (you need to register the database to start working with it in SQL Manager). If you intend to register the database later, uncheck this option.

When you are done, click the **Next** button to proceed to the Attaching database step of the wizard.

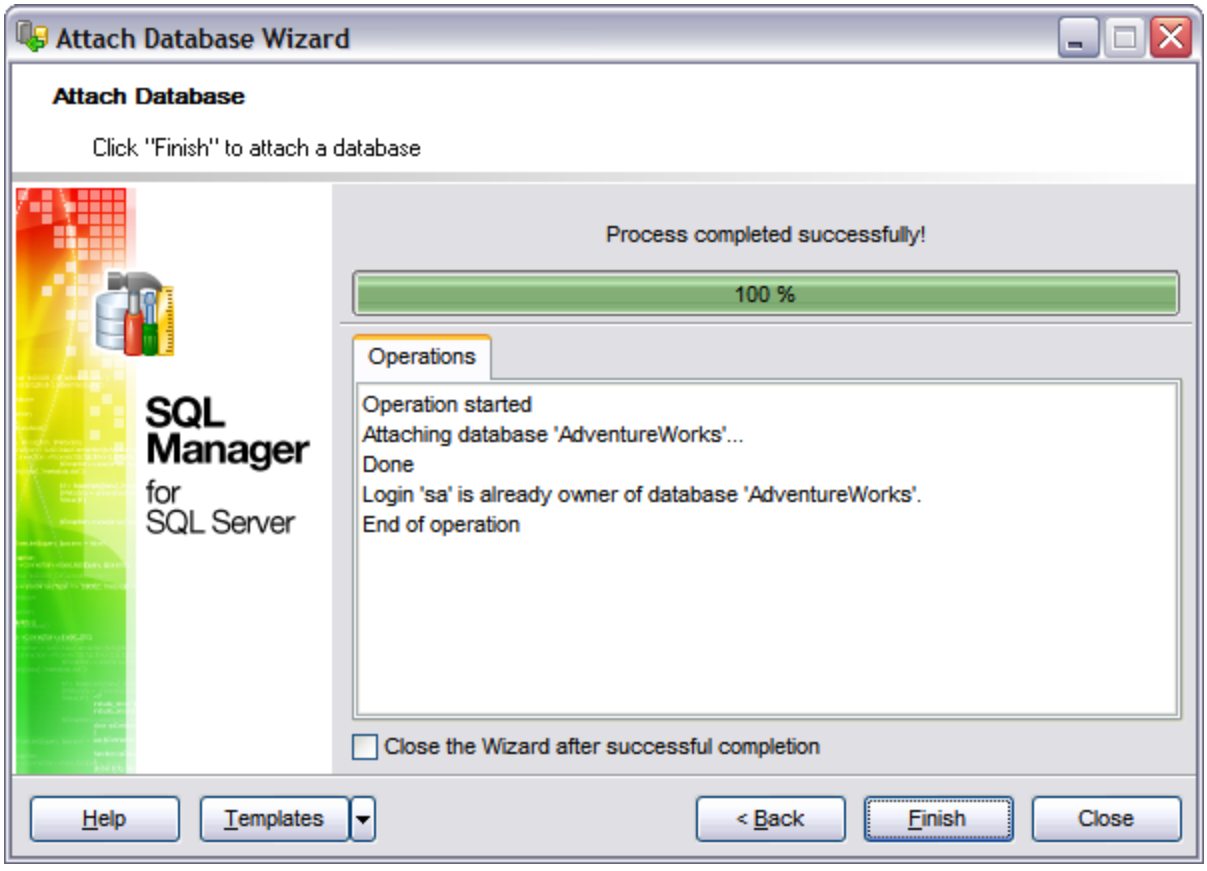
© 2008 EMS Database Management Solutions

9.2.1.3 Attaching database

This step of the wizard is intended to inform you that all necessary options have been set, and you can start the attach database process.

Close the Wizard after successful completion
If this option is selected, the wizard is closed automatically when the process is completed.

The **Operations** tab allows you to view the log of operations and errors (if any).



If necessary, you can save a template for future use.

Click the **Finish** button to complete the operation.

9.2.2 Detach Database

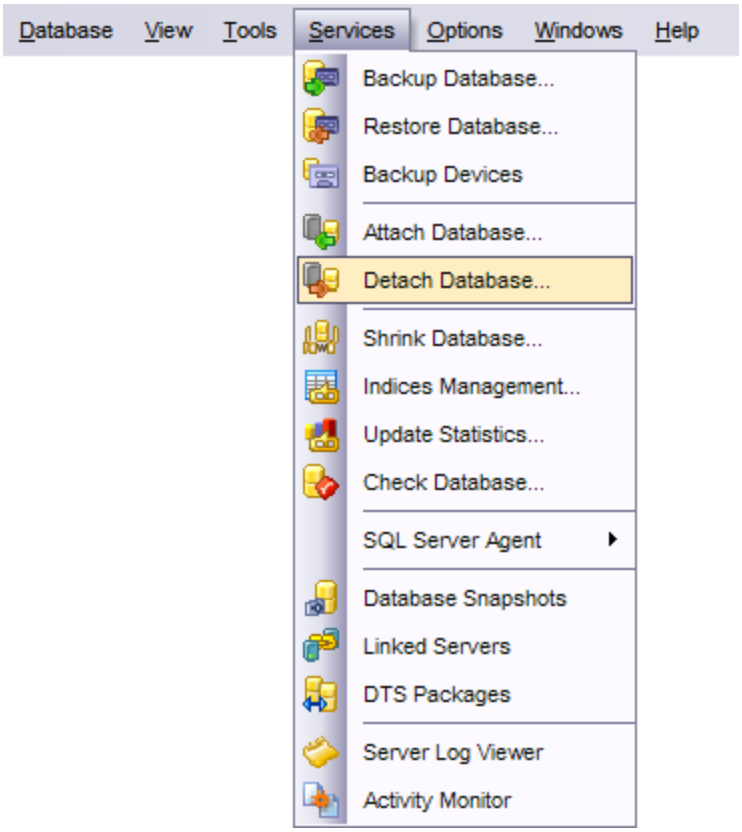
Detach Database Wizard allows you to detach a database from the instance of Microsoft® SQL Server™ (with SQL Server routines used).

© 2008 EMS Database Management Solutions

This operation is used to detach the specified database with or without updating statistics before the database is detached.

To run the wizard, select the **Services | Detach Database...** main menu item, or right-click the database alias in the DB Explorer tree and select the **Tasks | Detach Database...** item of the context menu.

- Setting connection properties
- Selecting the database
- Detaching database
- Using templates



Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**Yes**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

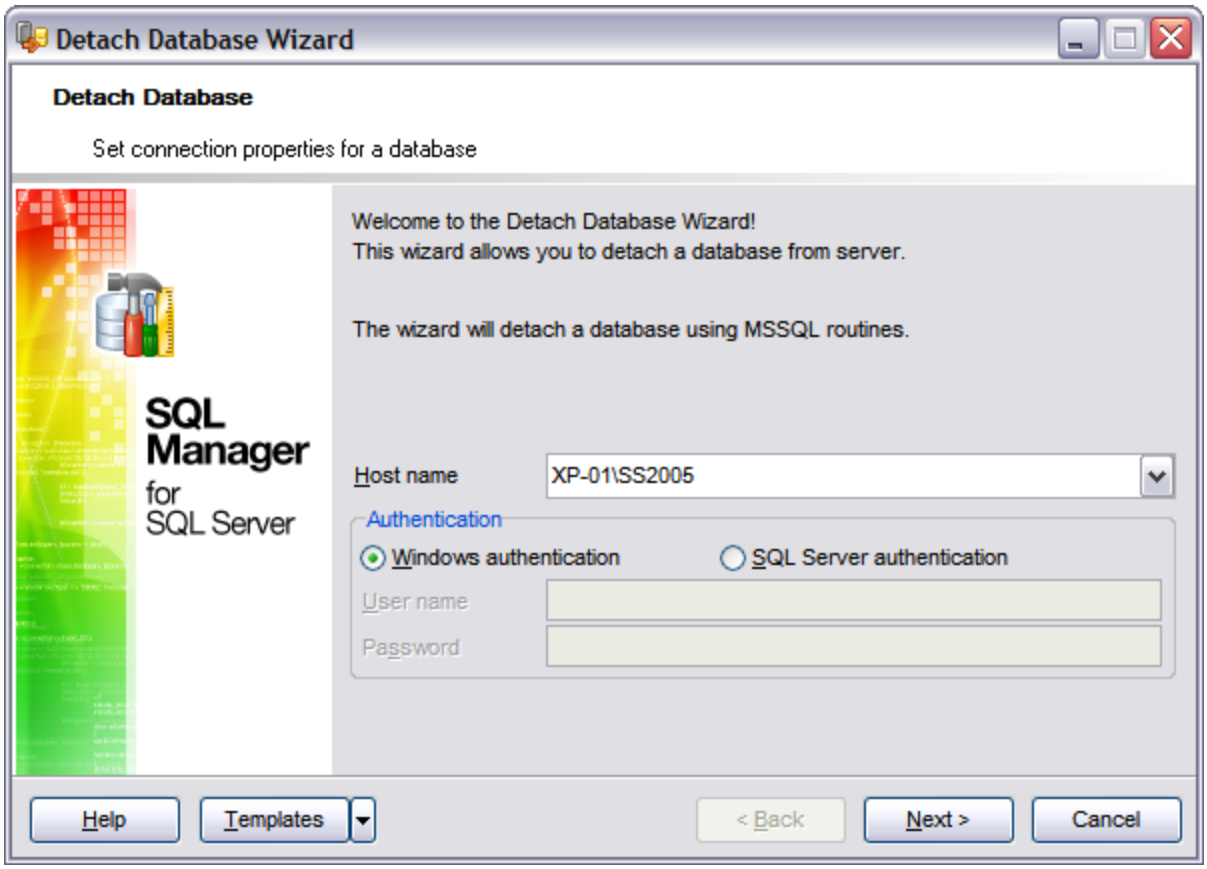
Attach Database

© 2008 EMS Database Management Solutions

Backup Database
Restore Database
Shrink Database
Indices Management
Update Statistics
Check Database
Grant Manager

9.2.2.1 Setting connection properties

This step of the wizard allows you to specify the **host/instance** to detach database from, and to provide **authentication** parameters.



Host
Type in or use the drop-down list to select the host/instance name to detach database from.

Authentication
Specify the type of Microsoft® SQL Server™ authentication to be used for the connection: the built-in **Windows authentication** or **SQL Server authentication**.

© 2008 EMS Database Management Solutions

If 'SQL Server' has been selected as the *authentication type*, you should also provide *authorization* settings: **User name** and **Password**.

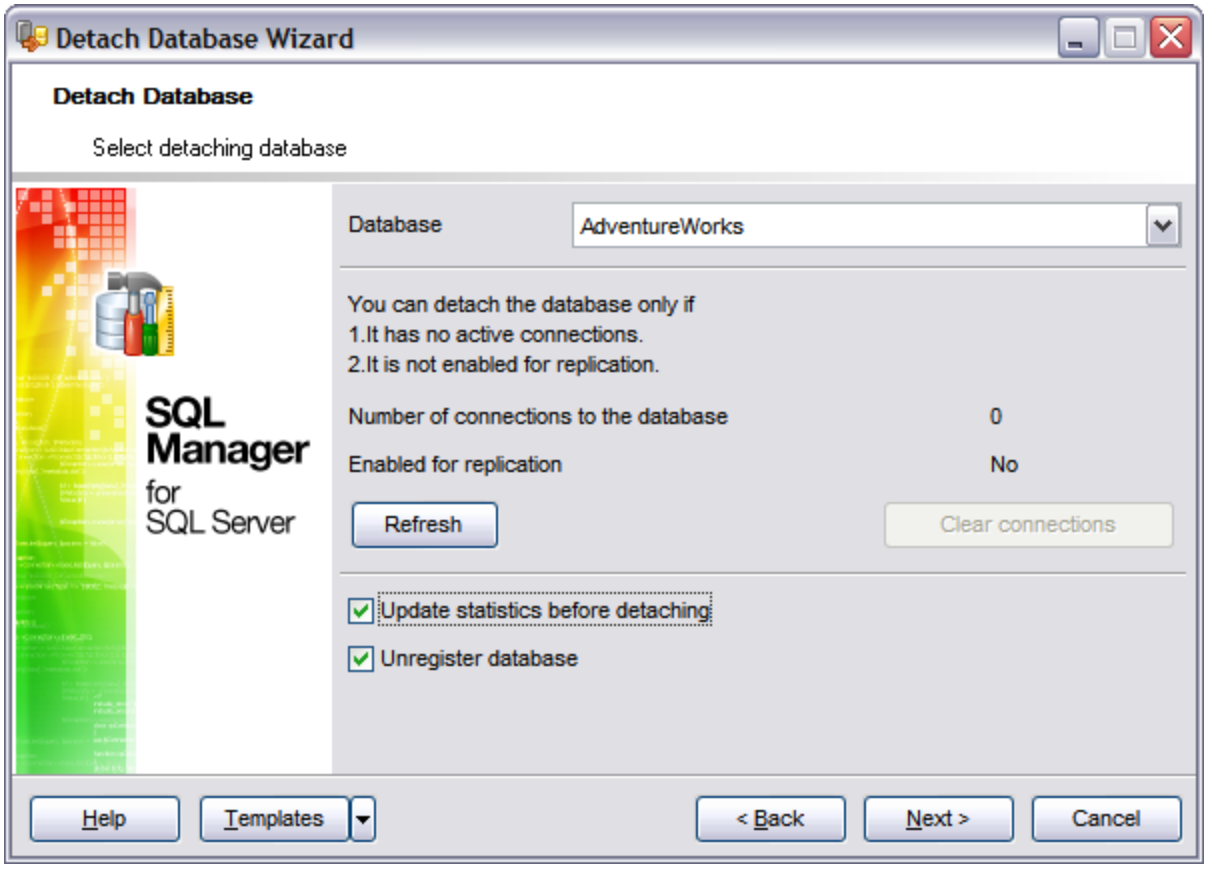
Click the **Next** button to proceed to the Selecting the database step of the wizard.

9.2.2.2 Selecting the database

At this step of the wizard you should **specify the database** to be detached from the instance of SQL Server.

Use the **Database** drop-down list to select the database to be detached.
Note that you can detach a database only if it has no active connections and it is not enabled for replication. To clear the current connections to the database, use the corresponding **Clear connections** button.

Press the **Refresh** button to get the current status of the selected database.



Update statistics before detaching
Check this option to update the database statistics before detaching the database.

Unregister database
Using this option you can unregister the database (if it has been registered in SQL

© 2008 EMS Database Management Solutions

Manager).

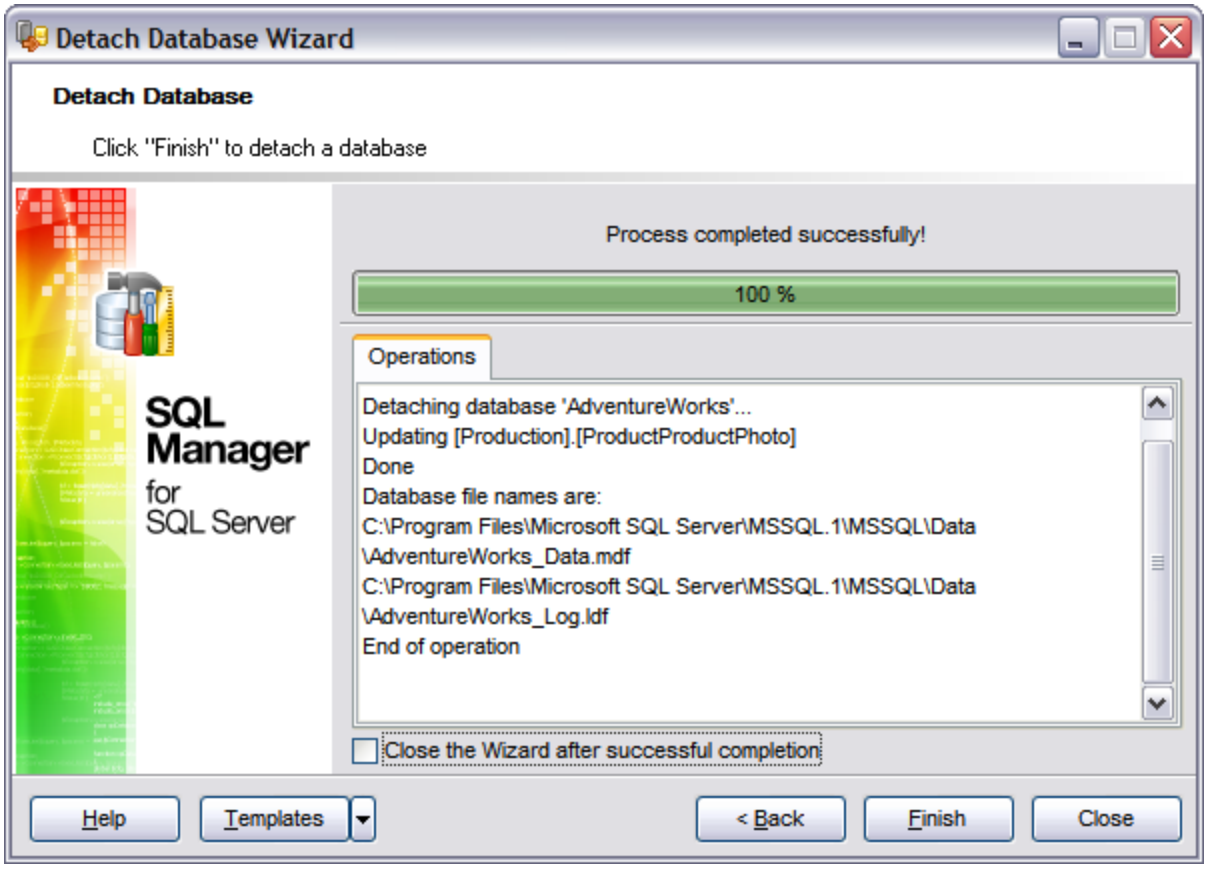
When you are done, click the **Next** button to proceed to the Detaching database step of the wizard.

9.2.2.3 Detaching database

This step of the wizard is intended to inform you that all necessary options have been set, and you can start the detach database process.

☑ **Close the Wizard after successful completion**
If this option is selected, the wizard is closed automatically when the process is completed.

The **Operations** tab allows you to view the log of operations and errors (if any).



If necessary, you can save a template for future use.

Click the **Finish** button to complete the operation.

© 2008 EMS Database Management Solutions

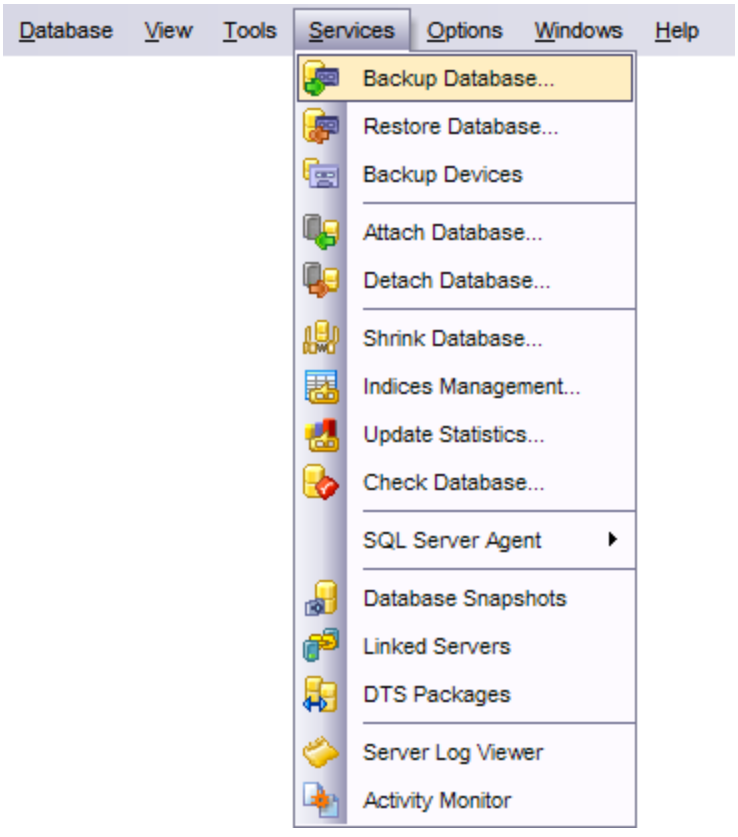
9.2.3 Backup Database

Backup Database Wizard allows you to perform the database backup operation on your SQL Server system (the *BACKUP DATABASE* Transact-SQL statement is issued).

This operation is used to create a backup copy of an entire database, transaction log or files/filegroups.

To run the wizard, select the **Services | Backup Database...** main menu item, or right-click the database alias in the DB Explorer tree and select the **Tasks | Backup Database...** item of the context menu.

- Selecting database to backup
- Setting backup type and options
- Selecting database files to backup
- Selecting backup devices
- Setting schedule options
- Setting media options
- Setting advanced backup options
- Setting transaction log backup options
- Setting data transfer options
- Running database backup
- Using templates



Availability:

© 2008 EMS Database Management Solutions

Full version (for Windows)	**Yes**
Lite version (for Windows)	**Yes**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

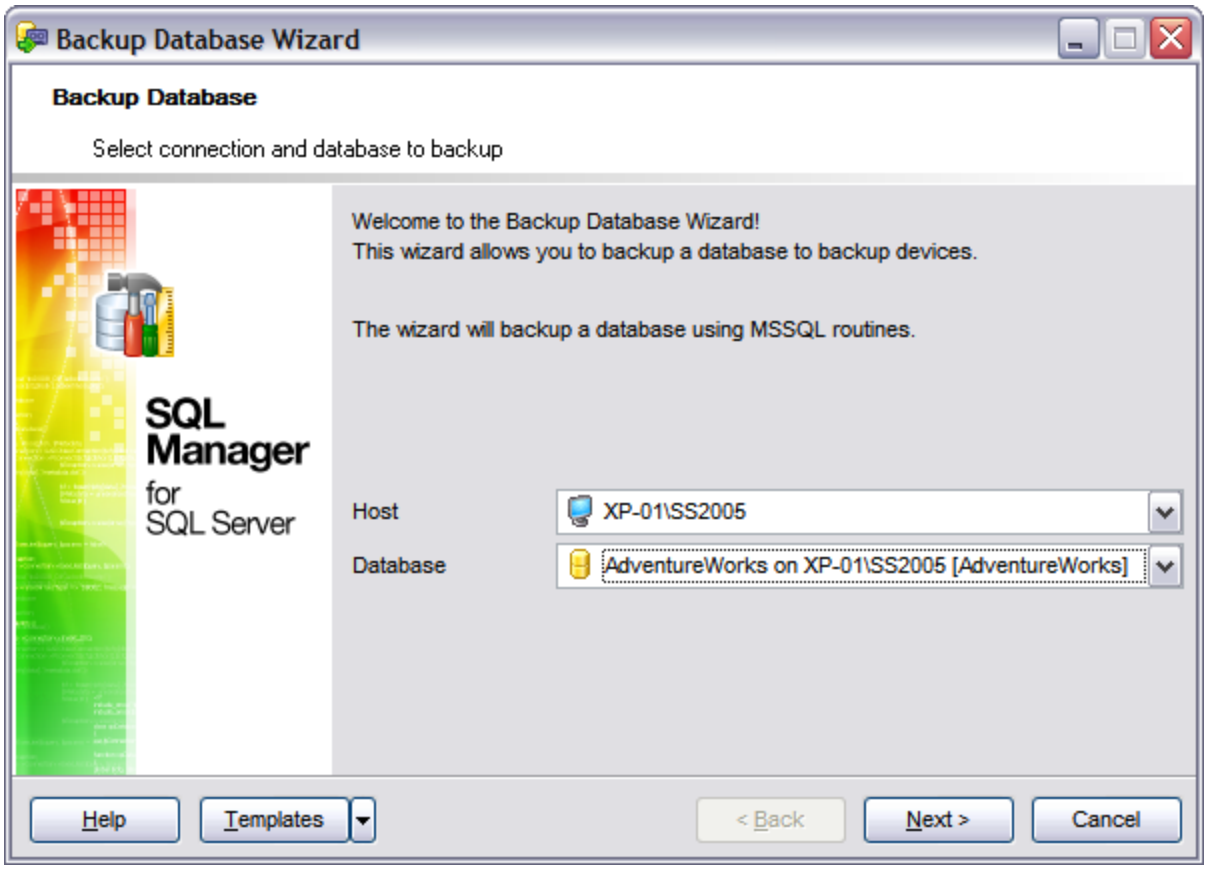
See also:

Attach Database
Detach Database
Restore Database
Shrink Database
Indices Management
Update Statistics
Check Database
Grant Manager

9.2.3.1 Selecting database to backup

This step of the wizard allows you to specify the **host/instance** containing the database for backup.

© 2008 EMS Database Management Solutions



Host
Type in or use the drop-down list to select the host/instance where the database resides.

Database
Use the drop-down list of all registered databases on the selected host to specify the database to backup.

Click the **Next** button to proceed to the Setting backup type and options step of the wizard.

9.2.3.2 Setting backup type and options

This step of the wizard allows you to set up basic **options** pertaining to the database backup operation.

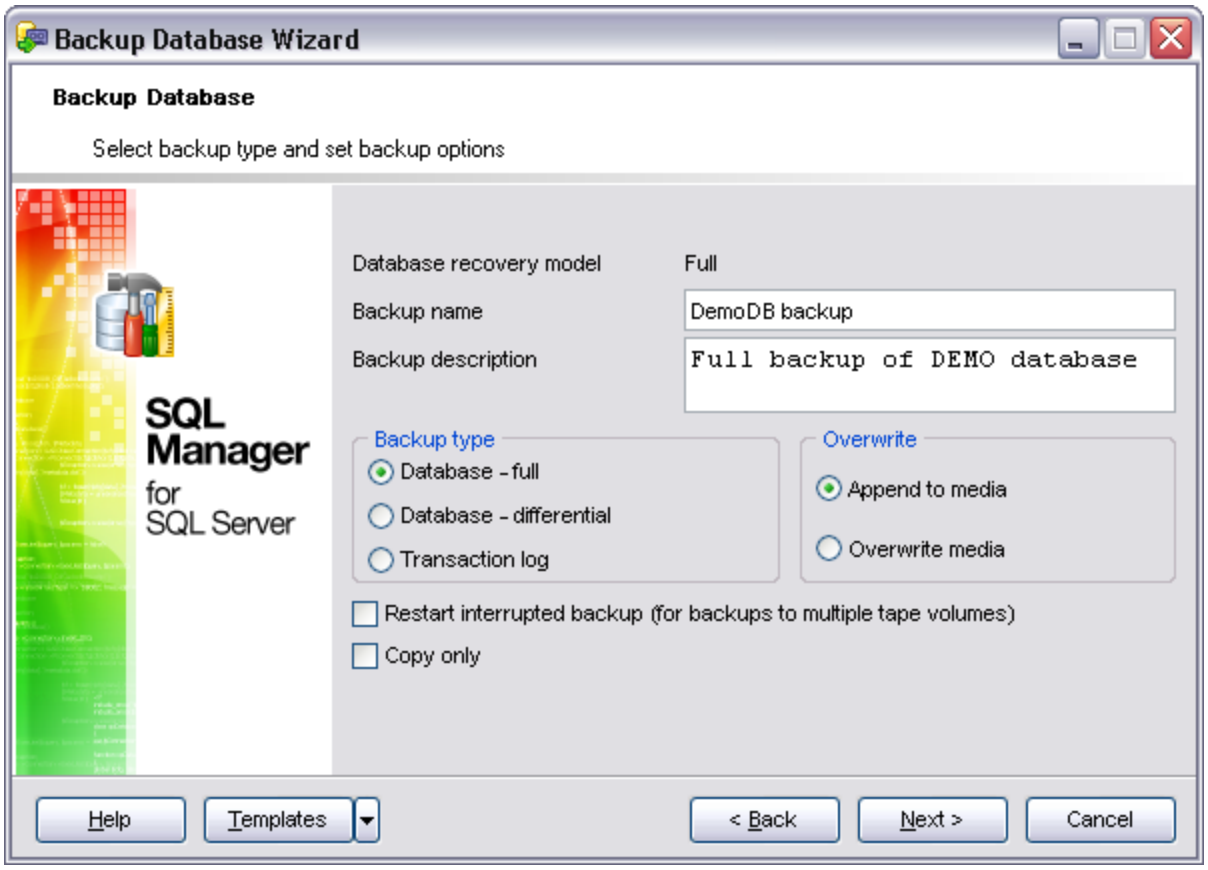
Database recovery model
This label displays the recovery model defined for the database.

Backup name
Enter a name for the current backup.

Backup description

© 2008 EMS Database Management Solutions

Enter any optional text to describe the backup operation.



Backup type

Use this parameter to define the backup type:
Database - Full (a full database backup which backs up the entire database including the transaction log)
Database - differential (a differential backup which records only the changes made to the data in the database after the last full database backup)
Transaction log (a sequence of log backups provided for a continuous chain of transaction information to support recovery forward from database, differential, or file backups)

Append to media / Overwrite media
Select if the media would be appended or overwritten during the backup operation.

Restart interrupted backup
Specifies that SQL Server restarts an interrupted backup operation. This option may save time because it restarts the backup operation from the point it was last interrupted.
Note: This option can only be used for backups directed to tape media and for backups that span multiple tape volumes. A restart operation never occurs on the first volume of the backup.

Copy only

© 2008 EMS Database Management Solutions

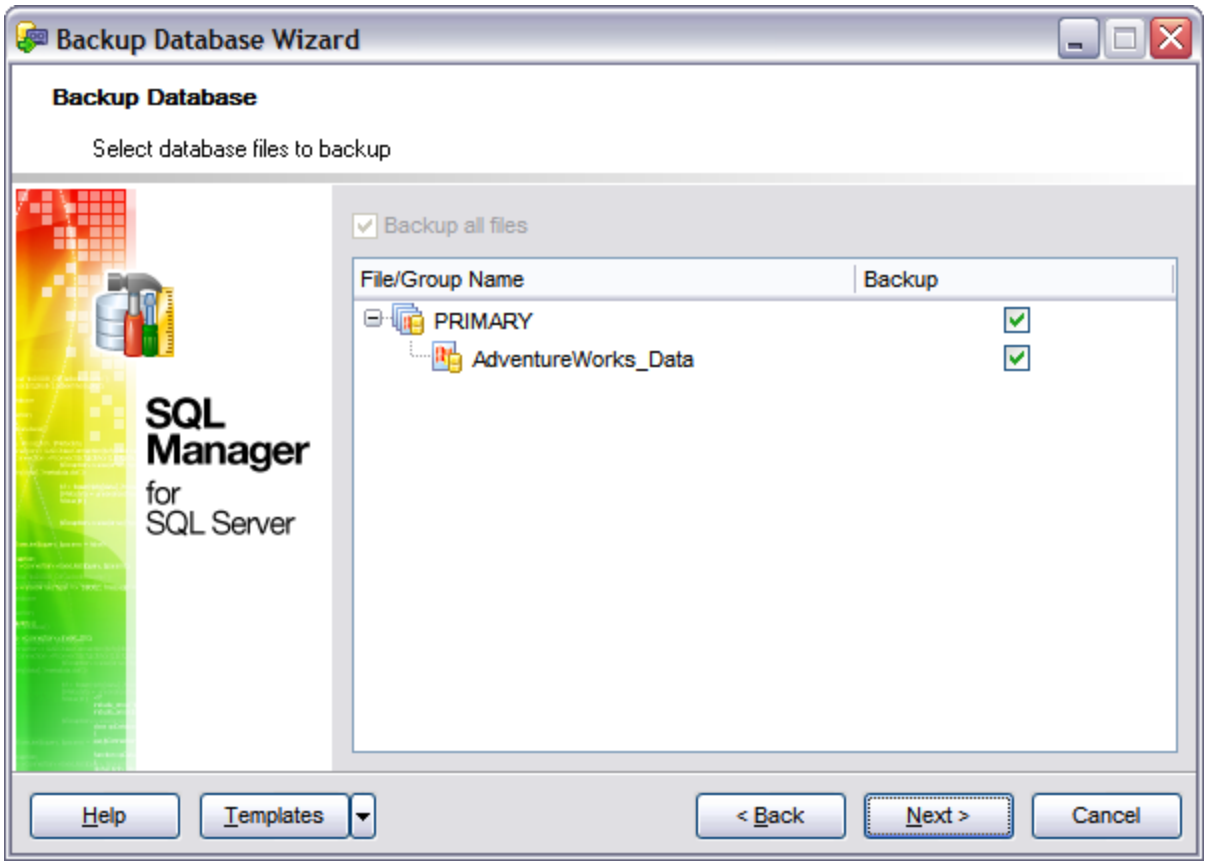
This option specifies that the backup does not affect the normal sequence of backups. A copy-only backup does not affect the overall backup and restore procedures for the database. You can create a copy-only backup for any type of backup.

Click the **Next** button to proceed to the Selecting database files to backup step of the wizard.

9.2.3.3 Selecting database files to backup

This step allows you to **select files** of the database to backup.

To add a file to the backup, select it in the list and set the respective flag in the **Backup** column.
To add all files to the backup, check the **Backup all files** option.



Click the **Next** button to proceed to the Selecting backup devices step of the wizard.

© 2008 EMS Database Management Solutions

9.2.3.4 Selecting backup devices

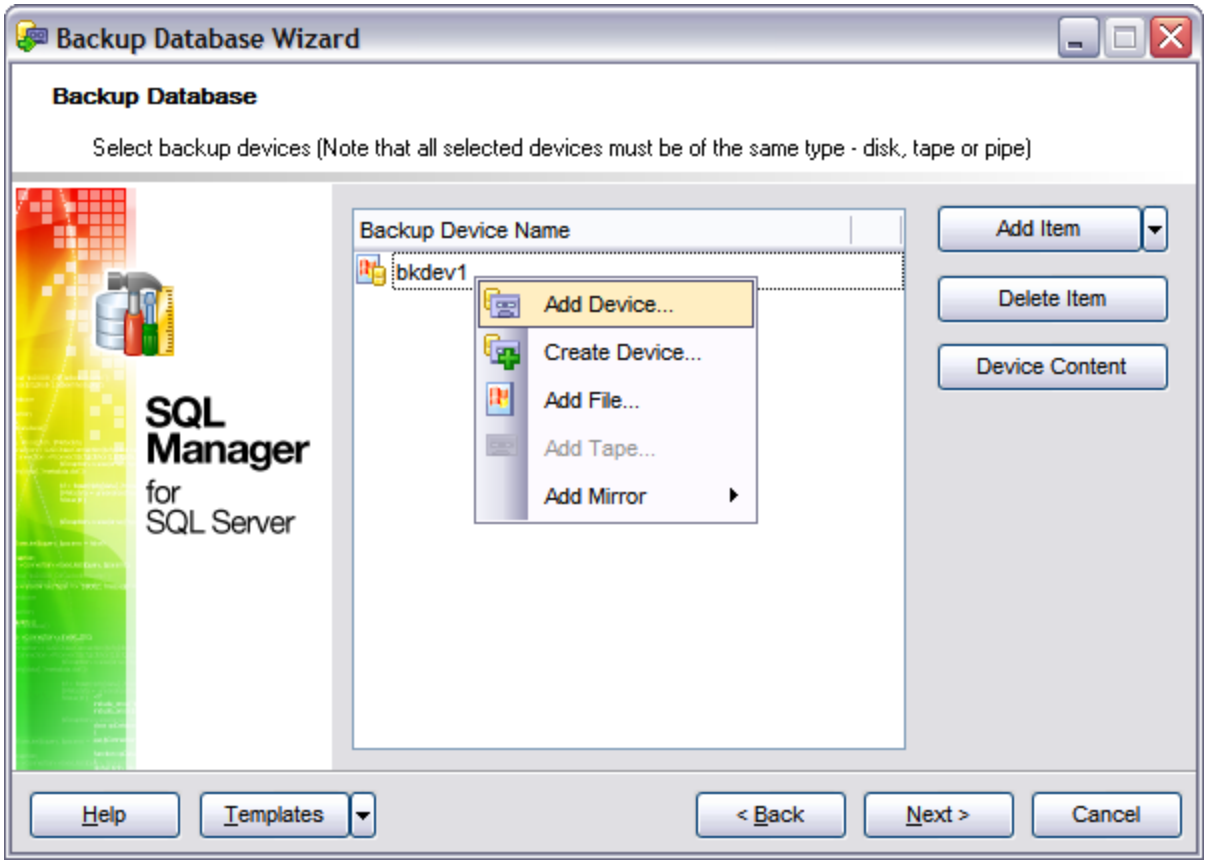
Use this step of the wizard to specify the **devices** to backup the database to. Note that all the selected backup devices must be of the same type.

To add a device, click the **Add Item** button and select a media type from the menu. Alternatively, you can use the context menu of the **Backup Device** list area for the same purpose. Select the required backup device using the corresponding dialog.

To remove a device from the list, select it and click the **Delete Item** button.

To browse the selected device, click the **Device Content** button.

Using the **Add Item** menu and items of the context menu you can also *create a new device* and specify its properties using the New Logical Backup Device dialog, *add a file*, *a tape* or *set mirroring* of backup media.



Disk backup devices are files on hard disks or other disk storage media are the same as regular operating system files. Referring to a disk backup device is the same as referring to any other operating system file. You can define disk backup devices on a local disk of a server, or on a remote disk on a shared network resource, and they can be as large or as small as needed. The maximum file size is the free disk space available on the disk.

Tape backup devices are used in the same way as disk devices, but the tape device

© 2008 EMS Database Management Solutions

must be connected physically to the computer running an instance of SQL Server, and if a tape backup device is filled during the backup operation, but more data still needs to be written, SQL Server prompts for a new tape and continues the backup operation. To back up SQL Server data to tape, use a tape backup device or tape drive supported by the Microsoft Windows platform. Additionally, use only the tapes recommended by the drive manufacturer for the specific tape drive.

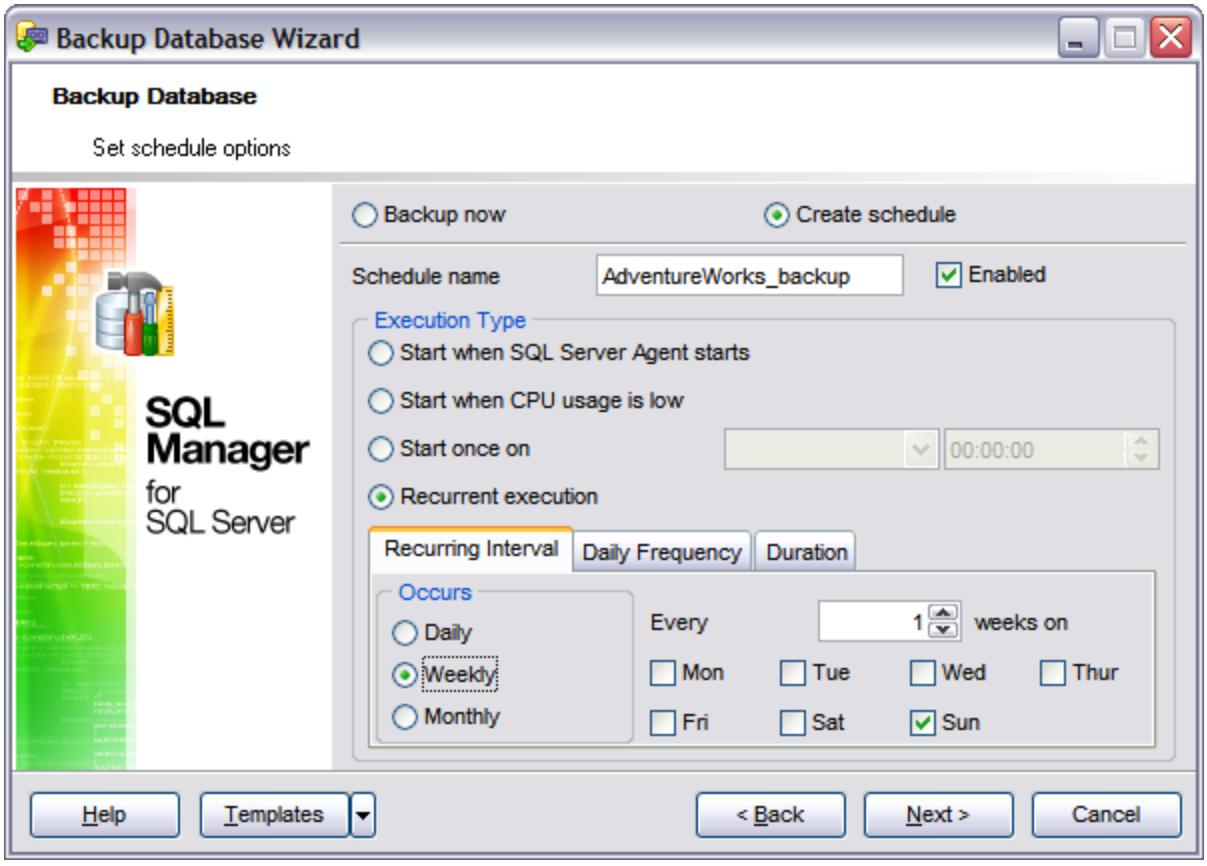
Click the **Next** button to proceed to the Setting schedule options step of the wizard.

9.2.3.5 Setting schedule options

Using this step you can choose to create a **scheduled** backup: select the **Create schedule** option and define the scheduling parameters.

Schedule name
Enter a name of the new backup schedule.

Enabled
Check this option to enable the scheduled backup immediately after its creation.



Execution Type

This panel allows you to define the execution behavior of the schedule:

© 2008 EMS Database Management Solutions

Start when SQL Server Agent starts
The scheduled backup operation will be started each time upon the SQL Server Agent startup.

Start when CPU usage is low
The scheduled backup operation will be started each time when the CPU usage is low.

Start once on...
The scheduled backup operation will be started only once on the defined date at the specified time.

Recurrent Execution
The scheduled backup operation will be started recurrently.

The following tabs allow you to setup the schedule routine:

Recurring Interval
This tab allows you to set the schedule recurrence periodicity in days: you can run your schedule *daily*, *weekly*, *monthly*, every several days, or on a certain day(s) of the week.

Daily Frequency
This tab allows you to set the schedule recurrence periodicity in hours: you can run your schedule once at a specified time, or run it every few hours or minutes after the specified interval.

Duration
This tab allows you to set the schedule lifetime: you can set the schedule start date and the expiration date in the respective edit fields.

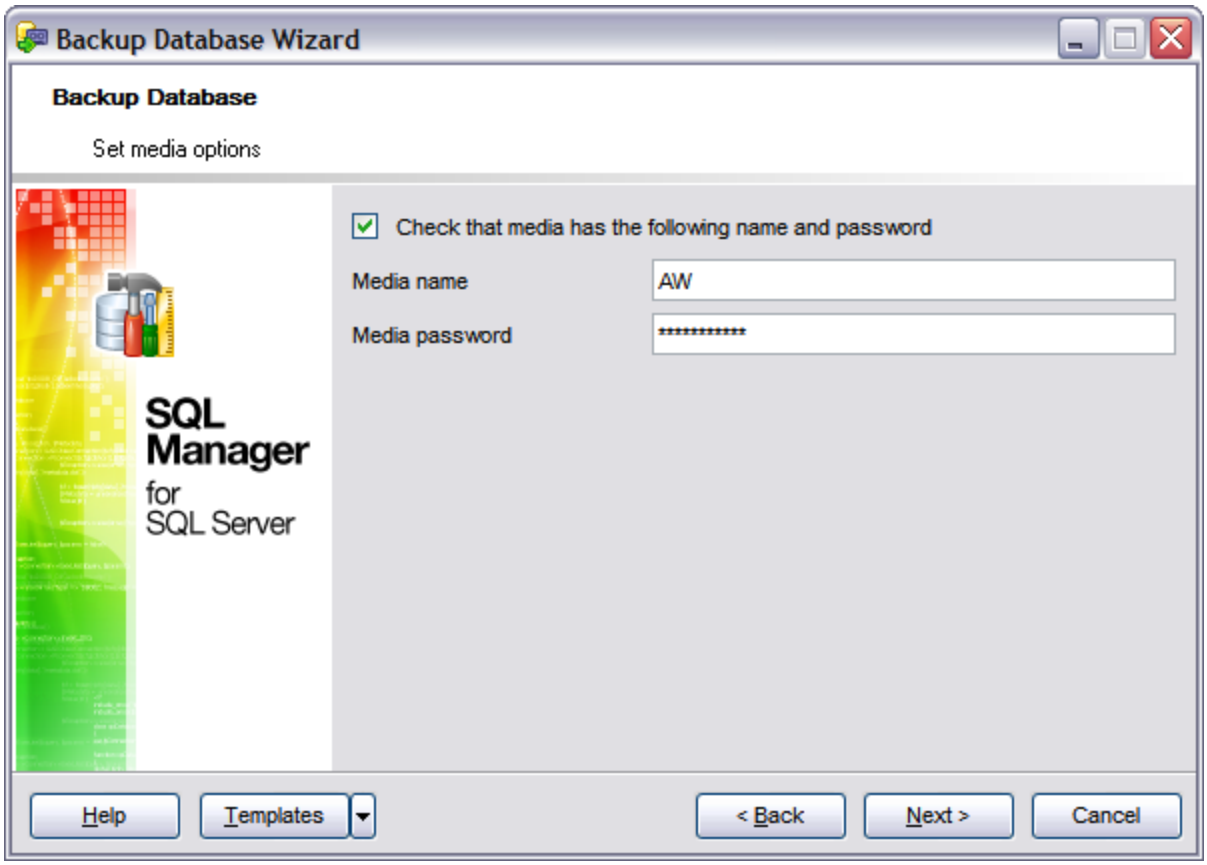
Click the **Next** button to proceed to the Setting media options step of the wizard.

9.2.3.6 Setting media options

This step is only available if the **Append to media** option was selected on the Setting backup type and options step of the wizard.

At this step you can specify whether the application should check the name and password combination for the media. To enable this feature, check the corresponding option and enter a **name** and a **password** for the media.

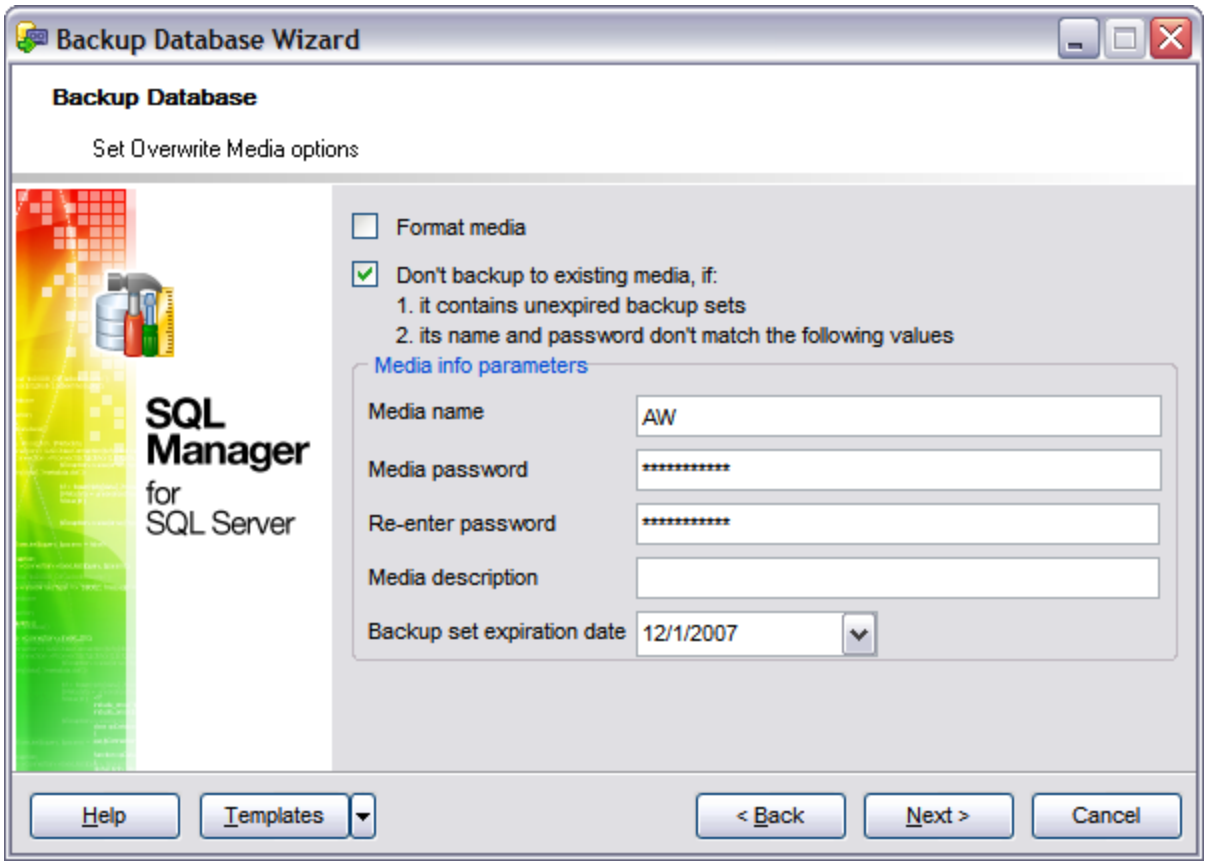
© 2008 EMS Database Management Solutions



The following step is only available if the **Overwrite media** option is selected on the Setting backup type and options step of the wizard.

At this step you can set a number of options pertaining to the media being overwritten.

© 2008 EMS Database Management Solutions



Format media
Specifies that the media header should be written on all volumes used for this backup operation. Any existing media header is overwritten. This option invalidates the entire media contents, ignoring any existing content.
Important: Use this option carefully. Formatting one backup device or medium renders the entire media set unusable. For example, if a single tape belonging to an existing striped media set is initialized, the entire media set is rendered useless.

Don't backup to existing media if...
If the option is ticked off, the media will be first examined for any existing unexpired backup sets and for the correctness of the password. If the password is incorrect or the media contains any unexpired backup sets, the current media will not be used for backup.

Media name
Enter the name of the media.

Media password
Enter the password for the media.

Re-enter password
Repeat the password to avoid misprints.

Media description
Supply any optional text to describe the media.

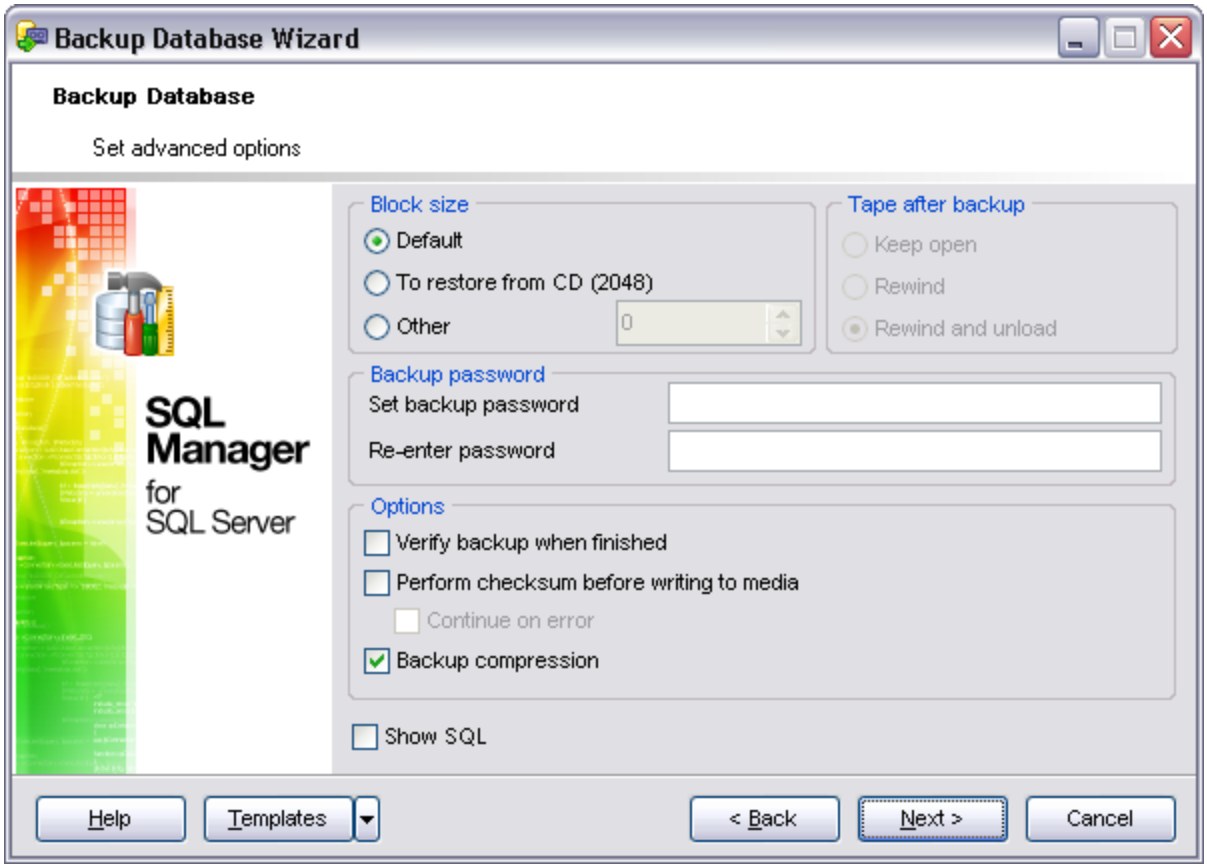
© 2008 EMS Database Management Solutions

Backup set expiration date
Define expiration date for the current backup set.

Click the **Next** button to proceed to the Setting advanced backup options step of the wizard.

9.2.3.7 Setting advanced backup options

This step offers some **additional options** that may be useful for the database backup operation.



Block size
Define the size of a data block. Possible values are: *Default*, *To restore from CD* (2048), *Other* (i.e. you can specify an arbitrary value).

Tape after backup
Specify whether the tape should be *rewound*, *rewound and unloaded*, or *kept open* when the backup operation is completed.

If necessary, you can set a password for the backup: enter the password twice in the

© 2008 EMS Database Management Solutions

respective edit fields.

Verify backup when finished
Check this option to enable backup verification.

Perform checksum before writing to media
This option enables/disables checksum verification before writing to the backup media.

Continue on error
This option determines whether the backup operation will be stopped or forced to continue if an error occurs.

Backup compression
Defines the values that are used to specify the backup compression option.

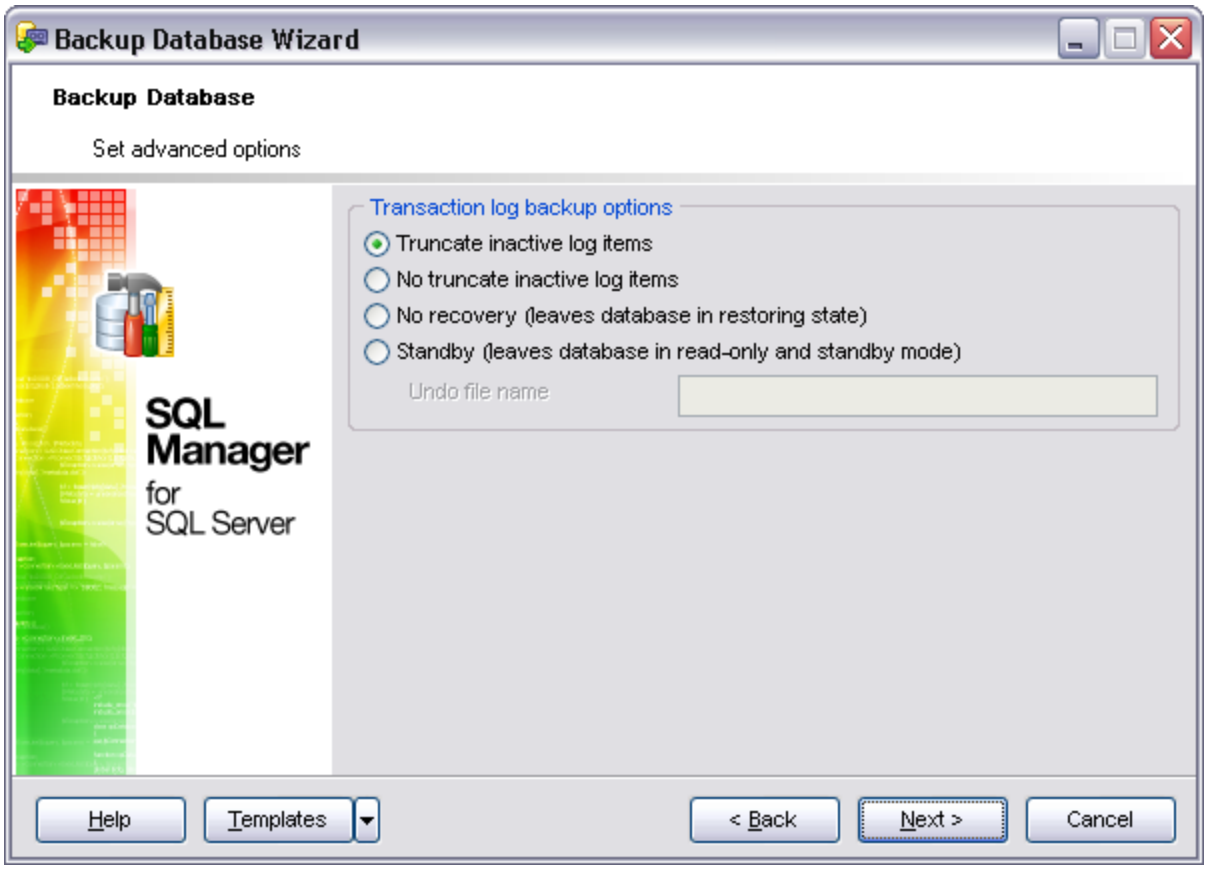
Show SQL
This option enables/disables SQL statements displayed within the **Operations** tab when running database backup.

When you are done, click the **Next** button to proceed to the next step of the wizard.

9.2.3.8 Setting transaction log backup options

This step offers some **additional options** that may be useful for the database backup operation.
(only displayed if the *Transaction log* is selected at the Setting backup type and options step and if server version lower than SQL Server 2005 is used.)

© 2008 EMS Database Management Solutions



Truncate inactive log items
The log is truncated after all the records within one or more virtual log files become inactive.

No truncate inactive log items
Specifies that the log not be truncated and causes the Database Engine to attempt the backup regardless of the state of the database. This option allows backing up the log in situations where the database is damaged.

No recovery (leaves database in restoring state)
Backs up the tail of the log and leaves the database in the RESTORING state. This is useful when failing over to a secondary database or when saving the tail of the log before a restore operation.

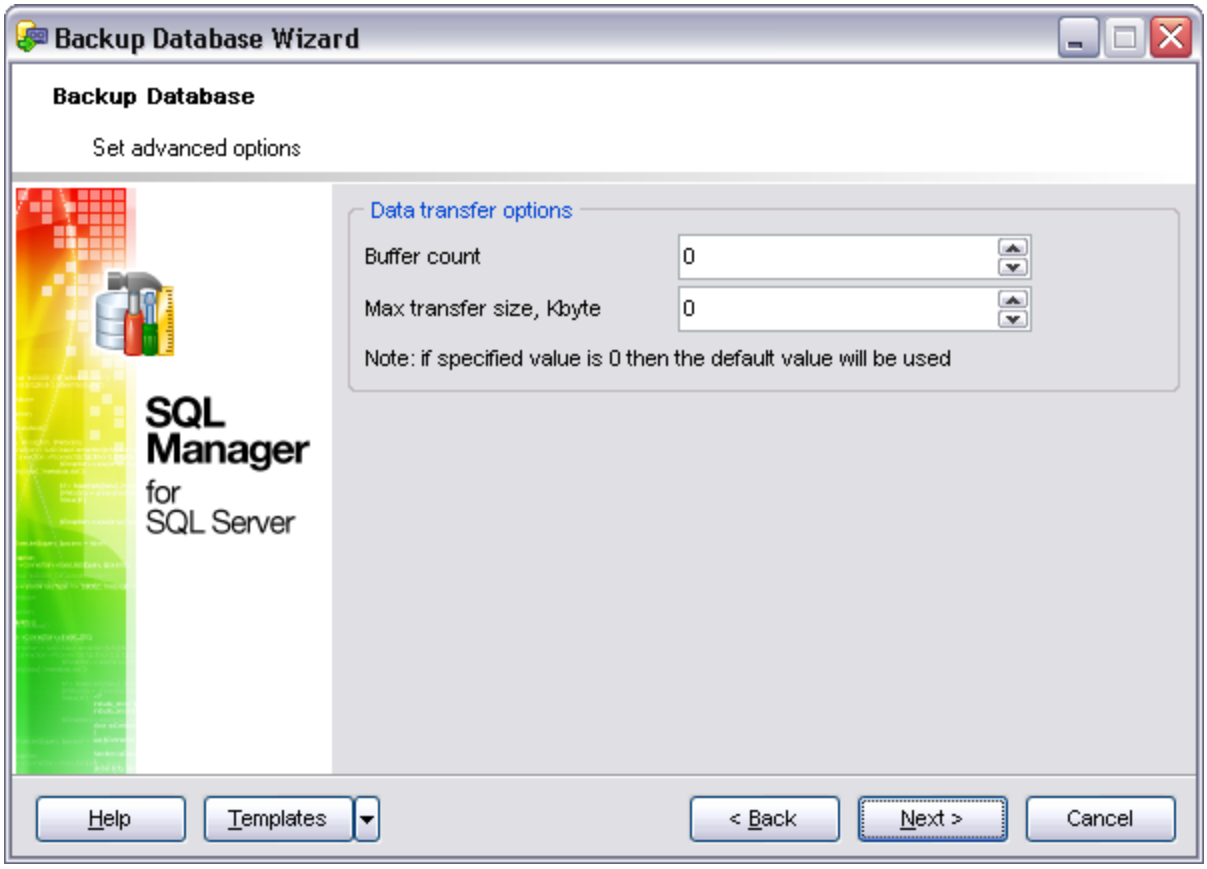
Standby (leaves database in read-only and standby mode)
Backs up the tail of the log and leaves the database in a read-only and STANDBY state. Using standby mode requires a standby file, specified by **Undo file name**.

When you are done, click the **Next** button to proceed to the next step of the wizard.

© 2008 EMS Database Management Solutions

9.2.3.9 Setting data transfer options

This step offers some **additional options** that may be useful for the database backup operation.
(only displayed for SQL Server 2008).



Buffer count
Specifies the total number of I/O buffers to be used for the backup operation. You can specify any positive integer; however, large numbers of buffers might cause "out of memory" errors because of inadequate virtual address space in the Sqlservr.exe process.

Max transfer size
Specifies the largest unit of transfer in bytes to be used between SQL Server and the backup media. The possible values are multiples of 65536 bytes (64 KB) ranging up to 4194304 bytes (4 MB).

When you are done, click the **Next** button to proceed to the last step of the wizard.

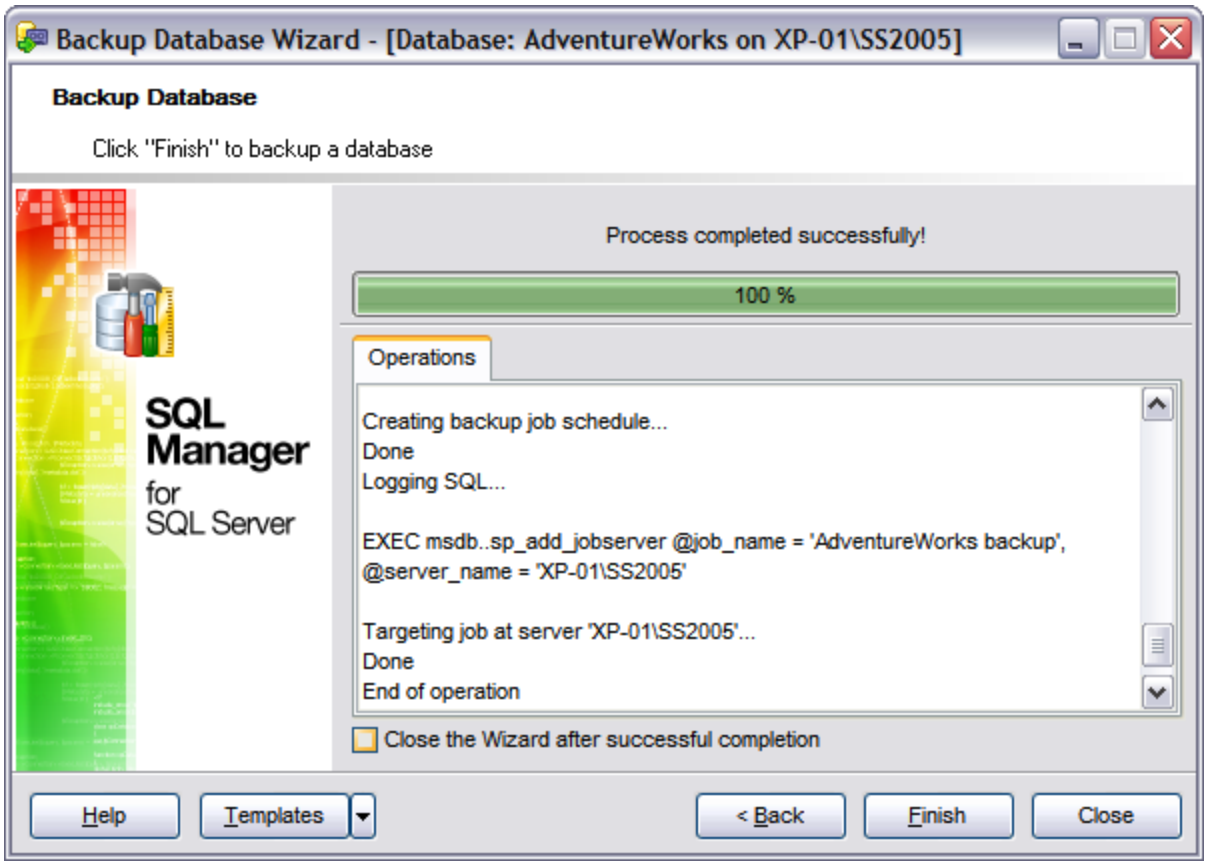
9.2.3.10 Running database backup

This step of the wizard is intended to inform you that all necessary options have been set, and you can start the backup database process.

© 2008 EMS Database Management Solutions

Close the Wizard after successful completion
If this option is selected, the wizard is closed automatically when the process is completed.

The **Operations** tab allows you to view the log of operations and errors (if any).



If necessary, you can save a template for future use.

Click the **Finish** button to run the backup database operation.

9.2.4 Restore Database

Restore Database Wizard allows you to perform the database restore operation on your SQL Server system (the *RESTORE DATABASE* Transact-SQL statement is issued).

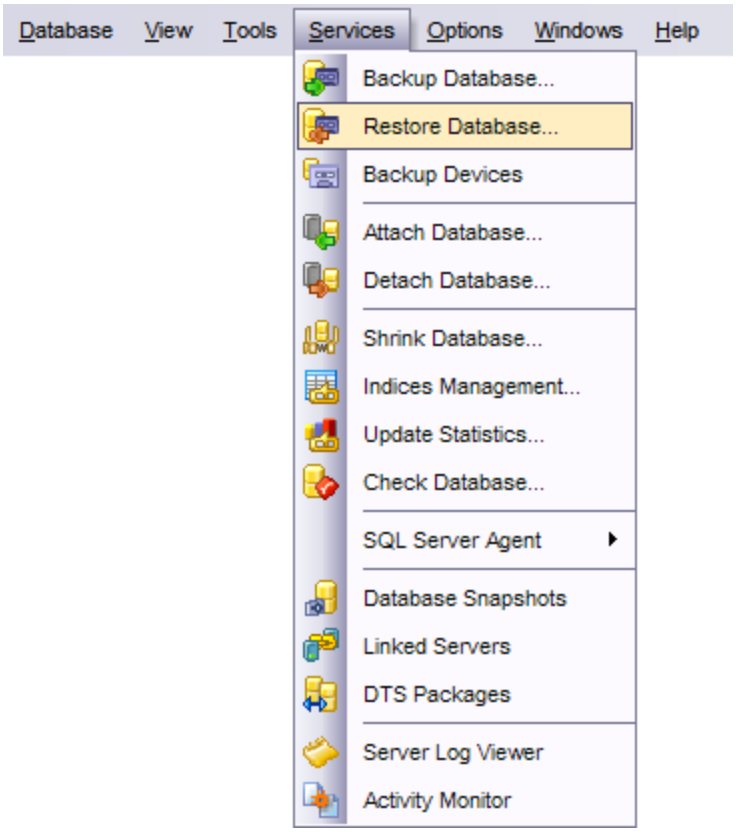
This operation is used to rebuild a damaged or corrupted database that has been backed up using Backup Database Wizard. You can restore an entire database, part of a database, specific files and filegroups, transaction logs.

To run the wizard, select the **Services | Restore Database...** main menu item, or right-click the database alias in the DB Explorer tree and select the **Tasks | Restore Database...** item of the context menu. Alternatively, you can right-click the **Databases**

© 2008 EMS Database Management Solutions

node and select the **Tasks | Restore Database...** context menu item.

- Selecting destination database
- Setting restore type and recovery options
- Selecting backup set to restore
- Selecting files to restore
- Setting advanced restore options
- Running database restore
- Using templates



Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**Yes**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

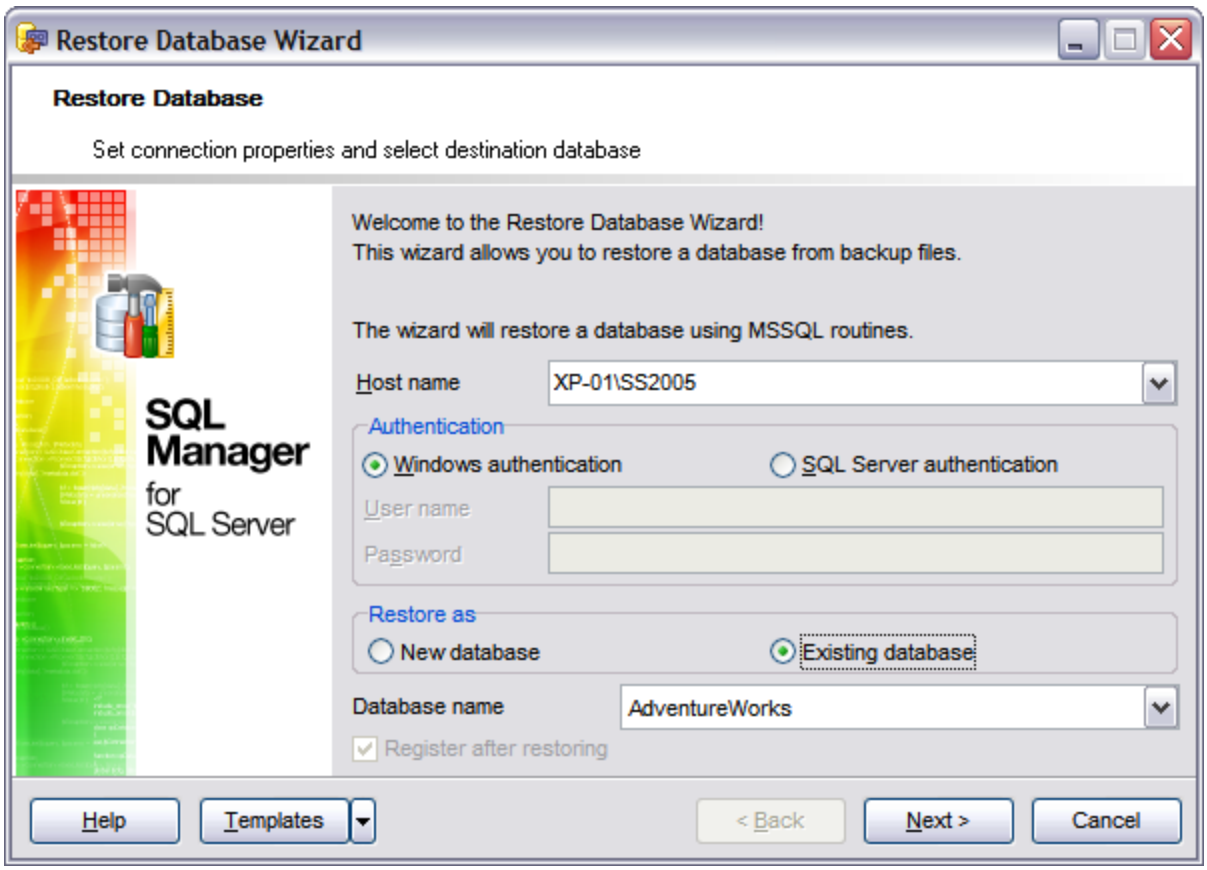
Attach Database
Detach Database
Backup Database

© 2008 EMS Database Management Solutions

Shrink Database
Indices Management
Update Statistics
Check Database
Grant Manager

9.2.4.1 Selecting destination database

This step of the wizard allows you to specify the **host/instance** to restore database, and to provide **authentication** parameters.



Host
Type in or use the drop-down list to select the host/instance name to restore database to.

Authentication
Specify the type of Microsoft® SQL Server™ authentication to be used for the connection: the built-in **Windows authentication** or **SQL Server authentication**.

If 'SQL Server' has been selected as the *authentication type*, you should also provide *authorization* settings: **User name** and **Password**.

© 2008 EMS Database Management Solutions

Restore as

Select *New database* if you want to create a new database, or select *Existing database* to restore to an existing database.

Database name
Enter a name for the new database or select the database to restore to from the drop-down list of all available databases on the specified host.

Register after restoring
This option indicates that the Database Registration Info dialog for the newly restored database will appear immediately after restoring the database (you need to register the database to start working with it in SQL Manager). If you intend to register the database later, uncheck this option.

Click the **Next** button to proceed to the Setting restore type and recovery options step of the wizard.

9.2.4.2 Setting restore type and recovery options

Use this step of the wizard to define a **restore type** and set **recovery options**.

Restore type

This group of options allows you to specify the restore type to be used for the restore operation.

Database
Full restore or differential restore of entire database or individual files and filegroups. Files and filegroups can be restored either from a file or filegroup backup operation, or from a full database backup operation.

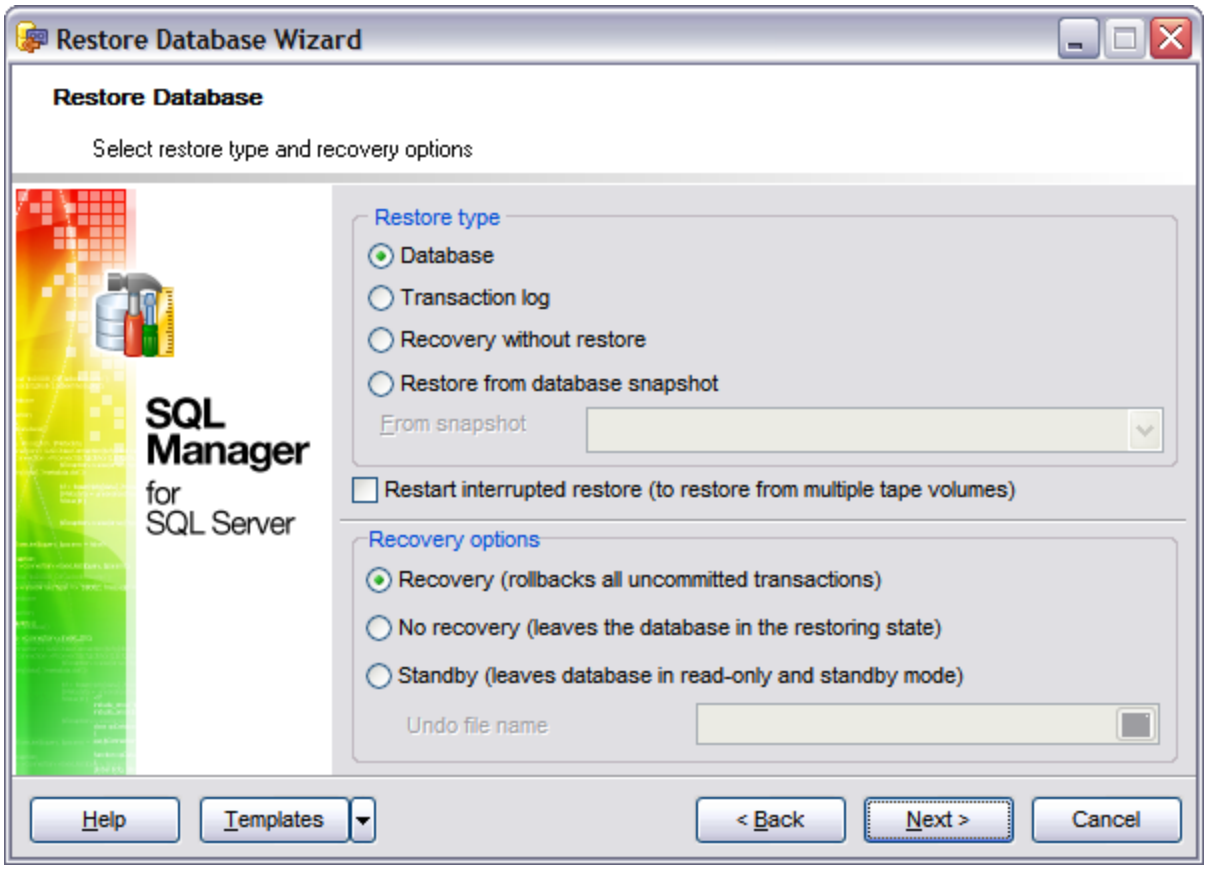
Transaction log
Specifies that a transaction log backup will be used to restore this database. SQL Server checks the backed up transaction log to ensure that the transactions are being loaded into the correct database and in the appropriate sequence.

Recovery without restore
Select this option to execute the database recovery only.

Recovery from database snapshot
Select this option to restore the database from a previously saved snapshot.

Restart interrupted restore
This option specifies that SQL Server should restart a restore operation that has been interrupted. The restore operation is restarted from the point it was interrupted.

© 2008 EMS Database Management Solutions



Recovery options

This group of options allows you to specify the recovery options to be applied to the restore operation.

Recovery
This option specifies that the restore operation rolls back any uncommitted transactions. After the recovery process the database is ready for use.

No recovery
This option specifies that the restore operation does not roll back any uncommitted transactions.
Note: When this option is selected, the database is not usable in this intermediate, non-recovered state.
When used with a file or filegroup restore operation, this option forces the database to remain in the restoring state after the restore operation. This is useful in either of the following situations:

- a restore script is being run and the log is always being applied;
- a sequence of file restores is used and the database is not intended to be usable between two of the restore operations.

Standby
This option allows the database to be brought up for read-only access between transaction log restores and can be used with either warm standby server situations or

© 2008 EMS Database Management Solutions

special recovery situations in which it is useful to inspect the database between log restores.

Undo file name
Specify the undo file name, so that the recovery effects can be undone. The size required for the undo file depends on the volume of undo actions resulting from uncommitted transactions. The same undo file can be used for consecutive restores of the same database.
Important: If free disk space is exhausted on the drive containing the specified undo file name, the restore operation stops.

Click the **Next** button to proceed to the Selecting backup set to restore step of the wizard.

9.2.4.3 Selecting backup set to restore

At this step of the wizard you need to **select a backup media** to restore the database from.

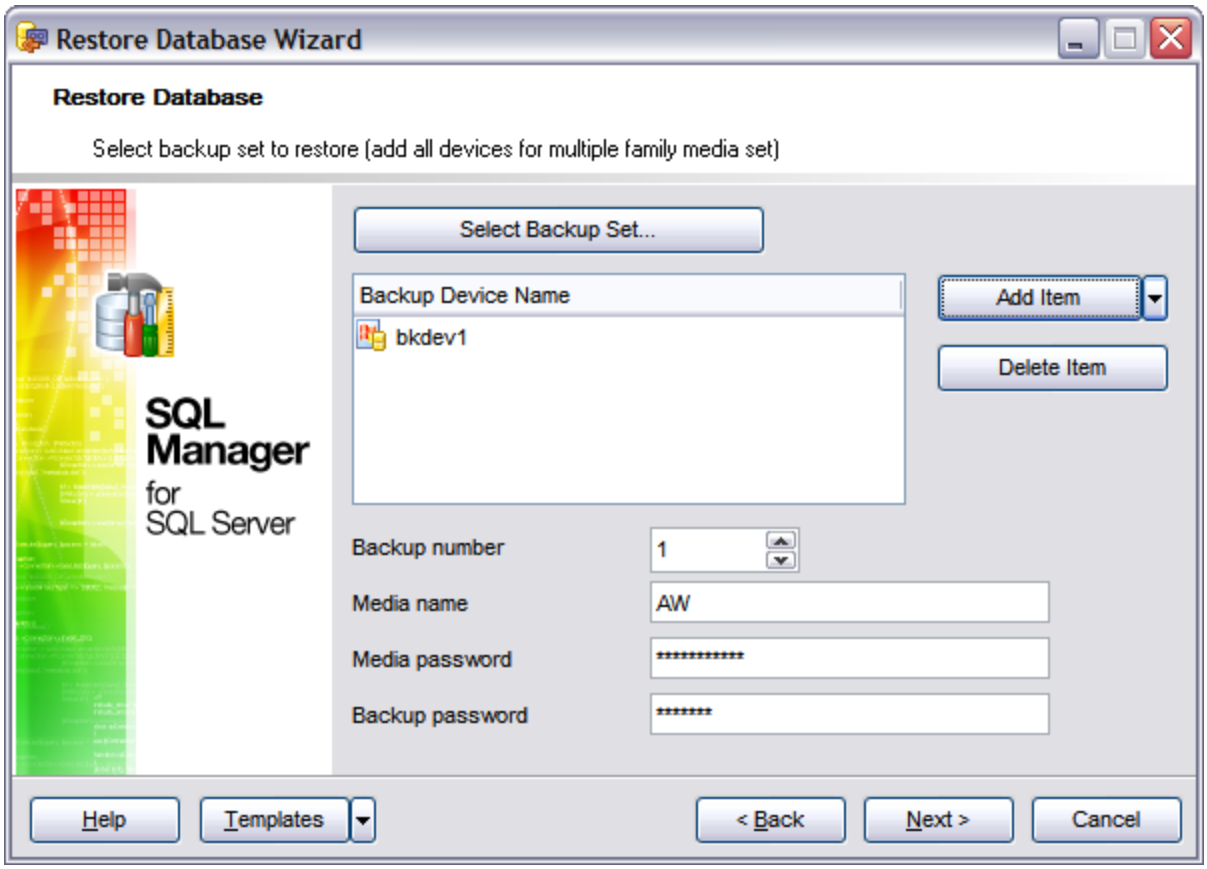
You can specify a backup set manually by adding devices to the device list.

To add a device, click the **Add Item** button and select a media type from the menu. Select the required backup device using the corresponding dialog.

To remove a device from the list, select it and click the **Delete Item** button.

Using the **Add Item** menu you can add a *logical device*, a *file*, or a *tape*.

© 2008 EMS Database Management Solutions



Backup number
Identifies the backup set to be restored. For example, *1* indicates the first backup set on the backup medium and *2* indicates the second backup set.

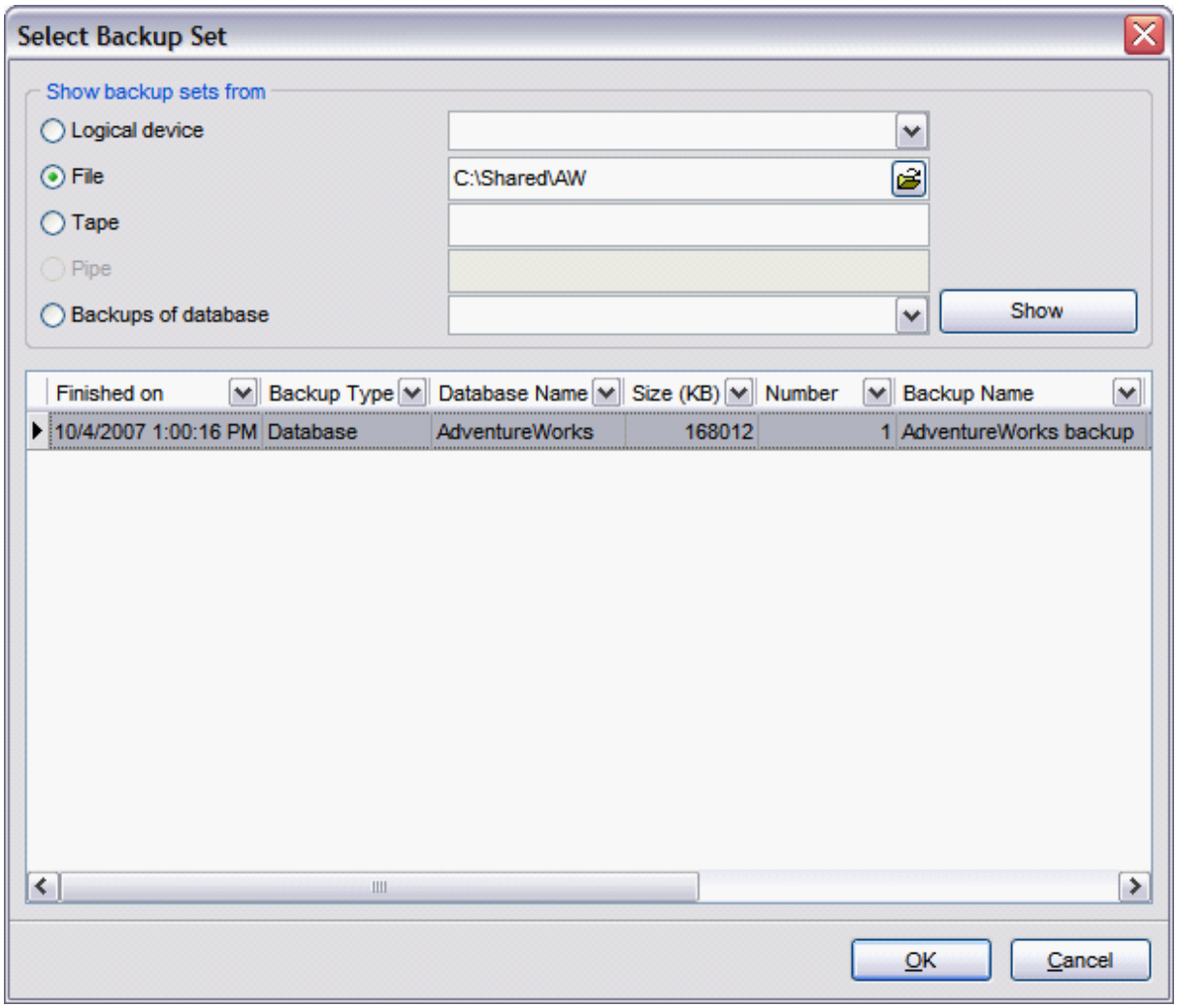
Media name
Specify the name for the media. If provided, the media name must match the media name on the backup volume(s); otherwise, the restore operation terminates. If no media name is given, no check for matching media name on the backup volume(s) is performed.
Important: Consistently using media names in backup and restore operations provides an extra safety check for the media selected for the restore operation.

Media password
Supply the password for the media set. If a password was provided when the media set was formatted, that password must be supplied to access any backup set on that media set.

Backup password
Provide the password for the backup set. If a password was provided when the backup set was created, the password must be supplied to perform any restore operation from the backup set.

To select a whole backup set, click the **Select Backup Set...** button and select a set in the corresponding dialog.

© 2008 EMS Database Management Solutions



Click the **Next** button to proceed to the Selecting files to restore step of the wizard.

9.2.4.4 Selecting files to restore

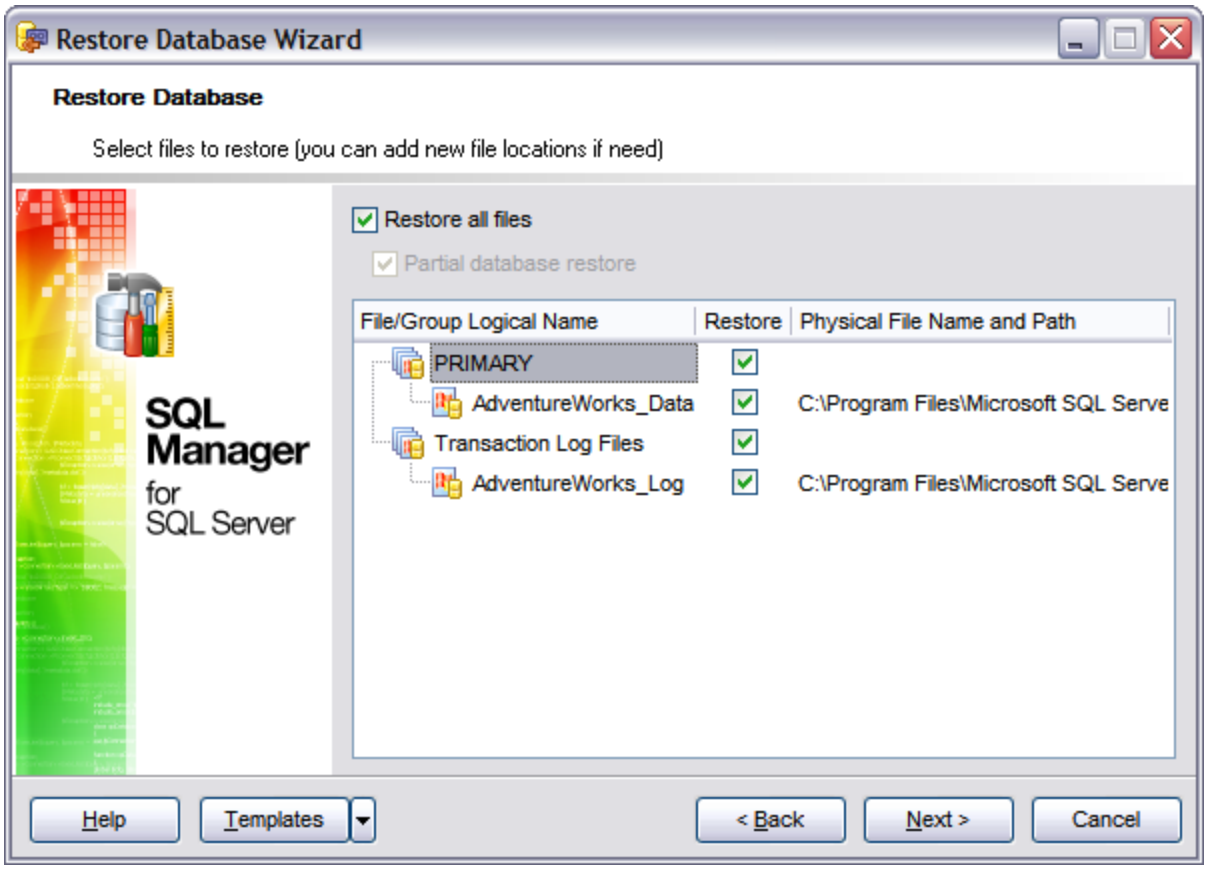
This step allows you to **select files** that will be used to restore the database.

By default the application restores all files. It is indicated by the **Restore all files** option selected. To select files manually, you should uncheck this option.

To add a file to the restore, select it in the list and set the respective flag in the **Restore** column.
To add all files to the restore, check the **Restore all files** option.

You can also edit file paths by clicking the corresponding rows of the **Physical File Name and Path** column.

© 2008 EMS Database Management Solutions



Partial database restore
This option specifies a partial restore operation. Application or user errors often affect an isolated portion of the database, such as a table. Examples of this type of error include an invalid update or a table dropped by mistake. To support recovery from these events, SQL Server provides a mechanism to restore part of the database to another location so that the damaged or missing data can be copied back to the original database.

The granularity of the partial restore operation is the database filegroup. The primary file and filegroup are always restored, along with the files that you specify and their corresponding filegroups.

Click the **Next** button to proceed to the Setting advanced restore options step of the wizard.

9.2.4.5 Setting advanced restore options

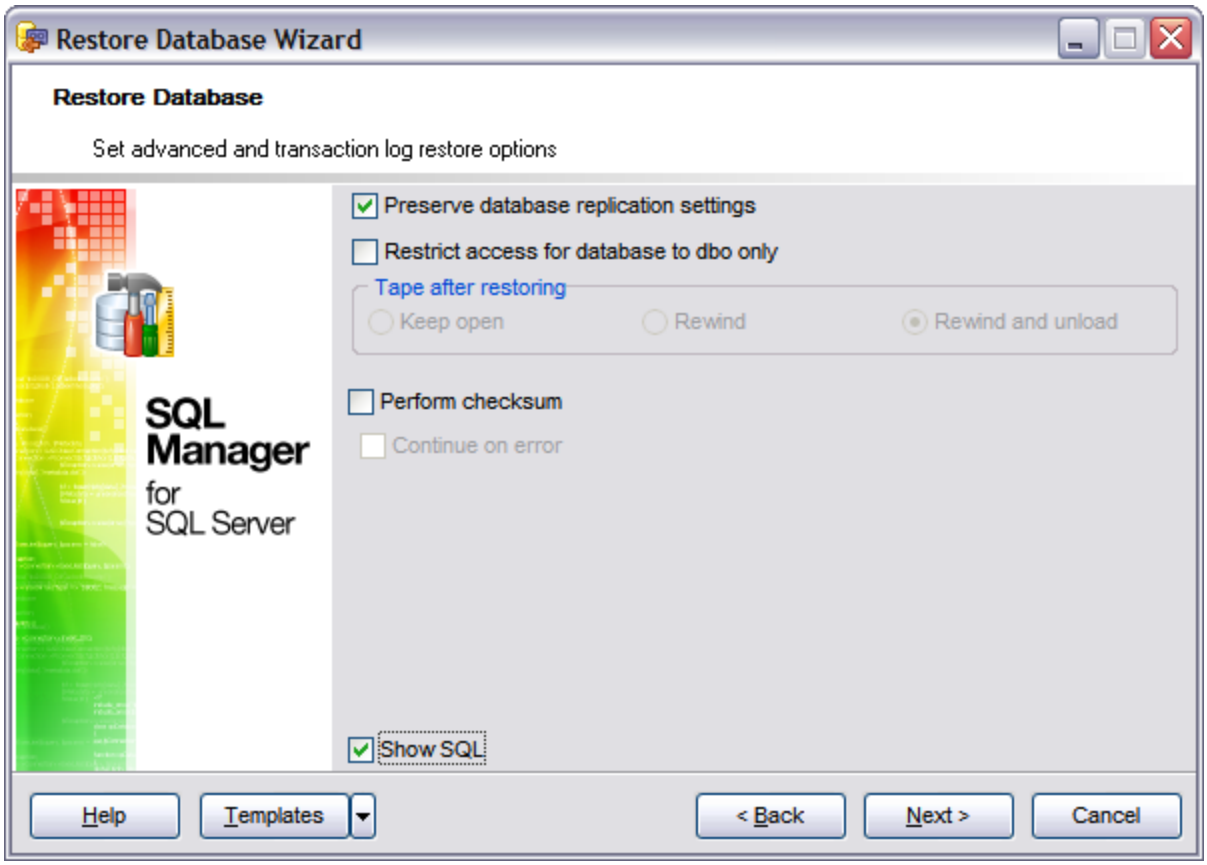
This step offers some **additional options** that are specific for each restoration mode.

Preserve database replication settings
This option specifies that the replication settings are preserved upon restoration of a published database to a server other than that on which it was created. It prevents replication settings from being removed when a database or log backup is restored on a

© 2008 EMS Database Management Solutions

warm standby server and the database is recovered.

Restrict access for database to dbo only
This option restricts access to the newly restored database to members of the *db_owner*, *dbcreator*, or *sysadmin* roles.



Tape after restoring
Specify whether the tape should be *rewound*, *rewound and unloaded*, or *kept open* when the restoration is completed.

Perform checksum
This option enables/disables checksum verification before restoring from the backup media.

Continue on error
This option determines whether the restore operation will be stopped or forced to continue if an error occurs.

Show SQL
This option enables/disables SQL statements displayed within the **Operations** tab when running database restore.

When you are done, click the **Next** button to proceed to the last step of the wizard.

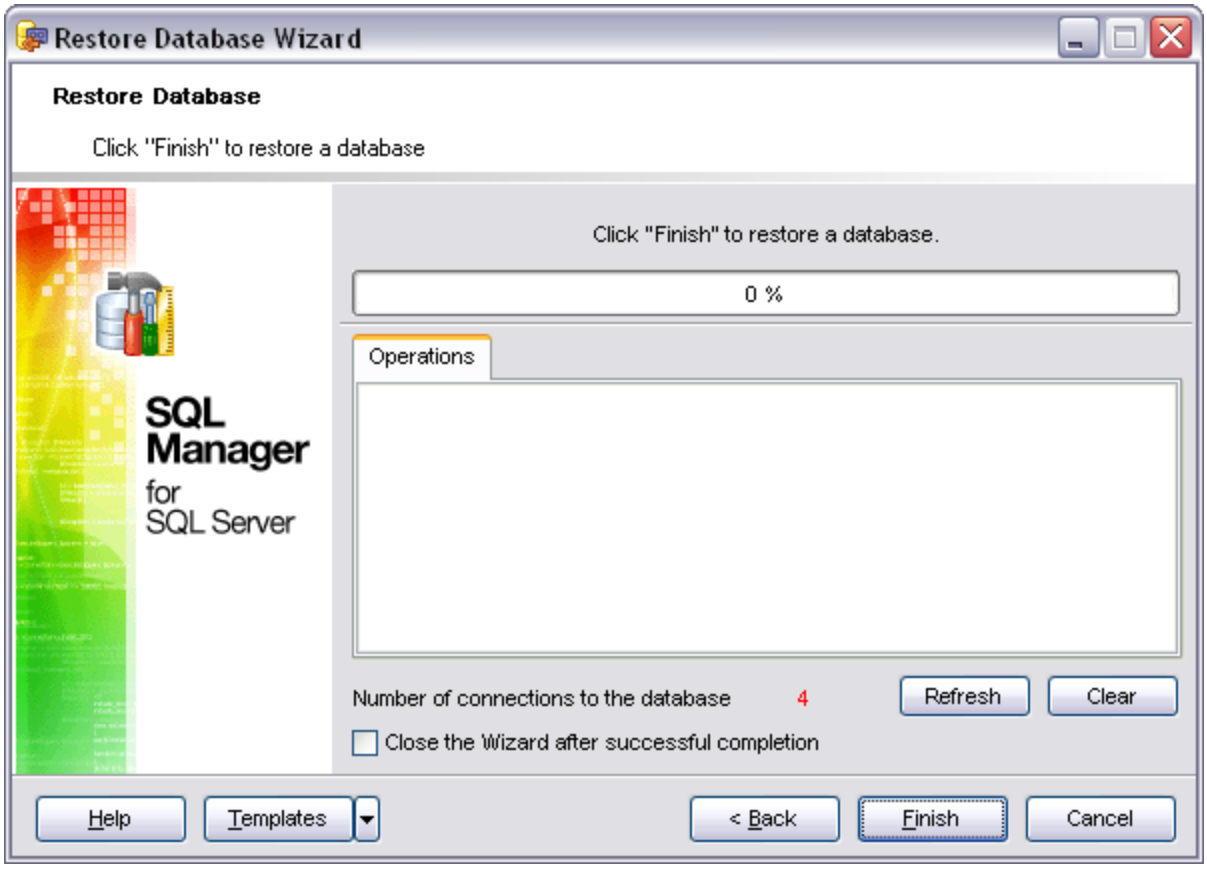
© 2008 EMS Database Management Solutions

9.2.4.6 Running database restore

This step of the wizard is intended to inform you that all necessary options have been set, and you can start the restore database process.

Close the Wizard after successful completion
If this option is selected, the wizard is closed automatically when the process is completed.

The **Operations** tab allows you to view the log of operations and errors (if any).



If necessary, you can save a template for future use.
Click the **Refresh** button to update the number of connections to the database.
Click the **Clear** button to drop all connections to the database you need to restore.

Click the **Finish** button to run the restore database operation.

© 2008 EMS Database Management Solutions

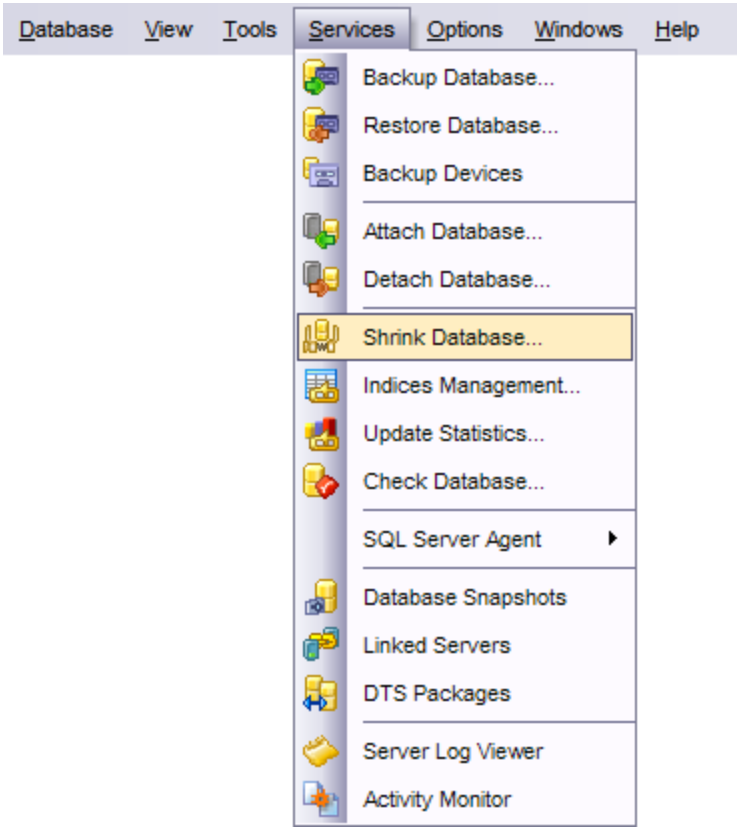
9.2.5 Shrink Database

Shrink Database Wizard allows you to perform database shrinking on your SQL Server system (the *DBCC SHRINKDATABASE* and *SHRINKFILE* Transact-SQL statements are issued).

This operation is used to reduce any file within the database to remove unused pages. Both data and transaction log files can be reduced, or shrunk.

To run the wizard, select the **Services | Shrink Database...** main menu item, or right-click the database alias in the DB Explorer tree and select the **Tasks | Shrink Database...** item of the context menu.

- Selecting database and setting shrink mode
- Defining shrink action for database
- Defining shrink action for files
- Shrinking database
- Using templates



Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**No**

© 2008 EMS Database Management Solutions

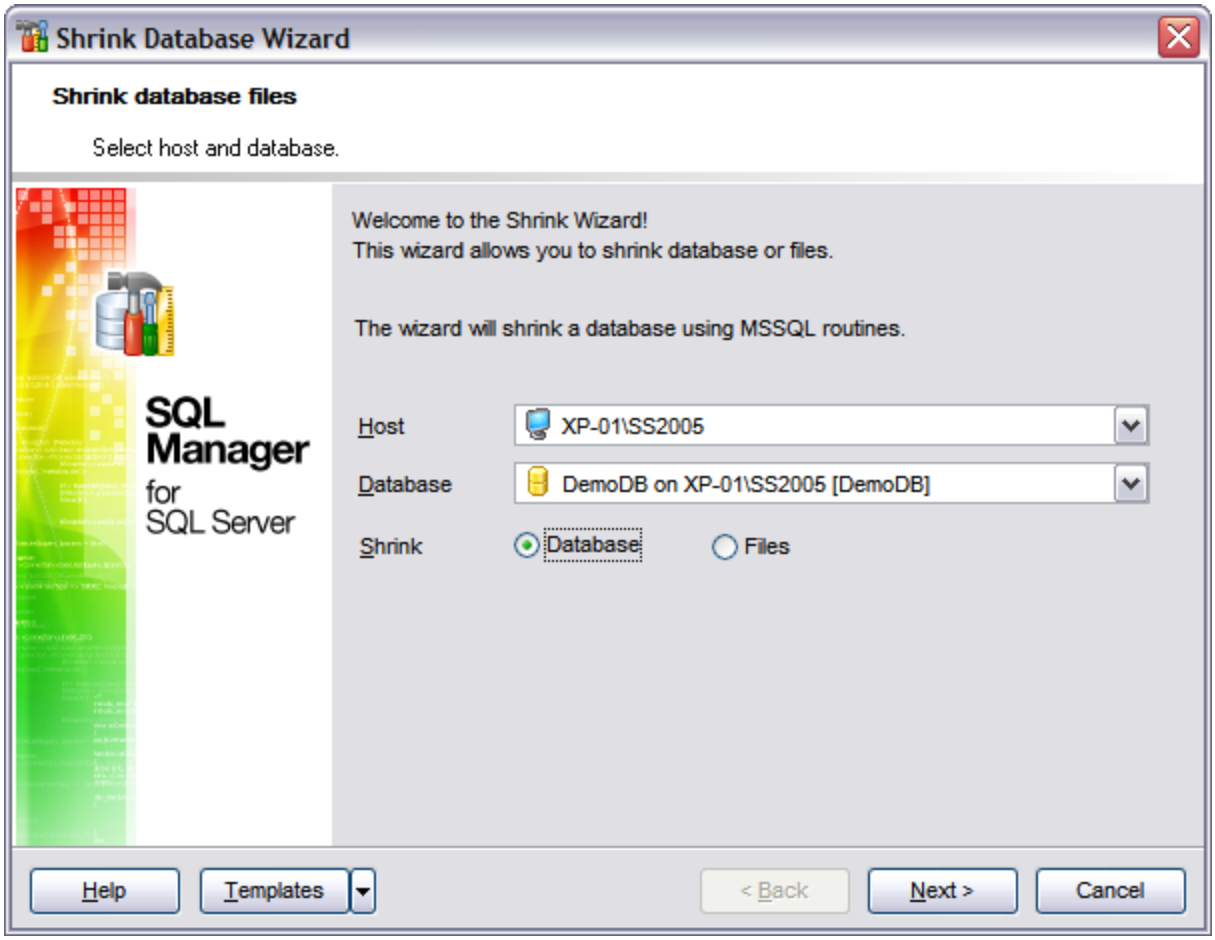
Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

Attach Database
Detach Database
Backup Database
Restore Database
Indices Management
Update Statistics
Check Database
Grant Manager

9.2.5.1 Selecting database and setting shrink mode

This step of the wizard allows you to specify the **host/instance** containing the database for shrinking, and to select the **shrink mode**.



© 2008 EMS Database Management Solutions

Host
Type in or use the drop-down list to select the host/instance where the database resides.

Database
Use the drop-down list of all registered databases on the selected host to specify the database to shrink.

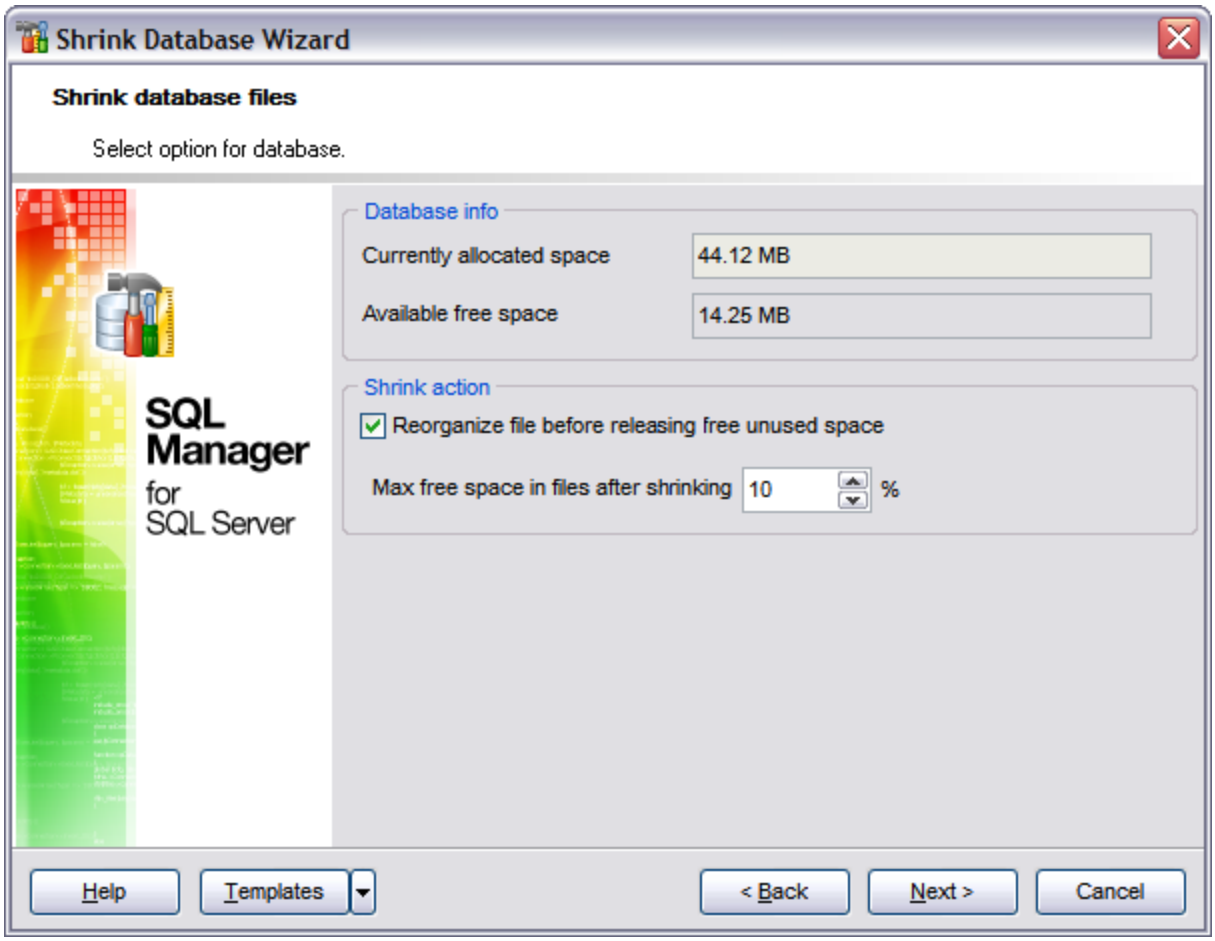
Shrink
Select the preferable shrink mode: **Database** (*SHRINKDATABASE*) or **Files** (*SHRINKFILE*).

Click the **Next** button to proceed to the Defining shrink action for database step of the wizard (if you have selected the *Database* shrink mode), or to the Defining shrink action for files step (if you have selected the *Files* shrink mode).

9.2.5.2 Defining shrink action for database

This step of the wizard provides you with information on the **allocated** and **free space**, and allows you to specify the **shrink action** for the database.

© 2008 EMS Database Management Solutions



Database info

This group displays the *currently allocated space* and the *available free space* values pertaining to the database being shrunk.

Shrink action

Reorganize file before releasing free unused space
This action causes the file to be reorganized before free space is released.

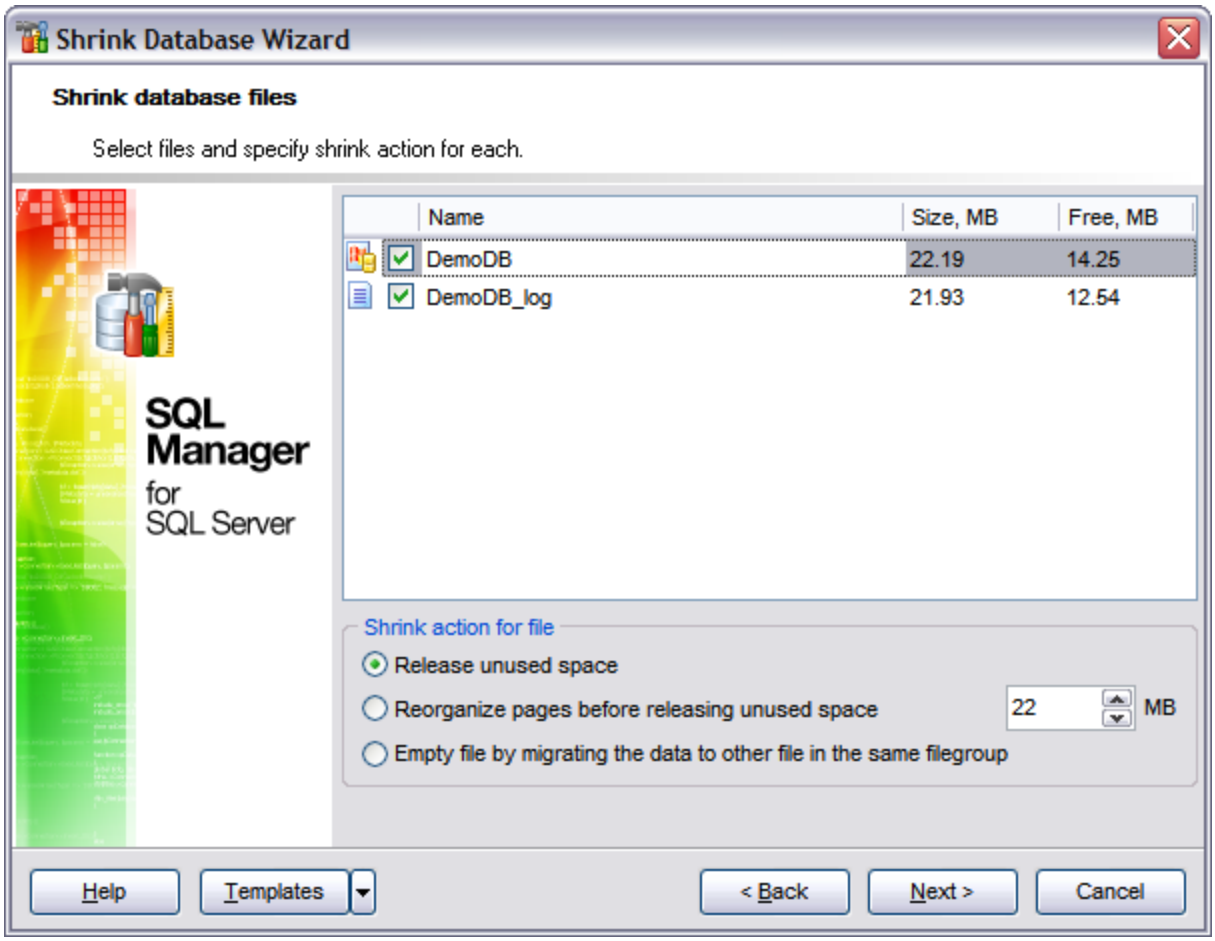
Max free space in files after shrinking
Specify the percentage of free space that will be left in the database file after the database is shrunk.

When you are done, click the **Next** button to proceed to the Shrinking database step of the wizard.

© 2008 EMS Database Management Solutions

9.2.5.3 Defining shrink action for files

This step of the wizard provides you with information on the **allocated** and **free space**, and allows you to specify the **shrink action** for each of the database files.



Shrink action for file

Release unused space
The free unused space is released explicitly (default).

Reorganize pages before releasing unused space
This action causes the pages to be reorganized before unused space is released.

Empty file by migrating the data to other file in the same filegroup
This action migrates all data from the selected file to other files in the same filegroup. In this case placing data on the file is not allowed.

If necessary, you can change the default value of the *target_size* edit-box to enforce shrinking the file to the specified size.

When you are done, click the **Next** button to proceed to the Shrinking database step of

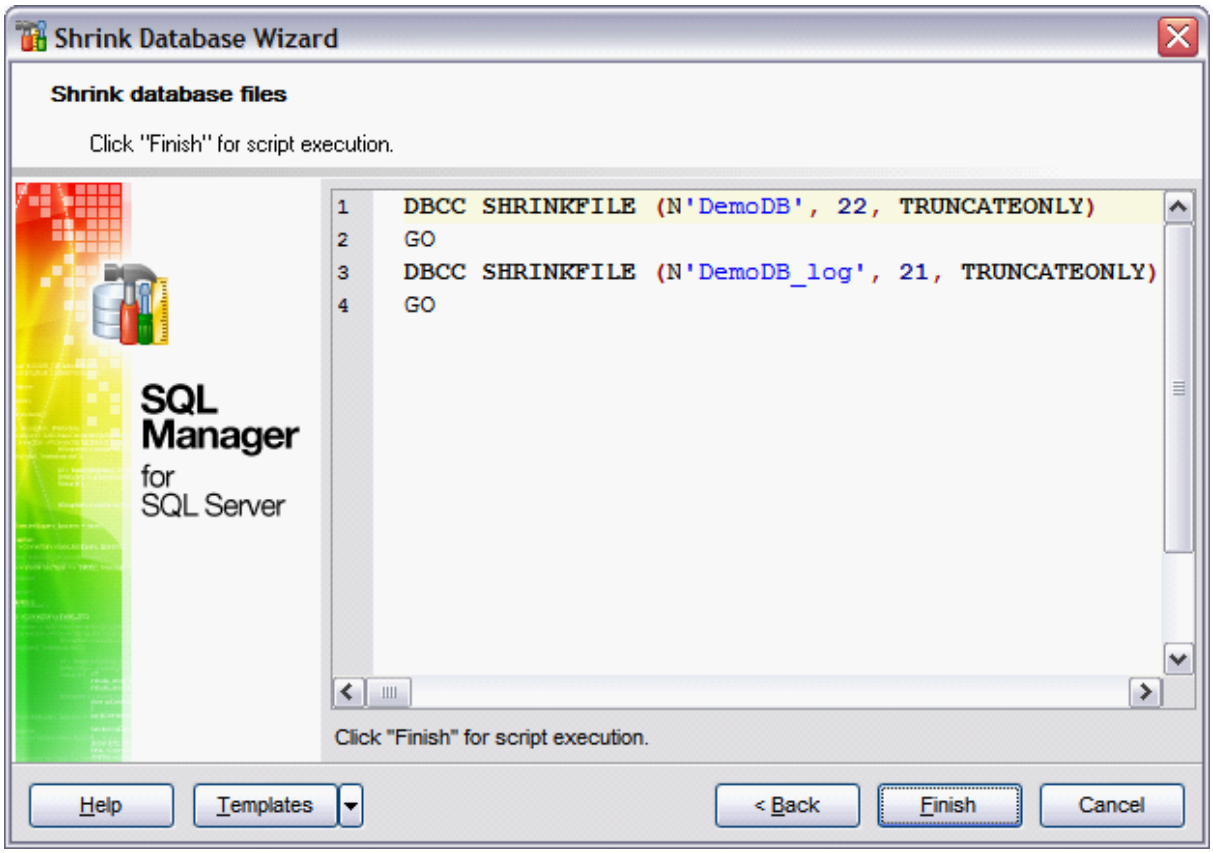
© 2008 EMS Database Management Solutions

the wizard.

9.2.5.4 Shrinking database

This step of the wizard is intended to inform you that all necessary options have been set, and you can start the shrink database/files process.

If necessary, you can edit the result script before execution.



If necessary, you can save a template for future use.

Click the **Finish** button to execute the result script and complete the operation.

9.2.6 Indices Management

Indices Management Wizard allows you to perform index management operations in your SQL Server database: *reorganizing*, *rebuilding*, *disabling* (the *ALTER INDEX REORGANIZE... REBUILD... DISABLE* Transact-SQL statements are issued).

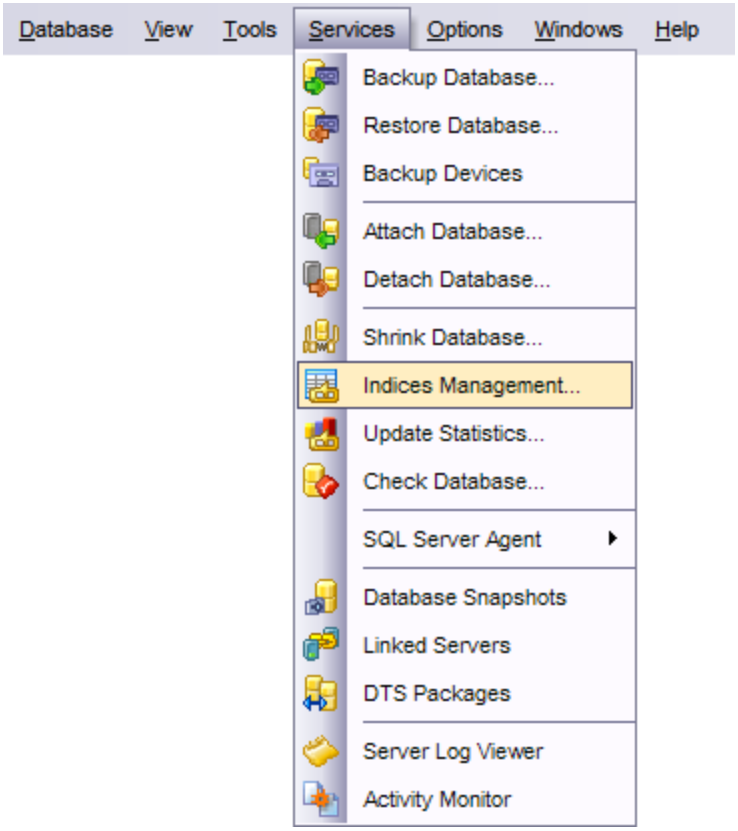
When indexes have pages in which the logical ordering based on the key value, does not match the physical ordering inside the data file, these indexes are called *fragmented*.

© 2008 EMS Database Management Solutions

Highly fragmented indexes may considerably decrease query performance. Index fragmentation is remedied by either reorganizing or by rebuilding an index.

To run the wizard, select the **Services | Indices Management...** main menu item, or right-click the database alias in the DB Explorer tree and select the **Tasks | Indices Management...** item of the context menu.

- Selecting database for indices management
- Selecting tables that contain indexes
- Specifying actions and setting options
- Editing and executing the result script
- Using templates



Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**No**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

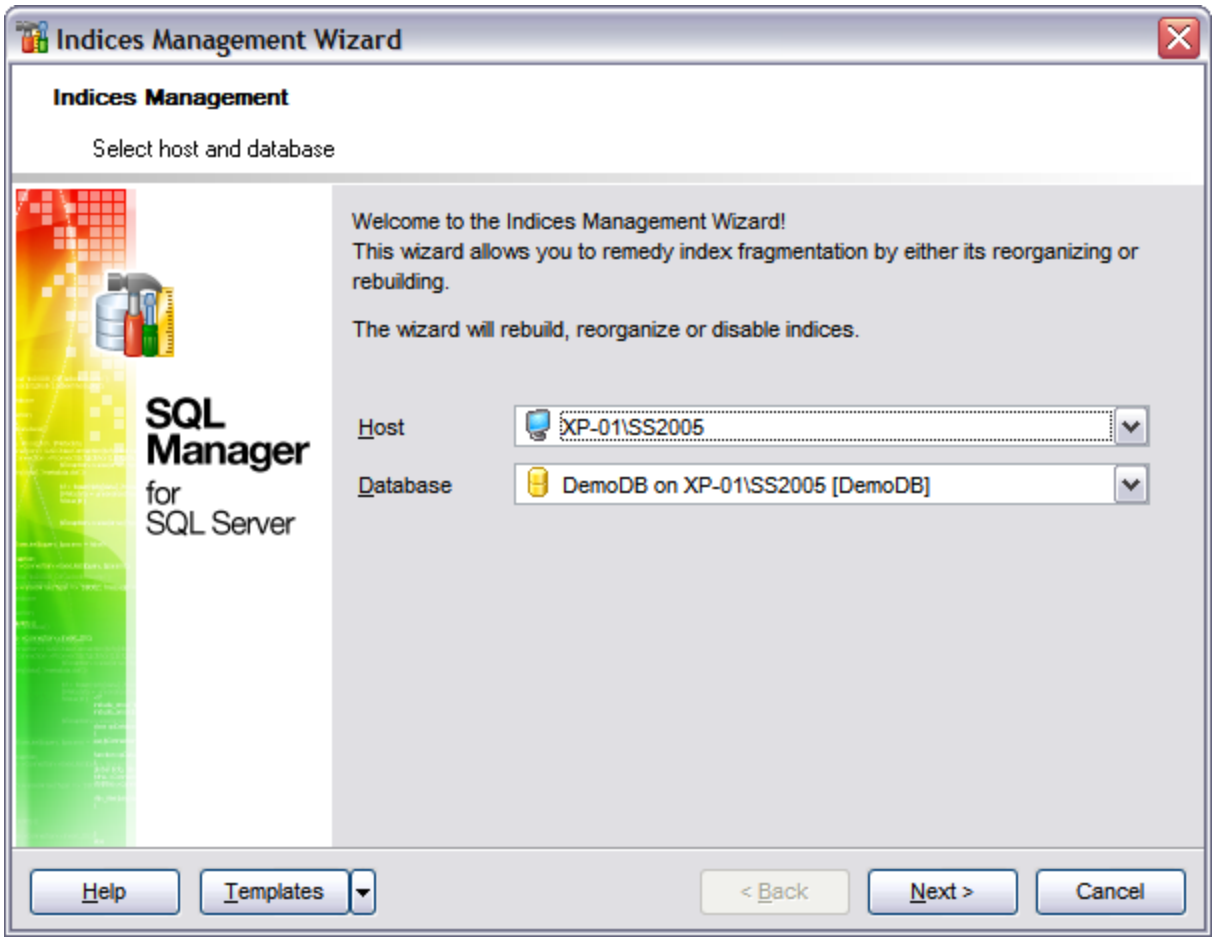
See also:

© 2008 EMS Database Management Solutions

Attach Database
Detach Database
Backup Database
Restore Database
Shrink Database
Update Statistics
Check Database
Grant Manager

9.2.6.1 Selecting database for indices management

This step of the wizard allows you to specify the **host/instance** containing the database for indexes management.



Host
Type in or use the drop-down list to select the host/instance where the database resides.

Database
Use the drop-down list of all registered databases on the selected host to specify the

© 2008 EMS Database Management Solutions

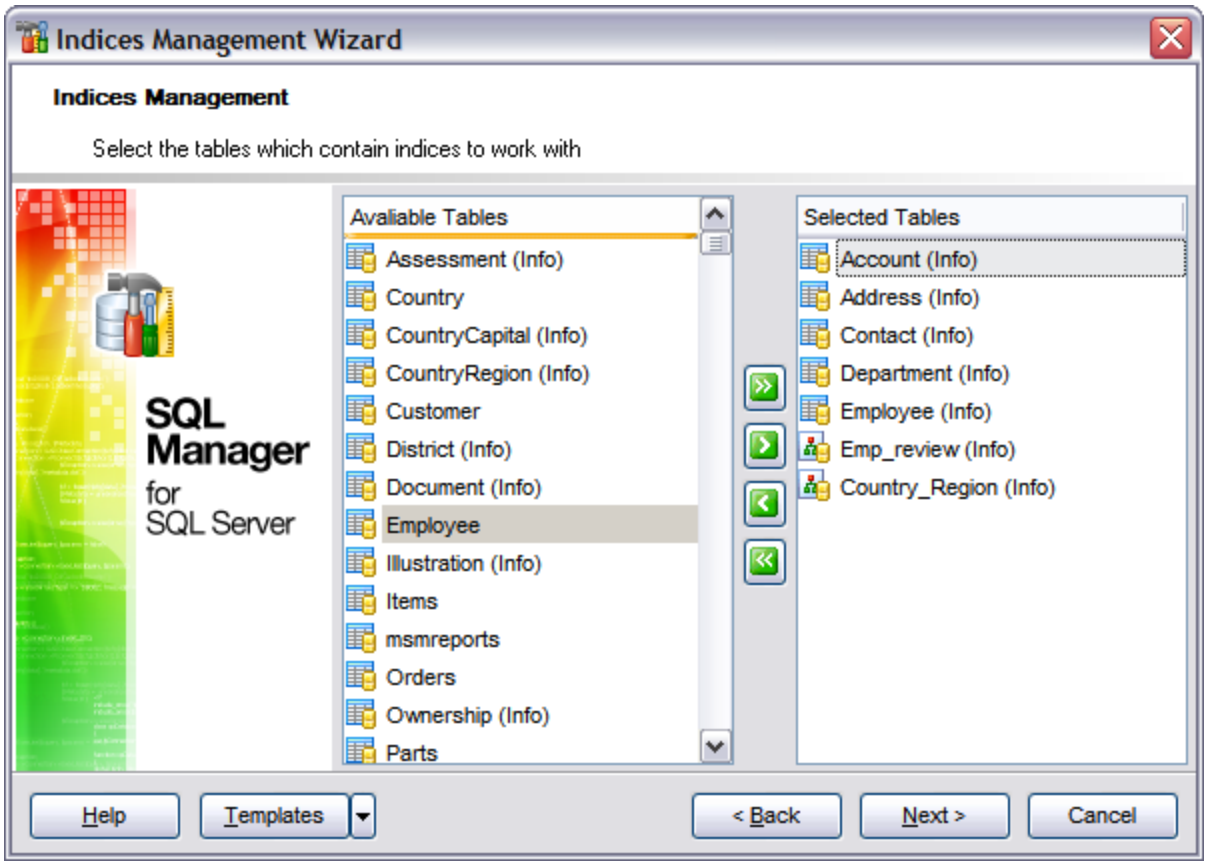
database to manage indexes in.

Click the **Next** button to proceed to the Selecting tables step of the wizard.

9.2.6.2 Selecting tables

This step of the wizard allows you to **select tables** containing indexes.

To select a table, you need to move it from the **Available Tables** list to the **Selected Tables** list. Use the buttons or drag-and-drop operations to move the tables from one list to another.



Click the **Next** button to proceed to the Specifying actions and setting options step of the wizard.

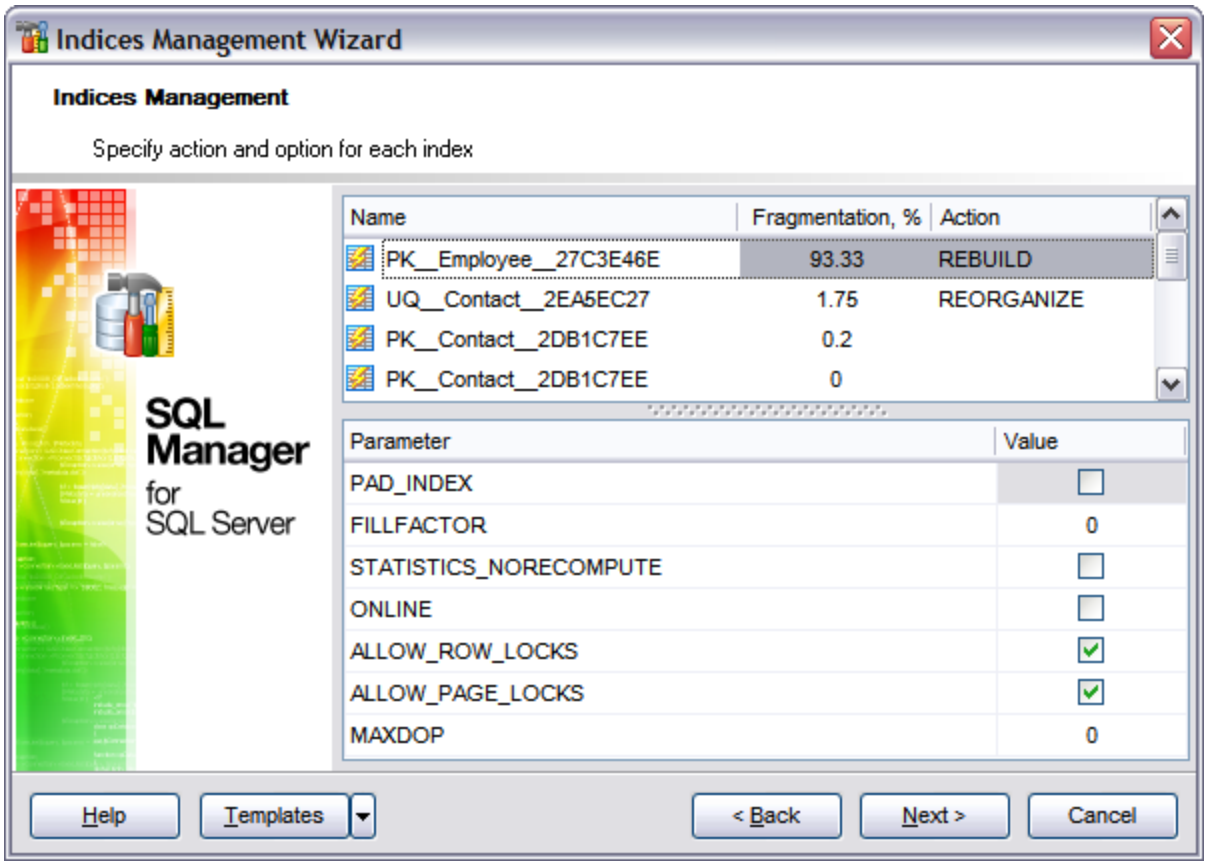
9.2.6.3 Specifying actions and setting options

Use this step of the wizard to specify an **action** for each index and set the respective **parameters**.

© 2008 EMS Database Management Solutions

The upper area lists all **indexes** of the tables selected at the previous step.

The **Fragmentation** column indicates the current fragmentation percentage of each index (*avg_fragmentation_in_percent*);
the **Action** column allows you to specify one of the available actions to be performed over the selected index: *REBUILD*, *REORGANIZE*, *DISABLE*.



The lower area lists the **parameters** that can be defined for the specified action (except for *DISABLE*).

- **REBUILD**:

PAD_INDEX
This parameter sets the percentage of free space in the intermediate level pages during index creation.

FILLFACTOR
This parameter sets the percentage of free space in the leaf level of each index page during index creation.

STATISTICS_NORECOMPUTE
This option specifies whether out-of-date index statistics should be automatically recomputed.

ONLINE
This option determines whether concurrent user access to the underlying table or

© 2008 EMS Database Management Solutions

clustered index data and any associated non-clustered indexes is allowed during index operations.

ALLOW_ROW_LOCKS
This option determines whether row locks are used in accessing index data.

ALLOW_PAGE_LOCKS
This option determines whether page locks are used in accessing index data.

MAXDOP
This parameter sets the maximum number of processors the query processor can use to execute a single index statement. Fewer processors may be used depending on the current system workload.

- **REORGANIZE**:

LOB_COMPARISON
Besides reorganizing one or more indexes, large object data types (LOBs) that are contained in the clustered index or underlying table can be compacted when an index is reorganized. The data types *image*, *text*, *ntext*, *varchar(max)*, *nvarchar(max)*, *varbinary(max)*, and *xml* are large object data types. Compacting this data can cause better disk space use.

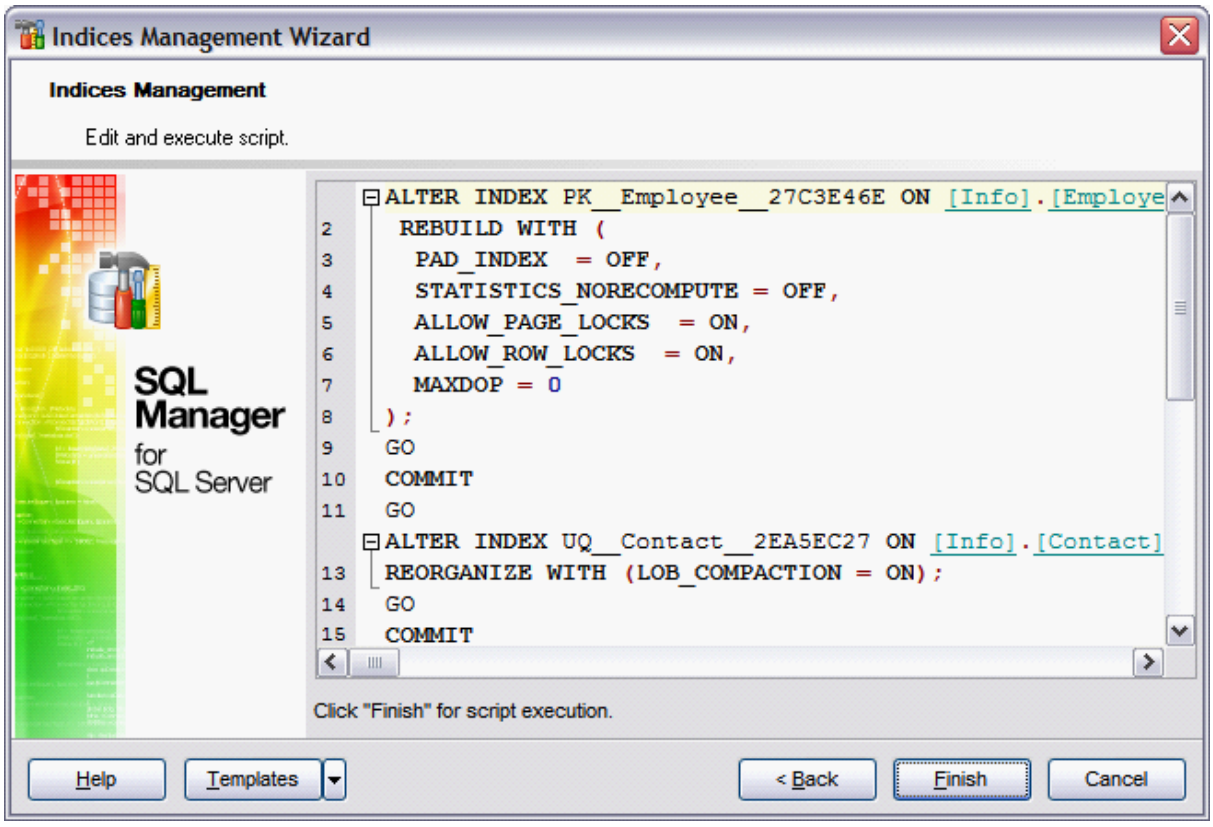
When you are done, click the **Next** button to proceed to Editing and executing the result script.

9.2.6.4 Editing and executing the result script

This step of the wizard is intended to inform you that all necessary options have been set, and you can start the indexes management process.

If necessary, you can edit the result script before execution.

© 2008 EMS Database Management Solutions



If necessary, you can save a template for future use.

Click the **Finish** button to execute the result script and complete the operation.

9.2.7 Update Statistics

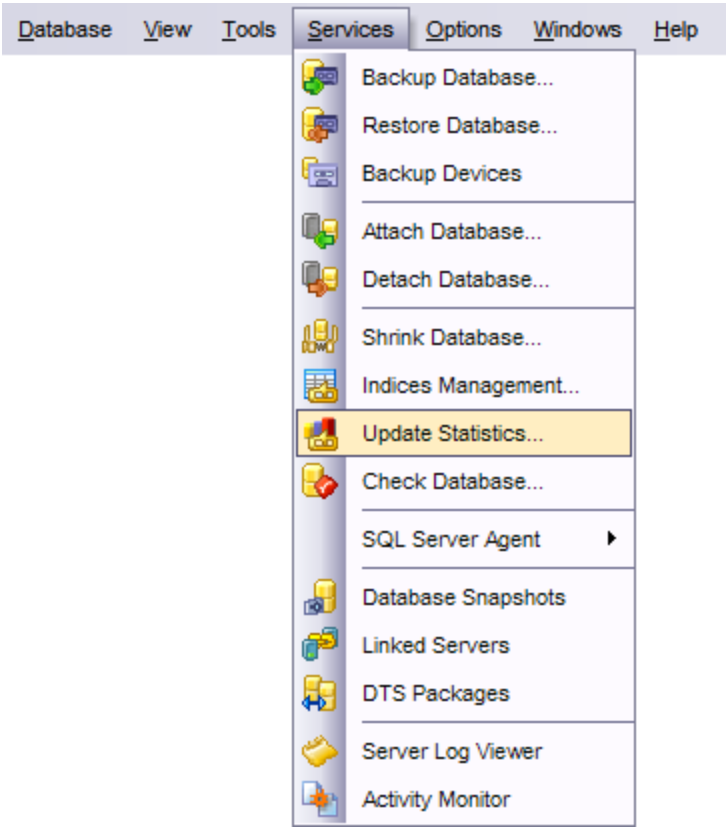
Statistics Update Wizard allows you to perform the update statistics operation in your SQL Server database (the *UPDATE STATISTICS* statement is issued).

This operation is used to updates the statistics for the specified table(s) and view(s).

To run the wizard, select the **Services | Update Statistics...** main menu item, or right-click the database alias in the DB Explorer tree and select the **Tasks | Update Statistics...** item of the context menu.

- Selecting database for statistics update
- Selecting tables and views
- Specifying update statistics options
- Editing and executing the result script
- Using templates

© 2008 EMS Database Management Solutions



Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**No**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

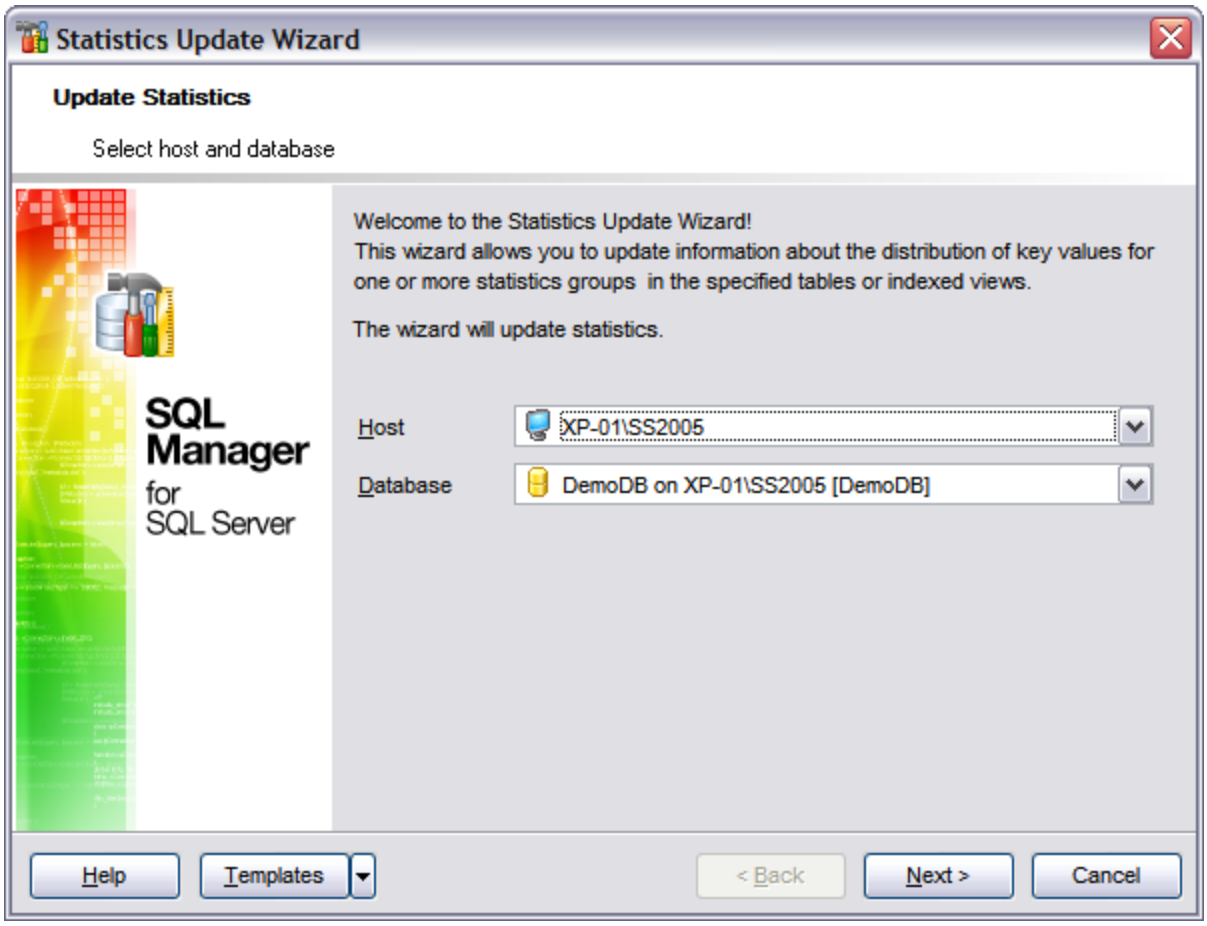
See also:

Attach Database
Detach Database
Backup Database
Restore Database
Shrink Database
Indices Management
Check Database
Grant Manager

© 2008 EMS Database Management Solutions

9.2.7.1 Selecting database for statistics update

This step of the wizard allows you to specify the **host/instance** containing the database for statistics update.



Host
Type in or use the drop-down list to select the host/instance where the database resides.

Database
Use the drop-down list of all registered databases on the selected host to specify the database to update statistics.

Click the **Next** button to proceed to the Selecting tables and views step of the wizard.

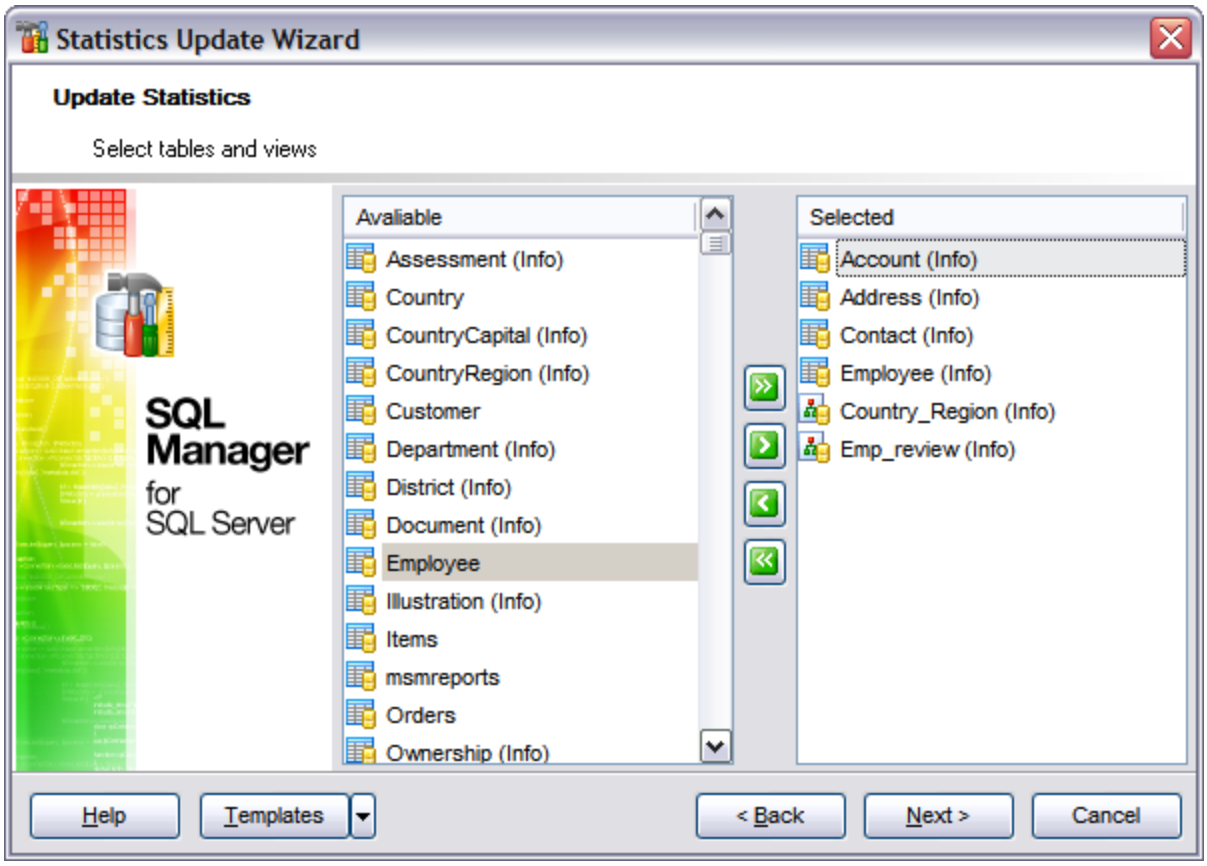
9.2.7.2 Selecting tables and views

This step of the wizard allows you to **select tables and views** to update statistics for.

To select a table/view, you need to move it from the **Available** list to the **Selected** list. Use the buttons or drag-and-drop operations to move the objects from

© 2008 EMS Database Management Solutions

one list to another.



Click the **Next** button to proceed to the Specifying update statistics options step of the wizard.

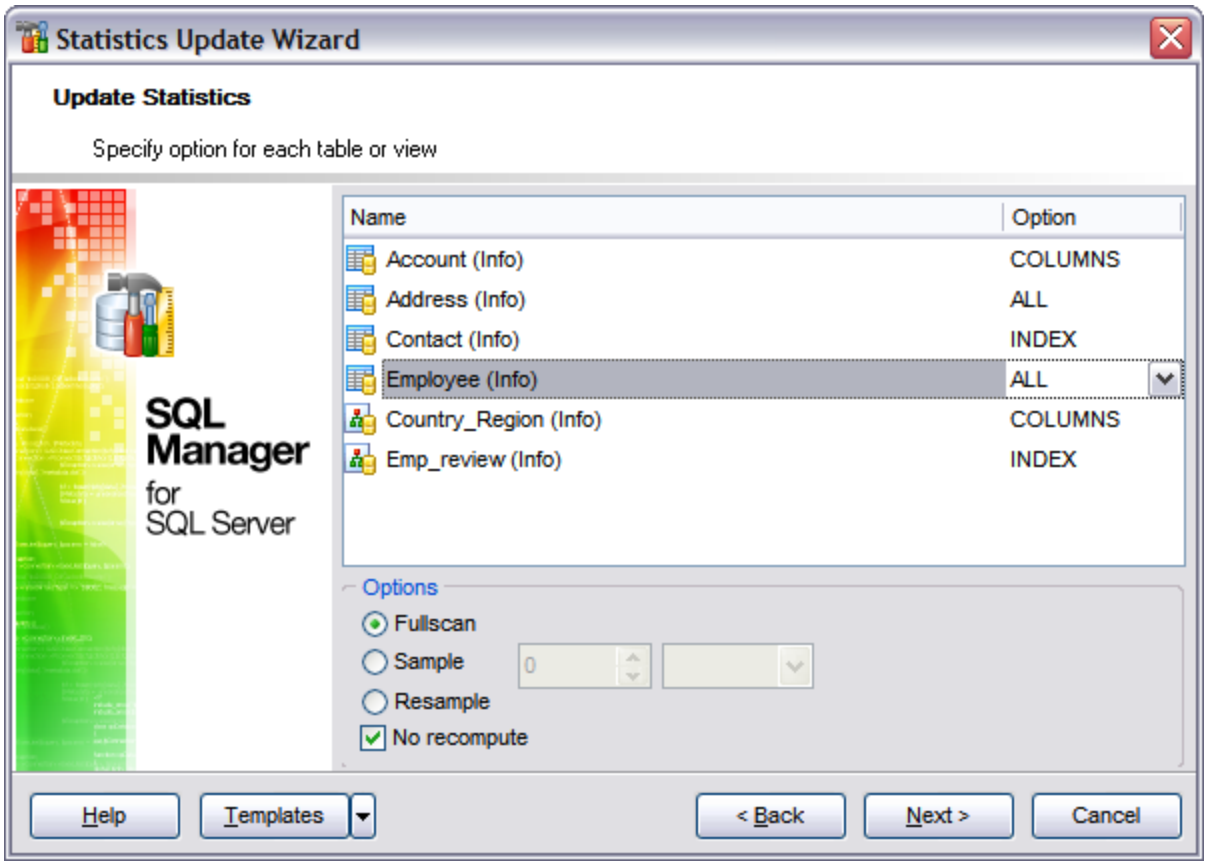
9.2.7.3 Specifying update statistics options

Use this step of the wizard to set **options** for each table/view.

The main area lists all **tables** and **views** selected at the previous step.

The **Option** column allows you to specify one of the available options to be applied to the selected object: *ALL*, *COLUMNS*, *INDEX*.

© 2008 EMS Database Management Solutions



Options

This group allows you to set update statistics options for tables.

Fullscan
Use this option to specify that all table rows should be read to gather the statistics. This option provides the same behavior as *SAMPLE 100 PERCENT*.

Sample
This option allows you to specify the percentage of the table, or the number of rows to sample when collecting statistics for each table. When *0 PERCENT* or *ROWS* is specified, the result is an empty statistics set.

Resample
Select this option to specify that statistics will be gathered using an inherited sampling ratio for all existing statistics including indexes. If the sampling ratio creates too few rows being sampled, the Database Engine automatically corrects the sampling based on the number of existing rows in the table.

No recompute
This option specifies that automatic recomputation of the statistics that become out of date will be disabled.

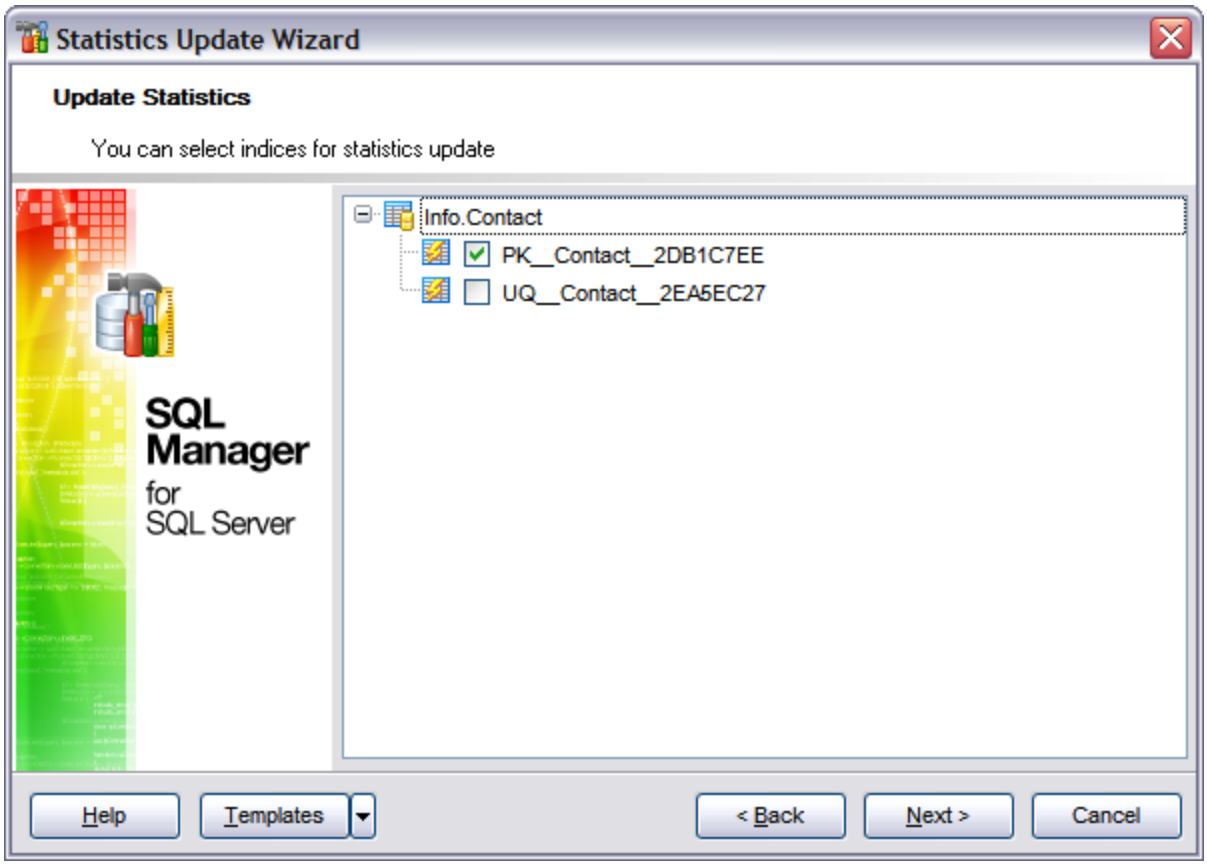
© 2008 EMS Database Management Solutions

Click the **Next** button to proceed to the Selecting indexes step of the wizard.

9.2.7.4 Selecting indexes

This step of the wizard allows you to **select indexes** to be included into the *UPDATE STATISTICS* statement.

For you convenience the indexes are represented in the form of a tree with objects-containers as tree nodes.



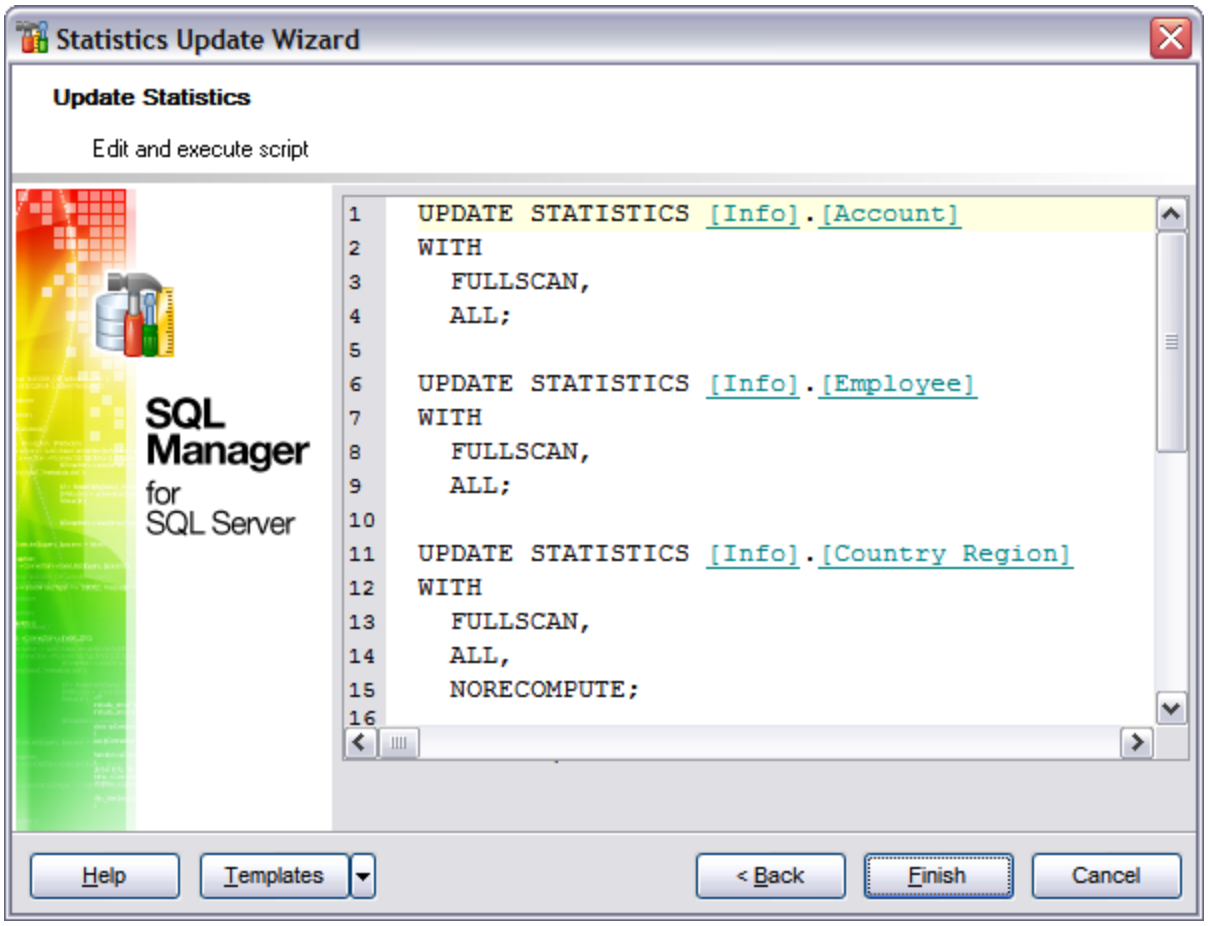
When you are done, click the **Next** button to proceed to Editing and executing the result script.

9.2.7.5 Editing and executing the result script

This step of the wizard is intended to inform you that all necessary options have been set, and you can start the update statistics process.

If necessary, you can edit the result script before execution.

© 2008 EMS Database Management Solutions



If necessary, you can save a template for future use.

Click the **Finish** button to execute the result script and complete the operation.

9.2.8 Check Database

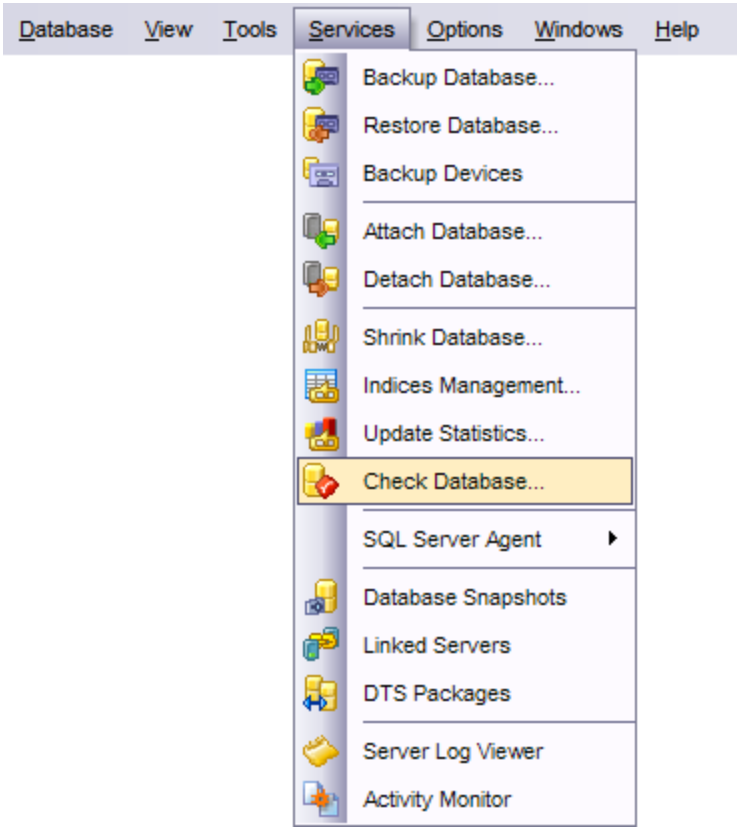
Check Database Wizard allows you to perform the check database operation on your SQL Server system (the *DBCC CHECKDB* Transact-SQL statement is issued).

This operation is used to check the allocation, structural, and logical integrity of all the objects in the specified database.

To run the wizard, select the **Services | Check Database...** main menu item, or right-click the database alias in the DB Explorer tree and select the **Tasks | Check Database...** item of the context menu.

- Selecting database to check
- Specifying arguments and options
- Editing and executing the result script
- Using templates

© 2008 EMS Database Management Solutions



Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**No**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

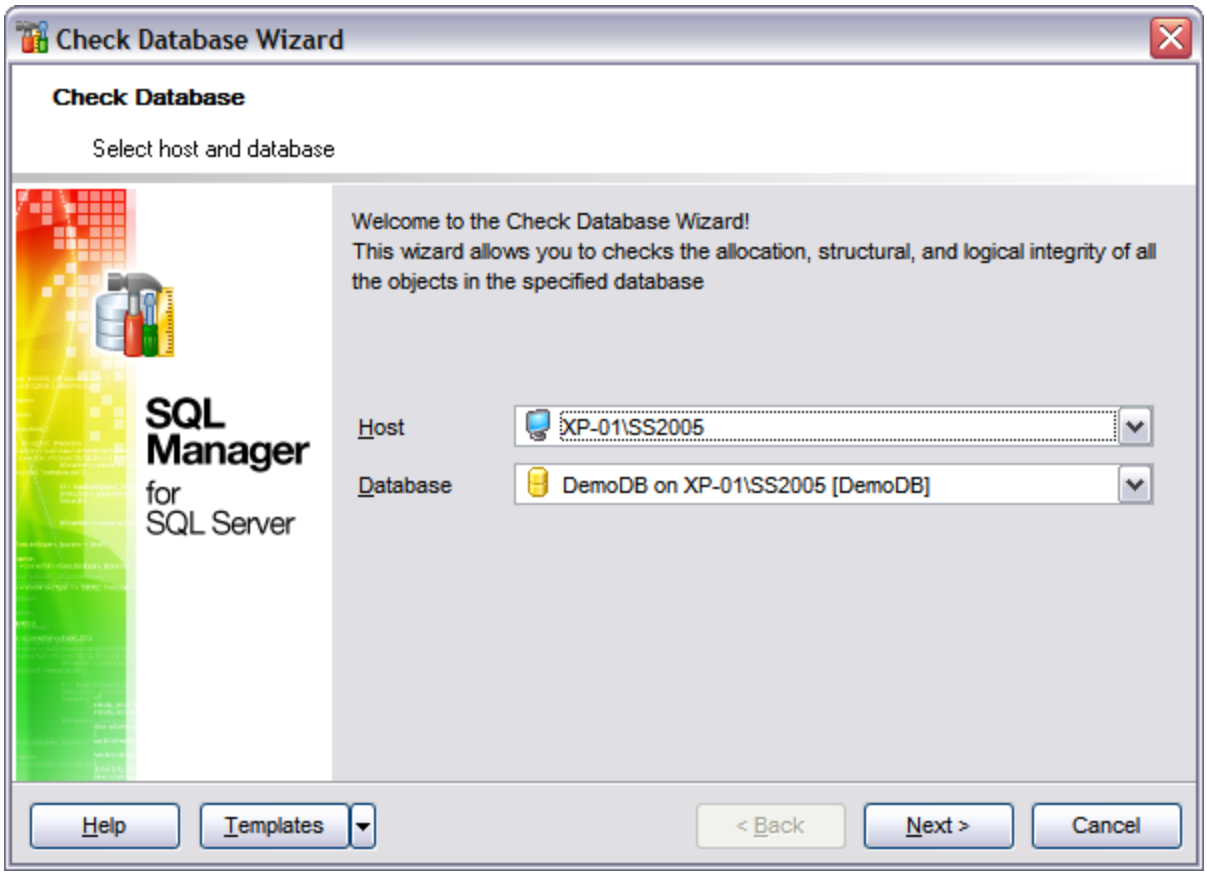
See also:

Attach Database
Detach Database
Backup Database
Restore Database
Shrink Database
Indices Management
Update Statistics
Grant Manager

© 2008 EMS Database Management Solutions

9.2.8.1 Selecting database to check

This step of the wizard allows you to select the **host/instance** containing the database for checking.



Host
Type in or use the drop-down list to select the host/instance where the database resides.

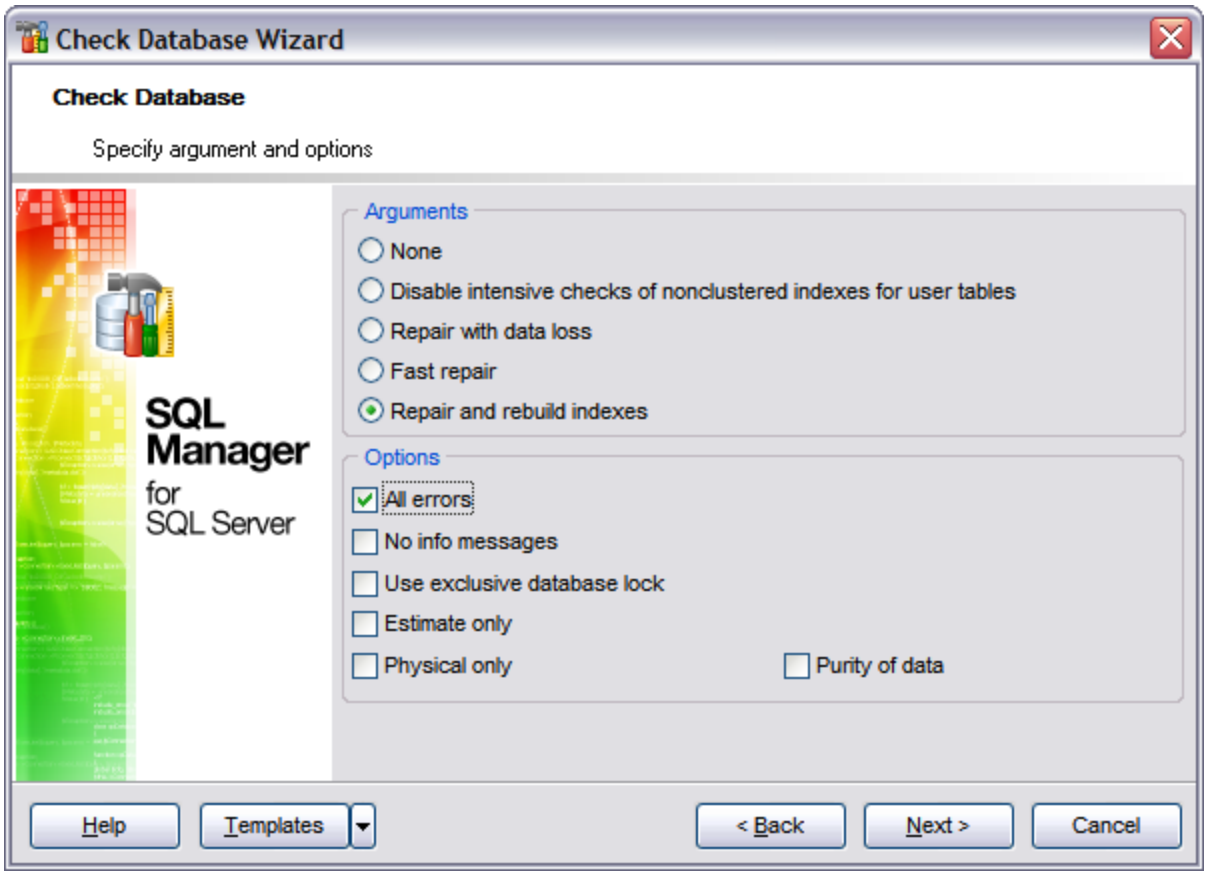
Database
Use the drop-down list of all registered databases on the selected host to specify the database to run integrity checks for.

Click the **Next** button to proceed to the Specifying arguments and options step of the wizard.

9.2.8.2 Specifying arguments and options

Use this step of the wizard to **specify arguments** and **options** for the check database operation.

© 2008 EMS Database Management Solutions



Arguments

Disable intensive checks of nonclustered indexes for user tables
This argument (*NOINDEX*) specifies that intensive checks of non-clustered indexes for user tables should not be performed; therefore, the overall execution time is decreased.

Repair with data loss
When this argument (*REPAIR_ALLOW_DATA_LOSS*) is used, all reported errors are repaired. These repairs can cause data loss.

Fast repair
This argument (*REPAIR_FAST*) enables fast repair of errors which is maintained for backward compatibility.

Repair and rebuild indexes
This argument (*REPAIR_REBUILD*) performs all repairs performed by *REPAIR_FAST* and includes time-consuming repairs such as rebuilding indexes. These repairs can be performed without risk of data loss.

Options

All errors

© 2008 EMS Database Management Solutions

This option (*ALL_ERRORMSGS*) displays an unlimited number of errors per object.

No info messages
This option (*NO_INFOMSGS*) suppresses all informational messages.

Use exclusive database lock
If this option (*TABLOCK*) is used, the *DBCC CHECKDB* statement obtains locks instead of using an internal database snapshot. This includes a short-term database exclusive lock.

Estimate only
This option (*ESTIMATEONLY*) displays the estimated amount of *tempdb* space that is required to run DBCC CHECKDB with all the other specified options. The actual database check is not performed in this case.

Physical only
This option (*PHYSICAL_ONLY*) limits the checking operation to the integrity of the physical structure of the page and record headers, the physical structure of B-trees and the allocation consistency of the database. Designed to provide a small overhead check of the physical consistency of the database, this check can also detect torn pages, checksum failures, and common hardware failures that can compromise a user's data.

Purity of data
If this option (*DATA_PURITY*) is enabled, the *DBCC CHECKDB* statement checks the database for column values that are not valid or out-of-range.

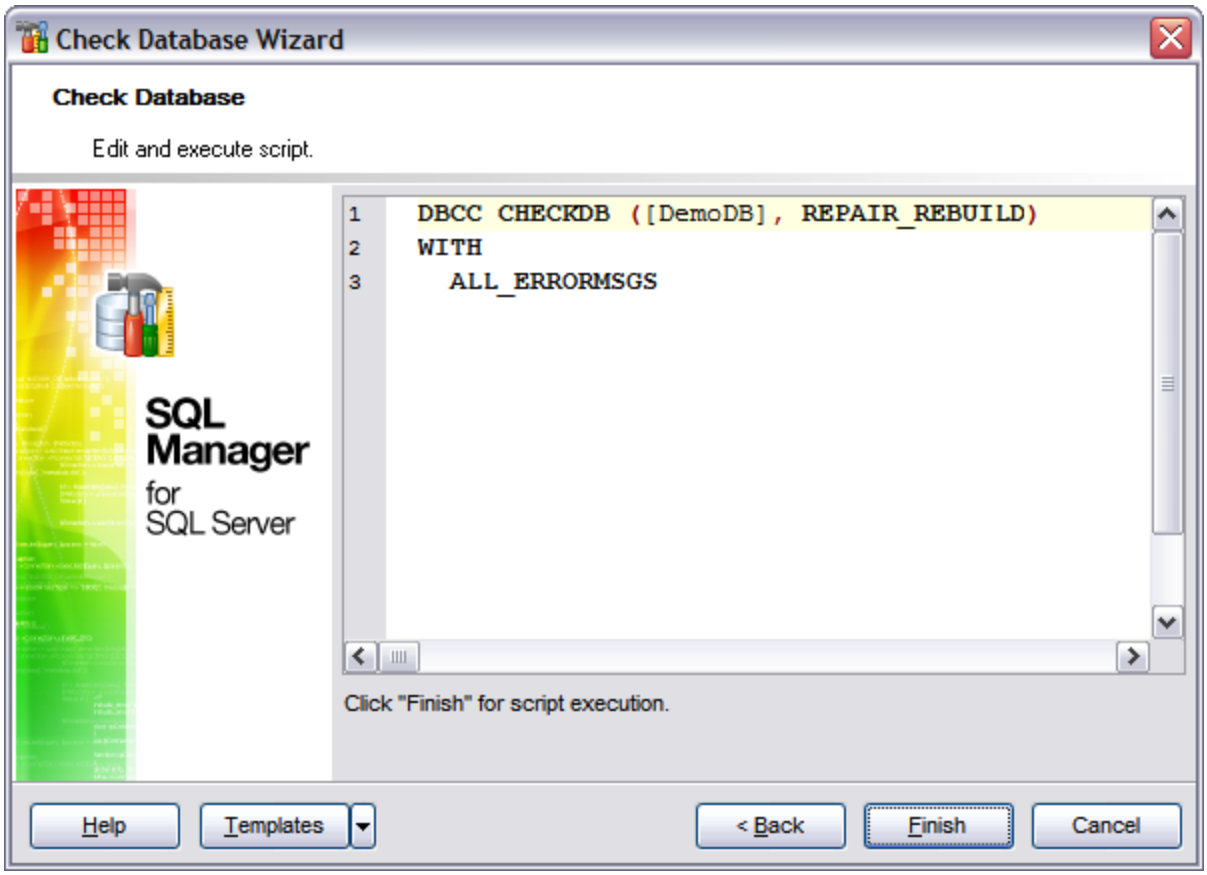
When you are done, click the **Next** button to proceed to Editing and executing the result script.

9.2.8.3 Editing and executing the result script

This step of the wizard is intended to inform you that all necessary options have been set, and you can start the check database process.

If necessary, you can edit the result script before execution.

© 2008 EMS Database Management Solutions



If necessary, you can save a template for future use.

Click the **Finish** button to execute the result script and complete the operation.

9.2.9 Grant Manager

Grant Manager allows you to set the user access grants for certain database objects: schemas, tables, views, procedures, synonyms, aggregates, XML schema collections, assemblies, etc.

'Global' user privileges define the user access rights to all the database objects on the server. Granting privileges on the selected database objects allows a user to perform the defined operation over the selected objects.

To open **Grant Manager**, select the **Tools | Grant Manager** main menu item.

- Using Navigation bar and context menu
- Managing database-specific privileges
- Managing column permissions
- Viewing effective permissions

© 2008 EMS Database Management Solutions

Managing grants

To define grants on database objects:

- select the **Tools | Grant Manager** main menu item, or use the corresponding toolbar button to open Grant Manager;
- select the object type using the drop-down list on the toolbar;
- select a User or Role from the **Privileges for** pane of the Navigation bar;
- edit user/role privileges using Grant Manager

or

- right-click an object within the **Users**/**Roles** node in DB Explorer and select the **Grants for <object_name>** context menu item;
- edit user/role privileges using Grant Manager

Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**Yes**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

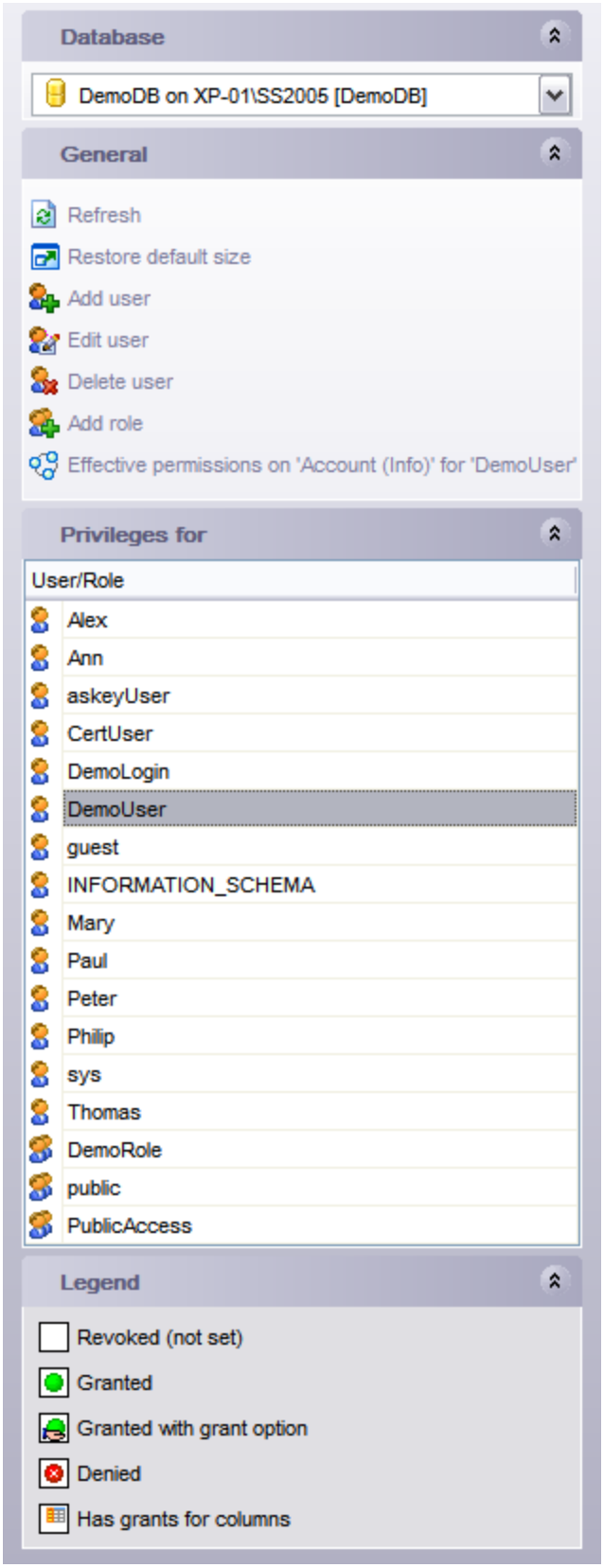
Attach Database
Detach Database
Backup Database
Restore Database
Shrink Database
Indices Management
Update Statistics
Check Database

9.2.9.1 Using Navigation bar and context menu

The **Navigation bar** of **Grant Manager** allows you to:

- select the database for grants management;
- refresh the content of the window;
- call User Editor to add a new database user;
- call User Editor to edit an existing user;
- delete a database user;
- call Role Editor to add a database role;
- view effective permissions on the object;
- select an existing database user/role to grant privileges for;
- view the legend;
- restore the default size of the window.

© 2008 EMS Database Management Solutions

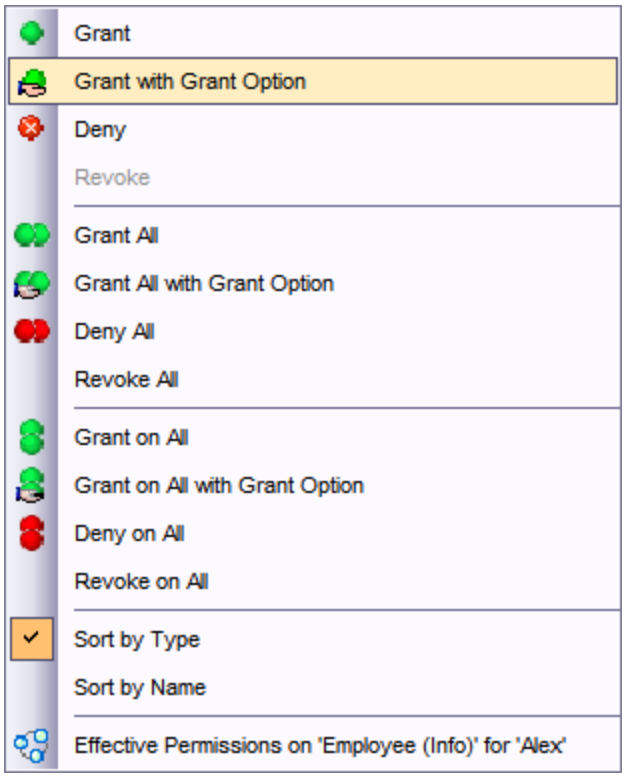


© 2008 EMS Database Management Solutions

The **context menu** is aimed at facilitating your work: you can perform a variety of operations using context menu items.

The **context menu** of **Grant Manager** allows you to:

- grant a permission on a securable to the selected principal;
- grant a permission (with Grant Option) on a securable to the selected principal;
- deny a permission on a securable to the selected principal;
- revoke a previously granted or denied permission;
- grant all permissions on a securable to the selected principal;
- grant all permissions (with Grant Option) on a securable to the selected principal;
- deny all permissions on a securable to the selected principal;
- revoke all previously granted or denied permissions on a securable;
- grant a permission on all objects to the selected principal;
- grant a permission (with Grant Option) on all objects to the selected principal;
- deny a permission on all objects to the selected principal;
- revoke a previously granted or denied permission on all objects;
- apply sorting of objects in grid (by type);
- apply sorting of objects in grid (by name);
- view effective permissions on the object.



See also:

Managing database-specific privileges
Managing column permissions
Viewing effective permissions

© 2008 EMS Database Management Solutions

9.2.9.2 Managing database-specific privileges

This window allows you to define privileges on database objects and grant privileges to a user or role.

To edit the privileges of a user/role on an object of a database, select the database using the **Database** pane of the Navigation bar, then select a *user* or *role* from the **Privileges for** list available within the Navigation bar. Then select the type of objects to be displayed in the main working window using the combo-box at the top.

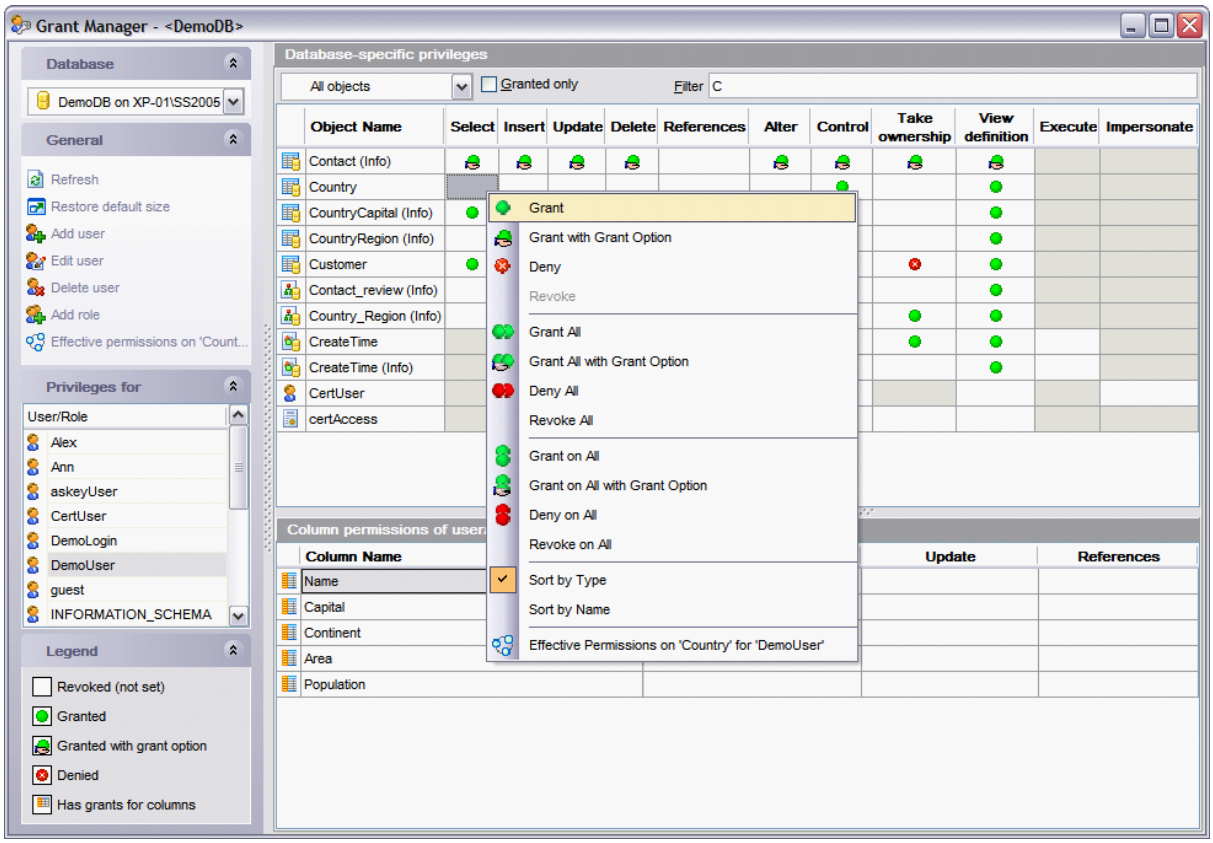
The **Object Name** column contains the list of the objects of the selected type; each subsequent column corresponds to the permission which can be granted on the selected object: *Select*, *Insert*, *Update*, *Delete*, *References*, *Alter*, *Control*, *Take ownership*, *View definition*, *Execute*, *Impersonate*.

The list of the objects can be configured in several ways: you can specify that only *granted* objects are displayed in the grid, or define an object name to *filter* the objects by that name.

Right-click a cell to grant a specific permission on a certain object. To grant a permission on an object, you should find the object in the **Object Name** list and the column with the corresponding permission. Note that the cells that are highlighted gray do not admit to setting grants for an obvious reason (e.g. you cannot execute a table). The context menu of a cell contains possible permissions that can be granted:

- *Grant*
- *Grant with Grant Option*
- *Deny* (prevents the principal from inheriting the permission through its group or role memberships)
- *Revoke* (removes a previously granted or denied permission)
- *Grant All*
- *Grant All with Grant Option*
- *Deny All*
- *Revoke All*
- *Grant on All*
- *Grant on All with Grant Option*
- *Deny on All*
- *Revoke on All*

© 2008 EMS Database Management Solutions



Check the **Granted only** option to display objects with at least one granted operation.

See also:

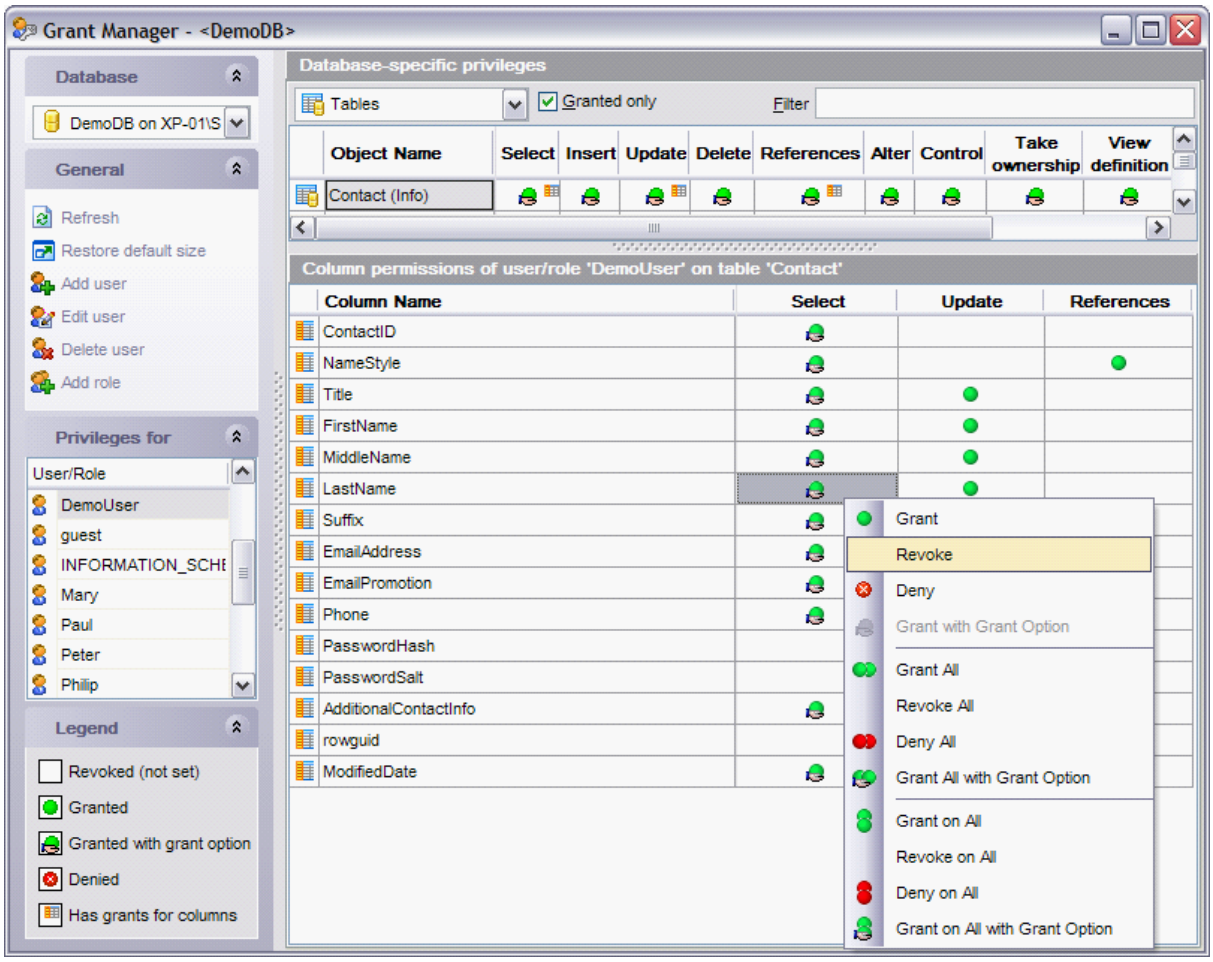
Using Navigation bar and context menu
Managing column permissions
Viewing effective permissions

9.2.9.3 Managing column permissions

The **Column permissions of user/role** area displays the grid with table columns and the privileges that can be granted to the selected user or role.

Use items of the context menu to grant/deny/revoke permissions on columns.

© 2008 EMS Database Management Solutions



If permissions on a column have been defined (for a table or view), the corresponding permission cell of the table/view contains a specific icon .

See also:

Using Navigation bar and context menu
Managing database-specific privileges
Viewing effective permissions

9.2.9.4 Viewing effective permissions

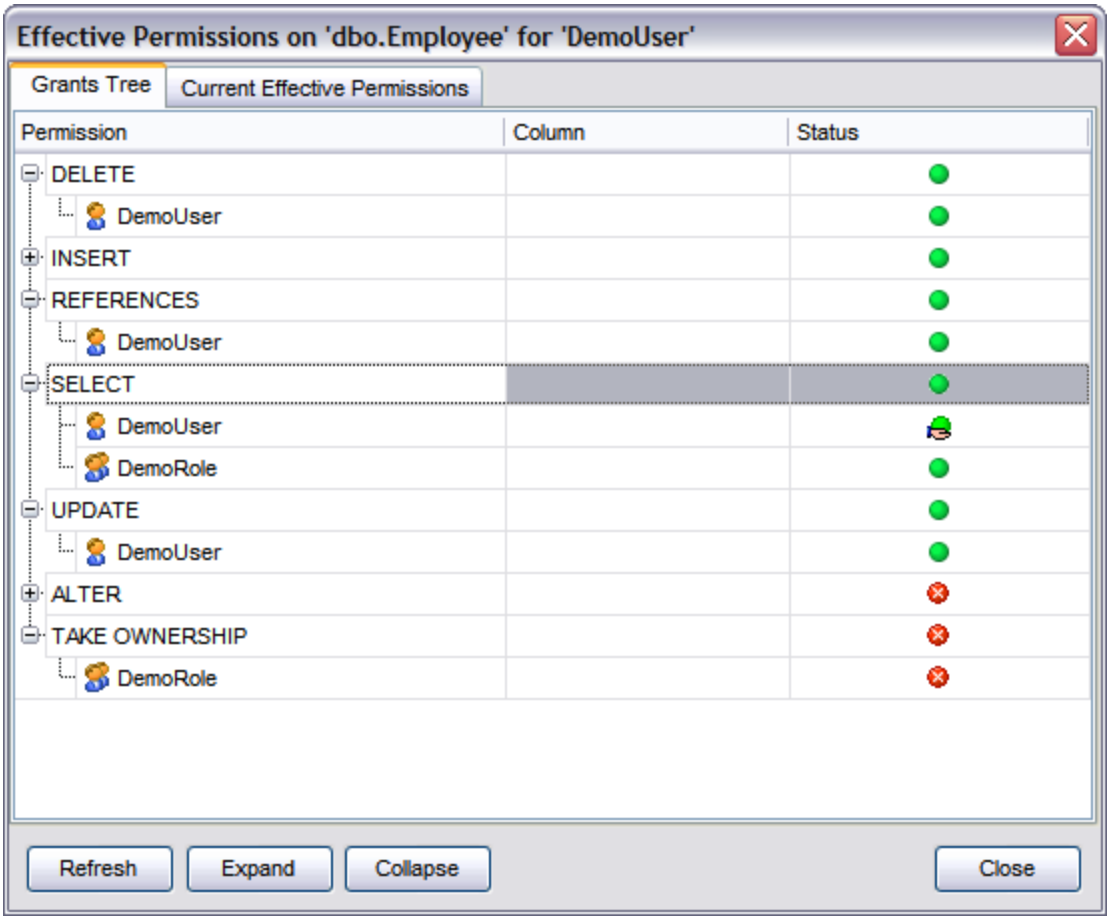
An **effective permission** is the one which is:

- granted directly to the principal, and not denied, or
- implied by a higher-level permission held by the principal and not denied, or
- granted to or held by, a role or group of which the principal is a member, and not denied.

The **Grants Tree** tab of the **Effective Permissions** dialog displays the entire list of permissions granted to the user and its role on a definite object in the form of a tree. For

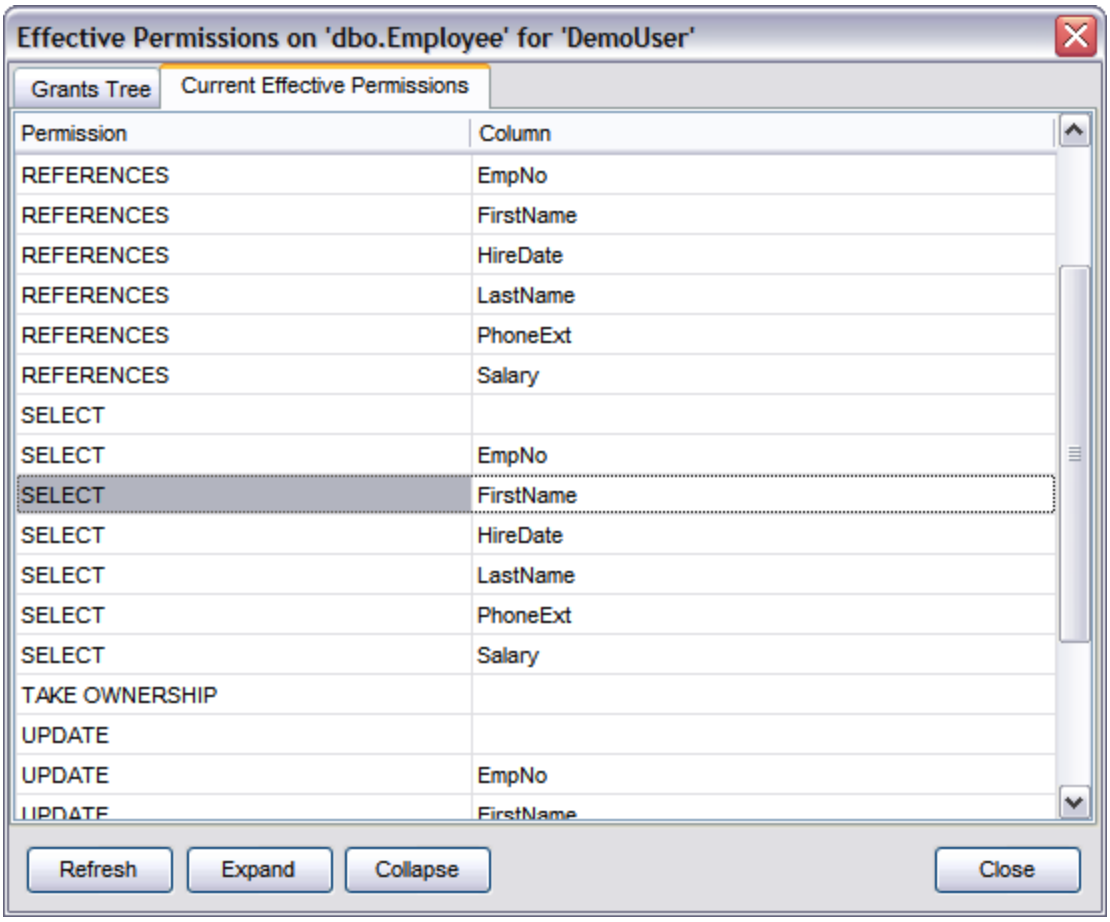
© 2008 EMS Database Management Solutions

your convenience the **Refresh**, **Expand**/**Collapse** buttons are available at the bottom of the dialog window.



The **Current Effective Permissions** tab of the **Effective Permissions** dialog displays a list of the permissions effectively granted to the specified principal on a definite object.

© 2008 EMS Database Management Solutions



See also:

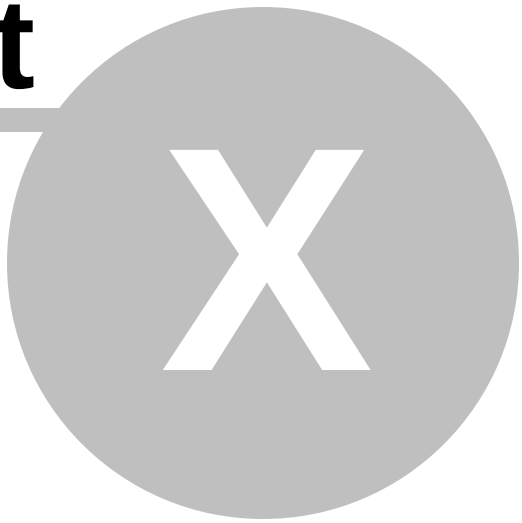
Using Navigation bar and context menu

Managing database-specific privileges

Managing column permissions

© 2008 EMS Database Management Solutions

Part



10 Server tools

SQL Manager 2008 for SQL Server provides a number of powerful tools for working with your database server.

- Server-level principals
- SQL Server objects and services

To obtain detailed information concerning SQL Server security system, objects and services, refer to the official Microsoft® SQL Server™ documentation.

See also:

Getting started
Database Explorer
Database management
Database objects management
Query management tools
Data management
Import/Export tools
Database tools
Personalization
External tools

© 2008 EMS Database Management Solutions

10.1 Server-level principals

Principals are individuals, groups, and processes that can request SQL Server resources.

SQL Manager 2008 for SQL Server provides management tools for the following server-level security principals:

- **Logins**
  Allows you to manage SQL Server logins.
- **Server Roles Manager**
  Allows you to manage SQL Server roles.

Adding logins

In order to add a new login:

- select the **Tools | Login Manager** main menu item or use the corresponding toolbar button to open Login Manager;
- select the **Add login...** item of the **context menu** or within the **Navigation bar**;
- define login properties and permissions using Login Editor

or

- right-click the **Logins** node (**Server Objects**) or any object within this node in DB Explorer and select the **New Login** context menu item, or use the *Ctrl+N* shortcut;
- define login properties and permissions using Login Editor.

Editing login properties

In order to edit an existing login:

- select the **Tools | Login Manager** main menu item or use the corresponding toolbar button to open Login Manager;
- select the **Edit login...** item of the **context menu** or within the **Navigation bar**;
- edit the login properties and permissions using Login Editor

or

- right-click the **Logins** node (**Server Objects**) or any object within this node in DB Explorer and select the **Edit Login <object_name>** context menu item, or use the *Ctrl+O* shortcut;
- edit the login properties and permissions using Login Editor.

Deleting logins

In order to delete an existing login:

- select the **Tools | Login Manager** main menu item or use the corresponding toolbar button to open Login Manager;
- right-click the login to delete and select the **Delete Login** item of the **context menu** or within the **Navigation bar**;
- confirm deleting in the dialog window

or

- right-click the login to delete within the **Logins** node (**Server Objects**) in DB Explorer and select the **Drop Login <object_name>** context menu item, or use the *Shift+Del* shortcut;
- confirm deleting in the dialog window

© 2008 EMS Database Management Solutions

Allocating server roles

In order to allocate server roles to logins:

- select the **Tools | Server Roles** main menu item or use the corresponding toolbar button;
- select a server role, and add logins to the Server Role Members list using Server Roles Manager.

See also:

SQL Server objects and services

10.1.1 Logins

Microsoft® SQL Server™ uses two ways to validate connections to SQL Server databases:
Windows Authentication which uses Windows-level principals
and
SQL Server Authentication which uses logins to validate the connection.

A **login** is an indivisible principal which is used to connect to the server and use its resources.

- Login Manager
- Login Editor

See also:

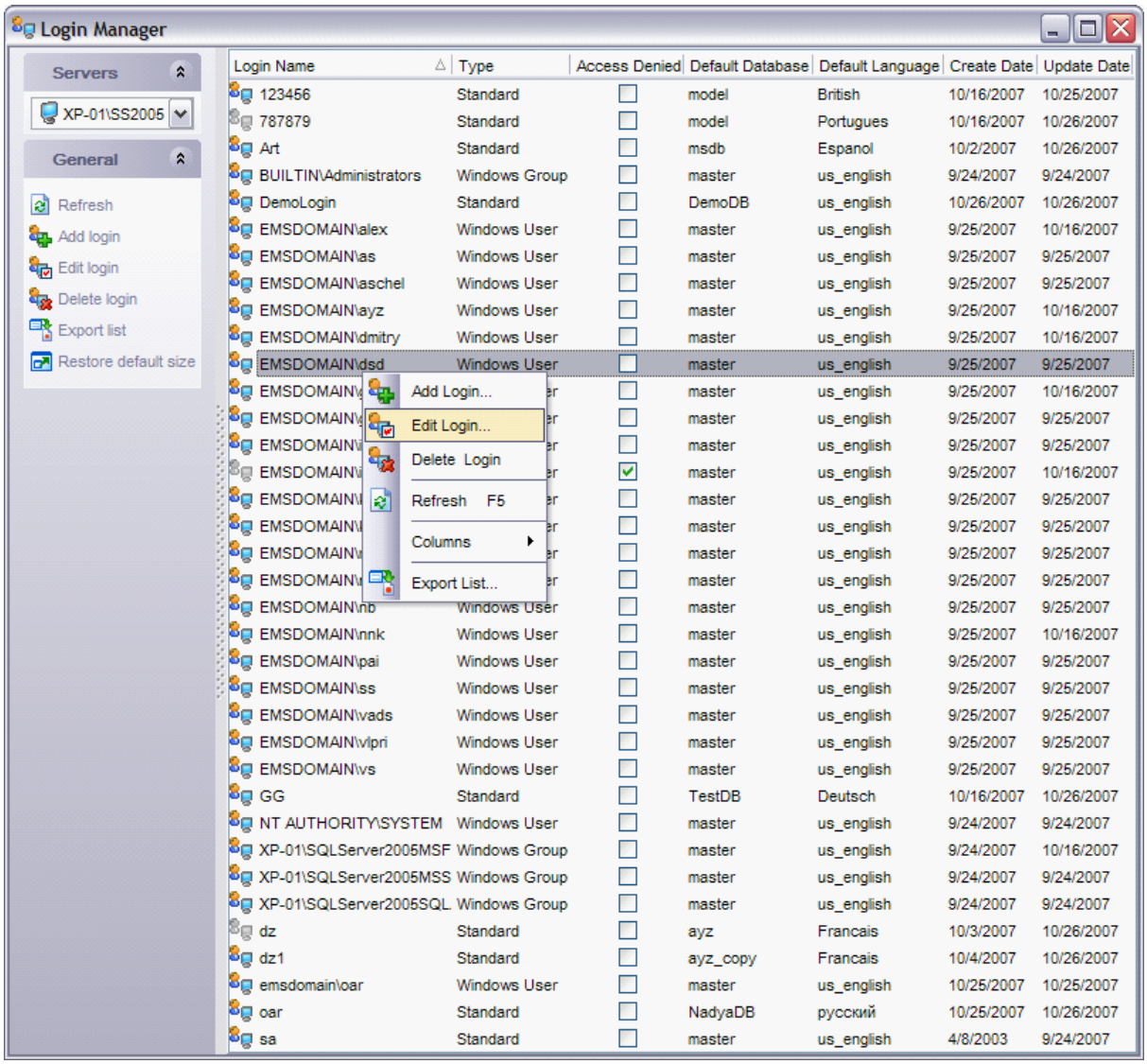
Server Roles

10.1.1.1 Login Manager

The **Login Manager** allows you to browse the list of existing on each of available instances of SQL Server, and manage them efficiently.

To launch the tool, select the **Tools | Login Manager** main menu item, or right-click the host alias in the DB Explorer tree and select the **Tasks | Logins** item from the context menu.

© 2008 EMS Database Management Solutions



The list displays the existing logins as a grid with the following columns: *Login Name*, *Type*, *Access Denied*, *Default Database*, *Default Language*, *Create Date*, *Update Date*.

Right-click an item within the list to call the **context menu** allowing you to *create* a new server login and specify its properties using Login Editor, *edit*, *delete* the selected login, *refresh* the list, or show/hide columns of the list. Using the context menu you can also export the list of logins to any of supported output file formats.

Logins management tasks are also available through the **Navigation bar** of the **Login Manager**.

Availability:
Full version (for Windows) **Yes**

© 2008 EMS Database Management Solutions

Lite version (for Windows)	**Yes**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

Login Editor
Server Roles

10.1.1.2 Login Editor

Login Editor allows you to define login properties. It opens automatically when you create a new server login and is available on editing an existing one (see Add login and Edit login for details).

To open a login in **Login Editor**, double-click it in the DB Explorer tree or in the Login Manager. Alternatively, you can right-click the login alias and select the **Edit Login** context menu item.

- Setting login parameters
- Allocating server roles
- Defining server permissions
- Defining endpoint permissions
- Defining database permissions
- User mapping
- Viewing DDL definition

Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**Yes**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

10.1.1.2.1 Setting login parameters

Use the **Login** tab of **Login Editor** to configure login authentication and default parameters.

Specify the way used by the login to connect to the instance of SQL Server:

- *Windows authentication*
- *SQL Server authentication*
- *Mapped to certificate*
- *Mapped to asymmetric key*

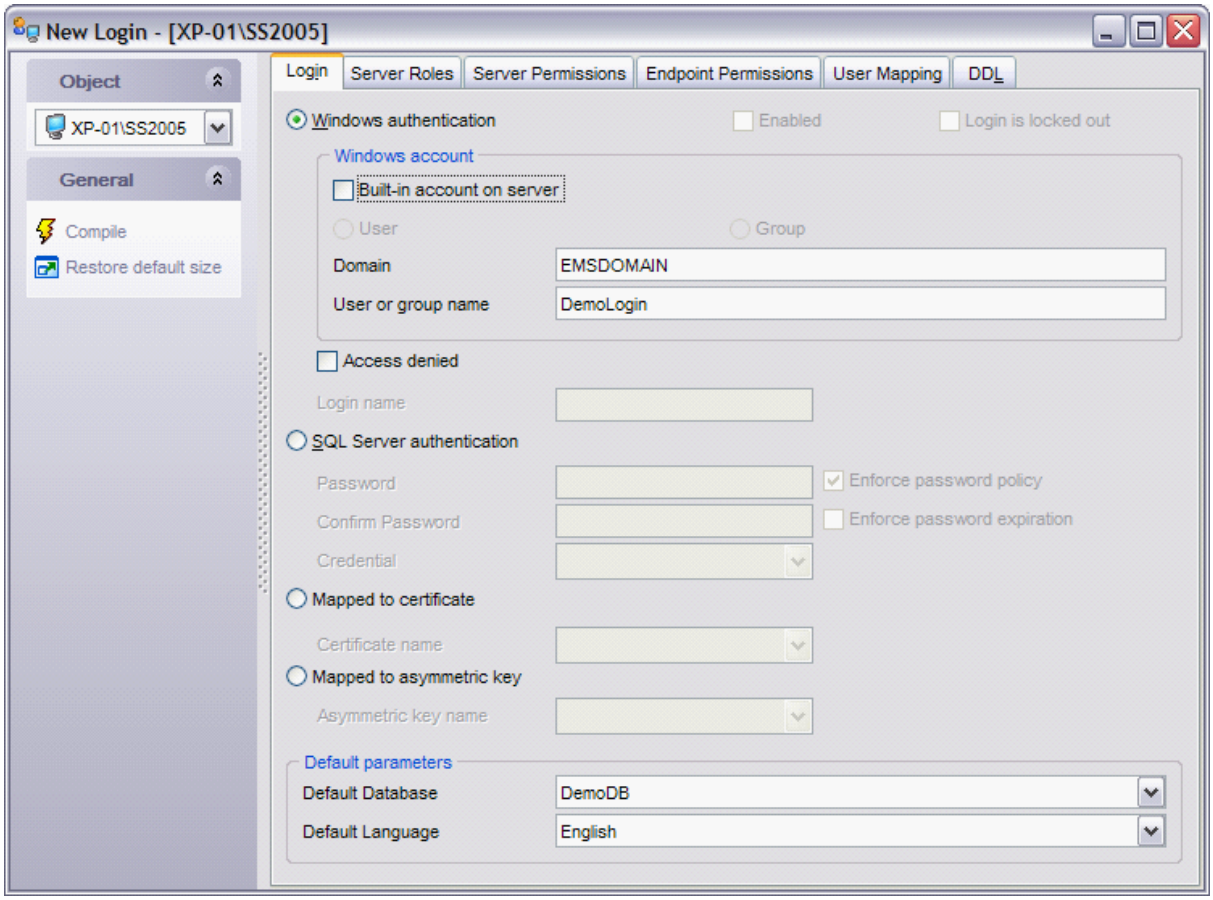
© 2008 EMS Database Management Solutions

Enabled
Enables/disables the login immediately after it is created. A disabled login still exists in the server, but it cannot be used to connect to the instance of SQL Server.

Login is locked out
This option is used to indicate whether the login is locked out.

Windows authentication
When Windows Authentication is used to connect to SQL Server, Microsoft Windows is completely responsible for authenticating the client. In this case, the client is identified by its Windows account (Windows-level login).

Built-in account on server
Select this option to specify a built-in server account for the login (*User* or *Group*), or specify the Windows **Domain** and **User or group name**.



Access denied
Select this option to forbid any connections to the server for the login.

Login name
Enter a name for the new login, or modify the name of the login being edited.

SQL Server authentication

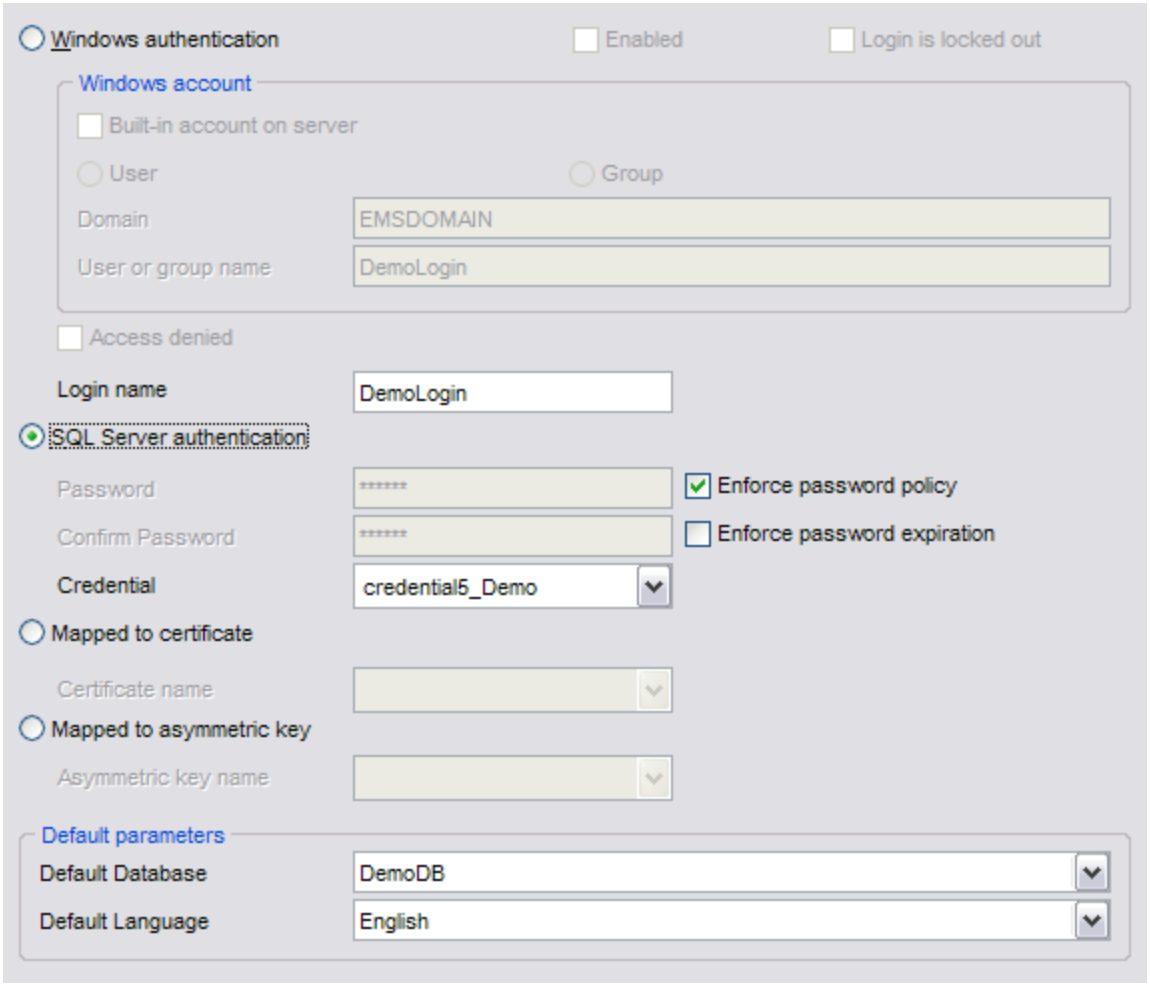
© 2008 EMS Database Management Solutions

When SQL Server Authentication is used, SQL Server authenticates the client by comparing the client-supplied user name and password to the list of valid user names and passwords maintained within SQL Server.

Enforce password policy
This option specifies that the Windows password policies of the computer where SQL Server is running should be enforced on this login.

Enforce password expiration
This option specifies whether password expiration policy should be enforced on this login.

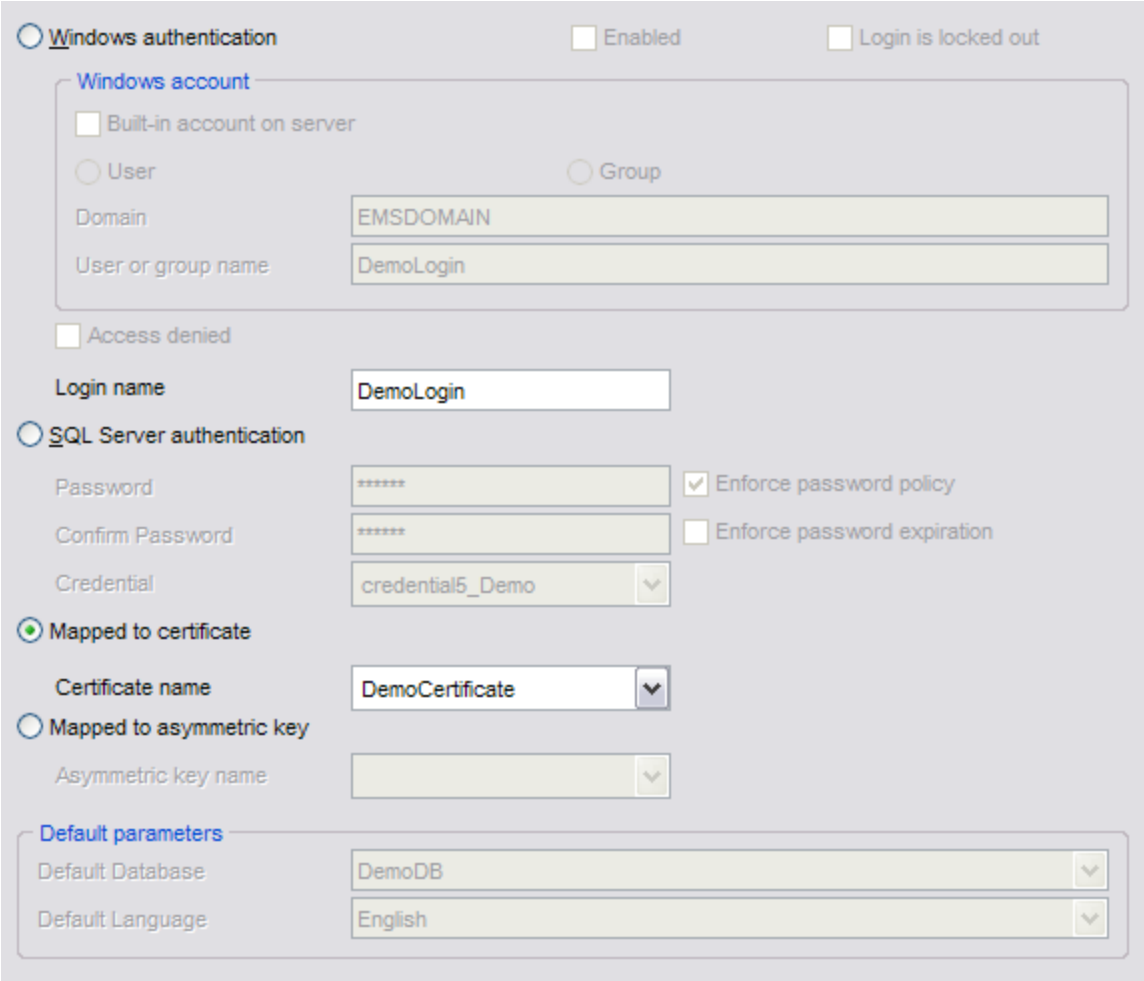
Credential
Use the drop-down list to select the credential that will be mapped to the new SQL Server login.



Mapped to certificate
Select this option to specify a certificate-mapped login.

Certificate name
Use the drop-down list to select the certificate that will be associated with this login.

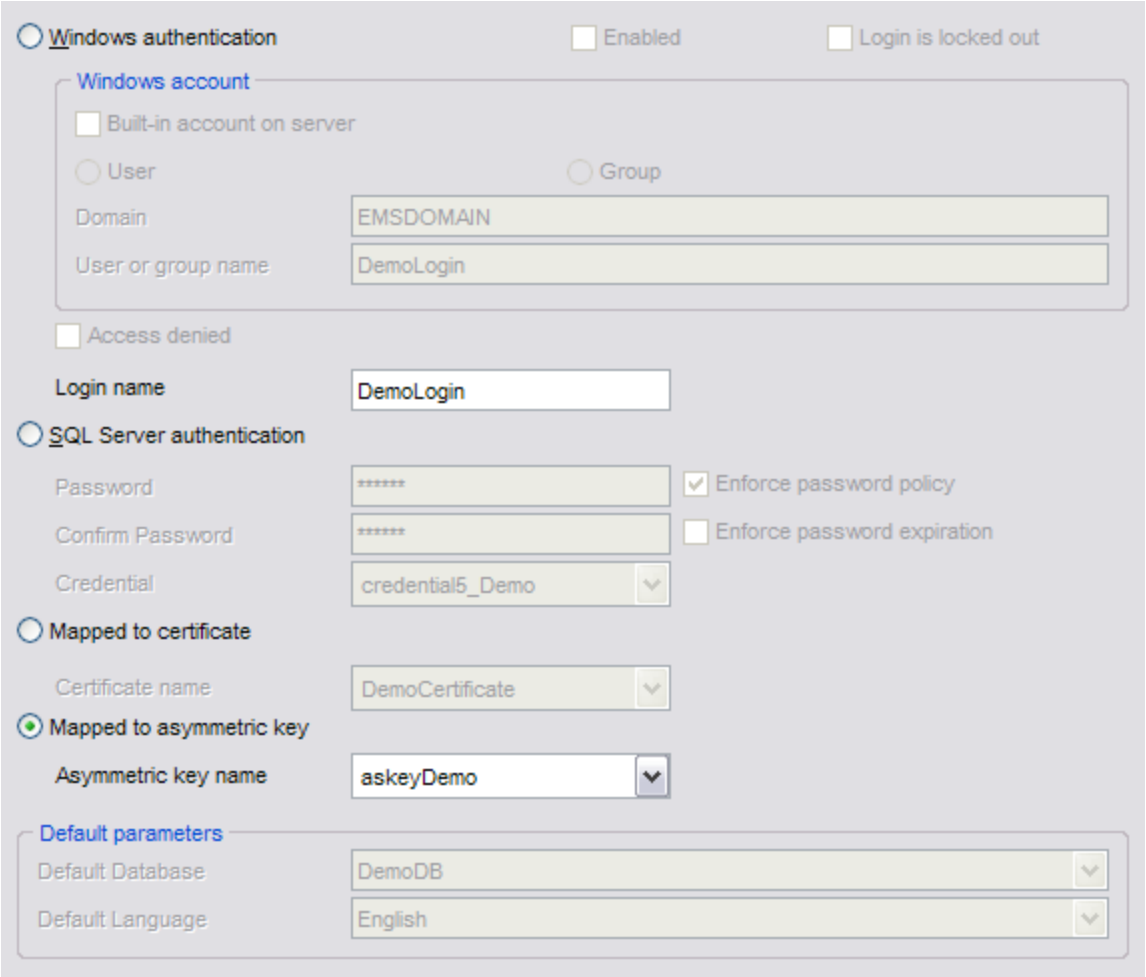
© 2008 EMS Database Management Solutions



Mapped to asymmetric key
Select this option to specify an asymmetric key-mapped login.

Asymmetric key name
Use the drop-down list to select the asymmetric key that will be associated with this login.

© 2008 EMS Database Management Solutions

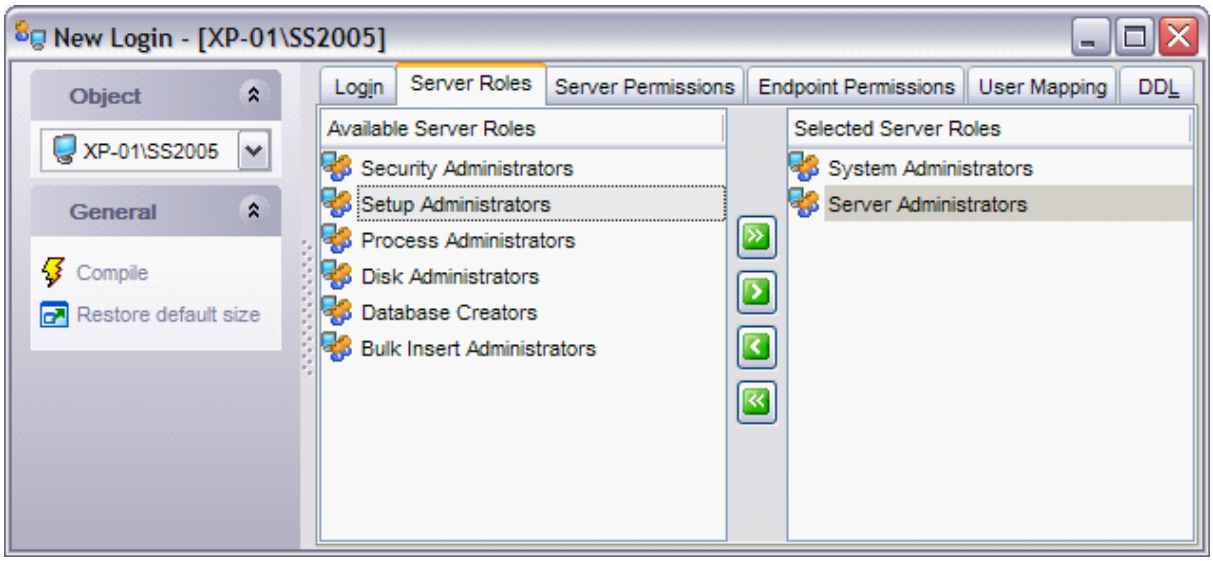


The lower group allows you to set the **Default parameters** pertaining to the login: **Default Database** and **Default Language**. Use the corresponding drop-down lists to select the required values that will be assigned to the login.

10.1.1.2.2 Allocating server roles

The **Server Roles** tab of **Login Editor** allows you to define the login membership, i.e. to select the built-in server role(s) the login will belong to. The selected server roles determine the tasks that can be performed through the login.

© 2008 EMS Database Management Solutions

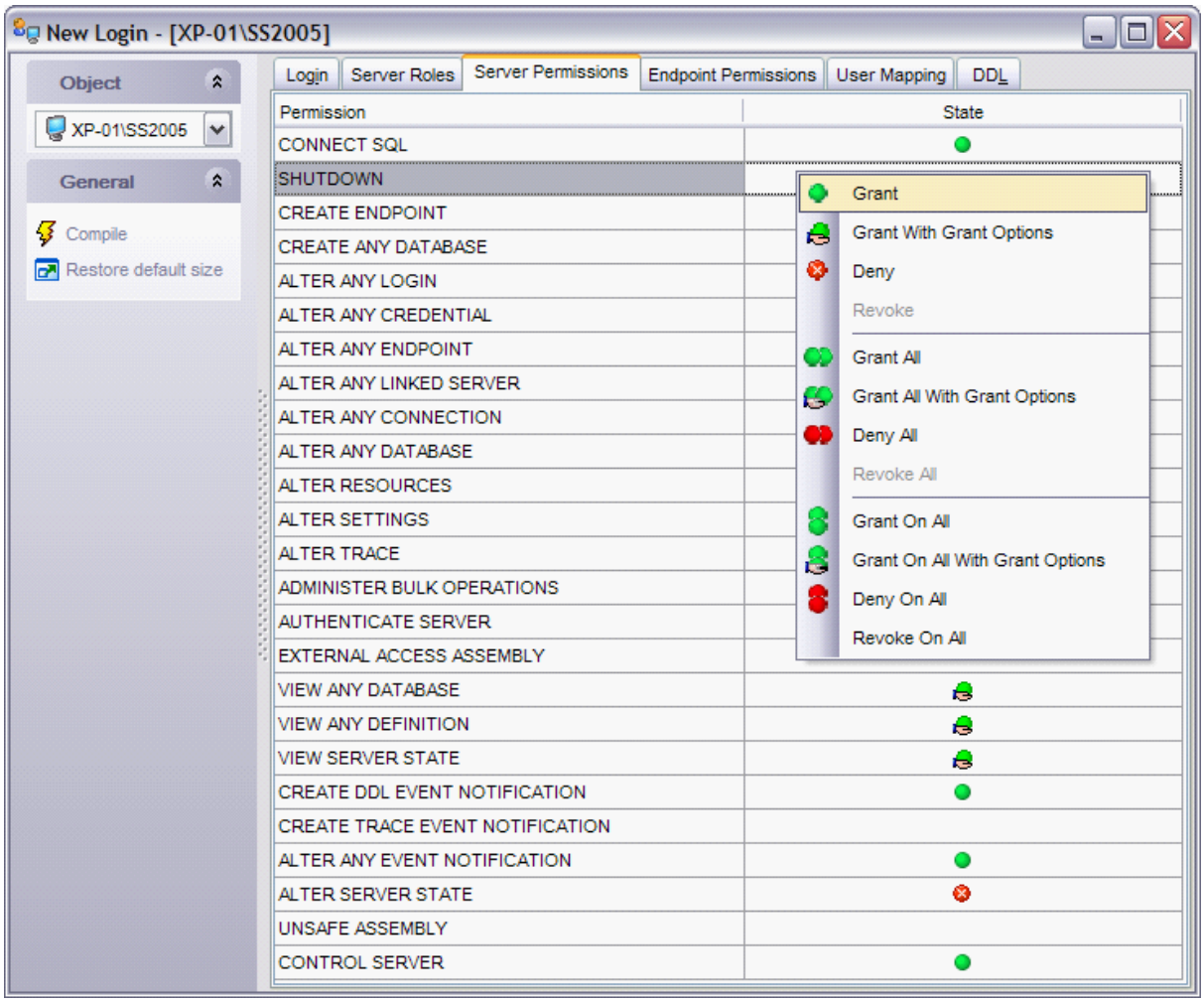


To select a server role, you need to move it from the **Available Server Roles** list to the **Selected Server Roles** list. Use the buttons or drag-and-drop operations to move the server roles from one list to another.

10.1.1.2.3 Defining server permissions

This tab allows you to grant **Server permissions** to the login.

© 2008 EMS Database Management Solutions



The **Permission** column contains the list of server permissions that can be granted to the login: *CONNECT SQL*, *SHUTDOWN*, *CREATE ANY DATABASE*, *CREATE ENDPOINT*, *ALTER ANY LOGIN*, *ALTER ANY CREDENTIAL*, *ALTER ANY LINKED SERVER*, *ALTER ANY CONNECTION*, etc.

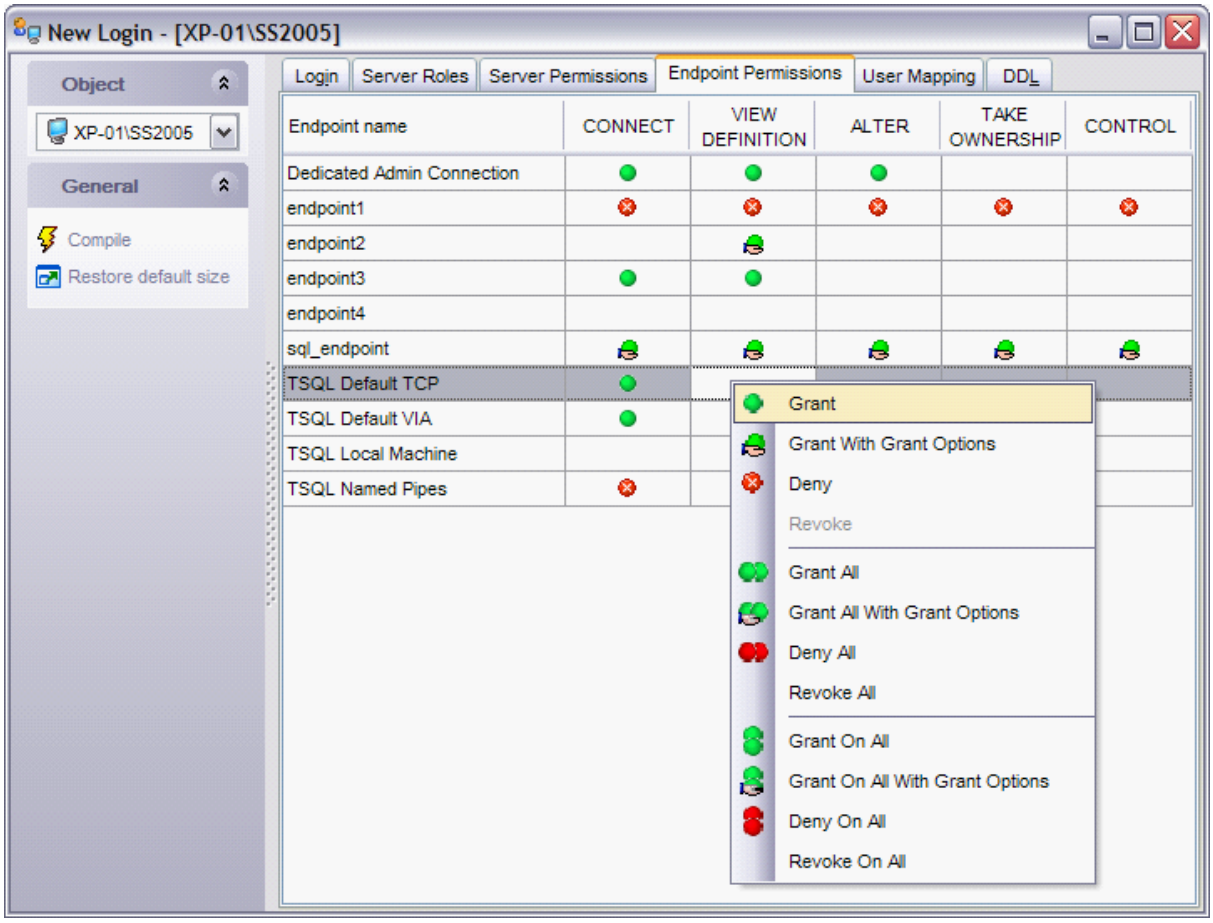
Right-click a cell within the **State** column to grant a permission to the login. The context menu of a cell allows you to:

- grant a permission to the login;
- grant a permission (with Grant Option) to the login;
- deny a permission to the login;
- revoke a previously granted or denied permission;
- grant all permissions to the login;
- grant all permissions (with Grant Option) to the login;
- deny all permissions to the login;
- revoke all previously granted or denied permissions.

© 2008 EMS Database Management Solutions

10.1.1.2.4 Defining endpoint permissions

This tab allows you to grant **Endpoint permissions** to the login.



The **Endpoint name** column contains the list of server endpoints; each subsequent column corresponds to the permission which can be granted on the selected endpoint: *CONNECT*, *VIEW DEFINITION*, *ALTER*, *TAKE OWNERSHIP*, *CONTROL*.

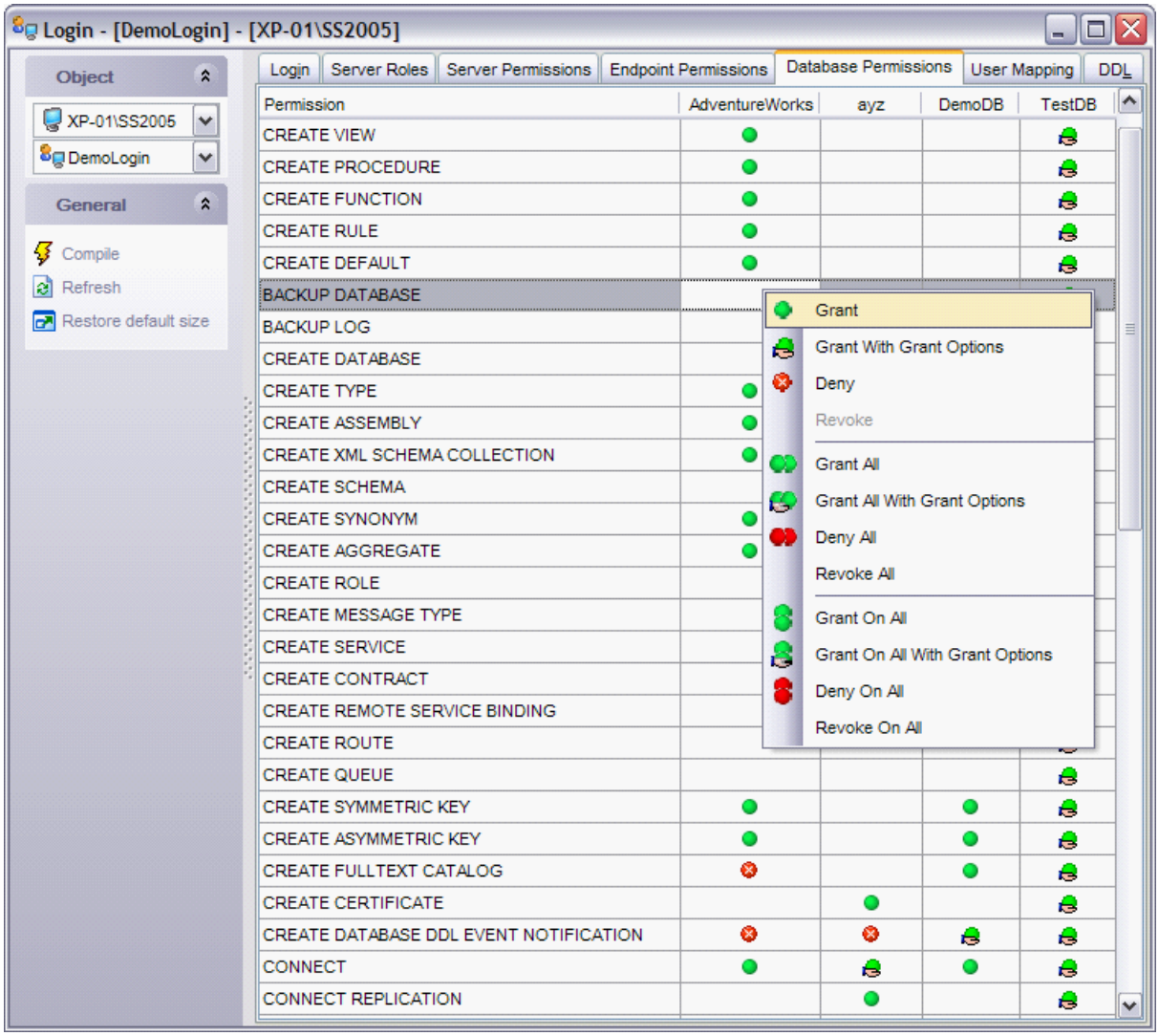
Right-click a cell to grant a specific permission on a certain endpoint. To grant a permission on an endpoint, you should find the object in the **Endpoint name** list and the column with the corresponding permission. The context menu of a cell allows you to:

- grant a permission on the endpoint to the login;
- grant a permission (with Grant Option) on the endpoint to the login;
- deny a permission on the endpoint to the login;
- revoke a previously granted or denied permission;
- grant all permissions on the endpoint to the login;
- grant all permissions (with Grant Option) on the endpoint to the login;
- deny all permissions on the endpoint to the login;
- revoke all previously granted or denied permissions;
- grant a permission on all endpoints to the login;
- grant a permission (with Grant Option) on all endpoints to the login;
- deny a permission on all endpoints to the login;
- revoke a previously granted or denied permission on all endpoints

© 2008 EMS Database Management Solutions

10.1.1.2.5 Defining database permissions

This tab allows you to grant **Database permissions** to the login. This tab is only available on editing an existing server login.



The **Permission** column contains the list of database permissions that can be granted to the login: *CREATE ANY VIEW*, *CREATE PROCEDURE*, *CREATE FUNCTION*, *CREATE RULE*, *CREATE DEFAULT*, *BACKUP DATABASE*, etc.; each subsequent column corresponds to a database that resides on the selected server.

Right-click a cell to grant a specific permission on a certain database that has a user mapped to the login. To grant a permission on a database, you should find the permission in the **Permission** list and the column with the corresponding database. The context menu of a cell allows you to:

- grant a permission on the database to the login;

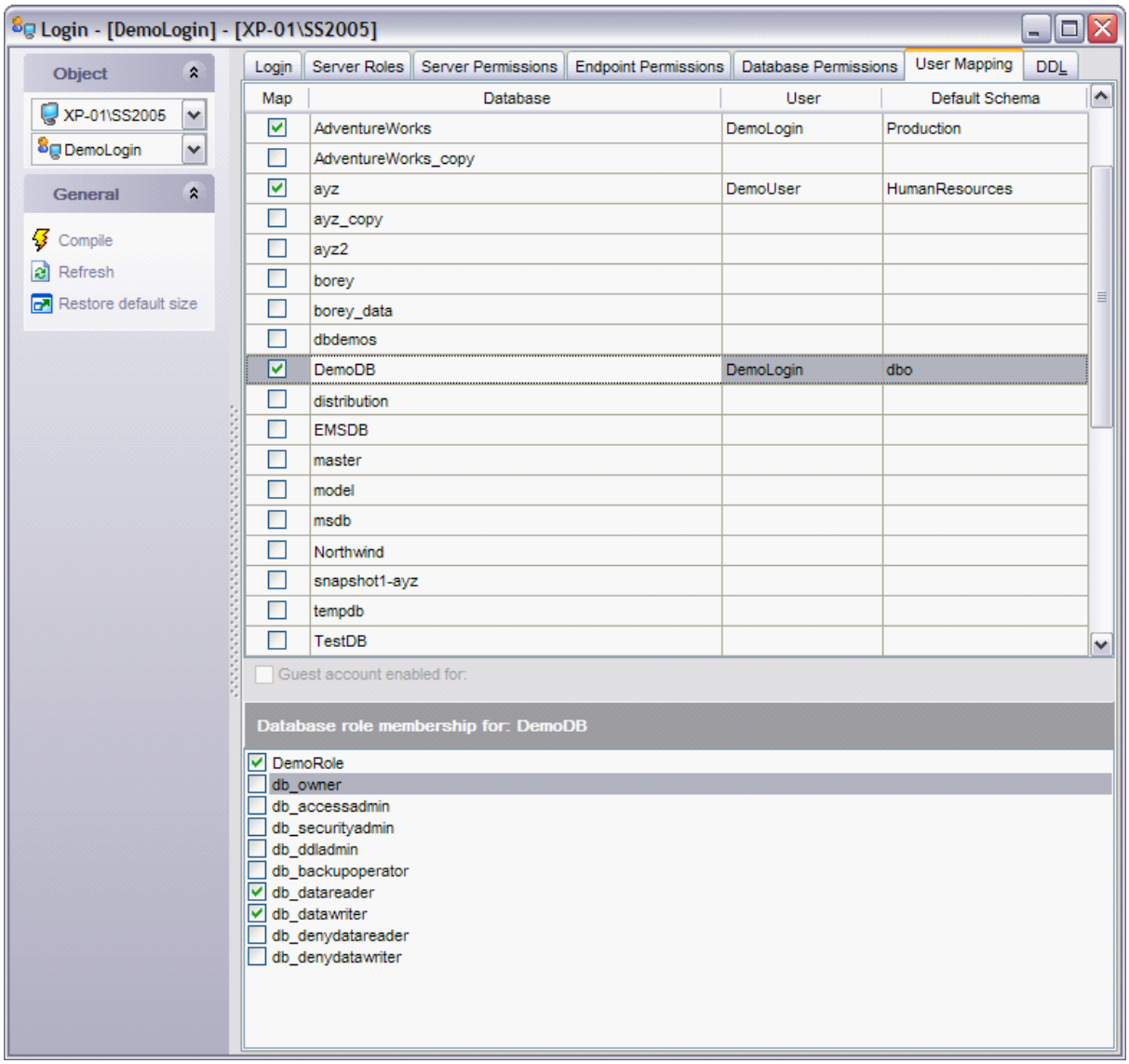
© 2008 EMS Database Management Solutions

- grant a permission (with Grant Option) on the database to the login;
- deny a permission on the database to the login;
- revoke a previously granted or denied permission;
- grant all permissions on the database to the login;
- grant all permissions (with Grant Option) on the database to the login;
- deny all permissions on the database to the login;
- revoke all previously granted or denied permissions;
- grant a permission on all databases to the login;
- grant a permission (with Grant Option) on all databases to the login;
- deny a permission on all databases to the login;
- revoke a previously granted or denied permission on all databases

10.1.1.2.6 User mapping

The **User Mapping** tab allows you to browse the list of databases available on the specified server, and to map their users to the login.

© 2008 EMS Database Management Solutions



Set the **Map** flag next to a database name, and specify the **User** and the **Default Schema** of the database using the corresponding drop-down lists.

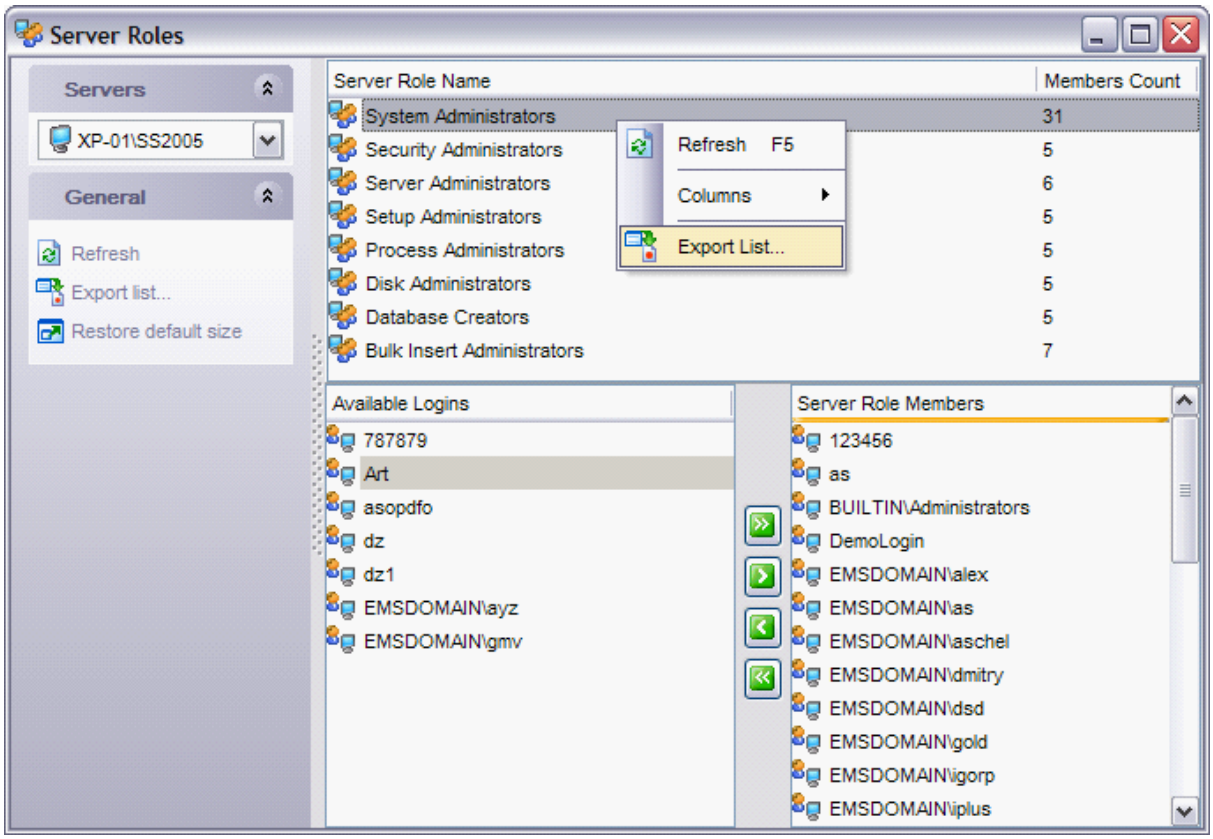
If necessary, use the **Database role membership for** area at the bottom to define database role(s) for each mapped user.

10.1.2 Server Roles

The **Server Roles manager** allows you to browse the list of SQL Server **roles**, and to define login membership for each of the roles.

To launch the tool, select the **Tools | Server Roles** main menu item, or right-click the host alias in the DB Explorer tree and select the **Tasks | Server Roles** item from the context menu.

© 2008 EMS Database Management Solutions



The upper list displays the server roles as a grid with the following columns: *Server Role Name*, *Members Count*.

Right-click an item within the list to call the **context menu** allowing you to *refresh* the list, or show/hide columns of the list. Using the context menu you can also export the list of server roles to any of supported output file formats.

Servers roles management tasks are also available through the **Navigation bar** of the **Server Roles manager**.

The lower area allows you to define login membership, i.e. to select the logins that will belong to the selected server role. The server roles determine the tasks that can be performed through the selected login.

To select a login, you need to move it from the **Available Logins** list to the **Server Role Members** list. Use the buttons or drag-and-drop operations to move the logins from one list to another.

Availability:
Full version (for Windows) **Yes**

© 2008 EMS Database Management Solutions

Lite version (for Windows)	**Yes**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

Logins

© 2008 EMS Database Management Solutions

10.2 SQL Server objects and services

SQL Manager 2008 for SQL Server provides graphical user interface for most SQL Server objects and native services.

The following *tools for SQL Server objects and services management* are available in SQL Manager 2008 for SQL Server:

- **Backup Devices**
  Provides tools for working with backup devices: Backup devices manager, Backup Device properties
- **Database Snapshots**
  Provides tools for working with database snapshots: Snapshots manager, Snapshot properties
- **Linked Servers**
  Provides tools for working with linked servers: Linked servers manager, Linked Server Editor
- **DTS Packages**
  Provides tools for working with DTS packages: DTS Packages manager, DTS Package properties
- **Jobs**
  Provides tools for working with jobs: Jobs manager, Job Editor
- **Job History**
  Provides a convenient tool to browse the list of recent jobs: Job History
- **Alerts**
  Provides tools for working with alerts: Alerts manager, Alert Editor
- **Operators**
  Provides tools for working with operators: Operators manager, Operator Editor
- **Shared Schedules**
  Provides tools for working with schedules: Shared schedules manager, Schedule Editor
- **Proxies**
  Provides tools for working with proxies: Proxies manager, Proxy Editor
- **Credentials**
  Provides a powerful tool for working with credentials: Credential Editor
- **Endpoints**
  Provides a powerful tool for working with endpoints: Endpoint Editor
- **DDL Triggers**
  Provides a powerful tool for working with DDL triggers: DDL Trigger Editor
- **Target Servers**
  Provides a convenient tool for target servers management: Target Servers manager
- **Server Logs**
  Provides a convenient tool to browse the server logs: Server Log Viewer
- **Server Activity**
  Provides a convenient tool to get information about connections and the locks that they hold: Activity Monitor

See also:

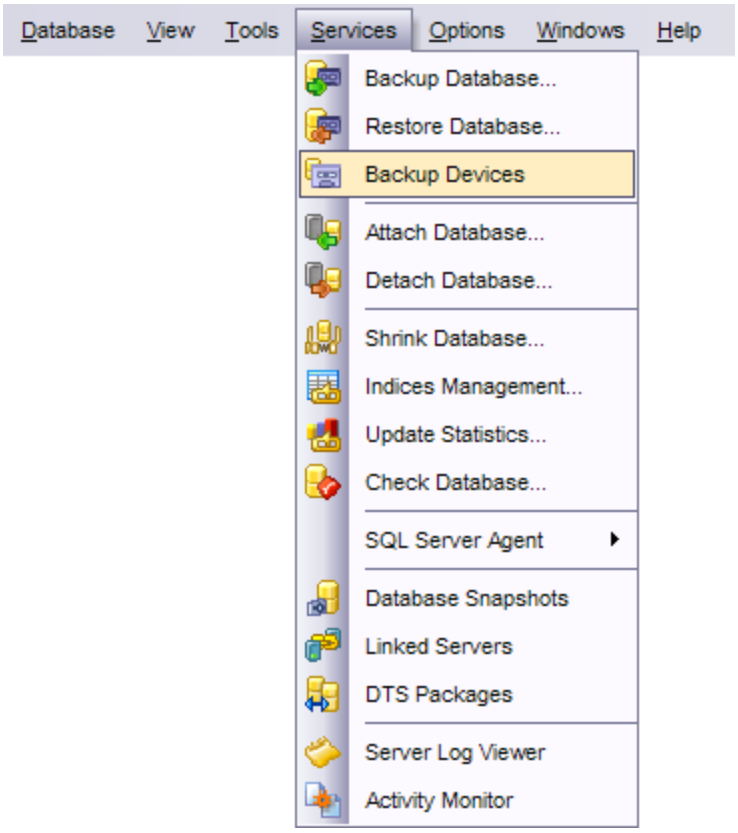
Server-level principals

© 2008 EMS Database Management Solutions

10.2.1 Backup Devices

A **backup device** is a tape drive or a disk drive used in backup and restore operations. When creating a backup, you must select a backup device where the data will be written. Microsoft SQL Server 2005 can back up databases, transaction logs, and files to disk and tape devices.

- Backup devices manager
- Backup Device properties



Creating Backup Devices

To create a new backup device:

- right-click the **Backup Devices** node (within the **Server Objects** branch) or any object within this node in the DB Explorer tree and select the **New Backup Device** item from the context menu;
- define backup device properties using the appropriate boxes of the New Logical Backup Device dialog

or

- select the **Database | New Object...** main menu item;
- select **Backup Device** in the Create New Object dialog;
- define backup device properties using the New Logical Backup Device dialog.

Hint: To create a new backup device, you can also select the **Services | Backup Devices** main menu item to open the **Backup devices manager** and select the **Add Device...** item from the context menu or on the Navigation bar.

© 2008 EMS Database Management Solutions

To create a new backup device with the same properties as one of existing backup devices has:
- select the **Database | Duplicate Object...** menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a backup device in the DB Explorer tree and select the **Duplicate Backup Device <backup_device_name>...** context menu item.

Duplicate Object Wizard allows you to select the server to create a new backup device and to edit the result SQL statement for creating the backup device.

Editing Backup Devices

To change the name of a backup device:
- select the backup device to rename in the DB Explorer tree;
- right-click the backup device alias and select the **Rename Backup Device <backup_device_name>...** item from the context menu;
- edit the backup device name using the **Rename Object...** dialog.

Dropping Backup Devices

To drop a backup device:
- select the backup device to drop in the DB Explorer tree;
- right-click the object and select the **Drop Backup Device <backup_device_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new backup device;
Ctrl+R to rename the backup device;
Shift+Del to drop the object.

To create, rename and drop backup devices, you can also use the **context menu** and the **Navigation bar** of Backup devices manager.

See also:
Database Snapshots
Linked Servers
DTS Packages
Jobs
Job History
Alerts
Operators
Schedules
Proxies
Credentials
Endpoints

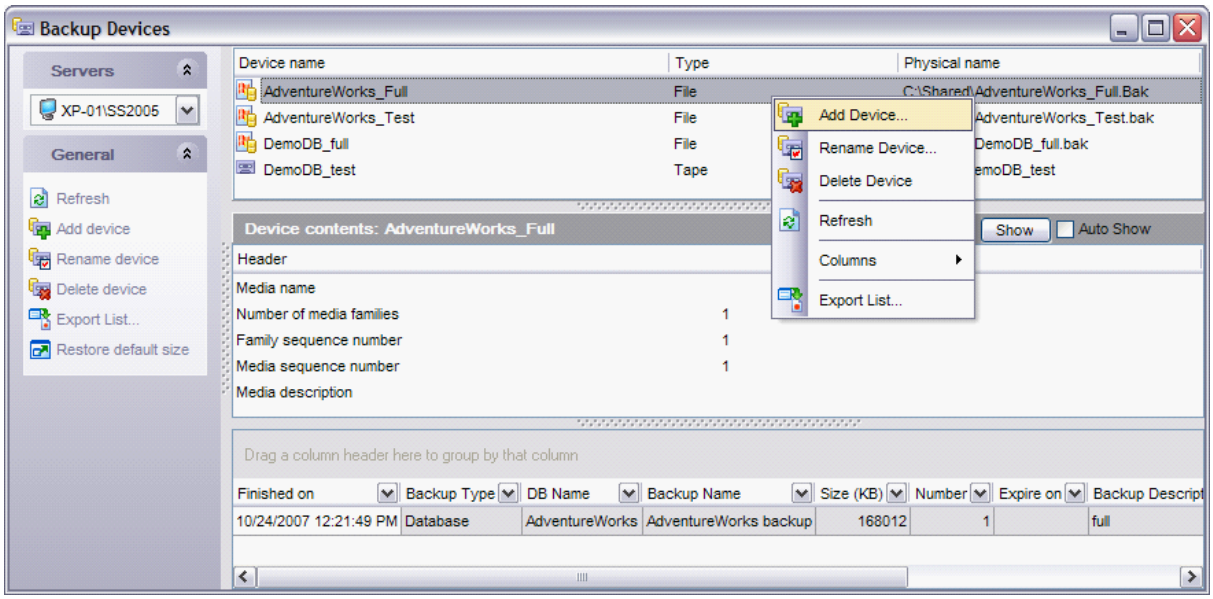
© 2008 EMS Database Management Solutions

DDL Triggers
Target Servers
Server Log Viewer
Activity Monitor

10.2.1.1 Backup devices manager

The **Backup devices manager** allows you to browse the list of SQL Server **backup devices** and manage them efficiently.

To launch the tool, select the **Services | Backup Devices** main menu item, or right-click the host alias in the DB Explorer tree and select the **Tasks | Backup Devices** item from the context menu.



The list displays the existing backup devices as a grid with the following columns: *Device name*, *Type*, *Physical name*.

Right-click an item within the list to call the **context menu** allowing you to *create* a new backup device and specify its properties using the New Logical Backup Device dialog, *rename*, *delete* the selected backup device, *refresh* the list, or show/hide columns of the list. Using the context menu you can also export the list of backup devices to any of supported output file formats.

Backup devices management tasks are also available through the **Navigation bar** of the **Backup devices manager**.

The **Device contents** area allows you to view the contents of the selected backup device: *Media name*, *Number of media families*, *Family sequence number*, *Media description*.

© 2008 EMS Database Management Solutions

To display the actual contents, press the **Show** button. If necessary, you can check the **Auto show** option - in this case the contents of each backup device will be retrieved upon the backup device selection in the list.

The lower list displays the backup operation(s) with the following attributes:
Finished on
Backup type
DB Name
Backup Name
Size (KB)
Number
Expire on
Backup description
Device Type
Created on
Created by

If necessary, you can **group the data in grid** by any of the columns. This operation is performed by dragging the column header to the gray **"Group by" box** area at the top. When grouping by a column is applied to the grid, all the rows are displayed as subnodes to the grouping row value. To reverse grouping, just drag the column header back.

Hint: These properties are also available within the modal Backup Device Properties dialog which is called through the **Backup Device Contents...** context menu item of the backup device alias in DB Explorer.

Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**Yes**

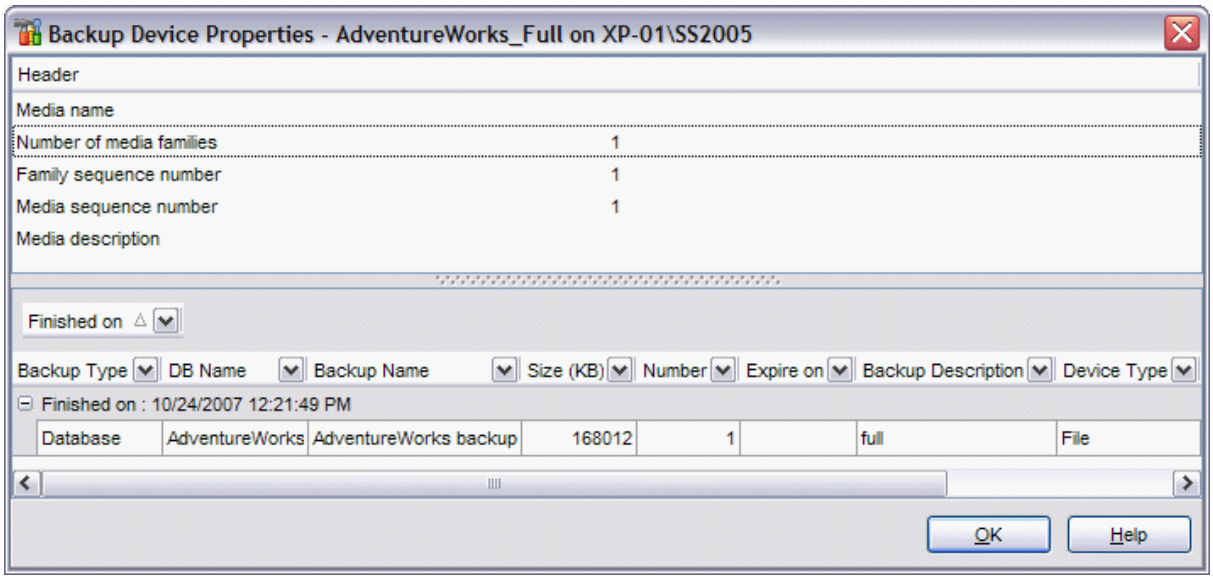
Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

10.2.1.2 Backup Device properties

The **Backup Device Properties** dialog allows you to browse the **backup device** contents and the backup operation(s) with this backup device used.

To open the dialog, right-click the backup device in DB Explorer and select the **Backup Device Contents...** context menu item.

© 2008 EMS Database Management Solutions



The **Device contents** area allows you to view the contents of the backup device: *Media name*, *Number of media families*, *Family sequence number*, *Media description*.

The lower list displays the backup operation(s) as a grid with the following columns: *Finished on*, *Backup type*, *DB Name*, *Backup Name*, *Size (KB)*, *Number*, *Expire on*, *Backup description*, *Device Type*, *Created on*, *Created by*.

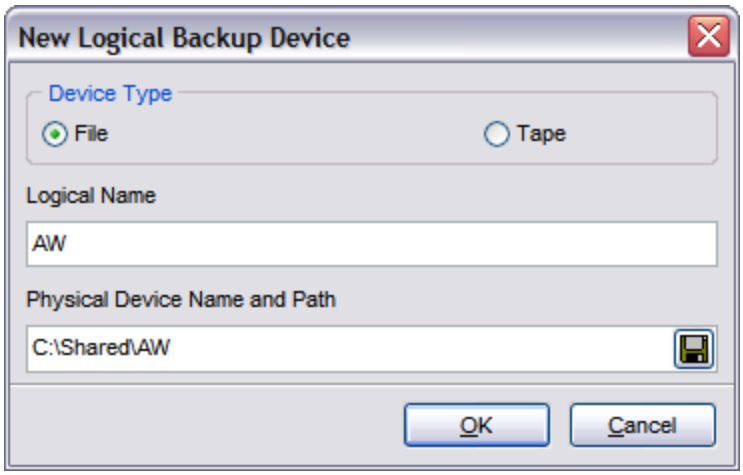
If necessary, you can **group the data in grid** by any of the columns. This operation is performed by dragging the column header to the gray **"Group by" box** area at the top. When grouping by a column is applied to the grid, all the rows are displayed as subnodes to the grouping row value. To reverse grouping, just drag the column header back.

10.2.1.3 New backup device

The **New Logical Backup Device** dialog opens automatically when you create a new backup device (see Create backup device for details) and allows you to define the new backup device parameters.

To call the dialog, you can right-click the **Backup Devices** node or any object within this node in the DB Explorer tree and use the *Ctrl+N* shortcut.

© 2008 EMS Database Management Solutions



Device Type
Select the logical backup device type: *File*, *Pipe* (for SQL Server versions earlier than 2005), or *Tape*.

Disk backup devices are files on hard disks or other disk storage media are the same as regular operating system files.
Tape backup devices are used in the same way as disk devices, but the tape device must be connected physically to the computer running an instance of SQL Server, and if a tape backup device is filled during the backup operation, but more data still needs to be written, SQL Server prompts for a new tape and continues the backup operation.

Logical Name
Enter a name for the new backup device.

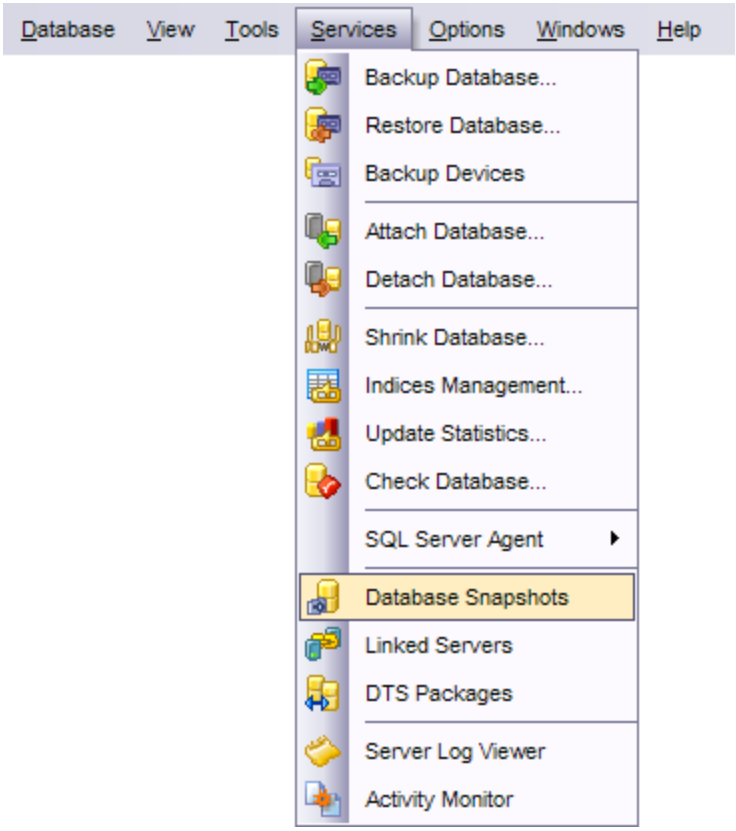
Physical Device Name and Path
Define the physical location of the device.
If **Device Type** is set to *File*, enter the path and the full name of the file.
If **Device Type** is set to *Tape*, enter the physical name of the device.

10.2.2 Database Snapshots

Database snapshots are a new feature of Microsoft® SQL Server 2005, and they are available only in the Enterprise Edition of Microsoft® SQL Server 2005.
A **database snapshot** is a read-only, static view of a database (the source database). Multiple snapshots can exist on a source database and always reside on the same server instance as the database. Each database snapshot is transactionally consistent with the source database as of the moment of the snapshot's creation.
Snapshots can be used for reporting purposes. Also, in the event of a user error on a source database, you can revert the source database to the state it was in when the snapshot was created. Data loss is confined to updates to the database since the snapshot's creation.

- Managing database snapshots

© 2008 EMS Database Management Solutions



See also:

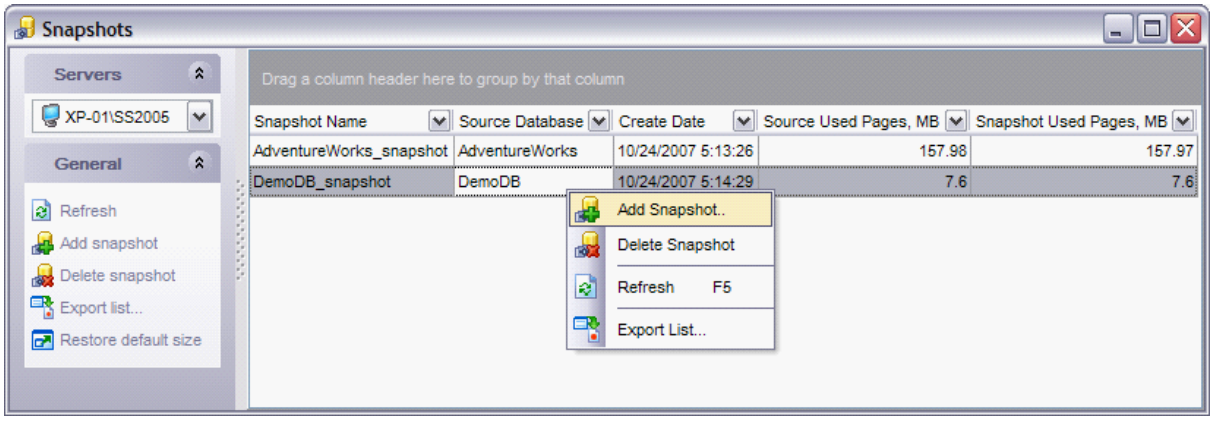
Backup Devices
Linked Servers
DTS Packages
Jobs
Job History
Alerts
Operators
Schedules
Proxies
Credentials
Endpoints
DDL Triggers
Target Servers
Server Log Viewer
Activity Monitor

© 2008 EMS Database Management Solutions

10.2.2.1 Managing database snapshots

The **Snapshots manager** allows you to browse the list of **database snapshots** and manage them efficiently.

To launch the tool, select the **Services | Database Snapshots** main menu item, or right-click the host alias in the DB Explorer tree and select the **Tasks | Database Snapshots** item from the context menu.



The list displays the existing database snapshots as a grid with the following columns: *Snapshot Name*, *Source Database*, *Create Date*, *Source Used Pages (MB)*, *Snapshot Used Pages (MB)*.

If necessary, you can **group the data in grid** by any of the columns. This operation is performed by dragging the column header to the gray **"Group by" box** area at the top. When grouping by a column is applied to the grid, all the rows are displayed as subnodes to the grouping row value. To reverse grouping, just drag the column header back.

Right-click an item within the list to call the **context menu** allowing you to *create* a new snapshot and specify its properties using the Add New Database Snapshot dialog, *delete* the selected snapshot, or *refresh* the list. Using the context menu you can also export the list of database snapshots to any of supported output file formats.

Database snapshots management tasks are also available through the **Navigation bar** of the **Snapshots manager**.

Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**No**

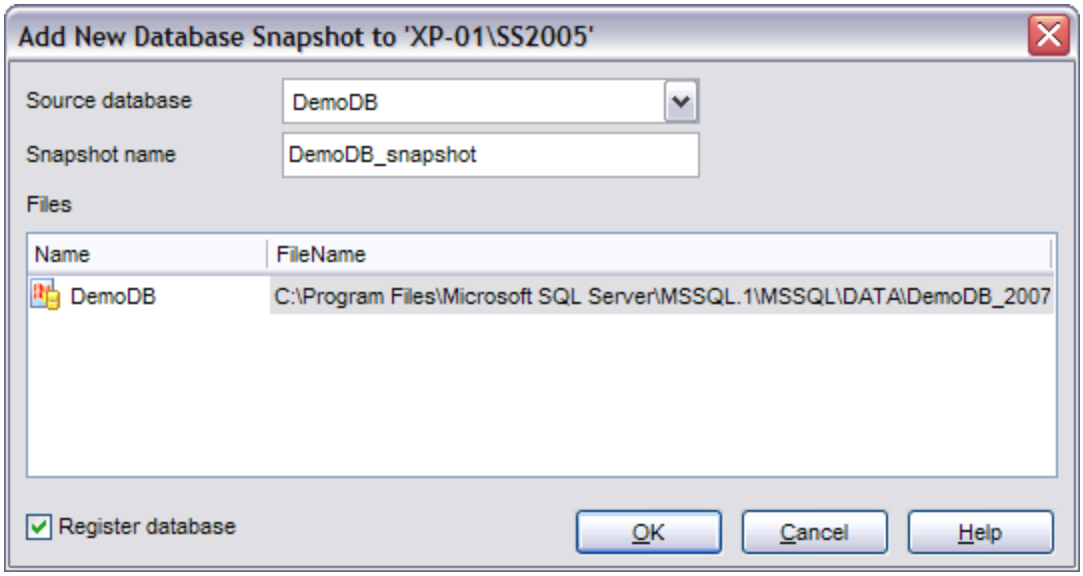
Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

© 2008 EMS Database Management Solutions

10.2.2.2 Database snapshot properties

The **Add New Database Snapshot** dialog opens automatically when you create a new database snapshot and allows you to define the new snapshot parameters.

To call the dialog, right-click within the snapshots list in the Snapshots manager and select the **Add Snapshot...** context menu item.



Source database
Use the drop-down list to select the source database for the new snapshot.

Snapshot name
By default, the snapshot name is generated by the application automatically on the basis of the source database name and the snapshot creation date. You can modify the name for the new snapshot, or leave the name unchanged. Note that each database snapshot requires a unique database name.

The **Files** area lists the database file(s) of the new database snapshot. The file name contains the name of the source database file and the snapshot creation date.

Register database
This option indicates that the Database Registration Info dialog for the new database will appear immediately after the database snapshot is created (you need to register the database to start working with it in SQL Manager). If you intend to register the database later, uncheck this option.

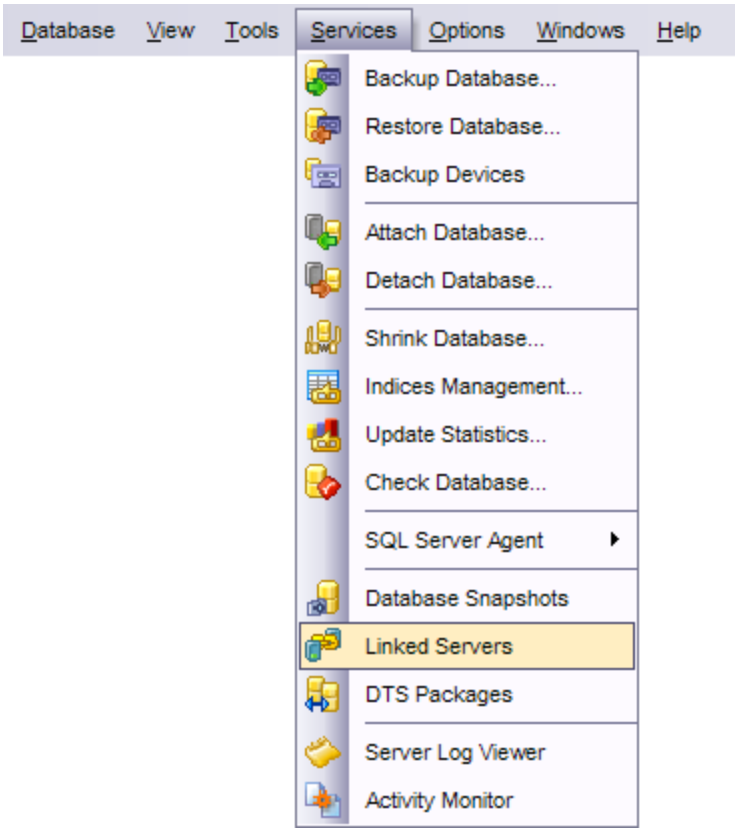
10.2.3 Linked Servers

A **linked server** is a virtual server that can be defined to SQL Server 2005 with all the information required to access an OLE DB data source. A **linked server** configuration enables SQL Server to execute commands against OLE DB data sources on remote servers. **Linked servers** provide a number of features: remote server access; the ability

© 2008 EMS Database Management Solutions

to issue distributed queries, updates, commands and transactions on heterogeneous data sources; the ability to address diverse data sources universally.

- Linked servers manager
- Linked Server Editor



Creating Linked Servers

To create a new linked server:

- right-click the **Linked Servers** node (within the **Server Objects** branch) or any object within this node in the DB Explorer tree and select the **New Linked Server** item from the context menu;
- define linked server properties using the appropriate tabs of Linked Server Editor.

or

- select the **Database | New Object...** main menu item;
- select **Linked Server** in the Create New Object dialog;
- define linked server properties using Linked Server Editor.

Hint: To create a new linked server, you can also select the **Services | Linked Servers** main menu item to open the **Linked servers manager** and select the **Add Linked Server** item from the context menu or on the Navigation bar.

To create a new linked server with the same properties as one of existing linked servers has:

- select the **Database | Duplicate Object...** menu item;

© 2008 EMS Database Management Solutions

- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a linked server in the DB Explorer tree and select the **Duplicate Linked Server <linked_server_name>...** context menu item.

Duplicate Object Wizard allows you to select the server to create a new linked server and to edit the result SQL statement for creating the linked server.

Editing Linked Servers

To edit an existing linked server:

- select the linked server for editing in the DB Explorer tree (type the first letters of the Linked Server name for quick search);
- right-click the object and select the **Edit Linked Server <linked_server_name>** context menu item, or simply double-click the linked server;
- edit linked server properties using the appropriate tabs of Linked Server Editor.

Dropping Linked Servers

To drop a linked server:

- select the linked server to drop in the DB Explorer tree;
- right-click the object and select the **Drop Linked Server <linked_server_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new linked server;
Ctrl+O to edit the selected linked server;
Shift+Del to drop the object.

To create, edit and drop linked servers, you can also use the **context menu** and the **Navigation bar** of Linked servers manager.

See also:

Backup Devices
Database Snapshots
DTS Packages
Jobs
Job History
Alerts
Operators
Schedules
Proxies
Credentials
Endpoints
DDL Triggers

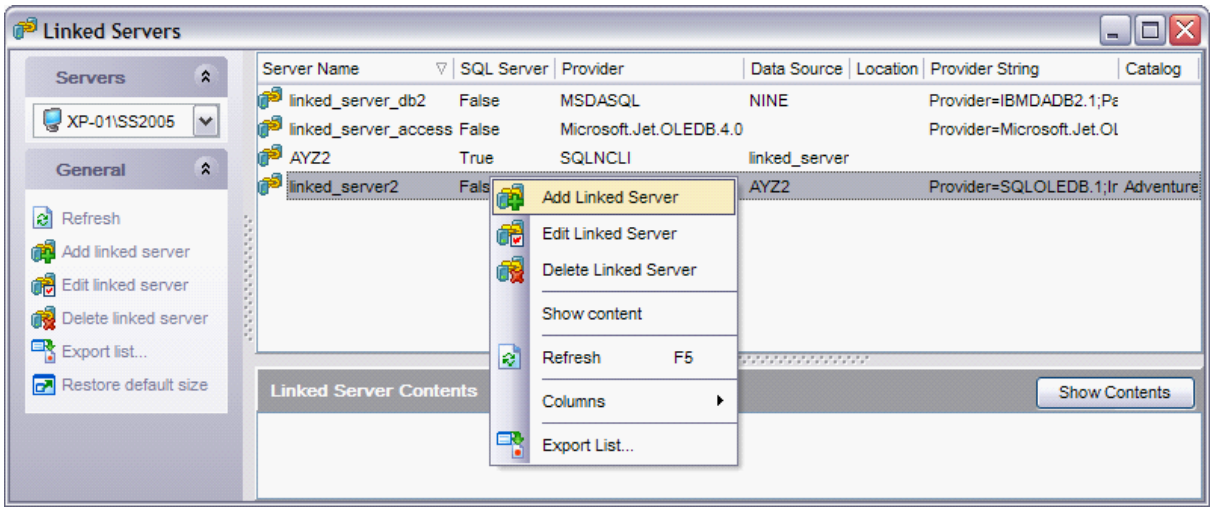
© 2008 EMS Database Management Solutions

Target Servers
Server Log Viewer
Activity Monitor

10.2.3.1 Linked servers manager

The **Linked servers manager** allows you to browse the list of **linked servers** and manage them efficiently.

To launch the tool, select the **Services | Linked Servers** main menu item, or right-click the host alias in the DB Explorer tree and select the **Tasks | Linked Servers** item from the context menu.



The list displays the existing linked servers as a grid with the following columns: *Server Name*, *SQL Server*, *Provider*, *Data Source*, *Location*, *Provider String*, *Catalog*.

Right-click an item within the list to call the **context menu** allowing you to *create* a new linked server and specify its properties using Linked Server Editor, *edit*, *delete* the selected linked server, retrieve its *content*, *refresh* the list, or show/hide columns of the list. Using the context menu you can also export the list of linked servers to any of supported output file formats.

Linked servers management tasks are also available through the **Navigation bar** of the **Linked servers manager**.

To display the actual linked server contents, press the **Show Contents** button.

Hint: Linked server contents are also available within the Content tab of Linked Server Editor.

Availability:

© 2008 EMS Database Management Solutions

Full version (for Windows)	**Yes**
Lite version (for Windows)	**No**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

10.2.3.2 Linked Server Editor

Linked Server Editor allows you to define linked server properties. It opens automatically when you create a new linked server and is available on editing an existing one (see Create linked server and Edit linked server for details).

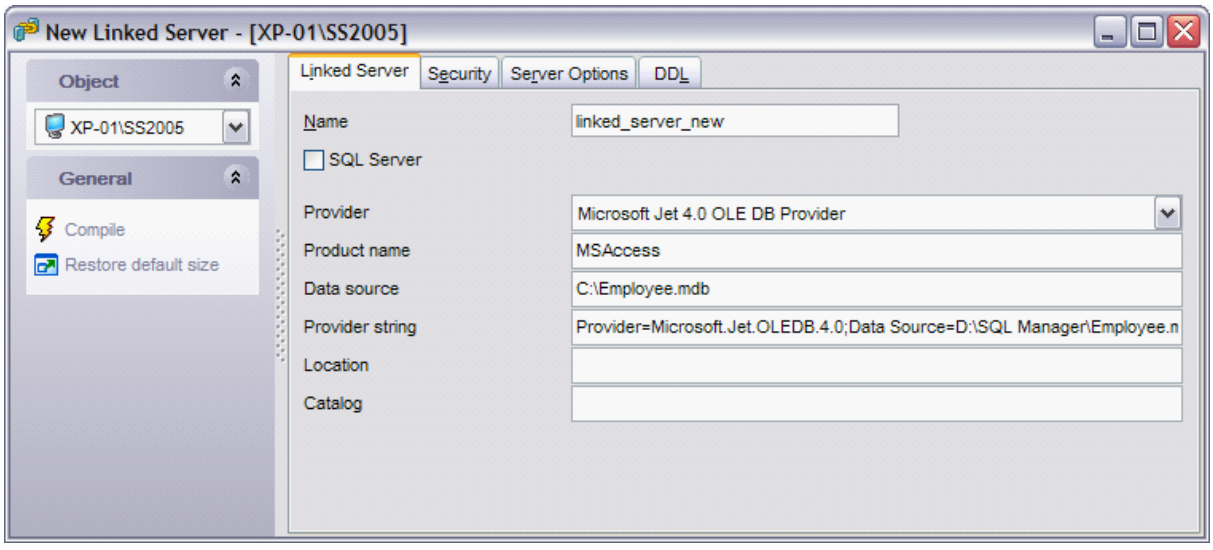
To open a linked server in **Linked Server Editor**, double-click it in the DB Explorer tree or in the Linked servers manager. Alternatively, you can right-click the linked server alias and select the **Edit Linked Server** context menu item.

- Configuring Linked server
- Security settings
- Setting server options
- Linked Server content
- Viewing DDL definition

10.2.3.2.1 Configuring Linked server

Use the **Linked Server** tab of **Linked Server Editor** to configure/view linked server properties.

Name
Enter a name for the new linked server, or view the name of the linked server being edited.



© 2008 EMS Database Management Solutions

SQL Server
Select this option to identify the linked server as an instance of Microsoft® SQL Server™. If you use this method of defining a SQL Server linked server, the name of the linked server must be the network name of the server. Also, any tables retrieved from the server are from the default database defined for the login on the linked server.

Provider
Use the drop-down list to select the name of the OLE DB Provider to be used for access to the specified linked server.

Product name
Specify the product name of the OLE DB data source you want to add as a linked server.

Data source
Specify the OLE DB data source property corresponding to the linked server.

Provider string
Specify the OLE DB provider string corresponding to the linked server.

Location
Specify the OLE DB location property corresponding to the linked server.

Catalog
Specify the OLE DB catalog property corresponding to the linked server.

10.2.3.2.2 Security settings

The **Security** tab provides security settings used for the specified linked server connection.

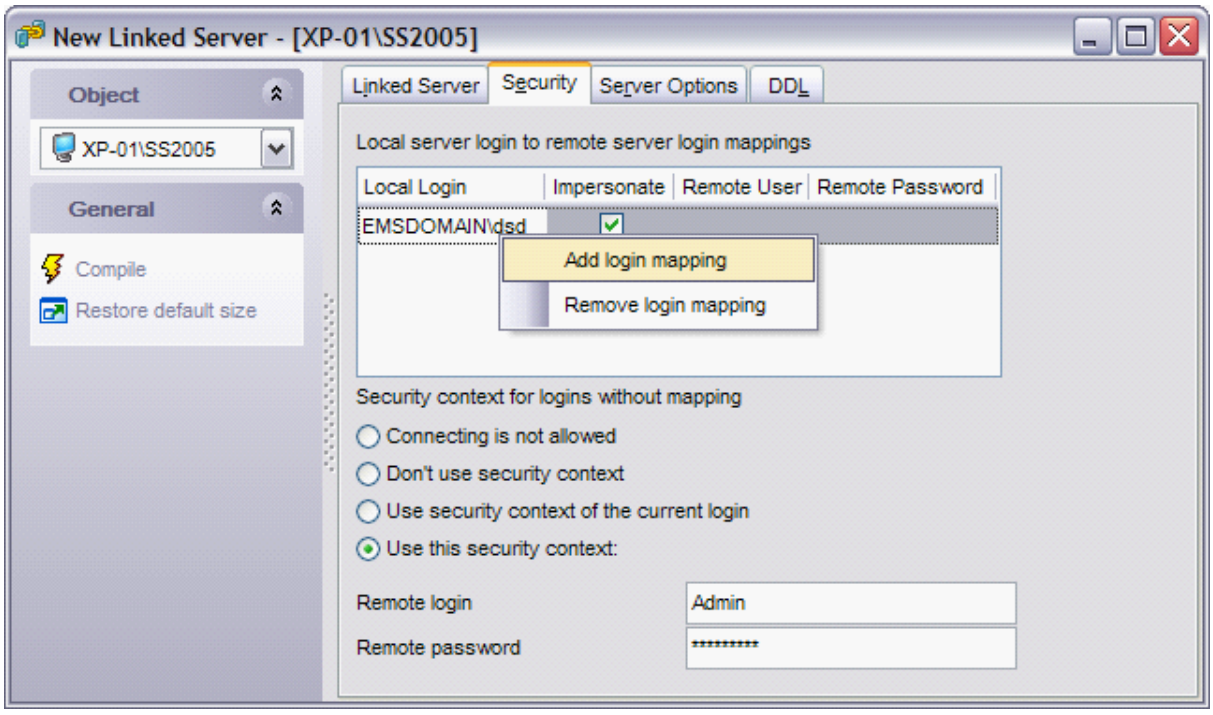
Local Login
Specify the local login IDs that can connect to the linked server.

Impersonate
Set this flag to specify that the local login ID will be used to connect to the linked server. Select this option if you are certain that the local login ID exactly matches a login ID with sufficient permissions on the linked server.

Remote user
Use the remote user to map the users that are not defined in **Local login**.

Remote password
Specify the password used to map the users that are not defined in **Local login**.

© 2008 EMS Database Management Solutions



Connecting is not allowed
Select this option to specify that the connection will not be established for logins that are not defined in the mapping list.

Don't use security context
Select this option to specify that for logins that are not defined in the list, the connection will be established without using a security context.

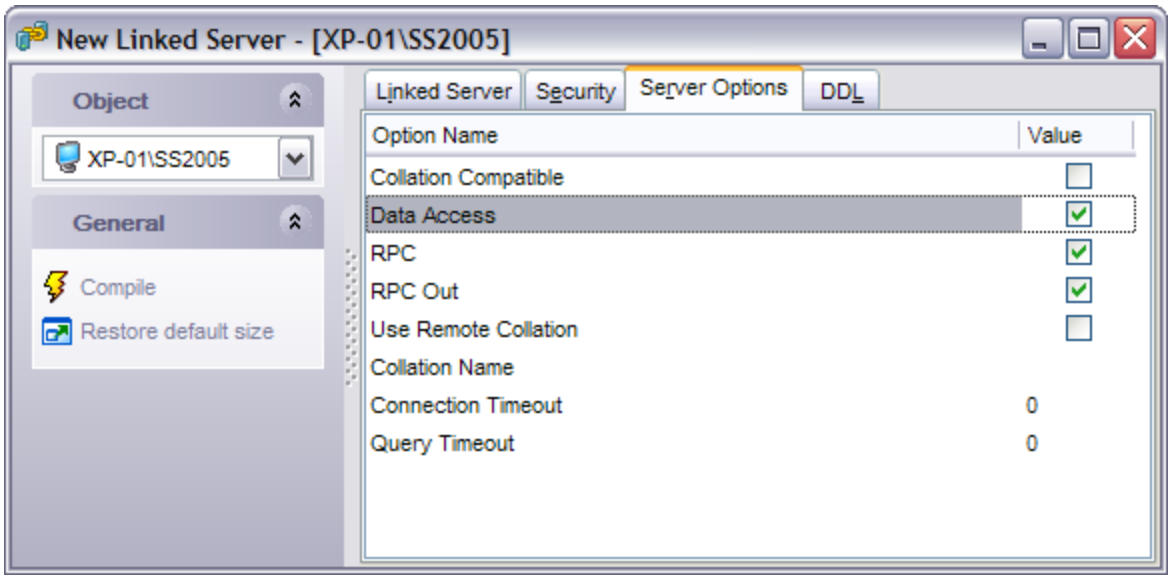
Use security context of the current login
Select this option to specify that for logins that are not defined in the list, the connection will be established using the security context of the current login.

Use this security context
Select this option to specify that for logins that are not defined in the list, the connection will be established using the login and password specified in the **Remote Login** and **Remote password** boxes below.

10.2.3.2.3 Setting server options

The **Security** tab allows you to set server options that will be used for the specified linked server connection.

© 2008 EMS Database Management Solutions



Collation Compatible
If this option is enabled, SQL Server 2005 assumes that all characters in the linked server are compatible with the local server, with regard to character set and collation sequence (or sort order). This enables SQL Server to send comparisons on character columns to the provider.
This option should be set only if it is certain that the data source corresponding to the linked server has the same character set and sort order as the local server.

Data Access
This option enables/disables a linked server for distributed query access.

RPC
This option enables/disables RPC from the given server.

RPC Out
This option enables/disables RPC to the given server.

Use Remote Collation
This option determines whether the collation of a remote column or of a local server will be used.
If checked, the collation of remote columns is used for SQL Server data sources, and the collation specified in collation name is used for non-SQL Server data sources.
If unchecked, distributed queries will always use the default collation of the local server, while collation name and the collation of remote columns are ignored.

Collation Name
This drop-down list allows you to specify the name of the collation used by the remote data source if use remote collation is true and the data source is not a SQL Server data source.

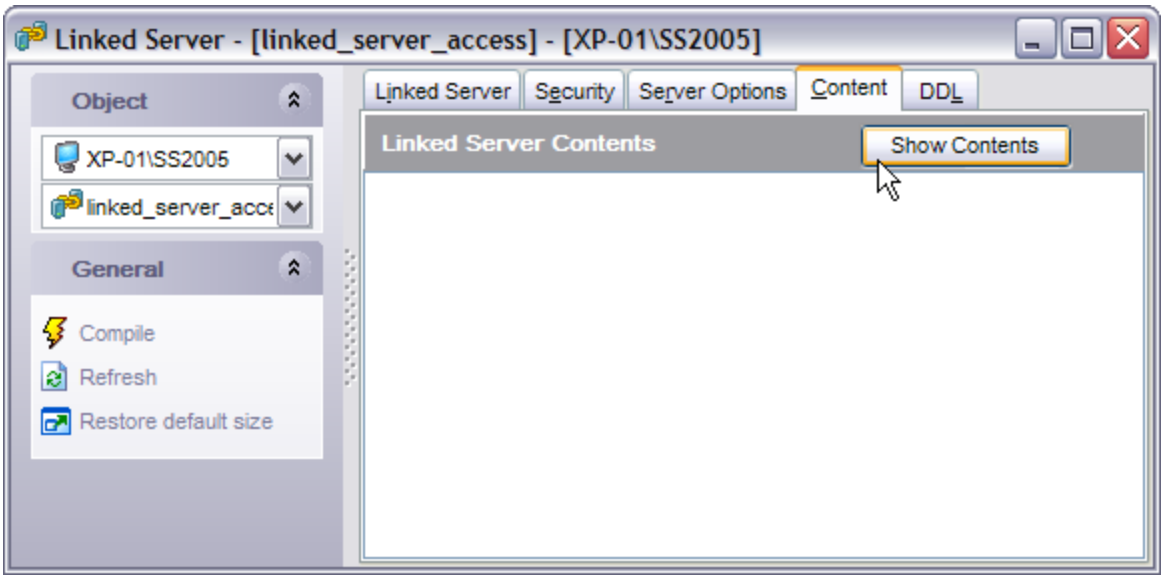
Connection Timeout
Use this edit-box to set the time-out value for connecting to the linked server.

© 2008 EMS Database Management Solutions

Query Timeout
Use this edit-box to set the time-out value for queries against the linked server.

10.2.3.2.4 Linked Server content

The **Content** tab allows you to browse the contents of the linked server.



To display the actual linked server contents, press the **Show Contents** button.

10.2.4 DTS Packages

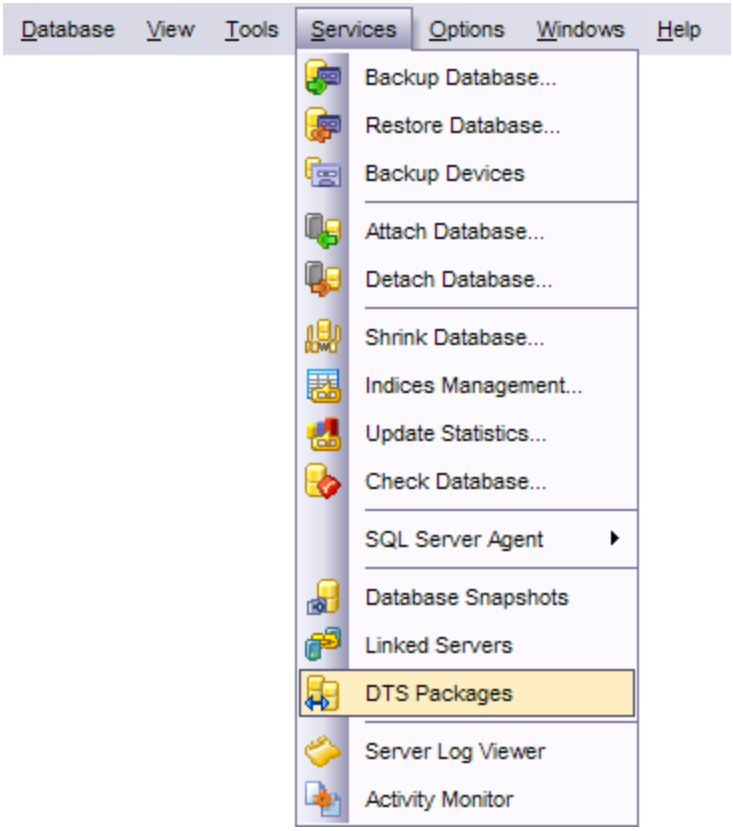
In SQL Server 2000, **Data Transformation Services** (**DTS**) is a component built to take data from one OLE DB data source, perform certain operations and store it in a destination OLE DB data source. DTS consists of **packages** which define a particular set of work that forms a logical work item. **DTS Packages** contain multiple connections to data sources, tasks to be performed, workflows. Examples of tasks include copying data from source to destination connections, transforming data from a source connection and placing the transformed data in the destination connection, executing a set of Microsoft ActiveX scripts or Transact-SQL statements against a connection.

In SQL Server 2005, **Integration Services** (**SSIS**) packages are provided for data warehousing purposes.

SQL Manager 2008 for SQL Server provides running **Data Transformation Services** (**DTS**) packages that were developed by using SQL Server 2000 tools. These can be run in SQL Server 2005, alongside **Integration Services** (**SSIS**) packages.

- DTS packages manager
- DTS package properties

© 2008 EMS Database Management Solutions



See also:

Backup Devices
Database Snapshots
Linked Servers
Jobs
Job History
Alerts
Operators
Schedules
Proxies
Credentials
Endpoints
DDL Triggers
Target Servers
Server Log Viewer
Activity Monitor

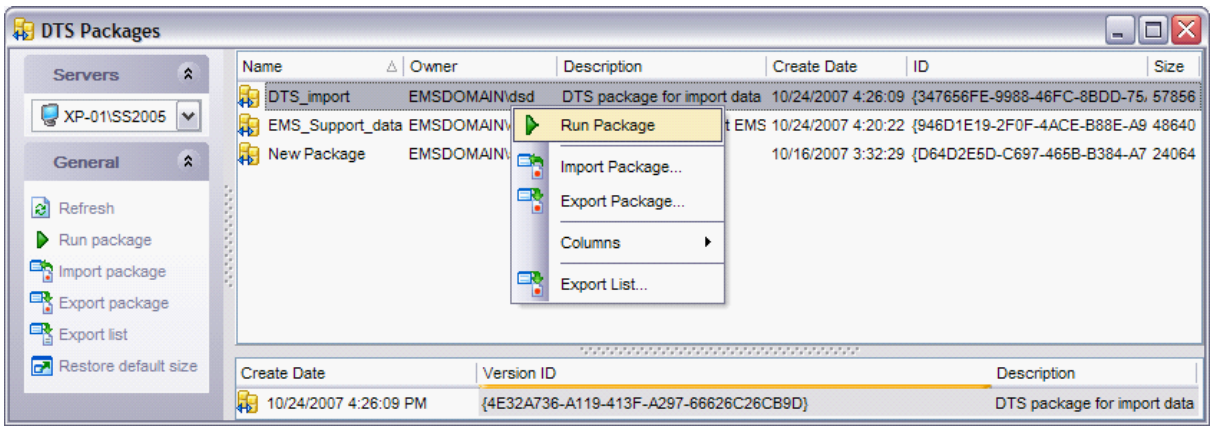
© 2008 EMS Database Management Solutions

10.2.4.1 DTS Packages manager

The **DTS Packages manager** allows you to browse the list of **DTS packages** and manage them efficiently.

Using this tool you can include SQL Server 2000 DTS packages in SQL Server 2005 data transformation solutions. A package can include both Execute Package tasks and Execute DTS 2000 Package tasks because each type of task uses a different version of the run-time engine.

To launch the tool, select the **Services | DTS Packages** main menu item, or right-click the host alias in the DB Explorer tree and select the **Tasks | DTS Packages** item from the context menu.



The list displays the existing DTS packages as a grid with the following columns: *Name*, *Owner*, *Description*, *Create Date*, *ID*, *Size*.

Right-click an item within the list to call the **context menu** allowing you to *run* a package, *import* a package, *export* the selected DTS package, or show/hide columns of the list. Using the context menu you can also export the list of DTS packages to any of supported output file formats.

DTS packages management tasks are also available through the **Navigation bar** of the **DTS Packages manager**.

At the bottom of the **DTS Packages manager** window the info area is located. It displays common information pertaining to the selected package: *creation date*, *version ID* and *description*.

Hint: These properties are also available within the modal DTS Package properties dialog which is called through the **DTS Package Properties...** context menu item of the DTS package alias in DB Explorer.

If you need to remove a package, right-click the package in the information area and select the **Delete** popup menu item.

© 2008 EMS Database Management Solutions

Availability:

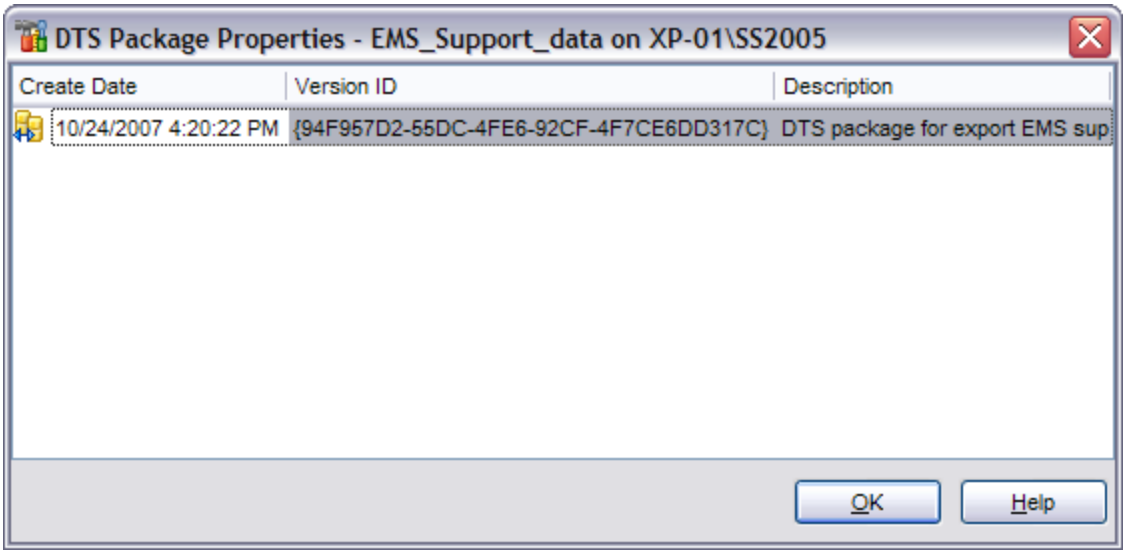
Full version (for Windows)	**Yes**
Lite version (for Windows)	**No**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

10.2.4.2 DTS Package properties

The **DTS Package Properties** dialog allows you to browse the **DTS package** common information.

To open the dialog, right-click the DTS package in DB Explorer and select the **DTS Package Properties...** context menu item.



The dialog displays common properties pertaining to the DTS package: *Create Date*, *Version ID*, *Description*.

10.2.5 Jobs

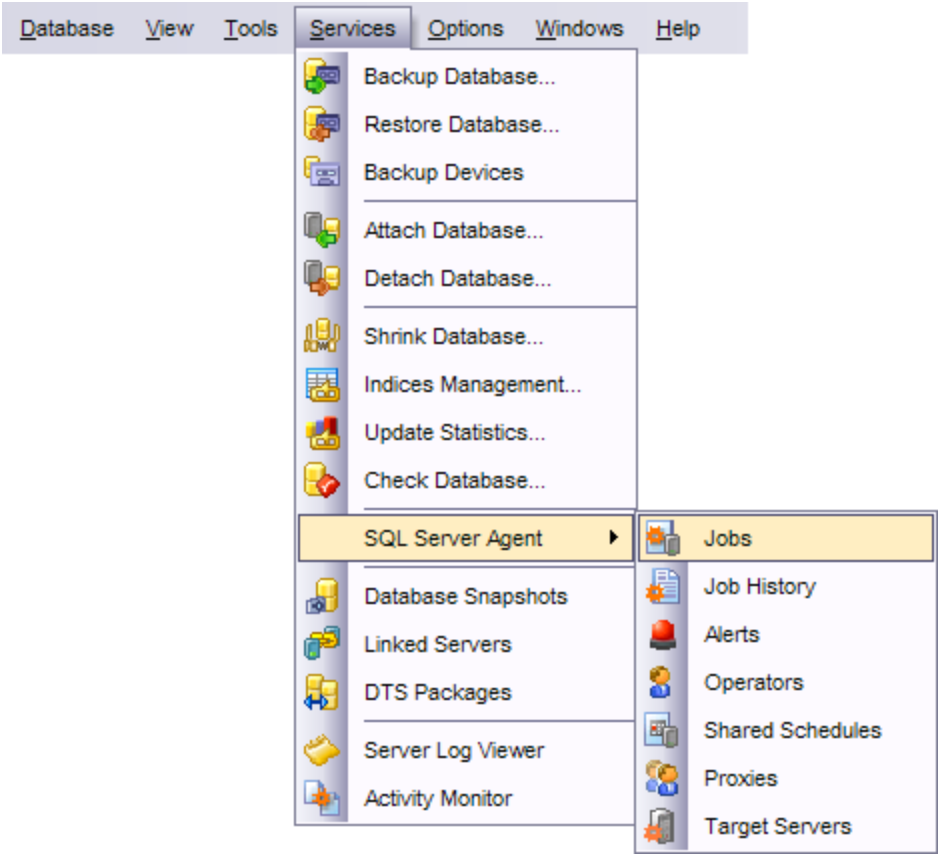
A **job** is a specified set of operations performed sequentially by SQL Server Agent. A job can perform a wide range of activities, including running Transact-SQL scripts, command-line applications, Microsoft ActiveX scripts, Integration Services packages, Analysis Services commands and queries, or Replication tasks. Jobs can run repetitive or scheduled tasks, and they can automatically notify operators of job status by generating alerts.

SQL Server Agent **jobs** can be used to simplify SQL Server administration and make it more efficient by automating routine administrative tasks.

© 2008 EMS Database Management Solutions

Before using SQL Server jobs, make sure that the **SQL Server Agent** service is running.

- Jobs manager
- Job Editor



Creating Jobs

To create a new job:

- right-click the **Jobs** node (within the **Server Objects** branch) or any object within this node in the DB Explorer tree and select the **New Job** item from the context menu;
- define job properties using the appropriate tabs of Job Editor.

or

- select the **Database | New Object...** main menu item;
- select **Job** in the Create New Object dialog;
- define job properties using Job Editor.

Hint: To create a new job, you can also select the **Services | SQL Server Agent | Jobs** main menu item to open the **Jobs manager** and select the **Add Job...** item from the context menu or on the Navigation bar.

To create a new job with the same properties as one of existing jobs has:

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

Alternatively, you can right-click a job in the DB Explorer tree and select the **Duplicate Job <job_name>...** context menu item.

© 2008 EMS Database Management Solutions

Duplicate Object Wizard allows you to select the server to create a new job and to edit the result SQL statement for creating the job.

Editing Jobs

To edit an existing job:
- select the job for editing in the DB Explorer tree (type the first letters of the job name for quick search);
- right-click the object and select the **Edit Job <job_name>** context menu item, or simply double-click the job;
- edit job properties using the appropriate tabs of Job Editor.

To change the name of a job:
- select the Job to rename in the DB Explorer tree;
- right-click the job alias and select the **Rename Job <job_name>...** item from the context menu;
- edit the job name using the **Rename Object...** dialog.

Dropping Jobs

To drop a job:
- select the job to drop in the DB Explorer tree;
- right-click the object and select the **Drop Job <job_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new job;
Ctrl+O to edit the selected job;
Ctrl+R to rename the job;
Shift+Del to drop the object.

To create, edit, rename and drop jobs, you can also use the **context menu** and the **Navigation bar** of Jobs manager.

See also:
Backup Devices
Database Snapshots
Linked Servers
DTS Packages
Job History
Alerts
Operators
Schedules
Proxies
Credentials
Endpoints
DDL Triggers

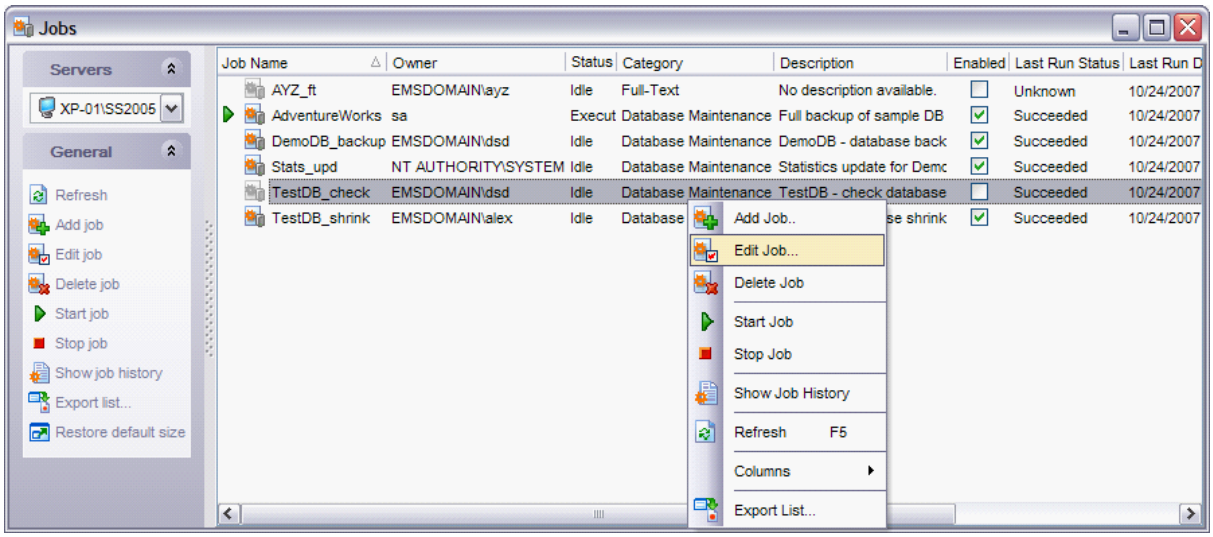
© 2008 EMS Database Management Solutions

Target Servers
Server Log Viewer
Activity Monitor

10.2.5.1 Jobs manager

The **Jobs manager** allows you to browse the list of **jobs** and manage them efficiently.

To launch the tool, select the **Services | SQL Server Agent | Jobs** main menu item, or right-click the host alias in the DB Explorer tree and select the **Tasks | Jobs** item from the context menu.



The list displays the existing jobs as a grid with the following columns: *Job Name*, *Owner*, *Status*, *Category*, *Description*, *Enabled*, *Last Run Status*, *Last Run Date*, *Next Run Date*, *Schedules*.

Hint: Enabled/disabled jobs are differentiated in the list by their icons: icons for disabled jobs are grayed out.

Note: Job events and properties are partially listed within the Job tab of Job Editor.

Right-click an item within the list to call the **context menu** allowing you to *create* a new job and specify its properties using Job Editor, *start* an idle job, *stop* a running job, *edit*, *delete* the selected job, show job history, *refresh* the list, or show/hide columns of the list. Using the context menu you can also export the list of jobs to any of supported output file formats.

Jobs management tasks are also available through the **Navigation bar** of the **Jobs manager**.

© 2008 EMS Database Management Solutions

Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**No**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

10.2.5.2 Job Editor

Job Editor allows you to define job properties. It opens automatically when you create a new job and is available on editing an existing one (see Create job and Edit job for details).

To open a job in **Job Editor**, double-click it in the DB Explorer tree or in the Jobs manager. Alternatively, you can right-click the job alias and select the **Edit Job** context menu item.

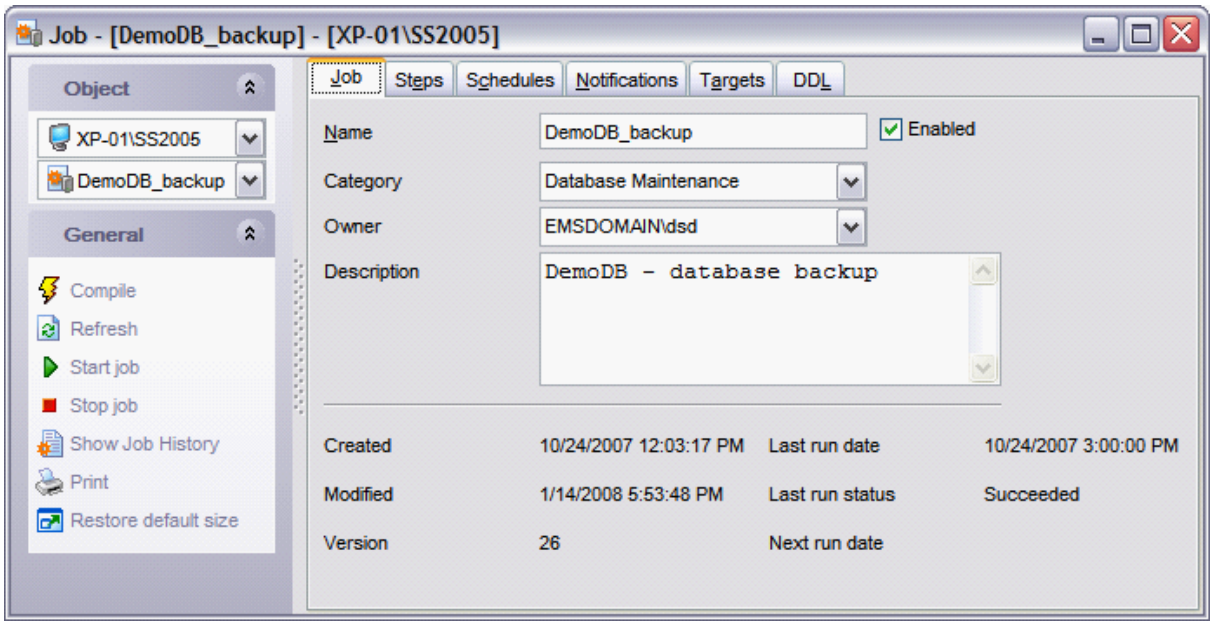
- Editing job definition
- Managing job steps
- Managing job schedules
- Managing notifications
- Managing target servers
- Viewing DDL definition

10.2.5.2.1 Editing job definition

Use the **Job** tab of **Job Editor** to configure/view job properties. If necessary, you can use items of the **Navigation bar** to manage the currently opened job (see Jobs manager).

Name
Enter a name for the new job, or modify the name of the job being edited. Enable/disable the job using the **Enabled** option.

© 2008 EMS Database Management Solutions



Category
Use the drop-down list to select the job category.
Job categories help you organize your jobs for easy filtering and grouping. For example, you can organize all your database backup jobs in the *Database Maintenance* category.

Owner
Use the drop-down list to select the job owner.
Note: To create a job, a user must be a member of one of the SQL Server Agent fixed database roles or the sysadmin fixed server role. A job can be edited only by its owner or members of the sysadmin role.

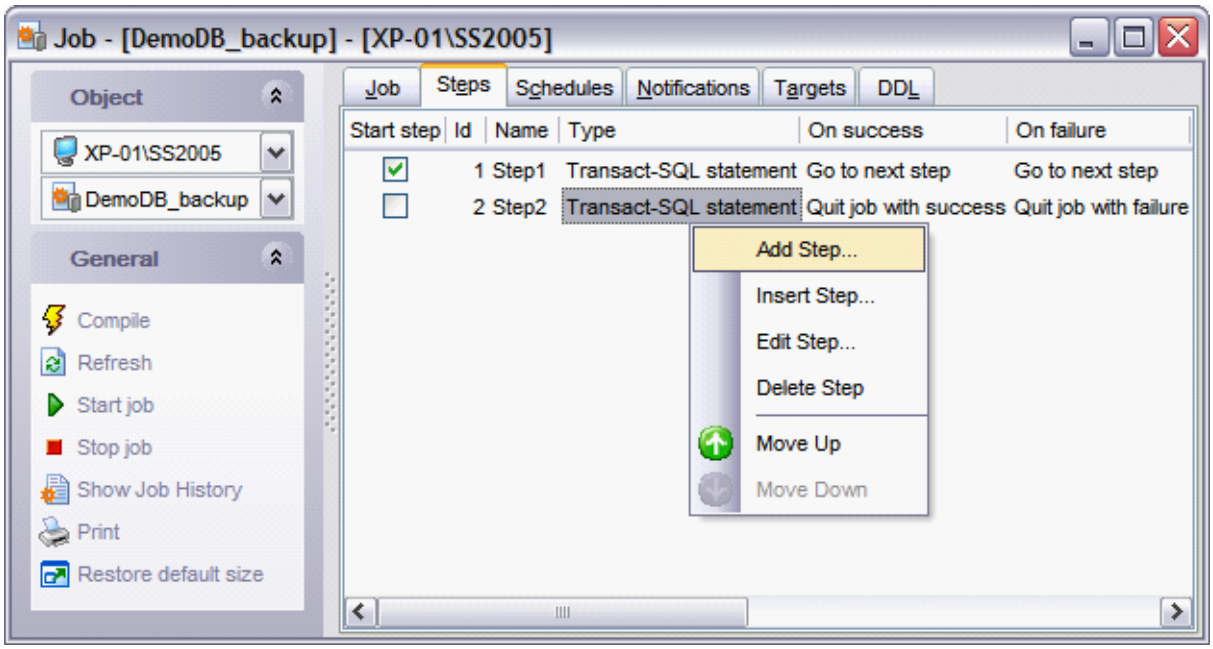
Description
In this memo field you can supply an optional description for the job.

The lower area displays the following job events and properties:
Created
Modified
Version
Last run date
Last run status
Next run date

10.2.5.2.2 Managing job steps

Use the **Steps** tab of **Job Editor** to manage job steps (operations). 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.

© 2008 EMS Database Management Solutions



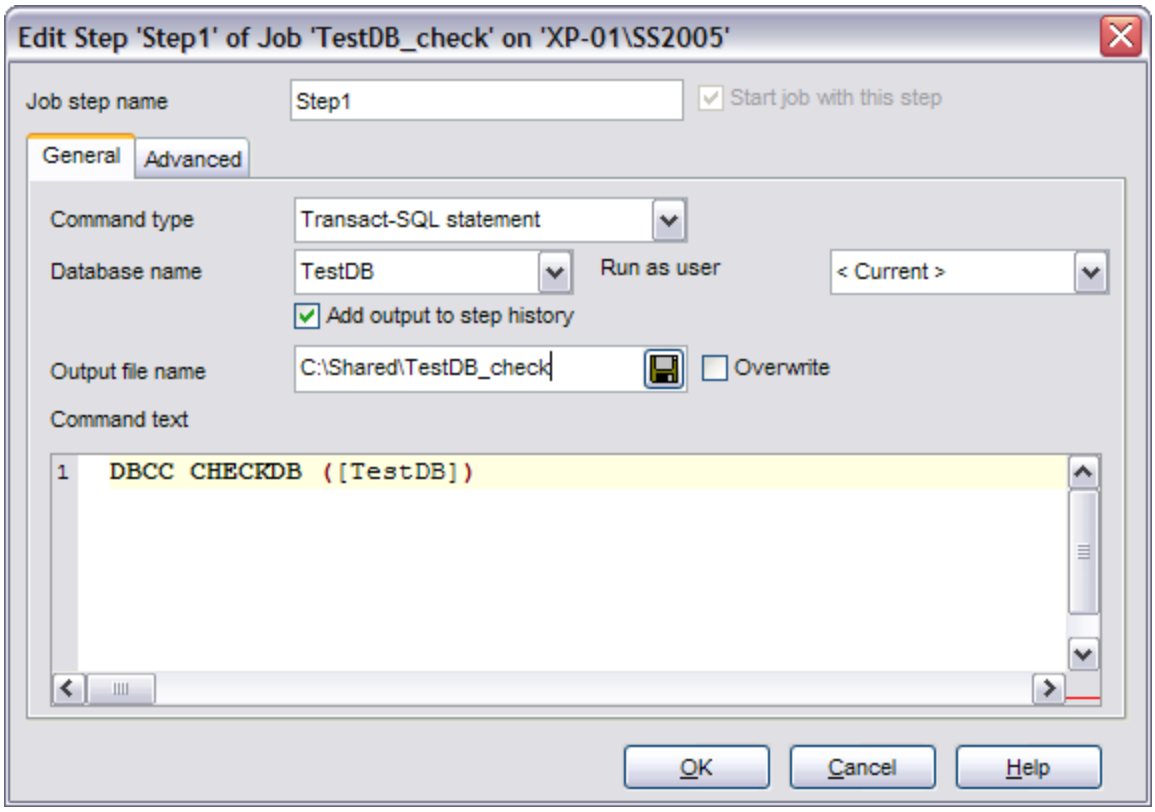
The current job steps are displayed as a grid with the following columns: *Start step*, *ID*, *Name*, *Type*, *On success*, *On failure*, *Last run outcome*, *Last run date*, *Command*.

Right-click an item within the list to call the **context menu** allowing you to *create* a new job step and specify its properties using the **Edit Step** dialog, *insert* a step, *edit* or *delete* the selected step.

The **Add New Job Step / Edit Step** dialog allows you to define job step parameters.

The **General** tab allows you to define basic parameters of the step.

© 2008 EMS Database Management Solutions



Job step name
Enter a name for the new step, or modify the name of the step being edited. If necessary, you can specify to **Start job with this step**. Note that a job can have only one start step.

Specify the **Command Type** of the step indicating the subsystem used by Microsoft SQL Server Agent service to execute the job command:

- *Transact-SQL statement*
- *Active Script*
- *Operating-system command or executable program*
- *Replication Distribution Agent job*
- *Replication Snapshot Agent job*
- *Replication Log Reader Agent job*
- *Replication Merge Agent job*

Use the **Database name** drop-down list to select the database for which the current step will be executed (for *Transact-SQL* command type step; for an *ActiveX* job step, you are to specify the name of the scripting language that the step uses), and specify the security context under which the command is executed using the **Run as user** drop-down list.

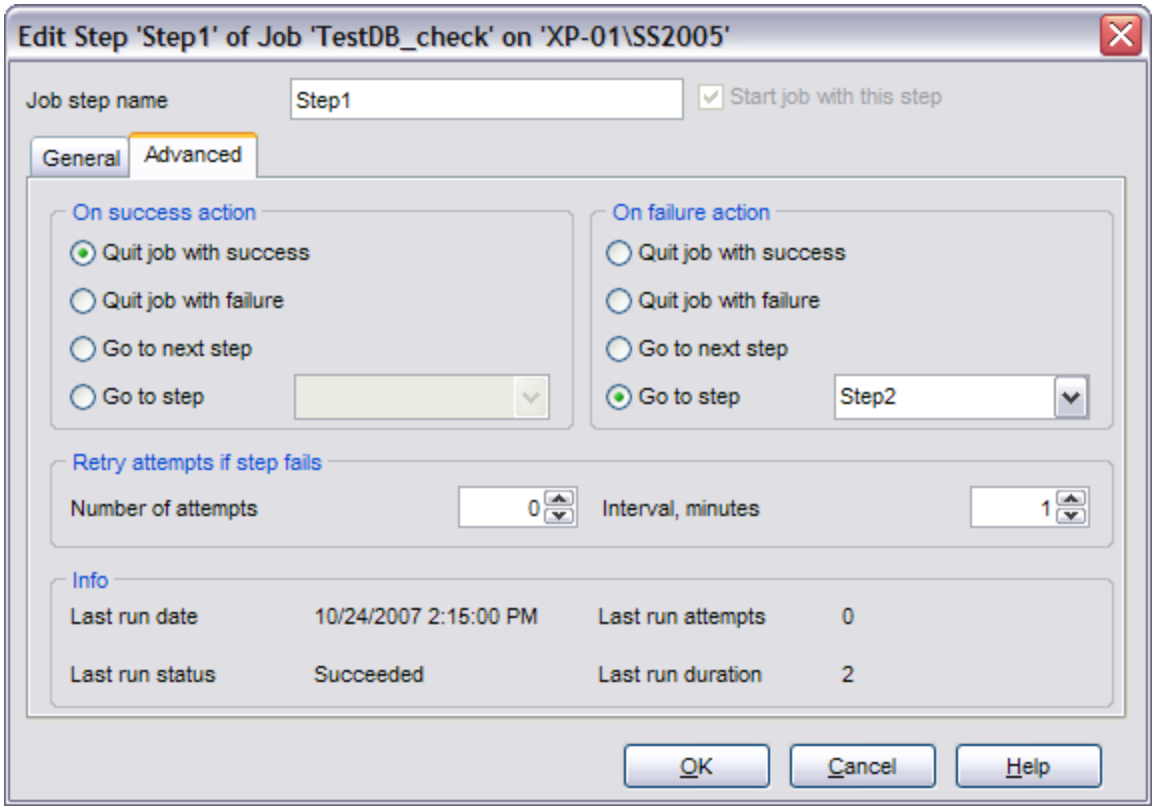
SQL Server Agent can write output from some job steps either to the sysjobsteplogs table in the msdb database or to an operating system file.
To use this feature, check the **Add output to step history** option and set the path

© 2008 EMS Database Management Solutions

and the file name in the **Output file name** field. You can select the **Overwrite** option to specify that the existing destination log file can be overwritten.

Use the **Command text** memo field to enter the job step command code: *SQL* code (for **Transact-SQL** step), *Visual Basic Script / Java Script / other script* code (for **Active Script** step), or *OS command / executable program* (for **Operating-system command** step).

The **Advanced** tab provides flexible job step control of flow for success and failure.



Specify the **On success action**, i.e. the action the job will take if the step succeeds:
- *Quit job with success*
- *Quit job with failure*
- *Go to next step*
- *Go to step ...* (select the step from the drop-down list)

Specify the **On failure action**, i.e. the action the job will take if the step fails:
- *Quit job with success*
- *Quit job with failure*
- *Go to next step*
- *Go to step ...* (select the step from the drop-down list)

The **Retry attempts if step fails** group allows you to set optional parameters for the job step: the *Number of attempts* and the *Interval* between attempts (in minutes).

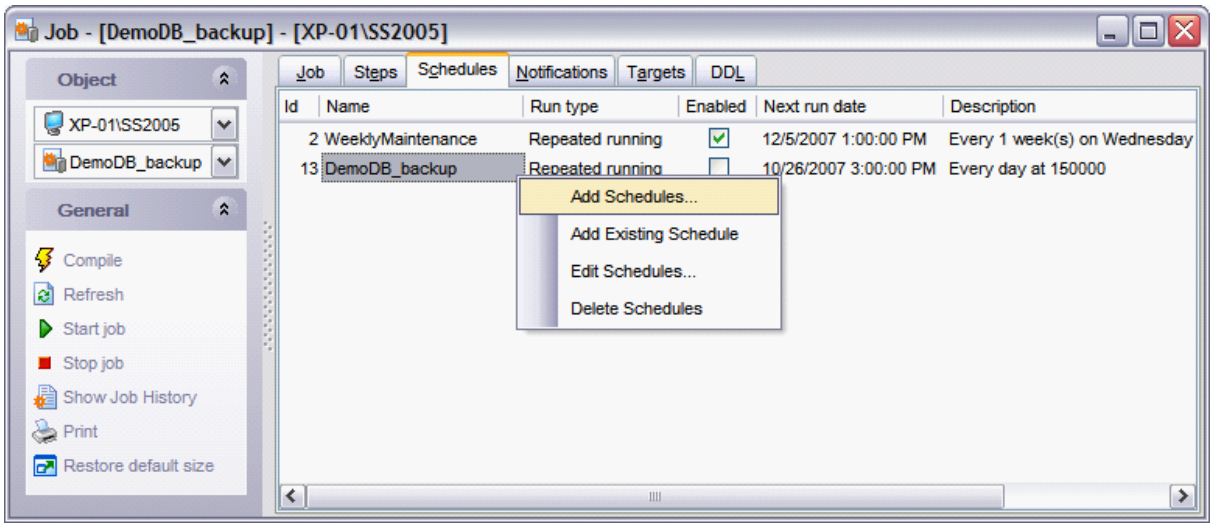
© 2008 EMS Database Management Solutions

The **Info** group contains common information about the job step:
Last run date
Last run status
Last run attempts
Last run duration

10.2.5.2.3 Managing job schedules

The **Schedules** tab allows you to manage schedules assigned to the job.

Scheduling administrative jobs consists in defining the condition(s) that cause the job to begin running. It is possible to schedule any type of job. More than one job can use the same job schedule.

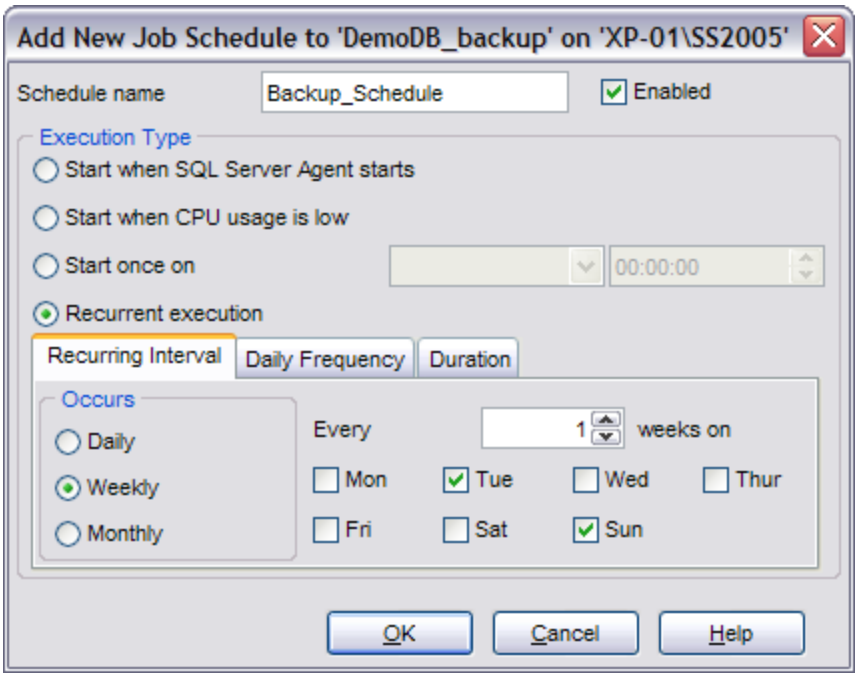


The current job schedules are displayed as a grid with the following columns: *ID*, *Name*, *Run type*, *Enabled*, *Next Run Date*, *Description*.

Right-click an item within the list to call the **context menu** allowing you to *create* a new job schedule and specify its properties using the **Add New Job Schedule** dialog, *add an existing schedule*, *edit* or *delete* the selected schedule.

The **Add New Job Schedule / Edit Schedule** dialog allows you to define schedule parameters.

© 2008 EMS Database Management Solutions



Schedule name
Enter a name for the new schedule, or modify the name of the schedule being edited. Enable/disable the schedule using the **Enabled** option.

Specify the **Execution Type** of the schedule indicating when the job is to be executed:

- *Start when SQL Server Agent starts*
- *Start when CPU usage is low*
- *Start once on...*
- *Recurrent execution*

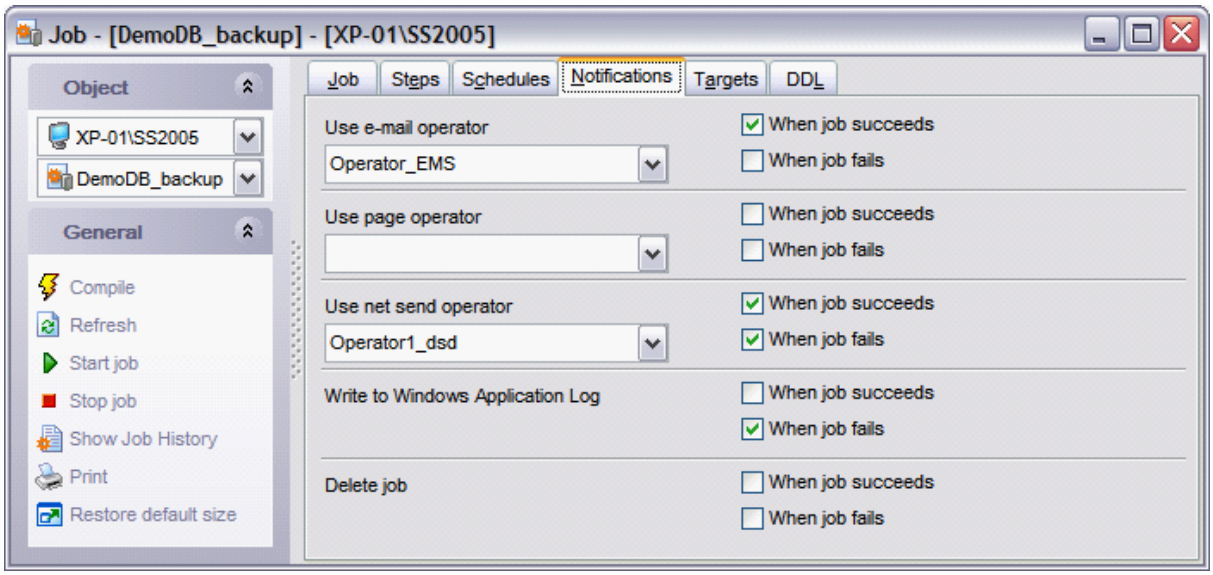
If the **Recurrent execution** type is selected, you should also set the **Recurring Interval**, **Daily Frequency** and **Duration** properties of the current job schedule.

For details see Schedule Editor.

10.2.5.2.4 Managing notifications

The **Notifications** tab allows you to specify actions that the SQL Server Agent service will take after the job completes (succeeds or fails). Job responses ensure that database administrators know when jobs complete and how frequently they run.

© 2008 EMS Database Management Solutions



Use e-mail operator
Use the drop-down list to select the operator that will be notified via *e-mail*, and specify the job status for the notification to be performed:
☑ *When job succeeds* and/or
☑ *When job fails.*

Use page operator
Use the drop-down list to select the operator that will be notified using *electronic paging*, and specify the job status for the notification to be performed:
☑ *When job succeeds* and/or
☑ *When job fails.*

Use net send operator
Use the drop-down list to select the operator that will be notified using a *net send* message, and specify the job status for the notification to be performed:
☑ *When job succeeds* and/or
☑ *When job fails.*

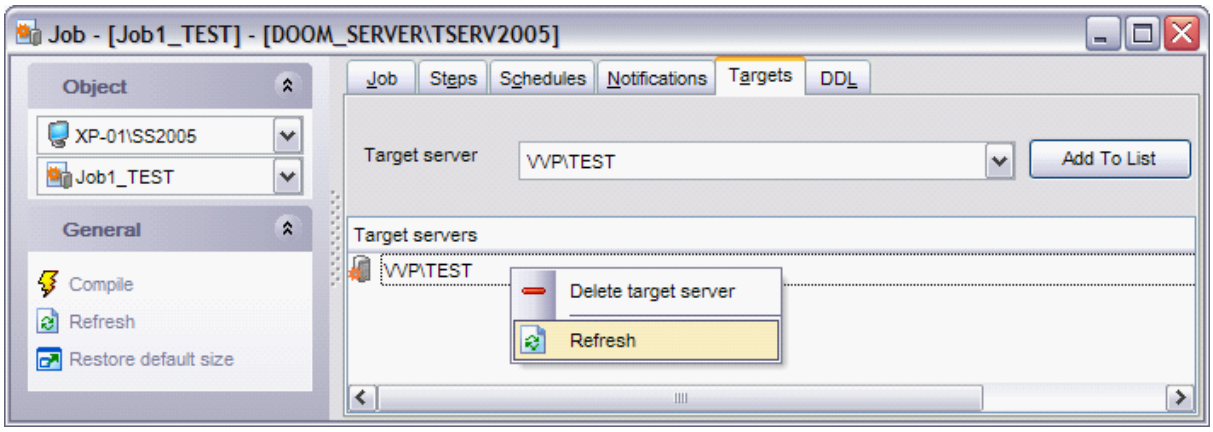
Write to Windows Application Log
If necessary, specify the job status for writing to Microsoft Windows Application Log:
☑ *When job fails.*

Delete job
If you are certain that you do not need to rerun this job later, you can specify the job status deleting the job:
☑ *When job succeeds* and/or
☑ *When job fails.*

10.2.5.2.5 Managing target servers

The **Targets** tab allows you to target the job at the specified target server.

© 2008 EMS Database Management Solutions



Target server
Use the drop-down list to select the target server and press the **Add To List** button to add the selected server to the list of target servers available for the multiserver job operation.

Right-click an item within the **Target servers** area to call the **context menu** allowing you to *delete* the selected target server or *refresh* the list.

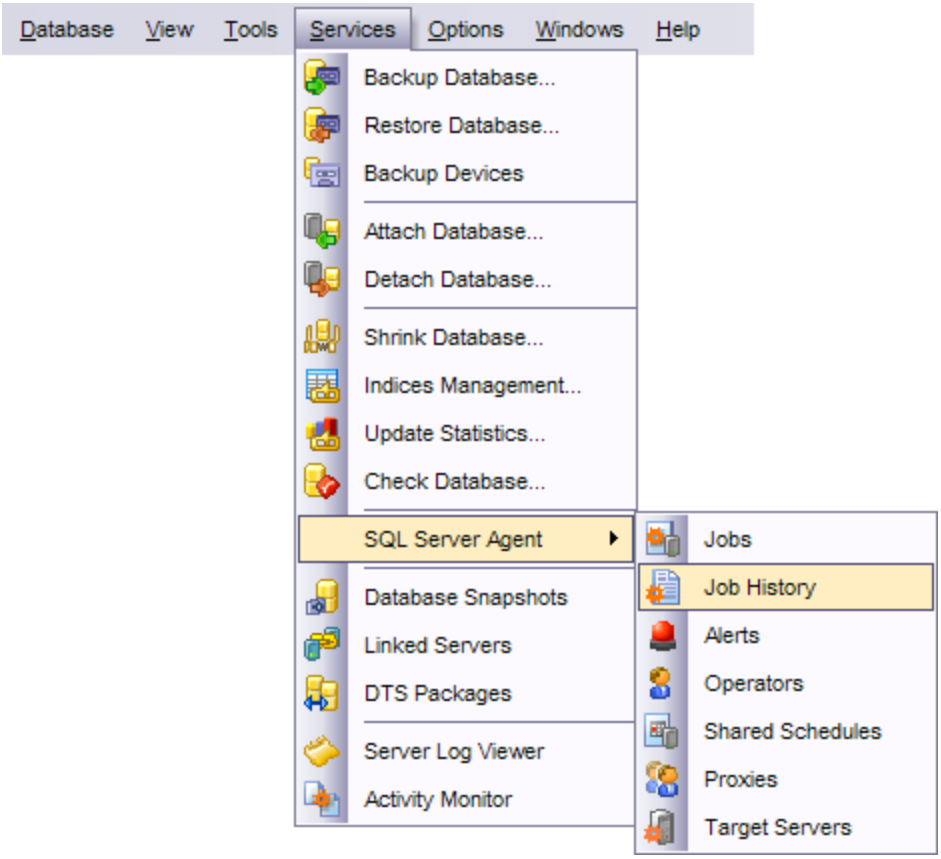
10.2.6 Job History

The **Job History** window allows you to view the list of recent jobs and the results of their execution.

Before using job history, make sure that the **SQL Server Agent** service is running.

- Viewing job history
- Filtering job history list

© 2008 EMS Database Management Solutions



Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**No**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

Backup Devices
Database Snapshots
Linked Servers
DTS Packages
Jobs
Alerts
Operators
Schedules
Proxies
Credentials

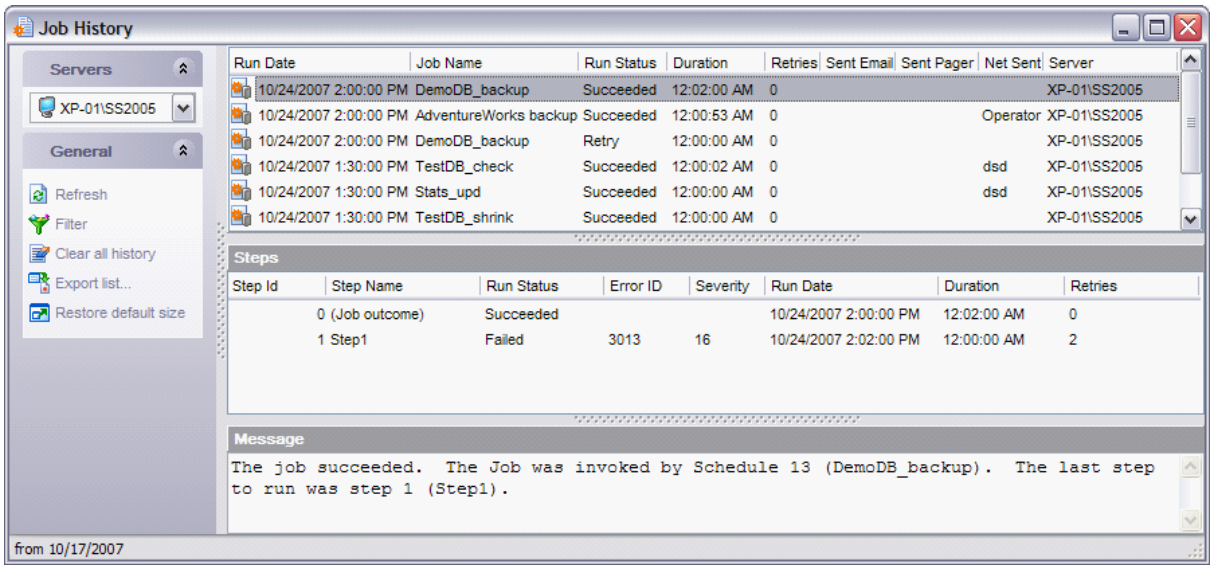
© 2008 EMS Database Management Solutions

Endpoints
DDL Triggers
Target Servers
Server Log Viewer
Activity Monitor

10.2.6.1 Viewing job history

The **Job History** window allows you to browse the list of recently tried **jobs** and manage this list efficiently.

To launch the tool, select the **Services | SQL Server Agent | Job History** main menu item, or right-click the host alias in the DB Explorer tree and select the **Tasks | Job History** item from the context menu.



The list displays the recent jobs as a grid with the following columns: *Run Date*, *Job Name*, *Run Status*, *Duration*, *Retries*, *Sent Email*, *Sent Pager*, *Net Sent*, *Server*.

Right-click an item within the list to call the **context menu** allowing you to filter jobs in list, *clear* the history of all jobs or of the selected job, *refresh* the list, or show/hide columns of the list. Using the context menu you can also export the job history list to any of supported output file formats, add, edit and delete jobs.

Job history list management tasks are also available through the **Navigation bar** of the **Job History** window.

The **Steps** area displays the list of steps defined for the selected job, with the following attributes pertaining to each step: *Step ID*, *Step Name*, *Run Status*, *Error ID*, *Severity*, *Run Date*, *Duration*, *Retries*.

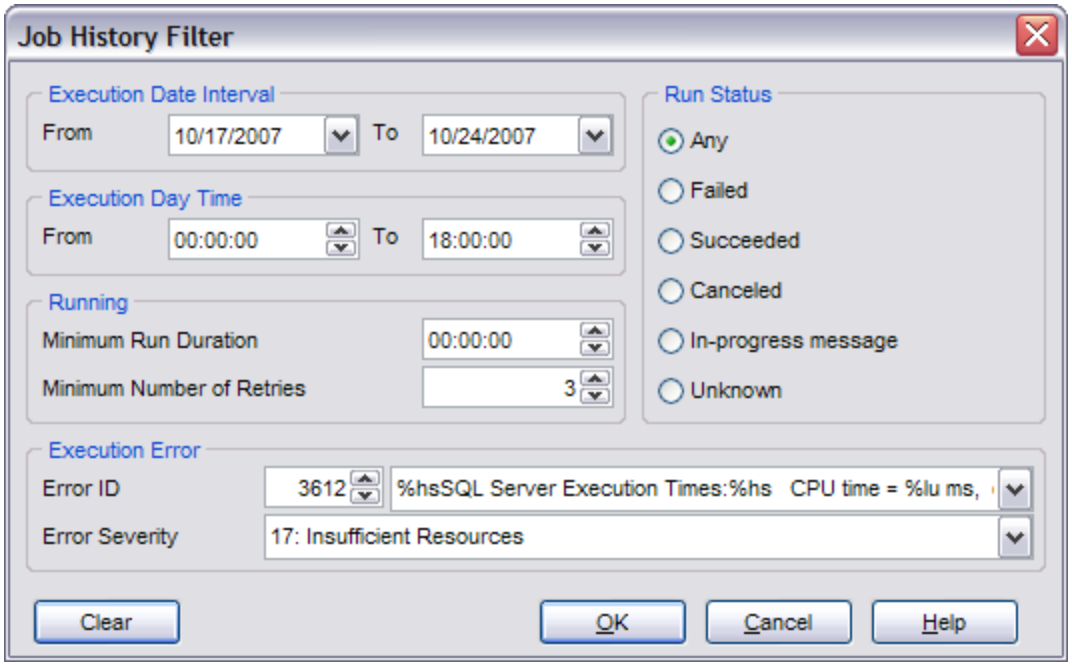
© 2008 EMS Database Management Solutions

At the bottom of the **Job History** window the **Message** area is located. It allows you to view the details of the selected step execution.

10.2.6.2 Filtering job history list

The **Job History Filter** dialog is a filtering facility implemented in **Job History** list for your convenience.

To open the dialog, right-click within the **Job History** list and select the **Filter...** context menu item, or use the corresponding item of the **Navigation bar**.



The **Job History Filter** dialog provides the following options for flexible **Job History** list filtering:

Execution Date Interval
From ... To ...
Set the date interval criteria using the **date editor** which is activated when you click the Arrow-Down element of the combo-box.

Execution Day Time
From ... To ...
Set the day time interval criteria using the spin-edit boxes.

Running
Minimum Run Duration
Minimum Number of Retries
Specify the minimum duration and number of job retries criteria using the spin-edit boxes.

Run Status

© 2008 EMS Database Management Solutions

Select the job run status criterion:
- *Any*
- *Failed*
- *Succeeded*
- *Canceled*
- *In-progress message*
- *Unknown*

Execution Error
This group allows you to specify criteria for the jobs that have been run with errors: *Error ID*, *Error message*, *Error Severity*.

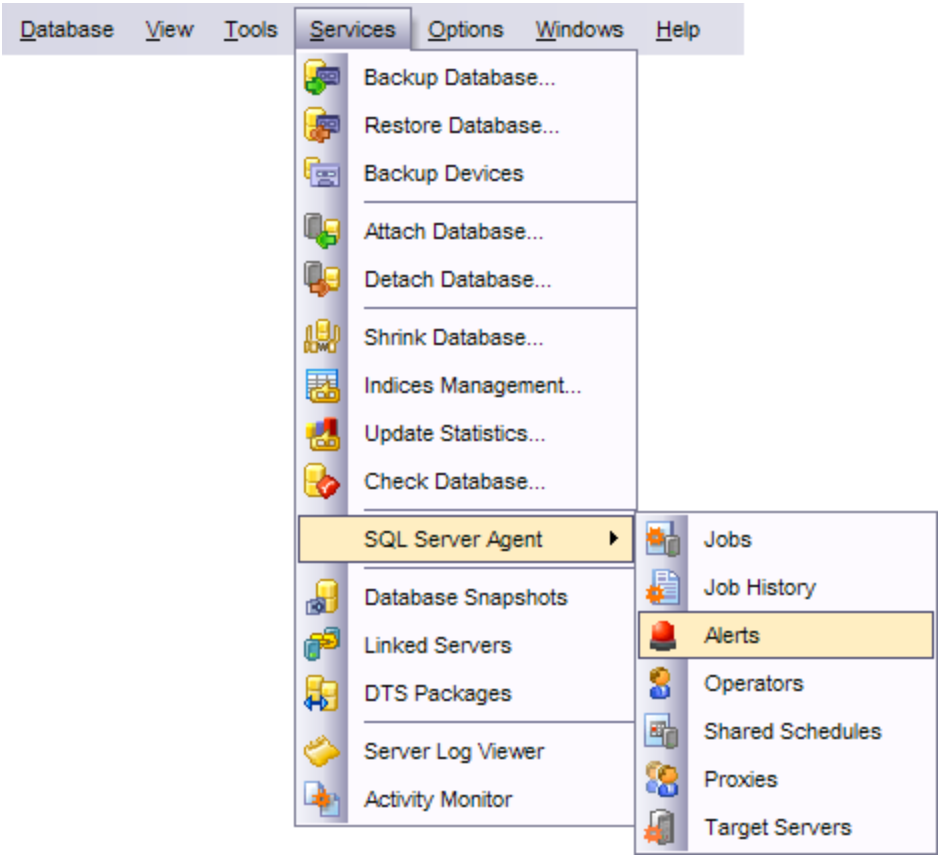
Hint: If necessary, you can reset the current filter criteria using the **Clear** button.

10.2.7 Alerts

An **alert** is as an automated response to one or more events. You can define an alert to specify how SQL Server Agent should respond to their occurrence. An alert can respond to an event by notifying an administrator or running a job, or both. An alert can also forward an event to the Microsoft Windows application log on a different computer. By defining alerts, database administrators can monitor and manage SQL Server more effectively.

Before using SQL Server alerts, make sure that the **SQL Server Agent** service is running.
- Alerts manager
- Alert Editor

© 2008 EMS Database Management Solutions



Creating Alerts

To create a new alert:

- right-click the **Alerts** node (within the **Server Objects** branch) or any object within this node in the DB Explorer tree and select the **New Alert** item from the context menu;
- define alert properties using the appropriate tabs of Alert Editor.

or

- select the **Database | New Object...** main menu item;
- select **Alert** in the Create New Object dialog;
- define alert properties using Alert Editor.

Hint: To create a new alert, you can also select the **Services | SQL Server Agent | Alerts** main menu item to open the **Alerts manager** and select the **Add Alert...** item from the context menu or on the Navigation bar.

To create a new alert with the same properties as one of existing alerts has:

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

Alternatively, you can right-click an alert in the DB Explorer tree and select the **Duplicate Alert <alert_name>...** context menu item.

Duplicate Object Wizard allows you to select the server to create a new alert and to edit the result SQL statement for creating the alert.

© 2008 EMS Database Management Solutions

Editing Alerts

To edit an existing alert:
- select the alert for editing in the DB Explorer tree (type the first letters of the alert name for quick search);
- right-click the object and select the **Edit Alert <alert_name>** context menu item, or simply double-click the alert;
- edit alert properties using the appropriate tabs of Alert Editor.

To change the name of an alert:
- select the alert to rename in the DB Explorer tree;
- right-click the alert alias and select the **Rename Alert <alert_name>...** item from the context menu;
- edit the alert name using the **Rename Object...** dialog.

Dropping Alerts

To drop an alert:
- select the alert to drop in the DB Explorer tree;
- right-click the object and select the **Drop Alert <alert_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new alert;
Ctrl+O to edit the selected alert;
Ctrl+R to rename the alert;
Shift+Del to drop the object.

To create, edit, rename and drop alerts, you can also use the **context menu** and the **Navigation bar** of Alerts manager.

See also:
Backup Devices
Database Snapshots
Linked Servers
DTS Packages
Jobs
Job History
Operators
Schedules
Proxies
Credentials
Endpoints
DDL Triggers
Target Servers
Server Log Viewer

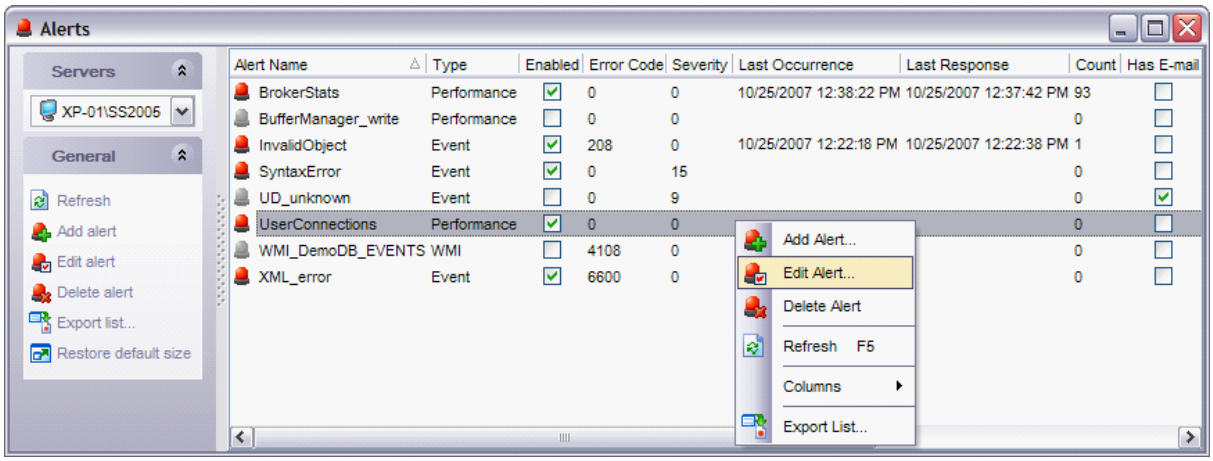
© 2008 EMS Database Management Solutions

Activity Monitor

10.2.7.1 Alerts manager

The **Alerts manager** allows you to browse the list of **alerts** and manage them efficiently.

To launch the tool, select the **Services | SQL Server Agent | Alerts** main menu item, or right-click the host alias in the DB Explorer tree and select the **Tasks | Alerts** item from the context menu.



The list displays the alerts as a grid with the following columns: *Alert Name*, *Type*, *Enabled*, *Error Code*, *Severity*, *Last Occurrence*, *Sent Pager*, *Net Sent*, *Server*.

Hint: Enabled/disabled alerts are differentiated in the list by their icons: icons for disabled alerts are grayed out.

Right-click an item within the list to call the **context menu** allowing you to *create* a new alert and specify its properties using Alert Editor, *edit*, *delete* the selected alert, *refresh* the list, or show/hide columns of the list. Using the context menu you can also export the list of alerts to any of supported output file formats.

Alerts management tasks are also available through the **Navigation bar** of the **Alerts manager**.

Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**No**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

© 2008 EMS Database Management Solutions

10.2.7.2 Alert Editor

Alert Editor allows you to define alert properties. It opens automatically when you create a new alert and is available on editing an existing one (see Create alert and Edit alert for details).

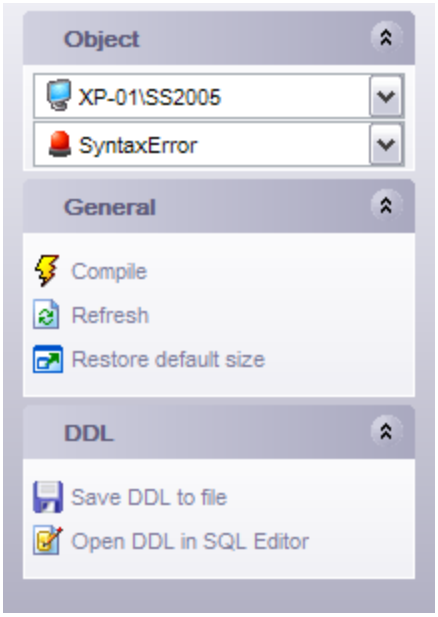
To open an alert in **Alert Editor**, double-click it in the DB Explorer tree or in the Alerts manager. Alternatively, you can right-click the alert alias and select the **Edit Alert** context menu item.

- Using Navigation bar
- Editing alert definition
- Setting additional properties
- Managing notifications
- Viewing DDL definition

10.2.7.2.1 Using Navigation bar

The **Navigation bar** of **Alert Editor** allows you to:

- select the server where the alert is located;
- select the alert for editing;
- compile the object (if it is being created/modified);
- refresh the content of the active tab;
- restore the default size of the editor window;
- save the alert DDL to an external **.sql* file;
- open the alert DDL in SQL Editor.



<< Back to Alert Editor

© 2008 EMS Database Management Solutions

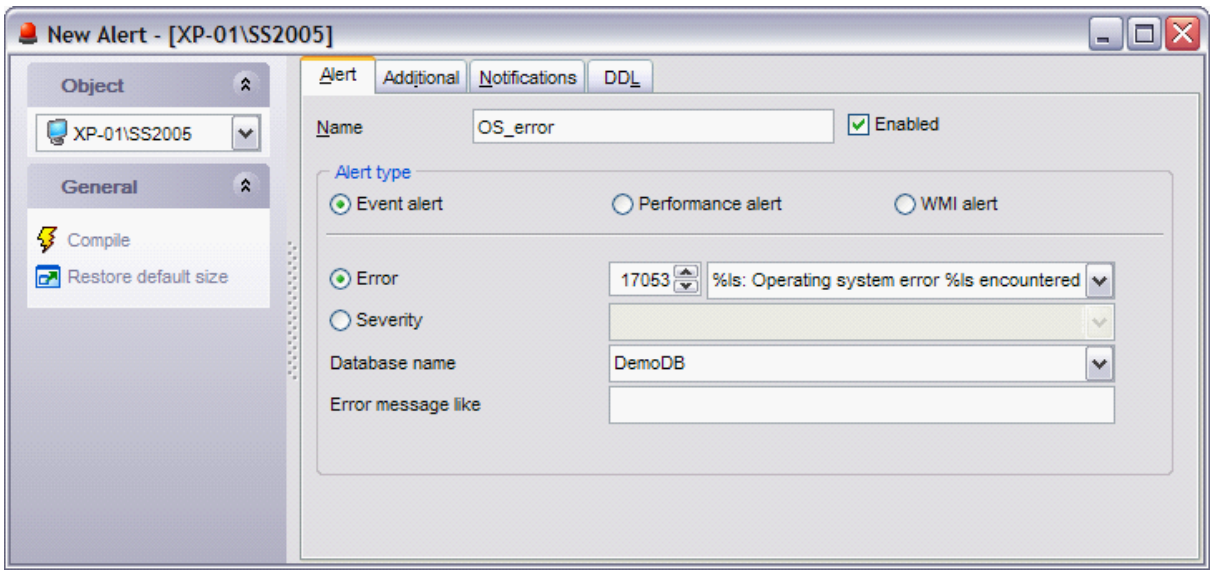
10.2.7.2.2 Editing alert definition

Use the **Alert** tab of **Alert Editor** to configure/view alert properties.

Name
Enter a name for the new alert, or modify the name of the alert being edited. Enable/disable the alert using the **Enabled** option.

An alert responds to an event of a specific type: *server event*, *server performance condition* or *WMI event*.

Event alert
Specify properties for the alert that responds to a *server event*.



Use the following parameters to specify the events that will trigger the alert:

Error
Select this option to specify that SQL Server Agent will fire the alert when a specific error occurs. Use the spin-edit box to specify the *Error ID* and the drop-down list to select the *Error message*.

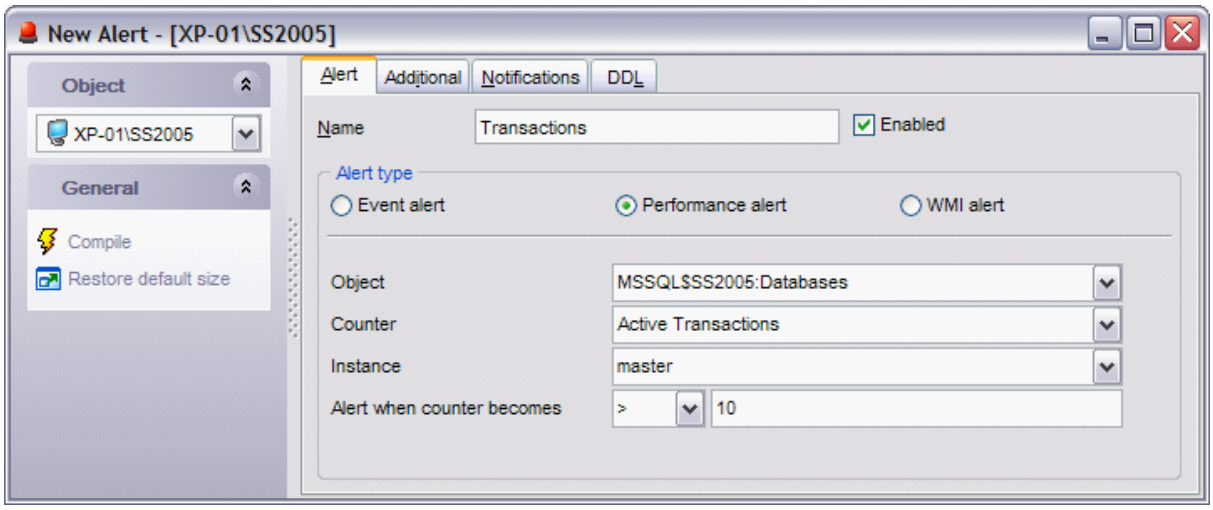
Severity
Select this option to specify that SQL Server Agent will fire the alert when any error of the specific severity occurs. Use the drop-down list to select the *Severity level*: 01-25. For example, you can specify a severity level of 15 to respond to syntax errors in Transact-SQL statements.

Database name
SQL Server Agent fires an alert only when the event occurs in a particular database. Select a database from the list of databases that reside on the specified SQL Server instance (see Using Navigation bar).

© 2008 EMS Database Management Solutions

Error messages like
SQL Server Agent fires an alert when the specified event contains a particular text string in the event message.

Performance alert
Specify properties for the alert that responds to a *server performance condition*.



Use the following parameters to specify the server performance conditions that will trigger the alert:

Object
Use the drop-down list to select the area of performance to be monitored.

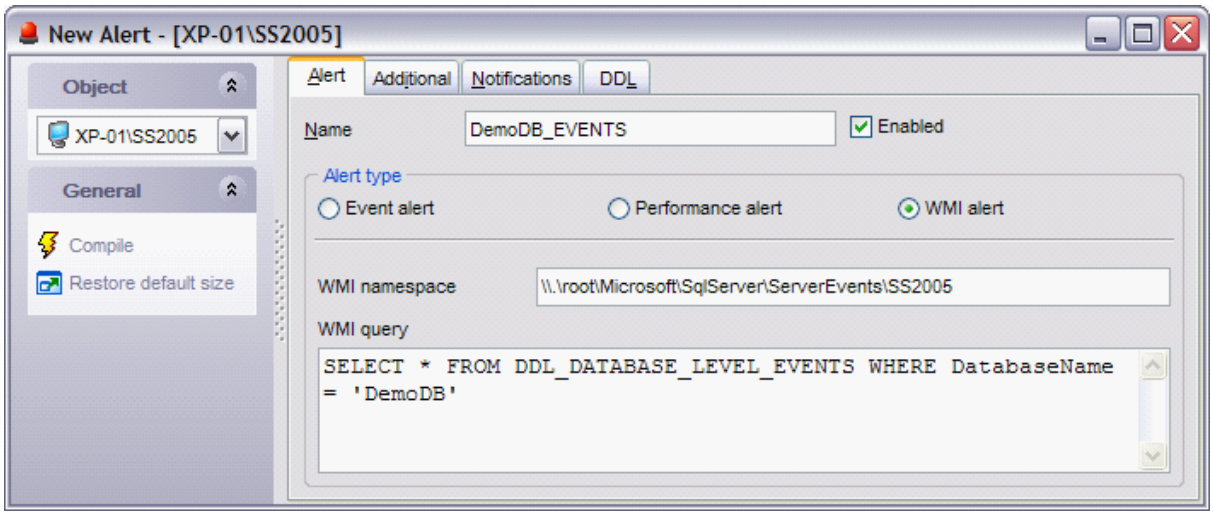
Counter
Use the drop-down list to select the attribute of the area to be monitored.

Instance
Use the drop-down list to define the specific instance (if any) of the attribute to be monitored.

Alert when counter becomes / Value
Using these boxes you should specify the threshold for the alert and the behavior that produces the alert. Use the drop-down list to select the behavior value: < (falls below), = (becomes equal to), > (rises above). Use the **Value** edit-box to specify the threshold number that describes the performance condition counter.

WMI alert
Specify properties for the alert that responds to a *WMI event*.

© 2008 EMS Database Management Solutions



WMI is an acronym for *Windows Management Instrumentation*. WMI is the Microsoft's implementation of Web-Based Enterprise Management (WBEM) - a new management technology that allows software to monitor and control managed resources throughout the network. Such managed resources include hard drives, file systems, settings of operating system, processes, services, shares, registry settings, networking components, event logs, users, groups, etc.
WMI allows monitoring of performance counters as well. Microsoft® SQL Server™ has the WMI Event Provider built-in.

One common way to use the WMI Event Provider is to create SQL Server Agent alerts that respond to specific events. SQL Server Agent submits a WQL request, receives WMI events, and runs a job in response to the event.

WMI namespace
Specify the WMI namespace on the machine where SQL Server is running.

WMI Query
Use this area to specify the WMI query text for the alert.

The WMI Query Language (WQL) is a subset of standard American National Standards Institute Structured Query Language (ANSI SQL) with minor semantic changes to support WMI.

Examples:

*SELECT * FROM Win32_OperatingSystem WHERE Caption="Microsoft Windows XP Professional" AND CSDVersion="Service Pack 2"*

If one needs to monitor the number of processes which use more than 10 threads, the following query can used:
SELECT ThreadCount FROM Win32_Process WHERE ThreadCount>10

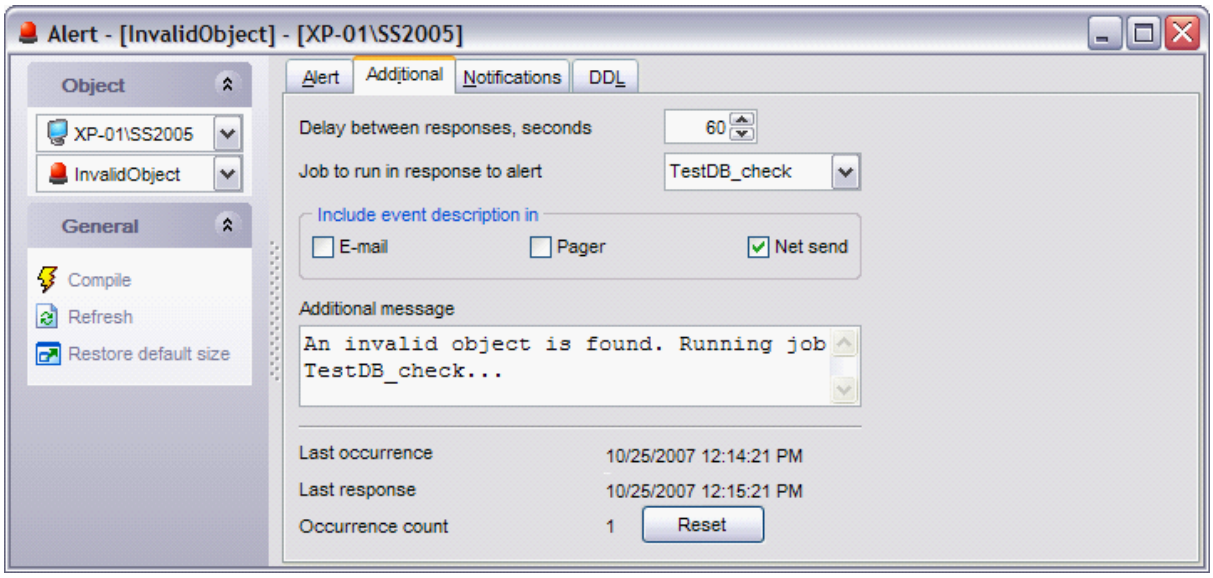
The following WQL query retrieves specific event properties for any event that occurs in the AdventureWorks database and exists under the DDL_DATABASE_LEVEL_EVENTS event group:

© 2008 EMS Database Management Solutions

SELECT SPID, SQLInstance, DatabaseName FROM DDL_DATABASE_LEVEL_EVENTS WHERE DatabaseName = 'AdventureWorks'

10.2.7.2.3 Setting additional properties

The **Additional** tab provides additional settings used for the specified alert.



Delay between responses, seconds
Specify the wait period, in seconds, between responses to the alert.

Job to run in response to alert
Use the drop-down list to select the name of the job to be executed in response to this alert.

Include event description in
Specify in which way the description of the server error should be included as part of the notification message:
☑ *E-mail*, ☑ *Pager*, ☑ *Net send*

Additional message
Use this box to specify an optional additional message sent to the operator as part of the e-mail, net send, or pager notification.

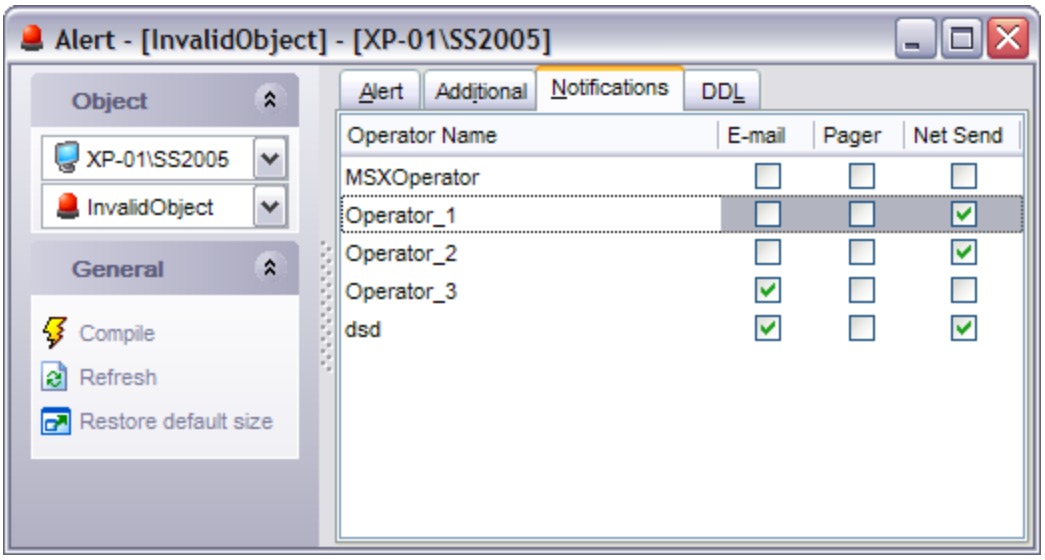
The lower area displays the following alert events:
Last occurrence
Last response
Occurrence count
If necessary, you can **reset** the occurrence history by clicking the corresponding button.

10.2.7.2.4 Managing notifications

The **Notifications** tab allows you to specify actions that the SQL Server Agent service will take after the alert fires. Alert notifications ensure that database administrators know

© 2008 EMS Database Management Solutions

when events/errors occur and which measures are taken in response.



The main area lists all currently available operators. Specify the method by which an operator is to be notified: *E-mail*, *Pager*, *Net Send*. Set the corresponding flag next to the operator name.

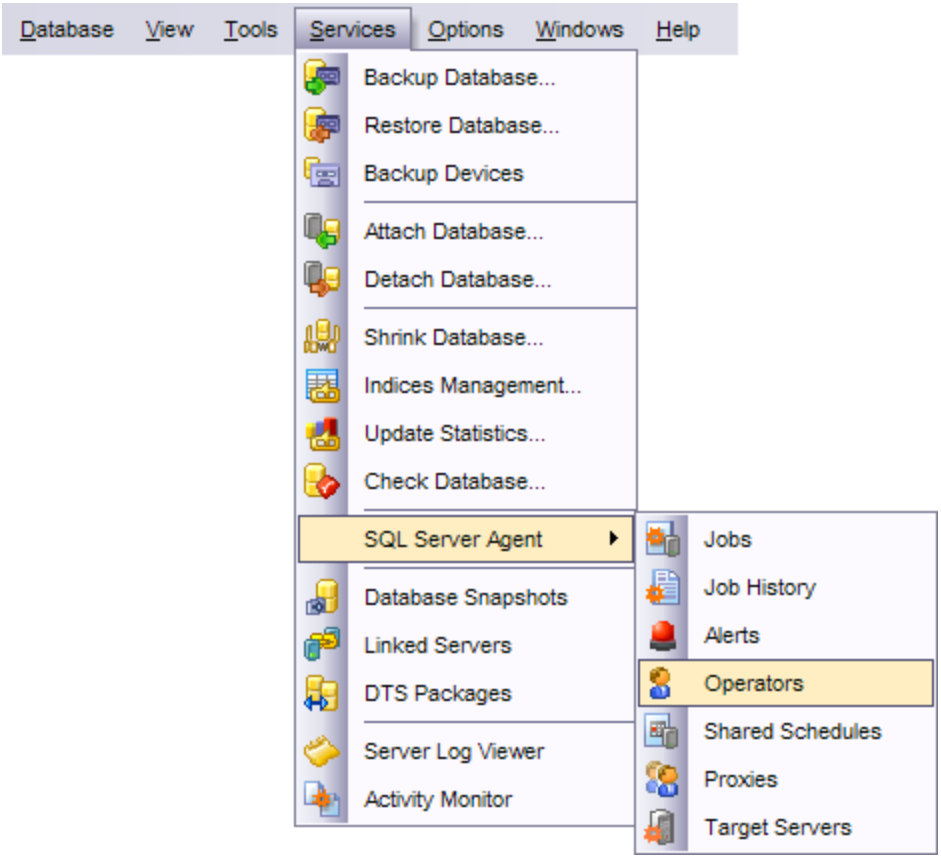
10.2.8 Operators

An **operator** is the object that represents a single SQL Server operator. SQL Server operators receive alert and job status notification in response to events generated by the server. You can assign alert notifications and job notifications for any operator defined for the instance of SQL Server.

Before using SQL Server operators, make sure that the **SQL Server Agent** service is running.

- Operators manager
- Operator Editor

© 2008 EMS Database Management Solutions



Creating Operators

To create a new operator:

- right-click the **Operators** node (within the **Server Objects** branch) or any object within this node in the DB Explorer tree and select the **New Operator** item from the context menu;
- define operator properties using the appropriate tabs of Operator Editor.

or

- select the **Database | New Object...** main menu item;
- select **Operator** in the Create New Object dialog;
- define operator properties using Operator Editor.

Hint: To create a new operator, you can also select the **Services | SQL Server Agent | Operators** main menu item to open the **Operators manager** and select the **Add Operator...** item from the context menu or on the Navigation bar.

To create a new operator with the same properties as one of existing operators has:

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

Alternatively, you can right-click an operator in the DB Explorer tree and select the **Duplicate Operator <operator_name>...** context menu item.

Duplicate Object Wizard allows you to select the server to create a new operator and to edit the result SQL statement for creating the operator.

© 2008 EMS Database Management Solutions

Editing Operators

To edit an existing operator:
- select the operator for editing in the DB Explorer tree (type the first letters of the operator name for quick search);
- right-click the object and select the **Edit Operator <operator_name>** context menu item, or simply double-click the operator;
- edit operator properties using the appropriate tabs of Operator Editor.

To change the name of an operator:
- select the operator to rename in the DB Explorer tree;
- right-click the operator alias and select the **Rename Operator <operator_name>...** item from the context menu;
- edit the operator name using the **Rename Object...** dialog.

Dropping Operators

To drop an operator:
- select the operator to drop in the DB Explorer tree;
- right-click the object and select the **Drop Operator <operator_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new operator;
Ctrl+O to edit the selected operator;
Ctrl+R to rename the operator;
Shift+Del to drop the object.

To create, edit, rename and drop operators, you can also use the **context menu** and the **Navigation bar** of Operators manager.

See also:
Backup Devices
Database Snapshots
Linked Servers
DTS Packages
Jobs
Job History
Alerts
Schedules
Proxies
Credentials
Endpoints
DDL Triggers
Target Servers

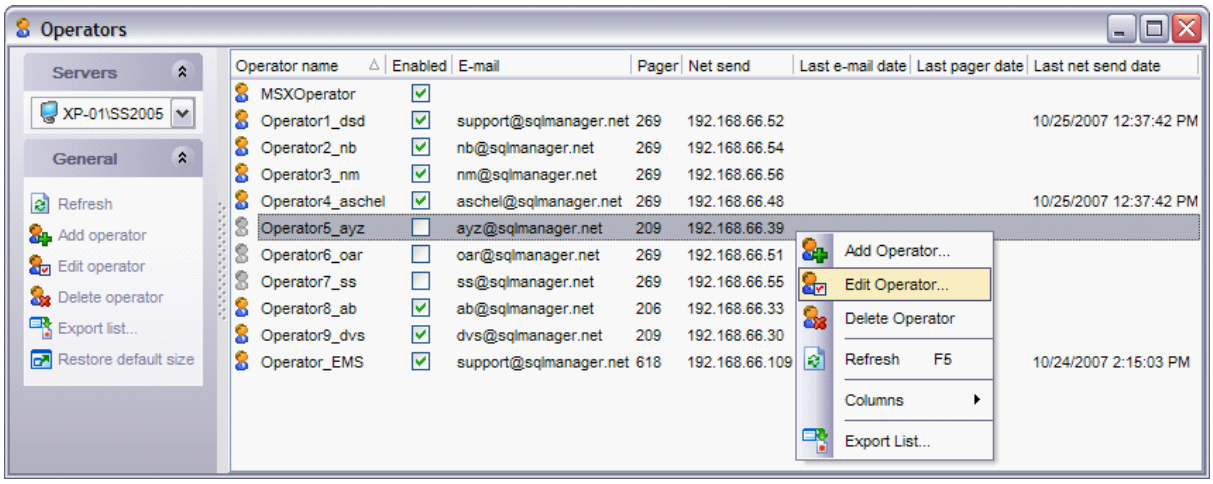
© 2008 EMS Database Management Solutions

Server Log Viewer
Activity Monitor

10.2.8.1 Operators manager

The **Operators manager** allows you to browse the list of **operators** and manage them efficiently.

To launch the tool, select the **Services | SQL Server Agent | Operators** main menu item, or right-click the host alias in the DB Explorer tree and select the **Tasks | Operators** item from the context menu.



The list displays the operators as a grid with the following columns: *Operator name*, *Enabled*, *E-mail*, *Pager*, *Net send*, *Last e-mail date*, *Last pager date*, *Last net send date*.

Hint: Enabled/disabled operators are differentiated in the list by their icons: icons for disabled operators are grayed out.

Right-click an item within the list to call the **context menu** allowing you to *create* a new operator and specify its properties using Operator Editor, *edit*, *delete* the selected operator, *refresh* the list, or show/hide columns of the list. Using the context menu you can also export the list of operators to any of supported output file formats.

Operators management tasks are also available through the **Navigation bar** of the **Operators manager**.

Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**No**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

© 2008 EMS Database Management Solutions

10.2.8.2 Operator Editor

Operator Editor allows you to define operator properties. It opens automatically when you create a new operator and is available on editing an existing one (see Create operator and Edit operator for details).

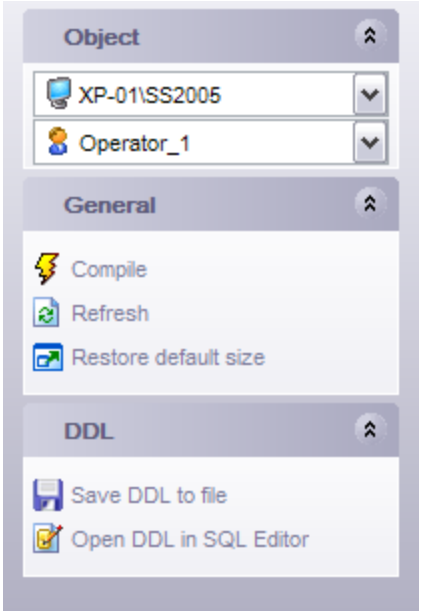
To open an operator in **Operator Editor**, double-click it in the DB Explorer tree or in the Operators manager. Alternatively, you can right-click the operator alias and select the **Edit Operator** context menu item.

- Using Navigation bar
- Editing operator definition
- Setting additional properties
- Managing alert notifications
- Managing job notifications
- Viewing DDL definition

10.2.8.2.1 Using Navigation bar

The **Navigation bar** of **Operator Editor** allows you to:

- select the server where the operator is located;
- select the operator for editing;
- compile the object (if it is being created/modified);
- refresh the content of the active tab;
- restore the default size of the editor window;
- save the operator DDL to an external **.sql* file;
- open the operator DDL in SQL Editor.



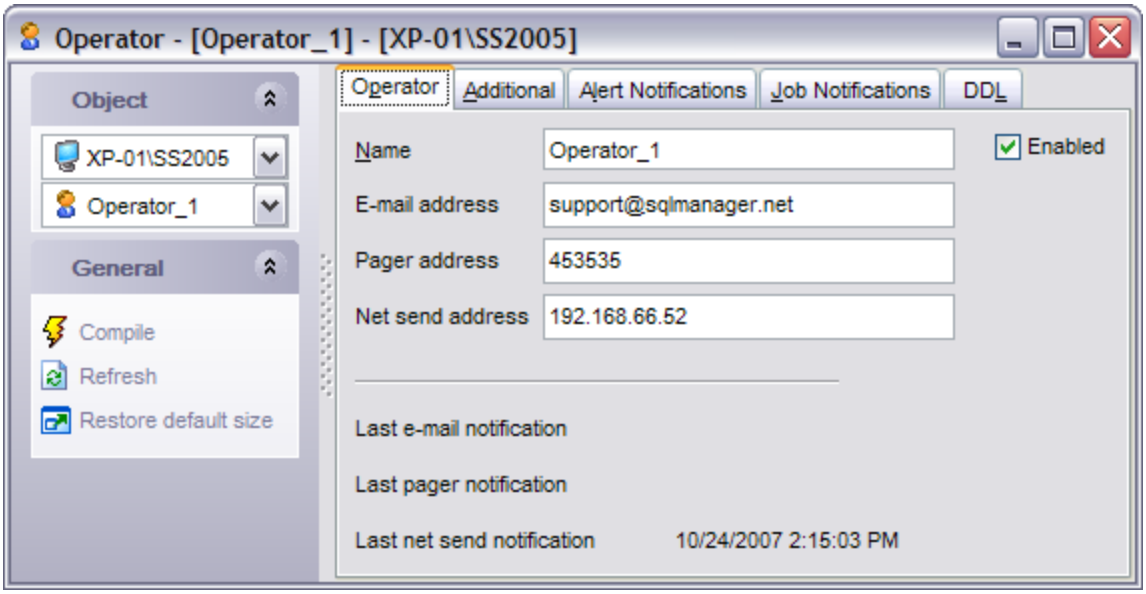
© 2008 EMS Database Management Solutions

<< Back to Operator Editor

10.2.8.2.2 Editing operator definition

Use the **Operator** tab of **Operator Editor** to configure/view operator (notification recipient) properties.

Name
Enter a name for the new operator, or modify the name of the operator being edited. Enable/disable the operator using the **Enabled** option.



E-mail address
Specify the e-mail address of the operator.

Pager address
Specify the pager address of the operator.

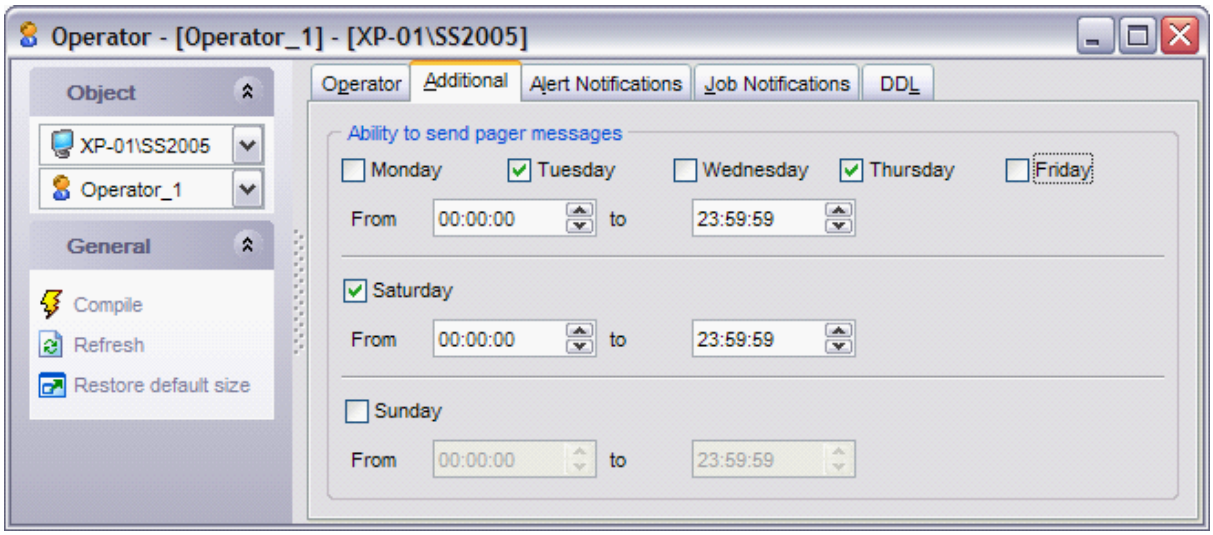
Net send address
Specify the network address of the operator to whom the network message will be sent.

The lower area displays the latest operator notifications:
Last e-mail notification
Last pager notification
Last net send notification

10.2.8.2.3 Setting additional properties

The **Additional** tab provides additional settings used for the specified operator.

© 2008 EMS Database Management Solutions

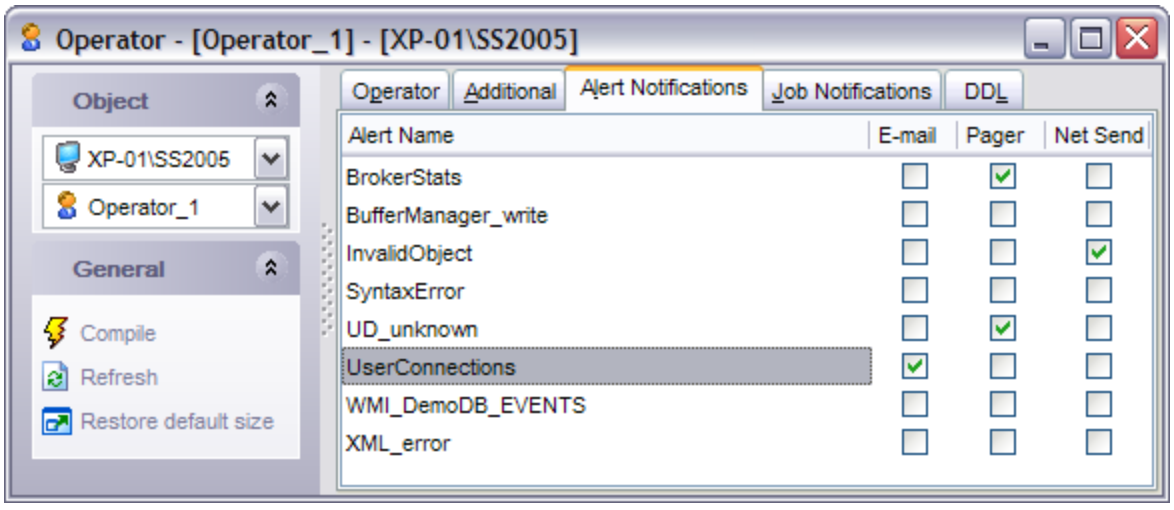


Ability to send pager messages
Specify when SQL Server Agent service can send pager notification to the operator: ☑ *Monday*, ☑ *Tuesday*, ☑ *Wednesday*, ☑ *Thursday*, ☑ *Friday*, ☑ *Saturday*, ☑ *Sunday*

From ... To ...
Set the day time interval values using the spin-edit boxes.

10.2.8.2.4 Managing alert notifications

The **Alert Notifications** tab allows you to specify methods that the SQL Server Agent service uses to notify the operator when each of existing alerts fires.

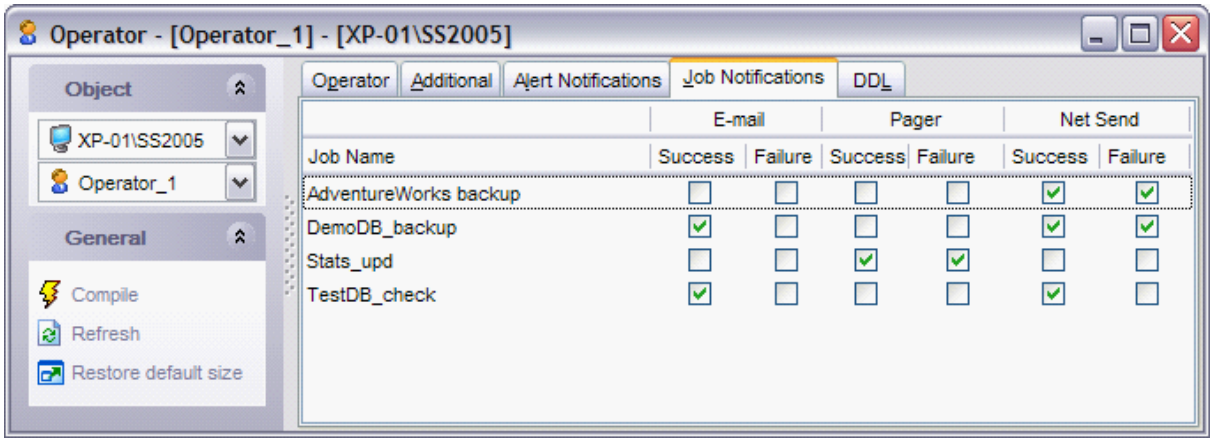


The main area lists all currently available alerts. Specify the method by which the operator is to be notified: *E-mail*, *Pager*, *Net Send*. Set the corresponding flag next to the alert name.

© 2008 EMS Database Management Solutions

10.2.8.2.5 Managing job notifications

The **Job Notifications** tab allows you to specify methods that the SQL Server Agent service uses to notify the operator when each of existing jobs is tried (succeeded or failed).



The main area lists all currently available jobs. Specify the method by which the operator is to be notified: *E-mail*, *Pager*, *Net Send*. Set the corresponding flag next to the alert name.

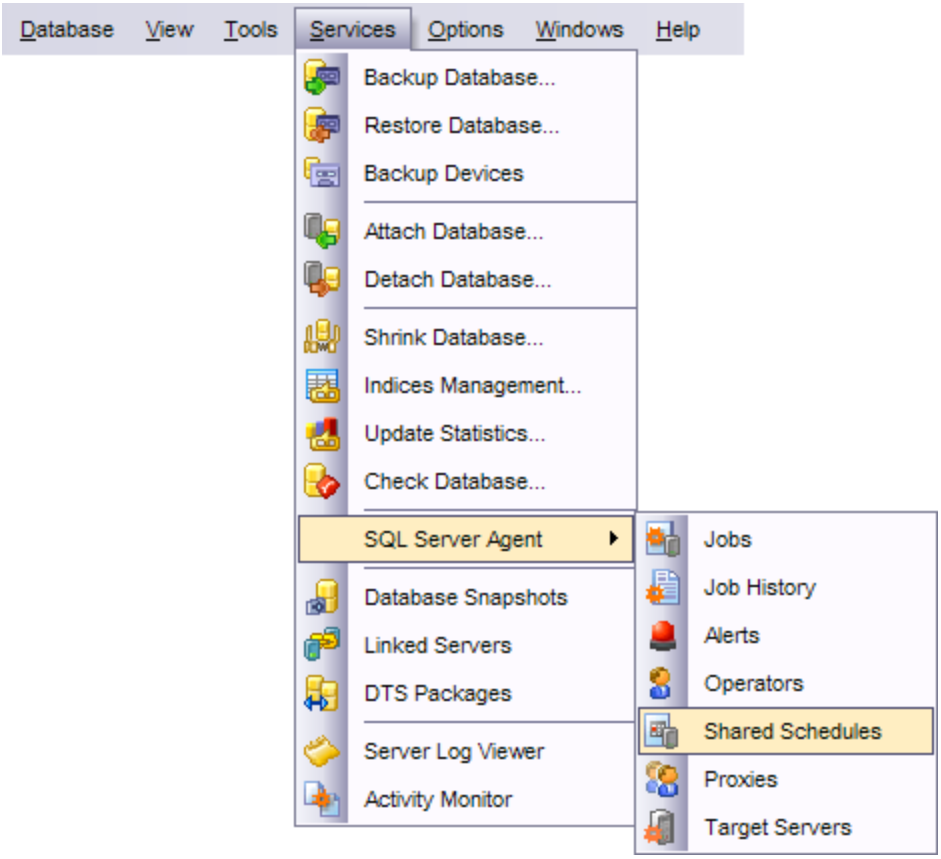
10.2.9 Schedules

Shared schedules are multipurpose items that contain ready-to-use schedule information. You can create a shared schedule once, and then reference it in a subscription or property page when you need to specify schedule information.

Before using SQL Server schedules, make sure that the **SQL Server Agent** service is running.

- Shared schedules manager
- Schedule Editor

© 2008 EMS Database Management Solutions



Creating Schedules

To create a new schedule:

- right-click the **Schedules** node (within the **Server Objects** branch) or any object within this node in the DB Explorer tree and select the **New Schedule** item from the context menu;
- define schedule properties using the appropriate tabs of Schedule Editor.

or

- select the **Database | New Object...** main menu item;
- select **Schedule** in the Create New Object dialog;
- define schedule properties using Schedule Editor.

Hint: To create a new schedule, you can also select the **Services | SQL Server Agent | Shared Schedules** main menu item to open the **Schedules manager** and select the **Add Schedule** item from the context menu or on the Navigation bar.

To create a new schedule with the same properties as one of existing schedules has:

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

Alternatively, you can right-click a schedule in the DB Explorer tree and select the **Duplicate Schedule <schedule_name>...** context menu item.

Duplicate Object Wizard allows you to select the server to create a new schedule and to edit the result SQL statement for creating the schedule.

© 2008 EMS Database Management Solutions

Editing Schedules

To edit an existing schedule:
- select the schedule for editing in the DB Explorer tree (type the first letters of the schedule name for quick search);
- right-click the object and select the **Edit Schedule <schedule_name>** context menu item, or simply double-click the schedule;
- edit schedule properties using the appropriate tabs of Schedule Editor.

To change the name of a schedule:
- select the schedule to rename in the DB Explorer tree;
- right-click the schedule alias and select the **Rename Schedule <schedule_name>...** item from the context menu;
- edit the schedule name using the **Rename Object...** dialog.

Dropping Schedules

To drop a schedule:
- select the schedule to drop in the DB Explorer tree;
- right-click the object and select the **Drop Schedule <schedule_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new schedule;
Ctrl+O to edit the selected schedule;
Ctrl+R to rename the schedule;
Shift+Del to drop the object.

To create, edit and drop schedules, you can also use the **context menu** and the **Navigation bar** of Shared schedules manager.

See also:
Backup Devices
Database Snapshots
Linked Servers
DTS Packages
Jobs
Job History
Alerts
Operators
Proxies
Credentials
Endpoints
DDL Triggers
Target Servers

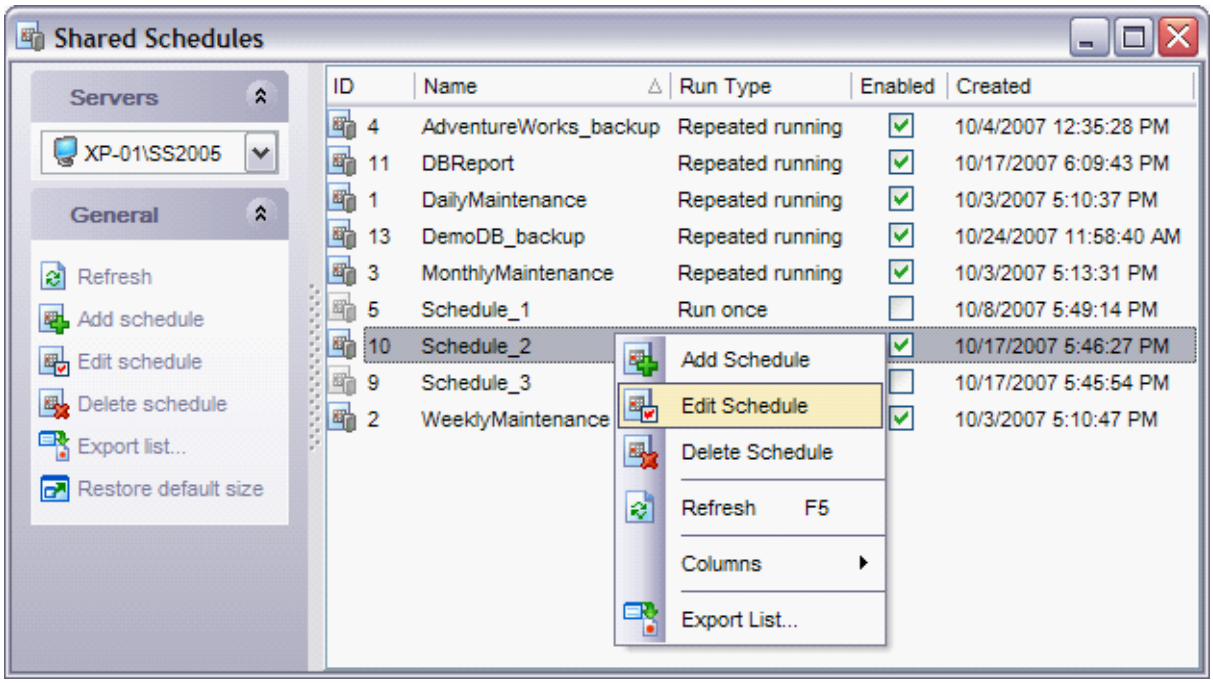
© 2008 EMS Database Management Solutions

Server Log Viewer
Activity Monitor

10.2.9.1 Shared schedules manager

The **Schedules manager** allows you to browse the list of **shared schedules** and manage them efficiently.

To launch the tool, select the **Services | SQL Server Agent | Shared Schedules** main menu item, or right-click the host alias in the DB Explorer tree and select the **Tasks | Shared Schedules** item from the context menu.



The list displays the shared schedules as a grid with the following columns: *ID*, *Name*, *Run Type*, *Enabled*, *Created*.

Hint: Enabled/disabled schedules are differentiated in the list by their icons: icons for disabled schedules are grayed out.

Right-click an item within the list to call the **context menu** allowing you to *create* a new schedule and specify its properties using Schedule Editor, *edit*, *delete* the selected schedule, *refresh* the list, or show/hide columns of the list. Using the context menu you can also export the list of schedules to any of supported output file formats.

Schedules management tasks are also available through the **Navigation bar** of the **Schedules manager**.

Availability:

© 2008 EMS Database Management Solutions

Full version (for Windows)	**Yes**
Lite version (for Windows)	**No**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

10.2.9.2 Schedule Editor

Schedule Editor allows you to define schedule properties. It opens automatically when you create a new schedule and is available on editing an existing one (see Create schedule and Edit schedule for details).

To open a schedule in **Schedule Editor**, double-click it in the DB Explorer tree or in the Shared schedules manager. Alternatively, you can right-click the schedule alias and select the **Edit Schedule** context menu item.

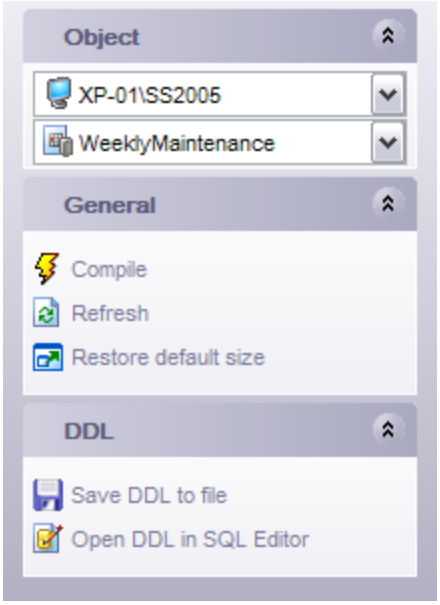
- Using Navigation bar
- Editing schedule definition
- Viewing scheduled jobs
- Viewing DDL definition

10.2.9.2.1 Using Navigation bar

The **Navigation bar** of **Schedule Editor** allows you to:

- select the server where the schedule is located;
- select the schedule for editing;
- compile the object (if it is being created/modified);
- refresh the content of the active tab;
- restore the default size of the editor window;
- save the schedule DDL to an external **.sql* file;
- open the schedule DDL in SQL Editor.

© 2008 EMS Database Management Solutions

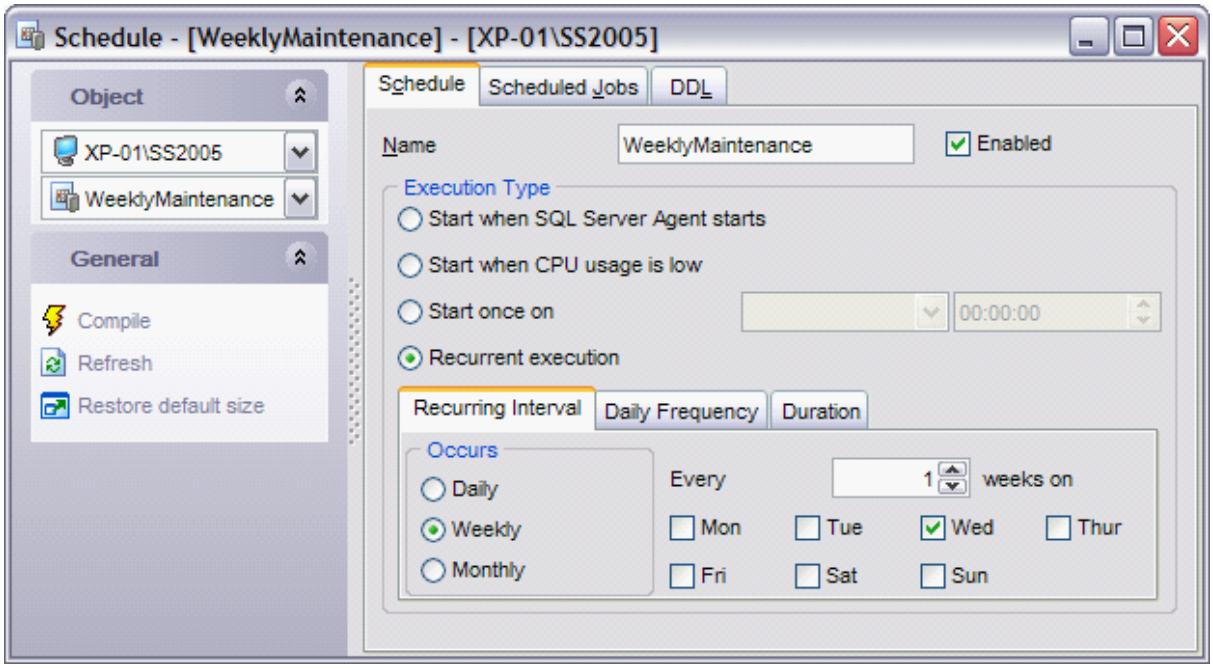


<< Back to Schedule Editor

10.2.9.2.2 Editing schedule definition

Use the **Schedule** tab of **Schedule Editor** to configure/view schedule properties.

Name
Enter a name for the new schedule, or modify the name of the schedule being edited. Enable/disable the schedule using the **Enabled** option.



© 2008 EMS Database Management Solutions

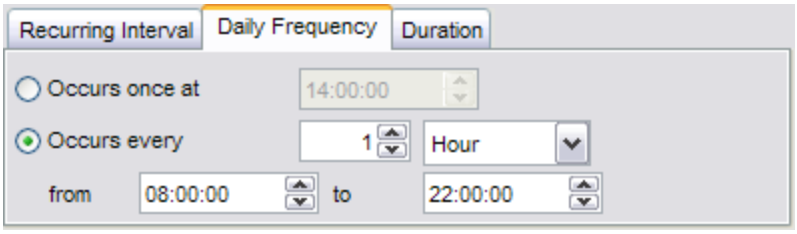
Specify the **Execution Type** of the schedule indicating when the scheduled job is to be executed:

- *Start when SQL Server Agent starts*
- *Start when CPU usage is low*
- *Start once on...* (set the date using the **date editor** which is activated when you click the Arrow-Down element of the combo-box; specify the time value using the spin-edit box)
- *Recurrent execution*

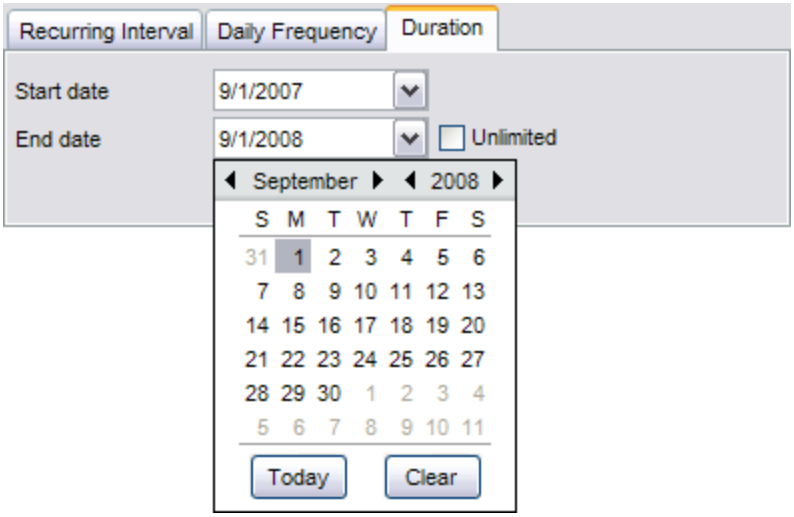
If the **Recurrent execution** type is selected, you should also set the **Recurring Interval**, **Daily Frequency** and **Duration** properties of the current schedule.

The **Recurring Interval** tab allows you to specify the **daily**/**weekly**/**monthly** recurrence rule, define the **Every ... days**/**weeks** value, and select the **days of the week**/**month** that can be set for the schedule.

The **Daily Frequency** tab allows you to select whether the scheduled job will be run **once at ...** or **every ... hours**/**minutes from ... to ...** (specify the day/time values using the drop-down menu and spin-edit boxes).



The **Duration** tab allows you to set the **Start date** and the **End date** threshold values for the interval using the **date editor** which is activated when you click the Arrow-Down element of the combo-box.

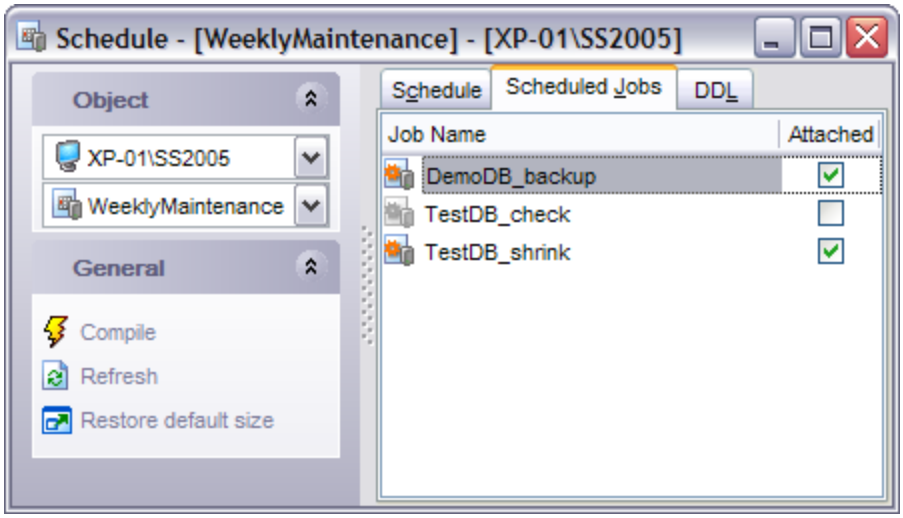


© 2008 EMS Database Management Solutions

Unlimited
Check this option if you are uncertain when the scheduled job should be finished, or if no limitations on the job duration are required.

10.2.9.2.3 Viewing scheduled jobs

The **Scheduled Jobs** tab allows you to browse the list of jobs that use the schedule, and to attach/detach them to/from the schedule.



Specify the **Attached** attribute for the jobs that should be attached to the schedule. Set the corresponding flag next to the job name.

10.2.10 Proxies

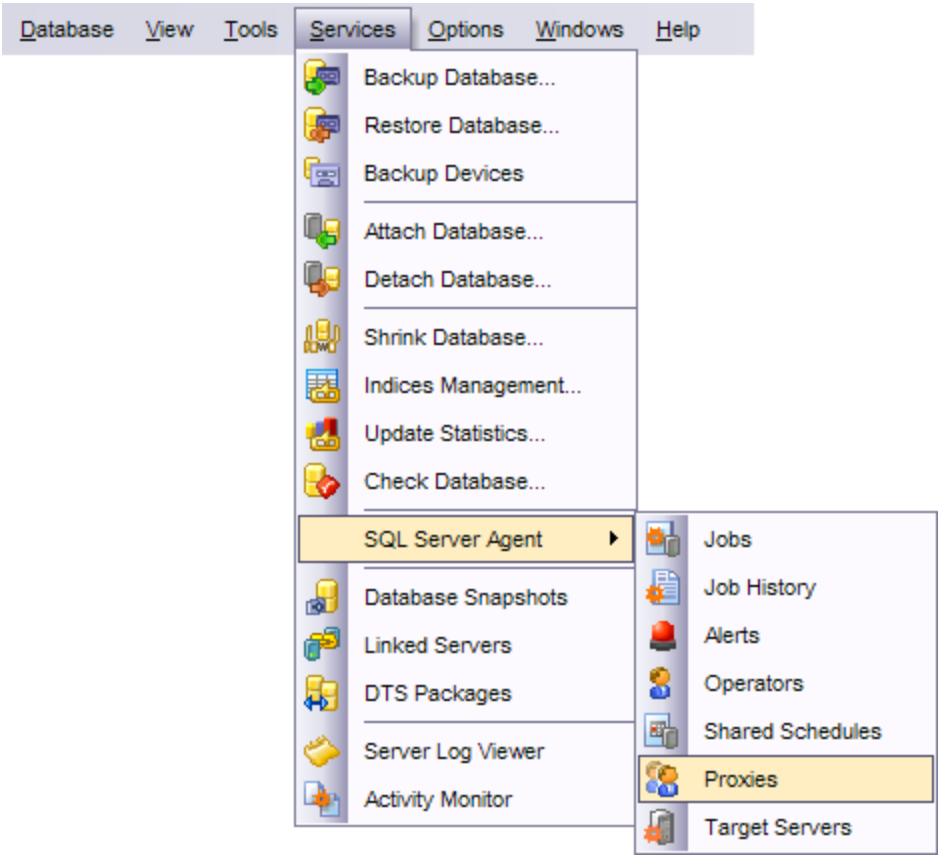
A Microsoft® SQL Server Agent **proxy** defines the security context for a job step. A proxy provides SQL Server Agent with access to the security credentials for a Microsoft Windows user. Each proxy can be associated with one or more subsystems. A job step that uses the proxy can access the specified subsystems by using the security context of the Windows user. Before SQL Server Agent runs a job step that uses a proxy, SQL Server Agent impersonates the credentials defined in the proxy, and then runs the job step by using that security context.

SQL Server **proxies** can be used to help database administrators ensure that each job step runs with the minimum permissions required to perform its task.

Before using SQL Server proxies, make sure that the **SQL Server Agent** service is running.

- Proxies manager
- Proxy Editor

© 2008 EMS Database Management Solutions



Creating Proxies

To create a new proxy:
- right-click the **Proxies** node (within the **Server Objects** branch) or any object within this node in the DB Explorer tree and select the **New Proxy** item from the context menu;
- define proxy properties using the appropriate tabs of Proxy Editor.

or
- select the **Database | New Object...** main menu item;
- select **Proxy** in the Create New Object dialog;
- define proxy properties using Proxy Editor.

Hint: To create a new proxy, you can also select the **Services | SQL Server Agent | Proxies** main menu item to open the **Proxies manager** and select the **Add Proxy...** item from the context menu or on the Navigation bar.

To create a new proxy with the same properties as one of existing proxies has:
- select the **Database | Duplicate Object...** menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a proxy in the DB Explorer tree and select the **Duplicate Proxy <proxy_name>...** context menu item.

Duplicate Object Wizard allows you to select the server to create a new proxy and to edit the result SQL statement for creating the proxy.

© 2008 EMS Database Management Solutions

Editing Proxies

To edit an existing proxy:
- select the proxy for editing in the DB Explorer tree (type the first letters of the proxy name for quick search);
- right-click the object and select the **Edit Proxy <proxy_name>** context menu item, or simply double-click the proxy;
- edit proxy properties using the appropriate tabs of Proxy Editor.

To change the name of a proxy:
- select the proxy to rename in the DB Explorer tree;
- right-click the proxy alias and select the **Rename Proxy <proxy_name>...** item from the context menu;
- edit the proxy name using the **Rename Object...** dialog.

Dropping Proxies

To drop a proxy:
- select the proxy to drop in the DB Explorer tree;
- right-click the object and select the **Drop Proxy <proxy_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new proxy;
Ctrl+O to edit the selected proxy;
Ctrl+R to rename the proxy;
Shift+Del to drop the object.

To create, edit and drop proxies, you can also use the **context menu** and the **Navigation bar** of Proxies manager.

See also:
Backup Devices
Database Snapshots
Linked Servers
DTS Packages
Jobs
Job History
Alerts
Operators
Schedules
Credentials
Endpoints
DDL Triggers

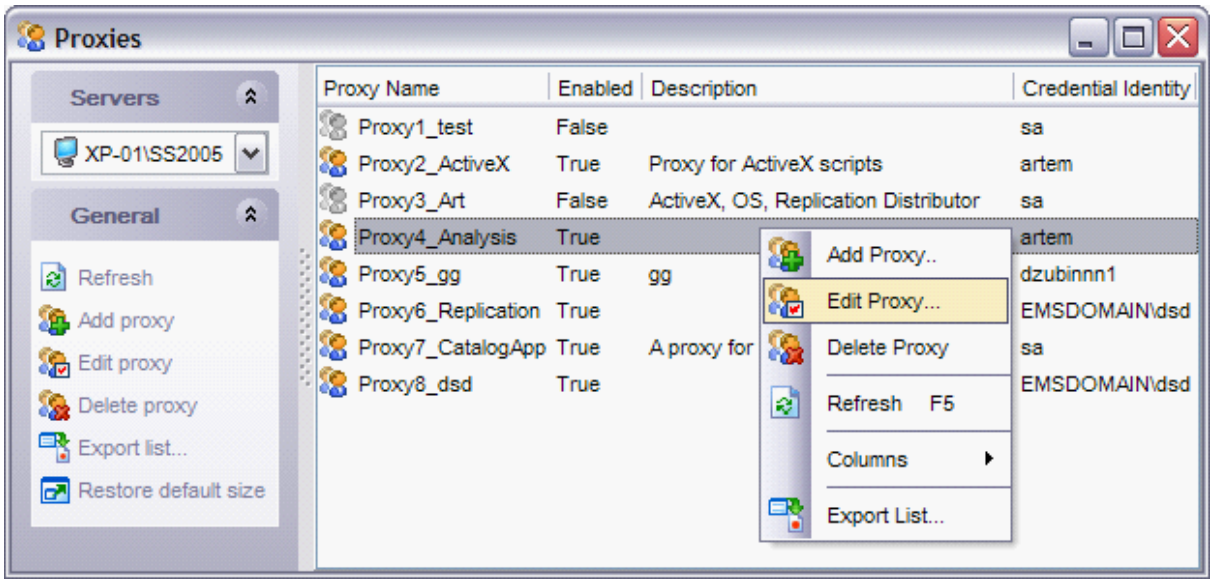
© 2008 EMS Database Management Solutions

Target Servers
Server Log Viewer
Activity Monitor

10.2.10.1 Proxies manager

The **Proxies manager** allows you to browse the list of **proxies** and manage them efficiently.

To launch the tool, select the **Services | SQL Server Agent | Proxies** main menu item, or right-click the host alias in the DB Explorer tree and select the **Tasks | Proxies** item from the context menu.



The list displays the proxies as a grid with the following columns: *Proxy Name*, *Enabled*, *Description*, *Credential Identity*.

Hint: Enabled/disabled proxies are differentiated in the list by their icons: icons for disabled proxies are grayed out.

Right-click an item within the list to call the **context menu** allowing you to *create* a new proxy and specify its properties using Proxy Editor, *edit*, *delete* the selected proxy, *refresh* the list, or show/hide columns of the list. Using the context menu you can also export the list of proxies to any of supported output file formats.

Proxies management tasks are also available through the **Navigation bar** of the **Proxies manager**.

Availability:
Full version (for Windows) **Yes**

© 2008 EMS Database Management Solutions

Lite version (for Windows)	**No**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

10.2.10.2 Proxy Editor

Proxy Editor allows you to define proxy properties. It opens automatically when you create a new proxy and is available on editing an existing one (see Create proxy and Edit proxy for details).

A SQL Server Agent proxy manages security for job steps that involve subsystems other than the Transact-SQL subsystem. Each proxy corresponds to a security credential. A proxy may have access to any number of subsystems.

To open a proxy in **Proxy Editor**, double-click it in the DB Explorer tree or in the Proxies manager. Alternatively, you can right-click the proxy alias and select the **Edit Proxy** context menu item.

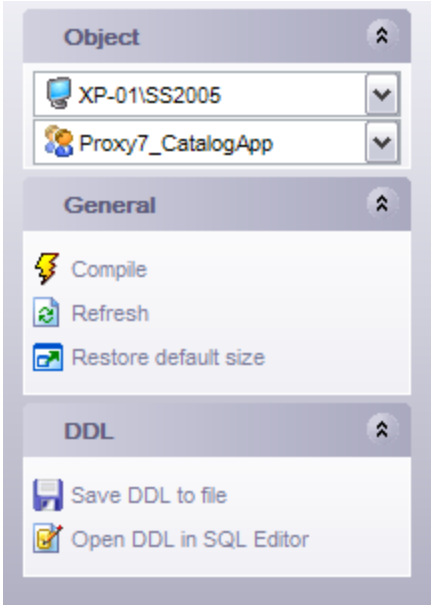
- Using Navigation bar
- Editing proxy definition
- Viewing proxy jobs
- Viewing DDL definition

10.2.10.2.1 Using Navigation bar

The **Navigation bar** of **Proxy Editor** allows you to:

- select the server where the proxy is located;
- select the proxy for editing;
- compile the object (if it is being created/modified);
- refresh the content of the active tab;
- restore the default size of the editor window;
- save the proxy DDL to an external **.sql* file;
- open the proxy DDL in SQL Editor.

© 2008 EMS Database Management Solutions



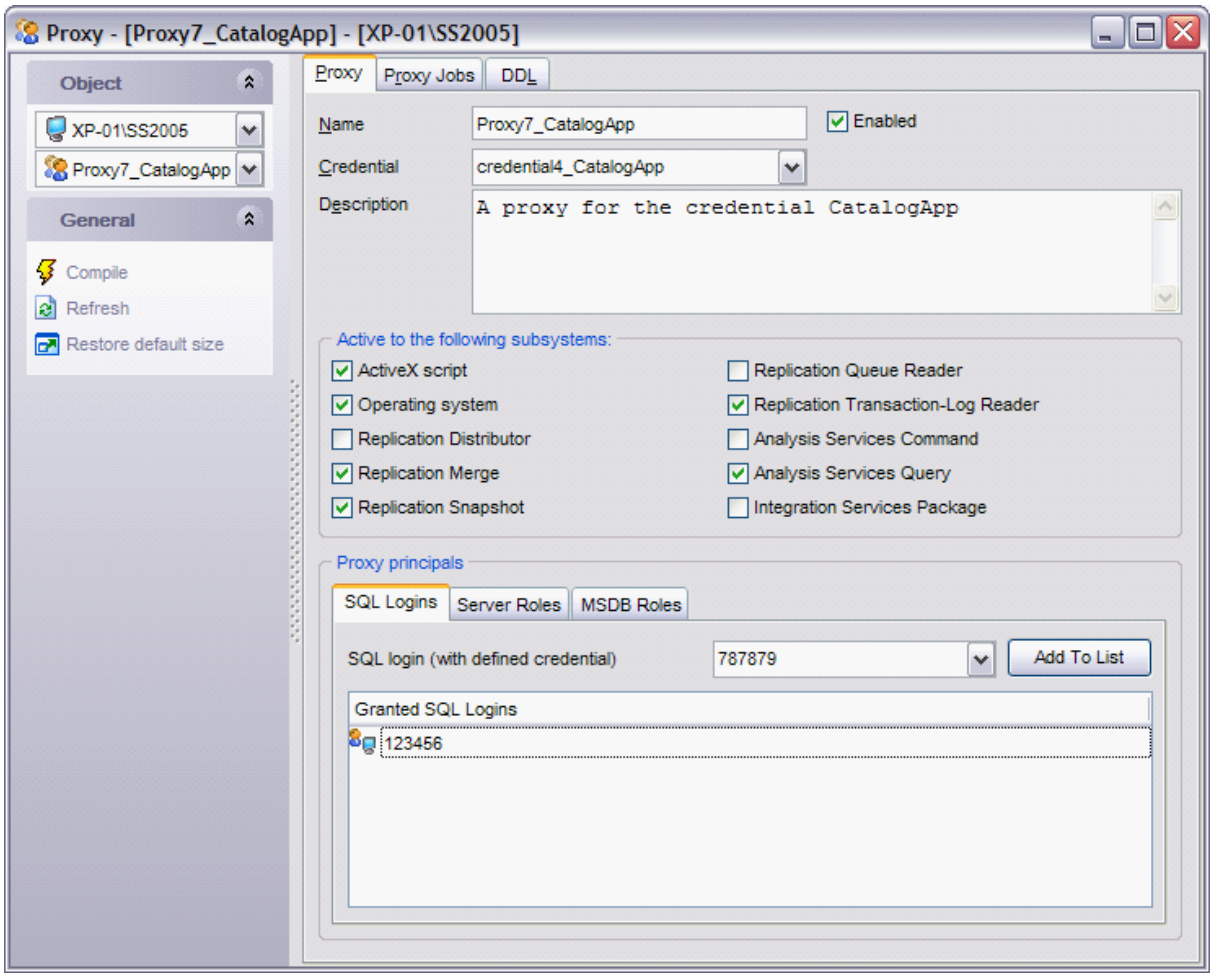
<< Back to Proxy Editor

10.2.10.2.2 Editing proxy definition

Use the **Proxy** tab of **Proxy Editor** to configure/view proxy properties. SQL Server Agent uses **proxies** to manage security contexts.

Name
Enter a name for the new proxy, or modify the name of the proxy being edited.
Enable/disable the proxy using the **Enabled** option.

© 2008 EMS Database Management Solutions



Credential
Each proxy corresponds to a security credential. Use the drop-down list to select a credential for the proxy.

Description
In this memo field you can supply an optional description for the proxy.

Each proxy can be associated with a set of subsystems and a set of logins.

Active to the following subsystems
Tick off the subsystem(s) to grant access to:
- ☑ *ActiveX script*
- ☑ *Operating system*
- ☑ *Replication Distributor*
- ☑ *Replication Merge*
- ☑ *Replication Snapshot*
- ☑ *Replication Queue Reader*
- ☑ *Replication Transaction-Log Reader*
- ☑ *Analysis Services Command*
- ☑ *Analysis Services Query*

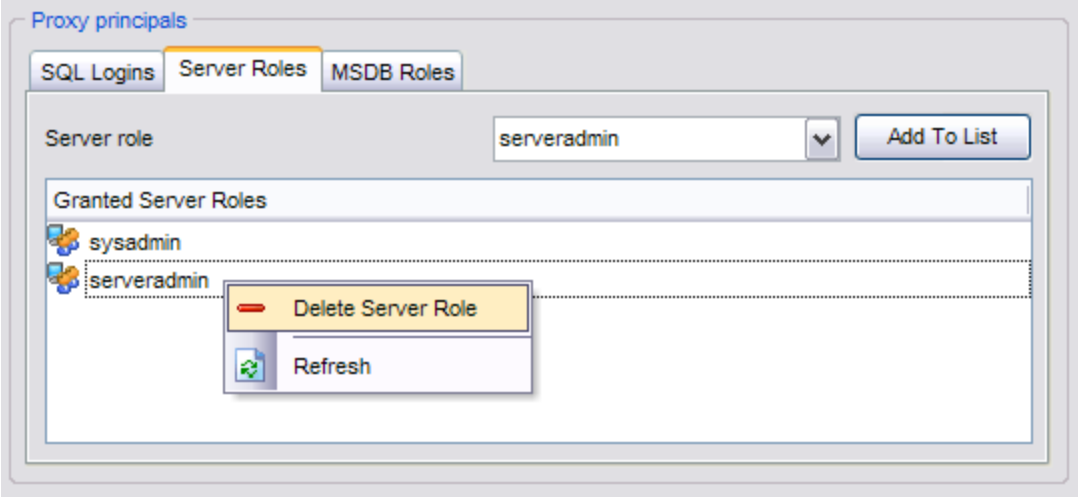
© 2008 EMS Database Management Solutions

☑ *Integration Services Package*

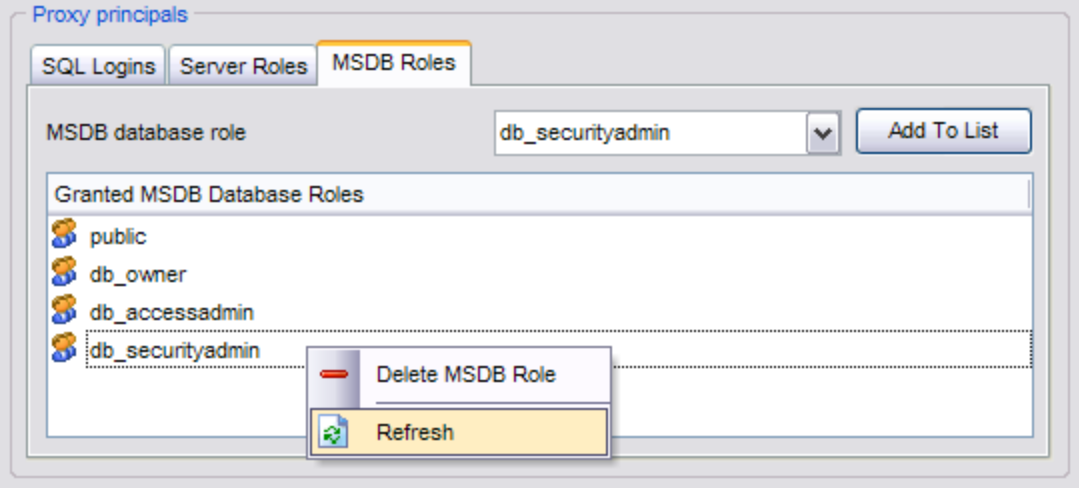
The **Proxy principals** group allows you to grant security principal access to the proxy.

Use the **SQL Logins** tab to grant access to SQL logins (with defined credentials).
To specify a login, select it in the drop-down list and click the **Add to List** button.

Use the **Server Roles** tab to grant access to fixed server roles.
To specify a server role, select it in the drop-down list and click the **Add to List** button.



Use the **MSDB Roles** tab to grant access to the database roles in the *msdb* database.
To specify a database role, select it in the drop-down list and click the **Add to List** button.



Right-click an item within the **Granted SQL Logins** / **Server Roles** / **MSDB Database Roles** list to call the **context menu** allowing you to *delete* a the selected principal and

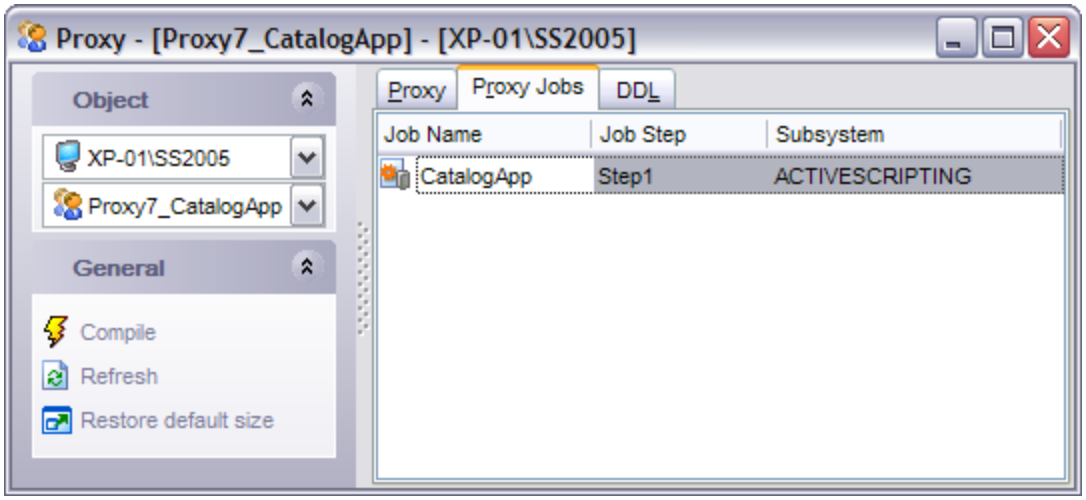
© 2008 EMS Database Management Solutions

to refresh the list.

10.2.10.2.3 Viewing proxy jobs

The **Proxy Jobs** tab allows you to browse the list of job steps that use the proxy.

Note: The proxy can be used only for job steps that use a subsystem associated with the proxy. To create a job step that uses a specific proxy, the job owner must either belong to a login associated with that proxy or be a member of a role with unrestricted access to proxies.



A proxy can be used in more than one job step.
The list displays the job steps as a grid with the following columns: *Job Name*, *Job Step*, *Subsystem*.

10.2.11 Credentials

A **credential** is a record containing the authentication information required to connect to a resource outside of SQL Server. Most credentials consist of a Windows login and password.
Credentials allow users that connect to SQL Server using SQL Authentication to connect to Windows or other resources outside of SQL Server.
After creating a credential, you can map it to a login. A single credential can be mapped to multiple SQL Server logins. But a SQL Server login can be mapped to only one credential.

- Credential Editor

Creating Credentials

To create a new credential:

- right-click the **Credentials** node (within the **Server Objects** branch) or any object within this node in the DB Explorer tree and select the **New Credential** item from the context menu;

© 2008 EMS Database Management Solutions

- define credential properties using the appropriate tabs of Credential Editor.

or

- select the **Database | New Object...** main menu item;
- select **Credential** in the Create New Object dialog;
- define credential properties using Credential Editor.

To create a new credential with the same properties as one of existing credentials has:

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

Alternatively, you can right-click a credential in the DB Explorer tree and select the **Duplicate Credential <credential_name>...** context menu item.

Duplicate Object Wizard allows you to select the server to create a new credential and to edit the result SQL statement for creating the credential.

Editing Credentials

To edit an existing credential:

- select the credential for editing in the DB Explorer tree (type the first letters of the credential name for quick search);
- right-click the object and select the **Edit Credential <credential_name>** context menu item, or simply double-click the credential;
- edit credential properties using the appropriate tabs of Credential Editor.

Dropping Credentials

To drop a credential:

- select the credential to drop in the DB Explorer tree;
- right-click the object and select the **Drop Credential <credential_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new credential;
Ctrl+O to edit the selected credential;
Shift+Del to drop the object.

See also:

Backup Devices
Database Snapshots
Linked Servers
DTS Packages
Jobs
Job History
Alerts
Operators
Schedules
Proxies

© 2008 EMS Database Management Solutions

Endpoints
DDL Triggers
Target Servers
Server Log Viewer
Activity Monitor

10.2.11.1 Credential Editor

Credential Editor allows you to define credential properties. It opens automatically when you create a new credential and is available on editing an existing one (see Create credential and Edit credential for details).

To open a credential in **Credential Editor**, double-click it in the DB Explorer tree. Alternatively, you can right-click the credential alias and select the **Edit Credential** context menu item.

- Using Navigation bar
- Editing credential definition
- Viewing DDL definition

Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**No**

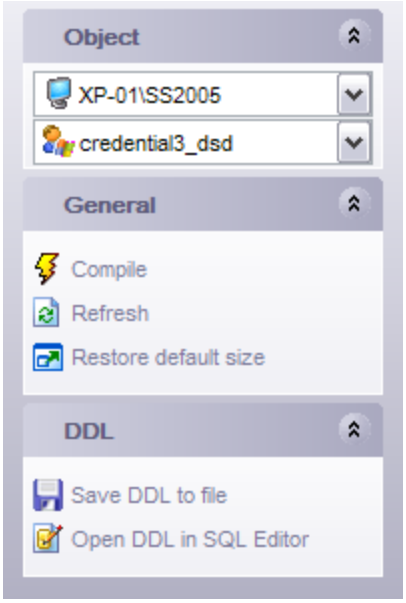
Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

10.2.11.1.1 Using Navigation bar

The **Navigation bar** of **Credential Editor** allows you to:

- select the server where the credential is located;
- select the credential for editing;
- compile the object (if it is being created/modified);
- refresh the content of the active tab;
- restore the default size of the editor window;
- save the credential DDL to an external **.sql* file;
- open the credential DDL in SQL Editor.

© 2008 EMS Database Management Solutions

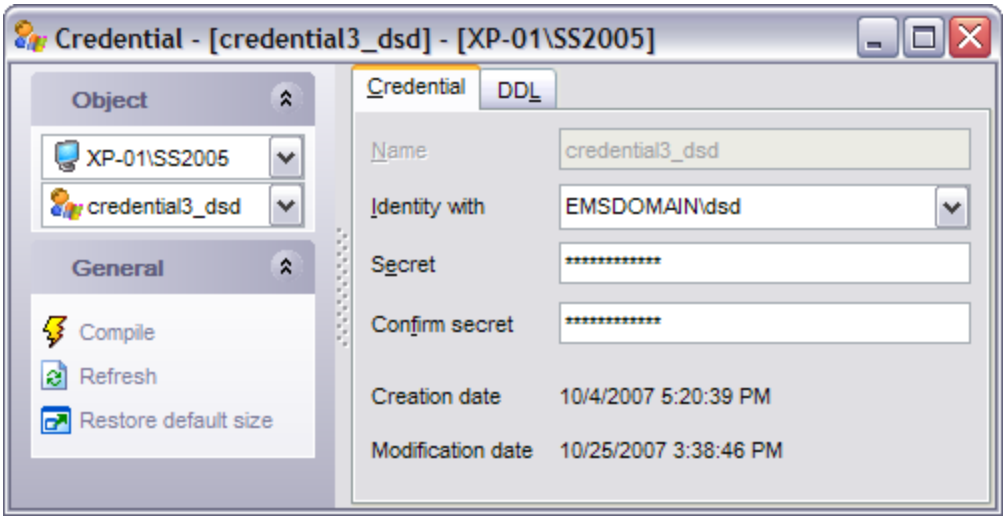


<< Back to Credential Editor

10.2.11.1.2 Editing credential definition

Use the **Credential** tab of **Credential Editor** to configure/view credential properties.

Name
Enter a name for the new credential, or view the name of the credential being edited.



Identity with
Use the drop-down list to select the name of the account to be used when connecting outside the server.

© 2008 EMS Database Management Solutions

Secret
Specify the secret required for outgoing authentication, and **confirm the secret** in the corresponding box.

The lower area displays the following credential events:
Creation date
Modification date

10.2.12 Endpoints

Communication of SQL Server Database Engine with an application is formatted in a Microsoft communication format called a tabular data stream (TDS) packet. The TDS packet is encapsulated inside a standard communication protocol (such as TCP/IP or Named Pipes) by the Net-Libraries on both the server and the client computers. On the server side the Net-Libraries are part of the Database Engine, whereas on the client side the Net-Libraries are part of the SQL Native Client. To configure Net-Libraries, the server and the SQL Native Client are configured to use a network protocol. The server creates a SQL Server object called a TDS **endpoint** for each network protocol.

- Endpoint Editor

Creating Endpoints

To create a new endpoint:

- right-click the **Endpoints** node (within the **Server Objects** branch) or any object within this node in the DB Explorer tree and select the **New Endpoint** item from the context menu;
- define endpoint properties using the appropriate tabs of Endpoint Editor.

or

- select the **Database | New Object...** main menu item;
- select **Endpoint** in the Create New Object dialog;
- define endpoint properties using Endpoint Editor.

To create a new endpoint with the same properties as one of existing endpoints has:

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

Alternatively, you can right-click an endpoint in the DB Explorer tree and select the **Duplicate Endpoint <endpoint_name>...** context menu item.

Duplicate Object Wizard allows you to select the server to create a new endpoint and to edit the result SQL statement for creating the endpoint.

Editing Endpoints

To edit an existing endpoint:

- select the endpoint for editing in the DB Explorer tree (type the first letters of the endpoint name for quick search);
- right-click the object and select the **Edit Endpoint <endpoint_name>** context menu item, or simply double-click the endpoint;
- edit endpoint properties using the appropriate tabs of Endpoint Editor.

© 2008 EMS Database Management Solutions

Dropping Endpoints

To drop an endpoint:
- select the endpoint to drop in the DB Explorer tree;
- right-click the object and select the **Drop Endpoint <endpoint_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new endpoint;
Ctrl+O to edit the selected endpoint;
Shift+Del to drop the object.

See also:
Backup Devices
Database Snapshots
Linked Servers
DTS Packages
Jobs
Job History
Alerts
Operators
Schedules
Proxies
Credentials
DDL Triggers
Target Servers
Server Log Viewer
Activity Monitor

10.2.12.1 Endpoint Editor

Endpoint Editor allows you to define endpoint properties. It opens automatically when you create a new endpoint and is available on editing an existing one (see Create endpoint and Edit endpoint for details).

To open an endpoint in **Endpoint Editor**, double-click it in the DB Explorer tree. Alternatively, you can right-click the endpoint alias and select the **Edit Endpoint** context menu item.
- Using Navigation bar
- Editing endpoint definition
- Setting protocol options
- Setting payload options
- Viewing DDL definition

© 2008 EMS Database Management Solutions

Availability:

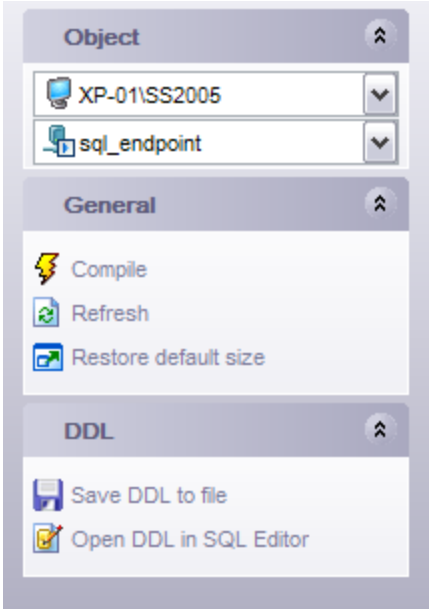
Full version (for Windows)	**Yes**
Lite version (for Windows)	**No**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

10.2.12.1.1 Using Navigation bar

The **Navigation bar** of **Endpoint Editor** allows you to:

- select the server where the endpoint is located;
- select the endpoint for editing;
- compile the object (if it is being created/modified);
- refresh the content of the active tab;
- restore the default size of the editor window;
- save the endpoint DDL to an external **.sql* file;
- open the endpoint DDL in SQL Editor.



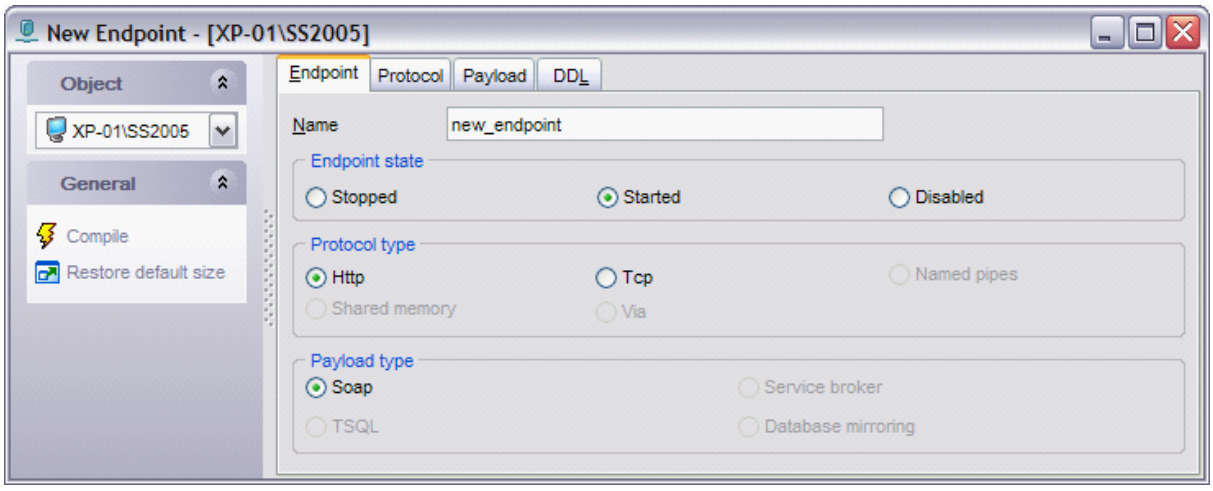
<< Back to Endpoint Editor

10.2.12.1.2 Editing endpoint definition

Use the **Endpoint** tab of **Endpoint Editor** to configure/view endpoint properties.

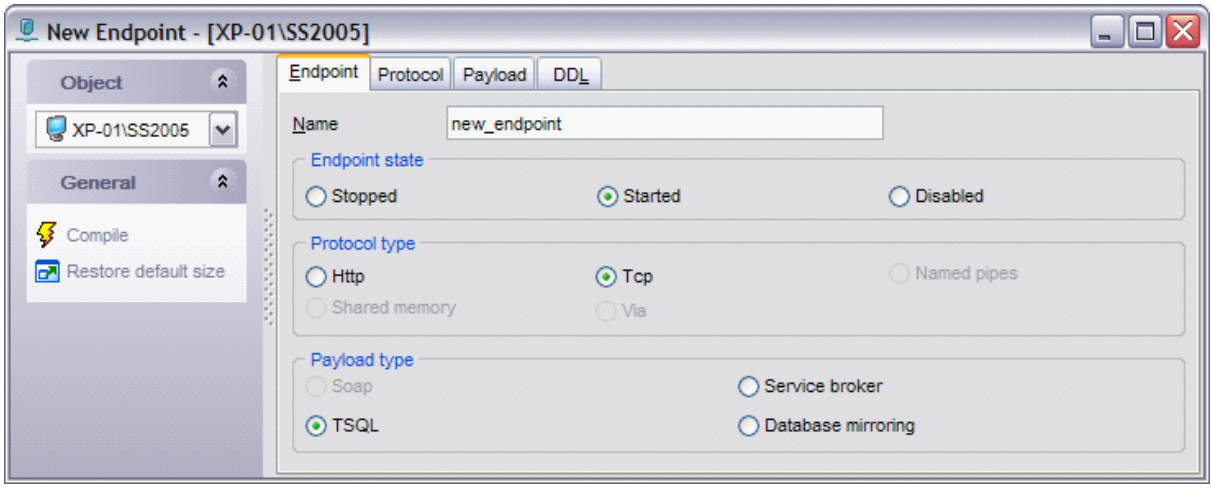
Name
Enter a name for the new endpoint, or view the name of the endpoint being edited.

© 2008 EMS Database Management Solutions



Endpoint state
Specify the state of the endpoint when it is created:
Stopped
This option indicates that the endpoint is stopped. In this state, the server listens to port requests but returns errors to clients.
Started
This option indicates that the endpoint is started and is actively listening for connections.
Disabled
This option indicates that the endpoint is disabled. In this state, the server does not listen to the endpoint port or respond to any attempted requests to use the endpoint.

Protocol type
Specify the transport protocol to use:
HTTP
TCP
Named Pipes
Shared memory
Via



© 2008 EMS Database Management Solutions

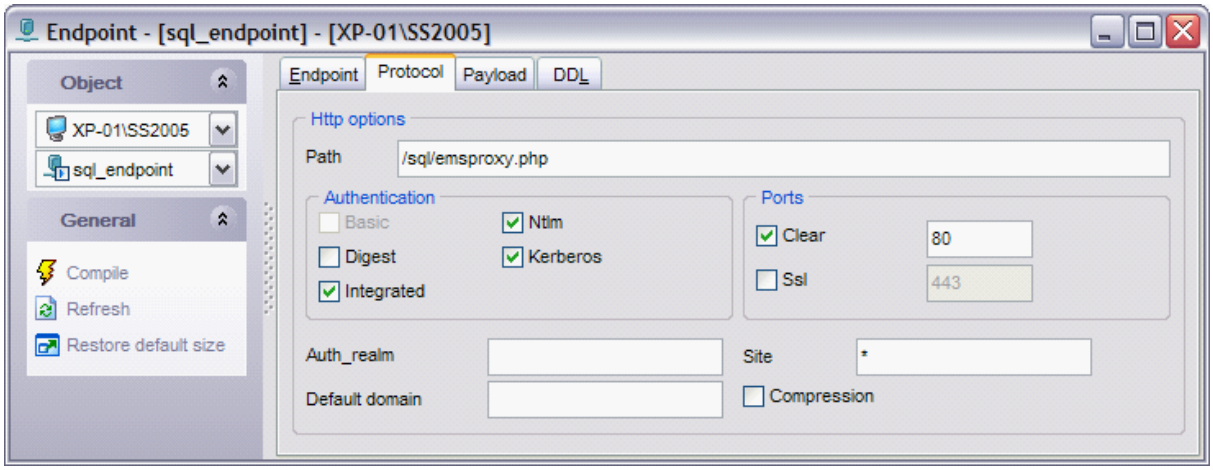
Payload type
Specify the payload type:
- *Soap* (available for HTTP protocol only)
- *TSQL* (available for TCP protocol only)
- *Service broker* (available for TCP protocol only)
- *Database mirroring* (available for TCP protocol only)

10.2.12.1.3 Setting protocol options

The **Protocol** tab allows you to specify HTTP/TCP protocol options pertaining to the endpoint.

HTTP options

Path
Specify the URL path that identifies the location of the endpoint on the host computer. This string is a logical partitioning of the URL namespace that is used by the listener to route requests appropriately.



Authentication
Specify the authentication type(s) to be used to authenticate users that log on to the instance of SQL Server:

☑ *Basic*
It is one of the two required authentication mechanisms in the HTTP 1.1 specification. Basic authentication is made up of an Authentication header that contains the base64-encoded user name and password separated by a colon.

☑ *Digest*
It is the second authentication mechanism required by HTTP 1.1. This authentication is made up of the user name and password. This is then hashed with MD5, a one-way hashing algorithm, and sent to the server. The server has access to either the raw password, or a stored MD5 hash that was created when the password was set. The

© 2008 EMS Database Management Solutions

server can then compare the stored calculated value to the one provided by the client. This way, the client can prove that it knows the password without actually giving it to the server.

☑ *Integrated*
Endpoints configured to support integrated authentication can respond with either of the following authentication types as part of the authentication challenge: *Kerberos* or *NTLM*. Under this configuration, the server will try to authenticate the client with whichever type the client uses in requesting authentication.

☑ *NTLM*
It is the authentication mechanism supported by Windows 9x and Windows NT 4.0 (client and server). This authentication mechanism is a challenge-response protocol that offers stronger authentication than either basic or digest. NTLM is implemented in Windows 2000 and later versions by a Security Support Provider Interface (SSPI).

☑ *Kerberos*
This authentication is an Internet standard authentication mechanism. Kerberos authentication is supported in Windows 2000 and later versions by an SSPI. When Kerberos authentication is used, the instance of SQL Server must associate a Service Principal Name (SPN) with the account it will be running on.

Ports
Use this group to specify the listening port type(s) to be associated with the endpoint:
☑ *Clear*
☑ *SSL*

Auth_realm
Specify the hint that returns to the client, which sent the SOAP request to the endpoint, as part of HTTP authentication challenge (applied when **Authentication** is set to *Digest*).

Default domain
Specify the default login domain (applied when **Authentication** is set to *Basic*).

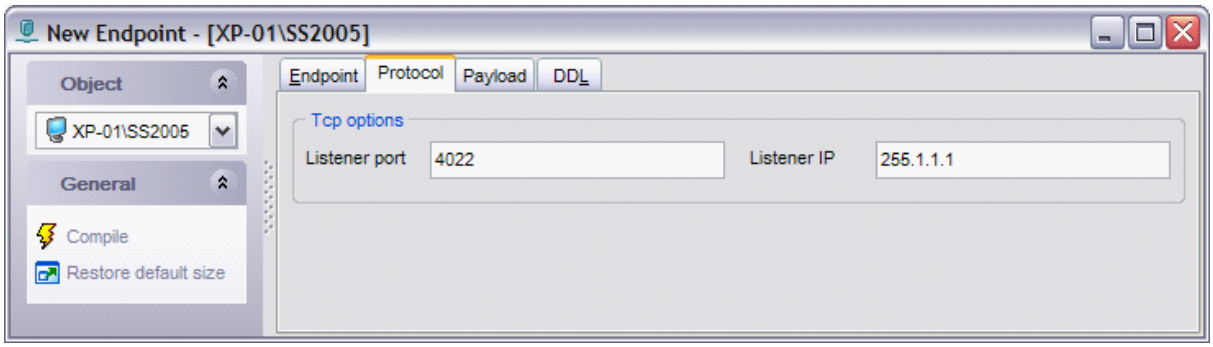
Site
Specify the name of the host computer, or one of the specific signs:
* (asterisk)
Implies that a listening operation applies to all possible host names for the computer that are not otherwise explicitly reserved.
+ (plus sign)
Implies that a listening operation applies to all possible host names for the computer.

☑ **Compression**
If this option is selected, SQL Server will honor requests where gzip encoding is accepted, and return compressed responses. That is, if a request comes in with an HTTP header specifying GZIP as a valid "accept-encoding", the server returns the response gzip-encoded.

TCP options

When TCP/IP is enabled for SQL Server, the Database Engine will listen for incoming connections on a connection point consisting of an IP address and TCP port number.

© 2008 EMS Database Management Solutions

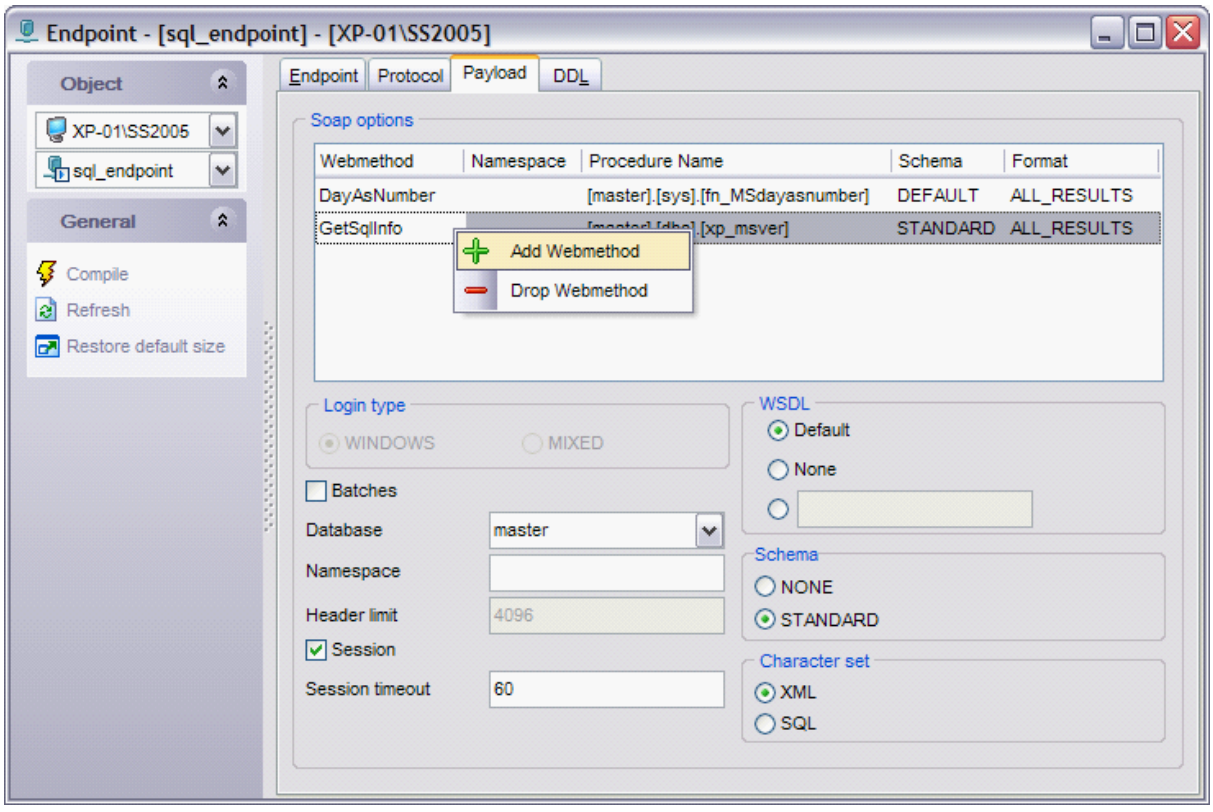


Listener port
Specify the port number that will be listened to for connections by the service broker TCP/IP protocol. The default port number is *4022*.

Listener IP
Specifies the IP address that the endpoint will listen on.

10.2.12.1.4 Setting payload options

The **Payload** tab allows you to specify payload options pertaining to the endpoint.



© 2008 EMS Database Management Solutions

Soap options

The upper area of the group displays the webmethods (the methods for which HTTP SOAP requests can be sent to the endpoint) as a grid with the following columns: *Webmethod*, *Namespace*, *Procedure name*, *Schema*, *Format*.

Right-click an item within the list to call the **context menu** allowing you to *create* a new method or *drop* the selected one.

Login type
Specify the SQL Server authentication mode for the endpoint:
WINDOWS
This option allows only Windows authentication to be used to authenticate endpoint users.
MIXED
This option allows either SQL Server authentication or Windows authentication to be used to authenticate endpoint users.

Batches
This option specifies whether special SQL requests are supported on the endpoint. If checked, SOAP requests for queries that use the *sqlbatch* method can be sent to this endpoint.

Database
Use the drop-down list to select the database in the context of which the requested operation will be executed.

Namespace
Specify the namespace for the endpoint.

Header limit
This value specifies the maximum size, of the header section in the SOAP envelope (in bytes).

Session
This option specifies whether the instance of SQL Server allows sessions support. If checked, SQL Server allows sessions support, whereby multiple SOAP request/response message pairs can be identified as part of a single SOAP session.

Session timeout
Specify time that is available before a SOAP session expires at the server when no further requests are received (in seconds).

WSDL
This group indicates whether WSDL document generation is supported for this endpoint:
None, *Default* (a default WSDL response is generated and returned for WSDL queries submitted to the endpoint), or specify stored procedure by name that will return a modified WSDL document.

Schema
This group specifies whether an XSD schema is returned by the endpoint when SOAP results are sent.

© 2008 EMS Database Management Solutions

NONE
This option omits inline schema from SOAP responses.
STANDARD
This option includes inline schema in endpoint responses.

Character set
This group defines the character set behavior: *XML* (by default), or *SQL* (the characters that are not valid as character references are encoded, and then returned in the result).

10.2.13 DDL Triggers

A **trigger** automatically executes when an event occurs in the database server.

DDL Triggers are a special kind of trigger that fire in response to Data Definition Language (DDL) statements. They can be used to perform administrative tasks in the database server.

- DDL Trigger Editor

Creating DDL Triggers

To create a new DDL trigger:
- right-click the **DDL Triggers** node (within the **Server Objects** branch) or any object within this node in the DB Explorer tree and select the **New DDL Trigger** item from the context menu;
- define DDL trigger properties using the appropriate tabs of DDL Trigger Editor.

or
- select the **Database | New Object...** main menu item;
- select **DDL Trigger** in the Create New Object dialog;
- define DDL trigger properties using DDL Trigger Editor.

To create a new DDL trigger with the same properties as one of existing DDL triggers has:
- select the **Database | Duplicate Object...** menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a DDL trigger in the DB Explorer tree and select the **Duplicate DDL Trigger <DDL_trigger_name>...** context menu item.

Duplicate Object Wizard allows you to select the server to create a new DDL trigger and to edit the result SQL statement for creating the DDL trigger.

Editing DDL Triggers

To edit an existing DDL trigger:
- select the DDL trigger for editing in the DB Explorer tree (type the first letters of the DDL trigger name for quick search);
- right-click the object and select the **Edit DDL Trigger <DDL_trigger_name>** context menu item, or simply double-click the DDL trigger;
- edit DDL trigger properties using the appropriate tabs of DDL Trigger Editor.

To change the name of a DDL trigger:
- select the DDL trigger to rename in the DB Explorer tree;
- right-click the DDL trigger alias and select the **Rename DDL Trigger**

© 2008 EMS Database Management Solutions

<DDL_trigger_name>... item from the context menu;
- edit the DDL trigger name using the **Rename Object...** dialog.

Dropping DDL Triggers

To drop a DDL trigger:
- select the DDL trigger to drop in the DB Explorer tree;
- right-click the object and select the **Drop DDL Trigger <DDL_trigger_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new DDL trigger;
Ctrl+O to edit the selected DDL trigger;
Ctrl+R to rename the DDL trigger;
Shift+Del to drop the object.

See also:
Backup Devices
Database Snapshots
Linked Servers
DTS Packages
Jobs
Job History
Alerts
Operators
Schedules
Proxies
Credentials
Endpoints
Target Servers
Server Log Viewer
Activity Monitor

10.2.13.1 DDL Trigger Editor

DDL Trigger Editor allows you to define DDL trigger properties (the scope of the DDL triggers is applied to the current server, i.e. the trigger fires whenever *event_type* or *event_group* happens anywhere in the current server). It opens automatically when you create a new DDL trigger and is available on editing an existing one (see Create DDL trigger and Edit DDL trigger for details).

To open a DDL trigger in **DDL Trigger Editor**, double-click it in the DB Explorer tree. Alternatively, you can right-click the DDL trigger alias and select the **Edit DDL Trigger** context menu item.
- Using Navigation bar

© 2008 EMS Database Management Solutions

- Editing DDL trigger definition
- Viewing DDL definition

Availability:

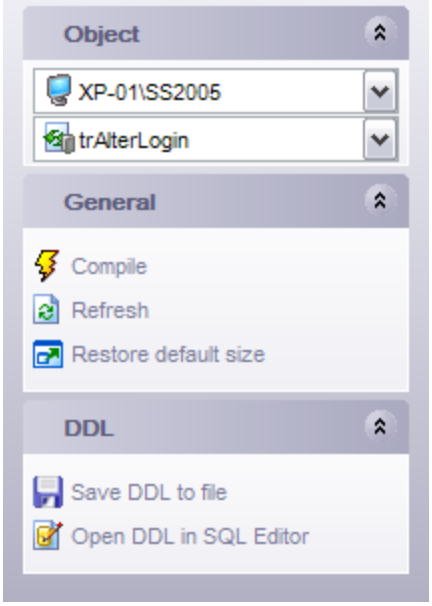
Full version (for Windows)	**Yes**
Lite version (for Windows)	**No**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

10.2.13.1.1 Using Navigation bar

The **Navigation bar** of **DDL Trigger Editor** allows you to:

- select the server where the DDL trigger is located;
- select the DDL trigger for editing;
- compile the object (if it is being created/modified);
- refresh the content of the active tab;
- restore the default size of the editor window;
- save the trigger DDL definition to an external **.sql* file;
- open the trigger DDL definition in SQL Editor.



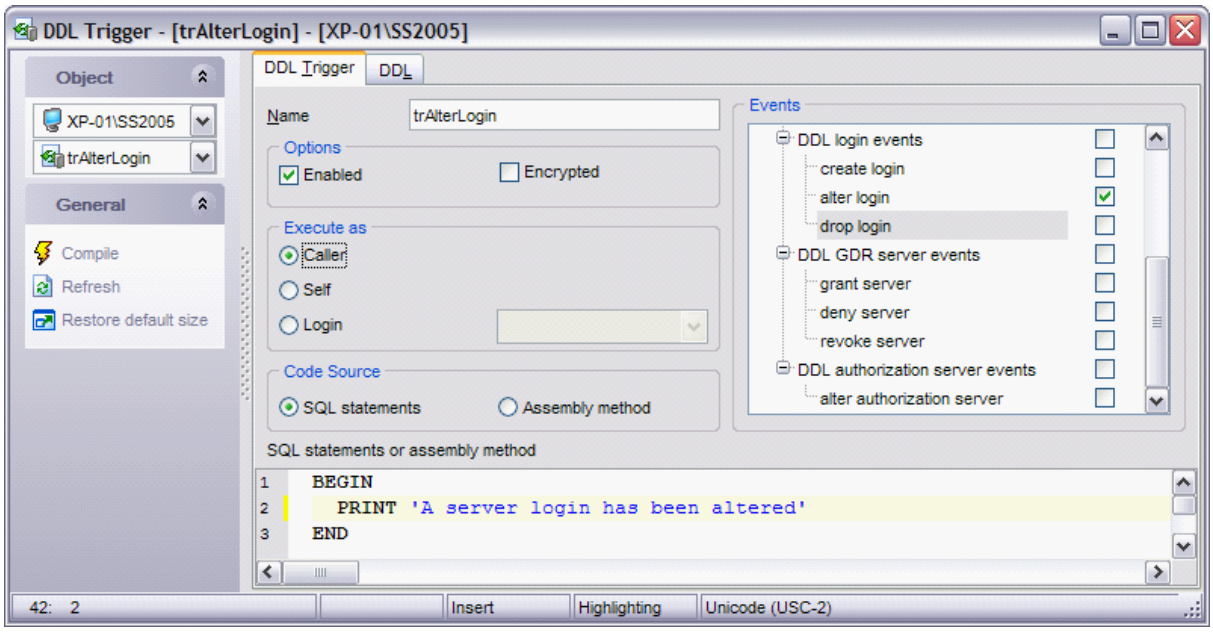
<< Back to DDL Trigger Editor

© 2008 EMS Database Management Solutions

10.2.13.1.2 Editing DDL trigger definition

Use the **DDL Trigger** tab of **DDL Trigger Editor** to configure/view DDL trigger properties.

Name
Enter a name for the new DDL trigger, or modify the name of the DDL trigger being edited.



Enabled
Enables/disables the trigger immediately after it is created. A disabled trigger still exists as an object in the server, but does not fire.

Encrypted
If this option is selected, the text of the *CREATE TRIGGER* statement is encrypted.

Execute as
Specify the security context under which the trigger is to be executed: *Caller*, *Self* or *Login* (select which login account should be used to validate permissions on any server objects that are referenced by the trigger).

Code Source
Select the trigger code source: it can be a direct set of SQL statements or an external procedure call from a .NET assembly.

Events
This area lists the DDL events as groups. The trigger will fire after execution of any data definition language event selected in this list. Set the corresponding flag next to an event or event group name.

SQL statements or assembly method
This area allows you to set the trigger conditions and actions. Trigger conditions specify additional criteria that determine whether the tried DDL statements cause the trigger

© 2008 EMS Database Management Solutions

actions to be performed.
The trigger actions take effect when the DDL operation is performed.

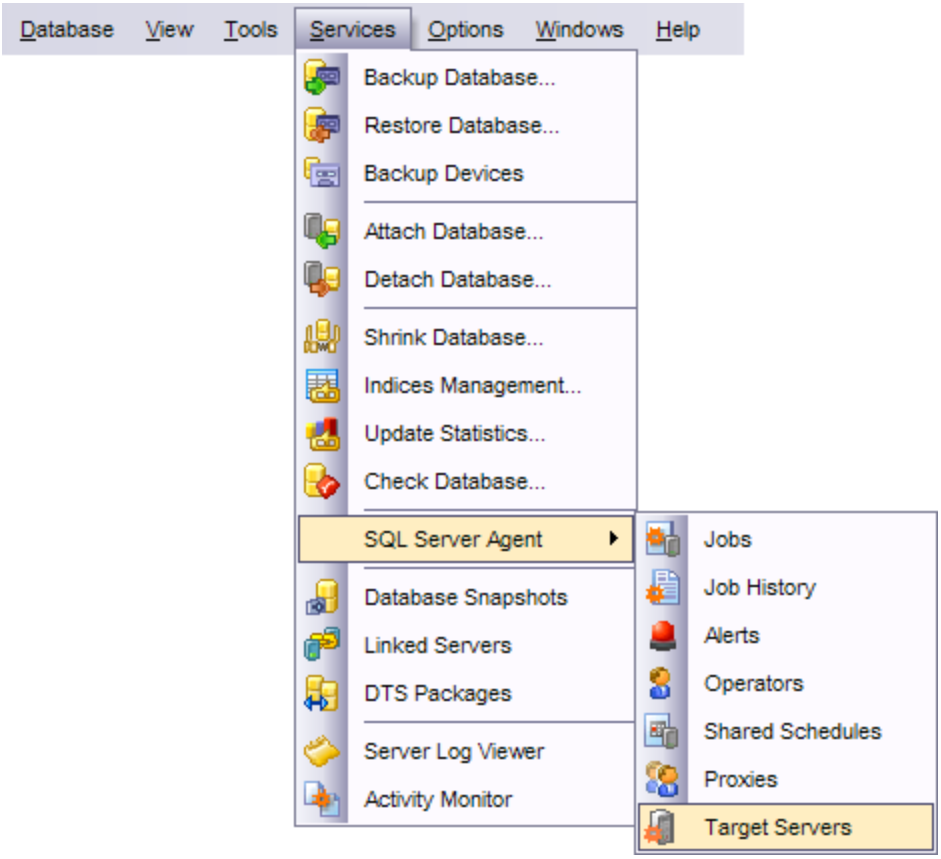
10.2.14 Target Servers

Multiserver administration which consists in automating administration across multiple instances of SQL Server is performed when there is a necessity to manage two or more servers and to schedule information flows between enterprise servers for data warehousing.

In a multiserver system, there is at least one **master server** and at least one **target server**.
A *master server* distributes jobs to *target servers*, and receives events from them. A master server also stores the central copy of job definitions for jobs that are run on target servers. Target servers connect periodically to the master server to update their schedule of jobs. If a new job exists on the master server, the target server downloads the job. After the target server completes the job, it reconnects to the master server and reports the status of the job.

Before using target servers, make sure that the **SQL Server Agent** service is running.

- Working with Target Servers manager



Availability:

© 2008 EMS Database Management Solutions

Full version (for Windows)	**Yes**
Lite version (for Windows)	**No**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

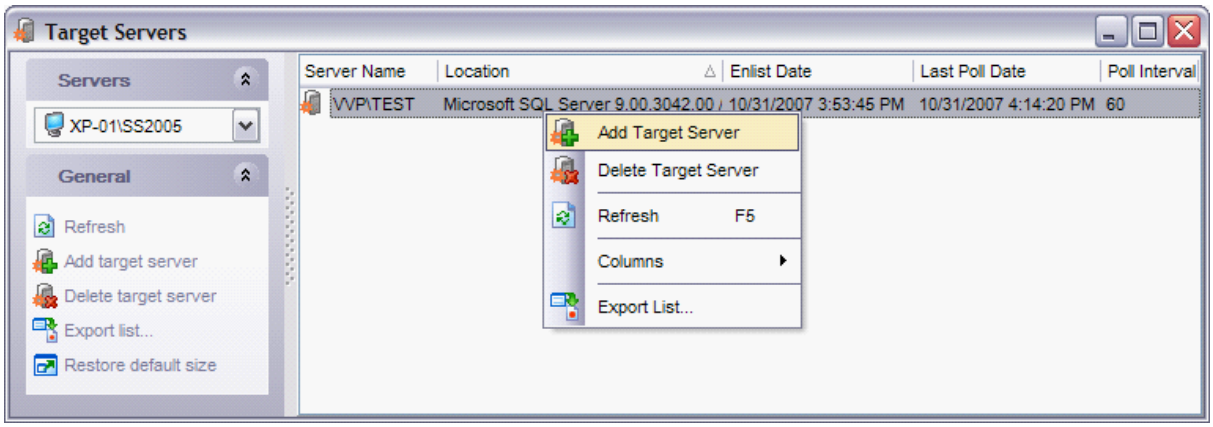
Backup Devices
Database Snapshots
Linked Servers
DTS Packages
Jobs
Job History
Alerts
Operators
Schedules
Proxies
Credentials
Endpoints
DDL Triggers
Server Log Viewer
Activity Monitor

10.2.14.1 Target Servers manager

The **Target Servers manager** is intended for easy and efficient management of target servers when multiserver administration is performed. The **Target Servers manager** allows you to browse the list of **target servers** and manage them efficiently.

To launch the tool, select the **Services | SQL Server Agent | Target Servers** main menu item, or right-click the host alias in the DB Explorer tree and select the **Tasks | Target Servers** item from the context menu.

© 2008 EMS Database Management Solutions

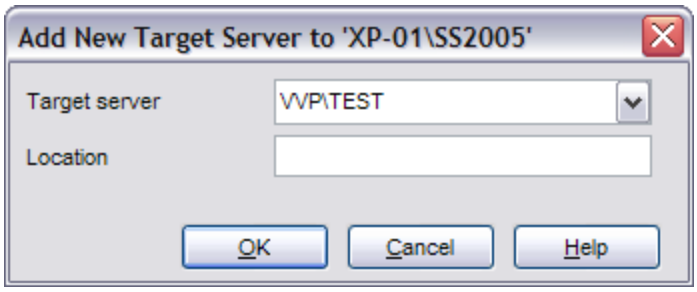


The list displays the target servers as a grid with the following columns: *Server Name*, *Location*, *Enlist Date*, *Last Poll Date*, *Poll Interval*.

Right-click an item within the list to call the **context menu** allowing you to *create* a new target server using the **Add New Target Server** dialog, *delete* the selected target server, *refresh* the list, or show/hide columns of the list. Using the context menu you can also export the list of target servers to any of supported output file formats.

Target servers management tasks are also available through the **Navigation bar** of the **Target Servers manager**.

The **Add New Target Server** dialog allows you to register a new target server.



Target server
Type in or use the drop-down list to select the server instance to be registered.

Location
Specify the target server location.

10.2.15 Server Log Viewer

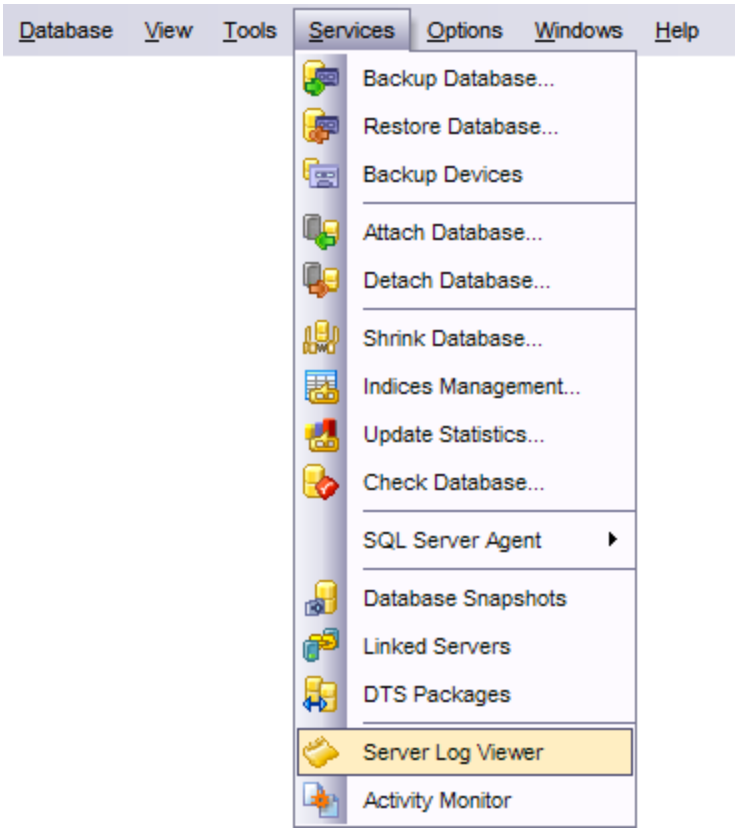
Server Log Viewer allows user to analyze SQL Server server logs in a simple way, providing the ability to sort, filter and group log statements.

To launch the tool, select the **Services | Server Log Viewer** main menu item, or right-click the host alias in the DB Explorer tree and select the **Tasks | Server Log**

© 2008 EMS Database Management Solutions

Viewer item from the context menu.

- Using Navigation bar
- Working with Server Log Viewer



Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**No**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

Backup Devices
Database Snapshots
Linked Servers
DTS Packages
Jobs
Job History
Alerts

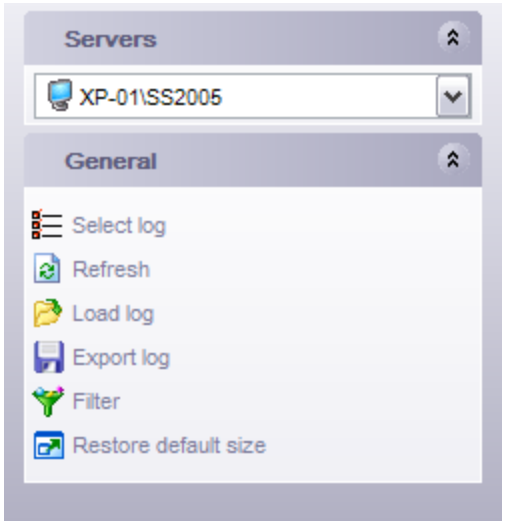
© 2008 EMS Database Management Solutions

Operators
Schedules
Proxies
Credentials
Endpoints
DDL Triggers
Target Servers
Activity Monitor

10.2.15.1 Using Navigation bar

The **Navigation bar** of **Server Log Viewer** allows you to:

- select the instance of SQL Server to view server logs;
- select a log;
- refresh the content of the window;
- load a log from an external **.txt* file;
- export log to an external **.txt* file;
- filter logs in the list;
- restore the default size of the window.

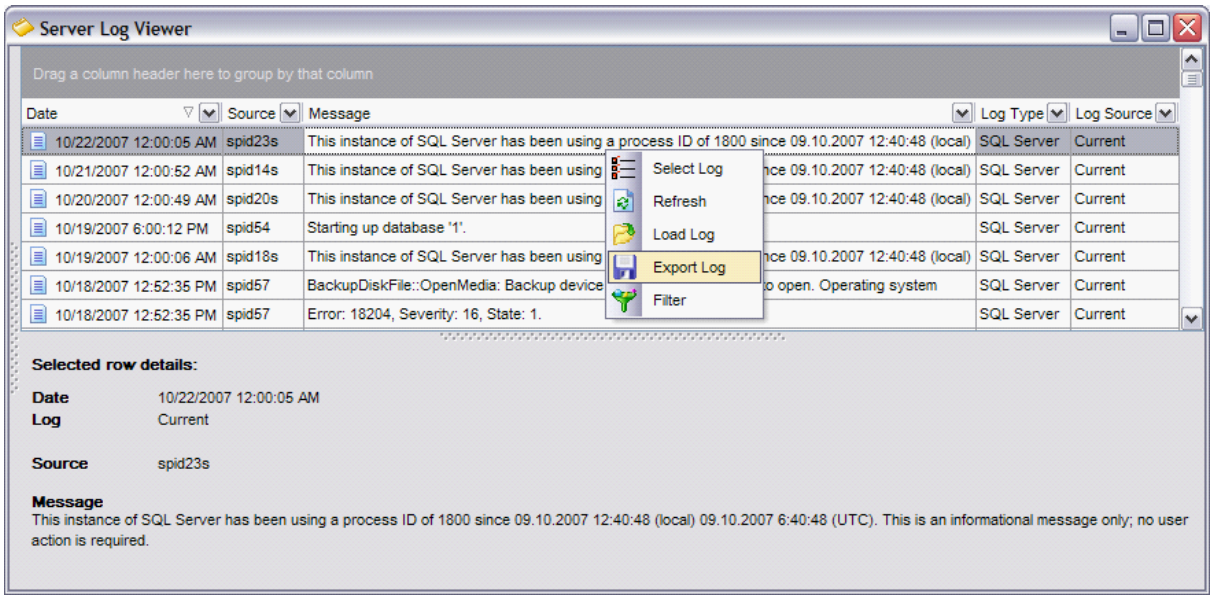


<< Back to Server Log Viewer

10.2.15.2 Working with Server Log Viewer

The **Server Log Viewer** window displays the list of server logs as a grid, and allows you to manage them efficiently.

© 2008 EMS Database Management Solutions



The list displays the server logs as a grid with the following columns: *Date*, *Source*, *Message*, *Log Type*, *Log Source*.

If necessary, you can **group the data in grid** by any of the columns. This operation is performed by dragging the column header to the gray **"Group by" box** area at the top. When grouping by a column is applied to the grid, all the rows are displayed as subnodes to the grouping row value. To reverse grouping, just drag the column header back.

Right-click an item within the list to call the **context menu** allowing you to *select* a log, *refresh* the list, load a log, export log and filter the logs in the grid.

Server logs management tasks are also available through the Navigation bar of **Server Log Viewer**.

The lower area displays the following server log details:
Date
Log
Source
Message

10.2.16 Activity Monitor

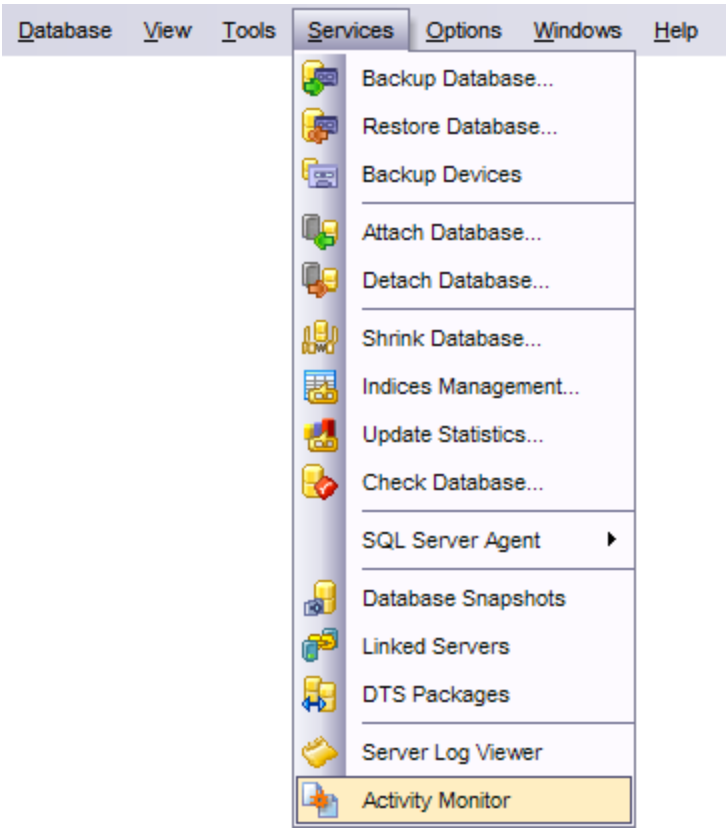
Activity Monitor allows you to get information about users connections to the Database Engine and the locks that they hold.

Use **Activity Monitor** when troubleshooting database locking issues, and to terminate a deadlocked or otherwise unresponsive process.

To launch the tool, select the **Services | Activity Monitor** main menu item, or right-click the host alias in the DB Explorer tree and select the **Tasks | Activity Monitor** item from the context menu.

© 2008 EMS Database Management Solutions

- Using Navigation bar
- Working with Activity Monitor



Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**No**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

Backup Devices
Database Snapshots
Linked Servers
DTS Packages
Jobs
Job History
Alerts
Operators

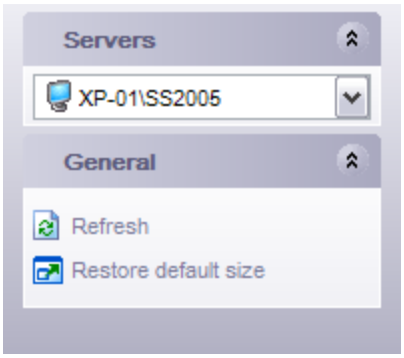
© 2008 EMS Database Management Solutions

Schedules
Proxies
Credentials
Endpoints
DDL Triggers
Target Servers
Server Log Viewer

10.2.16.1 Using Navigation bar

The **Navigation bar** of **Activity Monitor** allows you to:

- select the instance of SQL Server for activity monitoring;
- restore the default size of the window.

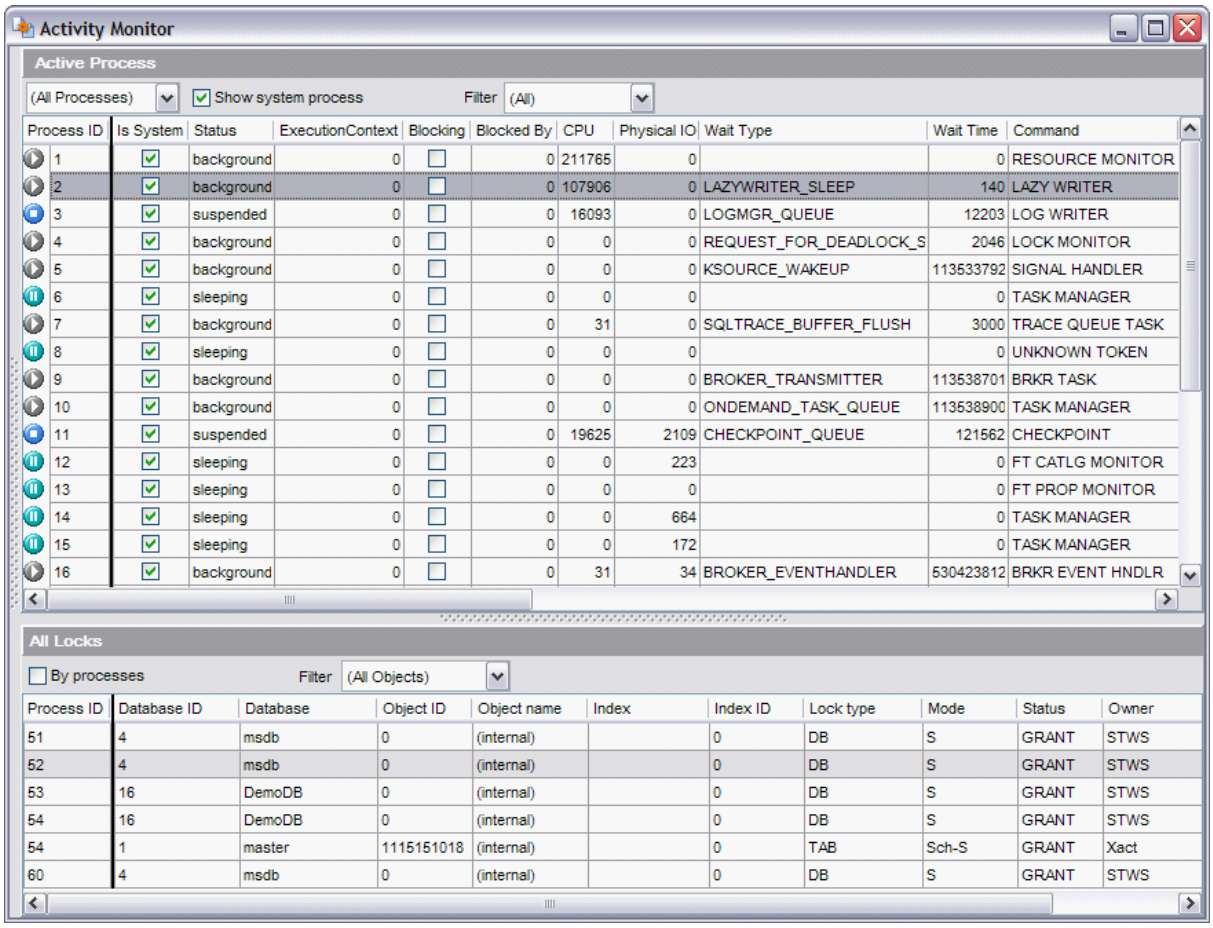


<< Back to Activity Monitor

10.2.16.2 Working with Activity Monitor

The **Activity Monitor** window consists of two basic areas: **Active Process** and **All Locks** .

© 2008 EMS Database Management Solutions



Hint: Background/suspended/sleeping processes are differentiated in the list by their icons next to the *Process ID* values.

The **Active Process** list displays the processes as a grid with the following columns: *Process ID*, *Is System*, *Status*, *ExecutionContext*, *Blocking*, *Blocked By*, *CPU*, *Physical IO*, *Wait Type*, *Wait Time*, *Command*, *Application*, *Open Transactions*, *Database*, *User*, *Net Address*, *Net Library*, *Host*, *Memory Usage*, *Login Time*, *Last Batch*.

For your convenience several filtering facilities are implemented: you can filter the rows in the **Active Process** list by process ID, by system/non-system attribute, or by status (*background*, *suspended*, *sleeping*, *running*). Use the corresponding controls in the upper area.

The **All Locks** list provides the following attributes of each lock: *Process ID*, *Database ID*, *Database*, *Object ID*, *Object name*, *Index*, *Index ID*, *Lock type*, *Mode*, *Status*, *Owner*, *Resource*.

If necessary, you can filter the rows in the **All Locks** list by processes or display all objects/internal.

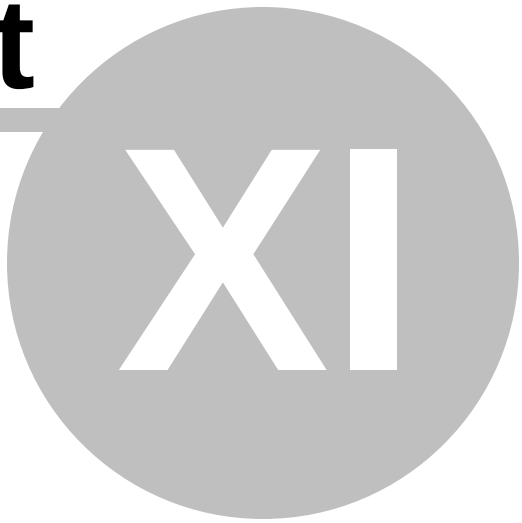
Right-click an item within the **Active Process** or **All Locks** lists to call the **context menu**

© 2008 EMS Database Management Solutions

allowing you to *kill* a process, show *SQL text*, or show/hide columns of the list. Using the context menu you can also export the list of processes to any of supported output file formats.

© 2008 EMS Database Management Solutions

Part



11 Personalization

SQL Manager 2008 for SQL Server provides you with capabilities for flexible personalization of the application.

Please see the chapters below to learn how to use personalization tools effectively.

- SQL Manager Options
- Localization
- Keyboard templates
- Find Option dialog

See also:

Getting started
Database Explorer
Database management
Database objects management
Query management tools
Data management
Import/Export tools
Database tools
Server tools
External tools

© 2008 EMS Database Management Solutions

11.1 SQL Manager Options

SQL Manager 2008 for SQL Server provides full customization of the program interface, windows behavior, various options of all tools and more. This chapter informs you how to use these options.

The following list contains the three basic groups of **options** available in SQL Manager 2008 for SQL Server.

Environment Options
General SQL Manager options: environment style, confirmations, window restrictions, DB Explorer, SQL Editor, SQL Monitor, Visual Query Builder options, and more.

Editor Options
Customizing of all the SQL editing tools - SQL Editor, SQL Script, and more.

Visual Options
Customizing program interface - bars, trees, menus, etc.

Besides, the **Options** menu allows you to export all program settings to a **.reg* file for future use, e.g. when you need to move the settings to another machine (see **Save Settings** for details).

See also:
Localization
Keyboard Templates
Find Option dialog

11.1.1 Environment Options

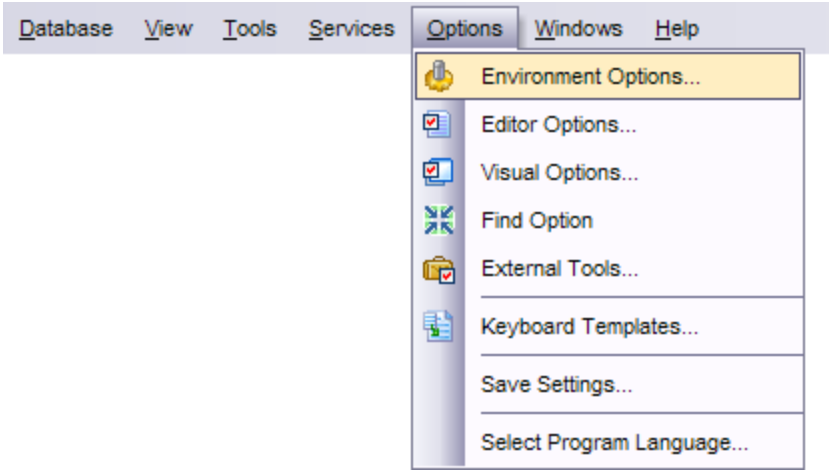
Environment Options allow you to customize general options of the SQL Manager application.

To open the **Environment Options** window, select the **Options | Environment Options...** main menu item, or use the **Environment Options** button on the main toolbar.

- **Preferences**
- **Full mode activation**
- **Confirmations**
- **Windows**
- **Tools**
- Timeouts
- DB Explorer
- Naming Rules
- Object Editors
- SQL Editor

© 2008 EMS Database Management Solutions

- SQL Monitor
- SQL Script
- **Query Builder**
- Style & Color Palette
- **Visual Database Designer**
- Color Palette
- Print Metadata
- Data Export
- SQL Server Reference
- **Fonts**
- **Grid**
- Data Options
- Print Data
- Color & Formats
- Advanced
- Column Options
- **Localization**
- **Find Option**



Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**Yes**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

Editor Options
Visual Options
Save Settings

© 2008 EMS Database Management Solutions

11.1.1.1 Preferences

Show splash screen at startup
Displays the splash screen of SQL Manager 2008 for SQL Server at the application startup.

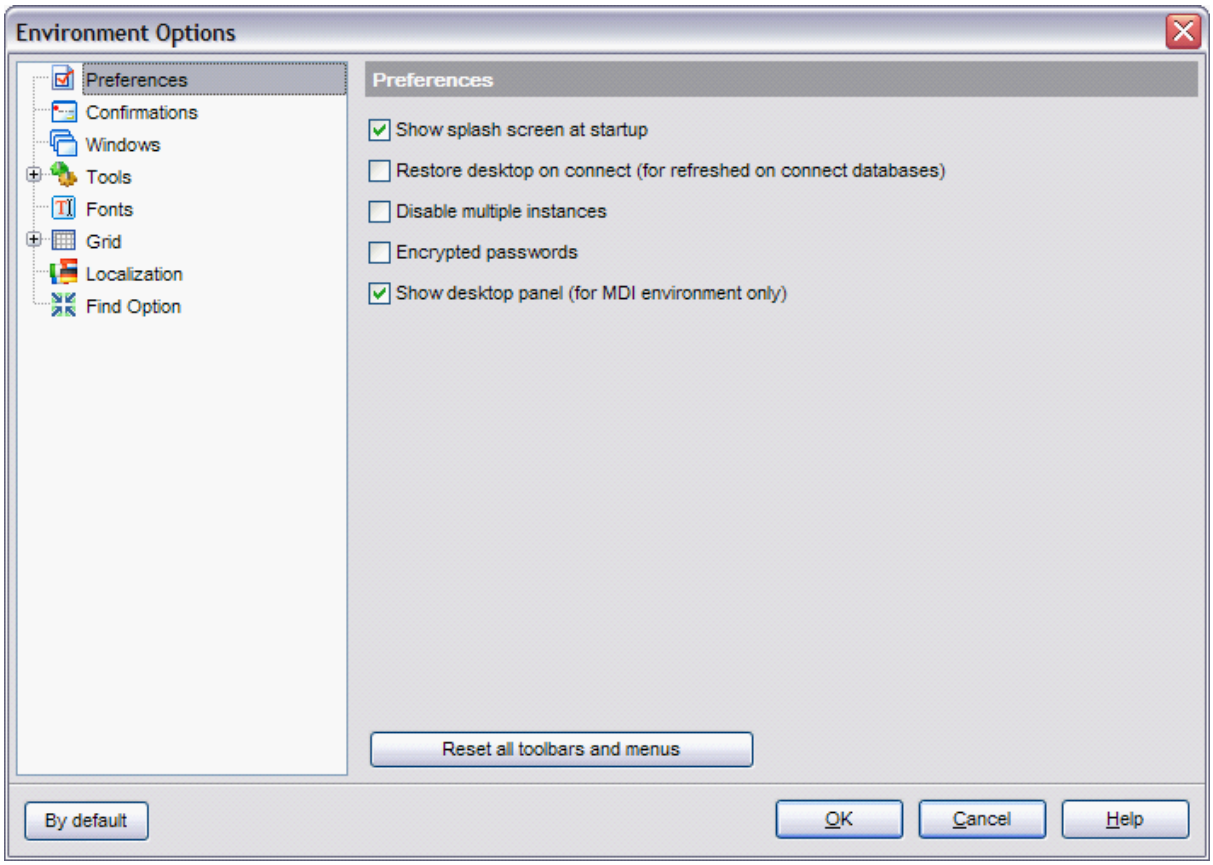
Restore desktop on connect
This option determines whether the previously opened windows and their positions should be restored upon connection to the database.

Disable multiple instances
Checking this option prevents one from running multiple instances of SQL Manager 2008 for SQL Server.

Encrypted passwords
Encrypts passwords for connecting to databases in Windows registry.

Show desktop panel (for MDI Environment style only)
Displays Desktop Panel when no child windows are open.

Show Full Version features
This option is available in the Lite version of SQL Manager. When selected, a 30-day period of fully-functional usage is activated.

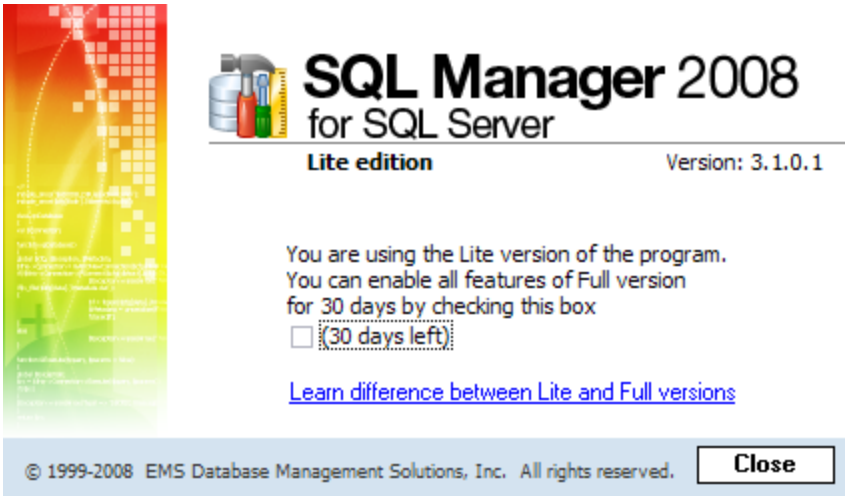


© 2008 EMS Database Management Solutions

See also:
Full mode activation
Confirmations
Windows
Tools
Fonts
Grid
Localization
Find Option

11.1.1.2 Full mode activation

Note that when using **the FREE Lite version of SQL Manager 2008 for SQL Server** (which contains functional limitations) you can activate a 30-day period of fully-functional usage. During this period you will get the splash screen displaying the number of days left every time you start the application. After the period expires, you will be able to continue using the Lite version.



To activate *the 30-day Full version mode*, please enable the **Show Full Version features** option available on the Preferences page of the **Environment Options** dialog (note that this option is only available in the Lite version of SQL Manager).

See also:
Preferences
Confirmations
Windows
Tools
Fonts

© 2008 EMS Database Management Solutions

Grid
Localization
Find Option

11.1.1.3 Confirmations

Confirm saving the object (or document) upon closing the editor
If this option is on, the program requires confirmation each time you want to save changes in a database object or document.

Confirm dropping of object
If this option is checked, the program requires confirmation of dropping a database object.

Confirm exit from SQL Manager
If this option is set, you are prompted for confirmation each time when you exit the application.

Confirm deleting records
This option enables/disables a confirmation dialog for deleting records.

Confirm metadata changing (Changing Metadata Window)
This option enables/disables the Changing Metadata window.

Confirm addition into spell checking dictionary
Check this option if you wish to be prompted for adding a word to the dictionary which is used for checking words spelling (see Spell Checking).

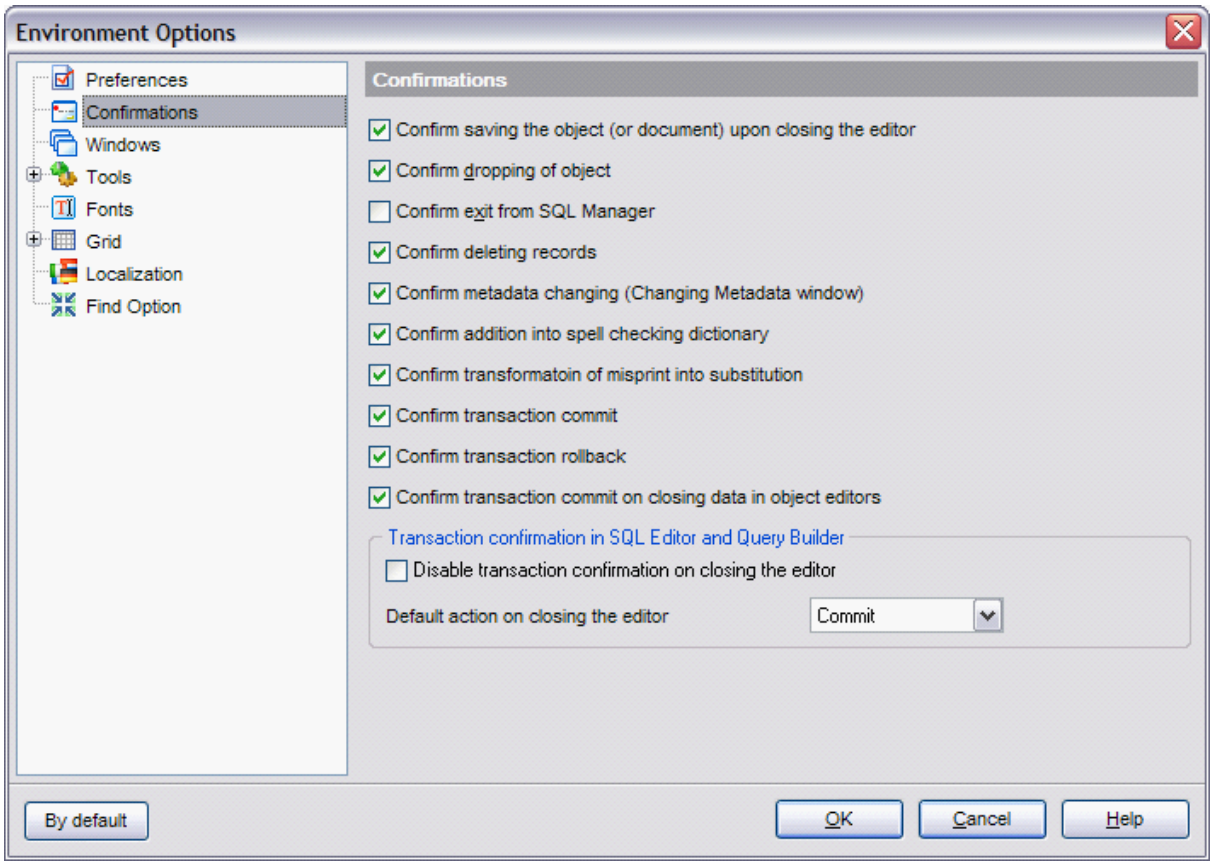
Confirm transformation of misprint into substitution
If this option is checked, you need to confirm replacing a misprinted word with a corresponding substitution word (see Spell Checking).

Confirm transaction commit
If this option is checked, the program requires confirmation on attempt to commit a transaction.

Confirm transaction rollback
If this option is checked, the program requires confirmation on attempt to rollback a transaction.

Confirm transaction commit on closing data in object editors
If this option is checked, the program prompts to commit the transaction upon closing the 'Data' tab in object editors.

© 2008 EMS Database Management Solutions



Transaction confirmation in SQL Editor and Query Builder

Disable transaction confirmation on closing the editor
If this option is checked, no transaction confirmation will be required on closing SQL Editor and Query Builder. Specify the **default action** (*Commit* or *Rollback*) and this action will be performed automatically each time when you exit **Query Builder** or **SQL Editor**.

See also:
Preferences
Full mode activation
Windows
Tools
Fonts
Grid
Localization
Find Option

© 2008 EMS Database Management Solutions

11.1.1.4 Windows

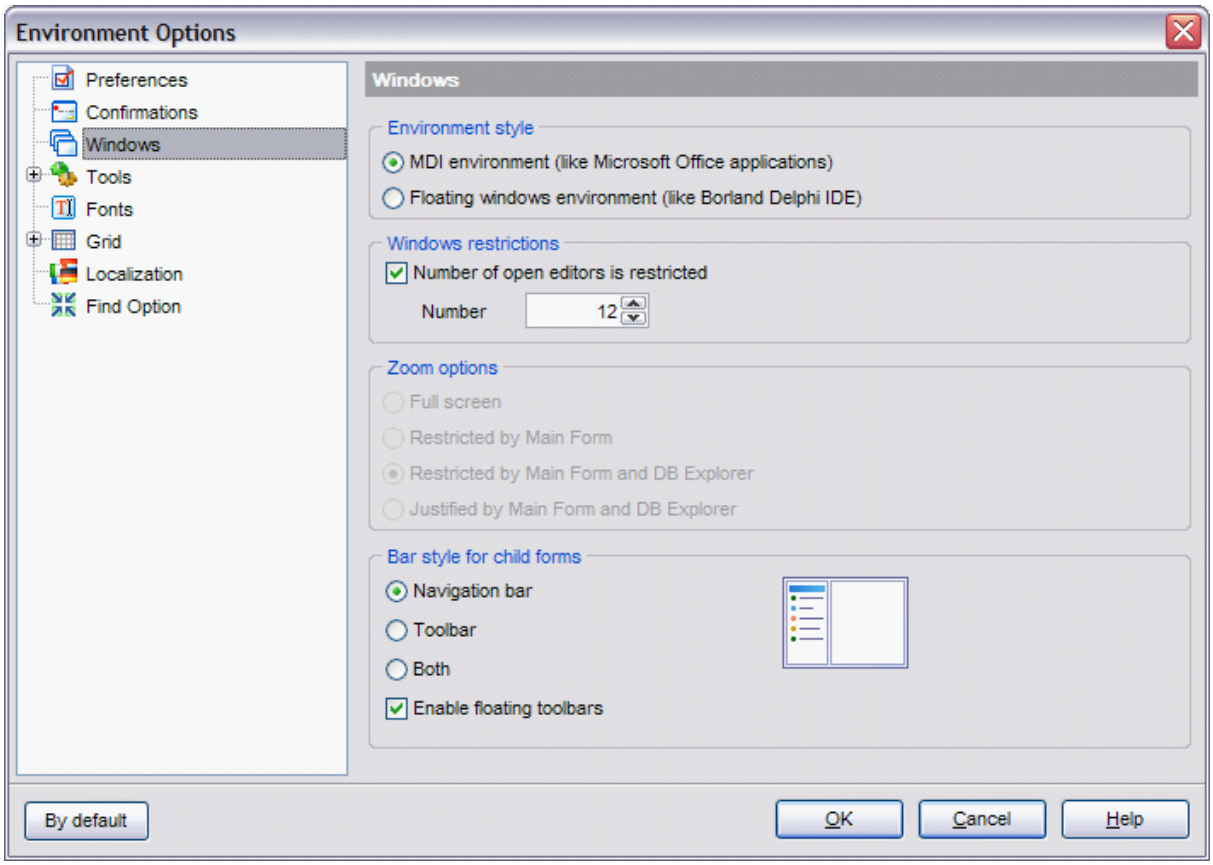
Environment style
This group allows you to define the basic window environment - *MDI* (like Microsoft® Office) or *Floating windows* (like Borland® Delphi IDE).

Windows restrictions
This option allows you to set the number of editors (Table Editor, SQL Editor, etc.) that can be opened simultaneously.

Zoom options
This group of options is only available if **Environment Style** is set to *Floating windows environment*. It allows you to set maximization size for child windows:
- *Full screen*
- *Restricted by Main Form*
- *Restricted by Main Form and DB Explorer*
- *Justified my Main Form and DB Explorer*

Bar style for child forms
Here you can define the location of action buttons: within the *Navigation bar* (on the left) and/or on the *Toolbar*.

If necessary, you can also **Enable floating toolbars** for your application.



© 2008 EMS Database Management Solutions

See also:

Preferences
Full mode activation
Confirmations
Tools
Fonts
Grid
Localization
Find Option

11.1.1.5 Tools

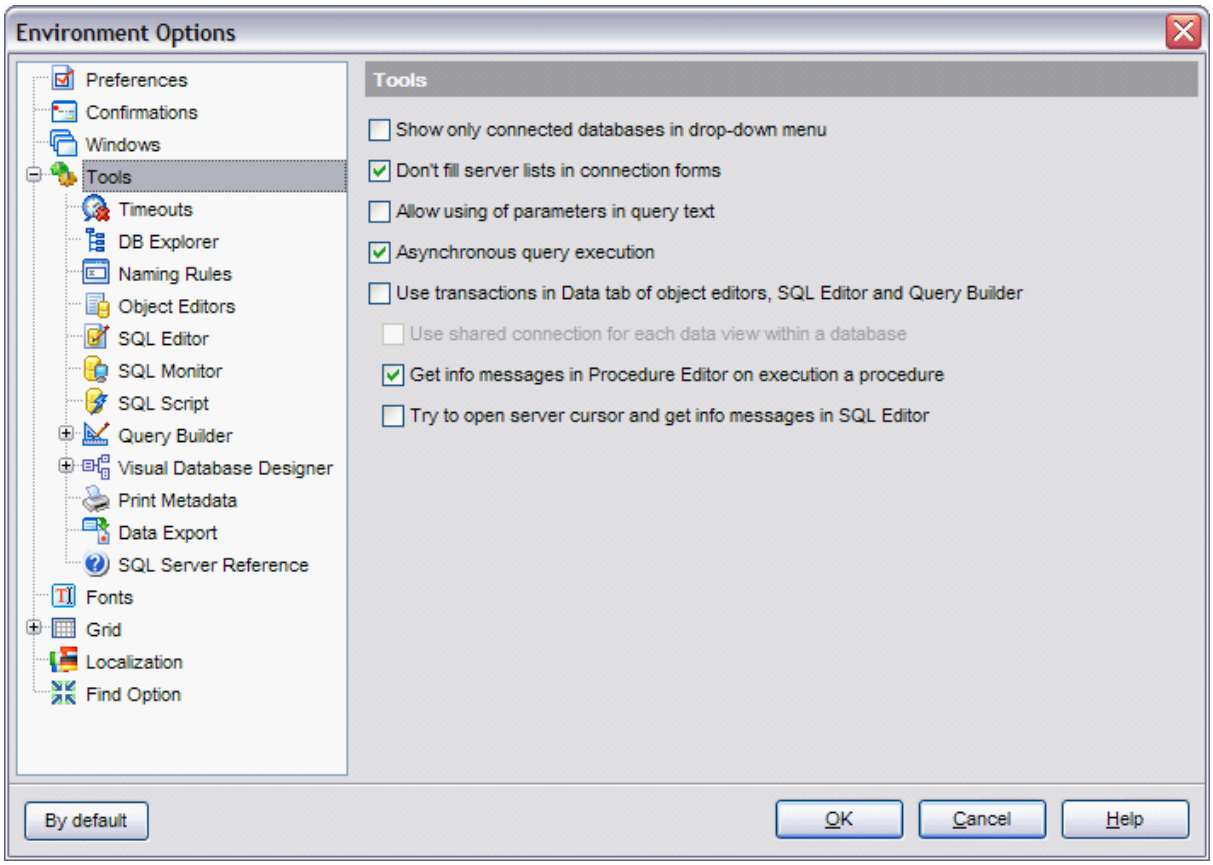
Show only connected databases in drop-down menu
If this option is checked, only connected databases are displayed in drop-down menus of such tools as Query Builder, SQL Script, etc.

Don't fill server lists in connection forms
If this option is checked, the server list will be filled up with the names of most recently used servers that are stored locally in the Windows registry. Otherwise, SQL Manager will scan the network to fill in the server lists in connection forms.

Allow using parameters in query text
This feature allows you to specify different values within a query in a popup dialog just before the query execution. Use the colon (':') character before an identifier (e.g. :P1) to specify a parameter within the query.

Asynchronous query execution
Check this option to allow executing queries in background mode (asynchronously).

© 2008 EMS Database Management Solutions



Use transactions in Data tab of object editors, SQL Editor and Query Builder
If this option is enabled, a transaction is active until the 'Data' tab is closed or the 'Commit' button is pressed; all edited records are blocked until the transaction is committed. If this option is disabled, the transaction starts and is immediately committed (autocommit) on saving each record which is blocked only for a short period of time.

Use shared connection for each data view within a database
Uncheck this option to use separate connection for each data view within a database. Note that this option is only available when the *Use transactions in Data tab of object editors, SQL Editor and Query Builder* option is enabled.

Get info messages in Procedure Editor on execution of a procedure
This option enables/disables information messages returned upon procedure execution.

Try to open server cursor and get info messages in SQL Editor
This option determines whether the application should attempt to open server cursor, and enables/disables information messages returned upon query execution in SQL Editor.

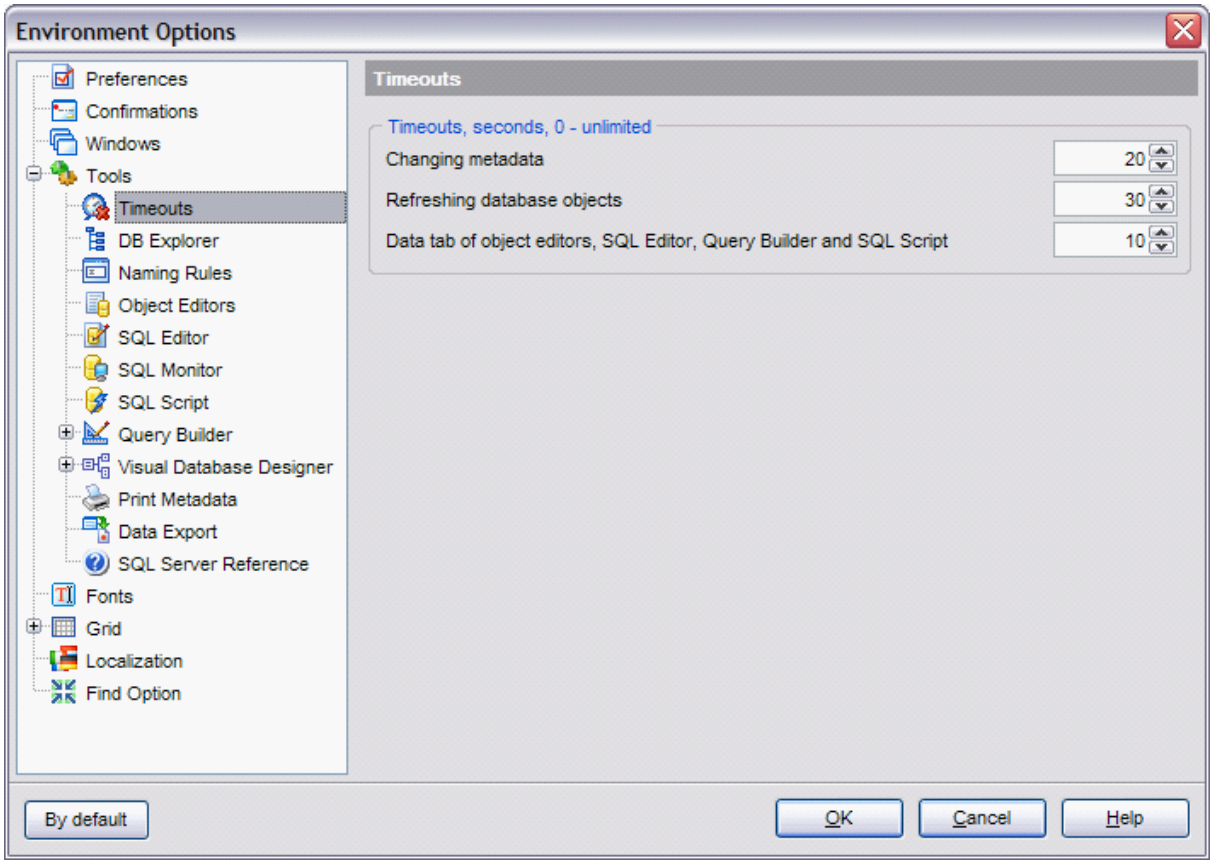
See also:
Preferences
Full mode activation

© 2008 EMS Database Management Solutions

Confirmations
Windows
Fonts
Grid
Localization
Find Option

11.1.1.5.1 Timeouts

This page allows you to set timeouts for some operations performed in SQL Manager. Timeouts are necessary for preventing program hang-up upon execution of a SQL statement due to transaction interlocking. You can set timeouts for *changing metadata*, *refreshing database objects* and *data tabs*.



Refresh of objects can also be terminated after timeout expiration if there is an active transaction that has made any changes to metadata.

11.1.1.5.2 DB Explorer

Show hosts in DB Explorer
Shows/hides database hosts in the DB Explorer tree.

© 2008 EMS Database Management Solutions

Show table subobjects
Shows/hides table subobjects (fields, indexes, etc.) in the DB Explorer tree.

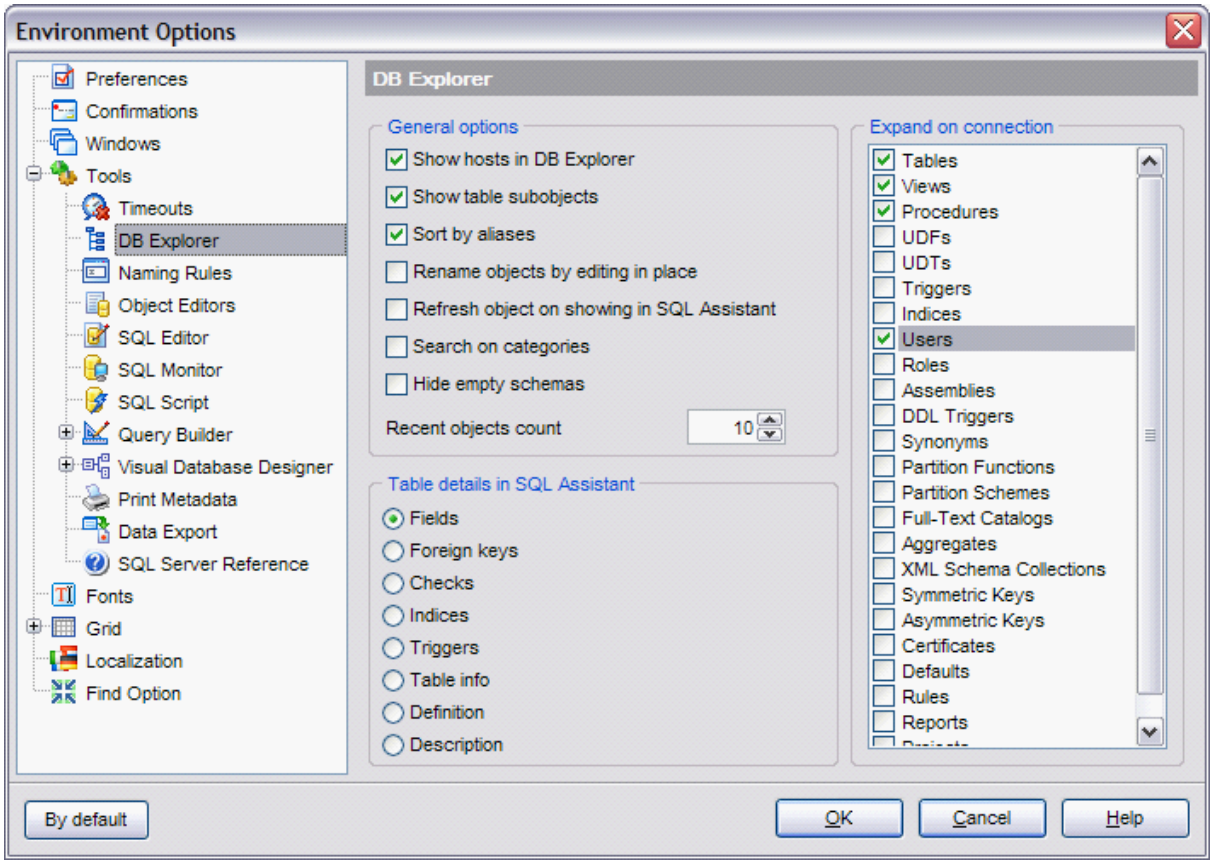
Sort by aliases
Use this option to apply sorting registered hosts and databases by their aliases in the DB Explorer tree.

Rename objects by editing in place
Allows you to edit object names in DB Explorer by selecting any object and clicking its alias one more time.

Refresh objects on showing in SQL Assistant
This option enables/disables refreshing objects each time they are displayed in SQL Assistant.

Search on categories
This option determines the search scope when the Find Item feature is used: if this option is selected, the search is performed within the currently selected category (node in the tree) only.

Hide empty schemas
This option allows you to specify whether schemas containing no objects will be displayed in the DB Explorer tree.



© 2008 EMS Database Management Solutions

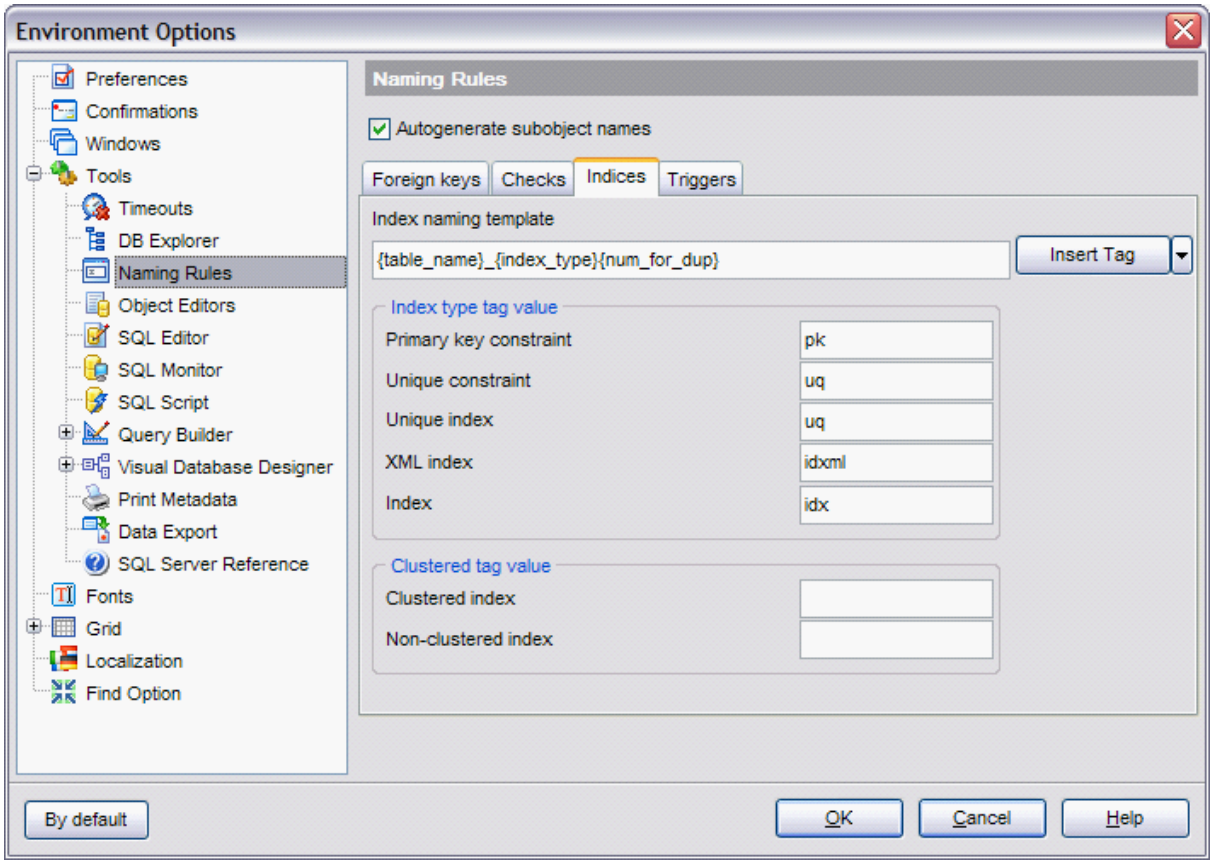
Recent objects count
Defines the number of objects displayed within the Recent menu of the DB Explorer.

Table details in SQL Assistant
These options switch the SQL Assistant mode for displaying table details (*fields*, *Foreign keys*, *checks*, *indexes*, *triggers*, *table info*, *definition* or *description*).

Expand on connection
This group of options allows you to specify the node(s) indicating type(s) of objects that will be expanded within DB Explorer upon successful connection to the database.

11.1.1.5.3 Naming Rules

Autogenerate subobject names
This option allows you to generate default names for newly created foreign keys, checks, indices and triggers automatically.



Using the **Foreign keys** tab you can set a template for foreign key naming. The **Insert Tag** drop-down list allows you to select one or more tags that may simplify template creation:
Table Name: *{table_name}*
Serial Number: *{num}*

© 2008 EMS Database Management Solutions

Serial Number (for duplicates): *{num_for_dup}*
Referenced Table Name: *{ref_table_name}*
Table Fields: *{fields}*
Referenced Table Fields: *{ref_fields}*

Using the **Checks** tab you can set a template for check naming. The **Insert Tag** drop-down list allows you to select one or more tags that may simplify template creation:
Table Name: *{table_name}*
Serial Number: *{num}*
Serial Number (for duplicates): *{num_for_dup}*

Using the **Indices** tab you can set a template for index naming. The **Insert Tag** drop-down list allows you to select one or more tags that may simplify template creation:
Table Name: *{table_name}*
Serial Number: *{num}*
Serial Number (for duplicates): *{num_for_dup}*
Index Type: {index_type}
Clustered: *{clustered}*
Index Fields: *{fields}*
Here you can also specify templates for **tags** added to default names of several index types: *Primary key constraint*, *Unique constraint*, *Unique index*, *XML index*, *Clustered*, *Non-clustered* and simple *index*.

Using the **Triggers** tab you can set a template for trigger naming. The **Insert Tag** drop-down list allows you to select one or more tags that may simplify template creation:
Table Name: *{table_name}*
Serial Number: *{num}*
Serial Number (for duplicates): *{num_for_dup}*
Trigger Type: *{trig_type}*
Trigger Events: *{trig_events}*
Here you can also specify templates for **tags** added to default names of *After* and *Instead of* trigger types and for **tag parts** added to trigger events: *Insert*, *Update*, *Delete*.

Use the **Insert Tag** button to add the selected tag value to the template, e.g. *{table_name}*, *{ref_fields}*, etc.

11.1.1.5.4 Object Editors

All

Convert created objects' names to lower case
Enable this option if you need to convert the names of all newly created objects to the lower case automatically.

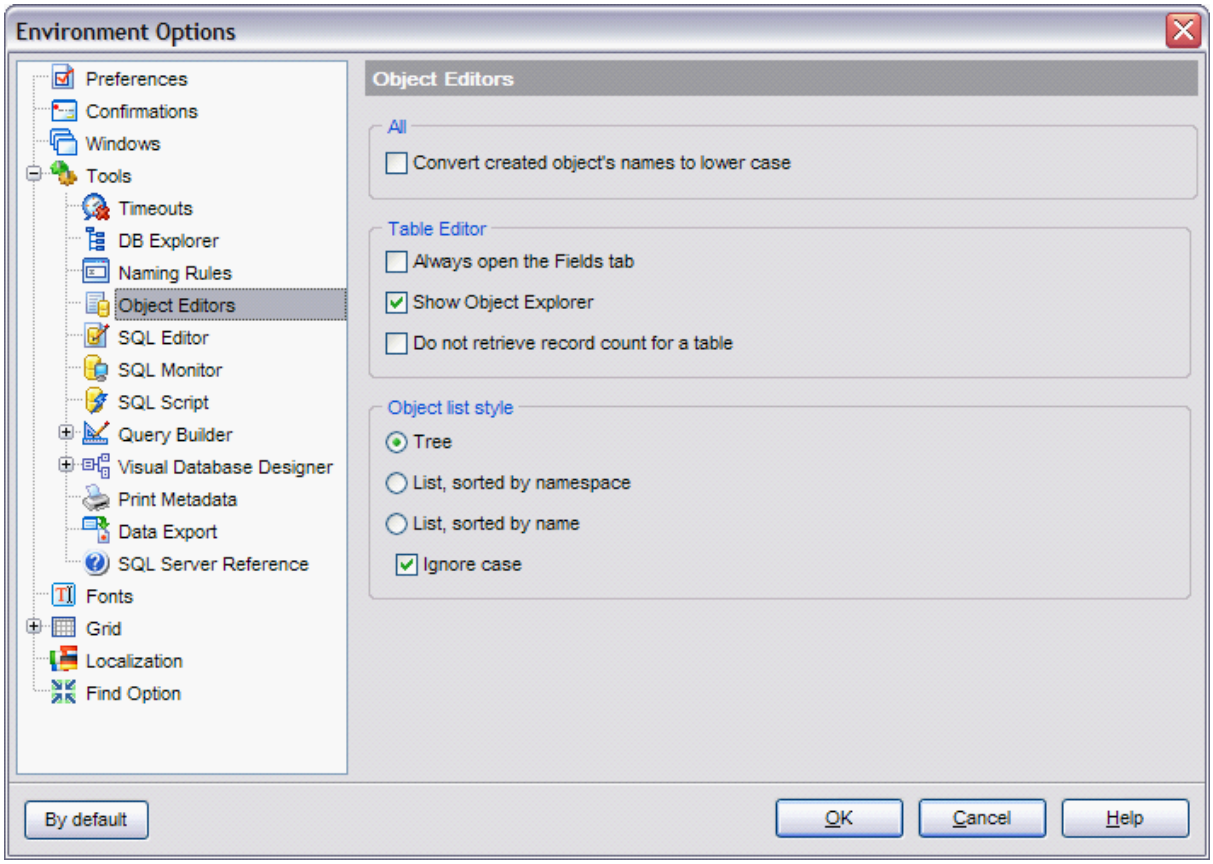
Table Editor

Always open the Fields tab
If this option is checked, the Fields tab is activated by default on opening a table in Table Editor.

Show Object Explorer
Enables/disables the Object Explorer panel within the Navigation bar of Table Editor.

© 2008 EMS Database Management Solutions

Do not retrieve record count for a table
Check this option to disable retrieving record count for tables (with this feature enabled, opening large tables may take much time).



Object list style

These options allow you to define the style of the combo-boxes used to select database objects (e.g. *Table or view* in Trigger Editor). Objects can be represented as a *tree*, a *list sorted by namespace* or a *list sorted by name*. Use the **Ignore case** option to enable/disable case sensitive sorting.

11.1.1.5.5 SQL Editor

Show actual execution plan on query execution
If this option is checked, the query plan is displayed at the bottom of the SQL Editor window.

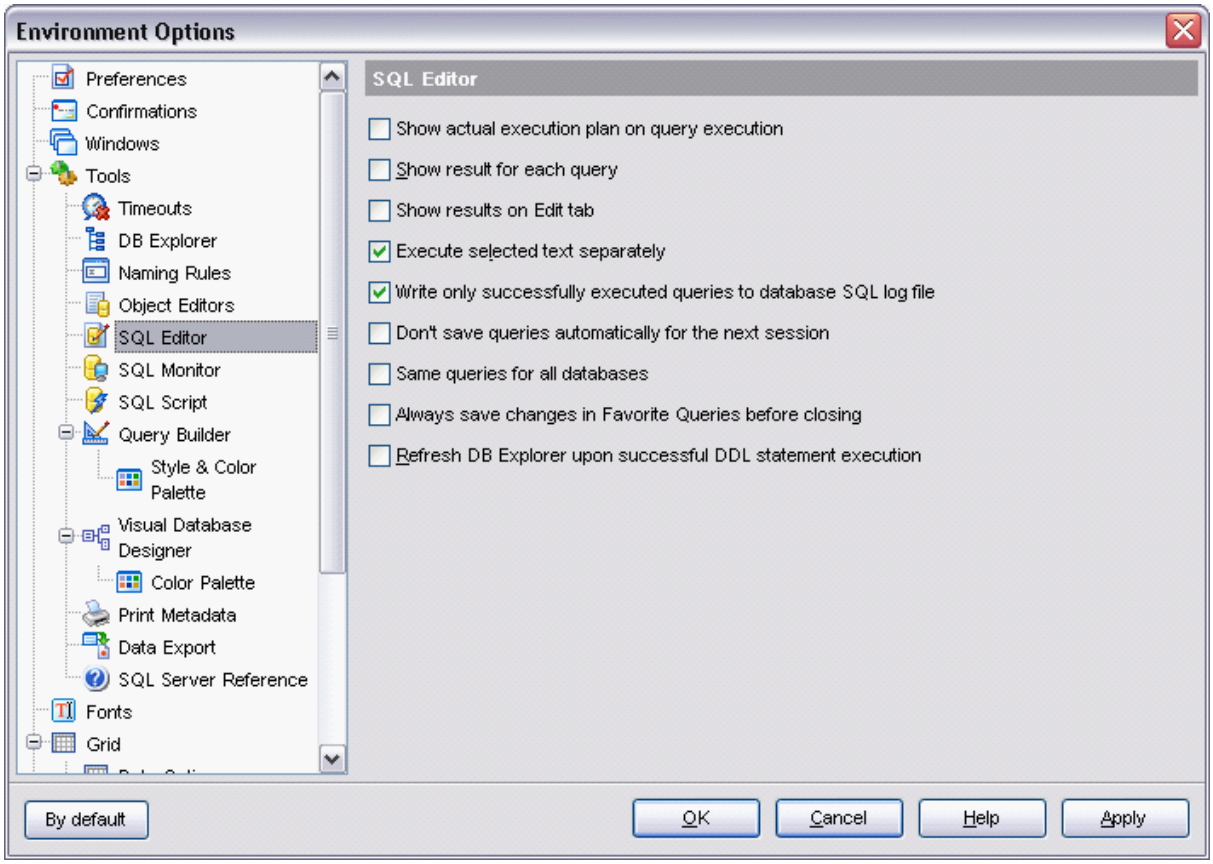
Show result for each query
With this option checked, when you execute two or more queries (divided by "GO"), the result of each query will be displayed. Otherwise, only the result of the last query will be displayed.

© 2008 EMS Database Management Solutions

Show results on Edit tab
If this option is checked, the **Results** tab is displayed as a separate tab.

Execute selected text separately
Check this option to allow execution of the selected statement separately.

Write only successfully executed queries to database SQL log file
If this option is checked, unsuccessful queries will not be saved to the SQL Editor log file (see Setting log options in the Database Registration Info dialog).



Don't save queries automatically for the next session
If this option is checked, the SQL query text will not be saved. Otherwise, it will be saved in Windows registry and will be therefore available in the next application sessions.

Same queries for all database
With this option enabled SQL Editor stores all queries in a shared repository so that switching to another database does not cause loading that database queries (applying that option does not affect the currently opened copies of the SQL Editor). The values of the option can be changed freely without any risk to lose the query repository content.

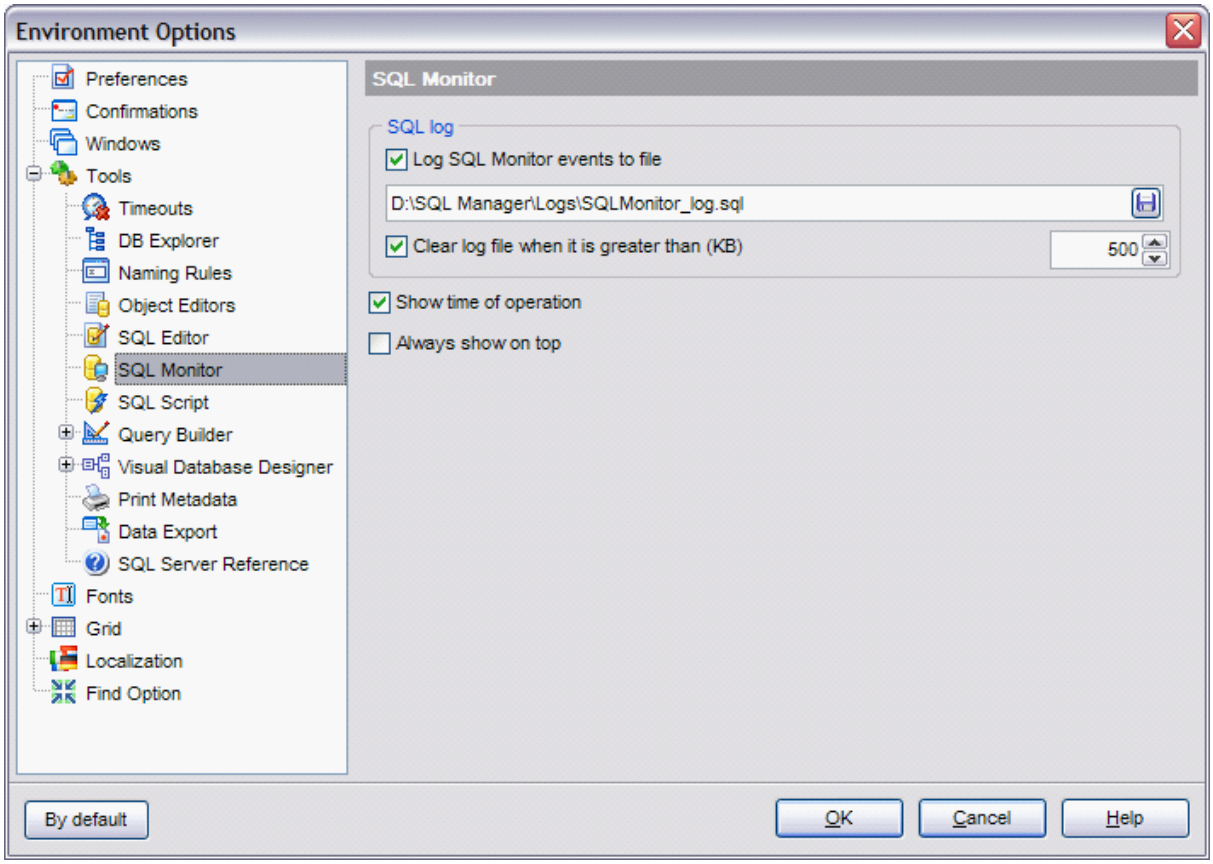
Always save changes in Favorite Queries before closing
This option enables/disables saving changes in SQL queries marked as Favorite automatically upon closing the editor.

© 2008 EMS Database Management Solutions

Refresh DB Explorer upon successful DDL statement execution
If this option is selected, the content of DB Explorer is refreshed each time a DDL statement is executed successfully in SQL Editor.

11.1.1.5.6 SQL Monitor

SQL log
This group of options allows you to enable logging of all SQL Monitor events to a file. Check the **Log SQL Monitor events to file** option, specify the path to the log file using the button, and enter a name for the **.sql* file. To clear the log file after it reaches some definite size, check the **Clear log file when it is greater than...** option and set the maximum file size (in Kilobytes).



Show time of operation
If this option is checked, the execution time of logged operations is added.

Always show on top
Select this option if you want to display the SQL Monitor window in the foreground permanently.

© 2008 EMS Database Management Solutions

11.1.1.5.7 SQL Script

Abort script on error
If this option is checked, script execution is aborted if an error occurs.

Rollback on abort
This option is only available if the **Abort script on error** option is checked. This option evokes automatic rollback when script execution is aborted.

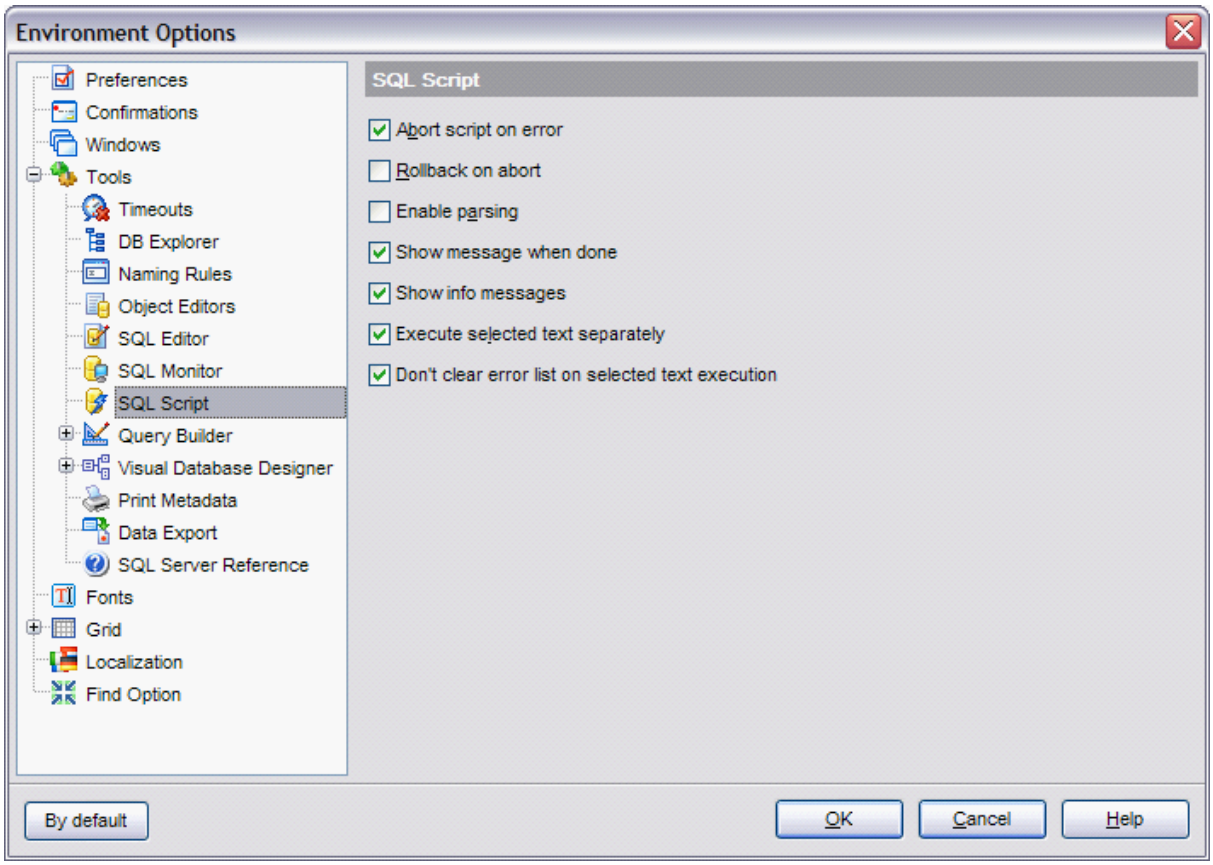
Enable parsing
With this option checked, SQL Script Editor parses the loaded script to enable fast navigation in the Script Explorer tool.

Show message when done
Displays a message box on finishing script execution.

Show info messages
Select this option to enable information messages when working with scripts.

Execute selected text separately
Check this option to allow execution of the selected statement separately.

Don't clear error list on selected text execution
If this option is checked, the error list is not cleared upon execution of the selected statement.



© 2008 EMS Database Management Solutions

11.1.1.5.8 Query Builder

General options

Allow SELECT queries only
When this option is checked, the *INSERT*, *UPDATE* and *DELETE* statements are not allowed in Query Builder.

Select condition row
Displays the selected condition in different rows on the **Criteria** and **Grouping Criteria** tabs of Query Builder.

Drag field name
Displays the dragged field name in the **Builder** area.

Hide selection when inactive
Hides the selection when the Query Builder window is inactive.

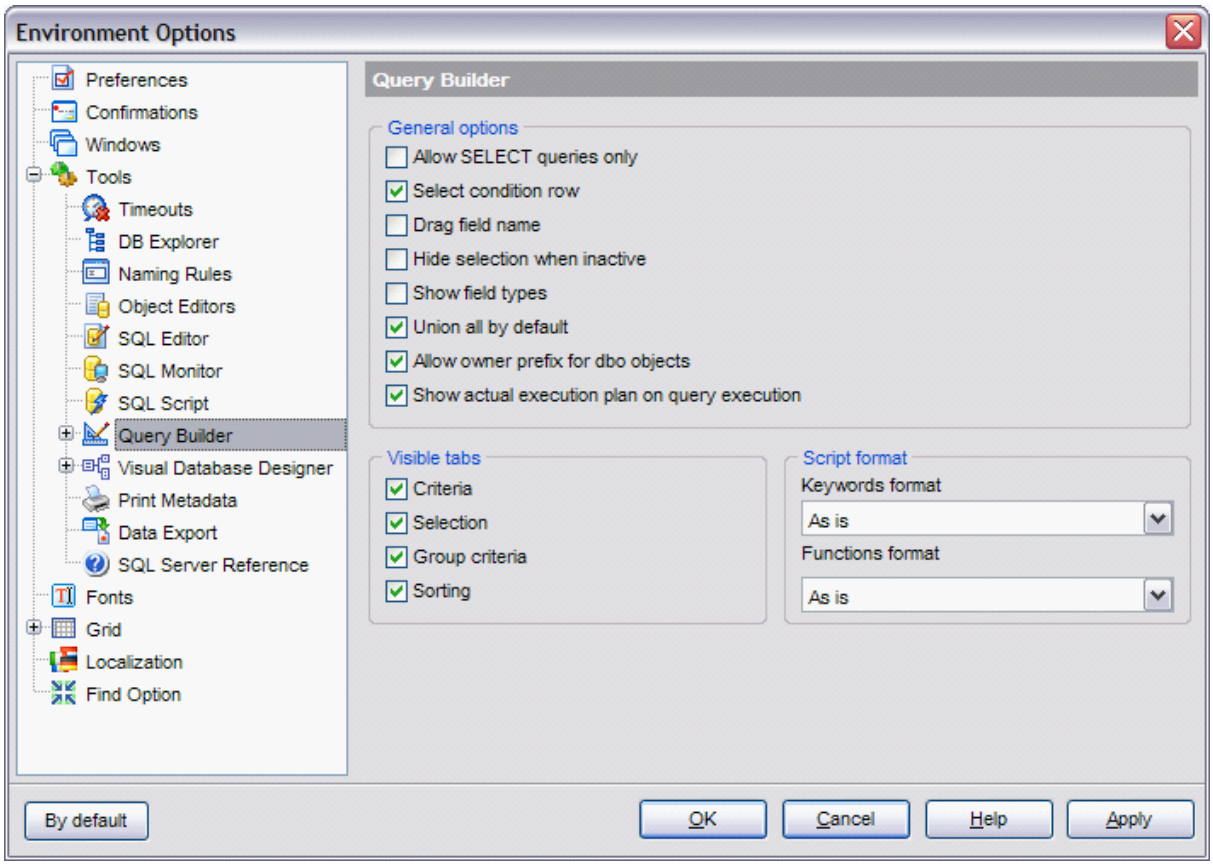
Show field types
Displays the field type next to the field name in the Table box.

Union all by default
Check this option to use the *UNION ALL* expression in Query Builder by default.
The *UNION* keyword allows you to include the results of two *SELECT* statements in one resulting table.
The *ALL* parameter incorporates all rows into the results, including duplicates. If not specified, duplicate rows are removed.

Allow owner prefix for dbo objects
Adds the schema name to the table name in the 'Table' box caption (for *[dbo]* schema objects)

Show actual execution plan on query execution
If this option is checked, the query plan is displayed within the corresponding tab of the Query Builder window.

© 2008 EMS Database Management Solutions



Visible tabs
These options specify which Query Builder tabs are available and which are not. Use the check boxes to make the corresponding tabs visible/invisible.

Script format
These options specify case formatting of keywords and functions in query text within the Edit tab: *As is* keeps the original case, *Uppercase* sets all the keywords/functions to the upper case, *Lowercase* sets all the keywords/functions to the lower case, and *First upper* sets the first letters of all keywords/functions to the upper case.

Additionally, you can set styles and color for all **Query Builder** objects by using Style and Color Palette.

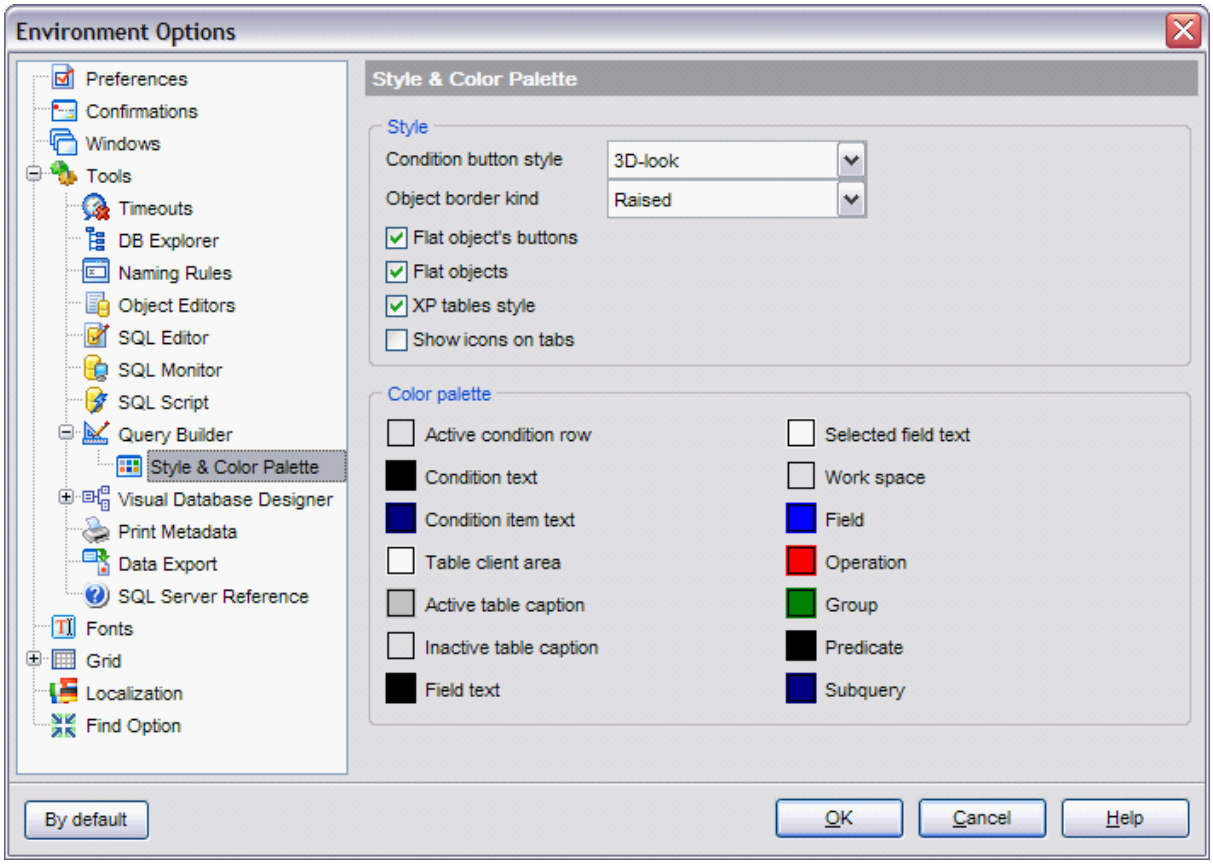
11.1.1.5.8.1 Style & Color Palette

Style
These options specify the way various Query Builder elements look: the **Condition button** : *Flat*, *3DLook*, *Raised*; **object borders**: *Bump*, *Etched*, *Raised*, *Sunken*. If necessary, you can also specify **flatness** for objects and buttons using the corresponding options.

☑ **XP tables style**
This option determines the appearance of non-client areas of tables in Query Builder.

© 2008 EMS Database Management Solutions

Show icons on tabs
With this option selected, you can see icons next to the tab names in Query Builder.



Color palette
These options define the colors of various Query Builder elements: *active condition row*, *condition text*, *condition item text*, *table client area*, etc. Click an item to select a color for the corresponding element using the **Color** dialog where you can specify the required color from the palette.

11.1.1.5.9 Visual Database Designer

Model notation
When you work in Visual Database Designer, you can choose one of the following modeling notations:

- Integration DEFinition for Information Modeling (IDEF1X);
- Information Engineering (IE).

The *IDEF1X* and *IE* notations use different symbols to represent relationships between entities (and tables).

Draw PRIMARY KEY fields separately
Separates Primary key fields from other fields with a horizontal line.

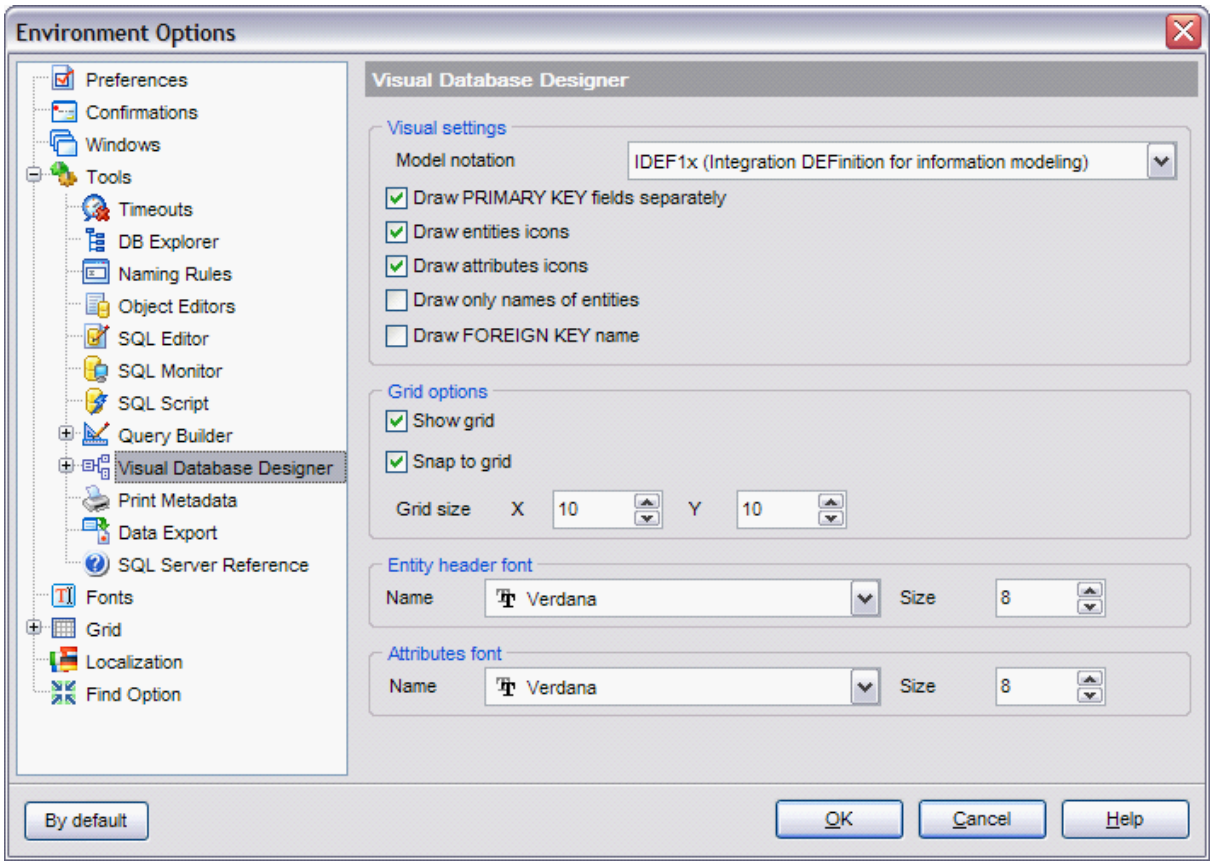
© 2008 EMS Database Management Solutions

Draw entities icons
Displays icons at the left of each entity header according to its type.

Draw attributes icons
Displays icons at the left of each attribute according to its type (Primary key, Foreign key, ordinary field).

Draw only names of entities
Enables highlight for entity names only.

Draw FOREIGN KEY fields separately
Enables highlight for Foreign key fields.



Show grid
Displays dots in the diagram area to make the grid visible.

Snap to grid
Automatically aligns entities on the form with the nearest grid line. You cannot place an entity in between grid lines.

Grid size
Sets grid spacing in pixels along the *x*- and *y*-axes. Specify a higher number to increase grid spacing.

© 2008 EMS Database Management Solutions

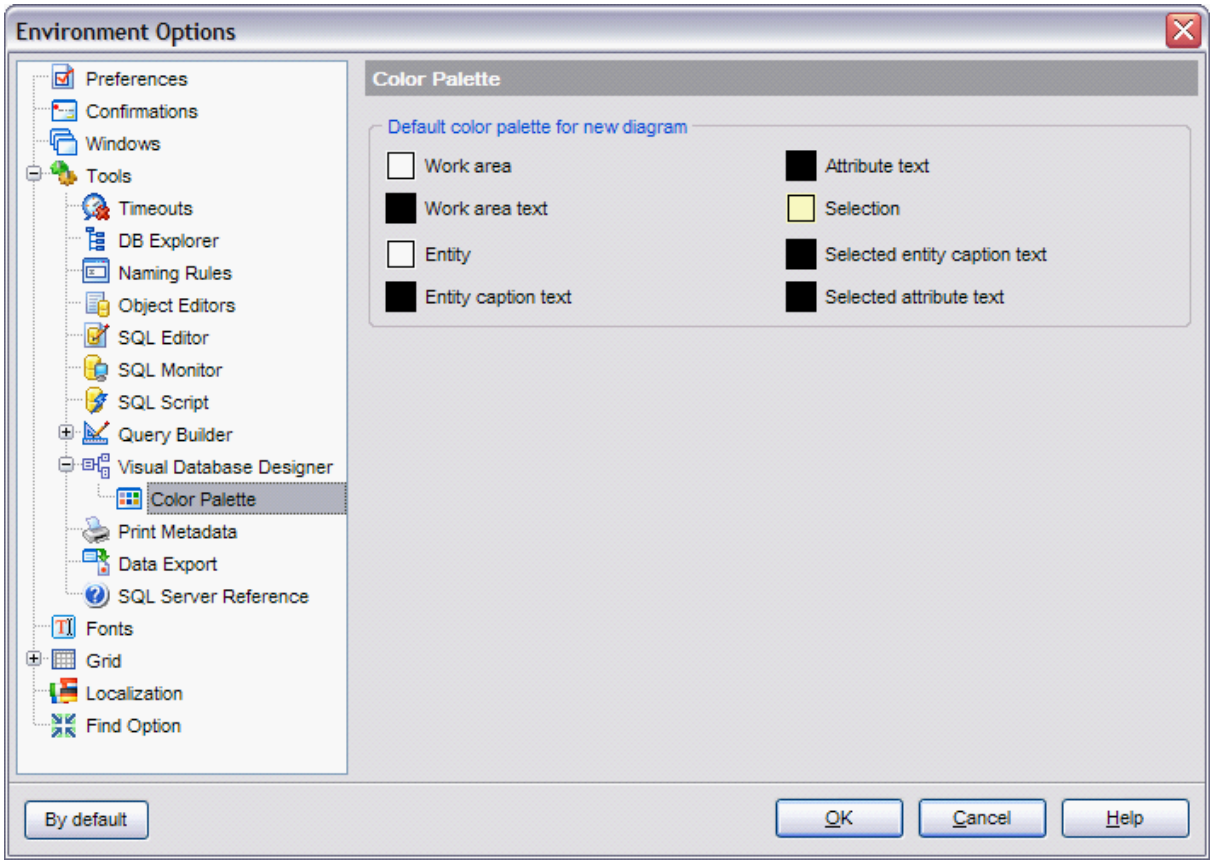
Entity header font
Select a font type and size to display entity headers.

Attributes font
Select a font type and size to display attributes.

Additionally, you can set color for all VDBD diagram objects using Color Palette.

11.1.1.5.9.1 Color Palette

These options define the colors of various Visual Database Designer elements: *work area*, *work area text*, *entity*, *entity caption text*, *attribute text*, etc. Click an item to select a color for the corresponding element using the **Color** dialog where you can specify the required color from the palette.

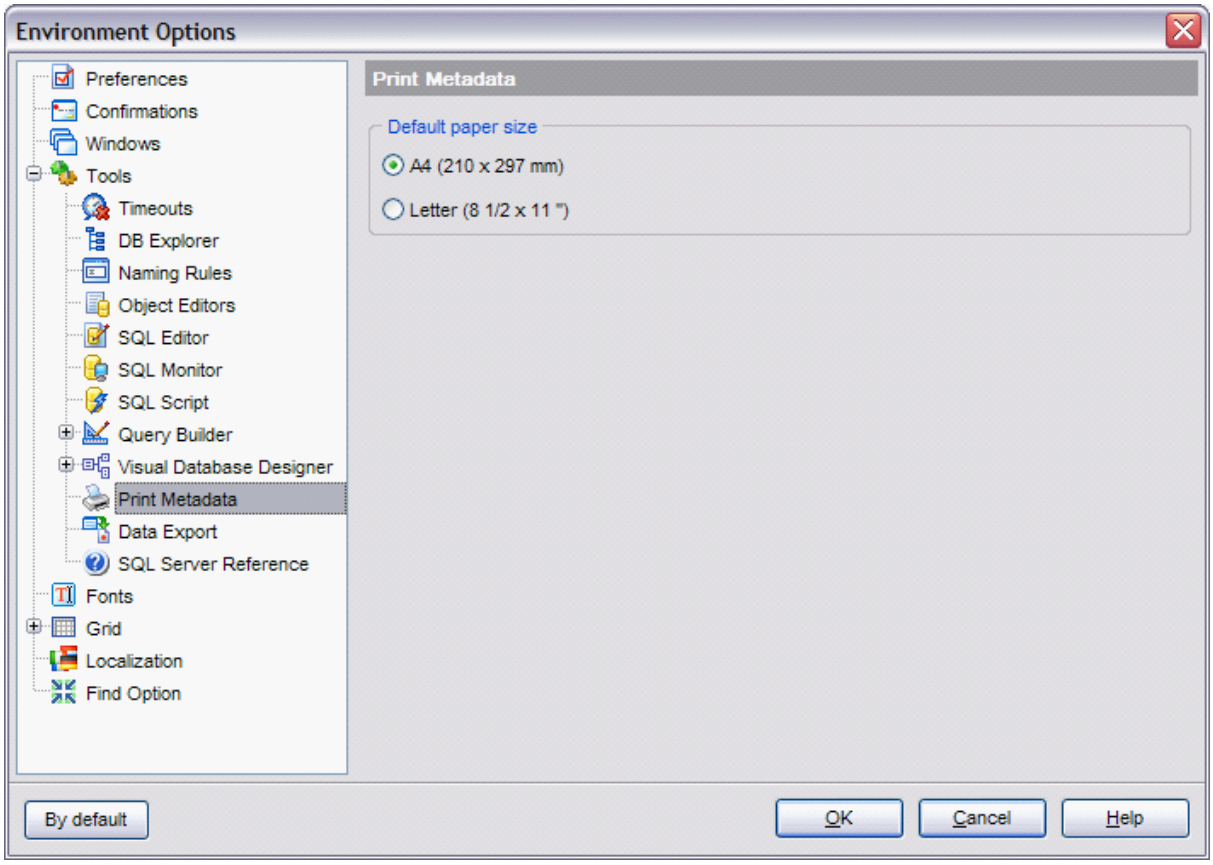


11.1.1.5.10 Print Metadata

Default paper size

© 2008 EMS Database Management Solutions

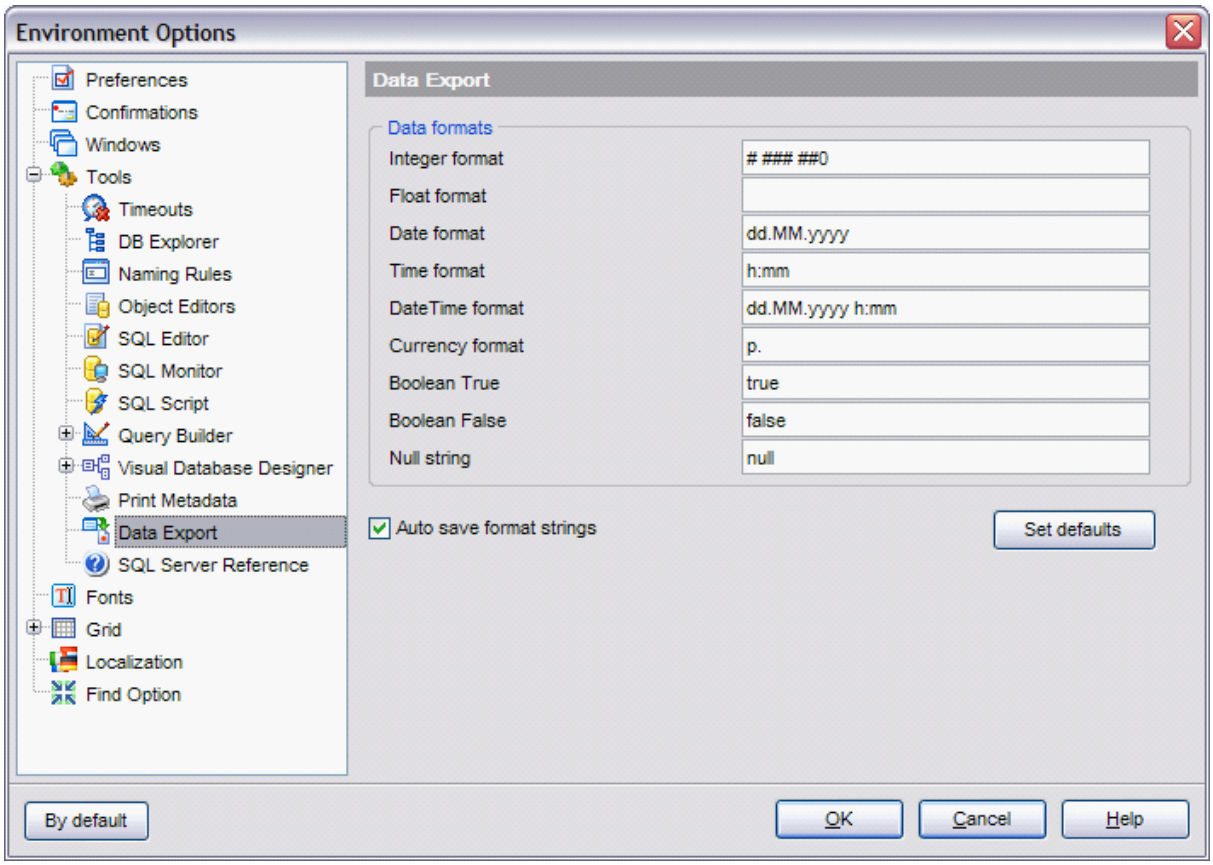
Define the default paper size for reports created with the Print Metadata tool used:
- *A4* (210 x 297 mm)
- *Letter* (8 1/2 x 11 ")



11.1.1.5.11 Data Export

This page allows you to customize formats applied to exported data. Edit the format masks to adjust the result format in the way you need, or click the **Set defaults** button if you wish to apply default data formats.

© 2008 EMS Database Management Solutions



Fore more details see the Adjusting Data Formats page.

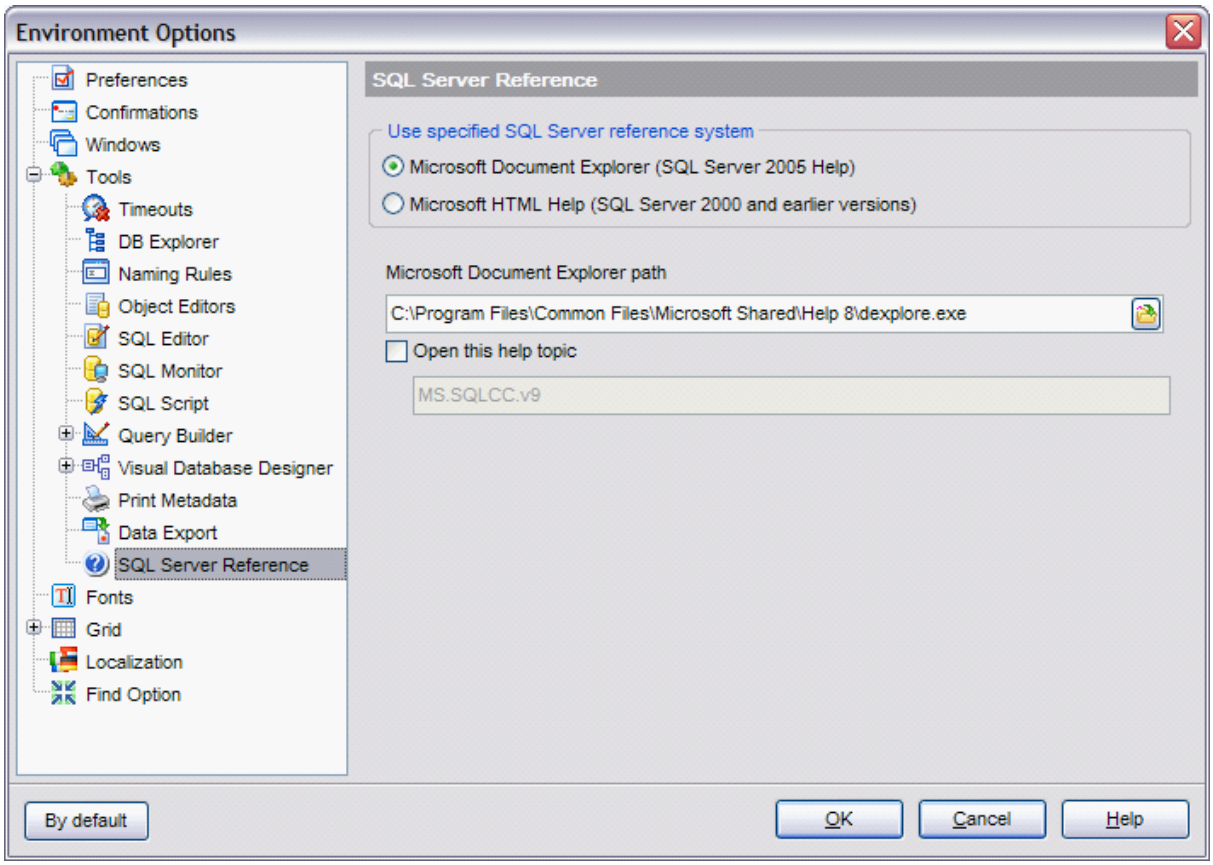
11.1.1.5.12 SQL Server Reference

Using this page you can set which Microsoft® SQL Server™ reference system will be further used in SQL Manager 2008 for SQL Server:

- *Microsoft Document Explorer (SQL Server 2005 Help)*
- *Microsoft HTML Help (SQL Server 2000 and earlier versions)*

For SQL Server 2005 documentation you should also specify the location of **Microsoft Document Explorer**: use the button to browse for the file on your hard disk.

© 2008 EMS Database Management Solutions



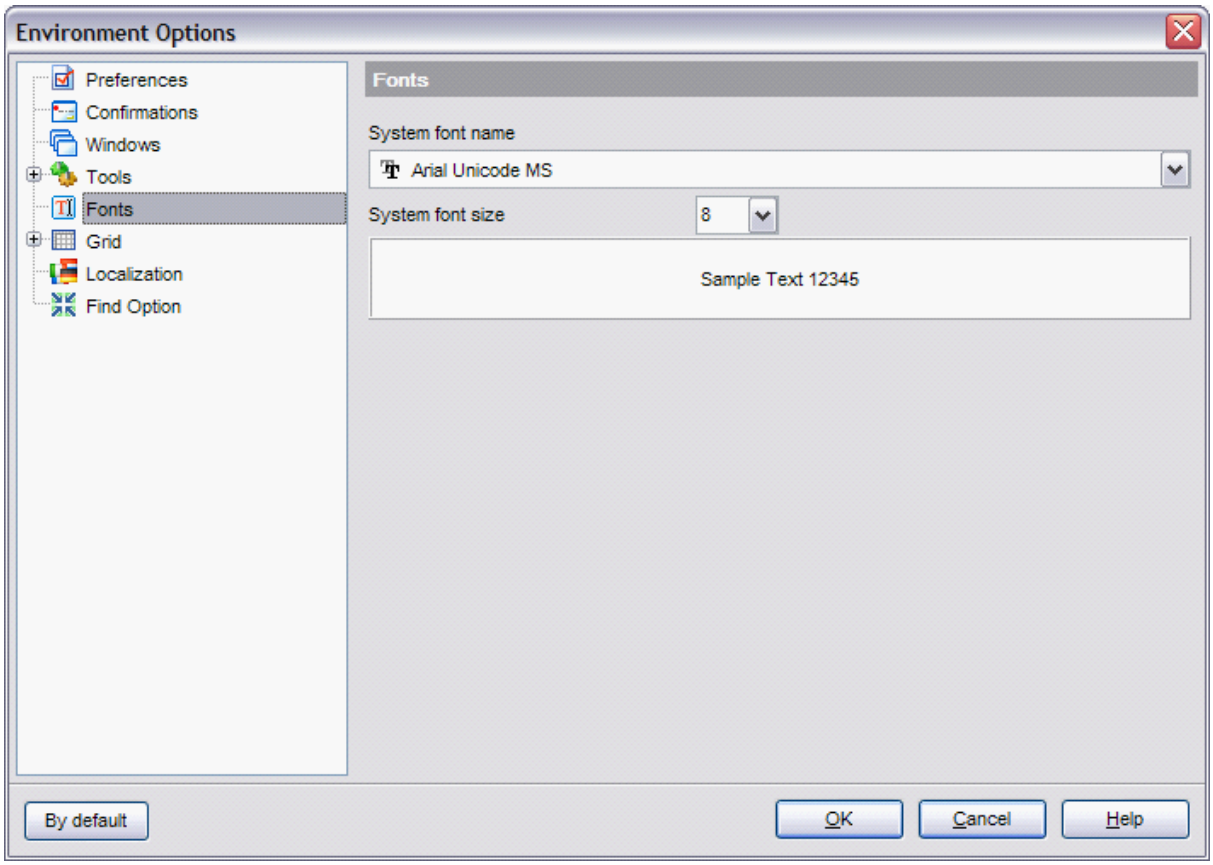
Optionally you can set the help topic to be opened by default at the help system startup.

11.1.1.6 Fonts

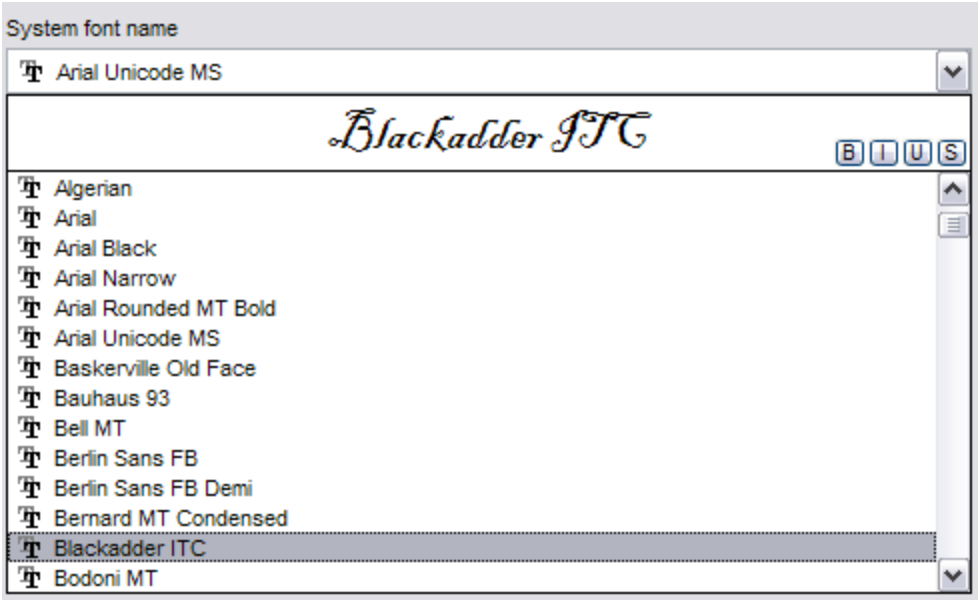
This section of the **Environment Options** dialog allows you to specify fonts used in the application.

The box below displays the *sample text* with the selected font applied.

© 2008 EMS Database Management Solutions



System font name
Defines the font used by SQL Manager 2008 for SQL Server. Select the font name from the drop-down list of available system fonts.



© 2008 EMS Database Management Solutions

System font size
Defines the font size used by SQL Manager 2008 for SQL Server. Type in or use the drop-down list to select the required value.

See also:
Preferences
Full mode activation
Confirmations
Windows
Tools
Grid
Localization
Find Option

11.1.1.7 Grid

General options

☑ **Striped grids**
Displays the odd grid rows in a different color defined by the **Strip** option available on the Color & Formats page.

☑ **Show editor immediately**
Allows editing the cell value right after the cell is clicked.

☑ **Always show editor**
Set this option to make the cell editors always active.

☑ **Enable auto-search in grid**
If this option is checked, the cursor is automatically forwarded to the closest match when you start typing.

☑ **Row multi-selection**
With this option set, multiple rows can be selected in grid.

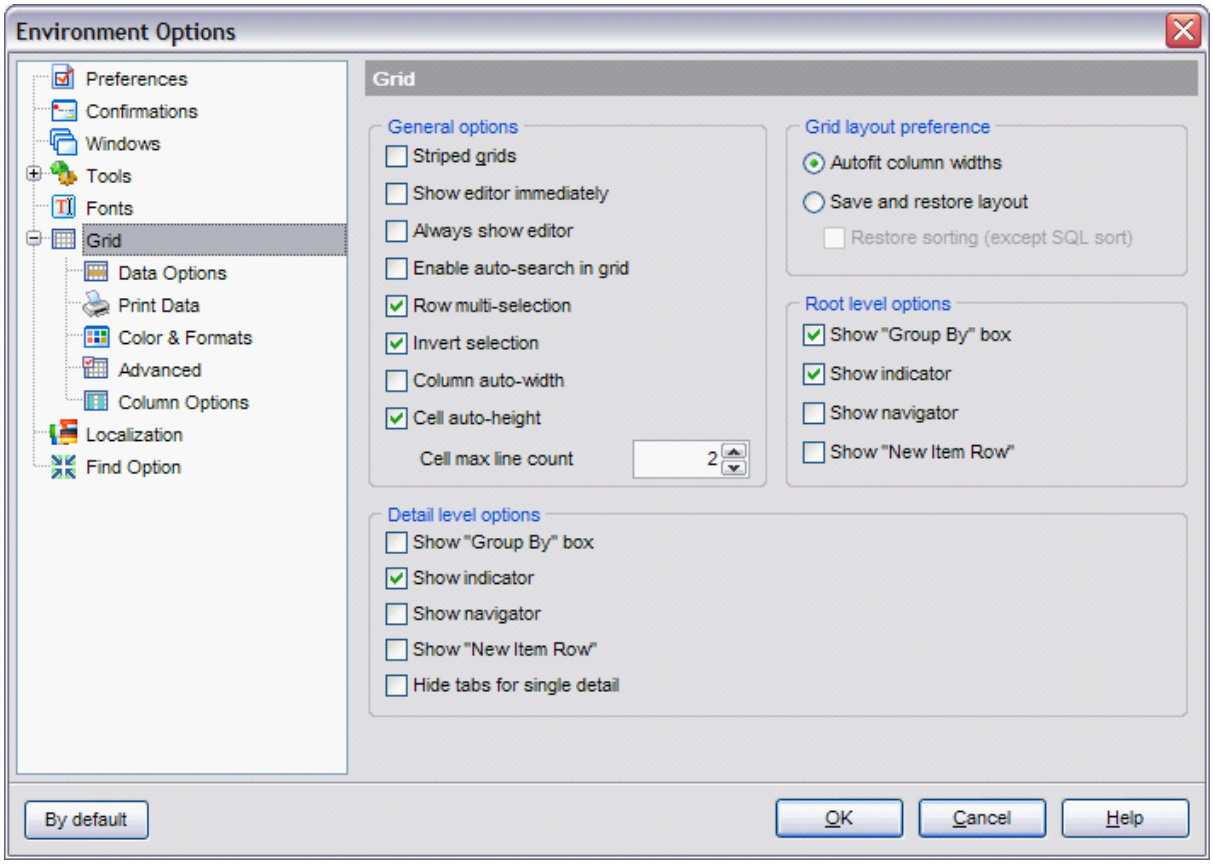
☑ **Invert selection**
Determines whether a single cell within the focused row or the entire row is highlighted when focused.

☑ **Column auto-width**
With this option set, column widths are changed in order to display all columns without using the horizontal scroll bar. If the content a column is still too large to display without a need to resize the grid, then the column values are truncated and the hidden characters are replaced with an ellipsis at the end.

☑ **Cell auto-height**

© 2008 EMS Database Management Solutions

If the widths of the columns are insufficient to display the full content, then text clipping occurs. Set this option to prevent this. If this option is set, the cell content is displayed in multiple lines where necessary. You can set the number of lines to display using the **Cell max line count** option.



Grid layout preference

Autofit column widths
Use this option to shrink the grid columns so that the longest visible column value fits.

Save and restore layout
Use this option to keep the original grid width. Check the **Restore sorting** option to apply defaults to sorting (except for SQL sorting) as well.

Root level options

These options are applied to the main view of the grid. See Grid View for details.

Detail level options

These options are applied to the detail view of the grid. See Grid View for details.

Show "Group by" box
Displays the gray area above the column caption allowing to group data in the grid.

© 2008 EMS Database Management Solutions

Show indicator
Displays the row indicator.

Show navigator
Activates/deactivates the data navigator. The navigator pane is available at the top within the **Data** tab of a view.

Show "New item row"
Displays an empty row at the bottom of a view which is a convenient way for adding data to the grid.

Hide tabs for single detail
This option is useful when only one view is present on the detail level. When checked, the view tab is hidden.

See also:
Preferences
Full mode activation
Confirmations
Windows
Tools
Fonts
Localization
Find Option

11.1.1.7.1 Data Options

Limit options in table and view editors

Define the number of records to be selected on opening the **Data** tab of Table Editor and View Editor:
Select all records from a table
Select only ... records (you should set the number of records using the corresponding spin editor)

Advanced

Use SQL sorting in data view
If the option is checked, the data will be sorted according to SQL sorting parameters.

Use SQL filter in data view of Table and View Editors
Check/uncheck this option to enable/disable SQL filter in data view of Table Editor and View Editor.

With the **Use SQL sorting in data view** option enabled, a click on the column header for

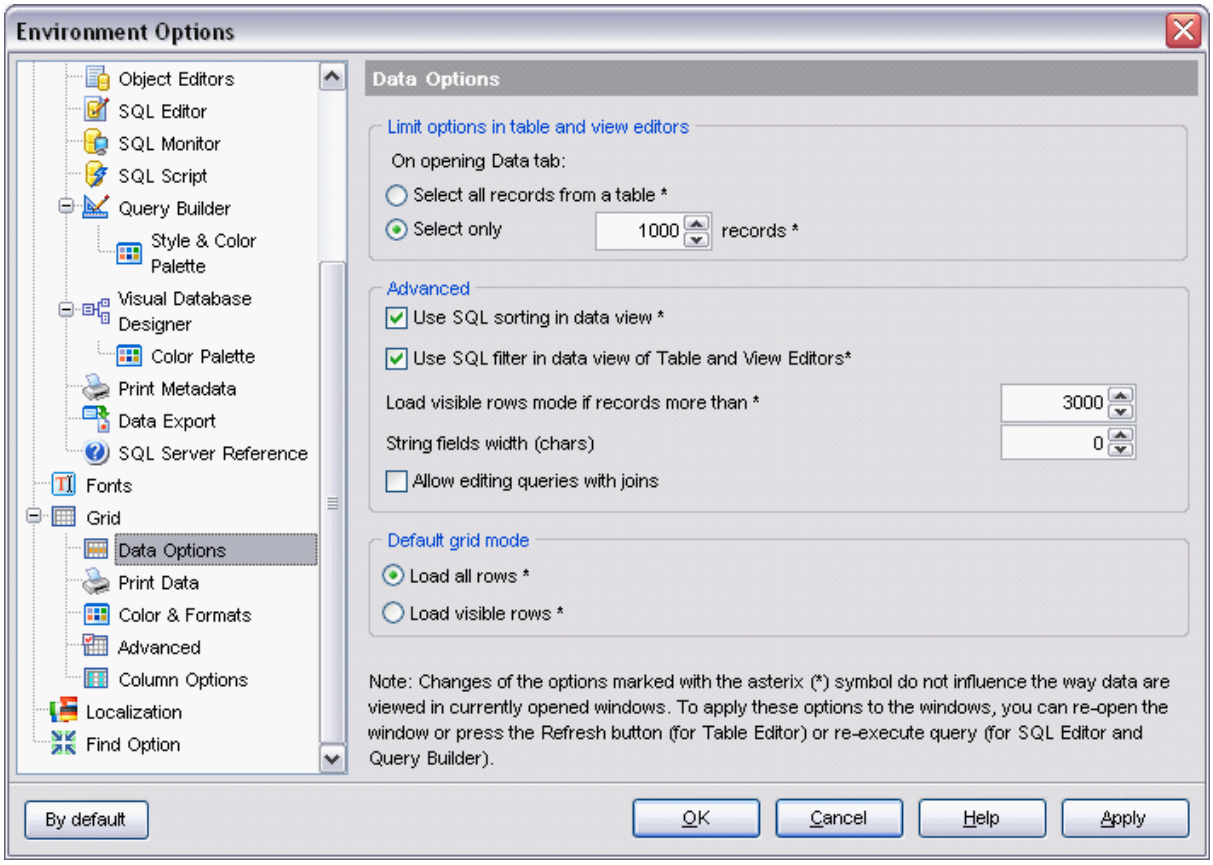
© 2008 EMS Database Management Solutions

sorting causes reloading all table data with the selected field in ORDER BY expression of the SELECT statement. Otherwise the sorting is performed by means of the grid. If the table contains a huge number of records and the **Select only XXX records** mode (see the **Limit options in table and view editors** group) is used, this mode is more preferable (e.g. all the records having values starting with "A" will be displayed, and not those which were in originally opened XXX records).
All above-mentioned is related to the **Use SQL filter in data view of Table and View Editors** option. If the filter is applied to a table containing a great number of records, it is strongly recommended to enable this option - in this case the filter will be applied to all table/view records, not only to those which are displayed at the present moment.

Load visible rows mode if records more than...
Set this option to switch to the **Load visible rows** mode when the number of records in the dataset exceeds the specified value.

String fields width (chars)
Using this option you can limit string fields width that may improve performance on large datasets.

Allow editing queries with joins
Enable this option to make editing queries with joins possible.



Default grid mode

© 2008 EMS Database Management Solutions

Load all rows
The grid loads all records from a dataset. This option increases the grid performance by reloading only changed dataset records when updating. In this mode all features (automatic sorting, filtering and summary calculations) are available.

Load visible rows
The grid loads only a fixed number of dataset records into memory. This option minimizes dataset loading time. Automatic sorting, filtering, summary calculations are not available in this mode.

The **Default grid mode** options allow you to define the grid mode which will be used by default.
With the **Load all rows** option enabled, when loading data, all the records are loaded into grid buffers. In this mode opening the tables with many records may take a considerable amount of time. But in this case you can make use of some advantages: in the filter drop-down list the column headers are displayed with the values for quick filtering; it is possible to open several sub-levels at the same time when viewing data in master-detail view, etc.
In case opening and other operations with an object consisting of many records takes sufficient time, the **Load visible rows** mode should be used instead. It can be set individually for each table and saved between sessions (can be set via the context menu of the grid).

See also:
EMS SQL Manager FAQ

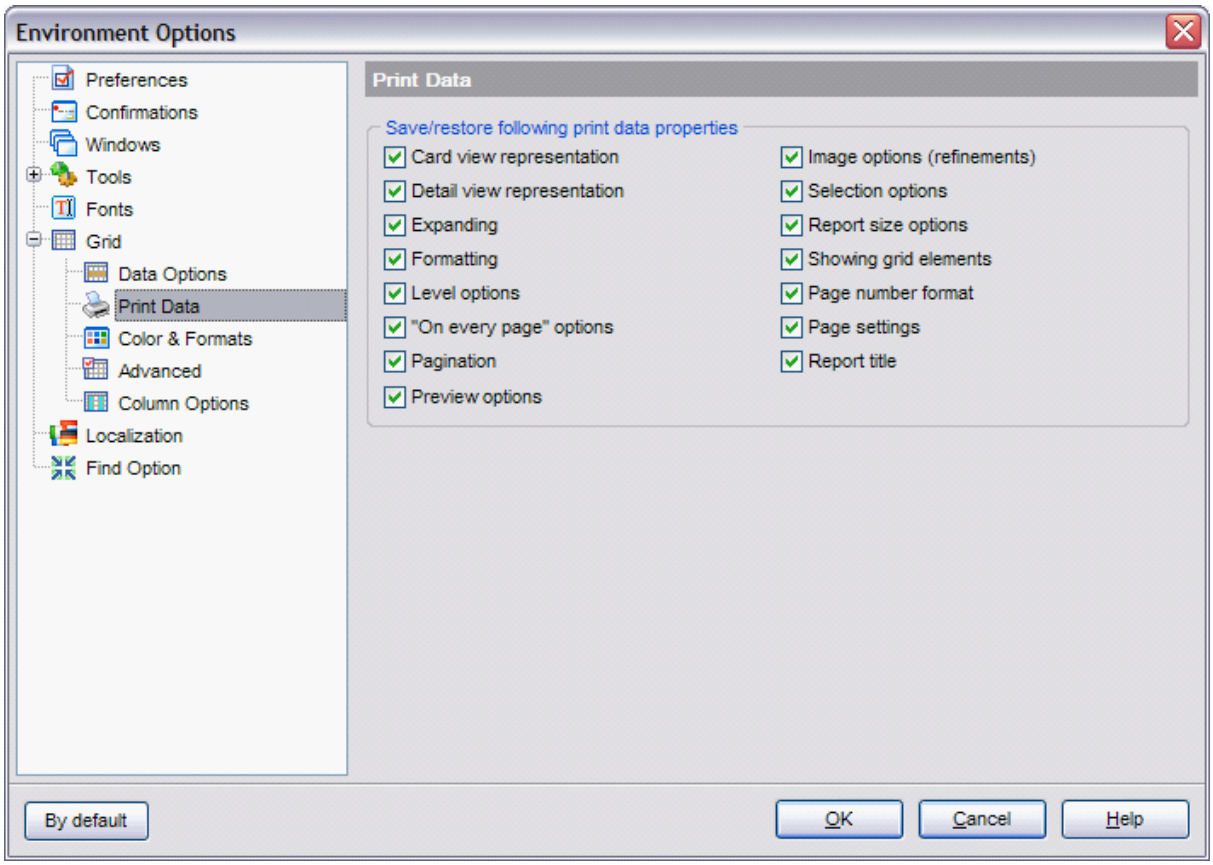
11.1.1.7.2 Print Data

Save/restore following print data properties

These options specify which Print Data properties will be saved between work sessions (e.g. if you tick off the *Page settings* item, those settings will be saved and stored between the sessions).

You can save/restore the following **Print Data properties**: *Card view representation*, *Detail view representation*, *Expanding*, *Formatting*, *Level options*, *"On every page" options*, *Pagination*, *Preview options*, *Image options*, *Selection options*, *Report size options*, *Showing grid elements*, *Page number format*, *Page settings*, *Report title*.

© 2008 EMS Database Management Solutions



11.1.1.7.3 Color & Formats

Display formats

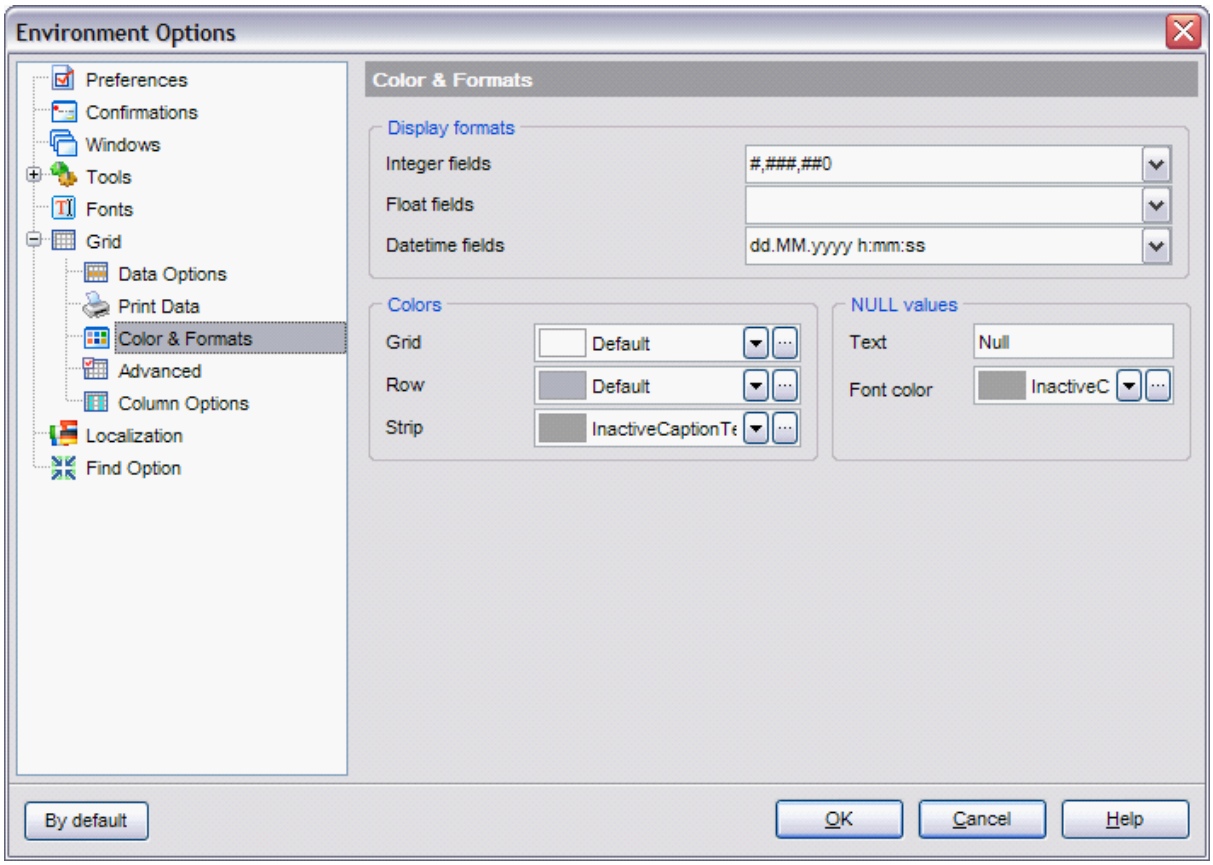
Integer fields
Defines the format for displaying *TINYINT*, *SMALLINT*, *INTEGER* and *BIGINT* fields.

Float fields
Defines the format for displaying *FLOAT*, *DOUBLE* and *DECIMAL* fields.

Datetime fields
Defines the format for displaying *DATETIME* fields.

For more information refer to the Format specifiers page.

© 2008 EMS Database Management Solutions



Colors
Options of this group allow you to set colors for basic grid elements. Use the ellipsis button to open the **Color** dialog allowing you to select the required color from the palette.

Grid
Defines the background color of the data grid.

Row
Defines the color of the selected row in the data grid.

Strip
Defines the color of the odd rows (applied if the **Striped grids** option is set on the Grid page).

NULL values

Text
Defines the text that stands for NULL values in grid.

Font color
Defines the font color for displaying NULL values in the grid. Use the ellipsis button to open the **Color** dialog allowing you to select the required color from the palette.

© 2008 EMS Database Management Solutions

11.1.1.7.4 Advanced

Advanced options

Cell hints for clipped text
Indicates whether a hint box is displayed when hovering over a cell containing clipped text.

Focus cell on cycle
Determines whether the focus moves to the next row after it reaches the right-most cell within the current row.

Focus first cell on new record
Determines whether the focus moves to the first cell of a newly created row.

Next cell on pressing Enter
Determines whether the current view columns can be navigated by using the **Enter** key.

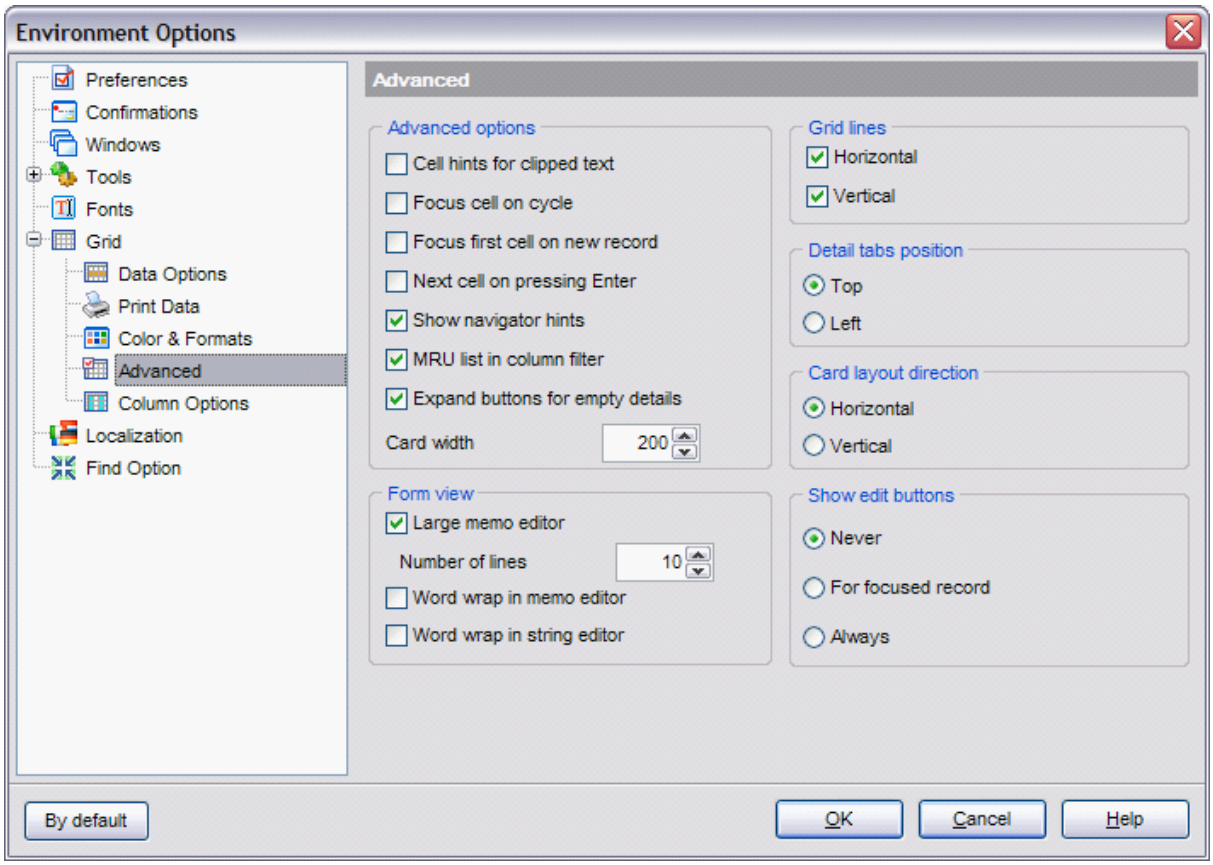
Show navigator hints
Indicates whether a hint box is displayed when hovering over navigation buttons.

MRU list in column filter
Enables showing of *Most Recently Used* items when filtering columns

Expand buttons for empty details
Specifies whether to display expand buttons within master rows that do not have associated details.

Card width
Defines the width of the card used in Card View mode.

© 2008 EMS Database Management Solutions



Form view

Large memo editor
Sets the number of lines for text-typed fields when viewing data in Form view.

Word wrap in memo editor
Determines whether long strings are wrapped within the memo editor area.

Word wrap in string editor
Determines whether long strings are wrapped within the string editor area.

Grid lines
Determines whether to display *vertical* and *horizontal* lines between cells.

Detail tabs position
Specifies the position of the tabs in detail level views: *top* or *left*.

Card layout direction
Specifies the direction of cards in Card View mode: *horizontal* or *vertical*.

Show edit buttons
Indicates when the edit buttons are displayed: *never*, *for focused record* or *always*.

© 2008 EMS Database Management Solutions

11.1.1.7.5 Column Options

Common options

Auto-select text
Determines whether all text within an editor is automatically selected when the editor gets focus.

Hide selection on losing focus
Determines whether the visual indication of the selected text remains when the editor loses focus.

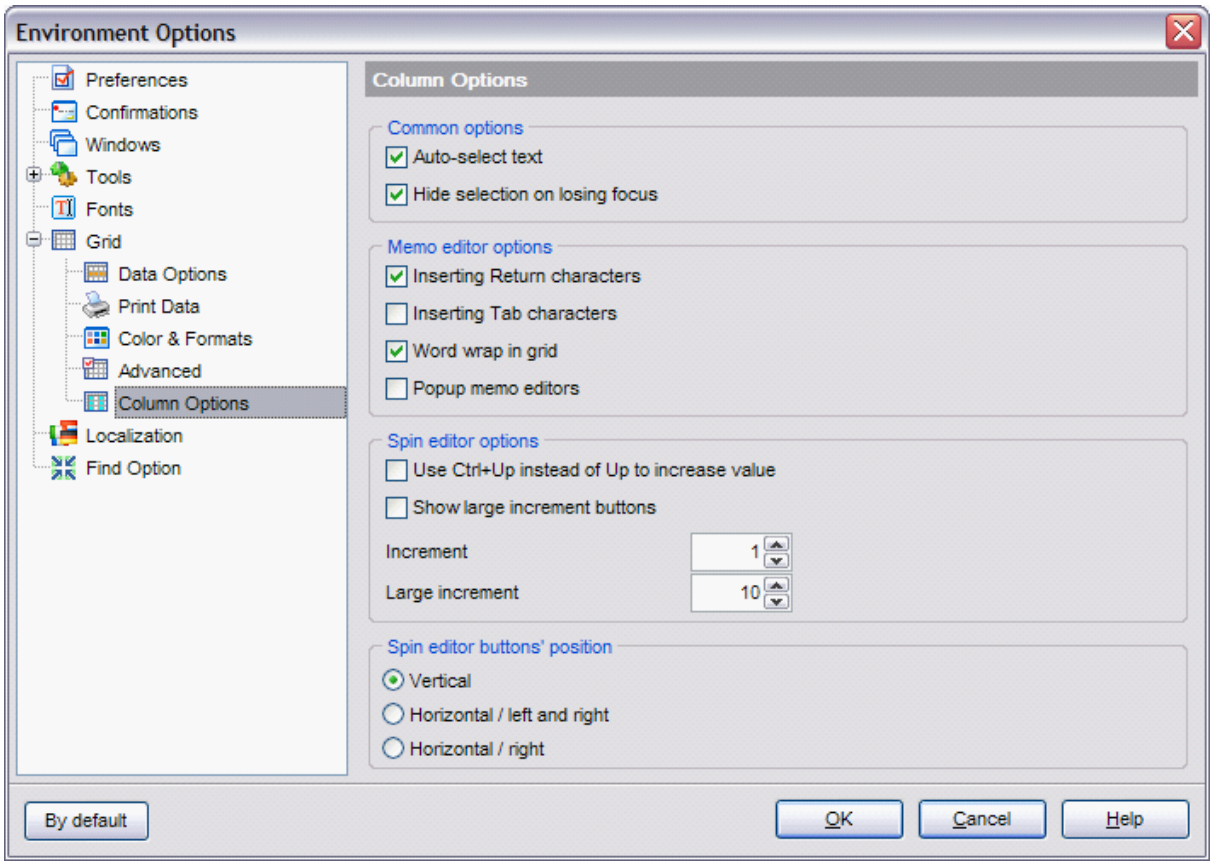
Memo editor options

Inserting Return characters
Specifies whether a user can insert return characters into text.

Inserting Tab characters
Specifies whether a user can insert tab characters into text.

Word wrap in grid
Determines whether long strings are wrapped in grid.

Popup memo editors
Turns on popup memo editors for text BLOB type fields.



© 2008 EMS Database Management Solutions

Spin editor options

☑ **Use Ctrl+Up instead of Up to increase value**
Allows you to use *Ctrl+Up* and *Ctrl+Down* key combinations for editing spin values (for INTEGER field values).

☑ **Show large increment buttons**
Determines whether fast buttons (for large increment) are visible within the editor.

Increment
Specifies the increment value for the spin editor.

Large increment
Specifies the large increment value for the spin editor.

Spin editor buttons' position
Specifies the position of spin editor buttons: *vertical*, *horizontal / left and right* or *horizontal / right*.

11.1.1.8 Localization

The **Localization** section of the **Environment Options** dialog is provided for managing the localization files of SQL Manager 2008 for SQL Server.

You can create your own **.lng* files similar to those available in the *%program_directory%\Languages* folder, add them to the list of available languages and set the new language as the program interface language.

Default directory
Use the **Explorer** button to specify the directory where the **.lng* files are to be stored by default.

Choose program language
Use the drop-down list of available languages to select the interface language to be applied to the application.

☑ **Auto scan languages on startup**
When checked, the directory with localization files will be scanned automatically at the application startup; all the languages found will be added to the list of available languages.

Available Languages
Lists all the languages available for localization and the corresponding **.lng* files. Double-click a language in the list to edit its name or the **.lng* file.

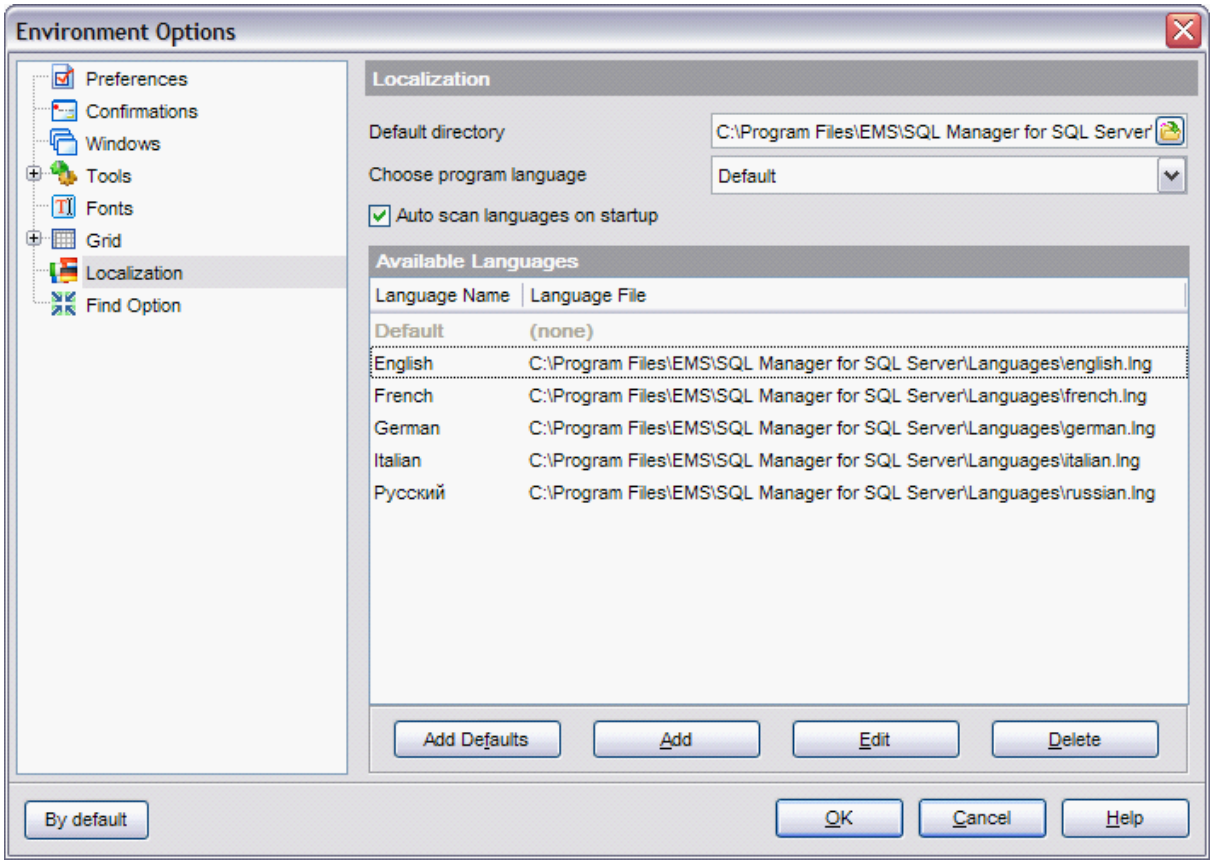
Add Defaults
This button is used to search for **.lng* files in the **Default directory** and add all of them to the **Available Languages** list.

© 2008 EMS Database Management Solutions

Add
Opens the Add language dialog where you can specify your own localization file and set the language name.

Edit
Opens the Edit language dialog where you can change the language name or select another localization file for the specified language.

Delete
Removes the selected language from the **Available languages** list (without confirmation).



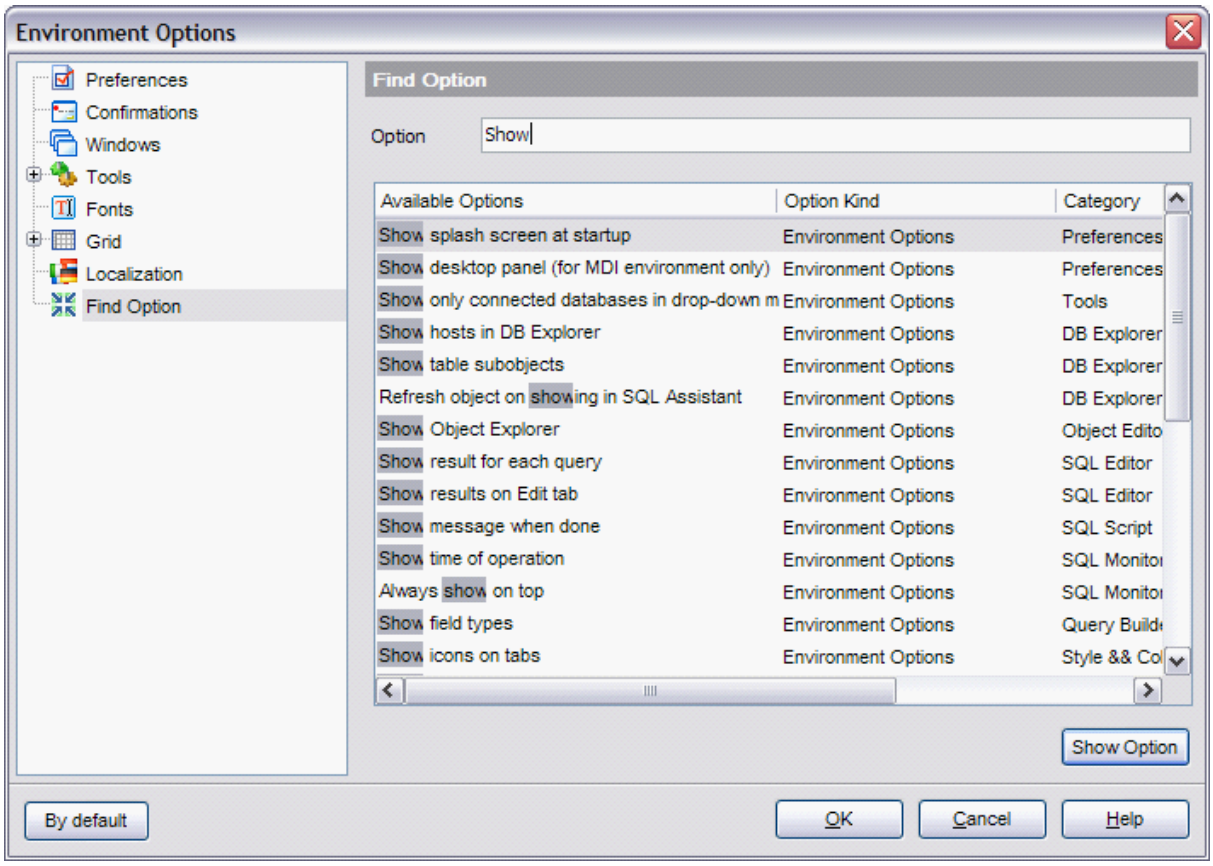
See also:
Preferences
Full mode activation
Confirmations
Windows
Tools
Fonts
Grid
Find Option

© 2008 EMS Database Management Solutions

11.1.1.9 Find Option

The **Find Option** section allows you to search for options available within the **Environment Options** dialog easily and quickly.

Option
In this field you can enter the name of the option to search for within SQL Manager Environment Options.



The **Available options** area lists all options of the Environment Options category according to the specified name. Select the required option in the list and click **Show Option** to open the corresponding section where you can view/edit the value of this option.

See also:
Preferences
Full mode activation
Confirmations

© 2008 EMS Database Management Solutions

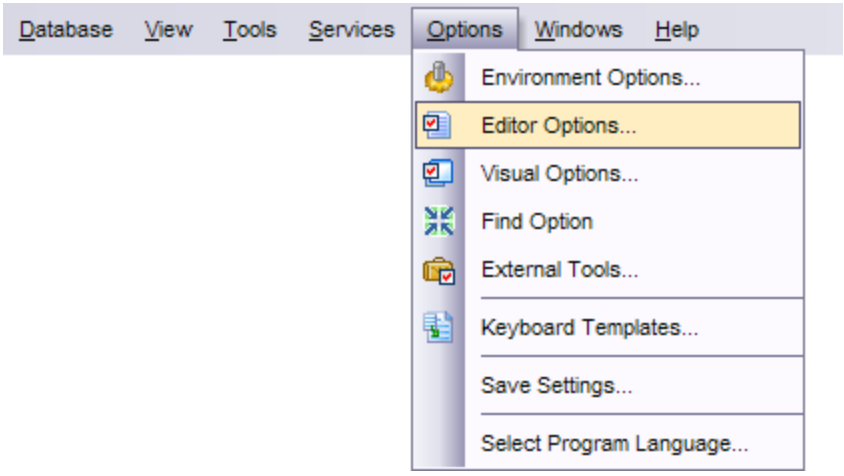
Windows
Tools
Fonts
Grid
Localization

11.1.2 Editor Options

Editor Options allow you to set the parameters of viewing and editing SQL statements within the SQL Editor tool of the SQL Manager application.

To open the **Editor Options** window, select the **Options | Editor Options...** main menu item, or use the **Editor Options** button on the main toolbar.

- General
- Display
- Color
- Quick Code
- SQL Formatter
- Key Mapping
- Spell Checking
- Find Option



Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**Yes**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

© 2008 EMS Database Management Solutions

See also:
Environment Options
Visual Options
Save Settings

11.1.2.1 General

Auto indent
If this option is checked, each new indention is the same as the previous one when editing SQL text.
Indents and outdents are used in the process of text editing to make the source code easier to read.

Insert mode
If this option is checked, the insert symbols mode is on by default.

Find text at cursor
If this option is checked, the **Text to find** field in the Find Text dialog is automatically filled with the text on which the cursor is set.

Always show hyperlinks
If this option is checked, hyperlinks are displayed in the editor window. To open a link, click it with the *Ctrl* key pressed.

Double click line
If this option is checked, double-clicking the line on which the cursor is set selects the whole line.

Trim trailing spaces
If this option is checked, all spaces after the last symbol in line will be trimmed.

Fixed line height
Prevents line height calculation. If this option is checked, the default line height is taken.

Persistent blocks
Keeps marked blocks selected even when the cursor is moved with the arrow keys used, unless a new block is selected.

Fixed column move
If this option is checked, the caret keeps its horizontal position when moved between lines.

Optimal fill
Check this option to enable optimal algorithm of filling text content in the working area of the editor.

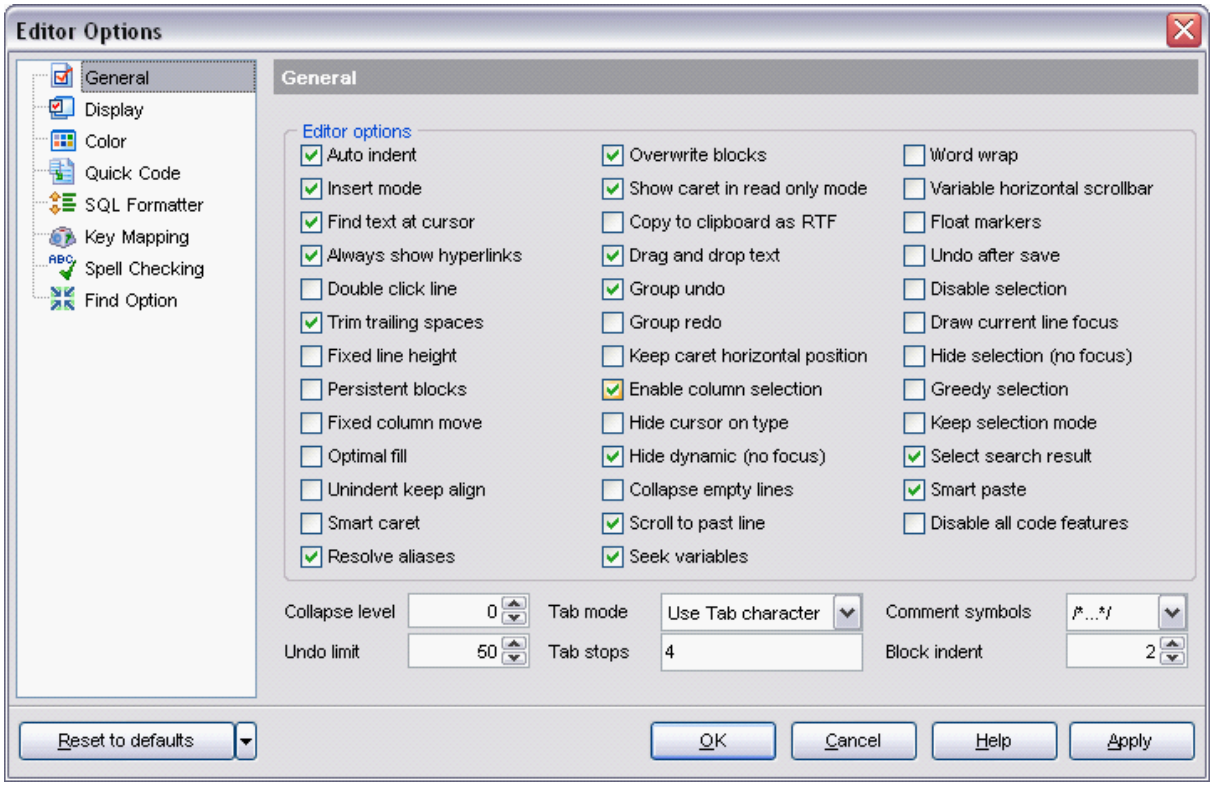
Unindent keep align
Keeps align for the lines that are not indented.

Smart caret
This option determines the caret movement (up, down, line start, line end). The caret is

© 2008 EMS Database Management Solutions

moved to the nearest position on the screen.

Resolve aliases
Disables highlighting and code completion feature for aliases.



Overwrite blocks
Replaces a marked block of text with whatever is typed next. If **Persistent Blocks** is also selected, the text you enter is appended to the currently selected block.

Show caret in read only mode
Displays/hides the caret in read-only mode.

Copy to clipboard as RTF
If this option is checked, the selected text is copied in RTF format.

Drag and drop text
This option allows to drag and drop selected text.

Group undo
This option allows you to undo multiple actions of the same kind.

Group redo
This option allows you to redo multiple actions of the same kind.

Keep caret horizontal position
If this option is checked, the horizontal position of a cursor is kept. If you move the cursor (using the *Up* and *Down* arrow keys) onto a line having length less than the current

© 2008 EMS Database Management Solutions

cursor horizontal position, it will be positioned after the last symbol of the line.

Enable column selection
Enables/disables column selection mode.

Hide cursor on type
Hides/displays mouse cursor within the working area while a user is typing some text.

Hide dynamic (no focus)
Hides dynamic highlights when an editor is not focused.

Collapse empty lines
Collapses empty lines after a text range when this range has been collapsed.

Scroll to past line
When the option is enabled, you can scroll to the last line of the text only, otherwise you can scroll to the end of the page.

Seek variables
Disables code completion feature for variables.

Word wrap
When on, text is wrapped at the right margin of the editor area to fit in the visible area.

Variable horizontal scrollbar
If this option is checked, the horizontal scrollbar varies according to the current content of the editor.

Float markers
When enabled, markers are linked to the text, and they will move with the text while the text is being edited; otherwise the markers are linked to the caret position, and stay unchanged while the text is being edited.

Undo after save
Keeps undo buffer unchanged after saving.

Disable selection
Disables any selection when editing.

Draw current line focus
Draws the focus rectangle around the current line when the editor has focus.

Hide selection (no focus)
Hides the selection when the editor loses focus.

Greedy selection
Selects an extra column/line in column/line selection modes.

Keep selection mode
Enables selection for caret movement commands (like in BRIEF).

Select search result
Determines whether the search result should be selected.

© 2008 EMS Database Management Solutions

Smart paste
When this option is enabled, the editor gets both Unicode and ANSI content from the clipboard, converts them using the selected character set and selects the best text to be pasted. This allows getting correct text copied from both ANSI and Unicode applications disregarding the currently selected keyboard language.

Disable all code features
This option disables code completion, code folding, highlight and all options that are set on the Quick Code page. For options that are set on the Color page, the defaults will be applied.

Collapse level
Specifies the level of text ranges that will be affected by the "Collapse all" command.

Undo limit
Defines the maximum number of changes possible to be undone.

Tab mode
Specifies the way the TAB key is processed. Possible values are: *Use tab character* (inserts a tab character); *Insert spaces* (inserts space characters); *Dialog behaviour* (when the edit control is in a dialog, the focus is switched to the next control); *Smart tab* (tabs to the first non-white space character in the preceding line).

Tab stops
Defines the tab length used when editing a text.

Comment symbols
Defines the symbols which will be used to comment code fragments.

Block indent
Specify the number of spaces to indent a marked block.

See also:
Display
Color
Quick Code
SQL Formatter
Key Mapping
Spell Checking
Find Option

11.1.2.2 Display

Gutter

Show line numbers
If this option is checked, line numbers are displayed in the SQL text editor window.

© 2008 EMS Database Management Solutions

Gutter auto width
Enable this option to specify that the gutter width will be adjusted automatically.

Display line state
If this option is checked, a colored line indicating the state of all altered lines in the text is displayed at the gutter of the editor window.

Use code folding
Check this option to enable to code folding feature of SQL Editor.

Width
Defines the gutter width in the editor window.

Color
Defines the gutter color in the editor window. Select an item from the drop-down list or click the ellipsis button to select a color using the **Color** dialog where you can specify the required color from the palette.

Right margin

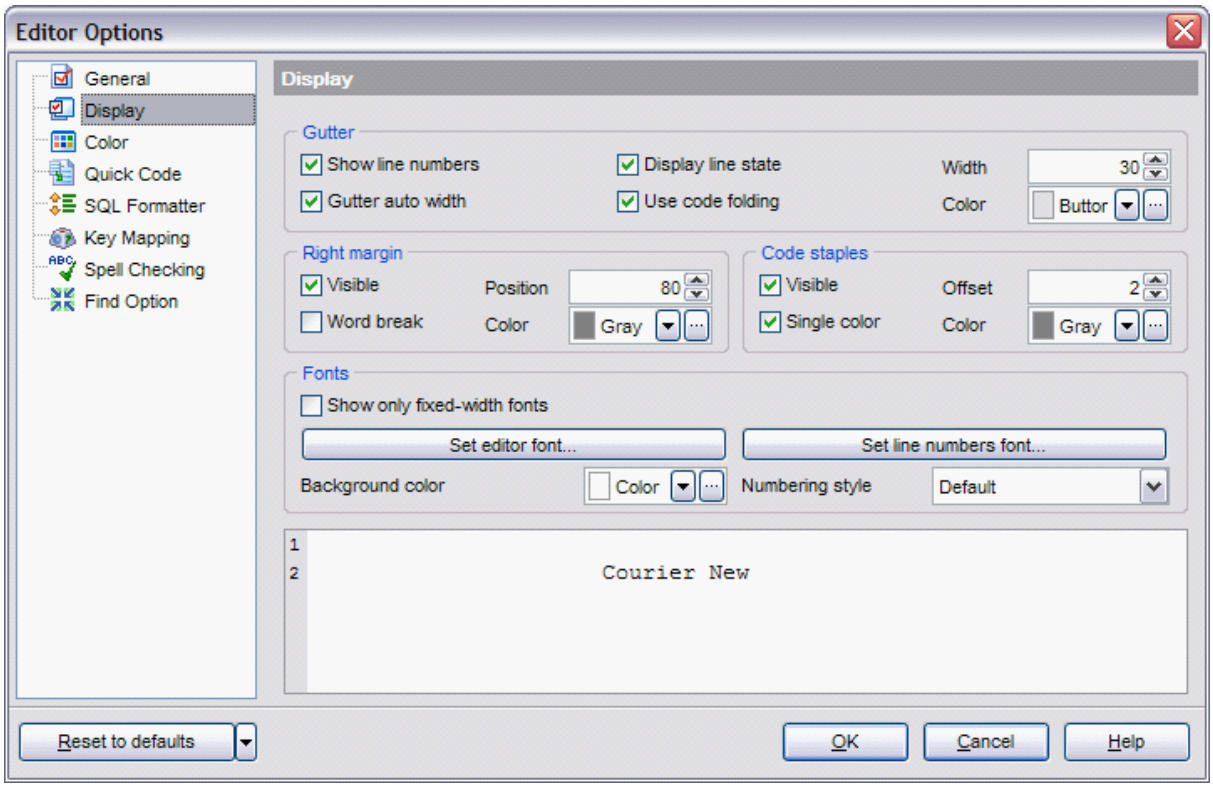
Visible
Makes the right text margin visible.

Word break
Allows breaking the words at the right margin.

Position
Defines the position of the right text margin in the editor window.

Color
Defines the color of the right margin in the editor window. Select an item from the drop-down list or click the ellipsis button to select a color using the **Color** dialog where you can specify the required color from the palette.

© 2008 EMS Database Management Solutions



Code staples

Visible
Makes the code staples visible in the editor window.

Single color
Check the option to apply a single color for code staples.

Offset
Specify the offset value for code staples.

Color
Defines the code staples color in the editor window (if the **Single color** option is deselected). Select an item from the drop-down list or click the ellipsis button to select a color using the **Color** dialog where you can specify the required color from the palette.

Fonts
Use these options to set the *fonts*, *style*, *size* and *color* used in the editor. If the **Show only fixed-width fonts** option is checked, only fonts with fixed width are displayed in the **Font** dialog.

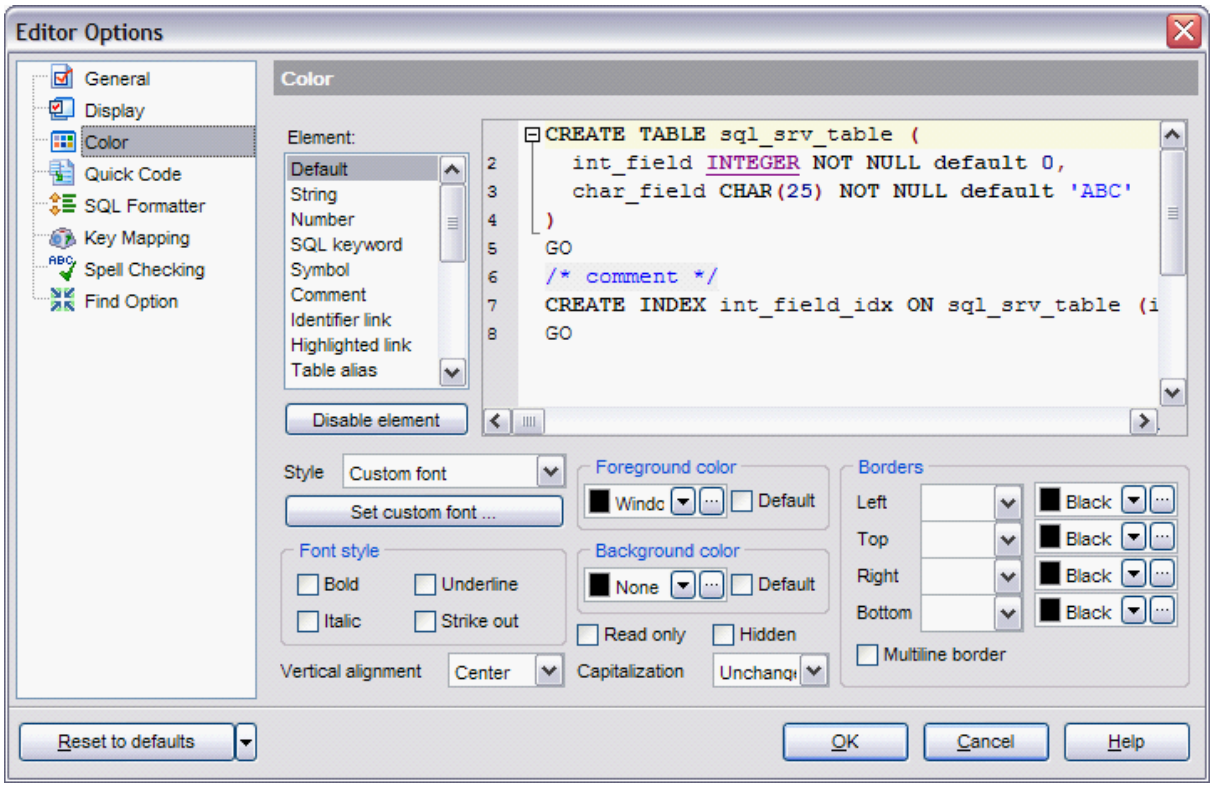
See also:
General

© 2008 EMS Database Management Solutions

Color
Quick Code
SQL Formatter
Key Mapping
Spell Checking
Find Option

11.1.2.3 Color

The **Color** section of the **Editor Options** dialog allows you to specify the **fonts**, **styles**, **foreground** and **background colors**, **borders** and other attributes of the text used by the editor to mark out various text fragments: *strings*, *numbers*, *SQL keywords*, *symbols*, *comments*, *identifier links*, *highlighted links*, *table aliases*, *SQL datatypes*, *SQL variables*, *SQL functions*, *wrong symbols*, *exceptions*, *parenthesis match*, *current lines*, *selected text*, *search marks*.



When setting colors, select an item from the drop-down list or click the ellipsis button to select a color using the **Color** dialog where you can specify the required color from the palette.

See also:

© 2008 EMS Database Management Solutions

General
Display
Quick Code
SQL Formatter
Key Mapping
Spell Checking
Find Option

11.1.2.4 Quick Code

Select an object in the **Completion list of objects** and set quick code parameters for the selected object. For your convenience the preview area (to the right of the **Completion list**) illustrates the changes being made to each of the objects.

Repeat this procedure for each of the objects.

☑ **Code completion**
If this option is checked, then on typing the first word characters in the SQL text editor you will be offered some variants for the word completion in a popup list (an analogue of the **Code Insight** feature in **Delphi IDE**). The popup list will appear after a period of time defined by the **Delay** option.

Sensitivity
This option allows you to set the number of characters to be typed before code completion is activated.

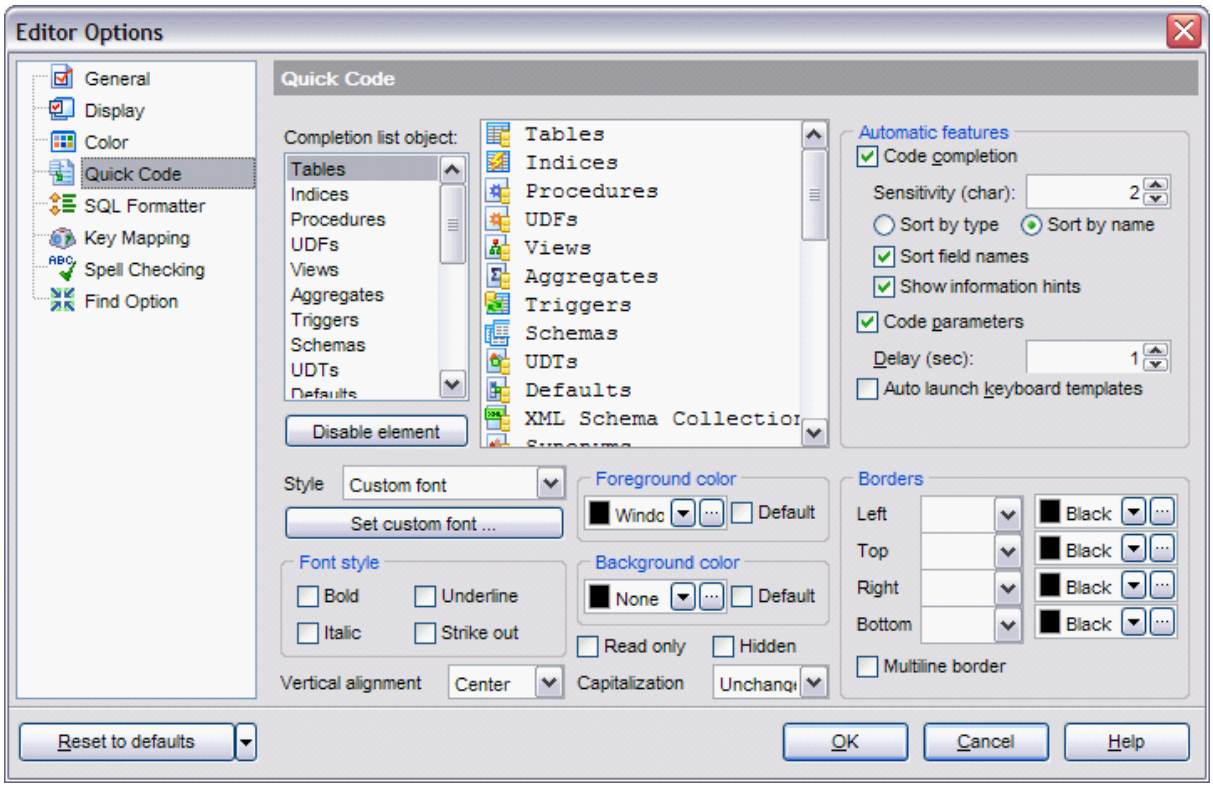
Specify whether items of the code completion list should be sorted ◉ *by type* or ◉ *by name*, and check the ☑ **Sort field names** option to apply sorting for field names as well.

☑ **Code parameters**
If this option is checked, the Delphi-like hint for functions is enabled.

Delay
Using this option you can change the time after which completion variants popup.

☑ **Auto launch keyboard templates**
Allows you to use keyboard templates for faster typing frequently used expressions (see Keyboard Templates).

© 2008 EMS Database Management Solutions



Select the **style** to be applied to the words inserted automatically (*Custom font*, *Font style and colors*, *Back and foreground* or *Background only*). Then, depending on the style selection, **set custom font**, **Font style**, **Foreground color**, **Background color**, **Borders**.

Vertical alignment
Allows you to set the vertical alignment of the words inserted automatically. Possible values are: *Top*, *Center*, *Bottom*.

Capitalization
Allows you to change the case of the words inserted automatically. Possible values are: *Unchanged*, *Upper case*, *Lower case*, *Toggle case*, *Initial caps*.

See also:
General
Display
Color
SQL Formatter
Key Mapping
Spell Checking
Find Option

© 2008 EMS Database Management Solutions

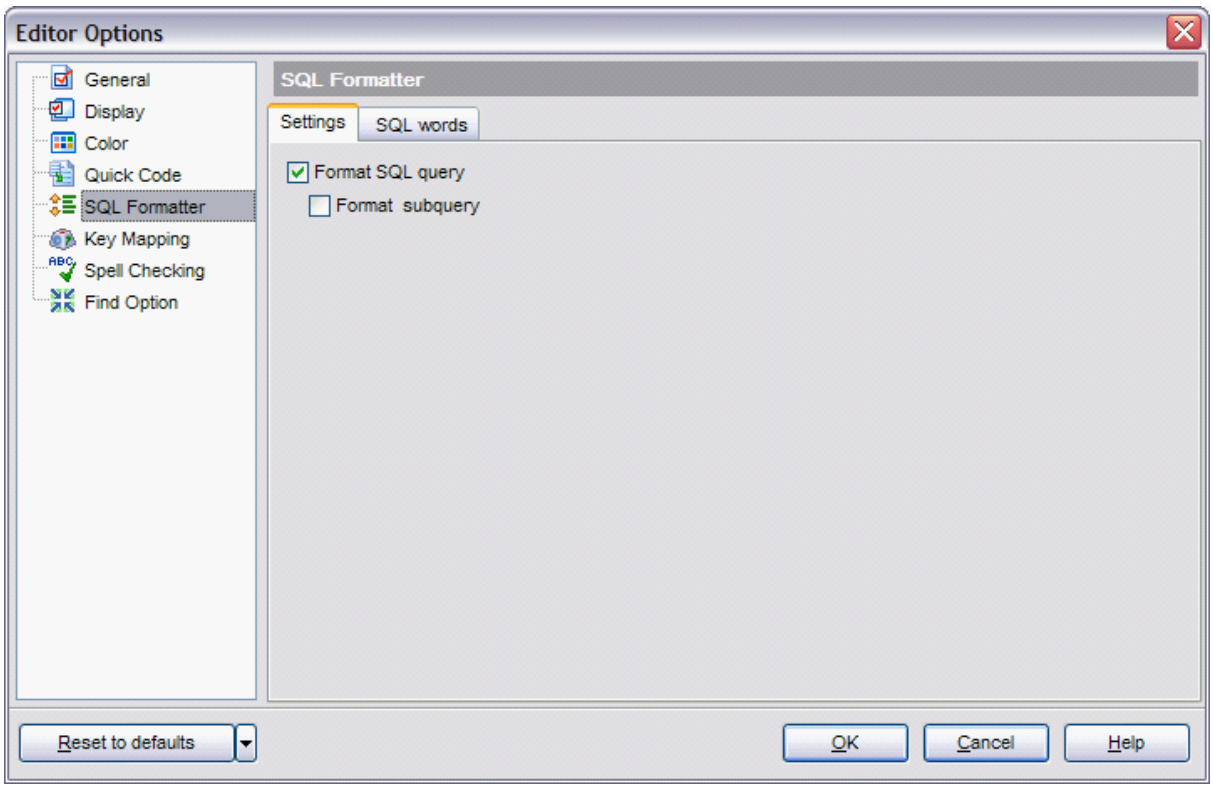
11.1.2.5 SQL Formatter

SQL Formatter is a feature implemented in SQL Editor and is a useful tool for formatting SQL queries, making SQL statements easy to read.

The **Settings** tab of the **SQL Formatter** section allows you to enable this feature and apply SQL formatting to subqueries, if necessary.

Format SQL query
Check this option to enable SQL formatting.

Format subquery
Enables SQL formatting for subqueries.

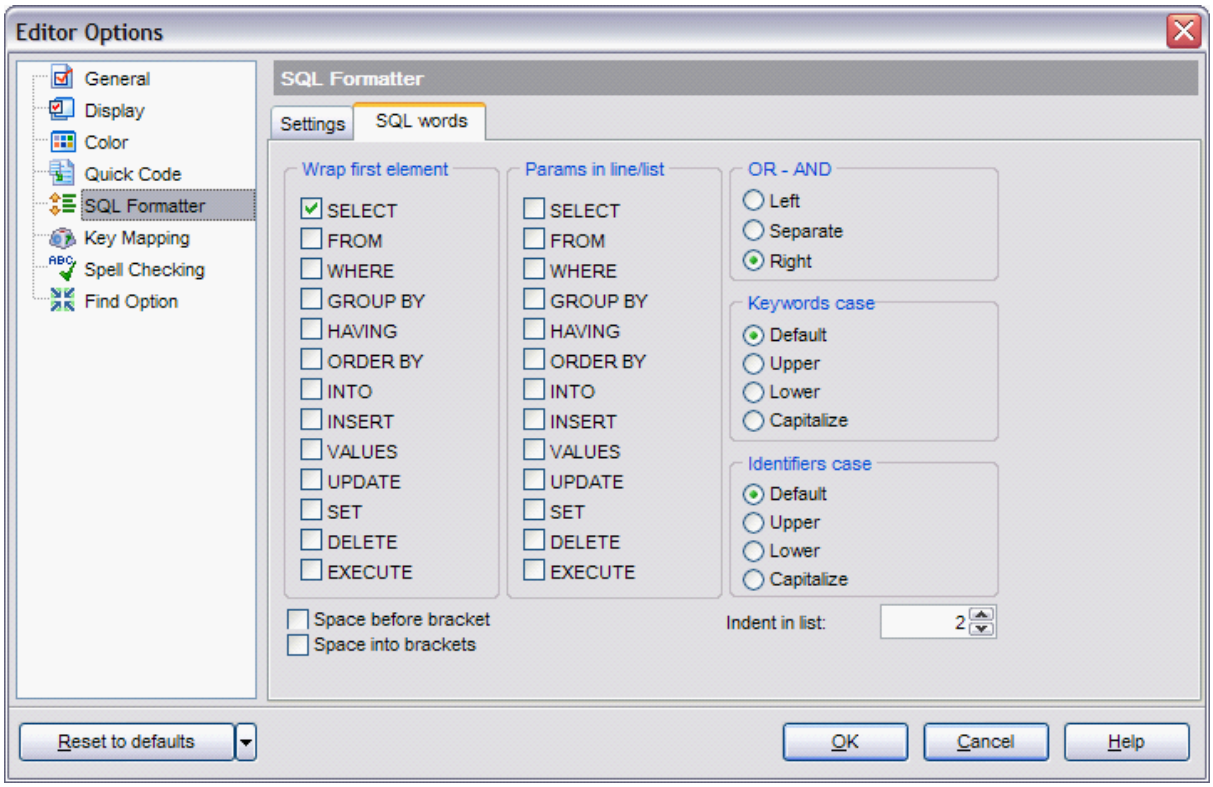


The **SQL words** tab of the **SQL Formatter** section allows you to select the key words for each action of SQL formatter and to set formatting parameters.

Wrap first element
Wraps the selected text at a specific column. Select the SQL key words after which formatting should be applied.

Params in line/list
Allows you to display the parameters followed by the defined key words in list or in line.

© 2008 EMS Database Management Solutions



OR - AND
Set the placement of the *AND* an *OR* operators according to the operands followed by them. See the example below.

Left

WHERE
AND ...
AND ...
AND ...

Separate

WHERE
...
AND
...
AND

Right

WHERE
... AND
... AND
... AND

© 2008 EMS Database Management Solutions

The **Keywords case** / **Identifiers case** options allow you to define the case of the corresponding items.
You can choose *UPPER*, *lower*, *Capitalize*.
Default case means that the name of the identifier/keyword remains "AS IS".

Space before bracket
Adds a "space" character before the opening bracket and after the closing one.

Space into brackets
Adds a "space" character after the opening bracket and before the closing one.

Indent in list
Sets the size of indent relatively to the previous string.

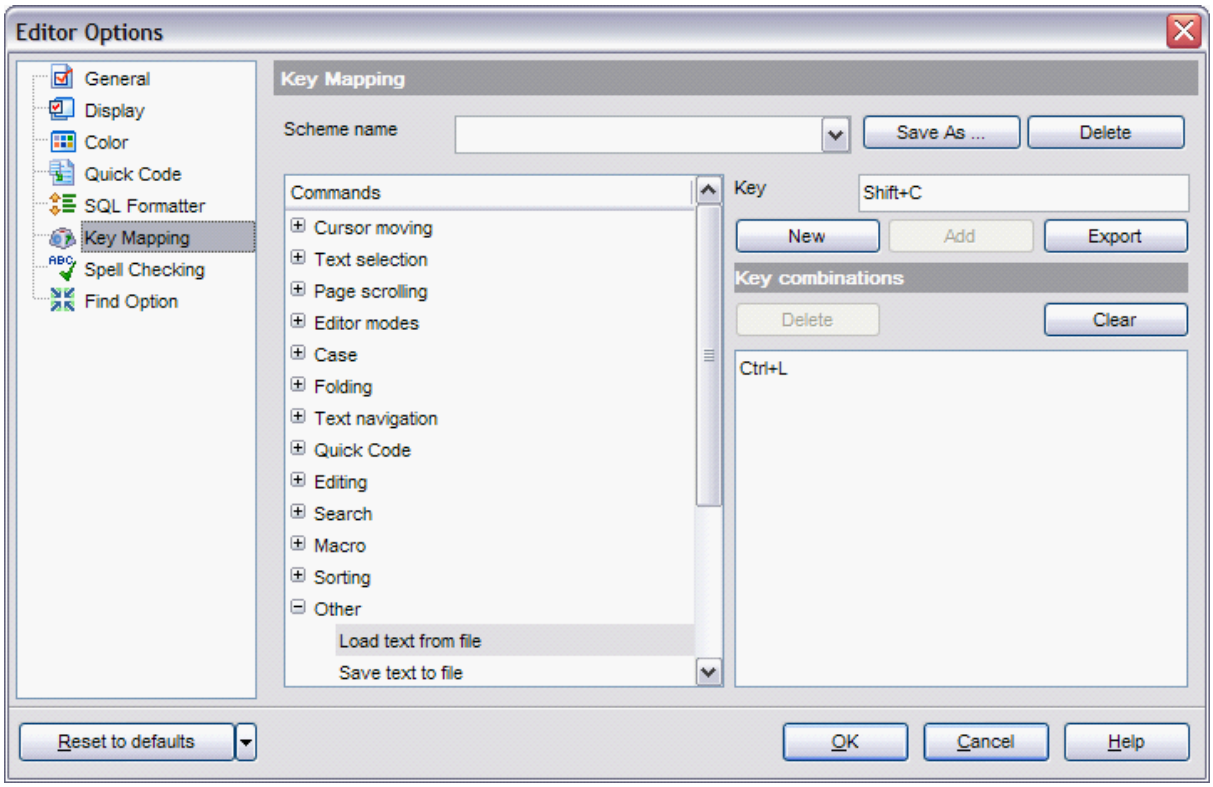
See also:
General
Display
Color
Quick Code
Key Mapping
Spell Checking
Find Option

11.1.2.6 Key Mapping

For your convenience **key mapping** is provided in SQL Manager 2008 for SQL Server. On this page you can set the shortcuts for various commands/operations according to your needs.

Use the **Commands** list on the right to select the command for which you need to make a shortcut, then place cursor into the **Key** editor and press the key combination you find useful (use *Ctrl Alt Shift* buttons). After setting the shortcut, press the **New** button to add it to the list of existing **Key combinations**.

© 2008 EMS Database Management Solutions



Note: it is possible to set more that one key combination for the same command/operation (e.g. *Ctrl-K*, *Ctrl-H*) using the **Add** button.

If necessary, you can export the current Key mapping list to an external file by pressing the **Export** button.

Manage the shortcuts within the Key combinations list using the **Delete** (to remove the selected item) and the **Clear** (to remove all shortcuts for this command/operation) buttons.

It is also possible to save a custom Key mapping scheme, if necessary:

- set the shortcuts for the appropriate commands/operations;
- click the **Save As...** button;
- input the new scheme name in the corresponding dialog.

To delete a scheme, select it in the Scheme name drop-down list and press the **Delete** button.

See also:

General
Display
Color
Quick Code
SQL Formatter

© 2008 EMS Database Management Solutions

Spell Checking
Find Option

11.1.2.7 Spell Checking

Spell checking is a new feature implemented in SQL Manager 2008 for SQL Server for your convenience.

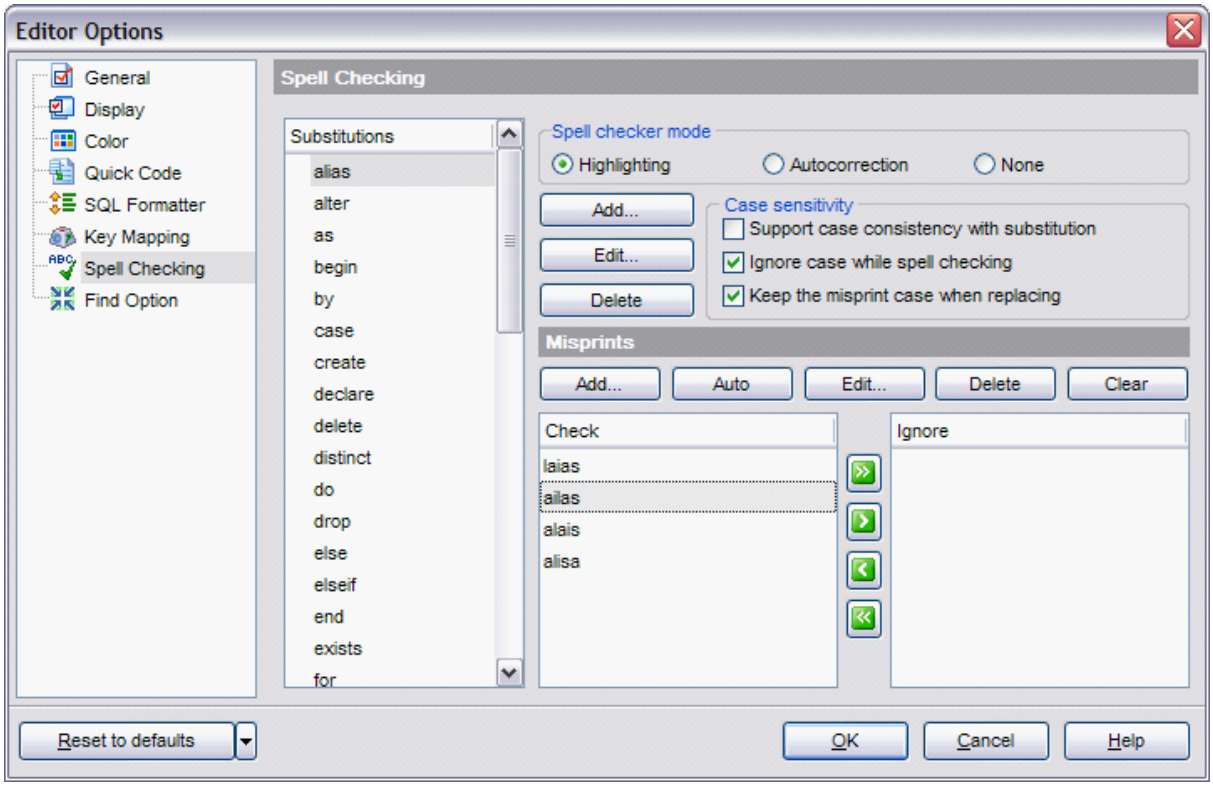
Set the necessary **Spell checker mode**:

Highlighting
In this mode incorrectly spelled and misprinted words are highlighted in the editor.

Autocorrection
In this mode incorrectly spelled and misprinted words are replaced with the corresponding words from the **Substitutions** list automatically.

None
In this mode the spelling checker is disabled.

Use the **Add...** button to add a new item to the **Substitutions** list, the **Edit...** button to alter the selected substitution, and the **Delete** button to remove the selected substitution from the spelling checker vocabulary.



© 2008 EMS Database Management Solutions

Case sensitivity

Support case consistency with substitution
If this option is selected, the spelling checker uses the case of words-substitutions when performing a replacement.

Ignore case while spell checking
Check this option to disable case checking.

Keep the misprint case when replacing
Check this option if you do not wish to change the case of the replaced word.

Misprints
Controls of this group allow you to manage the spelling checker vocabulary: use the **Add...** button to add a new misprint to the vocabulary, the **Auto** button to use the default list of misprints, the **Edit...** button to change the selected misprint, the **Delete** button to remove the selected misprint from the vocabulary, and the **Clear** button to empty the list of misprints for the currently selected substitution.

It is also possible to exclude a misprint from spell checking without deleting the misprint. This misprint will therefore remain in the vocabulary, but it will be ignored by the spelling checker.
To mark a misprint as excluded, you need to move it from the **Check** list to the **Ignore** list. Use the buttons or drag-and-drop operations to move the misprints from one list to another.

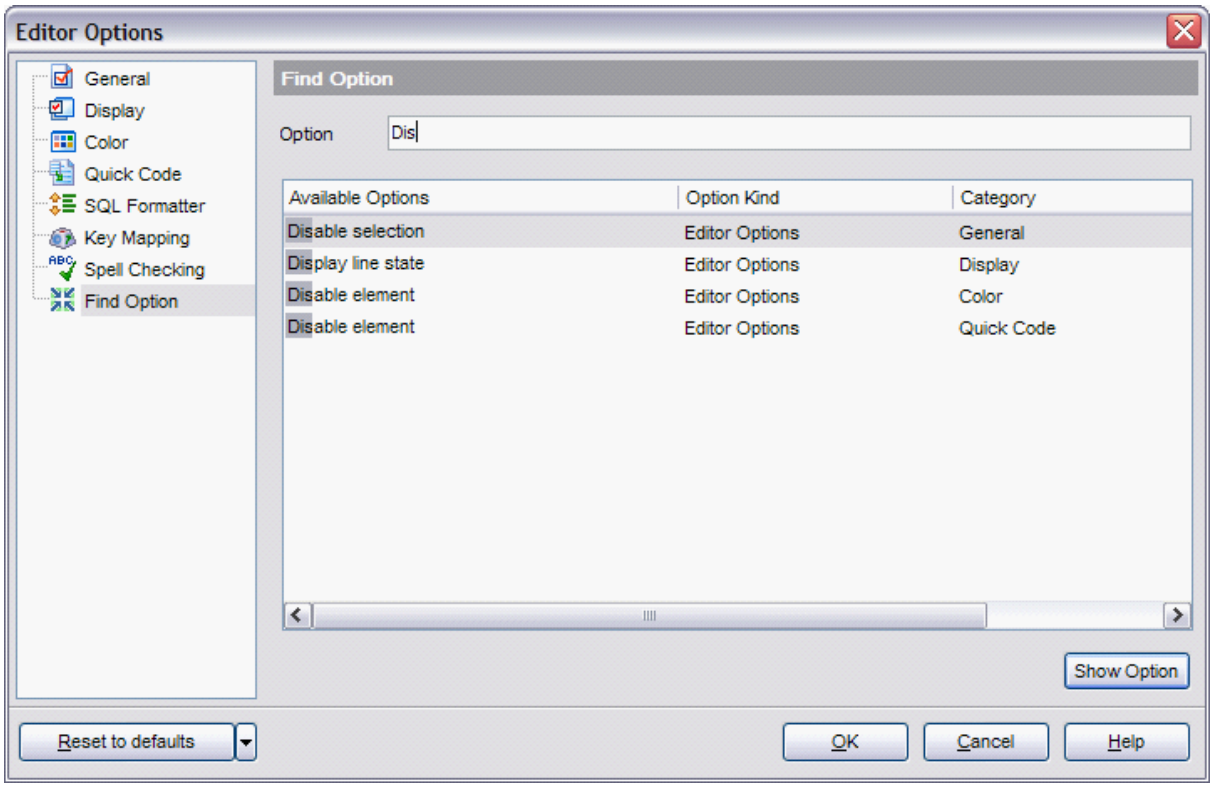
See also:
General
Display
Color
Quick Code
SQL Formatter
Key Mapping
Find Option

11.1.2.8 Find Option

The **Find Option** section allows you to search for options available within the **Editor Options** dialog easily and quickly.

Option
In this field you can enter the name of the option to search for within SQL Manager Editor Options.

© 2008 EMS Database Management Solutions



The **Available options** area lists all options of the Editor Options category according to the specified name. Select the required option in the list and click **Show Option** to open the corresponding section where you can view/edit the value of this option.

See also:

General
Display
Color
Quick Code
SQL Formatter
Key Mapping
Spell Checking

11.1.3 Visual Options

Visual Options allow you to customize the application interface style to your liking.

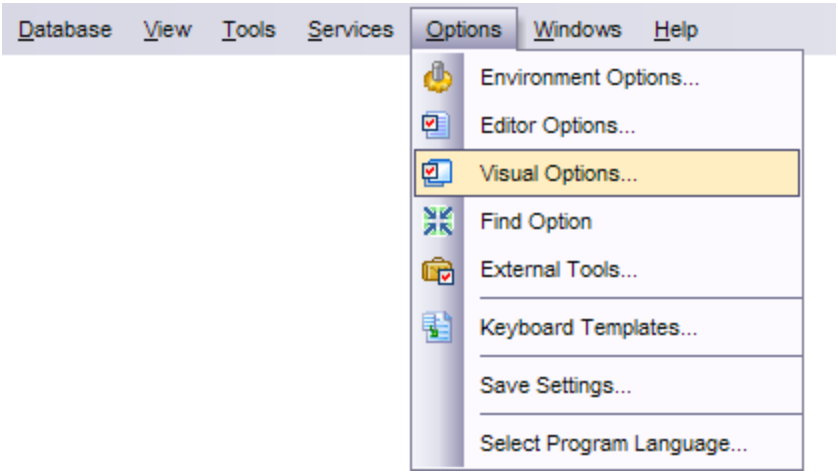
To open the **Visual Options** window, select the **Options | Visual Options...** main menu item, or use the **Visual Options** button on the main toolbar.

Use the **Scheme name** box to select the interface scheme you would like to be applied:

© 2008 EMS Database Management Solutions

MS Office 11 style, *MS Office XP style*, *MS Office 2000 style*, *Windows XP native style* or *Classic style*.

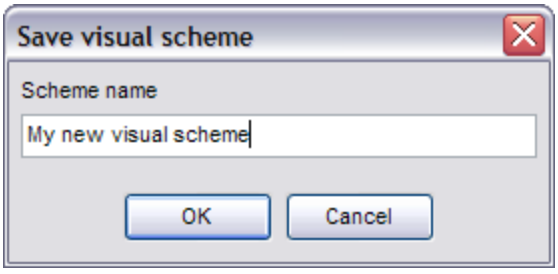
- Bars and menus
- Trees and lists
- Edit controls
- Check boxes
- Buttons
- Page controls
- Group boxes
- Splitters
- Navigation bar
- Find Option



For your convenience the previews illustrating the changes are displayed in the **Sample Group** area at the bottom of each section within the **Visual options** dialog.

It is also possible to create one's own interface scheme, if necessary:

- set your preferences within the available sections of the **Visual Options** dialog (Bars and menus, Trees and lists, Edit Controls, Check boxes, Buttons, etc.);
- click the **Save As...** button;
- specify the new scheme name in the **Save Visual Scheme** dialog.



Availability:
Full version (for Windows) **Yes**

© 2008 EMS Database Management Solutions

Lite version (for Windows)	**Yes**

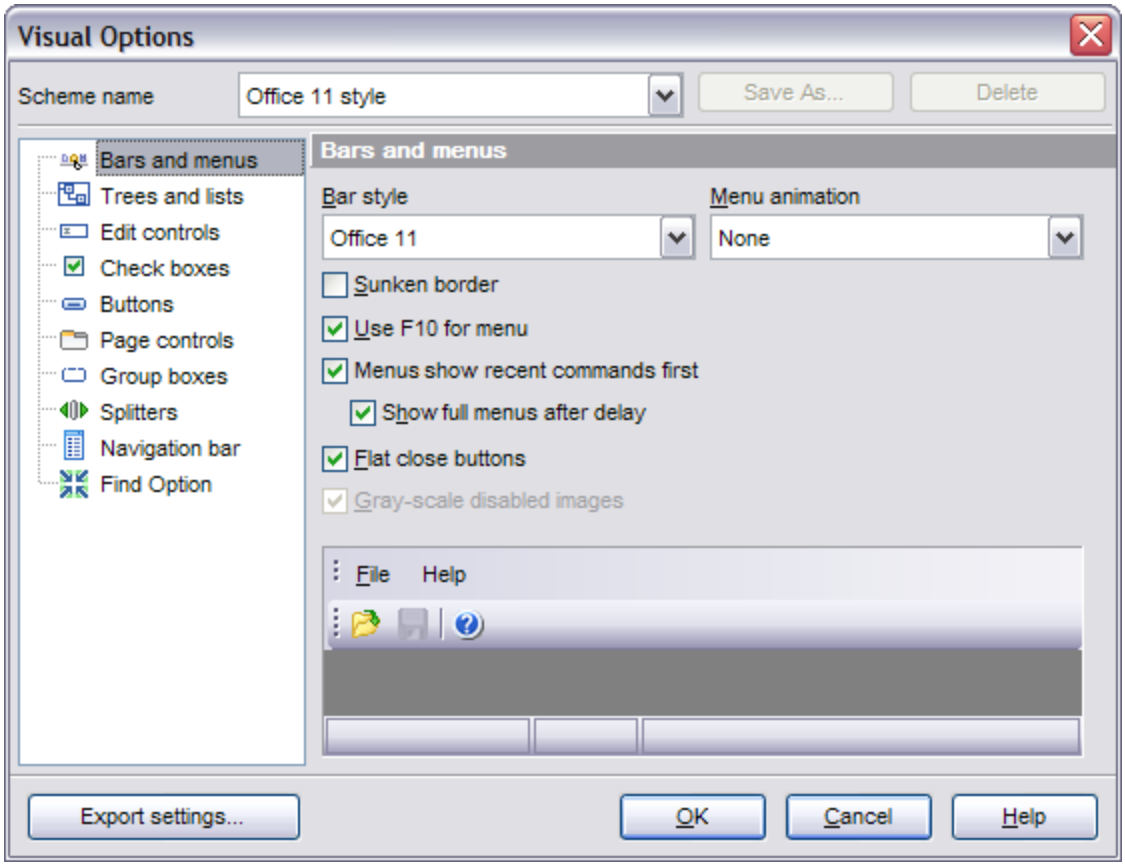
Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

Environment Options
Editor Options
Save Settings

11.1.3.1 Bars and menus

Use the **Bars and menus** section of the **Visual options** dialog to customize SQL Manager *bars style*, *menus animation* and a number of general options concerning toolbars and menus usage.



Bar style
Use the drop-down list to select the painting style that will be applied to the bars:
Standard
Enhanced

© 2008 EMS Database Management Solutions

Flat
XP native
Office 11

Note: If the *XP native* style is selected, bars use the currently applied XP theme to paint themselves. However, if the currently installed operating system is not Windows XP or the Windows Classic theme is currently applied, bars will be painted using the *Enhanced* style.

Menu animation
Use the drop-down list to specify the menu animation effects:
None (no animation)
Random (random choice: *Unfold*, *Slide*, *Fade*)
Unfold (unfolding menus)
Slide (sliding drop-down and popup menus)
Fade (menus fade in when appearing)

Sunken border
If this option is enabled, the border of each dockable bar is drawn using sunken borders. Otherwise, no border is drawn around the dockable bar.

Use F10 for menu
If this option is disabled, the application does not respond to the *F10* key press events and the main menu will not will be called.

Menus show recent commands first
This option determines whether the most frequently used items will be placed in menus at first position.
If this option is enabled, frequently used menu items are "promoted" and displayed higher on the list. Unused and infrequently used menu items are visually suppressed and appear "collapsed".

Hint: If you wish to disable this feature, you can also right-click the toolbar and select the **Customize...** popup menu item to call the Customize dialog, then proceed to the **Options** tab within the dialog, and deselect the *Menus show recently used commands first* option.

Show full menus after delay
This option is available only if the **Menus show recent commands first** option is selected.
If this option is enabled, infrequently used menu items (if they appear "collapsed") will be automatically expanded after a delay upon setting mouse cursor (or upon selection with the *Up*/*Down* keys) on the bottom of the menu. Otherwise, the menu expands only after clicking its bottom-most button (or using the *Ctrl+Down* shortcut).

Flat close buttons
This option determines the border style of the *Close* buttons.
If this option is enabled, the Close button is drawn flat. Otherwise, it has a 3D look.

Gray-scale disabled images
This option specifies whether default images must be painted faded.
By default, images within disabled links are painted grayscale when the *XP native* or *Office11* **bar style** is used. For other **bar styles**, such images are painted grayed out.
If this option is enabled, images of disabled links will be painted grayscale independently of

© 2008 EMS Database Management Solutions

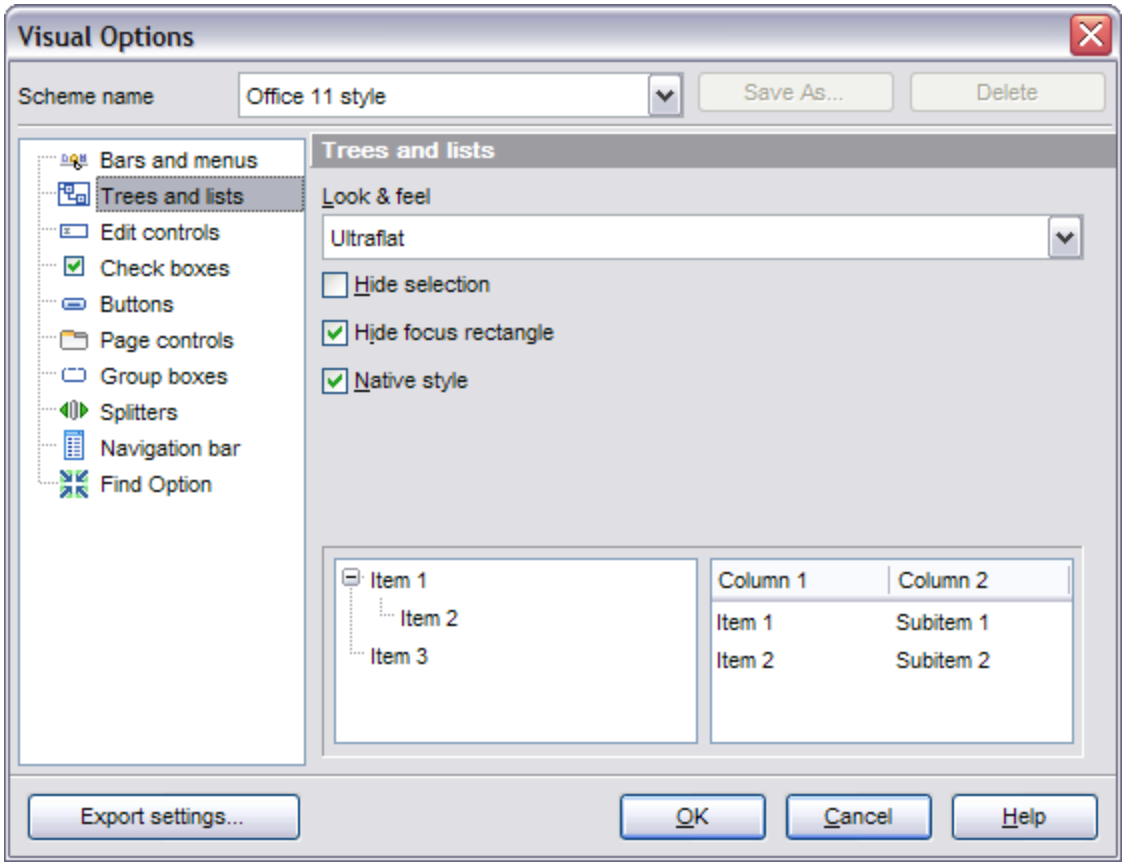
the selected **bar style**.

See also:

Trees and lists
Edit controls
Check boxes
Buttons
Page controls
Group boxes
Splitters
Navigation bar
Find Option

11.1.3.2 Trees and lists

Use the **Trees and lists** section of the **Visual options** dialog to view and edit the corresponding options.



© 2008 EMS Database Management Solutions

Look & feel
This setting determines the manner in which tree and list elements are painted. Use the drop-down list to select the painting style that will be applied to the trees and lists:
Standard
Flat
UltraFlat

Hide selection
This option specifies how selected tree nodes and list items are displayed when focus leaves the tree or list control.
If this option is enabled, selected nodes look like other nodes. Otherwise, selected nodes/items are highlighted within the tree/list.

Hide focus rectangle
This option determines whether a focus rectangle is displayed around the focused tree node or list item within the tree or list control.
If this option is disabled, the focused node/item is not highlighted but the focus rectangle is displayed around it.

Native style
This option determines whether the native Windows style will be applied to the trees and lists.
The option has the highest priority for trees and lists. If this option is selected, the tree nodes and list items are painted according to the native Windows style, regardless of other painting settings.

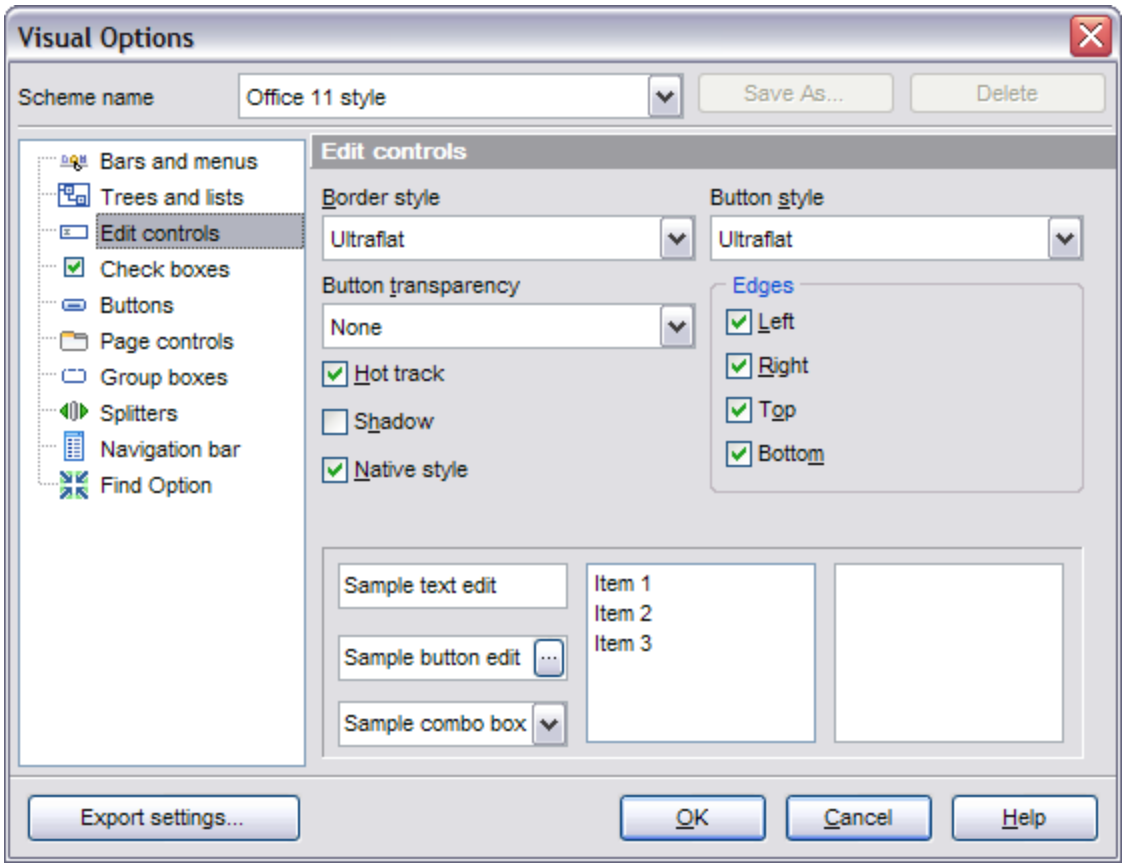
Note: The **Native style** option is currently supported for the Windows® XP operating system only.

See also:
Bars and menus
Edit controls
Check boxes
Buttons
Page controls
Group boxes
Splitters
Navigation bar
Find Option

11.1.3.3 Edit controls

Use the **Edit controls** section of the **Visual options** dialog to customize the appearance of various SQL Manager 2008 for SQL Server edit controls: *Border style*, *Button style*, *Button transparency*, etc.

© 2008 EMS Database Management Solutions



Border style / **Button style**
Use these drop-down lists to specify the style around an editor (the edit control **borders**) and select the painting style that will be applied to the edit control **buttons** (ellipsis button, arrow-down combo-box button, etc.) respectively:
None
Single
Thick
Flat
3D
UltraFlat

Button transparency
Represents the button transparency mode within an editor. Use the drop-down list to specify the transparency that will be applied to the edit control **buttons** (ellipsis button, arrow-down combo-box button, etc.):
None (a button is always displayed in a non-transparent fashion)
Inactive (a button is drawn when the editor has focus or when the mouse cursor is positioned over the button; otherwise, the button is transparent)
Always (a button is always transparent)
Hide inactive (a button is drawn only when the editor has focus; otherwise, the button is invisible)
Hide unselected (a button is drawn when the editor has focus or when the mouse cursor is positioned over the editor region; otherwise, the button is invisible).

© 2008 EMS Database Management Solutions

Edges
This group defines which edges are displayed within an editor. Check/uncheck the boxes to hide/show individual edges of the edit controls:
- *Left* (if unchecked, the left border edge is invisible)
- *Right* (if unchecked, the right border edge is invisible)
- *Top* (if unchecked, the top border edge is invisible)
- *Bottom* (if unchecked, the bottom border edge is invisible)

Hot track
This option specifies whether editor items are highlighted when the mouse cursor is positioned over an edit control. Select this option to highlight an edit control in response to mouse movements.

Shadow
If this option is selected, a shadow is displayed for the edit controls.

Native style
This option determines whether the native Windows style will be applied to the edit controls.
The option has the highest priority for edit controls. If this option is selected, the edit controls are painted according to the native Windows style, regardless of other painting settings.

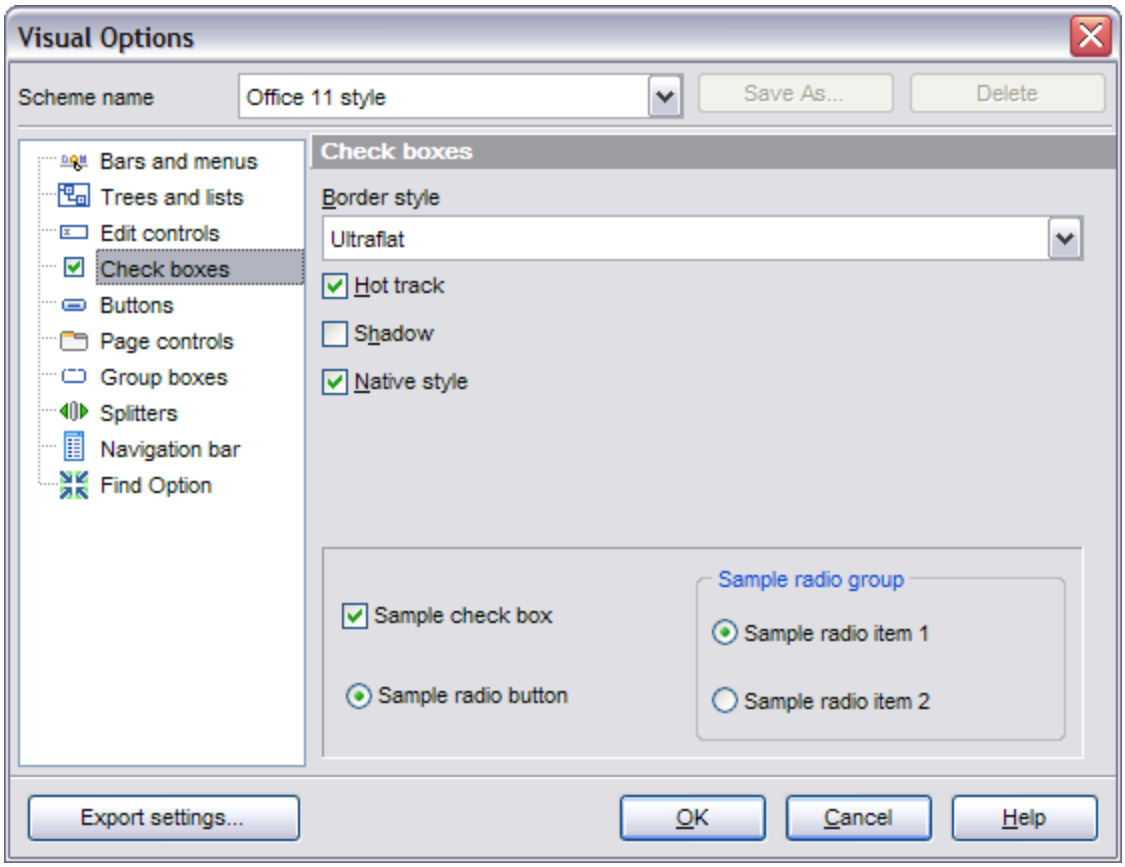
Note: The **Native style** option is currently supported for the Windows® XP operating system only.

See also:
Bars and menus
Trees and lists
Check boxes
Buttons
Page controls
Group boxes
Splitters
Navigation bar
Find Option

11.1.3.4 Check boxes

Use the **Check boxes** section of the **Visual options** dialog to customize the *border style* and the appearance of *check boxes* and *radio buttons*.

© 2008 EMS Database Management Solutions



Border style
This setting determines the manner in which check box and radio group borders are painted. Use the drop-down list to select the painting style that will be applied to the check boxes and radio groups:
None
Single
Thick
Flat
3D
UltraFlat

Shadow
If this option is selected, a shadow is displayed for the check boxes and radio groups.

Native style
This option determines whether the native Windows style will be applied to the check boxes and radio buttons.
The option has the highest priority for check boxes and radio buttons. If this option is selected, the check boxes and radio buttons are painted according to the native Windows style, regardless of other painting settings.

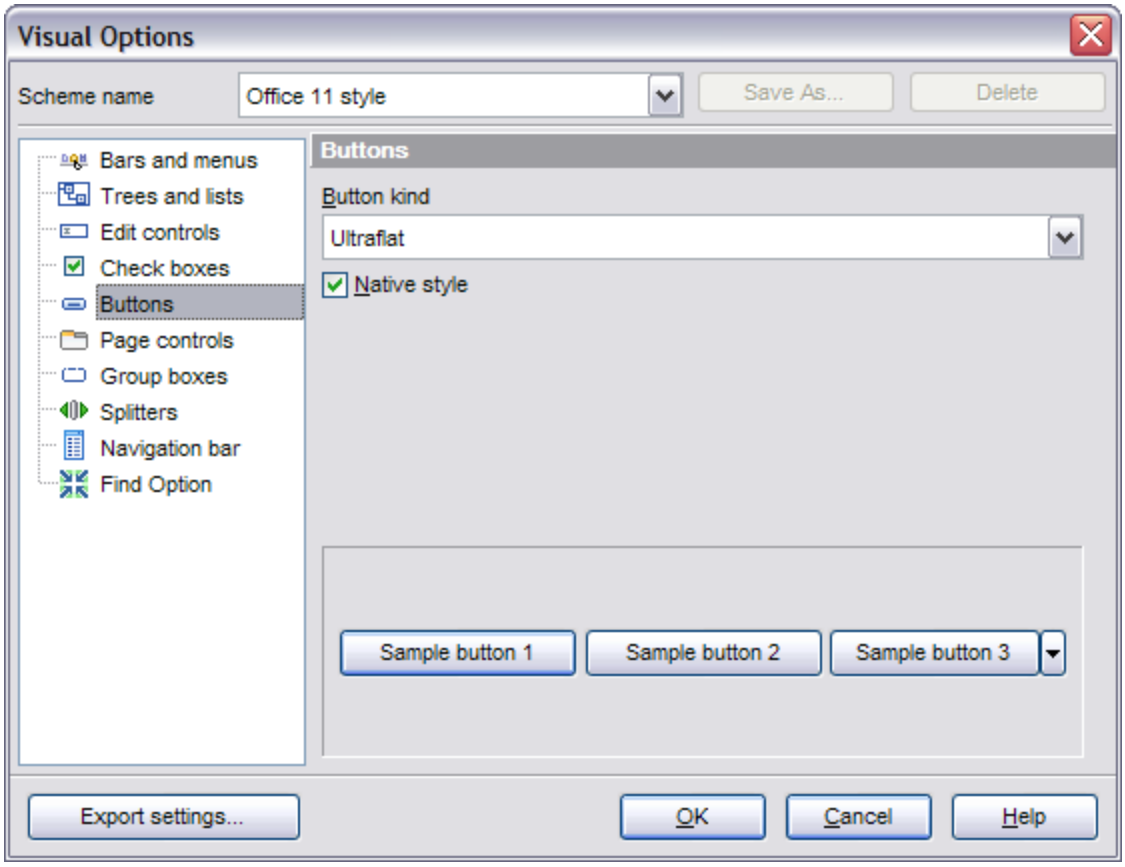
Note: The **Native style** option is currently supported for the Windows® XP operating system only.

© 2008 EMS Database Management Solutions

See also:
Bars and menus
Trees and lists
Edit controls
Buttons
Page controls
Group boxes
Splitters
Navigation bar
Find Option

11.1.3.5 Buttons

Use the **Buttons** section of the **Visual options** dialog to customize SQL Manager *buttons*.



Button kind
This setting determines the manner in which a button is painted. Use the drop-down list

© 2008 EMS Database Management Solutions

to select the painting style that will be applied to buttons:
Standard
Flat
UltraFlat

Native style
This option determines whether the native Windows style will be applied to the buttons. The option has a higher priority than the **Button kind** setting. If this option is selected, the buttons are painted according to the native Windows style, otherwise the **Button kind** selection is applied.

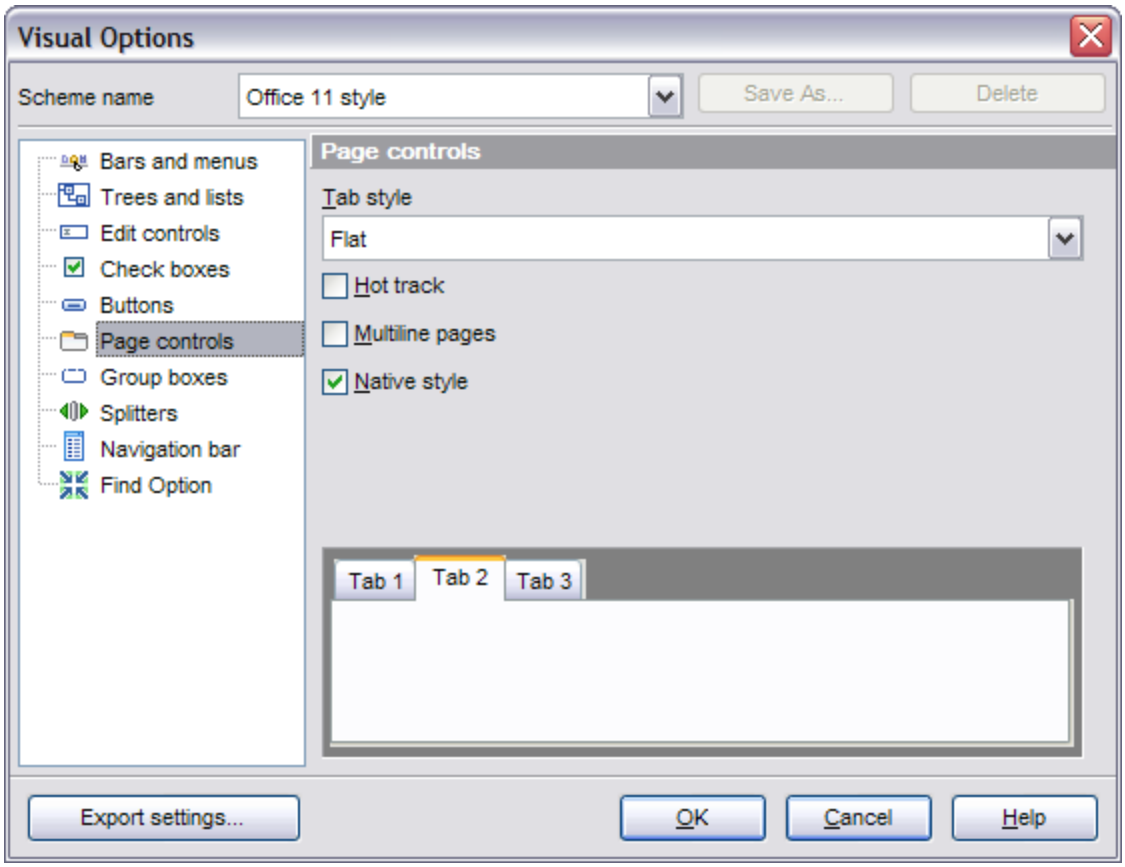
Note: The **Native style** option is currently supported for the Windows® XP operating system only.

See also:
Bars and menus
Trees and lists
Edit controls
Check boxes
Page controls
Group boxes
Splitters
Navigation bar
Find Option

11.1.3.6 Page controls

Use the **Page controls** section of the **Visual options** dialog to customize the style of all SQL Manager *page controls*.

© 2008 EMS Database Management Solutions



Tabs are visual elements of **tab controls**. Their purpose is to identify pages and switch between them. Once a tab is clicked, the corresponding page is selected.
Pages are container controls that represent the contents of tab controls. Tab controls contain a single page, whose context is to be updated each time the selected tab changes. **Page controls** contain the number of pages equal to the number of tabs.

Tab style
Use the drop-down list to select the painting style that will be applied to the tab controls:
Tabs (tabs are painted as notebook tabs)
Buttons (the selected tab is painted as a pressed button, other tabs are painted as released buttons)
Flat (tabs are painted as notebook tabs, but appear lowered slightly)

Hot track
This option specifies whether tab captions are highlighted when the mouse pointer hovers over tabs. Select this option to enable tab highlighting.

Multiline pages
This option specifies whether tabs are arranged across several rows.
If this option is enabled, tabs are automatically arranged into the minimum number of rows required to fit all of them. If this option is disabled, tabs are displayed within a single row.

Native style
This option determines whether the native Windows style will be applied to the tab

© 2008 EMS Database Management Solutions

controls.
The option has the highest priority for the tab controls. If this option is selected, the tabs are painted according to the native Windows style, regardless of other painting settings.

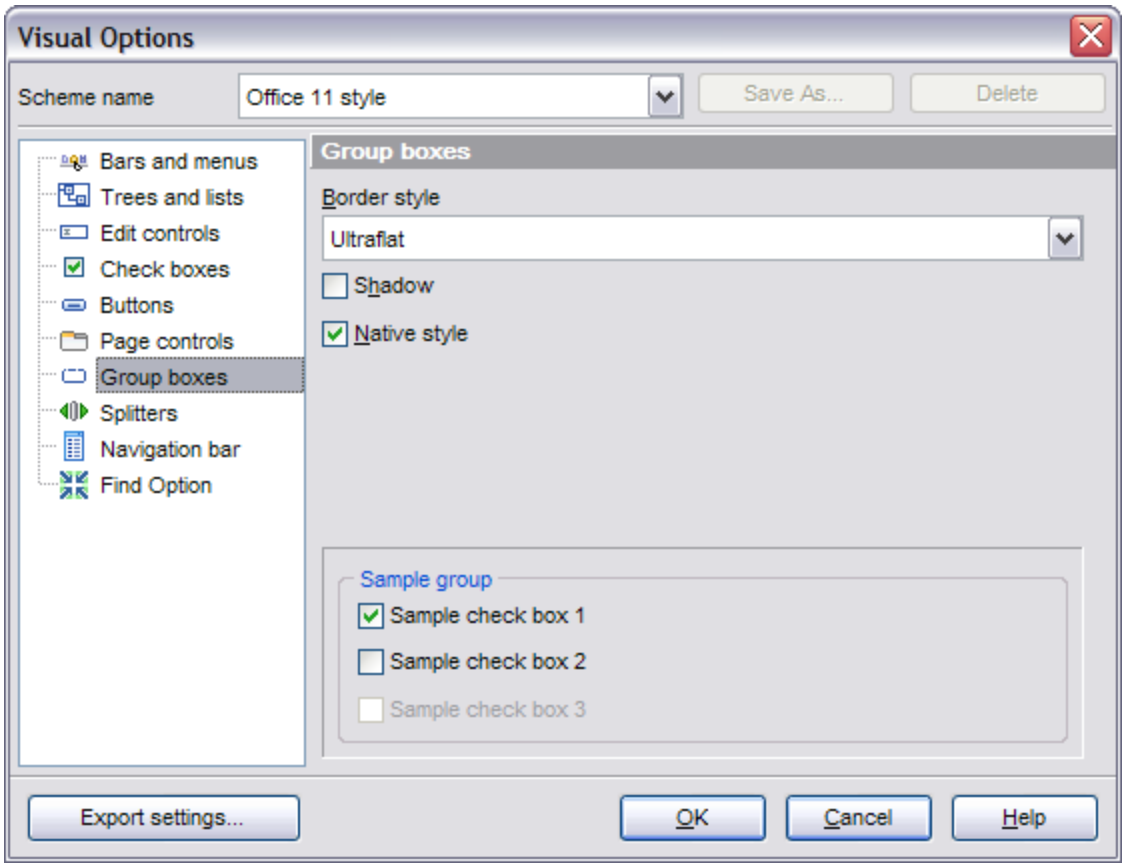
Note: The **Native style** option is currently supported for the Windows® XP operating system only.

See also:
Bars and menus
Trees and lists
Edit controls
Check boxes
Buttons
Group boxes
Splitters
Navigation bar
Find Option

11.1.3.7 Group boxes

Use the **Group boxes** section of the **Visual options** dialog to customize all SQL Manager *group boxes* to your liking.

© 2008 EMS Database Management Solutions



Border style
This setting determines the manner in which group box borders are painted. Use the drop-down list to select the painting style that will be applied to the group boxes:
None
Single
Thick
Flat
3D
UltraFlat

Shadow
If this option is selected, a shadow is displayed for the group boxes.

Native style
This option determines whether the native Windows style will be applied to the group boxes.
The option has the highest priority for the group boxes. If this option is selected, the group boxes are painted according to the native Windows style, regardless of other painting settings.

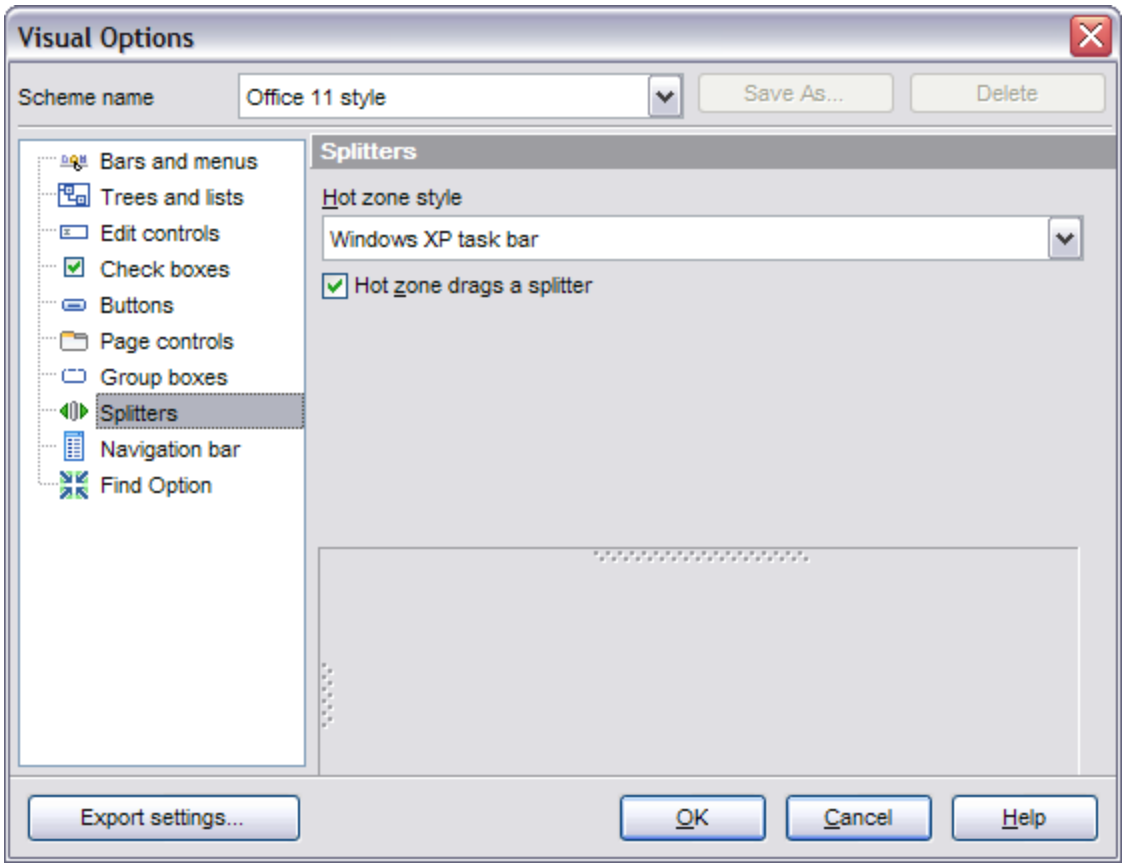
Note: The **Native style** option is currently supported for the Windows® XP operating system only.

© 2008 EMS Database Management Solutions

See also:
Bars and menus
Trees and lists
Edit controls
Check boxes
Buttons
Page controls
Splitters
Navigation bar
Find Option

11.1.3.8 Splitters

Use the **Splitters** section of the **Visual options** dialog to customize all SQL Manager *splitters* to your liking.



Hot zone style
This setting determines the manner in which splitter hot zones are painted. Use the drop-down list to select the hot zone style that will be applied to the splitters:

© 2008 EMS Database Management Solutions

Windows XP task bar
Media Player 8
Media Player 9
Simple
None (hot zone is disabled)

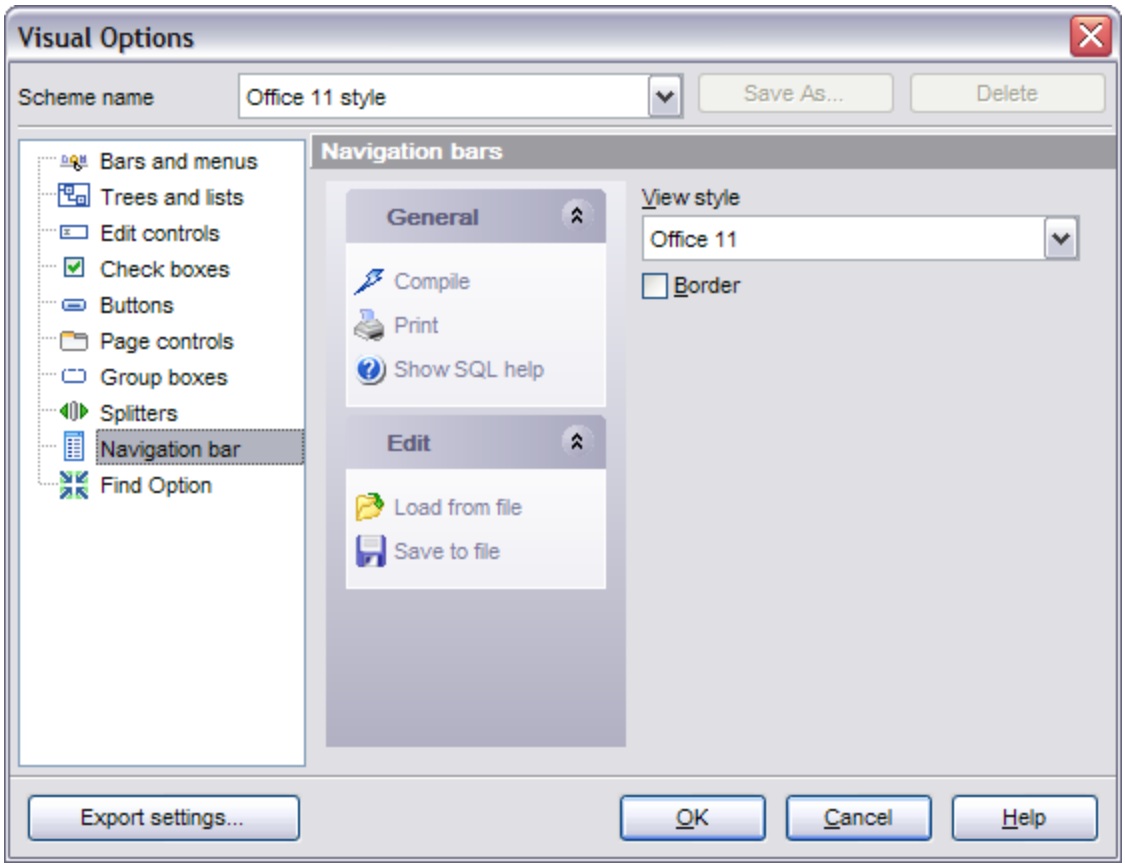
Hot zone drags a splitter
This option is applied when the **Hot zone style** value is different from *None*, and specifies whether the splitter can be dragged by its hot zone.
If this option is enabled, you can drag the splitter by its hot zone (i.e. use any part of the splitter's area for dragging). Otherwise, attempts to drag the splitter by the hot zone will have no effect.

See also:
Bars and menus
Trees and lists
Edit controls
Check boxes
Buttons
Page controls
Group boxes
Navigation bar
Find Option

11.1.3.9 Navigation bar

Use the **Navigation bar** section of the **Visual options** dialog to customize the *Navigation bars* of all SQL Manager tools according to your preferences.

© 2008 EMS Database Management Solutions



View style
Several types of control representations (views) are available for Navigation bars. View styles define the appearance of Navigation bar elements (background, scroll buttons, group headers, links/items and hint boxes). Use the drop-down list to select the view style that will be applied to the Navigation bars:
Office 11
XP
Simple
Advanced

Border
This option specifies whether a border style will be applied to the Navigation bars.
If this option is enabled, the Navigation bars have thin borders. Otherwise, no borders are drawn around the Navigation bars.

See also:
Bars and menus
Trees and lists
Edit controls
Check boxes
Buttons

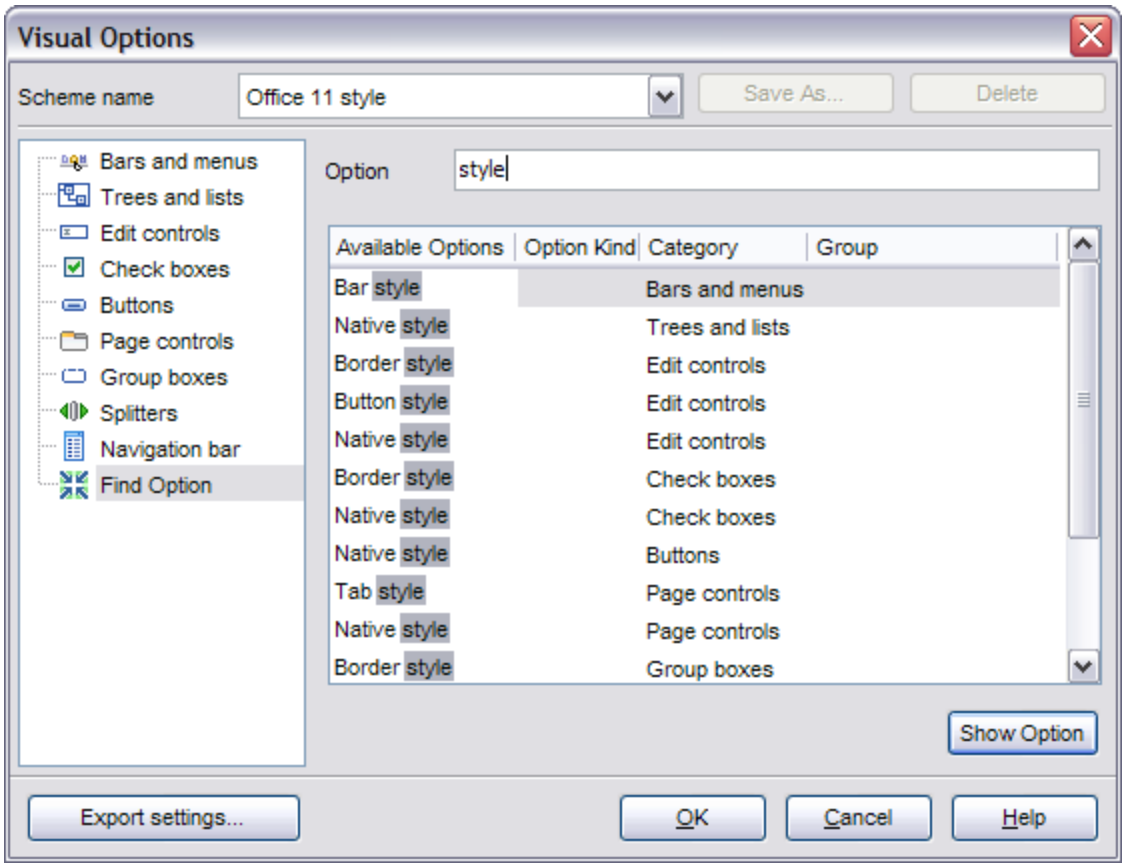
© 2008 EMS Database Management Solutions

Page controls
Group boxes
Splitters
Find Option

11.1.3.10 Find Option

The **Find Option** section allows you to search for options available within the **Visual Options** dialog easily and quickly.

Option
In this field you can enter the name of the option to search for within SQL Manager Visual Options.



The **Available options** area lists all options of the Visual Options category according to the specified name. Select the required option in the list and click **Show Option** to open the corresponding section where you can view/edit the value of this option.

See also:

© 2008 EMS Database Management Solutions

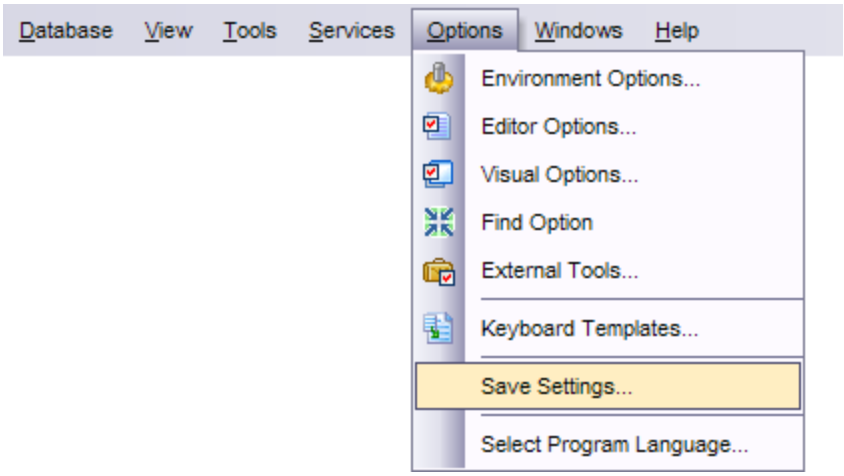
Bars and menus
Trees and lists
Edit controls
Check boxes
Buttons
Page controls
Group boxes
Splitters
Navigation bar

11.1.4 Save Settings

Save Settings Wizard allows you to export the settings of SQL Manager 2008 for SQL Server - wholly or partially - to a single **.reg* file which can be applied afterwards to SQL Manager 2008 for SQL Server installed on another machine, or it can be used to backup previous settings.

To start the wizard, select the **Options | Save Settings** main menu item.

- Specifying destination file
- Selecting settings
- Selecting databases
- Saving settings



Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**Yes**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

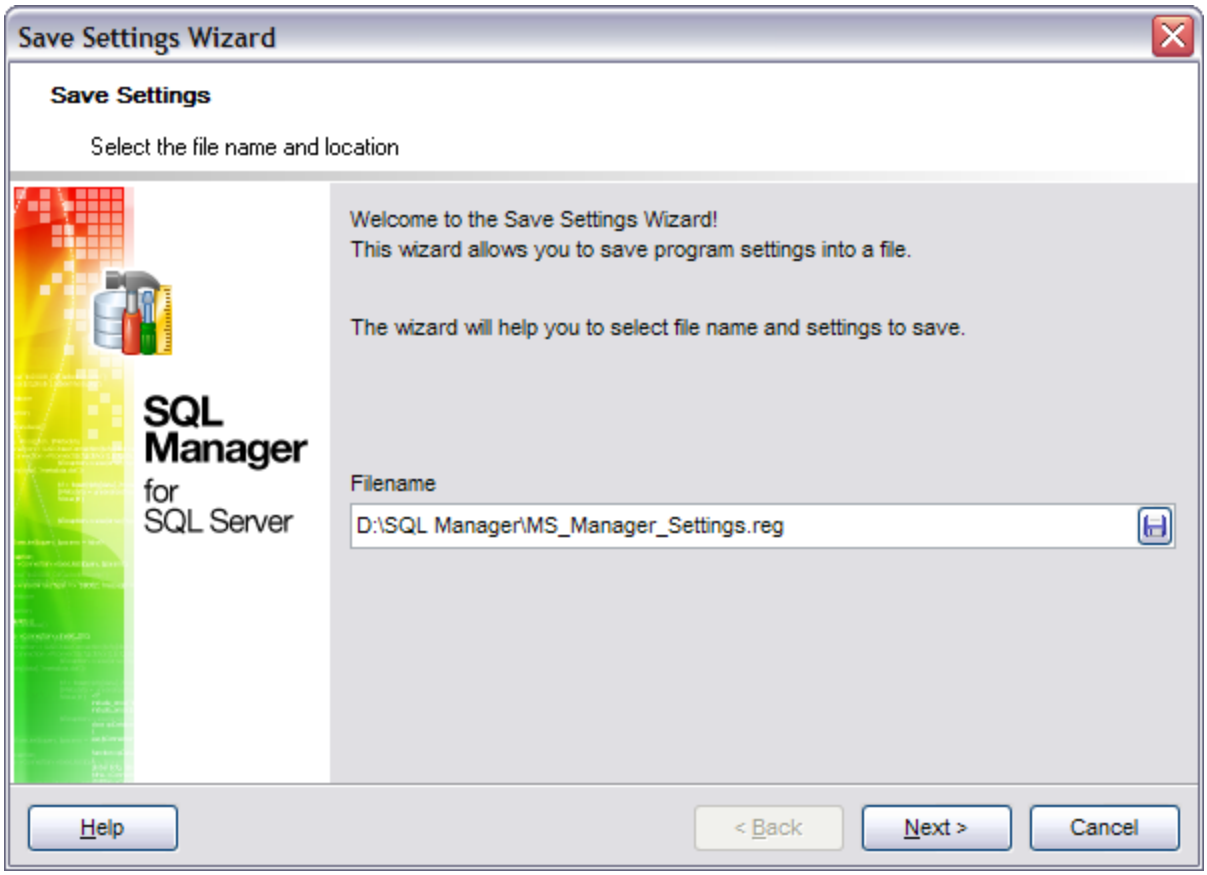
© 2008 EMS Database Management Solutions

See also:
Environment Options
Editor Options
Visual Options

11.1.4.1 Specifying destination file

This step of the wizard allows you to specify the location of the destination file.

Filename
Use the button to set the path to the **.reg* file where the application settings are to be saved.

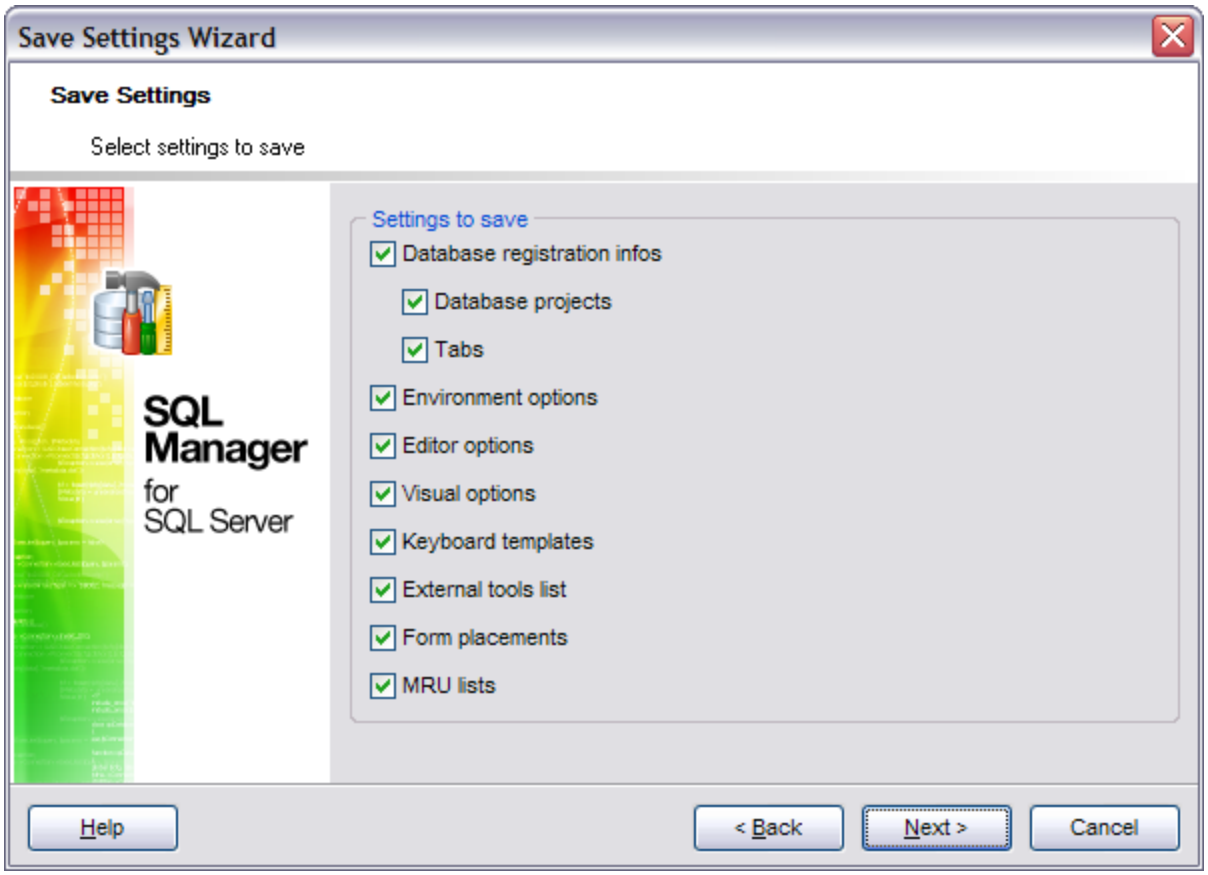


Press the **Next** button to proceed to the next step of the wizard.

© 2008 EMS Database Management Solutions

11.1.4.2 Selecting settings

This step of the wizard allows you to specify the information you need to be saved to the result file: *Database registration* info, *Database projects*, *Tabs*, *Environment options*, *Editor options*, *Visual options*, *Keyboard templates*, *External tools list*, *Form placements*, *MRU lists*.



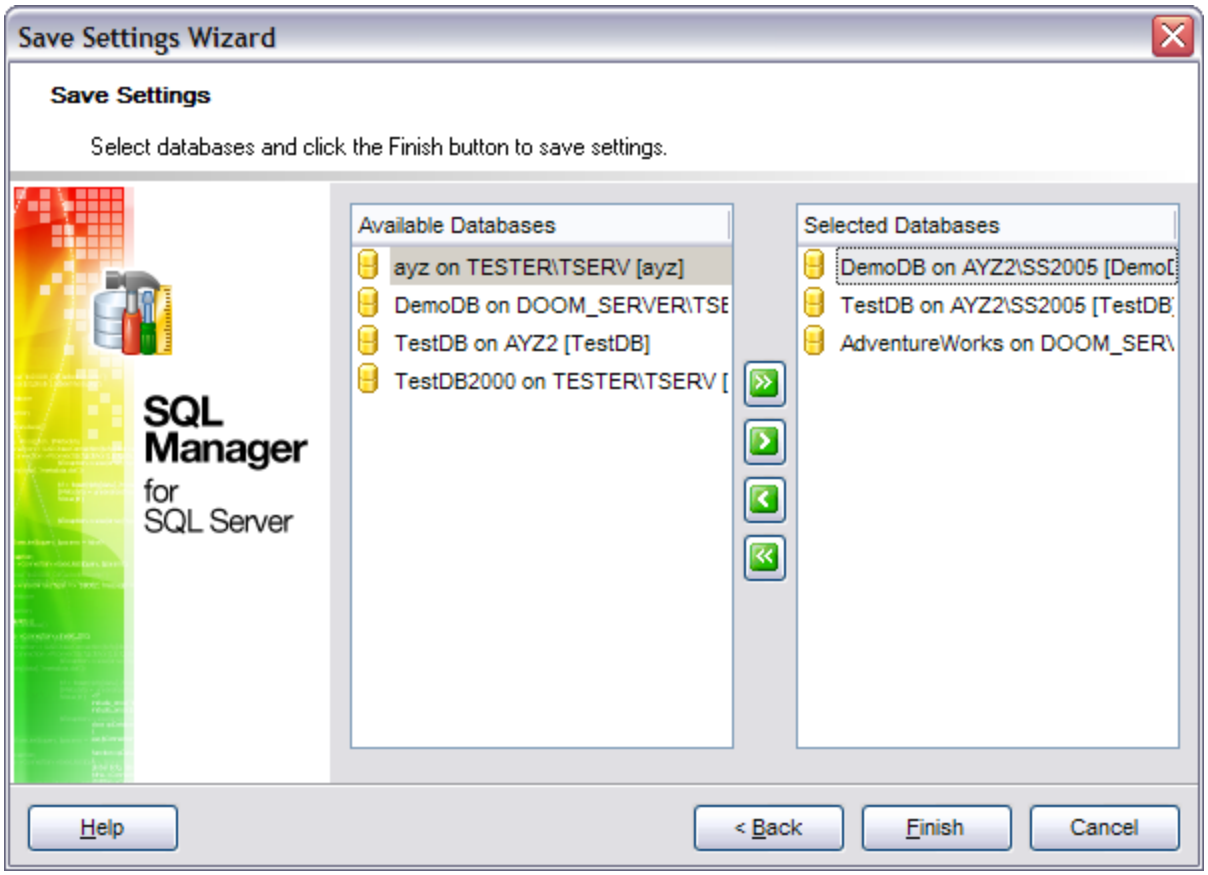
Press the **Next** button to proceed to the next step of the wizard.

11.1.4.3 Selecting databases

This step of the wizard allows you to select the database(s) to save the registration settings.

To select a database, you need to move its alias from the **Available Databases** list to the **Selected Databases** list. Use the buttons or drag-and-drop operations to move the databases from one list to another.

© 2008 EMS Database Management Solutions



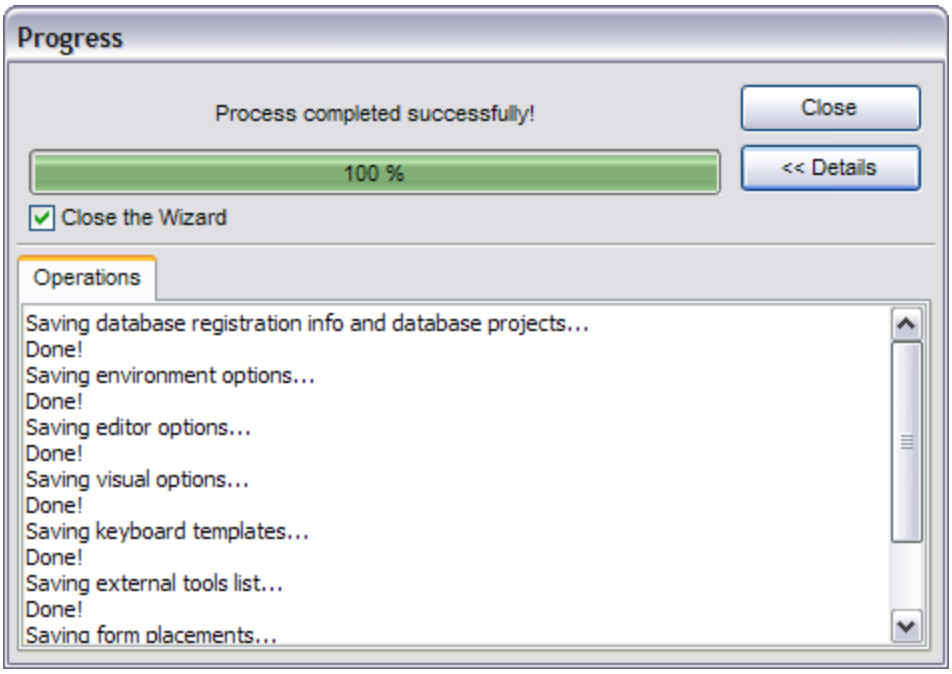
Click the **Finish** button to run the saving settings process.

11.1.4.4 Saving settings

After the saving settings operation has been configured, you can immediately start the process.

If all the settings are saved correctly, you will get the following message:

© 2008 EMS Database Management Solutions



If necessary, you can click **Details** to display/hide extended information about the operation.

© 2008 EMS Database Management Solutions

11.2 Localization

When using **SQL Manager 2008 for SQL Server**, your are provided with multi-language interface support. You can change the program language, specify the directories for your localization files easily, edit existing localizations and create your own localization files.

Changing program language

In order to select the program interface language:
- select the **Options | Select Program Language...** main menu item;
- select the interface language in the Select Language dialog;
- click **OK** to apply the language and close the dialog.

Editing program localization

In order to edit the interface localization:
- open one of the program windows (e.g. Table Editor, SQL Editor) where you wish to edit the localization of captions and hints;
- use the *Shift+Ctrl+L* keyboard shortcut to open the Localization Editor window;
- edit window captions and hints as necessary;
- click the **Save** button on the toolbar.

Note that the Localization Editor window is only available if the currently selected language is different from the default.

Creating new localization files

In order to create a new localization file:
- create a new localization file similar to those located in the *%program_directory%\Languages* folder;
- select the **Options | Environment Options** main menu item;
- proceed to the Localization section of the **Environment Options** dialog;
- click the **Add** button;
- set the language name and the path to the new **.lng* file within the Language Info Editor dialog.

The new language is added to the list of available languages. Now you can set it as the interface language using the Select Program Language dialog or the Localization section of the Environment Options dialog.

See also:

SQL Manager Options
Keyboard Templates
Find Option dialog

© 2008 EMS Database Management Solutions

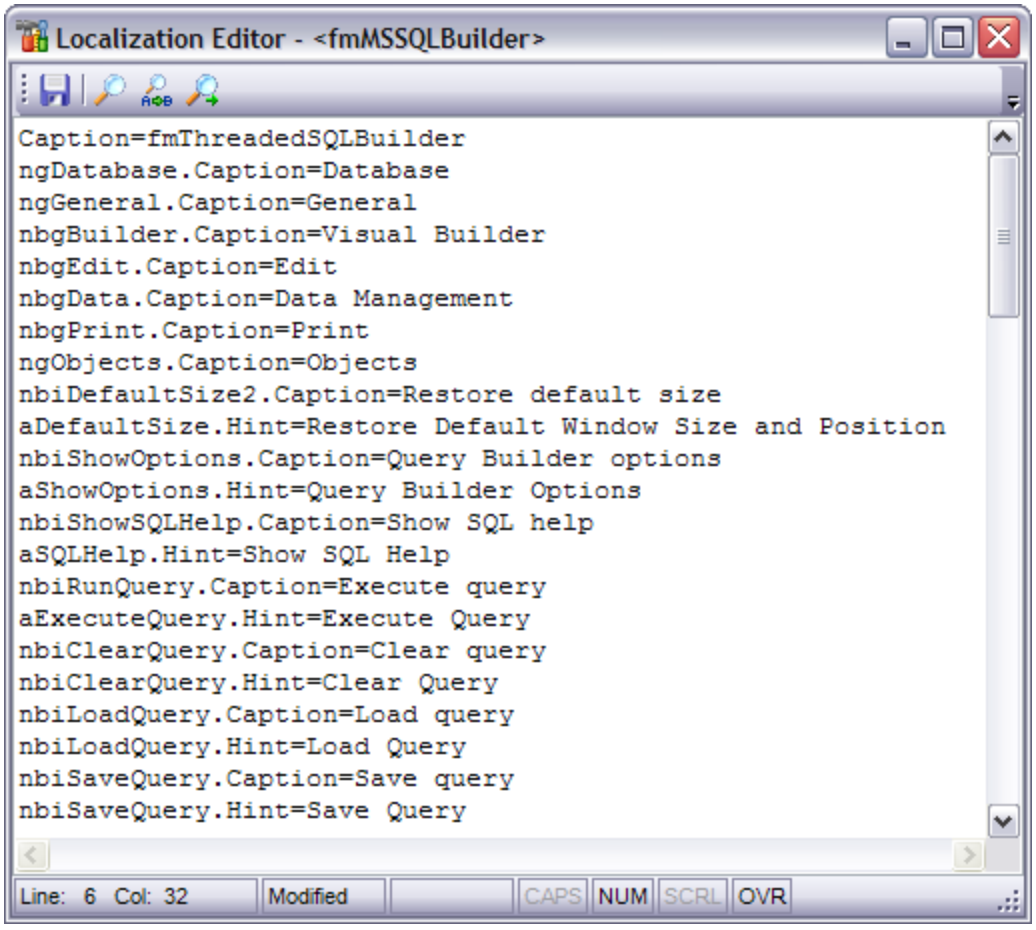
11.2.1 Localization Editor

The **Localization Editor** window allows you to edit the captions and hints of any SQL Manager window, if the selected program language is different from the default one.

To call this window, use the *Shift+Ctrl+L* shortcut in any child window of SQL Manager 2008 for SQL Server.

The working area of the window contains the element names and the corresponding strings divided by the "**=**" character. These strings are what you see in the program as menu items, window captions, button hints, etc. Edit them to change the program appearance. Be careful and do not edit the identifiers that stand before the "**=**" character - this will not produce any effect.

For your convenience the **Find** and **Replace** features are provided - the corresponding toolbar buttons are used to call the Find Text dialog or the Replace Text dialog respectively. The **Search Again** button enables the repeated search for the text that was last searched.



When you are done with editing, click the **Save** button on the toolbar to apply the changes you have made.

© 2008 EMS Database Management Solutions

Availability:
Full version (for Windows) **Yes**
Lite version (for Windows) **Yes**
Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

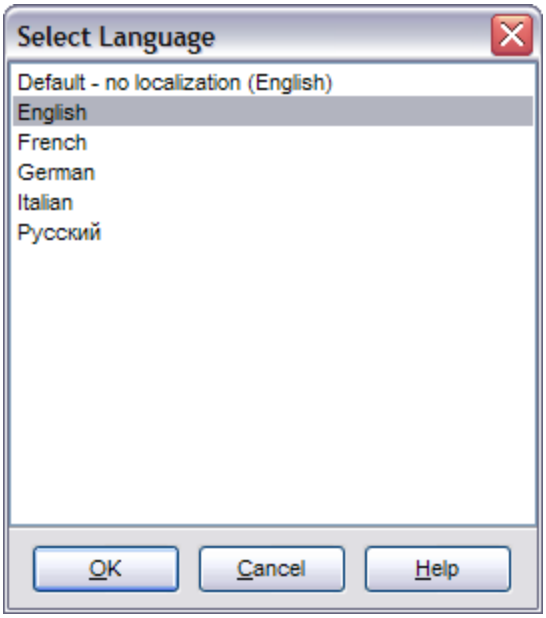
See also:
Select Program Language
Localization

11.2.2 Select Program Language

The **Select Language** dialog allows you to select a language for SQL Manager 2008 for SQL Server localization.

To open this dialog, select the **Options | Select Program Language...** main menu item.

The dialog displays the list of available languages configured on the Environment Options | Localization page. Select a language from the list and click **OK** to confirm your choice and close the dialog.



Availability:
Full version (for Windows) **Yes**

© 2008 EMS Database Management Solutions

Lite version (for Windows)	**Yes**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

Localization Editor

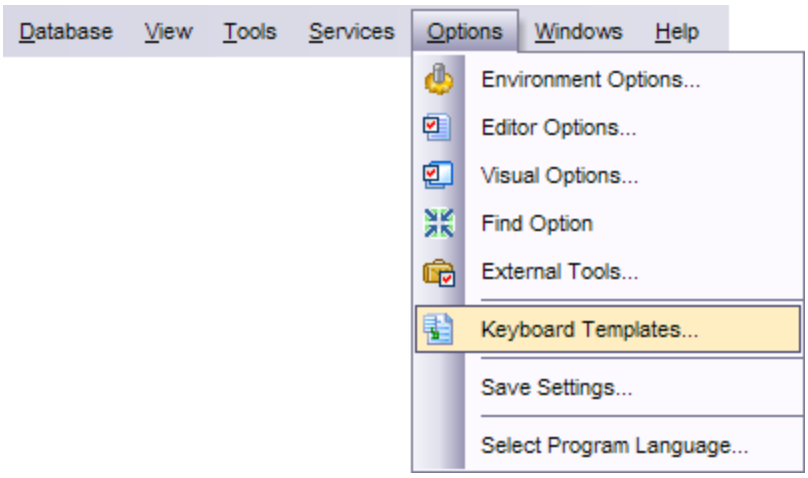
Localization

© 2008 EMS Database Management Solutions

11.3 Keyboard Templates

The **Keyboard Templates** window allows you to create new keyboard templates for quicker typing regularly used expressions and to edit the existing ones.

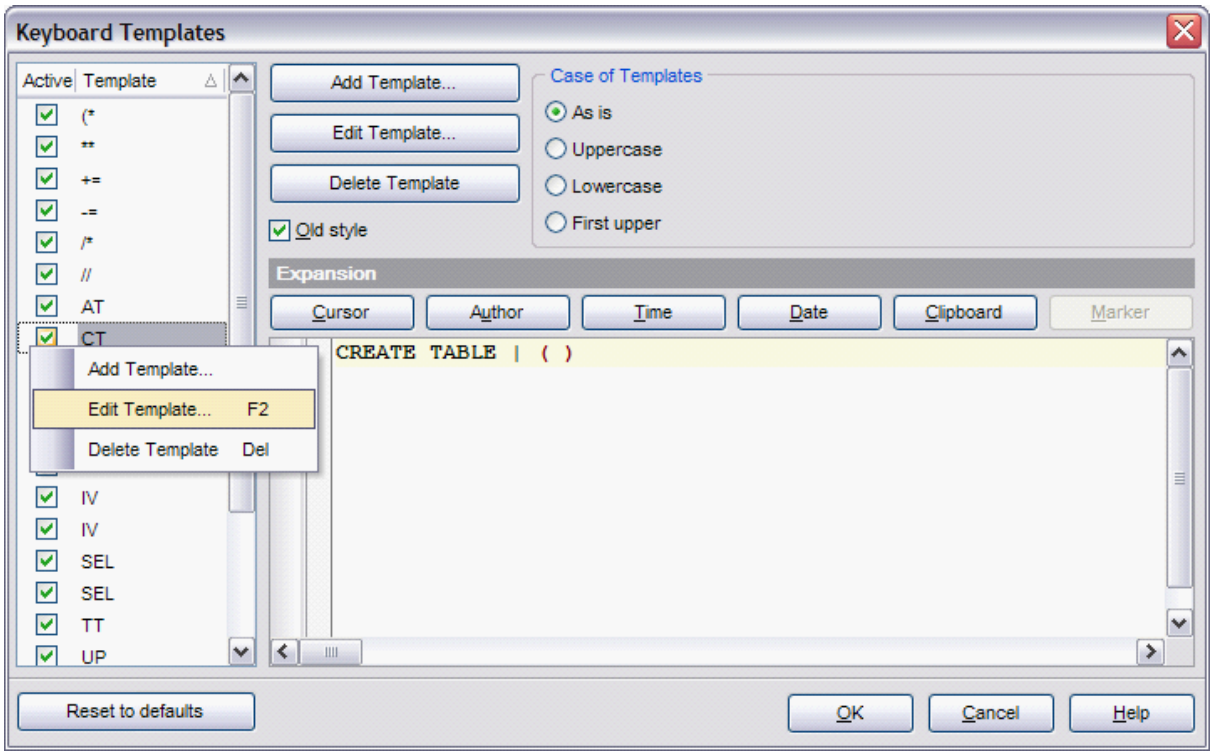
To open this window, select the **Options | Keyboard Templates...** main menu item.



To add a new keyboard template, click the **Add Template...** button, set the template name and define the template expression. In the upper right area of the window you can change the **case** of the template expression.

You can deactivate an existing template by selecting it from the list on the left and removing the **Active** flag of the template.

© 2008 EMS Database Management Solutions



If necessary, you can also edit the template name using the **Edit Template...** button, delete the template using the **Delete Template** button or edit the template expression within the **Expansion** area of the window. For faster editing you can use the *Cursor*, *Author*, *Time*, *Date*, *Clipboard*, *Marker* buttons.

Once you have defined the templates, you can use them in SQL Editor. First of all, make sure that the **Auto launch keyboard templates** option is selected on the Quick Code page of the Editor Options dialog. When editing SQL text in SQL Editor, type a template name and use the *Ctrl+J* shortcut: the text associated with the template (**Expansion**) will be inserted automatically.

Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**Yes**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

SQL Manager Options

Localization

© 2008 EMS Database Management Solutions

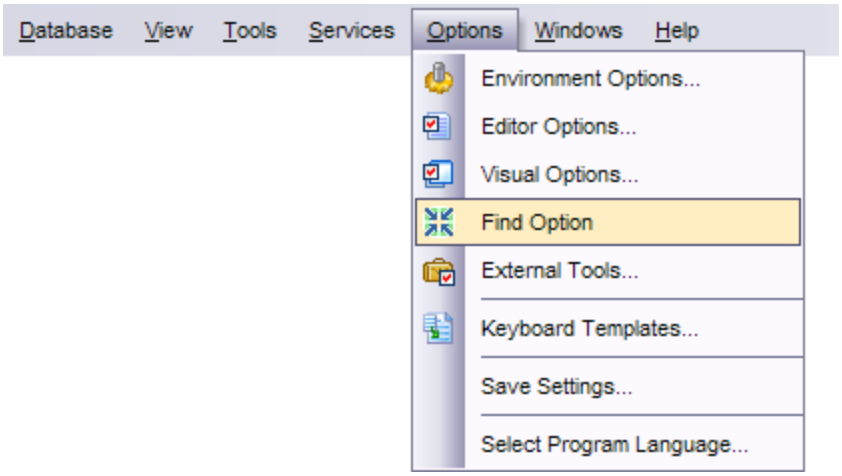
Find Option dialog

© 2008 EMS Database Management Solutions

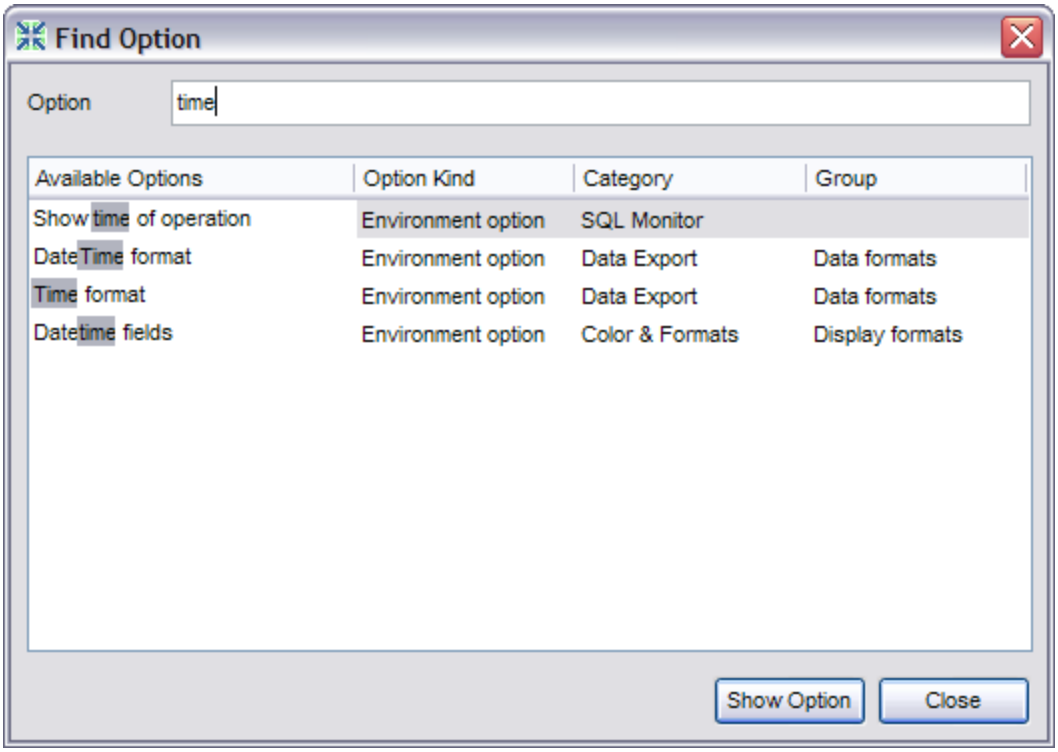
11.4 Find Option dialog

The **Find Option** dialog allows you to search for SQL Manager options easily.

To open this dialog, select the **Options | Find Option** main menu item.



Option
In this field you can enter the name of the option to search for within the specified category.



The **Available options** area lists all options of the specified category according to the

© 2008 EMS Database Management Solutions

specified name. Select the required option in the list and click **Show option** to open the corresponding dialog where you can view/edit the value of this option.

Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**Yes**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

SQL Manager Options
Localization
Keyboard Templates

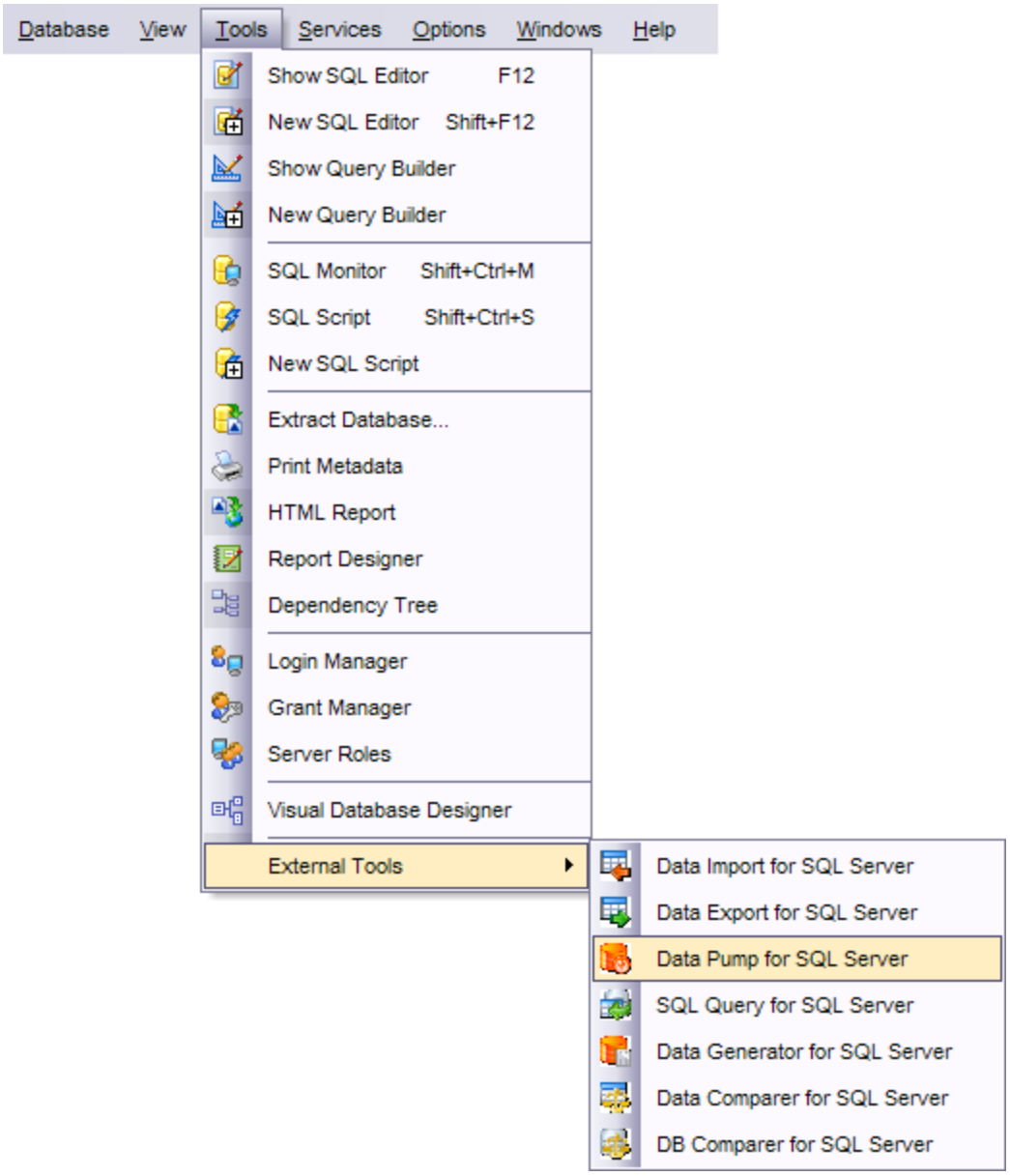
© 2008 EMS Database Management Solutions

Part

12 External tools

When using SQL Manager 2008 for SQL Server, you can add **external Windows applications** to make your work more efficient.

- External Tools dialog
- External Tool Info editor



Adding External Tools

In order to add an external program:

- select the **Options | External Tools...** main menu item;

© 2008 EMS Database Management Solutions

- click the **Add...** button in the External Tools dialog;
- specify parameters of the new external tool within the External Tool Info editor;
- confirm adding the new external tool by clicking **OK** in the External Tool Info editor and the External Tools dialog.

This adds the icon and the title of the application you have selected to the **Tools | External Tools** submenu. Now you can run this tool quickly without closing SQL Manager.

Removing External Tools

In order to remove an external program:

- select the **Options | External Tools...** main menu item;
- select the tool to be removed in the **Tools** list of the External Tools dialog;
- press the **Del** key or click the **Delete** button within the dialog;
- click **OK** to confirm removing the tool and closing the dialog.

The selected tool has been removed and is no longer accessible from the **Tools | External Tools** submenu.

See also:

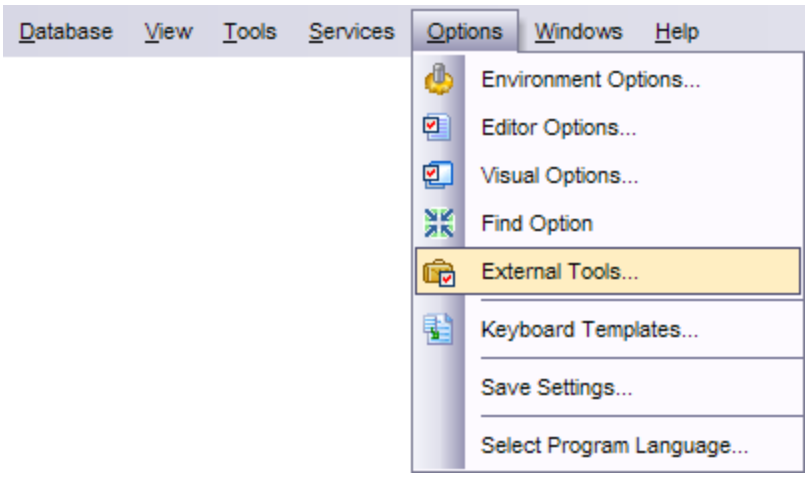
Getting started
Database Explorer
Database management
Database objects management
Query management tools
Data management
Import/Export tools
Database tools
Server tools
Personalization

© 2008 EMS Database Management Solutions

12.1 External Tools dialog

The **External Tools** dialog allows you to manage the list of external applications which can be easily run from SQL Manager environment.

To open this dialog, select the **Options | External Tools...** main menu item.



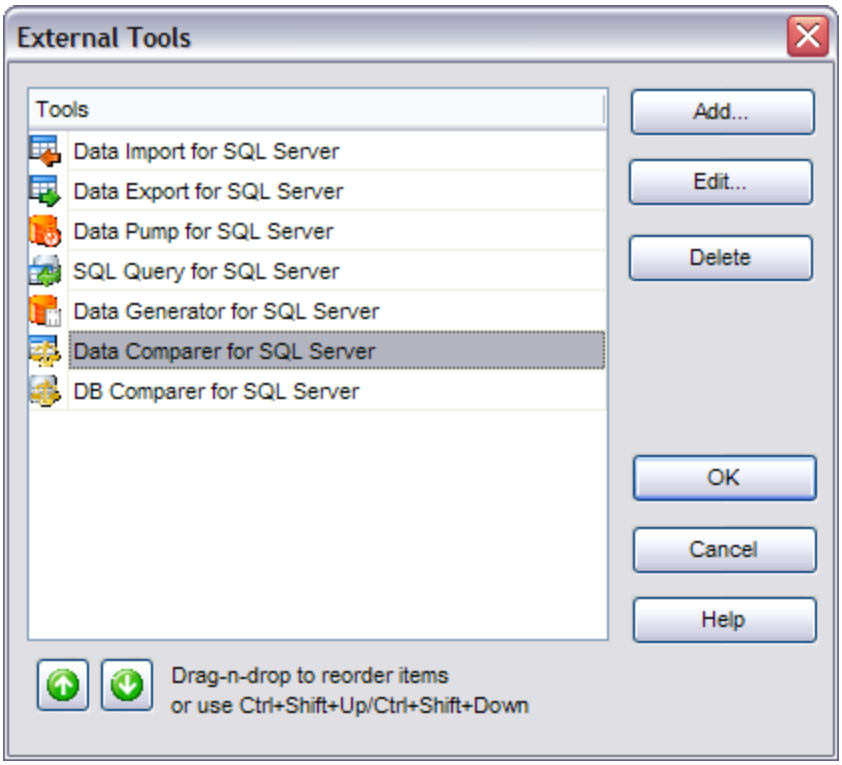
Tools
Lists all added external applications.

Add...
Opens the External Tool Info editor for adding a new tool to the **Tools | External Tools** submenu.

Edit...
Opens the External Tool Info editor for editing the title, the hot key, the path to the executable file, the working directory and execution parameters of the tool currently selected in the **Tools** list.

Delete
Removes the selected tool from the list of SQL Manager 2008 for SQL Server external tools.

© 2008 EMS Database Management Solutions



To change the order of tools in the list, use the arrow buttons at the bottom area of the dialog, or use the *Ctrl+Shift+Up* | *Ctrl+Shift+Down* shortcuts. You can also drag-and-drop items within the list box to change their positions.

Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**Yes**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

External Tool Info editor

© 2008 EMS Database Management Solutions

12.2 External Tool Info editor

The **External Tool Info** editor allows you to set common parameters of running added external programs from SQL Manager environment. This dialog is used both when adding external tools and editing their parameters (see Add External Tool and External Tools).

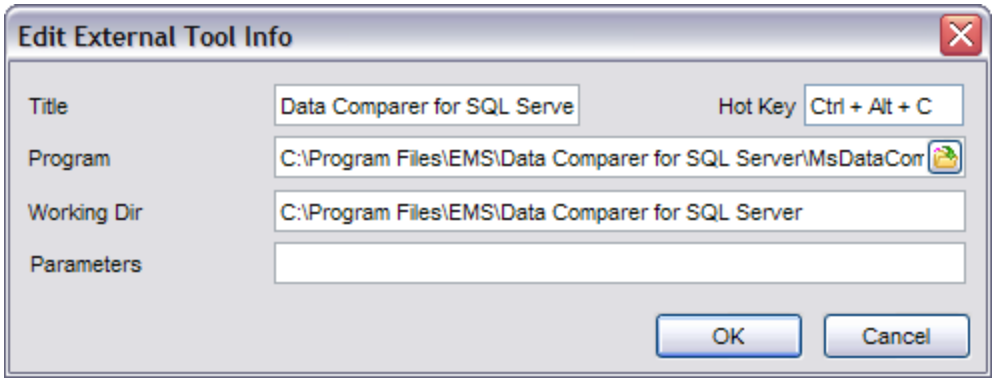
Title
Enter the title to be displayed in the **Tools | External Tools** submenu of SQL Manager.

Hot Key
Press a key or a key combination to set it as a hot key for running the tool.

Program
Use the **Explorer** button to specify the path to the **.exe* file of the external program.

Working Dir
Set the default working directory of the program.

Parameters
This box stores parameters for the program execution (if required).



Availability:
Full version (for Windows) **Yes**
Lite version (for Windows) **Yes**
Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:
External Tools dialog

© 2008 EMS Database Management Solutions

Part XIII

13 How to...

The succeeding pages of this chapter are intended to provide you with brief instructions on how to perform this or that operation correctly using SQL Manager 2008 for SQL Server .

Please see the chapters below to learn how to...

- Work with Databases
- Work with Database Objects
- Manage Server security

See also:

Getting started
Database Explorer
Database management
Database objects management
Query management tools
Data management
Import/Export tools
Database tools
Server tools
Personalization
External tools

© 2008 EMS Database Management Solutions

13.1 Work with databases

The following chapters inform you how to...

13.1.1 Create Database

SQL Manager 2008 for SQL Server provides a number of tools you may need to manage your SQL Server databases.
Find the list of common database management operations for working in SQL Manager below.

Registering Hosts

In order to register a Host in SQL Manager 2008 for SQL Server:
- select the **Database | Register Host...** main menu item or use the corresponding **Register Host** toolbar button

or
- right-click the Host and select the **Register Host...** context menu item in the DB Explorer tree;
- set all the necessary options using Register Host wizard which guides you through the entire process of Host registration.

Unregistering Hosts

In order to unregister a Host in SQL Manager 2008 for SQL Server:
- select the Host to unregister in the DB Explorer tree;
- select the **Database | Unregister Host** main menu item or use the corresponding **Unregister Host** toolbar button

or
- right-click the Host and select the **Unregister Host** context menu item in the DB Explorer tree;
- confirm unregistering in the corresponding dialog window.

Viewing and editing Server Properties

In order to view/edit Server properties in SQL Manager 2008 for SQL Server:
- select the host or a database that resides on this host in the DB Explorer tree;
- right-click the host or the database alias and select the **Server Properties** context menu item.

Creating Databases

In order to create a Database in SQL Manager 2008 for SQL Server:
- select the **Database | Create Database** main menu item or use the corresponding **Create Database** toolbar button;
- set all the necessary options using Create Database wizard which guides you through the entire process of creating a new database.

Dropping Databases

In order to drop a Database in SQL Manager 2008 for SQL Server:
- select the Database to drop in the DB Explorer tree;

© 2008 EMS Database Management Solutions

- select the **Database | Drop Database** main menu item;
- confirm dropping in the corresponding dialog window.

Registering Databases

In order to register a single Database in SQL Manager 2008 for SQL Server:
- select the **Database | Register Database...** main menu item or use the corresponding **Register Database** toolbar button

or
- right-click the Database and select the **Register Database...** context menu item in the DB Explorer tree;
- set all the necessary options using Register Database wizard which guides you through the entire process of Database registration.

Unregistering Databases

In order to unregister a Database in SQL Manager 2008 for SQL Server:
- select the Database to unregister in the DB Explorer tree;
- select the **Database | Unregister Database** main menu item or use the corresponding **Unregister Database** toolbar button

or
- right-click the Database and select the **Unregister Database** context menu item in the DB Explorer tree;
- confirm unregistering in the corresponding dialog window.

Connecting to Databases

In order to connect to a Database in SQL Manager 2008 for SQL Server:
- select the Database to connect to in the DB Explorer tree;
- select the **Database | Connect to Database** main menu item or use the corresponding **Connect to Database** toolbar button

or
- right-click the Database and select the **Connect to Database** context menu item in the DB Explorer tree.

Disconnecting from Databases

In order to disconnect from a Database in SQL Manager 2008 for SQL Server:
- select the Database to disconnect from in the DB Explorer tree;
- select the **Database | Disconnect from Database** main menu item or use the corresponding **Disconnect from Database** toolbar button

or
- right-click the Database and select the **Disconnect from Database** context menu item in the DB Explorer tree.

Viewing and editing Database Registration Info

In order to view/edit Database Registration Info in SQL Manager 2008 for SQL Server:
- select the Database or any of its objects in the DB Explorer tree;
- select the **Database | Database Registration Info...** main menu item

or
- right-click the Database or any of its objects and select the **Database Registration**

© 2008 EMS Database Management Solutions

Info... context menu item in the DB Explorer tree.

Viewing and editing Database Properties

In order to view/edit Database properties in SQL Manager 2008 for SQL Server:
- select the Database in the DB Explorer tree;
- right-click the Database and select the **Database Properties...** context menu item or use the corresponding **Properties** toolbar button

Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**Yes**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:
Getting started
Database Explorer
Database objects management
Query management tools
Data management
Import/Export tools
Database tools
Server tools
Personalization
External tools

13.1.2 Drop Database

SQL Manager 2008 for SQL Server provides a number of tools you may need to manage your SQL Server databases.
Find the list of common database management operations for working in SQL Manager below.

Registering Hosts

In order to register a Host in SQL Manager 2008 for SQL Server:
- select the **Database | Register Host...** main menu item or use the corresponding **Register Host** toolbar button

or
- right-click the Host and select the **Register Host...** context menu item in the DB

© 2008 EMS Database Management Solutions

Explorer tree;
- set all the necessary options using Register Host wizard which guides you through the entire process of Host registration.

Unregistering Hosts

In order to unregister a Host in SQL Manager 2008 for SQL Server:
- select the Host to unregister in the DB Explorer tree;
- select the **Database | Unregister Host** main menu item or use the corresponding **Unregister Host** toolbar button

or
- right-click the Host and select the **Unregister Host** context menu item in the DB Explorer tree;
- confirm unregistering in the corresponding dialog window.

Viewing and editing Server Properties

In order to view/edit Server properties in SQL Manager 2008 for SQL Server:
- select the host or a database that resides on this host in the DB Explorer tree;
- right-click the host or the database alias and select the **Server Properties** context menu item.

Creating Databases

In order to create a Database in SQL Manager 2008 for SQL Server:
- select the **Database | Create Database** main menu item or use the corresponding **Create Database** toolbar button;
- set all the necessary options using Create Database wizard which guides you through the entire process of creating a new database.

Dropping Databases

In order to drop a Database in SQL Manager 2008 for SQL Server:
- select the Database to drop in the DB Explorer tree;
- select the **Database | Drop Database** main menu item;
- confirm dropping in the corresponding dialog window.

Registering Databases

In order to register a single Database in SQL Manager 2008 for SQL Server:
- select the **Database | Register Database...** main menu item or use the corresponding **Register Database** toolbar button

or
- right-click the Database and select the **Register Database...** context menu item in the DB Explorer tree;
- set all the necessary options using Register Database wizard which guides you through the entire process of Database registration.

Unregistering Databases

In order to unregister a Database in SQL Manager 2008 for SQL Server:
- select the Database to unregister in the DB Explorer tree;

© 2008 EMS Database Management Solutions

- select the **Database | Unregister Database** main menu item or use the corresponding **Unregister Database** toolbar button

or

- right-click the Database and select the **Unregister Database** context menu item in the DB Explorer tree;
- confirm unregistering in the corresponding dialog window.

Connecting to Databases

In order to connect to a Database in SQL Manager 2008 for SQL Server:

- select the Database to connect to in the DB Explorer tree;
- select the **Database | Connect to Database** main menu item or use the corresponding **Connect to Database** toolbar button

or

- right-click the Database and select the **Connect to Database** context menu item in the DB Explorer tree.

Disconnecting from Databases

In order to disconnect from a Database in SQL Manager 2008 for SQL Server:

- select the Database to disconnect from in the DB Explorer tree;
- select the **Database | Disconnect from Database** main menu item or use the corresponding **Disconnect from Database** toolbar button

or

- right-click the Database and select the **Disconnect from Database** context menu item in the DB Explorer tree.

Viewing and editing Database Registration Info

In order to view/edit Database Registration Info in SQL Manager 2008 for SQL Server:

- select the Database or any of its objects in the DB Explorer tree;
- select the **Database | Database Registration Info...** main menu item

or

- right-click the Database or any of its objects and select the **Database Registration Info...** context menu item in the DB Explorer tree.

Viewing and editing Database Properties

In order to view/edit Database properties in SQL Manager 2008 for SQL Server:

- select the Database in the DB Explorer tree;
- right-click the Database and select the **Database Properties...** context menu item or use the corresponding **Properties** toolbar button

Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**Yes**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

© 2008 EMS Database Management Solutions

See also:
Getting started
Database Explorer
Database objects management
Query management tools
Data management
Import/Export tools
Database tools
Server tools
Personalization
External tools

13.1.3 Register Database

SQL Manager 2008 for SQL Server provides a number of tools you may need to manage your SQL Server databases.
Find the list of common database management operations for working in SQL Manager below.

Registering Hosts

In order to register a Host in SQL Manager 2008 for SQL Server:
- select the **Database | Register Host...** main menu item or use the corresponding **Register Host** toolbar button

or
- right-click the Host and select the **Register Host...** context menu item in the DB Explorer tree;
- set all the necessary options using Register Host wizard which guides you through the entire process of Host registration.

Unregistering Hosts

In order to unregister a Host in SQL Manager 2008 for SQL Server:
- select the Host to unregister in the DB Explorer tree;
- select the **Database | Unregister Host** main menu item or use the corresponding **Unregister Host** toolbar button

or
- right-click the Host and select the **Unregister Host** context menu item in the DB Explorer tree;
- confirm unregistering in the corresponding dialog window.

Viewing and editing Server Properties

In order to view/edit Server properties in SQL Manager 2008 for SQL Server:

© 2008 EMS Database Management Solutions

- select the host or a database that resides on this host in the DB Explorer tree;
- right-click the host or the database alias and select the **Server Properties** context menu item.

Creating Databases

In order to create a Database in SQL Manager 2008 for SQL Server:

- select the **Database | Create Database** main menu item or use the corresponding **Create Database** toolbar button;
- set all the necessary options using Create Database wizard which guides you through the entire process of creating a new database.

Dropping Databases

In order to drop a Database in SQL Manager 2008 for SQL Server:

- select the Database to drop in the DB Explorer tree;
- select the **Database | Drop Database** main menu item;
- confirm dropping in the corresponding dialog window.

Registering Databases

In order to register a single Database in SQL Manager 2008 for SQL Server:

- select the **Database | Register Database...** main menu item or use the corresponding **Register Database** toolbar button

or

- right-click the Database and select the **Register Database...** context menu item in the DB Explorer tree;
- set all the necessary options using Register Database wizard which guides you through the entire process of Database registration.

Unregistering Databases

In order to unregister a Database in SQL Manager 2008 for SQL Server:

- select the Database to unregister in the DB Explorer tree;
- select the **Database | Unregister Database** main menu item or use the corresponding **Unregister Database** toolbar button

or

- right-click the Database and select the **Unregister Database** context menu item in the DB Explorer tree;
- confirm unregistering in the corresponding dialog window.

Connecting to Databases

In order to connect to a Database in SQL Manager 2008 for SQL Server:

- select the Database to connect to in the DB Explorer tree;
- select the **Database | Connect to Database** main menu item or use the corresponding **Connect to Database** toolbar button

or

- right-click the Database and select the **Connect to Database** context menu item in the DB Explorer tree.

Disconnecting from Databases

© 2008 EMS Database Management Solutions

In order to disconnect from a Database in SQL Manager 2008 for SQL Server:
- select the Database to disconnect from in the DB Explorer tree;
- select the **Database | Disconnect from Database** main menu item or use the corresponding **Disconnect from Database** toolbar button

or
- right-click the Database and select the **Disconnect from Database** context menu item in the DB Explorer tree.

Viewing and editing Database Registration Info

In order to view/edit Database Registration Info in SQL Manager 2008 for SQL Server:
- select the Database or any of its objects in the DB Explorer tree;
- select the **Database | Database Registration Info...** main menu item

or
- right-click the Database or any of its objects and select the **Database Registration Info...** context menu item in the DB Explorer tree.

Viewing and editing Database Properties

In order to view/edit Database properties in SQL Manager 2008 for SQL Server:
- select the Database in the DB Explorer tree;
- right-click the Database and select the **Database Properties...** context menu item or use the corresponding **Properties** toolbar button

Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**Yes**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

Getting started
Database Explorer
Database objects management
Query management tools
Data management
Import/Export tools
Database tools
Server tools
Personalization
External tools

© 2008 EMS Database Management Solutions

13.1.4 Unregister Database

SQL Manager 2008 for SQL Server provides a number of tools you may need to manage your SQL Server databases.
Find the list of common database management operations for working in SQL Manager below.

Registering Hosts

In order to register a Host in SQL Manager 2008 for SQL Server:
- select the **Database | Register Host...** main menu item or use the corresponding **Register Host** toolbar button

or
- right-click the Host and select the **Register Host...** context menu item in the DB Explorer tree;
- set all the necessary options using Register Host wizard which guides you through the entire process of Host registration.

Unregistering Hosts

In order to unregister a Host in SQL Manager 2008 for SQL Server:
- select the Host to unregister in the DB Explorer tree;
- select the **Database | Unregister Host** main menu item or use the corresponding **Unregister Host** toolbar button

or
- right-click the Host and select the **Unregister Host** context menu item in the DB Explorer tree;
- confirm unregistering in the corresponding dialog window.

Viewing and editing Server Properties

In order to view/edit Server properties in SQL Manager 2008 for SQL Server:
- select the host or a database that resides on this host in the DB Explorer tree;
- right-click the host or the database alias and select the **Server Properties** context menu item.

Creating Databases

In order to create a Database in SQL Manager 2008 for SQL Server:
- select the **Database | Create Database** main menu item or use the corresponding **Create Database** toolbar button;
- set all the necessary options using Create Database wizard which guides you through the entire process of creating a new database.

Dropping Databases

In order to drop a Database in SQL Manager 2008 for SQL Server:
- select the Database to drop in the DB Explorer tree;
- select the **Database | Drop Database** main menu item;
- confirm dropping in the corresponding dialog window.

© 2008 EMS Database Management Solutions

Registering Databases

In order to register a single Database in SQL Manager 2008 for SQL Server:
- select the **Database | Register Database...** main menu item or use the corresponding **Register Database** toolbar button

or
- right-click the Database and select the **Register Database...** context menu item in the DB Explorer tree;
- set all the necessary options using Register Database wizard which guides you through the entire process of Database registration.

Unregistering Databases

In order to unregister a Database in SQL Manager 2008 for SQL Server:
- select the Database to unregister in the DB Explorer tree;
- select the **Database | Unregister Database** main menu item or use the corresponding **Unregister Database** toolbar button

or
- right-click the Database and select the **Unregister Database** context menu item in the DB Explorer tree;
- confirm unregistering in the corresponding dialog window.

Connecting to Databases

In order to connect to a Database in SQL Manager 2008 for SQL Server:
- select the Database to connect to in the DB Explorer tree;
- select the **Database | Connect to Database** main menu item or use the corresponding **Connect to Database** toolbar button

or
- right-click the Database and select the **Connect to Database** context menu item in the DB Explorer tree.

Disconnecting from Databases

In order to disconnect from a Database in SQL Manager 2008 for SQL Server:
- select the Database to disconnect from in the DB Explorer tree;
- select the **Database | Disconnect from Database** main menu item or use the corresponding **Disconnect from Database** toolbar button

or
- right-click the Database and select the **Disconnect from Database** context menu item in the DB Explorer tree.

Viewing and editing Database Registration Info

In order to view/edit Database Registration Info in SQL Manager 2008 for SQL Server:
- select the Database or any of its objects in the DB Explorer tree;
- select the **Database | Database Registration Info...** main menu item

or
- right-click the Database or any of its objects and select the **Database Registration Info...** context menu item in the DB Explorer tree.

© 2008 EMS Database Management Solutions

Viewing and editing Database Properties

In order to view/edit Database properties in SQL Manager 2008 for SQL Server:
- select the Database in the DB Explorer tree;
- right-click the Database and select the **Database Properties...** context menu item or use the corresponding **Properties** toolbar button

Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**Yes**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

Getting started
Database Explorer
Database objects management
Query management tools
Data management
Import/Export tools
Database tools
Server tools
Personalization
External tools

13.1.5 Connect to Database

SQL Manager 2008 for SQL Server provides a number of tools you may need to manage your SQL Server databases.
Find the list of common database management operations for working in SQL Manager below.

Registering Hosts

In order to register a Host in SQL Manager 2008 for SQL Server:
- select the **Database | Register Host...** main menu item or use the corresponding **Register Host** toolbar button

or
- right-click the Host and select the **Register Host...** context menu item in the DB Explorer tree;
- set all the necessary options using Register Host wizard which guides you through the

© 2008 EMS Database Management Solutions

entire process of Host registration.

Unregistering Hosts

In order to unregister a Host in SQL Manager 2008 for SQL Server:
- select the Host to unregister in the DB Explorer tree;
- select the **Database | Unregister Host** main menu item or use the corresponding **Unregister Host** toolbar button

or
- right-click the Host and select the **Unregister Host** context menu item in the DB Explorer tree;
- confirm unregistering in the corresponding dialog window.

Viewing and editing Server Properties

In order to view/edit Server properties in SQL Manager 2008 for SQL Server:
- select the host or a database that resides on this host in the DB Explorer tree;
- right-click the host or the database alias and select the **Server Properties** context menu item.

Creating Databases

In order to create a Database in SQL Manager 2008 for SQL Server:
- select the **Database | Create Database** main menu item or use the corresponding **Create Database** toolbar button;
- set all the necessary options using Create Database wizard which guides you through the entire process of creating a new database.

Dropping Databases

In order to drop a Database in SQL Manager 2008 for SQL Server:
- select the Database to drop in the DB Explorer tree;
- select the **Database | Drop Database** main menu item;
- confirm dropping in the corresponding dialog window.

Registering Databases

In order to register a single Database in SQL Manager 2008 for SQL Server:
- select the **Database | Register Database...** main menu item or use the corresponding **Register Database** toolbar button

or
- right-click the Database and select the **Register Database...** context menu item in the DB Explorer tree;
- set all the necessary options using Register Database wizard which guides you through the entire process of Database registration.

Unregistering Databases

In order to unregister a Database in SQL Manager 2008 for SQL Server:
- select the Database to unregister in the DB Explorer tree;
- select the **Database | Unregister Database** main menu item or use the corresponding **Unregister Database** toolbar button

© 2008 EMS Database Management Solutions

or
- right-click the Database and select the **Unregister Database** context menu item in the DB Explorer tree;
- confirm unregistering in the corresponding dialog window.

Connecting to Databases

In order to connect to a Database in SQL Manager 2008 for SQL Server:
- select the Database to connect to in the DB Explorer tree;
- select the **Database | Connect to Database** main menu item or use the corresponding **Connect to Database** toolbar button

or
- right-click the Database and select the **Connect to Database** context menu item in the DB Explorer tree.

Disconnecting from Databases

In order to disconnect from a Database in SQL Manager 2008 for SQL Server:
- select the Database to disconnect from in the DB Explorer tree;
- select the **Database | Disconnect from Database** main menu item or use the corresponding **Disconnect from Database** toolbar button

or
- right-click the Database and select the **Disconnect from Database** context menu item in the DB Explorer tree.

Viewing and editing Database Registration Info

In order to view/edit Database Registration Info in SQL Manager 2008 for SQL Server:
- select the Database or any of its objects in the DB Explorer tree;
- select the **Database | Database Registration Info...** main menu item

or
- right-click the Database or any of its objects and select the **Database Registration Info...** context menu item in the DB Explorer tree.

Viewing and editing Database Properties

In order to view/edit Database properties in SQL Manager 2008 for SQL Server:
- select the Database in the DB Explorer tree;
- right-click the Database and select the **Database Properties...** context menu item or use the corresponding **Properties** toolbar button

Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**Yes**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

© 2008 EMS Database Management Solutions

See also:
Getting started
Database Explorer
Database objects management
Query management tools
Data management
Import/Export tools
Database tools
Server tools
Personalization
External tools

13.1.6 Disconnect from Database

SQL Manager 2008 for SQL Server provides a number of tools you may need to manage your SQL Server databases.
Find the list of common database management operations for working in SQL Manager below.

Registering Hosts

In order to register a Host in SQL Manager 2008 for SQL Server:
- select the **Database | Register Host...** main menu item or use the corresponding **Register Host** toolbar button

or
- right-click the Host and select the **Register Host...** context menu item in the DB Explorer tree;
- set all the necessary options using Register Host wizard which guides you through the entire process of Host registration.

Unregistering Hosts

In order to unregister a Host in SQL Manager 2008 for SQL Server:
- select the Host to unregister in the DB Explorer tree;
- select the **Database | Unregister Host** main menu item or use the corresponding **Unregister Host** toolbar button

or
- right-click the Host and select the **Unregister Host** context menu item in the DB Explorer tree;
- confirm unregistering in the corresponding dialog window.

Viewing and editing Server Properties

In order to view/edit Server properties in SQL Manager 2008 for SQL Server:
- select the host or a database that resides on this host in the DB Explorer tree;
- right-click the host or the database alias and select the **Server Properties** context

© 2008 EMS Database Management Solutions

menu item.

Creating Databases

In order to create a Database in SQL Manager 2008 for SQL Server:
- select the **Database | Create Database** main menu item or use the corresponding **Create Database** toolbar button;
- set all the necessary options using Create Database wizard which guides you through the entire process of creating a new database.

Dropping Databases

In order to drop a Database in SQL Manager 2008 for SQL Server:
- select the Database to drop in the DB Explorer tree;
- select the **Database | Drop Database** main menu item;
- confirm dropping in the corresponding dialog window.

Registering Databases

In order to register a single Database in SQL Manager 2008 for SQL Server:
- select the **Database | Register Database...** main menu item or use the corresponding **Register Database** toolbar button

or
- right-click the Database and select the **Register Database...** context menu item in the DB Explorer tree;
- set all the necessary options using Register Database wizard which guides you through the entire process of Database registration.

Unregistering Databases

In order to unregister a Database in SQL Manager 2008 for SQL Server:
- select the Database to unregister in the DB Explorer tree;
- select the **Database | Unregister Database** main menu item or use the corresponding **Unregister Database** toolbar button

or
- right-click the Database and select the **Unregister Database** context menu item in the DB Explorer tree;
- confirm unregistering in the corresponding dialog window.

Connecting to Databases

In order to connect to a Database in SQL Manager 2008 for SQL Server:
- select the Database to connect to in the DB Explorer tree;
- select the **Database | Connect to Database** main menu item or use the corresponding **Connect to Database** toolbar button

or
- right-click the Database and select the **Connect to Database** context menu item in the DB Explorer tree.

Disconnecting from Databases

In order to disconnect from a Database in SQL Manager 2008 for SQL Server:

© 2008 EMS Database Management Solutions

- select the Database to disconnect from in the DB Explorer tree;
- select the **Database | Disconnect from Database** main menu item or use the corresponding **Disconnect from Database** toolbar button

or

- right-click the Database and select the **Disconnect from Database** context menu item in the DB Explorer tree.

Viewing and editing Database Registration Info

In order to view/edit Database Registration Info in SQL Manager 2008 for SQL Server:

- select the Database or any of its objects in the DB Explorer tree;
- select the **Database | Database Registration Info...** main menu item

or

- right-click the Database or any of its objects and select the **Database Registration Info...** context menu item in the DB Explorer tree.

Viewing and editing Database Properties

In order to view/edit Database properties in SQL Manager 2008 for SQL Server:

- select the Database in the DB Explorer tree;
- right-click the Database and select the **Database Properties...** context menu item or use the corresponding **Properties** toolbar button

Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**Yes**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

Getting started
Database Explorer
Database objects management
Query management tools
Data management
Import/Export tools
Database tools
Server tools
Personalization
External tools

© 2008 EMS Database Management Solutions

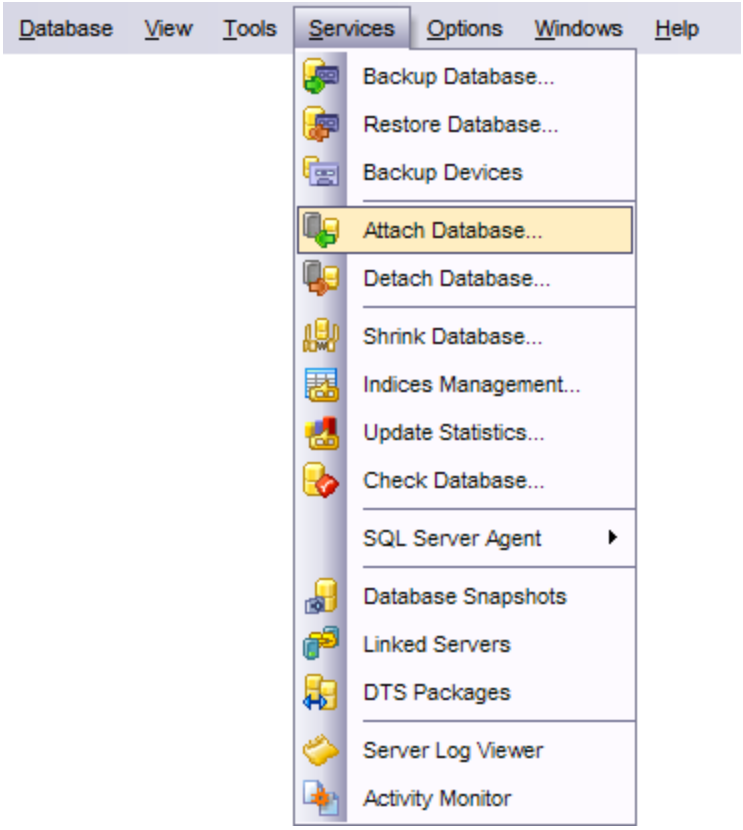
13.1.7 Attach Database

Attach Database Wizard allows you to attach an existing database that is made up of one or more files to the instance of Microsoft® SQL Server™ (with SQL Server routines used).

This operation is used to attach a database with the specified name and using the specified data files.

To run the wizard, select the **Services | Attach Database...** main menu item, or right-click the **Databases** node in the DB Explorer tree and select the **Tasks | Attach Database...** context menu item.

- Setting connection properties
- Defining database files and general options
- Attaching database
- Using templates



Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**Yes**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

© 2008 EMS Database Management Solutions

See also:
Detach Database
Backup Database
Restore Database
Shrink Database
Indices Management
Update Statistics
Check Database
Grant Manager

13.1.8 Detach Database

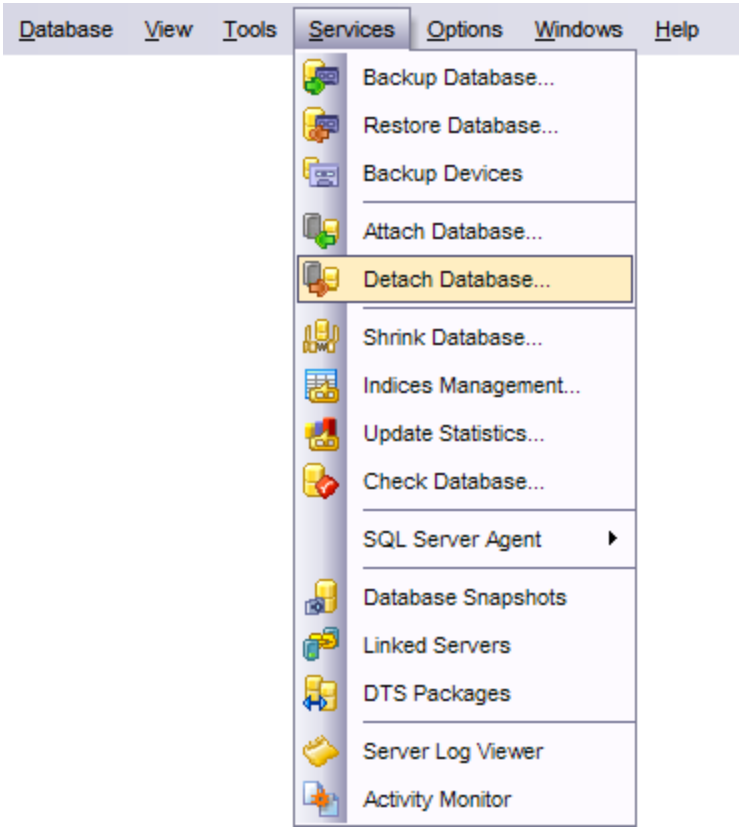
Detach Database Wizard allows you to detach a database from the instance of Microsoft® SQL Server™ (with SQL Server routines used).

This operation is used to detach the specified database with or without updating statistics before the database is detached.

To run the wizard, select the **Services | Detach Database...** main menu item, or right-click the database alias in the DB Explorer tree and select the **Tasks | Detach Database...** item of the context menu.

- Setting connection properties
- Selecting the database
- Detaching database
- Using templates

© 2008 EMS Database Management Solutions



Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**Yes**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

Attach Database
Backup Database
Restore Database
Shrink Database
Indices Management
Update Statistics
Check Database
Grant Manager

© 2008 EMS Database Management Solutions

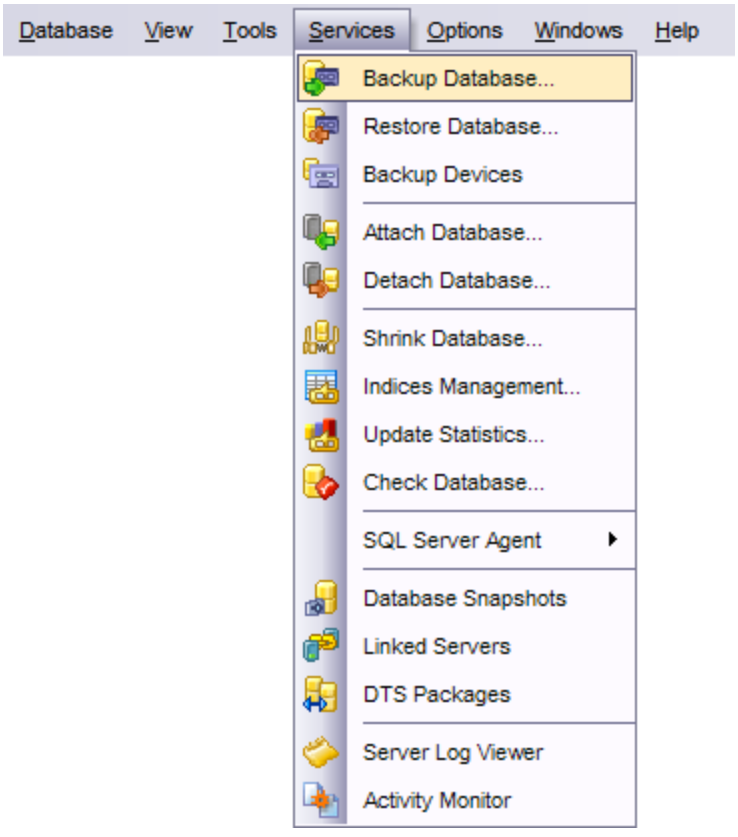
13.1.9 Backup Database

Backup Database Wizard allows you to perform the database backup operation on your SQL Server system (the *BACKUP DATABASE* Transact-SQL statement is issued).

This operation is used to create a backup copy of an entire database, transaction log or files/filegroups.

To run the wizard, select the **Services | Backup Database...** main menu item, or right-click the database alias in the DB Explorer tree and select the **Tasks | Backup Database...** item of the context menu.

- Selecting database to backup
- Setting backup type and options
- Selecting database files to backup
- Selecting backup devices
- Setting schedule options
- Setting media options
- Setting advanced backup options
- Setting transaction log backup options
- Setting data transfer options
- Running database backup
- Using templates



Availability:

© 2008 EMS Database Management Solutions

Full version (for Windows)	**Yes**
Lite version (for Windows)	**Yes**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

Attach Database
Detach Database
Restore Database
Shrink Database
Indices Management
Update Statistics
Check Database
Grant Manager

13.1.10 Restore Database

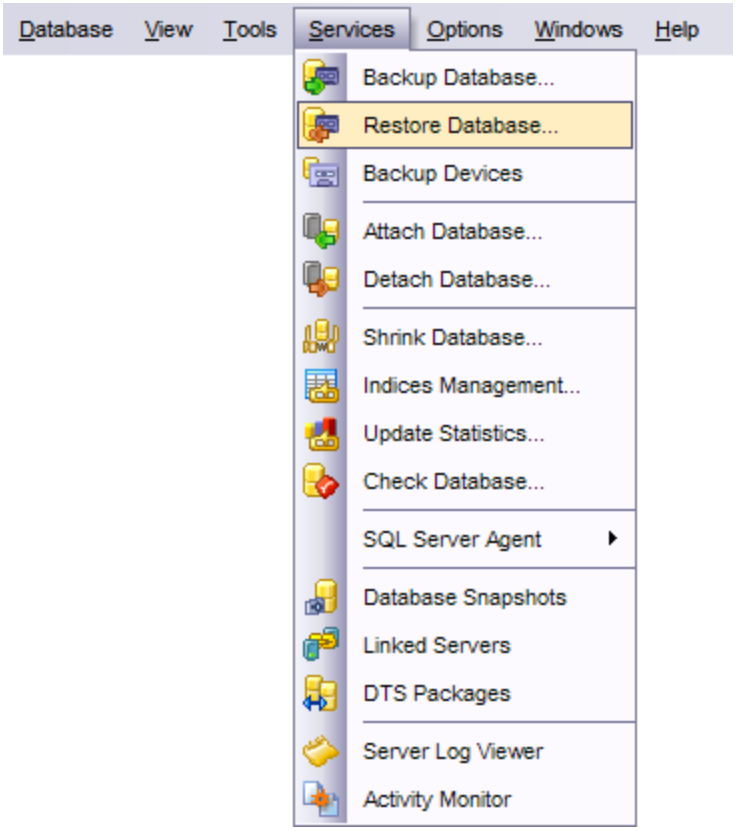
Restore Database Wizard allows you to perform the database restore operation on your SQL Server system (the *RESTORE DATABASE* Transact-SQL statement is issued).

This operation is used to rebuild a damaged or corrupted database that has been backed up using Backup Database Wizard. You can restore an entire database, part of a database, specific files and filegroups, transaction logs.

To run the wizard, select the **Services | Restore Database...** main menu item, or right-click the database alias in the DB Explorer tree and select the **Tasks | Restore Database...** item of the context menu. Alternatively, you can right-click the **Databases** node and select the **Tasks | Restore Database...** context menu item.

- Selecting destination database
- Setting restore type and recovery options
- Selecting backup set to restore
- Selecting files to restore
- Setting advanced restore options
- Running database restore
- Using templates

© 2008 EMS Database Management Solutions



Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**Yes**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

Attach Database
Detach Database
Backup Database
Shrink Database
Indices Management
Update Statistics
Check Database
Grant Manager

© 2008 EMS Database Management Solutions

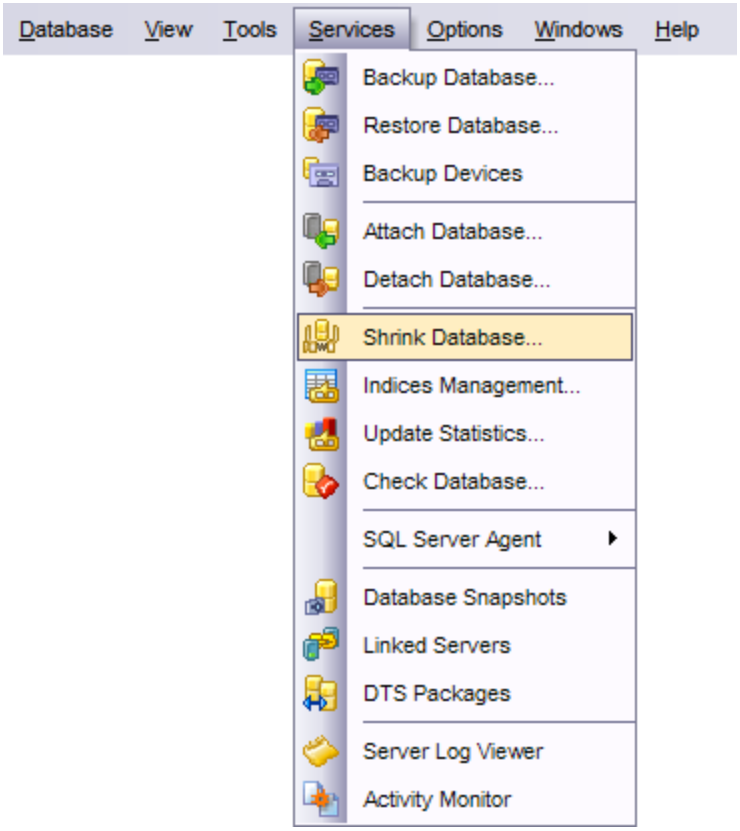
13.1.11 Shrink Database

Shrink Database Wizard allows you to perform database shrinking on your SQL Server system (the *DBCC SHRINKDATABASE* and *SHRINKFILE* Transact-SQL statements are issued).

This operation is used to reduce any file within the database to remove unused pages. Both data and transaction log files can be reduced, or shrunk.

To run the wizard, select the **Services | Shrink Database...** main menu item, or right-click the database alias in the DB Explorer tree and select the **Tasks | Shrink Database...** item of the context menu.

- Selecting database and setting shrink mode
- Defining shrink action for database
- Defining shrink action for files
- Shrinking database
- Using templates



Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**No**

© 2008 EMS Database Management Solutions

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:
Attach Database
Detach Database
Backup Database
Restore Database
Indices Management
Update Statistics
Check Database
Grant Manager

13.1.12 Manage indices

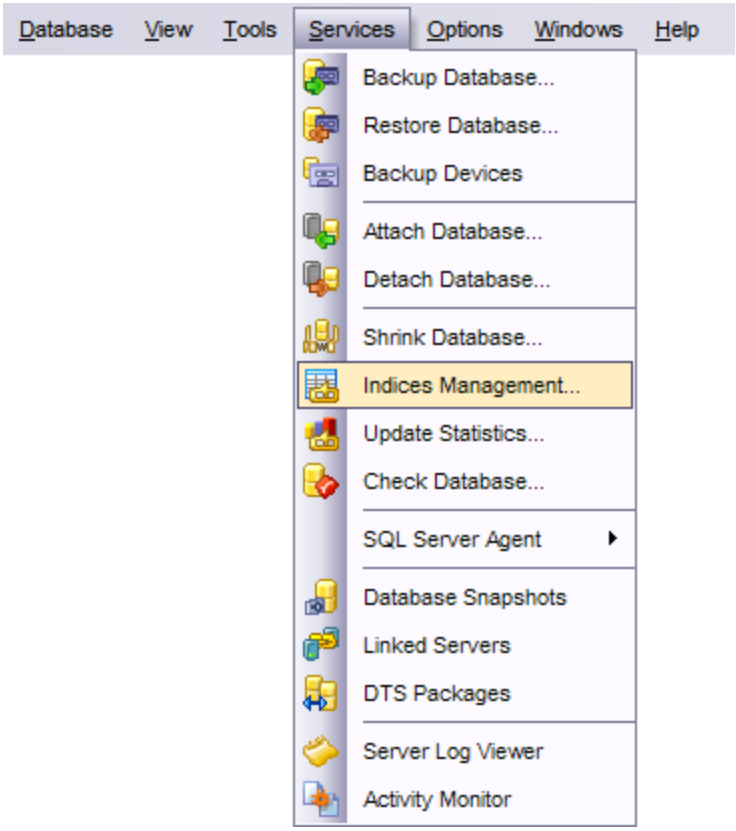
Indices Management Wizard allows you to perform index management operations in your SQL Server database: *reorganizing*, *rebuilding*, *disabling* (the *ALTER INDEX REORGANIZE... REBUILD... DISABLE* Transact-SQL statements are issued).

When indexes have pages in which the logical ordering based on the key value, does not match the physical ordering inside the data file, these indexes are called *fragmented*. Highly fragmented indexes may considerably decrease query performance. Index fragmentation is remedied by either reorganizing or by rebuilding an index.

To run the wizard, select the **Services | Indices Management...** main menu item, or right-click the database alias in the DB Explorer tree and select the **Tasks | Indices Management...** item of the context menu.

- Selecting database for indices management
- Selecting tables that contain indexes
- Specifying actions and setting options
- Editing and executing the result script
- Using templates

© 2008 EMS Database Management Solutions



Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**No**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

Attach Database
Detach Database
Backup Database
Restore Database
Shrink Database
Update Statistics
Check Database
Grant Manager

© 2008 EMS Database Management Solutions

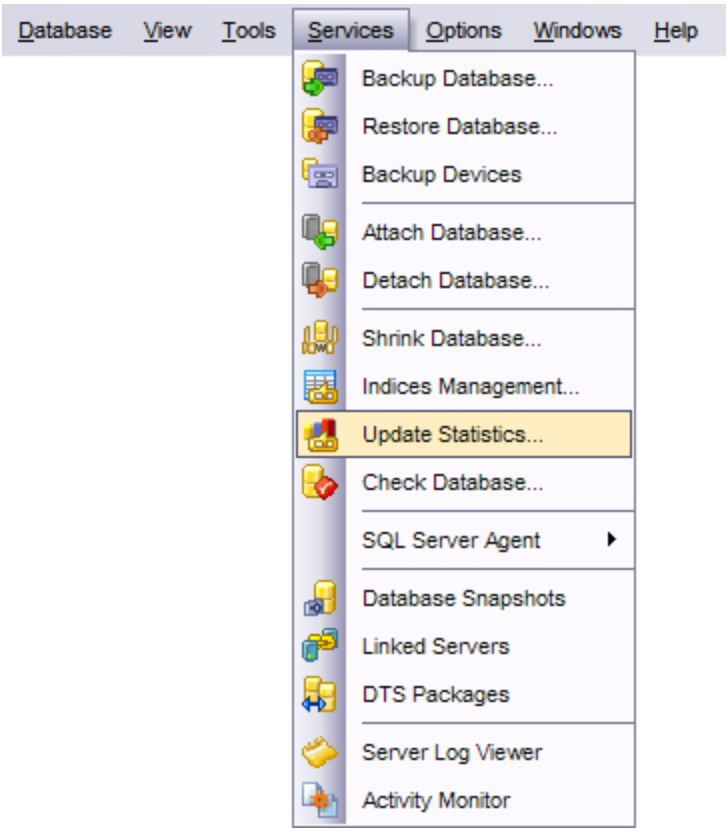
13.1.13 Update Statistics

Statistics Update Wizard allows you to perform the update statistics operation in your SQL Server database (the *UPDATE STATISTICS* statement is issued).

This operation is used to updates the statistics for the specified table(s) and view(s).

To run the wizard, select the **Services | Update Statistics...** main menu item, or right-click the database alias in the DB Explorer tree and select the **Tasks | Update Statistics...** item of the context menu.

- Selecting database for statistics update
- Selecting tables and views
- Specifying update statistics options
- Editing and executing the result script
- Using templates



Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**No**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

© 2008 EMS Database Management Solutions

See also:
Attach Database
Detach Database
Backup Database
Restore Database
Shrink Database
Indices Management
Check Database
Grant Manager

13.1.14 Check Database

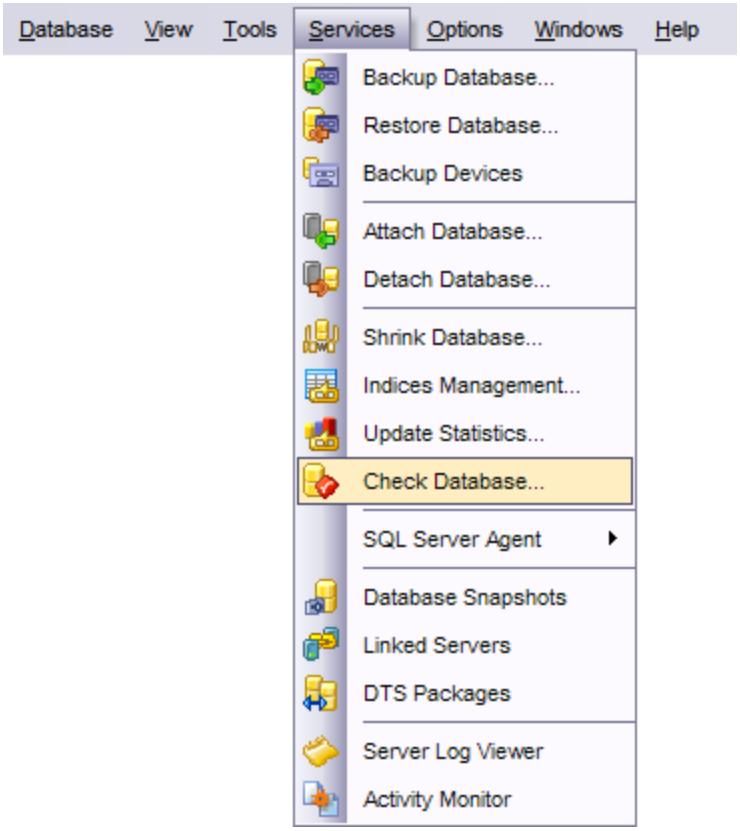
Check Database Wizard allows you to perform the check database operation on your SQL Server system (the *DBCC CHECKDB* Transact-SQL statement is issued).

This operation is used to check the allocation, structural, and logical integrity of all the objects in the specified database.

To run the wizard, select the **Services | Check Database...** main menu item, or right-click the database alias in the DB Explorer tree and select the **Tasks | Check Database...** item of the context menu.

- Selecting database to check
- Specifying arguments and options
- Editing and executing the result script
- Using templates

© 2008 EMS Database Management Solutions



Availability:

Full version (for Windows)	**Yes**
Lite version (for Windows)	**No**

Note: To compare all features of the **Full** and the **Lite** versions of SQL Manager, refer to the Feature Matrix page.

See also:

Attach Database
Detach Database
Backup Database
Restore Database
Shrink Database
Indices Management
Update Statistics
Grant Manager

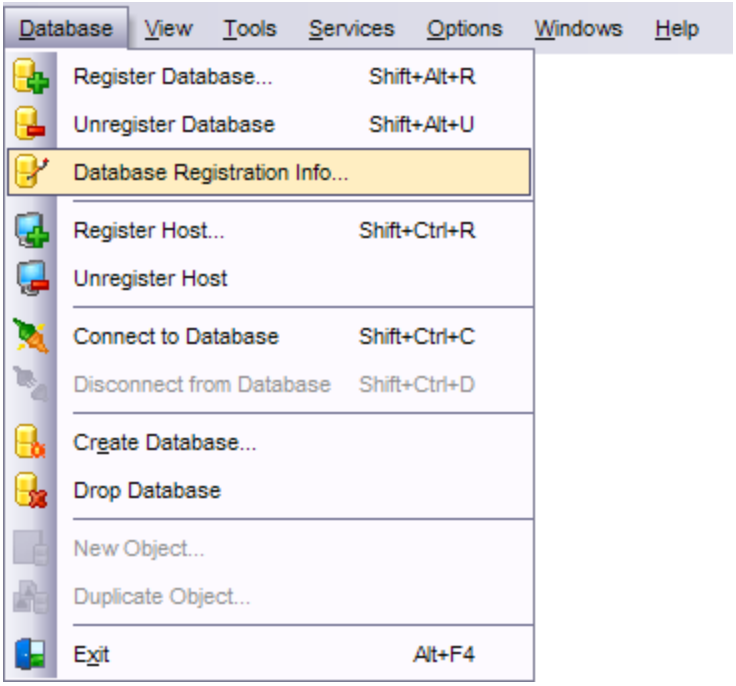
© 2008 EMS Database Management Solutions

13.1.15 Edit database registration info

Use the **Database Registration Info** dialog to view and edit the registration properties of the database.

To open the dialog, select the database or any of its objects in the DB Explorer tree, then select the **Database | Database Registration Info...** main menu item, or right-click the database alias in DB Explorer and use the **Database Registration Info...** context menu item. You can also use the **Database Registration Info...** button on the main toolbar.

- Editing connection properties
- Setting database options
- Setting display options
- Setting default directories
- Setting log options
- Find Option



See also:

Register Host wizard
Create Database wizard
Register Database wizard
Database Properties
Server Properties

© 2008 EMS Database Management Solutions

13.1.16 Edit database properties

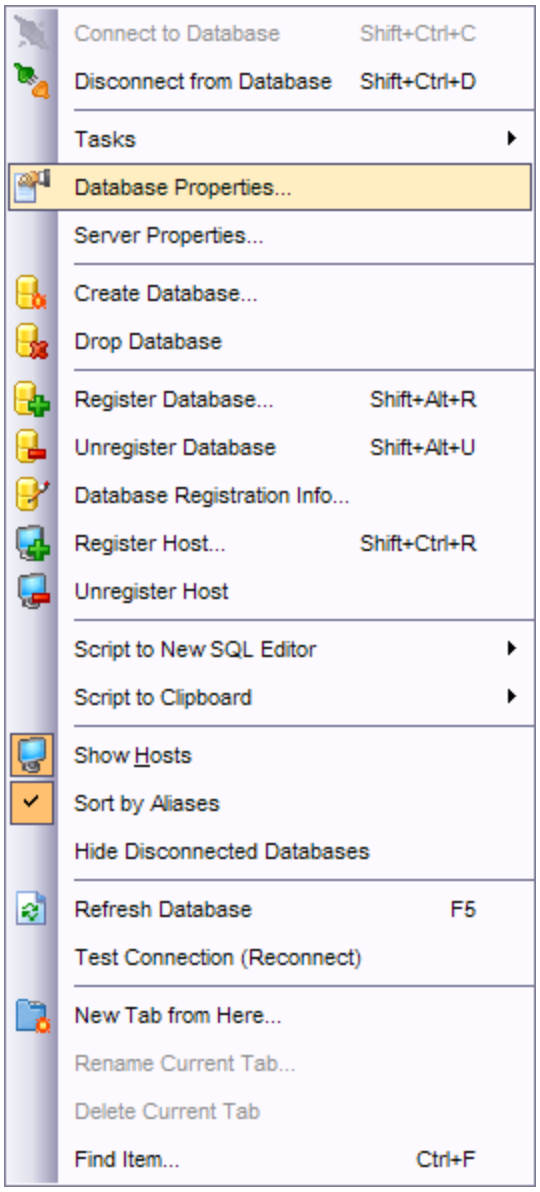
The **Database Properties** dialog allows you to view/edit a number of properties which can be changed to optimize database performance.

To open the dialog, right-click the Database in DB Explorer and select the **Database Properties...** context menu item, or use the **Properties** button on the main toolbar.

A number of database-level options that determine the characteristics of the database can be set for each database. Only the system administrator, database owner, members of the **sysadmin** and **dbcreator** fixed server roles and **db_owner** fixed database roles can modify these options. These options are unique to each database and do not affect other databases.

- Common database options
- Recovery options
- ANSI options
- SQL options
- Automation options
- Cursor options
- Advanced database options
- Data files
- Transaction log files
- Filegroups
- Find Option

© 2008 EMS Database Management Solutions



See also:

Register Host wizard
Create Database wizard
Register Database wizard
Database Registration Info
Server Properties

© 2008 EMS Database Management Solutions

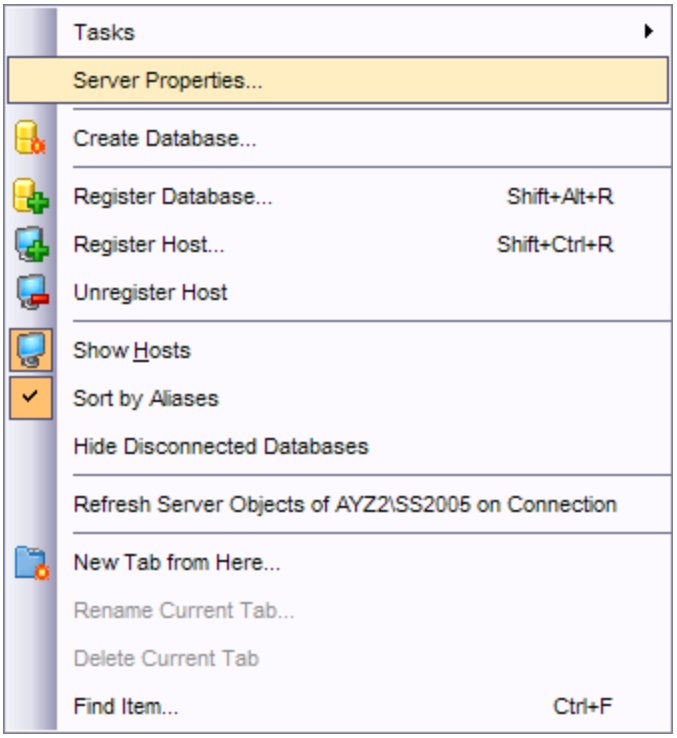
13.1.17 Edit server properties

The **Server Properties** dialog allows you to view/edit a number of server options which can be changed to optimize Microsoft® SQL Server performance.

- General options
- Memory options
- Processor options
- Security options
- Active Directory
- Connection settings
- Database Settings
- Misc Server Settings
- Find Option

Configured values
This option displays the configured values for options on this page. If you change these values, select **Running Values** to see whether the changes have taken effect. If they have not, the instance of SQL Server must be restarted first.

Running values
This option displays the currently running values for options on this page. These values are read-only.



See also:
Register Host wizard

© 2008 EMS Database Management Solutions

Create Database wizard
Register Database wizard
Database Registration Info
Database Properties

© 2008 EMS Database Management Solutions

13.2 Work with database objects

The following chapters inform you how to...

13.2.1 Work with Schemas

13.2.1.1 Create Schema

A **schema** is essentially a namespace: it can be defined as a collection of database objects that form a single namespace. A namespace is a set in which each element has a unique name.

Creating Schemas

To create a new schema:
- select the **Database | New Object...** main menu item;
- select **Schema** in the Create New Object dialog;
- define schema properties using the appropriate tabs of Schema Editor.

Hint: To create a new schema, you can also right-click the **Schemas** node or any object within this node in the DB Explorer tree and select the **New Schema** item from the context menu.

To create a new schema with the same properties as one of existing schemas has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a schema in the DB Explorer tree and select the **Duplicate Schema <schema_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new schema in, and to edit the result SQL statement for creating the schema.

Editing Schemas

To edit an existing schema:
- select the schema for editing in the DB Explorer tree (type the first letters of the schema name for quick search);
- right-click the object and select the **Edit Schema <schema_name>...** context menu item, or simply double-click the schema;
- edit schema properties using the appropriate tabs of Schema Editor.

Dropping Schemas

To drop a schema:
- select the schema to drop in the DB Explorer tree;
- right-click the object and select the **Drop Schema <schema_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new schema;
Ctrl+O to edit the selected schema;
Shift+Del to drop the object from the database.

© 2008 EMS Database Management Solutions

See also:
New Object dialog
Duplicate Object Wizard
Schema objects
Non-schema objects
Database-level principals

13.2.1.2 Edit Schema

A **schema** is essentially a namespace: it can be defined as a collection of database objects that form a single namespace. A namespace is a set in which each element has a unique name.

Creating Schemas

To create a new schema:
- select the **Database | New Object...** main menu item;
- select **Schema** in the Create New Object dialog;
- define schema properties using the appropriate tabs of Schema Editor.

Hint: To create a new schema, you can also right-click the **Schemas** node or any object within this node in the DB Explorer tree and select the **New Schema** item from the context menu.

To create a new schema with the same properties as one of existing schemas has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a schema in the DB Explorer tree and select the **Duplicate Schema <schema_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new schema in, and to edit the result SQL statement for creating the schema.

Editing Schemas

To edit an existing schema:
- select the schema for editing in the DB Explorer tree (type the first letters of the schema name for quick search);
- right-click the object and select the **Edit Schema <schema_name>...** context menu item, or simply double-click the schema;
- edit schema properties using the appropriate tabs of Schema Editor.

Dropping Schemas

To drop a schema:
- select the schema to drop in the DB Explorer tree;

© 2008 EMS Database Management Solutions

- right-click the object and select the **Drop Schema <schema_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new schema;
Ctrl+O to edit the selected schema;
Shift+Del to drop the object from the database.

See also:
New Object dialog
Duplicate Object Wizard
Schema objects
Non-schema objects
Database-level principals

13.2.1.3 Drop Schema

A **schema** is essentially a namespace: it can be defined as a collection of database objects that form a single namespace. A namespace is a set in which each element has a unique name.

Creating Schemas

To create a new schema:
- select the **Database | New Object...** main menu item;
- select **Schema** in the Create New Object dialog;
- define schema properties using the appropriate tabs of Schema Editor.

Hint: To create a new schema, you can also right-click the **Schemas** node or any object within this node in the DB Explorer tree and select the **New Schema** item from the context menu.

To create a new schema with the same properties as one of existing schemas has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a schema in the DB Explorer tree and select the **Duplicate Schema <schema_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new schema in, and to edit the result SQL statement for creating the schema.

Editing Schemas

To edit an existing schema:
- select the schema for editing in the DB Explorer tree (type the first letters of the schema name for quick search);

© 2008 EMS Database Management Solutions

- right-click the object and select the **Edit Schema <schema_name>...** context menu item, or simply double-click the schema;
- edit schema properties using the appropriate tabs of Schema Editor.

Dropping Schemas

To drop a schema:
- select the schema to drop in the DB Explorer tree;
- right-click the object and select the **Drop Schema <schema_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new schema;
Ctrl+O to edit the selected schema;
Shift+Del to drop the object from the database.

See also:
New Object dialog
Duplicate Object Wizard
Schema objects
Non-schema objects
Database-level principals

13.2.2 Work with Tables

13.2.2.1 Create Table

Relational databases store all their data in **tables**. A table is a data structure consisting of an unordered set of horizontal rows, each containing the same number of vertical columns. The intersection of an individual row and column is a field that contains a specific piece of information. Much of the power of relational databases comes from defining the relations among the tables.

Creating Tables

To create a new table:
- select the **Database | New Object...** main menu item;
- select **Table** in the Create New Object dialog;
- define table properties and fields using the appropriate tabs of Table Editor.

Hint: To create a new table, you can also right-click the **Tables** node or any object within this node in the DB Explorer tree and select the **New Table...** item from the context menu.

To create a new table with the same properties as one of existing tables has:
- select the **Database | Duplicate Object...** menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a table in the DB Explorer tree and select the **Duplicate**

© 2008 EMS Database Management Solutions

Table <table_name>... context menu item.

Duplicate Object Wizard allows you to select the database to create a new table in, and to edit the result SQL statement for creating the table.

Editing Tables

To edit an existing table (manage its fields, indexes, data, etc):
- select the table for editing in the DB Explorer tree (type the first letters of the table name for quick search);
- right-click the object and select the **Edit Table <table_name>** context menu item, or simply double-click the table;
- edit table subobjects and data using the appropriate tabs of Table Editor.

To change the name of a table:
- select the table to rename in the DB Explorer tree;
- right-click the table alias and select the **Rename Table <table_name>...** item from the context menu;
- edit the table name using the **Rename Object...** dialog.

Dropping Tables

To drop a table:
- select the table to drop in the DB Explorer tree;
- right-click the object and select the **Drop Table <table_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new table;
Ctrl+O to edit the selected table;
Ctrl+R to rename the table;
Shift+Del to drop the object from the database.

13.2.2.2 Edit Table

Relational databases store all their data in **tables**. A table is a data structure consisting of an unordered set of horizontal rows, each containing the same number of vertical columns. The intersection of an individual row and column is a field that contains a specific piece of information. Much of the power of relational databases comes from defining the relations among the tables.

Creating Tables

To create a new table:
- select the **Database | New Object...** main menu item;
- select **Table** in the Create New Object dialog;
- define table properties and fields using the appropriate tabs of Table Editor.

Hint: To create a new table, you can also right-click the **Tables** node or any object within this node in the DB Explorer tree and select the **New Table...** item from the context menu.

© 2008 EMS Database Management Solutions

To create a new table with the same properties as one of existing tables has:
- select the **Database | Duplicate Object...** menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a table in the DB Explorer tree and select the **Duplicate Table <table_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new table in, and to edit the result SQL statement for creating the table.

Editing Tables

To edit an existing table (manage its fields, indexes, data, etc):
- select the table for editing in the DB Explorer tree (type the first letters of the table name for quick search);
- right-click the object and select the **Edit Table <table_name>** context menu item, or simply double-click the table;
- edit table subobjects and data using the appropriate tabs of Table Editor.

To change the name of a table:
- select the table to rename in the DB Explorer tree;
- right-click the table alias and select the **Rename Table <table_name>...** item from the context menu;
- edit the table name using the **Rename Object...** dialog.

Dropping Tables

To drop a table:
- select the table to drop in the DB Explorer tree;
- right-click the object and select the **Drop Table <table_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new table;
Ctrl+O to edit the selected table;
Ctrl+R to rename the table;
Shift+Del to drop the object from the database.

13.2.2.3 Drop Table

Relational databases store all their data in **tables**. A table is a data structure consisting of an unordered set of horizontal rows, each containing the same number of vertical columns. The intersection of an individual row and column is a field that contains a specific piece of information. Much of the power of relational databases comes from defining the relations among the tables.

Creating Tables

To create a new table:
- select the **Database | New Object...** main menu item;

© 2008 EMS Database Management Solutions

- select **Table** in the Create New Object dialog;
- define table properties and fields using the appropriate tabs of Table Editor.

Hint: To create a new table, you can also right-click the **Tables** node or any object within this node in the DB Explorer tree and select the **New Table...** item from the context menu.

To create a new table with the same properties as one of existing tables has:
- select the **Database | Duplicate Object...** menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a table in the DB Explorer tree and select the **Duplicate Table <table_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new table in, and to edit the result SQL statement for creating the table.

Editing Tables

To edit an existing table (manage its fields, indexes, data, etc):
- select the table for editing in the DB Explorer tree (type the first letters of the table name for quick search);
- right-click the object and select the **Edit Table <table_name>** context menu item, or simply double-click the table;
- edit table subobjects and data using the appropriate tabs of Table Editor.

To change the name of a table:
- select the table to rename in the DB Explorer tree;
- right-click the table alias and select the **Rename Table <table_name>...** item from the context menu;
- edit the table name using the **Rename Object...** dialog.

Dropping Tables

To drop a table:
- select the table to drop in the DB Explorer tree;
- right-click the object and select the **Drop Table <table_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new table;
Ctrl+O to edit the selected table;
Ctrl+R to rename the table;
Shift+Del to drop the object from the database.

13.2.2.4 Manage table fields

13.2.2.4.1 Add Table Field

Table fields are managed within the **Fields** tab of Table Editor.

© 2008 EMS Database Management Solutions

Creating Fields

To create a new table field:
- open the table in Table Editor;
- proceed to the **Fields** tab there;
- right-click the tab area and select the **New Field** context menu item, or press the *Ins* key;
- define the field properties using the Field Editor dialog.

Editing Fields

To edit an existing table field:
- open the table in Table Editor;
- proceed to the **Fields** tab there;
- right-click the field and select the **Edit Field <field_name>** context menu item, or simply double-click the field;
- edit the field properties using the Field Editor dialog.

To change the order of table fields:
- open the table in Table Editor;
- proceed to the **Fields** tab there;
- right-click a field and select the **Reorder Fields <field_name>** context menu item;
- edit fields' order using the Reorder Fields dialog.

Dropping Fields

To drop a table field:
- open the table in Table Editor;
- proceed to the **Fields** tab there;
- right-click the field and select the **Drop Field <field_name>** context menu item;
- confirm dropping in the dialog window.

13.2.2.4.2 Edit Table Field

Table fields are managed within the **Fields** tab of Table Editor.

Creating Fields

To create a new table field:
- open the table in Table Editor;
- proceed to the **Fields** tab there;
- right-click the tab area and select the **New Field** context menu item, or press the *Ins* key;
- define the field properties using the Field Editor dialog.

Editing Fields

To edit an existing table field:
- open the table in Table Editor;
- proceed to the **Fields** tab there;
- right-click the field and select the **Edit Field <field_name>** context menu item, or simply double-click the field;

© 2008 EMS Database Management Solutions

- edit the field properties using the Field Editor dialog.

To change the order of table fields:

- open the table in Table Editor;
- proceed to the **Fields** tab there;
- right-click a field and select the **Reorder Fields <field_name>** context menu item;
- edit fields' order using the Reorder Fields dialog.

Dropping Fields

To drop a table field:

- open the table in Table Editor;
- proceed to the **Fields** tab there;
- right-click the field and select the **Drop Field <field_name>** context menu item;
- confirm dropping in the dialog window.

13.2.2.4.3 Drop Table Field

Table fields are managed within the **Fields** tab of Table Editor.

Creating Fields

To create a new table field:

- open the table in Table Editor;
- proceed to the **Fields** tab there;
- right-click the tab area and select the **New Field** context menu item, or press the *Ins* key;
- define the field properties using the Field Editor dialog.

Editing Fields

To edit an existing table field:

- open the table in Table Editor;
- proceed to the **Fields** tab there;
- right-click the field and select the **Edit Field <field_name>** context menu item, or simply double-click the field;
- edit the field properties using the Field Editor dialog.

To change the order of table fields:

- open the table in Table Editor;
- proceed to the **Fields** tab there;
- right-click a field and select the **Reorder Fields <field_name>** context menu item;
- edit fields' order using the Reorder Fields dialog.

Dropping Fields

To drop a table field:

- open the table in Table Editor;
- proceed to the **Fields** tab there;
- right-click the field and select the **Drop Field <field_name>** context menu item;
- confirm dropping in the dialog window.

© 2008 EMS Database Management Solutions

13.2.2.5 Manage foreign keys

13.2.2.5.1 Add Foreign Key

A **Foreign key** constraint (also called a *referential integrity constraint*) designates a column as the Foreign key and establishes a relationship between that foreign key and a specified *Primary* or *Unique* key called the *referenced key*. A composite Foreign key designates a combination of columns as the foreign key.

Table Foreign keys are managed within the **Foreign Keys** tab of Table Editor.

Creating Foreign Keys

To create a new Foreign key:
- open the table in Table Editor;
- proceed to the **Foreign Keys** tab there;
- right-click the tab area and select the **New Foreign Key** context menu item, or press the *Ins* key;
- define the Foreign key properties using the Foreign Key Editor dialog.

Editing Foreign Keys

To view/edit an existing Foreign key:
- open the table in Table Editor;
- proceed to the **Foreign Keys** tab there;
- right-click the Foreign key to edit and select the **Edit Foreign Key <foreign_key_name>** context menu item, or simply double-click the Foreign key;
- edit the Foreign key properties using the Foreign Key Editor dialog.

Dropping Foreign Keys

To drop a Foreign key:
- open the table in Table Editor;
- proceed to the **Foreign Keys** tab there;
- right-click the Foreign key and select the **Drop Foreign Key <foreign_key_name>** context menu item;
- confirm dropping in the dialog window.

13.2.2.5.2 Edit Foreign Key

A **Foreign key** constraint (also called a *referential integrity constraint*) designates a column as the Foreign key and establishes a relationship between that foreign key and a specified *Primary* or *Unique* key called the *referenced key*. A composite Foreign key designates a combination of columns as the foreign key.

Table Foreign keys are managed within the **Foreign Keys** tab of Table Editor.

Creating Foreign Keys

To create a new Foreign key:

© 2008 EMS Database Management Solutions

- open the table in Table Editor;
- proceed to the **Foreign Keys** tab there;
- right-click the tab area and select the **New Foreign Key** context menu item, or press the *Ins* key;
- define the Foreign key properties using the Foreign Key Editor dialog.

Editing Foreign Keys

To view/edit an existing Foreign key:
- open the table in Table Editor;
- proceed to the **Foreign Keys** tab there;
- right-click the Foreign key to edit and select the **Edit Foreign Key <foreign_key_name>** context menu item, or simply double-click the Foreign key;
- edit the Foreign key properties using the Foreign Key Editor dialog.

Dropping Foreign Keys

To drop a Foreign key:
- open the table in Table Editor;
- proceed to the **Foreign Keys** tab there;
- right-click the Foreign key and select the **Drop Foreign Key <foreign_key_name>** context menu item;
- confirm dropping in the dialog window.

13.2.2.5.3 Drop Foreign Key

A **Foreign key** constraint (also called a *referential integrity constraint*) designates a column as the Foreign key and establishes a relationship between that foreign key and a specified *Primary* or *Unique* key called the *referenced key*. A composite Foreign key designates a combination of columns as the foreign key.

Table Foreign keys are managed within the **Foreign Keys** tab of Table Editor.

Creating Foreign Keys

To create a new Foreign key:
- open the table in Table Editor;
- proceed to the **Foreign Keys** tab there;
- right-click the tab area and select the **New Foreign Key** context menu item, or press the *Ins* key;
- define the Foreign key properties using the Foreign Key Editor dialog.

Editing Foreign Keys

To view/edit an existing Foreign key:
- open the table in Table Editor;
- proceed to the **Foreign Keys** tab there;
- right-click the Foreign key to edit and select the **Edit Foreign Key <foreign_key_name>** context menu item, or simply double-click the Foreign key;
- edit the Foreign key properties using the Foreign Key Editor dialog.

© 2008 EMS Database Management Solutions

Dropping Foreign Keys

To drop a Foreign key:
- open the table in Table Editor;
- proceed to the **Foreign Keys** tab there;
- right-click the Foreign key and select the **Drop Foreign Key <foreign_key_name>** context menu item;
- confirm dropping in the dialog window.

13.2.2.6 Manage table checks

13.2.2.6.1 Create Check

A **check** specifies 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. Should 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. A check constraint specified as a column constraint should reference that column's value only, while an expression appearing in a table constraint may reference multiple columns.

Table checks are managed within the **Checks** tab of Table Editor.

Creating Checks

To create a new check:
- open the table in Table Editor;
- proceed to the **Checks** tab there;
- right-click the tab area and select the **New Check Constraint** context menu item, or press the *Ins* key;
- define the check properties using the Check Editor dialog.

Editing Checks

To view an existing check:
- open the table in Table Editor;
- proceed to the **Checks** tab there;
- right-click the check and select the **Edit Check <check_name>** context menu item, or simply double-click the check;
- edit the check properties using the Check Editor dialog.

Dropping Checks

To drop a check:
- open the table in Table Editor;
- proceed to the **Checks** tab there;
- right-click the check and select the **Drop Check <check_name>** context menu item;
- confirm dropping in the dialog window.

© 2008 EMS Database Management Solutions

13.2.2.6.2 Edit Check

A **check** specifies 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. Should 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. A check constraint specified as a column constraint should reference that column's value only, while an expression appearing in a table constraint may reference multiple columns.

Table checks are managed within the **Checks** tab of Table Editor.

Creating Checks

To create a new check:
- open the table in Table Editor;
- proceed to the **Checks** tab there;
- right-click the tab area and select the **New Check Constraint** context menu item, or press the *Ins* key;
- define the check properties using the Check Editor dialog.

Editing Checks

To view an existing check:
- open the table in Table Editor;
- proceed to the **Checks** tab there;
- right-click the check and select the **Edit Check <check_name>** context menu item, or simply double-click the check;
- edit the check properties using the Check Editor dialog.

Dropping Checks

To drop a check:
- open the table in Table Editor;
- proceed to the **Checks** tab there;
- right-click the check and select the **Drop Check <check_name>** context menu item;
- confirm dropping in the dialog window.

13.2.2.6.3 Drop Check

A **check** specifies 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. Should 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. A check constraint specified as a column constraint should reference that column's value only, while an expression appearing in a table constraint may reference multiple columns.

Table checks are managed within the **Checks** tab of Table Editor.

Creating Checks

© 2008 EMS Database Management Solutions

To create a new check:
- open the table in Table Editor;
- proceed to the **Checks** tab there;
- right-click the tab area and select the **New Check Constraint** context menu item, or press the *Ins* key;
- define the check properties using the Check Editor dialog.

Editing Checks

To view an existing check:
- open the table in Table Editor;
- proceed to the **Checks** tab there;
- right-click the check and select the **Edit Check <check_name>** context menu item, or simply double-click the check;
- edit the check properties using the Check Editor dialog.

Dropping Checks

To drop a check:
- open the table in Table Editor;
- proceed to the **Checks** tab there;
- right-click the check and select the **Drop Check <check_name>** context menu item;
- confirm dropping in the dialog window.

13.2.2.7 Manage table indexes

13.2.2.7.1 Create Index

Table indexes are managed within the **Indices** tab of Table Editor.

Creating Indexes

To create a new table index:
- open the table in Table Editor;
- proceed to the **Indexes** tab there;
- right-click the tab area and select the **New Index** context menu item, or press the *Ins* key;
- define the index properties using the Index Editor dialog.

Editing Indexes

To edit an existing table index:
- open the table in Table Editor;
- proceed to the **Indexes** tab there;
- right-click the index and select the **Edit Index** context menu item, or simply double-click the index;
- edit the index properties using the Index Editor dialog.

Dropping Indexes

To drop a table index:

© 2008 EMS Database Management Solutions

- open the table in Table Editor;
- proceed to the **Indexes** tab there;
- right-click the index and select the **Drop Index** context menu item;
- confirm dropping in the dialog window.

13.2.2.7.2 Edit Index

Table indexes are managed within the **Indices** tab of Table Editor.

Creating Indexes

To create a new table index:
- open the table in Table Editor;
- proceed to the **Indexes** tab there;
- right-click the tab area and select the **New Index** context menu item, or press the *Ins* key;
- define the index properties using the Index Editor dialog.

Editing Indexes

To edit an existing table index:
- open the table in Table Editor;
- proceed to the **Indexes** tab there;
- right-click the index and select the **Edit Index** context menu item, or simply double-click the index;
- edit the index properties using the Index Editor dialog.

Dropping Indexes

To drop a table index:
- open the table in Table Editor;
- proceed to the **Indexes** tab there;
- right-click the index and select the **Drop Index** context menu item;
- confirm dropping in the dialog window.

13.2.2.7.3 Drop Index

Table indexes are managed within the **Indices** tab of Table Editor.

Creating Indexes

To create a new table index:
- open the table in Table Editor;
- proceed to the **Indexes** tab there;
- right-click the tab area and select the **New Index** context menu item, or press the *Ins* key;
- define the index properties using the Index Editor dialog.

© 2008 EMS Database Management Solutions

Editing Indexes

To edit an existing table index:
- open the table in Table Editor;
- proceed to the **Indexes** tab there;
- right-click the index and select the **Edit Index** context menu item, or simply double-click the index;
- edit the index properties using the Index Editor dialog.

Dropping Indexes

To drop a table index:
- open the table in Table Editor;
- proceed to the **Indexes** tab there;
- right-click the index and select the **Drop Index** context menu item;
- confirm dropping in the dialog window.

13.2.2.8 Manage table triggers

13.2.2.8.1 Create Trigger

A **trigger** is a special kind of stored procedure that automatically executes when an event occurs in the database server.

DML triggers execute when a user tries to modify data through a data manipulation language (DML) event. DML events are *INSERT*, *UPDATE*, or *DELETE* statements on a table or view.

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

Creating Triggers

To create a new trigger:
- open the table in Table Editor;
- proceed to the **Triggers** tab there;
- right-click the tab area and select the **New Trigger** context menu item, or press the *Ins* key;
- define the trigger properties using the Trigger Editor dialog.

Editing Triggers

To edit an existing trigger:
- open the table in Table Editor;
- proceed to the **Triggers** tab there;
- right-click the trigger and select the **Edit Trigger <trigger_name>** context menu item, or simply double-click the trigger;
- edit the trigger properties using the Trigger Editor dialog.

Dropping Triggers

To drop a trigger:

© 2008 EMS Database Management Solutions

- open the table in Table Editor;
- proceed to the **Triggers** tab there;
- right-click the trigger and select the **Drop Trigger <trigger_name>** context menu item;
- confirm dropping in the dialog window.

13.2.2.8.2 Edit Trigger

A **trigger** is a special kind of stored procedure that automatically executes when an event occurs in the database server.

DML triggers execute when a user tries to modify data through a data manipulation language (DML) event. DML events are *INSERT*, *UPDATE*, or *DELETE* statements on a table or view.

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

Creating Triggers

To create a new trigger:
- open the table in Table Editor;
- proceed to the **Triggers** tab there;
- right-click the tab area and select the **New Trigger** context menu item, or press the *Ins* key;
- define the trigger properties using the Trigger Editor dialog.

Editing Triggers

To edit an existing trigger:
- open the table in Table Editor;
- proceed to the **Triggers** tab there;
- right-click the trigger and select the **Edit Trigger <trigger_name>** context menu item, or simply double-click the trigger;
- edit the trigger properties using the Trigger Editor dialog.

Dropping Triggers

To drop a trigger:
- open the table in Table Editor;
- proceed to the **Triggers** tab there;
- right-click the trigger and select the **Drop Trigger <trigger_name>** context menu item;
- confirm dropping in the dialog window.

13.2.2.8.3 Drop Trigger

A **trigger** is a special kind of stored procedure that automatically executes when an event occurs in the database server.

© 2008 EMS Database Management Solutions

DML triggers execute when a user tries to modify data through a data manipulation language (DML) event. DML events are *INSERT*, *UPDATE*, or *DELETE* statements on a table or view.

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

Creating Triggers

To create a new trigger:
- open the table in Table Editor;
- proceed to the **Triggers** tab there;
- right-click the tab area and select the **New Trigger** context menu item, or press the *Ins* key;
- define the trigger properties using the Trigger Editor dialog.

Editing Triggers

To edit an existing trigger:
- open the table in Table Editor;
- proceed to the **Triggers** tab there;
- right-click the trigger and select the **Edit Trigger <trigger_name>** context menu item, or simply double-click the trigger;
- edit the trigger properties using the Trigger Editor dialog.

Dropping Triggers

To drop a trigger:
- open the table in Table Editor;
- proceed to the **Triggers** tab there;
- right-click the trigger and select the **Drop Trigger <trigger_name>** context menu item;
- confirm dropping in the dialog window.

13.2.3 Work with Views

13.2.3.1 Create View

A **view** is a logical table based on one or more tables or views. A view contains no data itself. The tables upon which a view is based are called *base tables*.

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).

Creating Views

To create a new view:
- select the **Database | New Object...** main menu item;
- select **View** in the Create New Object dialog;
- define view properties using the appropriate tabs of View Editor.

© 2008 EMS Database Management Solutions

Hint: To create a new view, you can also right-click the **Views** node or any object within this node in the DB Explorer tree and select the **New View** item from the context menu.

Another way to create a view is to build a query in Visual Query Builder (click the **Create view** Navigation bar item after building).

To create a new view with the same properties as one of existing views has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a view in the DB Explorer tree and select the **Duplicate View <view_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new view in, and to edit the result SQL statement for creating the view.

Editing Views

To edit an existing view:
- select the view for editing in the DB Explorer tree (type the first letters of the view name for quick search);
- right-click the object and select the **Edit View <view_name>** context menu item, or simply double-click the view;
- edit view definition and data using the appropriate tabs of View Editor.

To change the name of a view:
- select the view to rename in the DB Explorer tree;
- right-click the view alias and select the **Rename View <view_name>...** item from the context menu;
- edit the view name using the **Rename Object...** dialog.

Dropping Views

To drop a view:
- select the view to drop in the DB Explorer tree;
- right-click the object and select the **Drop View <view_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new view;
Ctrl+O to edit the selected view;
Ctrl+R to rename the view;
Shift+Del to drop the object from the database.

13.2.3.2 Edit View

A **view** is a logical table based on one or more tables or views. A view contains no data itself. The tables upon which a view is based are called *base tables*.

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).

© 2008 EMS Database Management Solutions

Creating Views

To create a new view:
- select the **Database | New Object...** main menu item;
- select **View** in the Create New Object dialog;
- define view properties using the appropriate tabs of View Editor.

Hint: To create a new view, you can also right-click the **Views** node or any object within this node in the DB Explorer tree and select the **New View** item from the context menu.

Another way to create a view is to build a query in Visual Query Builder (click the **Create view** Navigation bar item after building).

To create a new view with the same properties as one of existing views has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a view in the DB Explorer tree and select the **Duplicate View <view_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new view in, and to edit the result SQL statement for creating the view.

Editing Views

To edit an existing view:
- select the view for editing in the DB Explorer tree (type the first letters of the view name for quick search);
- right-click the object and select the **Edit View <view_name>** context menu item, or simply double-click the view;
- edit view definition and data using the appropriate tabs of View Editor.

To change the name of a view:
- select the view to rename in the DB Explorer tree;
- right-click the view alias and select the **Rename View <view_name>...** item from the context menu;
- edit the view name using the **Rename Object...** dialog.

Dropping Views

To drop a view:
- select the view to drop in the DB Explorer tree;
- right-click the object and select the **Drop View <view_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new view;
Ctrl+O to edit the selected view;
Ctrl+R to rename the view;
Shift+Del to drop the object from the database.

© 2008 EMS Database Management Solutions

13.2.3.3 Drop View

A **view** is a logical table based on one or more tables or views. A view contains no data itself. The tables upon which a view is based are called *base tables*.

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).

Creating Views

To create a new view:
- select the **Database | New Object...** main menu item;
- select **View** in the Create New Object dialog;
- define view properties using the appropriate tabs of View Editor.

Hint: To create a new view, you can also right-click the **Views** node or any object within this node in the DB Explorer tree and select the **New View** item from the context menu.

Another way to create a view is to build a query in Visual Query Builder (click the **Create view** Navigation bar item after building).

To create a new view with the same properties as one of existing views has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a view in the DB Explorer tree and select the **Duplicate View <view_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new view in, and to edit the result SQL statement for creating the view.

Editing Views

To edit an existing view:
- select the view for editing in the DB Explorer tree (type the first letters of the view name for quick search);
- right-click the object and select the **Edit View <view_name>** context menu item, or simply double-click the view;
- edit view definition and data using the appropriate tabs of View Editor.

To change the name of a view:
- select the view to rename in the DB Explorer tree;
- right-click the view alias and select the **Rename View <view_name>...** item from the context menu;
- edit the view name using the **Rename Object...** dialog.

Dropping Views

To drop a view:
- select the view to drop in the DB Explorer tree;
- right-click the object and select the **Drop View <view_name>...** context menu item;
- confirm dropping in the dialog window.

© 2008 EMS Database Management Solutions

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new view;
Ctrl+O to edit the selected view;
Ctrl+R to rename the view;
Shift+Del to drop the object from the database.

13.2.4 Work with Procedures

13.2.4.1 Create Procedure

A **Procedure** is a set of procedural constructs and embedded SQL statements that is stored in the database and can be called by name.

When you create an application with Microsoft SQL Server 2005, the Transact-SQL programming language is the primary programming interface between your applications and the Microsoft SQL Server database. When you use Transact-SQL programs, two methods are available for storing and executing the programs. You can store the programs locally and create applications that send the commands to SQL Server and process the results, or you can store the programs as stored procedures in SQL Server and create applications that execute the stored procedures and process the results.

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.

Creating Procedures

To create a new procedure:
- select the **Database | New Object...** main menu item;
- select **Procedure** in the Create New Object dialog;
- define procedure properties using the appropriate tabs of Procedure Editor.

Hint: To create a new procedure, you can also right-click the **Procedures** node or any object within this node in the DB Explorer tree and select the **New Procedure...** item from the context menu.

To create a new procedure with the same properties as one of existing procedures has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a procedure in the DB Explorer tree and select the **Duplicate Procedure <procedure_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new procedure in, and to edit the result SQL statement for creating the procedure.

Editing Procedures

To edit an existing procedure:

© 2008 EMS Database Management Solutions

- select the procedure for editing in the DB Explorer tree (type the first letters of the procedure name for quick search);
- right-click the object and select the **Edit Procedure <procedure_name>** context menu item, or simply double-click the procedure;
- edit procedure definition using the appropriate tabs of Procedure Editor.

To change the name of a procedure:

- select the procedure to rename in the DB Explorer tree;
- right-click the procedure alias and select the **Rename Procedure <procedure_name>...** item from the context menu;
- edit the procedure name using the **Rename Object...** dialog.

Executing Procedures

To execute a procedure:

- select the procedure to execute in the DB Explorer tree (type the first letters of the procedure name for quick search);
- right-click the object and select the **Edit Procedure <procedure_name>...** context menu item, or simply double-click the procedure;
- execute the procedure using the **Execute Procedure** Navigation bar item of Procedure Editor.

Dropping Procedures

To drop a procedure:

- select the procedure to drop in the DB Explorer tree;
- right-click the object and select the **Drop Procedure <procedure_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new procedure;
Ctrl+O to edit the selected procedure;
Ctrl+R to rename the procedure;
Shift+Del to drop the object from the database.

13.2.4.2 Edit Procedure

A **Procedure** is a set of procedural constructs and embedded SQL statements that is stored in the database and can be called by name.

When you create an application with Microsoft SQL Server 2005, the Transact-SQL programming language is the primary programming interface between your applications and the Microsoft SQL Server database. When you use Transact-SQL programs, two methods are available for storing and executing the programs. You can store the programs locally and create applications that send the commands to SQL Server and process the results, or you can store the programs as stored procedures in SQL Server and create applications that execute the stored procedures and process the results.

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

© 2008 EMS Database Management Solutions

temporary use within a session, local temporary procedure, or for temporary use within all sessions, global temporary procedure.

Creating Procedures

To create a new procedure:
- select the **Database | New Object...** main menu item;
- select **Procedure** in the Create New Object dialog;
- define procedure properties using the appropriate tabs of Procedure Editor.

Hint: To create a new procedure, you can also right-click the **Procedures** node or any object within this node in the DB Explorer tree and select the **New Procedure...** item from the context menu.

To create a new procedure with the same properties as one of existing procedures has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a procedure in the DB Explorer tree and select the **Duplicate Procedure <procedure_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new procedure in, and to edit the result SQL statement for creating the procedure.

Editing Procedures

To edit an existing procedure:
- select the procedure for editing in the DB Explorer tree (type the first letters of the procedure name for quick search);
- right-click the object and select the **Edit Procedure <procedure_name>** context menu item, or simply double-click the procedure;
- edit procedure definition using the appropriate tabs of Procedure Editor.

To change the name of a procedure:
- select the procedure to rename in the DB Explorer tree;
- right-click the procedure alias and select the **Rename Procedure <procedure_name>...** item from the context menu;
- edit the procedure name using the **Rename Object...** dialog.

Executing Procedures

To execute a procedure:
- select the procedure to execute in the DB Explorer tree (type the first letters of the procedure name for quick search);
- right-click the object and select the **Edit Procedure <procedure_name>...** context menu item, or simply double-click the procedure;
- execute the procedure using the **Execute Procedure** Navigation bar item of Procedure Editor.

Dropping Procedures

To drop a procedure:
- select the procedure to drop in the DB Explorer tree;
- right-click the object and select the **Drop Procedure <procedure_name>...** context

© 2008 EMS Database Management Solutions

menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new procedure;
Ctrl+O to edit the selected procedure;
Ctrl+R to rename the procedure;
Shift+Del to drop the object from the database.

13.2.4.3 Execute Procedure

A **Procedure** is a set of procedural constructs and embedded SQL statements that is stored in the database and can be called by name.

When you create an application with Microsoft SQL Server 2005, the Transact-SQL programming language is the primary programming interface between your applications and the Microsoft SQL Server database. When you use Transact-SQL programs, two methods are available for storing and executing the programs. You can store the programs locally and create applications that send the commands to SQL Server and process the results, or you can store the programs as stored procedures in SQL Server and create applications that execute the stored procedures and process the results.

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.

Creating Procedures

To create a new procedure:
- select the **Database | New Object...** main menu item;
- select **Procedure** in the Create New Object dialog;
- define procedure properties using the appropriate tabs of Procedure Editor.

Hint: To create a new procedure, you can also right-click the **Procedures** node or any object within this node in the DB Explorer tree and select the **New Procedure...** item from the context menu.

To create a new procedure with the same properties as one of existing procedures has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a procedure in the DB Explorer tree and select the **Duplicate Procedure <procedure_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new procedure in, and to edit the result SQL statement for creating the procedure.

Editing Procedures

© 2008 EMS Database Management Solutions

To edit an existing procedure:
- select the procedure for editing in the DB Explorer tree (type the first letters of the procedure name for quick search);
- right-click the object and select the **Edit Procedure <procedure_name>** context menu item, or simply double-click the procedure;
- edit procedure definition using the appropriate tabs of Procedure Editor.

To change the name of a procedure:
- select the procedure to rename in the DB Explorer tree;
- right-click the procedure alias and select the **Rename Procedure <procedure_name>...** item from the context menu;
- edit the procedure name using the **Rename Object...** dialog.

Executing Procedures

To execute a procedure:
- select the procedure to execute in the DB Explorer tree (type the first letters of the procedure name for quick search);
- right-click the object and select the **Edit Procedure <procedure_name>...** context menu item, or simply double-click the procedure;
- execute the procedure using the **Execute Procedure** Navigation bar item of Procedure Editor.

Dropping Procedures

To drop a procedure:
- select the procedure to drop in the DB Explorer tree;
- right-click the object and select the **Drop Procedure <procedure_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new procedure;
Ctrl+O to edit the selected procedure;
Ctrl+R to rename the procedure;
Shift+Del to drop the object from the database.

13.2.4.4 Drop Procedure

A **Procedure** is a set of procedural constructs and embedded SQL statements that is stored in the database and can be called by name.

When you create an application with Microsoft SQL Server 2005, the Transact-SQL programming language is the primary programming interface between your applications and the Microsoft SQL Server database. When you use Transact-SQL programs, two methods are available for storing and executing the programs. You can store the programs locally and create applications that send the commands to SQL Server and process the results, or you can store the programs as stored procedures in SQL Server and create applications that execute the stored procedures and process the results.

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

© 2008 EMS Database Management Solutions

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.

Creating Procedures

To create a new procedure:
- select the **Database | New Object...** main menu item;
- select **Procedure** in the Create New Object dialog;
- define procedure properties using the appropriate tabs of Procedure Editor.

Hint: To create a new procedure, you can also right-click the **Procedures** node or any object within this node in the DB Explorer tree and select the **New Procedure...** item from the context menu.

To create a new procedure with the same properties as one of existing procedures has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a procedure in the DB Explorer tree and select the **Duplicate Procedure <procedure_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new procedure in, and to edit the result SQL statement for creating the procedure.

Editing Procedures

To edit an existing procedure:
- select the procedure for editing in the DB Explorer tree (type the first letters of the procedure name for quick search);
- right-click the object and select the **Edit Procedure <procedure_name>** context menu item, or simply double-click the procedure;
- edit procedure definition using the appropriate tabs of Procedure Editor.

To change the name of a procedure:
- select the procedure to rename in the DB Explorer tree;
- right-click the procedure alias and select the **Rename Procedure <procedure_name>...** item from the context menu;
- edit the procedure name using the **Rename Object...** dialog.

Executing Procedures

To execute a procedure:
- select the procedure to execute in the DB Explorer tree (type the first letters of the procedure name for quick search);
- right-click the object and select the **Edit Procedure <procedure_name>...** context menu item, or simply double-click the procedure;
- execute the procedure using the **Execute Procedure** Navigation bar item of Procedure Editor.

Dropping Procedures

To drop a procedure:
- select the procedure to drop in the DB Explorer tree;

© 2008 EMS Database Management Solutions

- right-click the object and select the **Drop Procedure <procedure_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new procedure;
Ctrl+O to edit the selected procedure;
Ctrl+R to rename the procedure;
Shift+Del to drop the object from the database.

13.2.5 Work with User-Defined Types

13.2.5.1 Create UDT

A **User-Defined Data Type** is internally represented as an existing type, but is considered to be a separate data type. In fact, this is an alias data type in an instance of SQL Server. An alias data type's implementation is based on a SQL Server native system type. A user-defined type is implemented through a class of an assembly in the Microsoft .NET Framework common language runtime (CLR).

Creating User-Defined Types

To create a new user-defined type:
- select the **Database | New Object...** main menu item;
- select **User-Defined Data Type** in the Create New Object dialog;
- define UDT properties using the appropriate tabs of UDT Editor.

Hint: To create a new user-defined type, you can also right-click the **UDTs** node or any object within this node in the DB Explorer tree and select the **New User-Defined Type** item from the context menu.

To create a new UDT with the same properties as one of existing UDTs has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a UDT in the DB Explorer tree and select the **Duplicate User-Defined Type <UDT_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new UDT in, and to edit the result SQL statement for creating the UDT.

Editing User-Defined Types

To edit an existing user-defined type:
- select the user-defined type for editing in the DB Explorer tree (type the first letters of the UDT name for quick search);
- right-click the object and select the **Edit User-Defined Type <UDT_name>** context menu item, or simply double-click the UDT;
- edit UDT definition using the appropriate tabs of UDT Editor.

To change the name of a user-defined type:
- select the user-defined type to rename in the DB Explorer tree;

© 2008 EMS Database Management Solutions

- right-click the UDT alias and select the **Rename User-Defined Type <UDT_name>...** item from the context menu;
- edit the UDT name using the **Rename Object...** dialog.

Dropping User-Defined Types

To drop a user-defined type:
- select the user-defined type to drop in the DB Explorer tree;
- right-click the object and select the **Drop User-Defined Type <UDT_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new UDT;
Ctrl+O to edit the selected UDT;
Ctrl+R to rename the UDT;
Shift+Del to drop the object from the database.

13.2.5.2 Edit UDT

A **User-Defined Data Type** is internally represented as an existing type, but is considered to be a separate data type. In fact, this is an alias data type in an instance of SQL Server. An alias data type's implementation is based on a SQL Server native system type. A user-defined type is implemented through a class of an assembly in the Microsoft .NET Framework common language runtime (CLR).

Creating User-Defined Types

To create a new user-defined type:
- select the **Database | New Object...** main menu item;
- select **User-Defined Data Type** in the Create New Object dialog;
- define UDT properties using the appropriate tabs of UDT Editor.

Hint: To create a new user-defined type, you can also right-click the **UDTs** node or any object within this node in the DB Explorer tree and select the **New User-Defined Type** item from the context menu.

To create a new UDT with the same properties as one of existing UDTs has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a UDT in the DB Explorer tree and select the **Duplicate User-Defined Type <UDT_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new UDT in, and to edit the result SQL statement for creating the UDT.

Editing User-Defined Types

To edit an existing user-defined type:
- select the user-defined type for editing in the DB Explorer tree (type the first letters

© 2008 EMS Database Management Solutions

of the UDT name for quick search);
- right-click the object and select the **Edit User-Defined Type <UDT_name>** context menu item, or simply double-click the UDT;
- edit UDT definition using the appropriate tabs of UDT Editor.

To change the name of a user-defined type:
- select the user-defined type to rename in the DB Explorer tree;
- right-click the UDT alias and select the **Rename User-Defined Type <UDT_name>...** item from the context menu;
- edit the UDT name using the **Rename Object...** dialog.

Dropping User-Defined Types

To drop a user-defined type:
- select the user-defined type to drop in the DB Explorer tree;
- right-click the object and select the **Drop User-Defined Type <UDT_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new UDT;
Ctrl+O to edit the selected UDT;
Ctrl+R to rename the UDT;
Shift+Del to drop the object from the database.

13.2.5.3 Drop UDT

A **User-Defined Data Type** is internally represented as an existing type, but is considered to be a separate data type. In fact, this is an alias data type in an instance of SQL Server. An alias data type's implementation is based on a SQL Server native system type. A user-defined type is implemented through a class of an assembly in the Microsoft .NET Framework common language runtime (CLR).

Creating User-Defined Types

To create a new user-defined type:
- select the **Database | New Object...** main menu item;
- select **User-Defined Data Type** in the Create New Object dialog;
- define UDT properties using the appropriate tabs of UDT Editor.

Hint: To create a new user-defined type, you can also right-click the **UDTs** node or any object within this node in the DB Explorer tree and select the **New User-Defined Type** item from the context menu.

To create a new UDT with the same properties as one of existing UDTs has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a UDT in the DB Explorer tree and select the **Duplicate User-Defined Type <UDT_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new UDT in, and to edit the result SQL statement for creating the UDT.

© 2008 EMS Database Management Solutions

Editing User-Defined Types

To edit an existing user-defined type:
- select the user-defined type for editing in the DB Explorer tree (type the first letters of the UDT name for quick search);
- right-click the object and select the **Edit User-Defined Type <UDT_name>** context menu item, or simply double-click the UDT;
- edit UDT definition using the appropriate tabs of UDT Editor.

To change the name of a user-defined type:
- select the user-defined type to rename in the DB Explorer tree;
- right-click the UDT alias and select the **Rename User-Defined Type <UDT_name>...** item from the context menu;
- edit the UDT name using the **Rename Object...** dialog.

Dropping User-Defined Types

To drop a user-defined type:
- select the user-defined type to drop in the DB Explorer tree;
- right-click the object and select the **Drop User-Defined Type <UDT_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new UDT;
Ctrl+O to edit the selected UDT;
Ctrl+R to rename the UDT;
Shift+Del to drop the object from the database.

13.2.6 Work with User-Defined Functions

13.2.6.1 Create UDF

A **Function** is a mapping embodied as a program (the function *body*) that can be invoked by using zero or more input values (*arguments*) to a single value (the *result*). A **User-Defined Function** is stored as a database object providing reusable code.

The Transact-SQL programming language provides the following types of functions:
Rowset functions (can be used like table references in an SQL statement)
Aggregate functions (operate on a collection of values but return a single, summarizing value)
Ranking functions (return a ranking value for each row in a partition)
Scalar functions (operate on a single value and then return a single value; scalar functions can be used wherever an expression is valid)

Creating User-Defined Functions

To create a new function:
- select the **Database | New Object...** main menu item;

© 2008 EMS Database Management Solutions

- select **User-Defined Function** in the Create New Object dialog;
- define function properties using the appropriate tabs of UDF Editor.

Hint: To create a new function, you can also right-click the **UDFs** node or any object within this node in the DB Explorer tree and select the **New User-Defined Function** item from the context menu.

To create a new function with the same properties as one of existing functions has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a function in the DB Explorer tree and select the **Duplicate User-Defined Function <UDF_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new function in, and to edit the result SQL statement for creating the function.

Editing User-Defined Functions

To edit an existing function:
- select the function for editing in the DB Explorer tree (type the first letters of the function name for quick search);
- right-click the object and select the **Edit User-Defined Function <UDF_name>** context menu item, or simply double-click the function;
- edit function definition using the appropriate tabs of UDF Editor.

To change the name of a UDF:
- select the UDF to rename in the DB Explorer tree;
- right-click the UDF alias and select the **Rename User-Defined Function <UDF_name>...** item from the context menu;
- edit the UDF name using the **Rename Object...** dialog.

Dropping User-Defined Functions

To drop a function:
- select the function to drop in the DB Explorer tree;
- right-click the object and select the **Drop User-Defined Function <UDF_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new UDF;
Ctrl+O to edit the selected UDF;
Ctrl+R to rename the UDF;
Shift+Del to drop the object from the database.

13.2.6.2 Edit UDF

A **Function** is a mapping embodied as a program (the function *body*) that can be invoked by using zero or more input values (*arguments*) to a single value (the *result*). A **User-Defined Function** is stored as a database object providing reusable code.

The Transact-SQL programming language provides the following types of functions:

© 2008 EMS Database Management Solutions

Rowset functions (can be used like table references in an SQL statement)
Aggregate functions (operate on a collection of values but return a single, summarizing value)
Ranking functions (return a ranking value for each row in a partition)
Scalar functions (operate on a single value and then return a single value; scalar functions can be used wherever an expression is valid)

Creating User-Defined Functions

To create a new function:
- select the **Database | New Object...** main menu item;
- select **User-Defined Function** in the Create New Object dialog;
- define function properties using the appropriate tabs of UDF Editor.

Hint: To create a new function, you can also right-click the **UDFs** node or any object within this node in the DB Explorer tree and select the **New User-Defined Function** item from the context menu.

To create a new function with the same properties as one of existing functions has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a function in the DB Explorer tree and select the **Duplicate User-Defined Function <UDF_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new function in, and to edit the result SQL statement for creating the function.

Editing User-Defined Functions

To edit an existing function:
- select the function for editing in the DB Explorer tree (type the first letters of the function name for quick search);
- right-click the object and select the **Edit User-Defined Function <UDF_name>** context menu item, or simply double-click the function;
- edit function definition using the appropriate tabs of UDF Editor.

To change the name of a UDF:
- select the UDF to rename in the DB Explorer tree;
- right-click the UDF alias and select the **Rename User-Defined Function <UDF_name>...** item from the context menu;
- edit the UDF name using the **Rename Object...** dialog.

Dropping User-Defined Functions

To drop a function:
- select the function to drop in the DB Explorer tree;
- right-click the object and select the **Drop User-Defined Function <UDF_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new UDF;
Ctrl+O to edit the selected UDF;

© 2008 EMS Database Management Solutions

Ctrl+R to rename the UDF;
Shift+Del to drop the object from the database.

13.2.6.3 Drop UDF

A **Function** is a mapping embodied as a program (the function *body*) that can be invoked by using zero or more input values (*arguments*) to a single value (the *result*). A **User-Defined Function** is stored as a database object providing reusable code.

The Transact-SQL programming language provides the following types of functions:
Rowset functions (can be used like table references in an SQL statement)
Aggregate functions (operate on a collection of values but return a single, summarizing value)
Ranking functions (return a ranking value for each row in a partition)
Scalar functions (operate on a single value and then return a single value; scalar functions can be used wherever an expression is valid)

Creating User-Defined Functions

To create a new function:
- select the **Database | New Object...** main menu item;
- select **User-Defined Function** in the Create New Object dialog;
- define function properties using the appropriate tabs of UDF Editor.

Hint: To create a new function, you can also right-click the **UDFs** node or any object within this node in the DB Explorer tree and select the **New User-Defined Function** item from the context menu.

To create a new function with the same properties as one of existing functions has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a function in the DB Explorer tree and select the **Duplicate User-Defined Function <UDF_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new function in, and to edit the result SQL statement for creating the function.

Editing User-Defined Functions

To edit an existing function:
- select the function for editing in the DB Explorer tree (type the first letters of the function name for quick search);
- right-click the object and select the **Edit User-Defined Function <UDF_name>** context menu item, or simply double-click the function;
- edit function definition using the appropriate tabs of UDF Editor.

To change the name of a UDF:
- select the UDF to rename in the DB Explorer tree;
- right-click the UDF alias and select the **Rename User-Defined Function <UDF_name>...** item from the context menu;
- edit the UDF name using the **Rename Object...** dialog.

© 2008 EMS Database Management Solutions

Dropping User-Defined Functions

To drop a function:
- select the function to drop in the DB Explorer tree;
- right-click the object and select the **Drop User-Defined Function <UDF_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new UDF;
Ctrl+O to edit the selected UDF;
Ctrl+R to rename the UDF;
Shift+Del to drop the object from the database.

13.2.7 Work with Synonyms

13.2.7.1 Create Synonym

You can use **synonyms** in place of their referenced base object in a number of SQL statements and expression contexts. When working with synonyms in these contexts, the base object is affected. For example, if you have a synonym that references a base object that is a table, when you insert a row into the synonym, you are actually inserting a row into the referenced table.

Creating Synonyms

To create a new synonym:
- select the **Database | New Object...** main menu item;
- select **Synonym** in the Create New Object dialog;
- define synonym properties using the appropriate tabs of Synonym Editor.

Hint: To create a new synonym, you can also right-click the **Synonyms** node or any object within this node in the DB Explorer tree and select the **New Synonym** item from the context menu.

To create a new synonym with the same properties as one of existing synonyms has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a synonym in the DB Explorer tree and select the **Duplicate Synonym <synonym_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new synonym in, and to edit the result SQL statement for creating the synonym.

Editing Synonyms

To edit an existing synonym:
- select the synonym for editing in the DB Explorer tree (type the first letters of the synonym name for quick search);

© 2008 EMS Database Management Solutions

- right-click the object and select the **Edit Synonym <synonym_name>** context menu item, or simply double-click the synonym;
- edit synonym properties using the appropriate tabs of Synonym Editor.

To change the name of a synonym:

- select the synonym to rename in the DB Explorer tree;
- right-click the synonym alias and select the **Rename Synonym <synonym_name>...** item from the context menu;
- edit the synonym name using the **Rename Object...** dialog.

Dropping Synonyms

To drop a synonym:

- select the synonym to drop in the DB Explorer tree;
- right-click the object and select the **Drop Synonym <synonym_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new synonym;
Ctrl+O to edit the selected synonym;
Ctrl+R to rename the synonym;
Shift+Del to drop the object from the database.

13.2.7.2 Edit Synonym

You can use **synonyms** in place of their referenced base object in a number of SQL statements and expression contexts. When working with synonyms in these contexts, the base object is affected. For example, if you have a synonym that references a base object that is a table, when you insert a row into the synonym, you are actually inserting a row into the referenced table.

Creating Synonyms

To create a new synonym:

- select the **Database | New Object...** main menu item;
- select **Synonym** in the Create New Object dialog;
- define synonym properties using the appropriate tabs of Synonym Editor.

Hint: To create a new synonym, you can also right-click the **Synonyms** node or any object within this node in the DB Explorer tree and select the **New Synonym** item from the context menu.

To create a new synonym with the same properties as one of existing synonyms has:

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

Alternatively, you can right-click a synonym in the DB Explorer tree and select the **Duplicate Synonym <synonym_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new synonym in, and to edit the result SQL statement for creating the synonym.

© 2008 EMS Database Management Solutions

Editing Synonyms

To edit an existing synonym:
- select the synonym for editing in the DB Explorer tree (type the first letters of the synonym name for quick search);
- right-click the object and select the **Edit Synonym <synonym_name>** context menu item, or simply double-click the synonym;
- edit synonym properties using the appropriate tabs of Synonym Editor.

To change the name of a synonym:
- select the synonym to rename in the DB Explorer tree;
- right-click the synonym alias and select the **Rename Synonym <synonym_name>...** item from the context menu;
- edit the synonym name using the **Rename Object...** dialog.

Dropping Synonyms

To drop a synonym:
- select the synonym to drop in the DB Explorer tree;
- right-click the object and select the **Drop Synonym <synonym_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new synonym;
Ctrl+O to edit the selected synonym;
Ctrl+R to rename the synonym;
Shift+Del to drop the object from the database.

13.2.7.3 Drop Synonym

You can use **synonyms** in place of their referenced base object in a number of SQL statements and expression contexts. When working with synonyms in these contexts, the base object is affected. For example, if you have a synonym that references a base object that is a table, when you insert a row into the synonym, you are actually inserting a row into the referenced table.

Creating Synonyms

To create a new synonym:
- select the **Database | New Object...** main menu item;
- select **Synonym** in the Create New Object dialog;
- define synonym properties using the appropriate tabs of Synonym Editor.

Hint: To create a new synonym, you can also right-click the **Synonyms** node or any object within this node in the DB Explorer tree and select the **New Synonym** item from the context menu.

To create a new synonym with the same properties as one of existing synonyms has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

© 2008 EMS Database Management Solutions

Alternatively, you can right-click a synonym in the DB Explorer tree and select the **Duplicate Synonym <synonym_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new synonym in, and to edit the result SQL statement for creating the synonym.

Editing Synonyms

To edit an existing synonym:
- select the synonym for editing in the DB Explorer tree (type the first letters of the synonym name for quick search);
- right-click the object and select the **Edit Synonym <synonym_name>** context menu item, or simply double-click the synonym;
- edit synonym properties using the appropriate tabs of Synonym Editor.

To change the name of a synonym:
- select the synonym to rename in the DB Explorer tree;
- right-click the synonym alias and select the **Rename Synonym <synonym_name>...** item from the context menu;
- edit the synonym name using the **Rename Object...** dialog.

Dropping Synonyms

To drop a synonym:
- select the synonym to drop in the DB Explorer tree;
- right-click the object and select the **Drop Synonym <synonym_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new synonym;
Ctrl+O to edit the selected synonym;
Ctrl+R to rename the synonym;
Shift+Del to drop the object from the database.

13.2.8 Work with Aggregates

13.2.8.1 Create Aggregate

Aggregate functions, or **aggregates**, perform a calculation on a set of values and return a single value. Traditionally Microsoft® SQL Server supported only built-in aggregate functions, such as *SUM* or *MAX*, that operate on a set of input scalar values and generate a single aggregate value from that set. SQL Server integration with the .NET Framework common language runtime (CLR) now allows developers to create custom aggregate functions in managed code and make these functions accessible to Transact-SQL or other managed code.

Creating Aggregates

To create a new aggregate:

© 2008 EMS Database Management Solutions

- select the **Database | New Object...** main menu item;
- select **Aggregate** in the Create New Object dialog;
- define aggregate properties using the appropriate tabs of Aggregate Editor.

Hint: To create a new aggregate, you can also right-click the **Aggregates** node or any object within this node in the DB Explorer tree and select the **New Aggregate** item from the context menu.

To create a new aggregate with the same properties as one of existing aggregates has:

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

Alternatively, you can right-click a aggregate in the DB Explorer tree and select the **Duplicate Aggregate <aggregate_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new aggregate in, and to edit the result SQL statement for creating the aggregate.

Editing Aggregates

To edit an existing aggregate:

- select the aggregate for editing in the DB Explorer tree (type the first letters of the aggregate name for quick search);
- right-click the object and select the **Edit Aggregate <aggregate_name>** context menu item, or simply double-click the aggregate;
- edit aggregate properties using the appropriate tabs of Aggregate Editor.

To change the name of an aggregate:

- select the aggregate to rename in the DB Explorer tree;
- right-click the aggregate alias and select the **Rename Aggregate <aggregate_name>...** item from the context menu;
- edit the aggregate name using the **Rename Object...** dialog.

Dropping Aggregates

To drop an aggregate:

- select the aggregate to drop in the DB Explorer tree;
- right-click the object and select the **Drop Aggregate <aggregate_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new aggregate;
Ctrl+O to edit the selected aggregate;
Ctrl+R to rename the aggregate;
Shift+Del to drop the object from the database.

13.2.8.2 Edit Aggregate

Aggregate functions, or **aggregates**, perform a calculation on a set of values and return a single value. Traditionally Microsoft® SQL Server supported only built-in aggregate functions, such as *SUM* or *MAX*, that operate on a set of input scalar values and generate a single aggregate value from that set. SQL Server integration with the .NET Framework common language runtime (CLR) now allows developers to create custom

© 2008 EMS Database Management Solutions

aggregate functions in managed code and make these functions accessible to Transact-SQL or other managed code.

Creating Aggregates

To create a new aggregate:
- select the **Database | New Object...** main menu item;
- select **Aggregate** in the Create New Object dialog;
- define aggregate properties using the appropriate tabs of Aggregate Editor.

Hint: To create a new aggregate, you can also right-click the **Aggregates** node or any object within this node in the DB Explorer tree and select the **New Aggregate** item from the context menu.

To create a new aggregate with the same properties as one of existing aggregates has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a aggregate in the DB Explorer tree and select the **Duplicate Aggregate <aggregate_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new aggregate in, and to edit the result SQL statement for creating the aggregate.

Editing Aggregates

To edit an existing aggregate:
- select the aggregate for editing in the DB Explorer tree (type the first letters of the aggregate name for quick search);
- right-click the object and select the **Edit Aggregate <aggregate_name>** context menu item, or simply double-click the aggregate;
- edit aggregate properties using the appropriate tabs of Aggregate Editor.

To change the name of an aggregate:
- select the aggregate to rename in the DB Explorer tree;
- right-click the aggregate alias and select the **Rename Aggregate <aggregate_name>...** item from the context menu;
- edit the aggregate name using the **Rename Object...** dialog.

Dropping Aggregates

To drop an aggregate:
- select the aggregate to drop in the DB Explorer tree;
- right-click the object and select the **Drop Aggregate <aggregate_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new aggregate;
Ctrl+O to edit the selected aggregate;
Ctrl+R to rename the aggregate;
Shift+Del to drop the object from the database.

© 2008 EMS Database Management Solutions

13.2.8.3 Drop Aggregate

Aggregate functions, or **aggregates**, perform a calculation on a set of values and return a single value. Traditionally Microsoft® SQL Server supported only built-in aggregate functions, such as *SUM* or *MAX*, that operate on a set of input scalar values and generate a single aggregate value from that set. SQL Server integration with the .NET Framework common language runtime (CLR) now allows developers to create custom aggregate functions in managed code and make these functions accessible to Transact-SQL or other managed code.

Creating Aggregates

To create a new aggregate:
- select the **Database | New Object...** main menu item;
- select **Aggregate** in the Create New Object dialog;
- define aggregate properties using the appropriate tabs of Aggregate Editor.

Hint: To create a new aggregate, you can also right-click the **Aggregates** node or any object within this node in the DB Explorer tree and select the **New Aggregate** item from the context menu.

To create a new aggregate with the same properties as one of existing aggregates has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a aggregate in the DB Explorer tree and select the **Duplicate Aggregate <aggregate_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new aggregate in, and to edit the result SQL statement for creating the aggregate.

Editing Aggregates

To edit an existing aggregate:
- select the aggregate for editing in the DB Explorer tree (type the first letters of the aggregate name for quick search);
- right-click the object and select the **Edit Aggregate <aggregate_name>** context menu item, or simply double-click the aggregate;
- edit aggregate properties using the appropriate tabs of Aggregate Editor.

To change the name of an aggregate:
- select the aggregate to rename in the DB Explorer tree;
- right-click the aggregate alias and select the **Rename Aggregate <aggregate_name>...** item from the context menu;
- edit the aggregate name using the **Rename Object...** dialog.

Dropping Aggregates

To drop an aggregate:
- select the aggregate to drop in the DB Explorer tree;
- right-click the object and select the **Drop Aggregate <aggregate_name>...** context menu item;
- confirm dropping in the dialog window.

© 2008 EMS Database Management Solutions

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new aggregate;
Ctrl+O to edit the selected aggregate;
Ctrl+R to rename the aggregate;
Shift+Del to drop the object from the database.

13.2.9 Work with XML Schema Collections

13.2.9.1 Create XML Schema Collection

Microsoft® SQL Server™ provides native storage of XML data through the XML data type. 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 they are stored in the database

The XML stored in a column or variable that a schema is associated with is referred to as typed XML, because the schema provides the necessary data type information for the instance data. SQL Server uses this type information to optimize data storage.

Note that XML Schema Collection is a metadata entity like a table in the database, hence the XML Schema Collection can be created, modified and dropped.

Creating XML Schema Collections

To create a new XML Schema Collection:

- select the **Database | New Object...** main menu item;
- select **XML Schema Collection** in the Create New Object dialog;
- define XML Schema Collection properties using the appropriate tabs of XML Schema Collection Editor.

Hint: To create a new XML Schema Collection, you can also right-click the **XML Schema Collections** node or any object within this node in the DB Explorer tree and select the **New XML Schema Collection** item from the context menu.

To create a new XML Schema Collection with the same properties as one of existing XML Schema Collections has:

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

Alternatively, you can right-click a XML Schema Collection in the DB Explorer tree and select the **Duplicate XML Schema Collection <XML_Schema_Collection_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new XML Schema Collection in, and to edit the result SQL statement for creating the XML Schema Collection.

Editing XML Schema Collections

To edit an existing XML Schema Collection:

- select the XML Schema Collection for editing in the DB Explorer tree (type the first letters of the XML Schema Collection name for quick search);

© 2008 EMS Database Management Solutions

- right-click the object and select the **Edit XML Schema Collection <XML_Schema_Collection_name>** context menu item, or simply double-click the XML Schema Collection;
- edit XML Schema Collection properties using the appropriate tabs of XML Schema Collection Editor.

To change the name of a XML Schema Collection:

- select the XML Schema Collection to rename in the DB Explorer tree;
- right-click the XML Schema Collection alias and select the **Rename XML Schema Collection <XML_Schema_Collection_name>...** item from the context menu;
- edit the XML Schema Collection name using the **Rename Object...** dialog.

Dropping XML Schema Collections

To drop a XML Schema Collection:

- select the XML Schema Collection to drop in the DB Explorer tree;
- right-click the object and select the **Drop XML Schema Collection <XML_Schema_Collection_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new XML Schema Collection;
Ctrl+O to edit the selected XML Schema Collection;
Ctrl+R to rename the XML Schema Collection;
Shift+Del to drop the object from the database.

13.2.9.2 Edit XML Schema Collection

Microsoft® SQL Server™ provides native storage of XML data through the XML data type. 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 they are stored in the database

The XML stored in a column or variable that a schema is associated with is referred to as typed XML, because the schema provides the necessary data type information for the instance data. SQL Server uses this type information to optimize data storage.

Note that XML Schema Collection is a metadata entity like a table in the database, hence the XML Schema Collection can be created, modified and dropped.

Creating XML Schema Collections

To create a new XML Schema Collection:

- select the **Database | New Object...** main menu item;
- select **XML Schema Collection** in the Create New Object dialog;
- define XML Schema Collection properties using the appropriate tabs of XML Schema Collection Editor.

Hint: To create a new XML Schema Collection, you can also right-click the **XML Schema Collections** node or any object within this node in the DB Explorer tree and select the **New XML Schema Collection** item from the context menu.

© 2008 EMS Database Management Solutions

To create a new XML Schema Collection with the same properties as one of existing XML Schema Collections has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a XML Schema Collection in the DB Explorer tree and select the **Duplicate XML Schema Collection <XML_Schema_Collection_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new XML Schema Collection in, and to edit the result SQL statement for creating the XML Schema Collection.

Editing XML Schema Collections

To edit an existing XML Schema Collection:
- select the XML Schema Collection for editing in the DB Explorer tree (type the first letters of the XML Schema Collection name for quick search);
- right-click the object and select the **Edit XML Schema Collection <XML_Schema_Collection_name>** context menu item, or simply double-click the XML Schema Collection;
- edit XML Schema Collection properties using the appropriate tabs of XML Schema Collection Editor.

To change the name of a XML Schema Collection:
- select the XML Schema Collection to rename in the DB Explorer tree;
- right-click the XML Schema Collection alias and select the **Rename XML Schema Collection <XML_Schema_Collection_name>...** item from the context menu;
- edit the XML Schema Collection name using the **Rename Object...** dialog.

Dropping XML Schema Collections

To drop a XML Schema Collection:
- select the XML Schema Collection to drop in the DB Explorer tree;
- right-click the object and select the **Drop XML Schema Collection <XML_Schema_Collection_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new XML Schema Collection;
Ctrl+O to edit the selected XML Schema Collection;
Ctrl+R to rename the XML Schema Collection;
Shift+Del to drop the object from the database.

13.2.9.3 Drop XML Schema Collection

Microsoft® SQL Server™ provides native storage of XML data through the XML data type. 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 they are stored in the database

The XML stored in a column or variable that a schema is associated with is referred to as

© 2008 EMS Database Management Solutions

typed XML, because the schema provides the necessary data type information for the instance data. SQL Server uses this type information to optimize data storage.

Note that XML Schema Collection is a metadata entity like a table in the database, hence the XML Schema Collection can be created, modified and dropped.

Creating XML Schema Collections

To create a new XML Schema Collection:
- select the **Database | New Object...** main menu item;
- select **XML Schema Collection** in the Create New Object dialog;
- define XML Schema Collection properties using the appropriate tabs of XML Schema Collection Editor.

Hint: To create a new XML Schema Collection, you can also right-click the **XML Schema Collections** node or any object within this node in the DB Explorer tree and select the **New XML Schema Collection** item from the context menu.

To create a new XML Schema Collection with the same properties as one of existing XML Schema Collections has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a XML Schema Collection in the DB Explorer tree and select the **Duplicate XML Schema Collection <XML_Schema_Collection_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new XML Schema Collection in, and to edit the result SQL statement for creating the XML Schema Collection.

Editing XML Schema Collections

To edit an existing XML Schema Collection:
- select the XML Schema Collection for editing in the DB Explorer tree (type the first letters of the XML Schema Collection name for quick search);
- right-click the object and select the **Edit XML Schema Collection <XML_Schema_Collection_name>** context menu item, or simply double-click the XML Schema Collection;
- edit XML Schema Collection properties using the appropriate tabs of XML Schema Collection Editor.

To change the name of a XML Schema Collection:
- select the XML Schema Collection to rename in the DB Explorer tree;
- right-click the XML Schema Collection alias and select the **Rename XML Schema Collection <XML_Schema_Collection_name>...** item from the context menu;
- edit the XML Schema Collection name using the **Rename Object...** dialog.

Dropping XML Schema Collections

To drop a XML Schema Collection:
- select the XML Schema Collection to drop in the DB Explorer tree;
- right-click the object and select the **Drop XML Schema Collection <XML_Schema_Collection_name>...** context menu item;

© 2008 EMS Database Management Solutions

- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new XML Schema Collection;
Ctrl+O to edit the selected XML Schema Collection;
Ctrl+R to rename the XML Schema Collection;
Shift+Del to drop the object from the database.

13.2.10 Work with Defaults

13.2.10.1 Create Default

When bound to a column or a user-defined data type, a **default** specifies a value to be inserted into the column to which the object is bound (or into all columns, in case of UDF) when no value is explicitly supplied during an insert operation.

Creating Defaults

To create a new default:
- select the **Database | New Object...** main menu item;
- select **Default** in the Create New Object dialog;
- define default properties using the appropriate tabs of Default Editor.

Hint: To create a new default, you can also right-click the **Defaults** node or any object within this node in the DB Explorer tree and select the **New Default** item from the context menu.

To create a new default with the same properties as one of existing defaults has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a default in the DB Explorer tree and select the **Duplicate Default <default_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new default in, and to edit the result SQL statement for creating the default.

Editing Defaults

To edit an existing default:
- select the default for editing in the DB Explorer tree (type the first letters of the default name for quick search);
- right-click the object and select the **Edit Default <default_name>** context menu item, or simply double-click the default;
- edit default properties using the appropriate tabs of Default Editor.

To change the name of a default:
- select the default to rename in the DB Explorer tree;
- right-click the default alias and select the **Rename Default <default_name>...** item from the context menu;
- edit the default name using the **Rename Object...** dialog.

© 2008 EMS Database Management Solutions

Dropping Defaults

To drop a default:
- select the default to drop in the DB Explorer tree;
- right-click the object and select the **Drop Default <default_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new default;
Ctrl+O to edit the selected default;
Ctrl+R to rename the default;
Shift+Del to drop the object from the database.

13.2.10.2 Edit Default

When bound to a column or a user-defined data type, a **default** specifies a value to be inserted into the column to which the object is bound (or into all columns, in case of UDF) when no value is explicitly supplied during an insert operation.

Creating Defaults

To create a new default:
- select the **Database | New Object...** main menu item;
- select **Default** in the Create New Object dialog;
- define default properties using the appropriate tabs of Default Editor.

Hint: To create a new default, you can also right-click the **Defaults** node or any object within this node in the DB Explorer tree and select the **New Default** item from the context menu.

To create a new default with the same properties as one of existing defaults has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a default in the DB Explorer tree and select the **Duplicate Default <default_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new default in, and to edit the result SQL statement for creating the default.

Editing Defaults

To edit an existing default:
- select the default for editing in the DB Explorer tree (type the first letters of the default name for quick search);
- right-click the object and select the **Edit Default <default_name>** context menu item, or simply double-click the default;
- edit default properties using the appropriate tabs of Default Editor.

To change the name of a default:
- select the default to rename in the DB Explorer tree;

© 2008 EMS Database Management Solutions

- right-click the default alias and select the **Rename Default <default_name>...** item from the context menu;
- edit the default name using the **Rename Object...** dialog.

Dropping Defaults

To drop a default:
- select the default to drop in the DB Explorer tree;
- right-click the object and select the **Drop Default <default_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new default;
Ctrl+O to edit the selected default;
Ctrl+R to rename the default;
Shift+Del to drop the object from the database.

13.2.10.3 Drop Default

When bound to a column or a user-defined data type, a **default** specifies a value to be inserted into the column to which the object is bound (or into all columns, in case of UDF) when no value is explicitly supplied during an insert operation.

Creating Defaults

To create a new default:
- select the **Database | New Object...** main menu item;
- select **Default** in the Create New Object dialog;
- define default properties using the appropriate tabs of Default Editor.

Hint: To create a new default, you can also right-click the **Defaults** node or any object within this node in the DB Explorer tree and select the **New Default** item from the context menu.

To create a new default with the same properties as one of existing defaults has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a default in the DB Explorer tree and select the **Duplicate Default <default_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new default in, and to edit the result SQL statement for creating the default.

Editing Defaults

To edit an existing default:
- select the default for editing in the DB Explorer tree (type the first letters of the default name for quick search);
- right-click the object and select the **Edit Default <default_name>** context menu

© 2008 EMS Database Management Solutions

item, or simply double-click the default;
- edit default properties using the appropriate tabs of Default Editor.

To change the name of a default:
- select the default to rename in the DB Explorer tree;
- right-click the default alias and select the **Rename Default <default_name>...** item from the context menu;
- edit the default name using the **Rename Object...** dialog.

Dropping Defaults

To drop a default:
- select the default to drop in the DB Explorer tree;
- right-click the object and select the **Drop Default <default_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new default;
Ctrl+O to edit the selected default;
Ctrl+R to rename the default;
Shift+Del to drop the object from the database.

13.2.11 Work with Rules

13.2.11.1 Create Rule

When bound to a column or a user-defined data type, a **rule** specifies the acceptable values that can be inserted into that column or to the fields having the specified data type.

Creating Rules

To create a new rule:
- select the **Database | New Object...** main menu item;
- select **Rule** in the Create New Object dialog;
- define rule properties using the appropriate tabs of Rule Editor.

Hint: To create a new rule, you can also right-click the **Rules** node or any object within this node in the DB Explorer tree and select the **New rule** item from the context menu.

To create a new rule with the same properties as one of existing rules has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a rule in the DB Explorer tree and select the **Duplicate Rule <rule_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new rule in, and to edit the result SQL statement for creating the rule.

Editing Rules

© 2008 EMS Database Management Solutions

To edit an existing rule:
- select the rule for editing in the DB Explorer tree (type the first letters of the rule name for quick search);
- right-click the object and select the **Edit Rule <rule_name>** context menu item, or simply double-click the rule;
- edit rule properties using the appropriate tabs of Rule Editor.

To change the name of a rule:
- select the rule to rename in the DB Explorer tree;
- right-click the rule alias and select the **Rename Rule <rule_name>...** item from the context menu;
- edit the rule name using the **Rename Object...** dialog.

Dropping Rules

To drop a rule:
- select the rule to drop in the DB Explorer tree;
- right-click the object and select the **Drop Rule <rule_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new rule;
Ctrl+O to edit the selected rule;
Ctrl+R to rename the rule;
Shift+Del to drop the object from the database.

13.2.11.2 Edit Rule

When bound to a column or a user-defined data type, a **rule** specifies the acceptable values that can be inserted into that column or to the fields having the specified data type.

Creating Rules

To create a new rule:
- select the **Database | New Object...** main menu item;
- select **Rule** in the Create New Object dialog;
- define rule properties using the appropriate tabs of Rule Editor.

Hint: To create a new rule, you can also right-click the **Rules** node or any object within this node in the DB Explorer tree and select the **New rule** item from the context menu.

To create a new rule with the same properties as one of existing rules has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a rule in the DB Explorer tree and select the **Duplicate Rule <rule_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new rule in, and to edit the result SQL statement for creating the rule.

© 2008 EMS Database Management Solutions

Editing Rules

To edit an existing rule:
- select the rule for editing in the DB Explorer tree (type the first letters of the rule name for quick search);
- right-click the object and select the **Edit Rule <rule_name>** context menu item, or simply double-click the rule;
- edit rule properties using the appropriate tabs of Rule Editor.

To change the name of a rule:
- select the rule to rename in the DB Explorer tree;
- right-click the rule alias and select the **Rename Rule <rule_name>...** item from the context menu;
- edit the rule name using the **Rename Object...** dialog.

Dropping Rules

To drop a rule:
- select the rule to drop in the DB Explorer tree;
- right-click the object and select the **Drop Rule <rule_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new rule;
Ctrl+O to edit the selected rule;
Ctrl+R to rename the rule;
Shift+Del to drop the object from the database.

13.2.11.3 Drop Rule

When bound to a column or a user-defined data type, a **rule** specifies the acceptable values that can be inserted into that column or to the fields having the specified data type.

Creating Rules

To create a new rule:
- select the **Database | New Object...** main menu item;
- select **Rule** in the Create New Object dialog;
- define rule properties using the appropriate tabs of Rule Editor.

Hint: To create a new rule, you can also right-click the **Rules** node or any object within this node in the DB Explorer tree and select the **New rule** item from the context menu.

To create a new rule with the same properties as one of existing rules has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a rule in the DB Explorer tree and select the **Duplicate Rule <rule_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new rule in, and to

© 2008 EMS Database Management Solutions

edit the result SQL statement for creating the rule.

Editing Rules

To edit an existing rule:
- select the rule for editing in the DB Explorer tree (type the first letters of the rule name for quick search);
- right-click the object and select the **Edit Rule <rule_name>** context menu item, or simply double-click the rule;
- edit rule properties using the appropriate tabs of Rule Editor.

To change the name of a rule:
- select the rule to rename in the DB Explorer tree;
- right-click the rule alias and select the **Rename Rule <rule_name>...** item from the context menu;
- edit the rule name using the **Rename Object...** dialog.

Dropping Rules

To drop a rule:
- select the rule to drop in the DB Explorer tree;
- right-click the object and select the **Drop Rule <rule_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new rule;
Ctrl+O to edit the selected rule;
Ctrl+R to rename the rule;
Shift+Del to drop the object from the database.

13.2.12 Work with Assemblies

13.2.12.1 Create Assembly

An **assembly** is a managed application module containing class metadata and managed code as an object in SQL Server, against which CLR functions, stored procedures, triggers, user-defined aggregates, and user-defined types can be created in SQL Server.

Creating Assemblies

To create a new assembly:
- select the **Database | New Object...** main menu item;
- select **Assembly** in the Create New Object dialog;
- define assembly properties using the appropriate tabs of Assembly Editor.

Hint: To create a new assembly, you can also right-click the **Assemblies** node or any object within this node in the DB Explorer tree and select the **New Assembly** item from the context menu.

To create a new assembly with the same properties as one of existing assemblys has:
- select the **Database | Duplicate Object...** main menu item;

© 2008 EMS Database Management Solutions

- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a assembly in the DB Explorer tree and select the **Duplicate Assembly <assembly_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new assembly in, and to edit the result SQL statement for creating the assembly.

Editing Assemblies

To edit an existing assembly:

- select the assembly for editing in the DB Explorer tree (type the first letters of the assembly name for quick search);
- right-click the object and select the **Edit Assembly <assembly_name>** context menu item, or simply double-click the assembly;
- edit assembly properties using the appropriate tabs of Assembly Editor.

Dropping Assemblies

To drop a assembly:

- select the assembly to drop in the DB Explorer tree;
- right-click the object and select the **Drop Assembly <assembly_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new assembly;
Ctrl+O to edit the selected assembly;
Shift+Del to drop the object from the database.

13.2.12.2 Edit Assembly

An **assembly** is a managed application module containing class metadata and managed code as an object in SQL Server, against which CLR functions, stored procedures, triggers, user-defined aggregates, and user-defined types can be created in SQL Server.

Creating Assemblies

To create a new assembly:

- select the **Database | New Object...** main menu item;
- select **Assembly** in the Create New Object dialog;
- define assembly properties using the appropriate tabs of Assembly Editor.

Hint: To create a new assembly, you can also right-click the **Assemblies** node or any object within this node in the DB Explorer tree and select the **New Assembly** item from the context menu.

To create a new assembly with the same properties as one of existing assemblys has:

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

Alternatively, you can right-click a assembly in the DB Explorer tree and select the **Duplicate Assembly <assembly_name>...** context menu item.

© 2008 EMS Database Management Solutions

Duplicate Object Wizard allows you to select the database to create a new assembly in, and to edit the result SQL statement for creating the assembly.

Editing Assemblies

To edit an existing assembly:
- select the assembly for editing in the DB Explorer tree (type the first letters of the assembly name for quick search);
- right-click the object and select the **Edit Assembly <assembly_name>** context menu item, or simply double-click the assembly;
- edit assembly properties using the appropriate tabs of Assembly Editor.

Dropping Assemblies

To drop a assembly:
- select the assembly to drop in the DB Explorer tree;
- right-click the object and select the **Drop Assembly <assembly_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new assembly;
Ctrl+O to edit the selected assembly;
Shift+Del to drop the object from the database.

13.2.12.3 Drop Assembly

An **assembly** is a managed application module containing class metadata and managed code as an object in SQL Server, against which CLR functions, stored procedures, triggers, user-defined aggregates, and user-defined types can be created in SQL Server.

Creating Assemblies

To create a new assembly:
- select the **Database | New Object...** main menu item;
- select **Assembly** in the Create New Object dialog;
- define assembly properties using the appropriate tabs of Assembly Editor.

Hint: To create a new assembly, you can also right-click the **Assemblies** node or any object within this node in the DB Explorer tree and select the **New Assembly** item from the context menu.

To create a new assembly with the same properties as one of existing assemblys has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a assembly in the DB Explorer tree and select the **Duplicate Assembly <assembly_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new assembly in, and to edit the result SQL statement for creating the assembly.

© 2008 EMS Database Management Solutions

Editing Assemblies

To edit an existing assembly:
- select the assembly for editing in the DB Explorer tree (type the first letters of the assembly name for quick search);
- right-click the object and select the **Edit Assembly <assembly_name>** context menu item, or simply double-click the assembly;
- edit assembly properties using the appropriate tabs of Assembly Editor.

Dropping Assemblies

To drop a assembly:
- select the assembly to drop in the DB Explorer tree;
- right-click the object and select the **Drop Assembly <assembly_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new assembly;
Ctrl+O to edit the selected assembly;
Shift+Del to drop the object from the database.

13.2.13 Work with DDL Triggers

13.2.13.1 Create DDL Trigger

A **trigger** is a special kind of stored procedure that automatically executes when an event occurs in the database.

DDL triggers fire in response to a variety of data definition language (DDL) events which are primarily *CREATE*, *ALTER* and *DROP* statements; logon triggers fire in response to the *LOGON* event that is raised when a user sessions is being established.

Creating DDL Triggers

To create a new DDL Trigger:
- select the **Database | New Object...** main menu item;
- select **DDL Trigger** in the Create New Object dialog;
- define DDL Trigger properties using the appropriate tabs of DDL Trigger Editor.

Hint: To create a new DDL Trigger, you can also right-click the **DDL Triggers** node or any object within this node in the DB Explorer tree and select the **New DDL Trigger** item from the context menu.

To create a new DDL Trigger with the same properties as one of existing DDL Triggers has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a DDL Trigger in the DB Explorer tree and select the **Duplicate DDL Trigger <DDL_trigger_name>...** context menu item.

© 2008 EMS Database Management Solutions

Duplicate Object Wizard allows you to select the database to create a new DDL Trigger in, and to edit the result SQL statement for creating the DDL Trigger.

Editing DDL Triggers

To edit an existing DDL Trigger:
- select the DDL Trigger for editing in the DB Explorer tree (type the first letters of the DDL Trigger name for quick search);
- right-click the object and select the **Edit DDL Trigger <DDL_trigger_name>** context menu item, or simply double-click the DDL Trigger;
- edit DDL Trigger properties using the appropriate tabs of DDL Trigger Editor.

To change the name of a DDL Trigger:
- select the DDL Trigger to rename in the DB Explorer tree;
- right-click the DDL Trigger alias and select the **Rename DDL Trigger <DDL_trigger_name>...** item from the context menu;
- edit the DDL Trigger name using the **Rename Object...** dialog.

Dropping DDL Triggers

To drop a DDL Trigger:
- select the DDL Trigger to drop in the DB Explorer tree;
- right-click the object and select the **Drop DDL Trigger <DDL_trigger_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new DDL Trigger;
Ctrl+O to edit the selected DDL Trigger;
Ctrl+R to rename the DDL Trigger;
Shift+Del to drop the object from the database.

13.2.13.2 Edit DDL Trigger

A **trigger** is a special kind of stored procedure that automatically executes when an event occurs in the database.

DDL triggers fire in response to a variety of data definition language (DDL) events which are primarily *CREATE*, *ALTER* and *DROP* statements; logon triggers fire in response to the *LOGON* event that is raised when a user sessions is being established.

Creating DDL Triggers

To create a new DDL Trigger:
- select the **Database | New Object...** main menu item;
- select **DDL Trigger** in the Create New Object dialog;
- define DDL Trigger properties using the appropriate tabs of DDL Trigger Editor.

Hint: To create a new DDL Trigger, you can also right-click the **DDL Triggers** node or any object within this node in the DB Explorer tree and select the **New DDL Trigger** item from the context menu.

© 2008 EMS Database Management Solutions

To create a new DDL Trigger with the same properties as one of existing DDL Triggers has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a DDL Trigger in the DB Explorer tree and select the **Duplicate DDL Trigger <DDL_trigger_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new DDL Trigger in, and to edit the result SQL statement for creating the DDL Trigger.

Editing DDL Triggers

To edit an existing DDL Trigger:
- select the DDL Trigger for editing in the DB Explorer tree (type the first letters of the DDL Trigger name for quick search);
- right-click the object and select the **Edit DDL Trigger <DDL_trigger_name>** context menu item, or simply double-click the DDL Trigger;
- edit DDL Trigger properties using the appropriate tabs of DDL Trigger Editor.

To change the name of a DDL Trigger:
- select the DDL Trigger to rename in the DB Explorer tree;
- right-click the DDL Trigger alias and select the **Rename DDL Trigger <DDL_trigger_name>...** item from the context menu;
- edit the DDL Trigger name using the **Rename Object...** dialog.

Dropping DDL Triggers

To drop a DDL Trigger:
- select the DDL Trigger to drop in the DB Explorer tree;
- right-click the object and select the **Drop DDL Trigger <DDL_trigger_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new DDL Trigger;
Ctrl+O to edit the selected DDL Trigger;
Ctrl+R to rename the DDL Trigger;
Shift+Del to drop the object from the database.

13.2.13.3 Drop DDL Trigger

A **trigger** is a special kind of stored procedure that automatically executes when an event occurs in the database.

DDL triggers fire in response to a variety of data definition language (DDL) events which are primarily *CREATE*, *ALTER* and *DROP* statements; logon triggers fire in response to the *LOGON* event that is raised when a user sessions is being established.

Creating DDL Triggers

© 2008 EMS Database Management Solutions

To create a new DDL Trigger:
- select the **Database | New Object...** main menu item;
- select **DDL Trigger** in the Create New Object dialog;
- define DDL Trigger properties using the appropriate tabs of DDL Trigger Editor.

Hint: To create a new DDL Trigger, you can also right-click the **DDL Triggers** node or any object within this node in the DB Explorer tree and select the **New DDL Trigger** item from the context menu.

To create a new DDL Trigger with the same properties as one of existing DDL Triggers has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a DDL Trigger in the DB Explorer tree and select the **Duplicate DDL Trigger <DDL_trigger_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new DDL Trigger in, and to edit the result SQL statement for creating the DDL Trigger.

Editing DDL Triggers

To edit an existing DDL Trigger:
- select the DDL Trigger for editing in the DB Explorer tree (type the first letters of the DDL Trigger name for quick search);
- right-click the object and select the **Edit DDL Trigger <DDL_trigger_name>** context menu item, or simply double-click the DDL Trigger;
- edit DDL Trigger properties using the appropriate tabs of DDL Trigger Editor.

To change the name of a DDL Trigger:
- select the DDL Trigger to rename in the DB Explorer tree;
- right-click the DDL Trigger alias and select the **Rename DDL Trigger <DDL_trigger_name>...** item from the context menu;
- edit the DDL Trigger name using the **Rename Object...** dialog.

Dropping DDL Triggers

To drop a DDL Trigger:
- select the DDL Trigger to drop in the DB Explorer tree;
- right-click the object and select the **Drop DDL Trigger <DDL_trigger_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new DDL Trigger;
Ctrl+O to edit the selected DDL Trigger;
Ctrl+R to rename the DDL Trigger;
Shift+Del to drop the object from the database.

© 2008 EMS Database Management Solutions

13.2.14 Work with Partition Functions

13.2.14.1 Create partition function

A **partition function** specifies how a table or index is partitioned. The function maps the domain into a set of partitions. When creating a partition function, you specify the number of partitions, the partitioning column, and the range of partition column values for each partition. Note that when you specify the partitioning column, you can only specify one.

Creating Partition Functions

To create a new partition function:
- select the **Database | New Object...** main menu item;
- select **Partition Function** in the Create New Object dialog;
- define partition function properties using the appropriate tabs of Partition Function Editor.

Hint: To create a new partition function, you can also right-click the **Partition Functions** node or any object within this node in the DB Explorer tree and select the **New Partition Function** item from the context menu.

To create a new partition function with the same properties as one of existing partition functions has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a partition function in the DB Explorer tree and select the **Duplicate Partition Function <partition_function_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new partition function in, and to edit the result SQL statement for creating the partition function.

Editing Partition Functions

To edit an existing partition function:
- select the partition function for editing in the DB Explorer tree (type the first letters of the partition function name for quick search);
- right-click the object and select the **Edit Partition Function <partition_function_name>** context menu item, or simply double-click the partition function;
- edit partition function properties using the appropriate tabs of Partition Function Editor.

To change the name of a partition function:
- select the partition function to rename in the DB Explorer tree;
- right-click the partition function alias and select the **Rename Partition Function <partition_function_name>...** item from the context menu;
- edit the partition function name using the **Rename Object...** dialog.

Dropping Partition Functions

To drop a partition function:
- select the partition function to drop in the DB Explorer tree;

© 2008 EMS Database Management Solutions

- right-click the object and select the **Drop Partition Function <partition_function_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new partition function;
Ctrl+O to edit the selected partition function;
Ctrl+R to rename the partition function;
Shift+Del to drop the object from the database.

13.2.14.2 Edit partition function

A **partition function** specifies how a table or index is partitioned. The function maps the domain into a set of partitions. When creating a partition function, you specify the number of partitions, the partitioning column, and the range of partition column values for each partition. Note that when you specify the partitioning column, you can only specify one.

Creating Partition Functions

To create a new partition function:
- select the **Database | New Object...** main menu item;
- select **Partition Function** in the Create New Object dialog;
- define partition function properties using the appropriate tabs of Partition Function Editor.

Hint: To create a new partition function, you can also right-click the **Partition Functions** node or any object within this node in the DB Explorer tree and select the **New Partition Function** item from the context menu.

To create a new partition function with the same properties as one of existing partition functions has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a partition function in the DB Explorer tree and select the **Duplicate Partition Function <partition_function_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new partition function in, and to edit the result SQL statement for creating the partition function.

Editing Partition Functions

To edit an existing partition function:
- select the partition function for editing in the DB Explorer tree (type the first letters of the partition function name for quick search);
- right-click the object and select the **Edit Partition Function <partition_function_name>** context menu item, or simply double-click the partition function;
- edit partition function properties using the appropriate tabs of Partition Function Editor.

To change the name of a partition function:

© 2008 EMS Database Management Solutions

- select the partition function to rename in the DB Explorer tree;
- right-click the partition function alias and select the **Rename Partition Function <partition_function_name>...** item from the context menu;
- edit the partition function name using the **Rename Object...** dialog.

Dropping Partition Functions

To drop a partition function:
- select the partition function to drop in the DB Explorer tree;
- right-click the object and select the **Drop Partition Function <partition_function_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new partition function;
Ctrl+O to edit the selected partition function;
Ctrl+R to rename the partition function;
Shift+Del to drop the object from the database.

13.2.14.3 Drop partition function

A **partition function** specifies how a table or index is partitioned. The function maps the domain into a set of partitions. When creating a partition function, you specify the number of partitions, the partitioning column, and the range of partition column values for each partition. Note that when you specify the partitioning column, you can only specify one.

Creating Partition Functions

To create a new partition function:
- select the **Database | New Object...** main menu item;
- select **Partition Function** in the Create New Object dialog;
- define partition function properties using the appropriate tabs of Partition Function Editor.

Hint: To create a new partition function, you can also right-click the **Partition Functions** node or any object within this node in the DB Explorer tree and select the **New Partition Function** item from the context menu.

To create a new partition function with the same properties as one of existing partition functions has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a partition function in the DB Explorer tree and select the **Duplicate Partition Function <partition_function_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new partition function in, and to edit the result SQL statement for creating the partition function.

Editing Partition Functions

© 2008 EMS Database Management Solutions

To edit an existing partition function:
- select the partition function for editing in the DB Explorer tree (type the first letters of the partition function name for quick search);
- right-click the object and select the **Edit Partition Function <partition_function_name>** context menu item, or simply double-click the partition function;
- edit partition function properties using the appropriate tabs of Partition Function Editor.

To change the name of a partition function:
- select the partition function to rename in the DB Explorer tree;
- right-click the partition function alias and select the **Rename Partition Function <partition_function_name>...** item from the context menu;
- edit the partition function name using the **Rename Object...** dialog.

Dropping Partition Functions

To drop a partition function:
- select the partition function to drop in the DB Explorer tree;
- right-click the object and select the **Drop Partition Function <partition_function_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new partition function;
Ctrl+O to edit the selected partition function;
Ctrl+R to rename the partition function;
Shift+Del to drop the object from the database.

13.2.15 Work with Partition Schemes

13.2.15.1 Create partition scheme

A **partition scheme** maps the partitions produced by a partition function to a set of filegroups that you define. A partition scheme can use only one partition function. However, a partition function can participate in more than one partition scheme.

Creating Partition Schemes

To create a new partition scheme:
- select the **Database | New Object...** main menu item;
- select **Partition Scheme** in the Create New Object dialog;
- define partition scheme properties using the appropriate tabs of Partition Scheme Editor.

Hint: To create a new partition scheme, you can also right-click the **Partition Schemes** node or any object within this node in the DB Explorer tree and select the **New Partition Scheme** item from the context menu.

To create a new partition scheme with the same properties as one of existing partition schemes has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

© 2008 EMS Database Management Solutions

Alternatively, you can right-click a partition scheme in the DB Explorer tree and select the **Duplicate Partition Scheme <partition_scheme_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new partition scheme in, and to edit the result SQL statement for creating the partition scheme.

Editing Partition Schemes

To edit an existing partition scheme:
- select the partition scheme for editing in the DB Explorer tree (type the first letters of the partition scheme name for quick search);
- right-click the object and select the **Edit Partition Scheme <partition_scheme_name>** context menu item, or simply double-click the partition scheme;
- edit partition scheme properties using the appropriate tabs of Partition Scheme Editor.

To change the name of a partition scheme:
- select the partition scheme to rename in the DB Explorer tree;
- right-click the partition scheme alias and select the **Rename Partition Scheme <partition_scheme_name>...** item from the context menu;
- edit the partition scheme name using the **Rename Object...** dialog.

Dropping Partition Schemes

To drop a partition scheme:
- select the partition scheme to drop in the DB Explorer tree;
- right-click the object and select the **Drop Partition Scheme <partition_scheme_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new partition scheme;
Ctrl+O to edit the selected partition scheme;
Ctrl+R to rename the partition scheme;
Shift+Del to drop the object from the database.

13.2.15.2 Edit partition scheme

A **partition scheme** maps the partitions produced by a partition function to a set of filegroups that you define. A partition scheme can use only one partition function. However, a partition function can participate in more than one partition scheme.

Creating Partition Schemes

To create a new partition scheme:
- select the **Database | New Object...** main menu item;
- select **Partition Scheme** in the Create New Object dialog;
- define partition scheme properties using the appropriate tabs of Partition Scheme Editor.

Hint: To create a new partition scheme, you can also right-click the **Partition Schemes**

© 2008 EMS Database Management Solutions

node or any object within this node in the DB Explorer tree and select the **New Partition Scheme** item from the context menu.

To create a new partition scheme with the same properties as one of existing partition schemes has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a partition scheme in the DB Explorer tree and select the **Duplicate Partition Scheme <partition_scheme_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new partition scheme in, and to edit the result SQL statement for creating the partition scheme.

Editing Partition Schemes

To edit an existing partition scheme:
- select the partition scheme for editing in the DB Explorer tree (type the first letters of the partition scheme name for quick search);
- right-click the object and select the **Edit Partition Scheme <partition_scheme_name>** context menu item, or simply double-click the partition scheme;
- edit partition scheme properties using the appropriate tabs of Partition Scheme Editor.

To change the name of a partition scheme:
- select the partition scheme to rename in the DB Explorer tree;
- right-click the partition scheme alias and select the **Rename Partition Scheme <partition_scheme_name>...** item from the context menu;
- edit the partition scheme name using the **Rename Object...** dialog.

Dropping Partition Schemes

To drop a partition scheme:
- select the partition scheme to drop in the DB Explorer tree;
- right-click the object and select the **Drop Partition Scheme <partition_scheme_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new partition scheme;
Ctrl+O to edit the selected partition scheme;
Ctrl+R to rename the partition scheme;
Shift+Del to drop the object from the database.

13.2.15.3 Drop partition scheme

A **partition scheme** maps the partitions produced by a partition function to a set of filegroups that you define. A partition scheme can use only one partition function. However, a partition function can participate in more than one partition scheme.

Creating Partition Schemes

© 2008 EMS Database Management Solutions

To create a new partition scheme:
- select the **Database | New Object...** main menu item;
- select **Partition Scheme** in the Create New Object dialog;
- define partition scheme properties using the appropriate tabs of Partition Scheme Editor.

Hint: To create a new partition scheme, you can also right-click the **Partition Schemes** node or any object within this node in the DB Explorer tree and select the **New Partition Scheme** item from the context menu.

To create a new partition scheme with the same properties as one of existing partition schemes has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a partition scheme in the DB Explorer tree and select the **Duplicate Partition Scheme <partition_scheme_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new partition scheme in, and to edit the result SQL statement for creating the partition scheme.

Editing Partition Schemes

To edit an existing partition scheme:
- select the partition scheme for editing in the DB Explorer tree (type the first letters of the partition scheme name for quick search);
- right-click the object and select the **Edit Partition Scheme <partition_scheme_name>** context menu item, or simply double-click the partition scheme;
- edit partition scheme properties using the appropriate tabs of Partition Scheme Editor.

To change the name of a partition scheme:
- select the partition scheme to rename in the DB Explorer tree;
- right-click the partition scheme alias and select the **Rename Partition Scheme <partition_scheme_name>...** item from the context menu;
- edit the partition scheme name using the **Rename Object...** dialog.

Dropping Partition Schemes

To drop a partition scheme:
- select the partition scheme to drop in the DB Explorer tree;
- right-click the object and select the **Drop Partition Scheme <partition_scheme_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new partition scheme;
Ctrl+O to edit the selected partition scheme;
Ctrl+R to rename the partition scheme;
Shift+Del to drop the object from the database.

© 2008 EMS Database Management Solutions

13.2.16 Work with Full-Text Catalogs

13.2.16.1 Create full-text catalog

Full-text catalogs and **full-text indexes** are the objects that are implemented in Microsoft® SQL Server for efficient full-text administration which is carried out at the server, database, table and column levels.

Like regular indexes, **full-text indexes** can be automatically updated as data are modified in the associated tables. Index updating is normally performed as an asynchronous process that runs in the background during periods of low database activity. Tables with the same update characteristics (e.g. small number of changes and large number of changes) should be grouped together and assigned to the same **full-text catalog**.

One **full-text catalog** can have several *full-text indexes*, but a full-text index can only be part of one full-text catalog.

Creating Full-Text Catalogs

To create a new full-text catalog:
- select the **Database | New Object...** main menu item;
- select **Full-Text Catalog** in the Create New Object dialog;
- define full-text catalog properties using the appropriate tabs of Full-Text Catalog Editor.

Hint: To create a new full-text catalog, you can also right-click the **Full-Text Catalogs** node or any object within this node in the DB Explorer tree and select the **New Full-Text Catalog** item from the context menu.

To create a new full-text catalog with the same properties as one of existing full-text catalogs has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a full-text catalog in the DB Explorer tree and select the **Duplicate Full-Text Catalog <full-text_catalog_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new full-text catalog in, and to edit the result SQL statement for creating the full-text catalog.

Editing Full-Text Catalogs

To edit an existing full-text catalog:
- select the full-text catalog for editing in the DB Explorer tree (type the first letters of the full-text catalog name for quick search);
- right-click the object and select the **Edit Full-Text Catalog <full-text_catalog_name>** context menu item, or simply double-click the full-text catalog;
- edit full-text catalog properties using the appropriate tabs of Full-Text Catalog Editor.

Dropping Full-Text Catalogs

To drop a full-text catalog:

© 2008 EMS Database Management Solutions

- select the full-text catalog to drop in the DB Explorer tree;
- right-click the object and select the **Drop Full-Text Catalog <full-text_catalog_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new full-text catalog;
Ctrl+O to edit the selected full-text catalog;
Shift+Del to drop the object from the database.

13.2.16.2 Edit full-text catalog

Full-text catalogs and **full-text indexes** are the objects that are implemented in Microsoft® SQL Server for efficient full-text administration which is carried out at the server, database, table and column levels.

Like regular indexes, **full-text indexes** can be automatically updated as data are modified in the associated tables. Index updating is normally performed as an asynchronous process that runs in the background during periods of low database activity. Tables with the same update characteristics (e.g. small number of changes and large number of changes) should be grouped together and assigned to the same **full-text catalog**.

One **full-text catalog** can have several *full-text indexes*, but a full-text index can only be part of one full-text catalog.

Creating Full-Text Catalogs

To create a new full-text catalog:
- select the **Database | New Object...** main menu item;
- select **Full-Text Catalog** in the Create New Object dialog;
- define full-text catalog properties using the appropriate tabs of Full-Text Catalog Editor.

Hint: To create a new full-text catalog, you can also right-click the **Full-Text Catalogs** node or any object within this node in the DB Explorer tree and select the **New Full-Text Catalog** item from the context menu.

To create a new full-text catalog with the same properties as one of existing full-text catalogs has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a full-text catalog in the DB Explorer tree and select the **Duplicate Full-Text Catalog <full-text_catalog_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new full-text catalog in, and to edit the result SQL statement for creating the full-text catalog.

Editing Full-Text Catalogs

To edit an existing full-text catalog:
- select the full-text catalog for editing in the DB Explorer tree (type the first letters of

© 2008 EMS Database Management Solutions

the full-text catalog name for quick search);
- right-click the object and select the **Edit Full-Text Catalog <full-text_catalog_name>** context menu item, or simply double-click the full-text catalog;
- edit full-text catalog properties using the appropriate tabs of Full-Text Catalog Editor.

Dropping Full-Text Catalogs

To drop a full-text catalog:
- select the full-text catalog to drop in the DB Explorer tree;
- right-click the object and select the **Drop Full-Text Catalog <full-text_catalog_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new full-text catalog;
Ctrl+O to edit the selected full-text catalog;
Shift+Del to drop the object from the database.

13.2.16.3 Drop full-text catalog

Full-text catalogs and **full-text indexes** are the objects that are implemented in Microsoft® SQL Server for efficient full-text administration which is carried out at the server, database, table and column levels.

Like regular indexes, **full-text indexes** can be automatically updated as data are modified in the associated tables. Index updating is normally performed as an asynchronous process that runs in the background during periods of low database activity. Tables with the same update characteristics (e.g. small number of changes and large number of changes) should be grouped together and assigned to the same **full-text catalog**.

One **full-text catalog** can have several *full-text indexes*, but a full-text index can only be part of one full-text catalog.

Creating Full-Text Catalogs

To create a new full-text catalog:
- select the **Database | New Object...** main menu item;
- select **Full-Text Catalog** in the Create New Object dialog;
- define full-text catalog properties using the appropriate tabs of Full-Text Catalog Editor.

Hint: To create a new full-text catalog, you can also right-click the **Full-Text Catalogs** node or any object within this node in the DB Explorer tree and select the **New Full-Text Catalog** item from the context menu.

To create a new full-text catalog with the same properties as one of existing full-text catalogs has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a full-text catalog in the DB Explorer tree and select the **Duplicate Full-Text Catalog <full-text_catalog_name>...** context menu item.

© 2008 EMS Database Management Solutions

Duplicate Object Wizard allows you to select the database to create a new full-text catalog in, and to edit the result SQL statement for creating the full-text catalog.

Editing Full-Text Catalogs

To edit an existing full-text catalog:
- select the full-text catalog for editing in the DB Explorer tree (type the first letters of the full-text catalog name for quick search);
- right-click the object and select the **Edit Full-Text Catalog <full-text_catalog_name>** context menu item, or simply double-click the full-text catalog;
- edit full-text catalog properties using the appropriate tabs of Full-Text Catalog Editor.

Dropping Full-Text Catalogs

To drop a full-text catalog:
- select the full-text catalog to drop in the DB Explorer tree;
- right-click the object and select the **Drop Full-Text Catalog <full-text_catalog_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new full-text catalog;
Ctrl+O to edit the selected full-text catalog;
Shift+Del to drop the object from the database.

13.2.17 Work with Symmetric Keys

13.2.17.1 Create symmetric key

In SQL Server 2005, data can be encrypted with a hierarchical encryption and key management infrastructure. Each layer encrypts the layer below it by using a combination of the following mechanisms for encryption: **symmetric keys**, asymmetric keys, certificates.

A **symmetric key** is a single key that is used for both encryption and decryption. Encryption and decryption by using a symmetric key is fast and suitable for routine use with sensitive data in the database.

Creating Symmetric Keys

To create a new symmetric key:
- select the **Database | New Object...** main menu item;
- select **Symmetric Key** in the Create New Object dialog;
- define symmetric key properties using the appropriate tabs of Symmetric Key Editor.

Hint: To create a new symmetric key, you can also right-click the **Symmetric Keys** node or any object within this node in the DB Explorer tree and select the **New Symmetric Key** item from the context menu.

© 2008 EMS Database Management Solutions

To create a new symmetric key with the same properties as one of existing symmetric keys has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a symmetric key in the DB Explorer tree and select the **Duplicate Symmetric Key <symmetric_key_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new symmetric key in, and to edit the result SQL statement for creating the symmetric key.

Editing Symmetric Keys

To edit an existing symmetric key:
- select the symmetric key for editing in the DB Explorer tree (type the first letters of the symmetric key name for quick search);
- right-click the object and select the **Edit Symmetric Key <symmetric_key_name>** context menu item, or simply double-click the symmetric key;
- edit symmetric key properties using the appropriate tabs of Symmetric Key Editor.

Dropping Symmetric Keys

To drop a symmetric key:
- select the symmetric key to drop in the DB Explorer tree;
- right-click the object and select the **Drop Symmetric Key <symmetric_key_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new symmetric key;
Ctrl+O to edit the selected symmetric key;
Shift+Del to drop the object from the database.

13.2.17.2 Edit symmetric key

In SQL Server 2005, data can be encrypted with a hierarchical encryption and key management infrastructure. Each layer encrypts the layer below it by using a combination of the following mechanisms for encryption: **symmetric keys**, asymmetric keys, certificates.

A **symmetric key** is a single key that is used for both encryption and decryption. Encryption and decryption by using a symmetric key is fast and suitable for routine use with sensitive data in the database.

Creating Symmetric Keys

To create a new symmetric key:
- select the **Database | New Object...** main menu item;
- select **Symmetric Key** in the Create New Object dialog;
- define symmetric key properties using the appropriate tabs of Symmetric Key Editor.

Hint: To create a new symmetric key, you can also right-click the **Symmetric Keys** node

© 2008 EMS Database Management Solutions

or any object within this node in the DB Explorer tree and select the **New Symmetric Key** item from the context menu.

To create a new symmetric key with the same properties as one of existing symmetric keys has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a symmetric key in the DB Explorer tree and select the **Duplicate Symmetric Key <symmetric_key_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new symmetric key in, and to edit the result SQL statement for creating the symmetric key.

Editing Symmetric Keys

To edit an existing symmetric key:
- select the symmetric key for editing in the DB Explorer tree (type the first letters of the symmetric key name for quick search);
- right-click the object and select the **Edit Symmetric Key <symmetric_key_name>** context menu item, or simply double-click the symmetric key;
- edit symmetric key properties using the appropriate tabs of Symmetric Key Editor.

Dropping Symmetric Keys

To drop a symmetric key:
- select the symmetric key to drop in the DB Explorer tree;
- right-click the object and select the **Drop Symmetric Key <symmetric_key_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new symmetric key;
Ctrl+O to edit the selected symmetric key;
Shift+Del to drop the object from the database.

13.2.17.3 Drop symmetric key

In SQL Server 2005, data can be encrypted with a hierarchical encryption and key management infrastructure. Each layer encrypts the layer below it by using a combination of the following mechanisms for encryption: **symmetric keys**, asymmetric keys, certificates.

A **symmetric key** is a single key that is used for both encryption and decryption. Encryption and decryption by using a symmetric key is fast and suitable for routine use with sensitive data in the database.

Creating Symmetric Keys

To create a new symmetric key:
- select the **Database | New Object...** main menu item;

© 2008 EMS Database Management Solutions

- select **Symmetric Key** in the Create New Object dialog;
- define symmetric key properties using the appropriate tabs of Symmetric Key Editor.

Hint: To create a new symmetric key, you can also right-click the **Symmetric Keys** node or any object within this node in the DB Explorer tree and select the **New Symmetric Key** item from the context menu.

To create a new symmetric key with the same properties as one of existing symmetric keys has:

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

Alternatively, you can right-click a symmetric key in the DB Explorer tree and select the **Duplicate Symmetric Key <symmetric_key_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new symmetric key in, and to edit the result SQL statement for creating the symmetric key.

Editing Symmetric Keys

To edit an existing symmetric key:

- select the symmetric key for editing in the DB Explorer tree (type the first letters of the symmetric key name for quick search);
- right-click the object and select the **Edit Symmetric Key <symmetric_key_name>** context menu item, or simply double-click the symmetric key;
- edit symmetric key properties using the appropriate tabs of Symmetric Key Editor.

Dropping Symmetric Keys

To drop a symmetric key:

- select the symmetric key to drop in the DB Explorer tree;
- right-click the object and select the **Drop Symmetric Key <symmetric_key_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new symmetric key;
Ctrl+O to edit the selected symmetric key;
Shift+Del to drop the object from the database.

13.2.18 Work with Asymmetric Keys

13.2.18.1 Create asymmetric key

In SQL Server 2005, data can be encrypted with a hierarchical encryption and key management infrastructure. Each layer encrypts the layer below it by using a combination of the following mechanisms for encryption: symmetric keys, **asymmetric keys**, certificates.

An **asymmetric key** is composed of a private key and the corresponding public key. Each key can decrypt data encrypted by the other. Asymmetric encryption and decryption provide a higher level of security than symmetric encryption. An asymmetric key can be used to encrypt a symmetric key for storage in a database.

© 2008 EMS Database Management Solutions

Creating Asymmetric Keys

To create a new asymmetric key:
- select the **Database | New Object...** main menu item;
- select **Asymmetric Key** in the Create New Object dialog;
- define asymmetric key properties using the appropriate tabs of Asymmetric Key Editor.

Hint: To create a new asymmetric key, you can also right-click the **Asymmetric Keys** node or any object within this node in the DB Explorer tree and select the **New Asymmetric Key** item from the context menu.

To create a new asymmetric key with the same properties as one of existing asymmetric keys has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a asymmetric key in the DB Explorer tree and select the **Duplicate Asymmetric Key <asymmetric_key_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new asymmetric key in, and to edit the result SQL statement for creating the asymmetric key.

Editing Asymmetric Keys

To edit an existing asymmetric key:
- select the asymmetric key for editing in the DB Explorer tree (type the first letters of the asymmetric key name for quick search);
- right-click the object and select the **Edit Asymmetric Key <asymmetric_key_name>** context menu item, or simply double-click the asymmetric key;
- edit asymmetric key properties using the appropriate tabs of Asymmetric Key Editor.

Dropping Asymmetric Keys

To drop a asymmetric key:
- select the asymmetric key to drop in the DB Explorer tree;
- right-click the object and select the **Drop Asymmetric Key <asymmetric_key_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new asymmetric key;
Ctrl+O to edit the selected asymmetric key;
Shift+Del to drop the object from the database.

13.2.18.2 Edit asymmetric key

In SQL Server 2005, data can be encrypted with a hierarchical encryption and key management infrastructure. Each layer encrypts the layer below it by using a combination of the following mechanisms for encryption: symmetric keys, **asymmetric keys**,

© 2008 EMS Database Management Solutions

certificates.

An **asymmetric key** is composed of a private key and the corresponding public key. Each key can decrypt data encrypted by the other. Asymmetric encryption and decryption provide a higher level of security than symmetric encryption. An asymmetric key can be used to encrypt a symmetric key for storage in a database.

Creating Asymmetric Keys

To create a new asymmetric key:
- select the **Database | New Object...** main menu item;
- select **Asymmetric Key** in the Create New Object dialog;
- define asymmetric key properties using the appropriate tabs of Asymmetric Key Editor.

Hint: To create a new asymmetric key, you can also right-click the **Asymmetric Keys** node or any object within this node in the DB Explorer tree and select the **New Asymmetric Key** item from the context menu.

To create a new asymmetric key with the same properties as one of existing asymmetric keys has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a asymmetric key in the DB Explorer tree and select the **Duplicate Asymmetric Key <asymmetric_key_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new asymmetric key in, and to edit the result SQL statement for creating the asymmetric key.

Editing Asymmetric Keys

To edit an existing asymmetric key:
- select the asymmetric key for editing in the DB Explorer tree (type the first letters of the asymmetric key name for quick search);
- right-click the object and select the **Edit Asymmetric Key <asymmetric_key_name>** context menu item, or simply double-click the asymmetric key;
- edit asymmetric key properties using the appropriate tabs of Asymmetric Key Editor.

Dropping Asymmetric Keys

To drop a asymmetric key:
- select the asymmetric key to drop in the DB Explorer tree;
- right-click the object and select the **Drop Asymmetric Key <asymmetric_key_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new asymmetric key;
Ctrl+O to edit the selected asymmetric key;
Shift+Del to drop the object from the database.

© 2008 EMS Database Management Solutions

13.2.18.3 Drop asymmetric key

In SQL Server 2005, data can be encrypted with a hierarchical encryption and key management infrastructure. Each layer encrypts the layer below it by using a combination of the following mechanisms for encryption: symmetric keys, **asymmetric keys**, certificates.

An **asymmetric key** is composed of a private key and the corresponding public key. Each key can decrypt data encrypted by the other. Asymmetric encryption and decryption provide a higher level of security than symmetric encryption. An asymmetric key can be used to encrypt a symmetric key for storage in a database.

Creating Asymmetric Keys

To create a new asymmetric key:
- select the **Database | New Object...** main menu item;
- select **Asymmetric Key** in the Create New Object dialog;
- define asymmetric key properties using the appropriate tabs of Asymmetric Key Editor.

Hint: To create a new asymmetric key, you can also right-click the **Asymmetric Keys** node or any object within this node in the DB Explorer tree and select the **New Asymmetric Key** item from the context menu.

To create a new asymmetric key with the same properties as one of existing asymmetric keys has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a asymmetric key in the DB Explorer tree and select the **Duplicate Asymmetric Key <asymmetric_key_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new asymmetric key in, and to edit the result SQL statement for creating the asymmetric key.

Editing Asymmetric Keys

To edit an existing asymmetric key:
- select the asymmetric key for editing in the DB Explorer tree (type the first letters of the asymmetric key name for quick search);
- right-click the object and select the **Edit Asymmetric Key <asymmetric_key_name>** context menu item, or simply double-click the asymmetric key;
- edit asymmetric key properties using the appropriate tabs of Asymmetric Key Editor.

Dropping Asymmetric Keys

To drop a asymmetric key:
- select the asymmetric key to drop in the DB Explorer tree;
- right-click the object and select the **Drop Asymmetric Key <asymmetric_key_name>...** context menu item;
- confirm dropping in the dialog window.

© 2008 EMS Database Management Solutions

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new asymmetric key;
Ctrl+O to edit the selected asymmetric key;
Shift+Del to drop the object from the database.

13.2.19 Work with Certificates

13.2.19.1 Create certificate

In SQL Server 2005, data can be encrypted with a hierarchical encryption and key management infrastructure. Each layer encrypts the layer below it by using a combination of the following mechanisms for encryption: symmetric keys, asymmetric keys, **certificates**.

A **certificate** is a digitally-signed statement that binds the value of a public key to the identity of the person, device or service that holds the corresponding private key. Certificates are database-level securable objects that follow the X.509 standard and support X.509 V1 fields. A certificate can be loaded from a file or assembly upon its creation.

Creating Certificates

To create a new certificate:
- select the **Database | New Object...** main menu item;
- select **Certificate** in the Create New Object dialog;
- define certificate properties using the appropriate tabs of Certificate Editor.

Hint: To create a new certificate, you can also right-click the **Certificates** node or any object within this node in the DB Explorer tree and select the **New Certificate** item from the context menu.

To create a new certificate with the same properties as one of existing certificates has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a certificate in the DB Explorer tree and select the **Duplicate Certificate <certificate_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new certificate in, and to edit the result SQL statement for creating the certificate.

Editing Certificates

To edit an existing certificate:
- select the certificate for editing in the DB Explorer tree (type the first letters of the certificate name for quick search);
- right-click the object and select the **Edit Certificate <certificate_name>** context menu item, or simply double-click the certificate;
- edit certificate properties using the appropriate tabs of Certificate Editor.

Dropping Certificates

© 2008 EMS Database Management Solutions

To drop a certificate:
- select the certificate to drop in the DB Explorer tree;
- right-click the object and select the **Drop Certificate <certificate_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new certificate;
Ctrl+O to edit the selected certificate;
Shift+Del to drop the object from the database.

13.2.19.2 Edit certificate

In SQL Server 2005, data can be encrypted with a hierarchical encryption and key management infrastructure. Each layer encrypts the layer below it by using a combination of the following mechanisms for encryption: symmetric keys, asymmetric keys, **certificates**.

A **certificate** is a digitally-signed statement that binds the value of a public key to the identity of the person, device or service that holds the corresponding private key. Certificates are database-level securable objects that follow the X.509 standard and support X.509 V1 fields. A certificate can be loaded from a file or assembly upon its creation.

Creating Certificates

To create a new certificate:
- select the **Database | New Object...** main menu item;
- select **Certificate** in the Create New Object dialog;
- define certificate properties using the appropriate tabs of Certificate Editor.

Hint: To create a new certificate, you can also right-click the **Certificates** node or any object within this node in the DB Explorer tree and select the **New Certificate** item from the context menu.

To create a new certificate with the same properties as one of existing certificates has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a certificate in the DB Explorer tree and select the **Duplicate Certificate <certificate_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new certificate in, and to edit the result SQL statement for creating the certificate.

Editing Certificates

To edit an existing certificate:
- select the certificate for editing in the DB Explorer tree (type the first letters of the certificate name for quick search);
- right-click the object and select the **Edit Certificate <certificate_name>** context

© 2008 EMS Database Management Solutions

menu item, or simply double-click the certificate;
- edit certificate properties using the appropriate tabs of Certificate Editor.

Dropping Certificates

To drop a certificate:
- select the certificate to drop in the DB Explorer tree;
- right-click the object and select the **Drop Certificate <certificate_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new certificate;
Ctrl+O to edit the selected certificate;
Shift+Del to drop the object from the database.

13.2.19.3 Drop certificate

In SQL Server 2005, data can be encrypted with a hierarchical encryption and key management infrastructure. Each layer encrypts the layer below it by using a combination of the following mechanisms for encryption: symmetric keys, asymmetric keys, **certificates**.

A **certificate** is a digitally-signed statement that binds the value of a public key to the identity of the person, device or service that holds the corresponding private key. Certificates are database-level securable objects that follow the X.509 standard and support X.509 V1 fields. A certificate can be loaded from a file or assembly upon its creation.

Creating Certificates

To create a new certificate:
- select the **Database | New Object...** main menu item;
- select **Certificate** in the Create New Object dialog;
- define certificate properties using the appropriate tabs of Certificate Editor.

Hint: To create a new certificate, you can also right-click the **Certificates** node or any object within this node in the DB Explorer tree and select the **New Certificate** item from the context menu.

To create a new certificate with the same properties as one of existing certificates has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a certificate in the DB Explorer tree and select the **Duplicate Certificate <certificate_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new certificate in, and to edit the result SQL statement for creating the certificate.

Editing Certificates

© 2008 EMS Database Management Solutions

To edit an existing certificate:
- select the certificate for editing in the DB Explorer tree (type the first letters of the certificate name for quick search);
- right-click the object and select the **Edit Certificate <certificate_name>** context menu item, or simply double-click the certificate;
- edit certificate properties using the appropriate tabs of Certificate Editor.

Dropping Certificates

To drop a certificate:
- select the certificate to drop in the DB Explorer tree;
- right-click the object and select the **Drop Certificate <certificate_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new certificate;
Ctrl+O to edit the selected certificate;
Shift+Del to drop the object from the database.

13.2.20 Work with Users

13.2.20.1 Create User

A database **user** is a database-level security principal enabling object access permission control at the finest level of granularity. A user represents a single SQL Server login within the scope of the database in which the user is defined.

Creating Users

To create a new user:
- select the **Database | New Object...** main menu item;
- select **User** in the Create New Object dialog;
- define user properties using the appropriate tabs of User Editor.

Hint: To create a new user, you can also right-click the **Users** node or any object within this node in the DB Explorer tree and select the **New User** item from the context menu.

To create a new user with the same properties as one of existing users has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a user in the DB Explorer tree and select the **Duplicate User <user_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new user in, and to edit the result SQL statement for creating the user.

Editing Users

To edit an existing user:
- select the user for editing in the DB Explorer tree (type the first letters of the user

© 2008 EMS Database Management Solutions

name for quick search);
- right-click the object and select the **Edit User <user_name>** context menu item, or simply double-click the user;
- edit user properties using the appropriate tabs of User Editor.

Dropping Users

To drop a user:
- select the user to drop in the DB Explorer tree;
- right-click the object and select the **Drop User <user_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new user;
Ctrl+O to edit the selected user;
Shift+Del to drop the object from the database.

13.2.20.2 Edit User

A database **user** is a database-level security principal enabling object access permission control at the finest level of granularity. A user represents a single SQL Server login within the scope of the database in which the user is defined.

Creating Users

To create a new user:
- select the **Database | New Object...** main menu item;
- select **User** in the Create New Object dialog;
- define user properties using the appropriate tabs of User Editor.

Hint: To create a new user, you can also right-click the **Users** node or any object within this node in the DB Explorer tree and select the **New User** item from the context menu.

To create a new user with the same properties as one of existing users has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a user in the DB Explorer tree and select the **Duplicate User <user_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new user in, and to edit the result SQL statement for creating the user.

Editing Users

To edit an existing user:
- select the user for editing in the DB Explorer tree (type the first letters of the user name for quick search);
- right-click the object and select the **Edit User <user_name>** context menu item, or simply double-click the user;
- edit user properties using the appropriate tabs of User Editor.

© 2008 EMS Database Management Solutions

Dropping Users

To drop a user:
- select the user to drop in the DB Explorer tree;
- right-click the object and select the **Drop User <user_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new user;
Ctrl+O to edit the selected user;
Shift+Del to drop the object from the database.

13.2.20.3 Drop User

A database **user** is a database-level security principal enabling object access permission control at the finest level of granularity. A user represents a single SQL Server login within the scope of the database in which the user is defined.

Creating Users

To create a new user:
- select the **Database | New Object...** main menu item;
- select **User** in the Create New Object dialog;
- define user properties using the appropriate tabs of User Editor.

Hint: To create a new user, you can also right-click the **Users** node or any object within this node in the DB Explorer tree and select the **New User** item from the context menu.

To create a new user with the same properties as one of existing users has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a user in the DB Explorer tree and select the **Duplicate User <user_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new user in, and to edit the result SQL statement for creating the user.

Editing Users

To edit an existing user:
- select the user for editing in the DB Explorer tree (type the first letters of the user name for quick search);
- right-click the object and select the **Edit User <user_name>** context menu item, or simply double-click the user;
- edit user properties using the appropriate tabs of User Editor.

Dropping Users

To drop a user:
- select the user to drop in the DB Explorer tree;
- right-click the object and select the **Drop User <user_name>...** context menu item;

© 2008 EMS Database Management Solutions

- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new user;
Ctrl+O to edit the selected user;
Shift+Del to drop the object from the database.

13.2.21 Work with Roles

13.2.21.1 Create Role

All users interact with SQL Server within the context of a database-level principal - **role**. A user can belong to multiple groups and have multiple roles, and the operations that are permitted by each role determine the actions that the user can perform. It is necessary to create SQL Server database roles when a group of users needs to perform a specified set of activities in the database.

Creating Roles

To create a new role:
- select the **Database | New Object...** main menu item;
- select **Role** in the Create New Object dialog;
- define role properties using the appropriate tabs of Role Editor.

Hint: To create a new role, you can also right-click the **Roles** node or any object within this node in the DB Explorer tree and select the **New Role** item from the context menu.

To create a new role with the same properties as one of existing roles has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a role in the DB Explorer tree and select the **Duplicate Role <role_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new role in, and to edit the result SQL statement for creating the role.

Editing Roles

To edit an existing role:
- select the role for editing in the DB Explorer tree (type the first letters of the role name for quick search);
- right-click the object and select the **Edit Role <role_name>** context menu item, or simply double-click the role;
- edit role properties using the appropriate tabs of Role Editor.

Dropping Roles

To drop a role:
- select the role to drop in the DB Explorer tree;
- right-click the object and select the **Drop Role <role_name>...** context menu item;
- confirm dropping in the dialog window.

© 2008 EMS Database Management Solutions

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new role;
Ctrl+O to edit the selected role;
Shift+Del to drop the object from the database.

13.2.21.2 Edit Role

All users interact with SQL Server within the context of a database-level principal - **role**. A user can belong to multiple groups and have multiple roles, and the operations that are permitted by each role determine the actions that the user can perform. It is necessary to create SQL Server database roles when a group of users needs to perform a specified set of activities in the database.

Creating Roles

To create a new role:
- select the **Database | New Object...** main menu item;
- select **Role** in the Create New Object dialog;
- define role properties using the appropriate tabs of Role Editor.

Hint: To create a new role, you can also right-click the **Roles** node or any object within this node in the DB Explorer tree and select the **New Role** item from the context menu.

To create a new role with the same properties as one of existing roles has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a role in the DB Explorer tree and select the **Duplicate Role <role_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new role in, and to edit the result SQL statement for creating the role.

Editing Roles

To edit an existing role:
- select the role for editing in the DB Explorer tree (type the first letters of the role name for quick search);
- right-click the object and select the **Edit Role <role_name>** context menu item, or simply double-click the role;
- edit role properties using the appropriate tabs of Role Editor.

Dropping Roles

To drop a role:
- select the role to drop in the DB Explorer tree;
- right-click the object and select the **Drop Role <role_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:

© 2008 EMS Database Management Solutions

Ctrl+N to create a new role;
Ctrl+O to edit the selected role;
Shift+Del to drop the object from the database.

13.2.21.3 Drop Role

All users interact with SQL Server within the context of a database-level principal - **role**. A user can belong to multiple groups and have multiple roles, and the operations that are permitted by each role determine the actions that the user can perform. It is necessary to create SQL Server database roles when a group of users needs to perform a specified set of activities in the database.

Creating Roles

To create a new role:
- select the **Database | New Object...** main menu item;
- select **Role** in the Create New Object dialog;
- define role properties using the appropriate tabs of Role Editor.

Hint: To create a new role, you can also right-click the **Roles** node or any object within this node in the DB Explorer tree and select the **New Role** item from the context menu.

To create a new role with the same properties as one of existing roles has:
- select the **Database | Duplicate Object...** main menu item;
- follow the instructions of Duplicate Object Wizard.

Alternatively, you can right-click a role in the DB Explorer tree and select the **Duplicate Role <role_name>...** context menu item.

Duplicate Object Wizard allows you to select the database to create a new role in, and to edit the result SQL statement for creating the role.

Editing Roles

To edit an existing role:
- select the role for editing in the DB Explorer tree (type the first letters of the role name for quick search);
- right-click the object and select the **Edit Role <role_name>** context menu item, or simply double-click the role;
- edit role properties using the appropriate tabs of Role Editor.

Dropping Roles

To drop a role:
- select the role to drop in the DB Explorer tree;
- right-click the object and select the **Drop Role <role_name>...** context menu item;
- confirm dropping in the dialog window.

Note: If more convenient, you can also use the following shortcuts:
Ctrl+N to create a new role;
Ctrl+O to edit the selected role;

© 2008 EMS Database Management Solutions

Shift+Del to drop the object from the database.

© 2008 EMS Database Management Solutions

13.3 Manage Server security

The following chapters inform you how to...

13.3.1 Add server login

Principals are individuals, groups, and processes that can request SQL Server resources.

SQL Manager 2008 for SQL Server provides management tools for the following server-level security principals:

- **Logins**
  Allows you to manage SQL Server logins.
- **Server Roles Manager**
  Allows you to manage SQL Server roles.

Adding logins

In order to add a new login:

- select the **Tools | Login Manager** main menu item or use the corresponding toolbar button to open Login Manager;
- select the **Add login...** item of the **context menu** or within the **Navigation bar**;
- define login properties and permissions using Login Editor

or

- right-click the **Logins** node (**Server Objects**) or any object within this node in DB Explorer and select the **New Login** context menu item, or use the *Ctrl+N* shortcut;
- define login properties and permissions using Login Editor.

Editing login properties

In order to edit an existing login:

- select the **Tools | Login Manager** main menu item or use the corresponding toolbar button to open Login Manager;
- select the **Edit login...** item of the **context menu** or within the **Navigation bar**;
- edit the login properties and permissions using Login Editor

or

- right-click the **Logins** node (**Server Objects**) or any object within this node in DB Explorer and select the **Edit Login <object_name>** context menu item, or use the *Ctrl+O* shortcut;
- edit the login properties and permissions using Login Editor.

Deleting logins

In order to delete an existing login:

- select the **Tools | Login Manager** main menu item or use the corresponding toolbar button to open Login Manager;
- right-click the login to delete and select the **Delete Login** item of the **context menu** or within the **Navigation bar**;
- confirm deleting in the dialog window

or

- right-click the login to delete within the **Logins** node (**Server Objects**) in DB Explorer

© 2008 EMS Database Management Solutions

and select the **Drop Login <object_name>** context menu item, or use the *Shift+Del* shortcut;
- confirm deleting in the dialog window

Allocating server roles

In order to allocate server roles to logins:
- select the **Tools | Server Roles** main menu item or use the corresponding toolbar button;
- select a server role, and add logins to the Server Role Members list using Server Roles Manager.

See also:
SQL Server objects and services

13.3.2 Edit login properties

Principals are individuals, groups, and processes that can request SQL Server resources.

SQL Manager 2008 for SQL Server provides management tools for the following server-level security principals:
- **Logins**
Allows you to manage SQL Server logins.
- **Server Roles Manager**
Allows you to manage SQL Server roles.

Adding logins

In order to add a new login:
- select the **Tools | Login Manager** main menu item or use the corresponding toolbar button to open Login Manager;
- select the **Add login...** item of the **context menu** or within the **Navigation bar**;
- define login properties and permissions using Login Editor

or
- right-click the **Logins** node (**Server Objects**) or any object within this node in DB Explorer and select the **New Login** context menu item, or use the *Ctrl+N* shortcut;
- define login properties and permissions using Login Editor.

Editing login properties

In order to edit an existing login:
- select the **Tools | Login Manager** main menu item or use the corresponding toolbar button to open Login Manager;
- select the **Edit login...** item of the **context menu** or within the **Navigation bar**;
- edit the login properties and permissions using Login Editor

© 2008 EMS Database Management Solutions

or

- right-click the **Logins** node (**Server Objects**) or any object within this node in DB Explorer and select the **Edit Login <object_name>** context menu item, or use the *Ctrl+O* shortcut;
- edit the login properties and permissions using Login Editor.

Deleting logins

In order to delete an existing login:

- select the **Tools | Login Manager** main menu item or use the corresponding toolbar button to open Login Manager;
- right-click the login to delete and select the **Delete Login** item of the **context menu** or within the **Navigation bar**;
- confirm deleting in the dialog window

or

- right-click the login to delete within the **Logins** node (**Server Objects**) in DB Explorer and select the **Drop Login <object_name>** context menu item, or use the *Shift+Del* shortcut;
- confirm deleting in the dialog window

Allocating server roles

In order to allocate server roles to logins:

- select the **Tools | Server Roles** main menu item or use the corresponding toolbar button;
- select a server role, and add logins to the Server Role Members list using Server Roles Manager.

See also:

SQL Server objects and services

13.3.3 Drop login

Principals are individuals, groups, and processes that can request SQL Server resources.

SQL Manager 2008 for SQL Server provides management tools for the following server-level security principals:

- **Logins**
  Allows you to manage SQL Server logins.
- **Server Roles Manager**
  Allows you to manage SQL Server roles.

Adding logins

In order to add a new login:

© 2008 EMS Database Management Solutions

- select the **Tools | Login Manager** main menu item or use the corresponding toolbar button to open Login Manager;
- select the **Add login...** item of the **context menu** or within the **Navigation bar**;
- define login properties and permissions using Login Editor

or

- right-click the **Logins** node (**Server Objects**) or any object within this node in DB Explorer and select the **New Login** context menu item, or use the *Ctrl+N* shortcut;
- define login properties and permissions using Login Editor.

Editing login properties

In order to edit an existing login:

- select the **Tools | Login Manager** main menu item or use the corresponding toolbar button to open Login Manager;
- select the **Edit login...** item of the **context menu** or within the **Navigation bar**;
- edit the login properties and permissions using Login Editor

or

- right-click the **Logins** node (**Server Objects**) or any object within this node in DB Explorer and select the **Edit Login <object_name>** context menu item, or use the *Ctrl+O* shortcut;
- edit the login properties and permissions using Login Editor.

Deleting logins

In order to delete an existing login:

- select the **Tools | Login Manager** main menu item or use the corresponding toolbar button to open Login Manager;
- right-click the login to delete and select the **Delete Login** item of the **context menu** or within the **Navigation bar**;
- confirm deleting in the dialog window

or

- right-click the login to delete within the **Logins** node (**Server Objects**) in DB Explorer and select the **Drop Login <object_name>** context menu item, or use the *Shift+Del* shortcut;
- confirm deleting in the dialog window

Allocating server roles

In order to allocate server roles to logins:

- select the **Tools | Server Roles** main menu item or use the corresponding toolbar button;
- select a server role, and add logins to the Server Role Members list using Server Roles Manager.

See also:

SQL Server objects and services

© 2008 EMS Database Management Solutions

13.3.4 Allocate server role

Principals are individuals, groups, and processes that can request SQL Server resources.

SQL Manager 2008 for SQL Server provides management tools for the following server-level security principals:

- **Logins**
  Allows you to manage SQL Server logins.
- **Server Roles Manager**
  Allows you to manage SQL Server roles.

Adding logins

In order to add a new login:

- select the **Tools | Login Manager** main menu item or use the corresponding toolbar button to open Login Manager;
- select the **Add login...** item of the **context menu** or within the **Navigation bar**;
- define login properties and permissions using Login Editor

or

- right-click the **Logins** node (**Server Objects**) or any object within this node in DB Explorer and select the **New Login** context menu item, or use the *Ctrl+N* shortcut;
- define login properties and permissions using Login Editor.

Editing login properties

In order to edit an existing login:

- select the **Tools | Login Manager** main menu item or use the corresponding toolbar button to open Login Manager;
- select the **Edit login...** item of the **context menu** or within the **Navigation bar**;
- edit the login properties and permissions using Login Editor

or

- right-click the **Logins** node (**Server Objects**) or any object within this node in DB Explorer and select the **Edit Login <object_name>** context menu item, or use the *Ctrl+O* shortcut;
- edit the login properties and permissions using Login Editor.

Deleting logins

In order to delete an existing login:

- select the **Tools | Login Manager** main menu item or use the corresponding toolbar button to open Login Manager;
- right-click the login to delete and select the **Delete Login** item of the **context menu** or within the **Navigation bar**;
- confirm deleting in the dialog window

or

- right-click the login to delete within the **Logins** node (**Server Objects**) in DB Explorer and select the **Drop Login <object_name>** context menu item, or use the *Shift+Del* shortcut;

© 2008 EMS Database Management Solutions

- confirm deleting in the dialog window

Allocating server roles

In order to allocate server roles to logins:

- select the **Tools | Server Roles** main menu item or use the corresponding toolbar button;
- select a server role, and add logins to the Server Role Members list using Server Roles Manager.

See also:

SQL Server objects and services

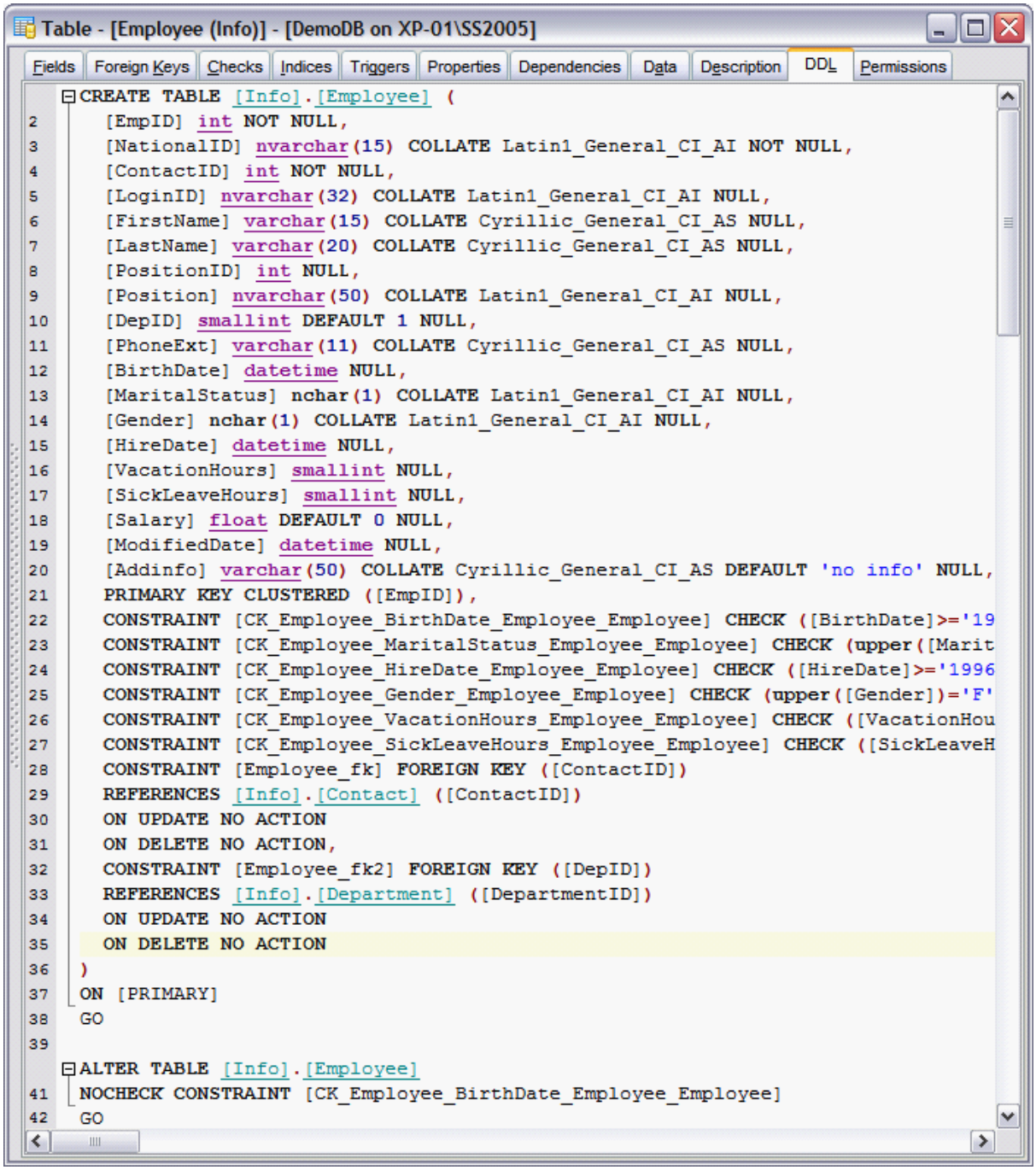
© 2008 EMS Database Management Solutions

Part

14 Appendix

14.1 Viewing object DDL structure

The **DDL** (Data Definition Language) tab displays the SQL statement for creating the object with all its subobjects, if any. 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 the SQL Editor or SQL Script Editor.



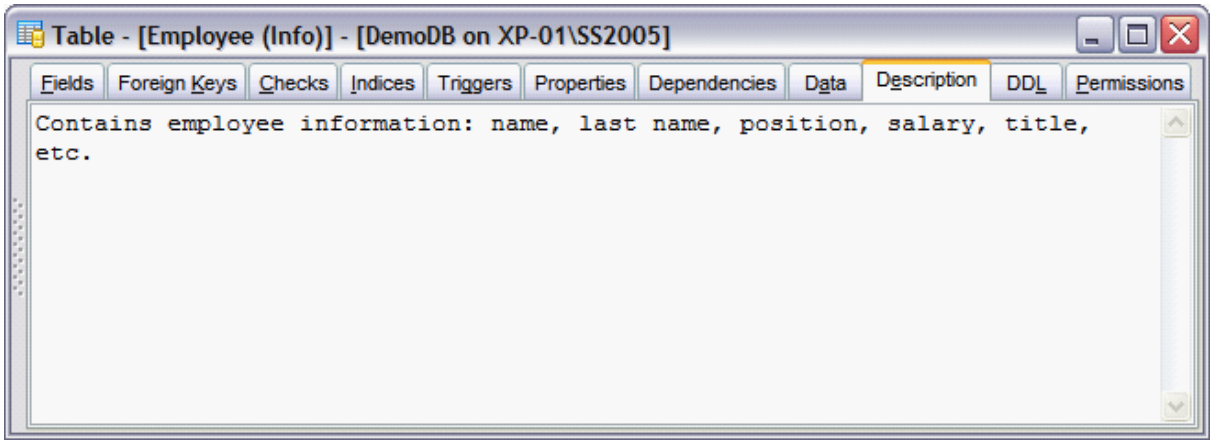
© 2008 EMS Database Management Solutions

Hint: If more convenient, you can use the **Save DDL to file** and **Open DDL in SQL Editor** items available on the DDL pane within the Navigation bar of object editors.

© 2008 EMS Database Management Solutions

14.2 Editing object description

The **Description** tab allows you to view and edit the comment for the object (optional).

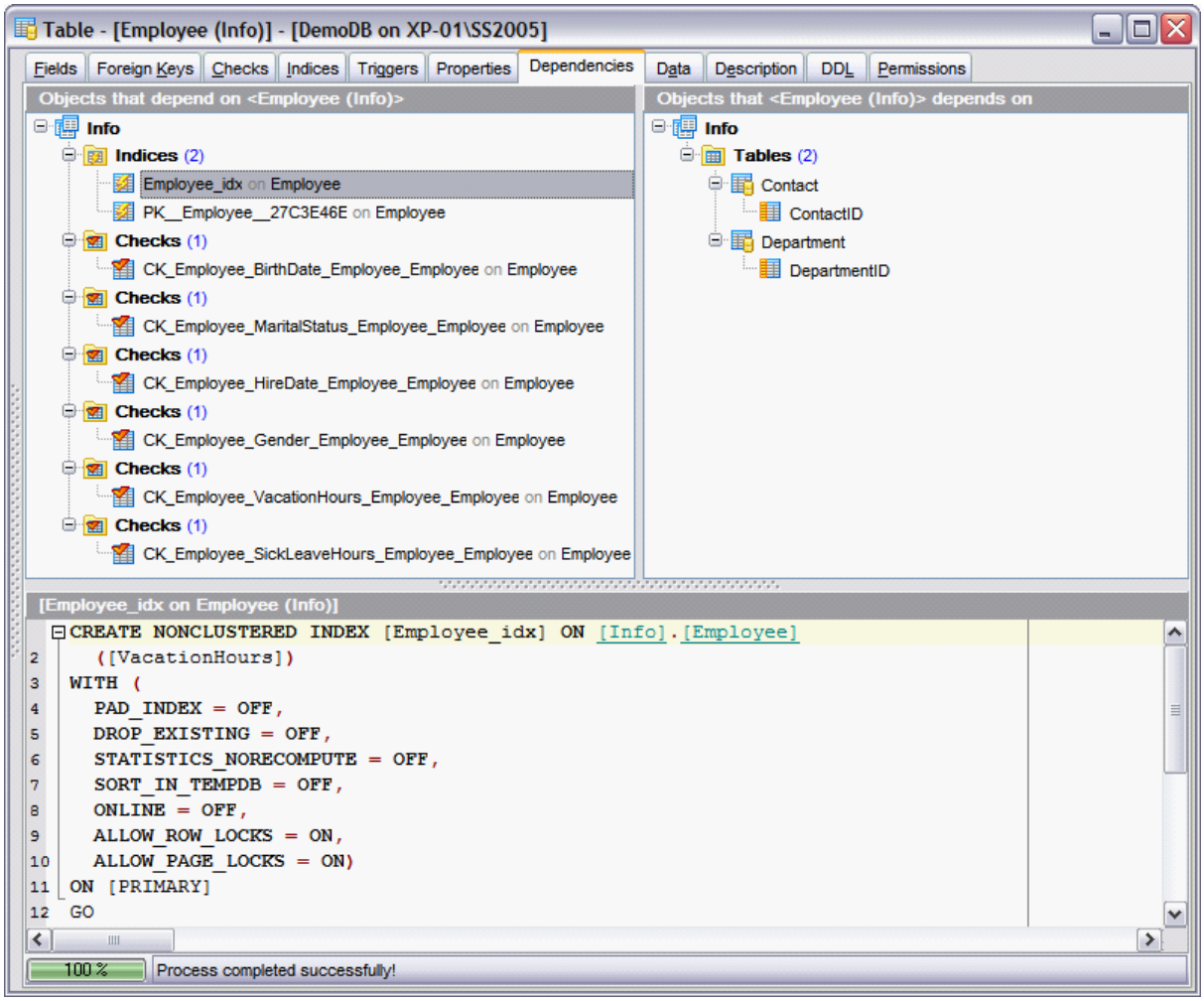


Note: Changing object description is performed by means of *sp_updateextendedproperty 'MS_Description'*.

© 2008 EMS Database Management Solutions

14.3 Browsing object dependencies

The **Dependencies** tab allows you to view objects that depend on the object being edited, and the objects that the edited object depends on.

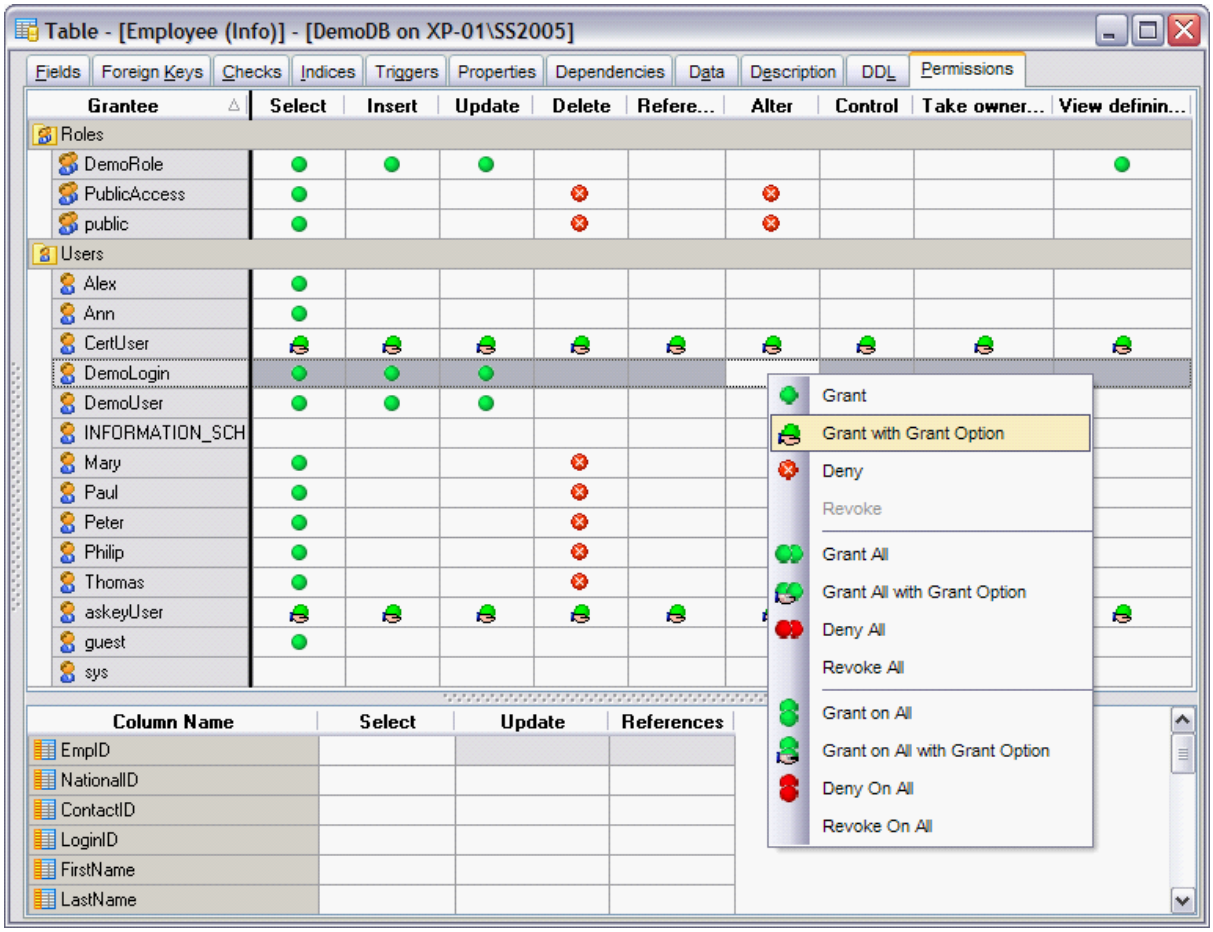


Hint: To open a dependent object or a depending object in its editor, you can simply double-click the object alias in the **Objects that <object_name> depends on** and **Objects that depend on <object_name>** lists.

© 2008 EMS Database Management Solutions

14.4 Setting object permissions

The **Permissions** tab allows you to view the permissions currently allocated for this object, and to grant permissions on the object to any of the existing principals.



For details see Grant Manager.

© 2008 EMS Database Management Solutions

14.5 Changing Metadata window

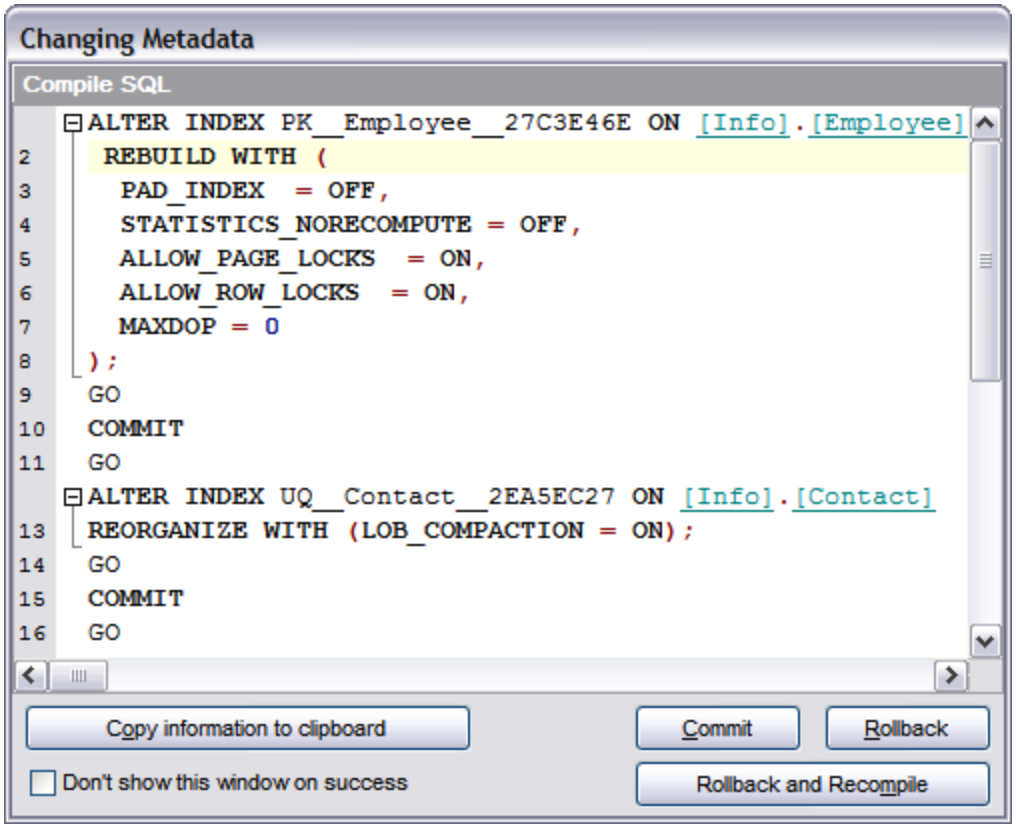
The **Changing Metadata** window is used to trace the errors and edit SQL statements during their compilation. The compilation window appears each time metadata is changed, both when the compilation is successful and when there are compilation errors. To hide this window for successful metadata changes, select the ☑ **Don't show this window on success** option.

Compile SQL
This area displays the SQL statement pending to be executed to perform metadata changing. In this area you can view and edit the SQL statement.
In case of a compilation error the **Error** tab also becomes visible - here you can view the error description returned by the server.

Commit
This button starts execution of the statement(s). Click it to commit the current transaction. This button is available only if there were no errors in compilation.

Rollback
This button cancels the script execution and allows you to return to the previous stage (editor window or DB Explorer).

Rollback and Recompile
This button calls for recompilation with the changes you made in the **Compile SQL** area. Use this button after correcting the SQL statement.



© 2008 EMS Database Management Solutions

If necessary, you can **copy information to clipboard** and save it in a text editor afterwards (the button is only enabled when a compilation error occurs).

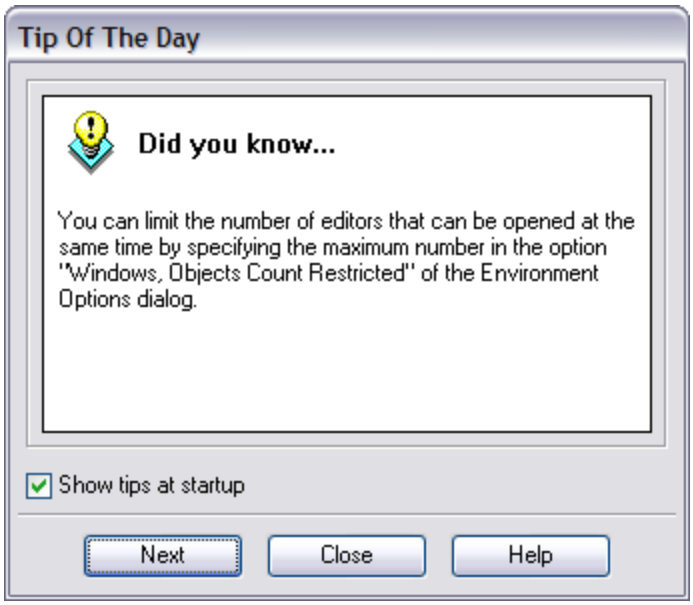
If you want this window to appear only in case of an error, uncheck the **Confirm metadata changing (Changing Metadata Window)** option (checked by default) available within the **Confirmations** section of the Environment Options dialog.

© 2008 EMS Database Management Solutions

14.6 Tip of the Day

This window allows you to see short messages notifying you about interesting particularities and useful features of SQL Manager 2008 for SQL Server.

To open this window, select the **Help | Tip of the Day** main menu item.



If you check the **Show tips at startup** option, this window will popup each time you start SQL Manager 2008 for SQL Server.

© 2008 EMS Database Management Solutions

14.7 Find Text dialog

The **Find Text** dialog is provided for quick and flexible searching for specified text within the working area of SQL Manager editors.

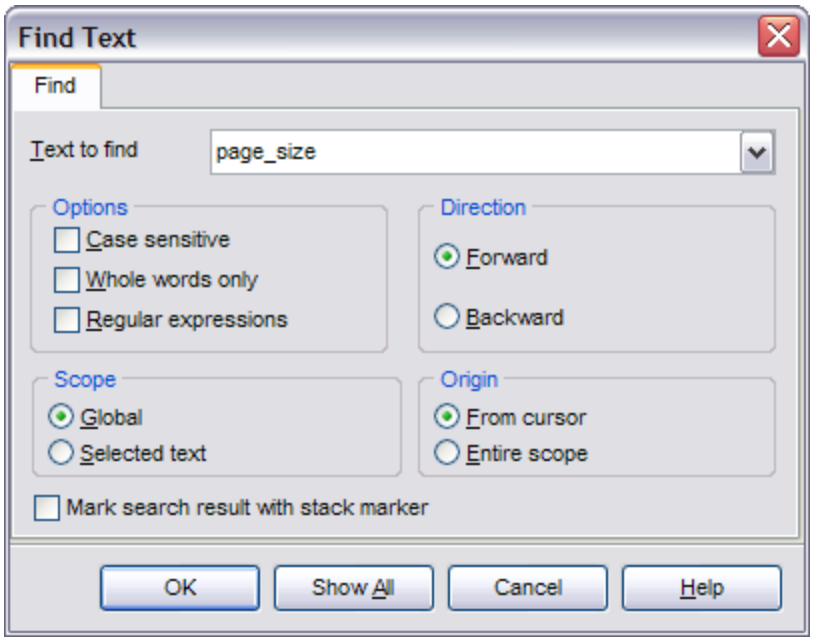
Text to find
Enter a search string in this box. The Arrow-Down button which can be found next to the input box allows you to select any of the previously entered search strings.

Options

Case sensitive
This option can be used to differentiate uppercase characters from lowercase ones during the search process.

Whole words only
Use this option to search 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.



Direction

Forward
Searches from the current position to the end of the working area.

Backward
Searches from the current position to the beginning of the working area.

Scope

© 2008 EMS Database Management Solutions

Global
Searches within the entire working area, in the direction specified by the *Direction* setting.

Selected text
Searches only within the currently selected text, in the direction specified by the *Direction* setting. You can use the mouse or block commands to select a block of text.

Origin

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.

Entire scope
The search covers either the entire block of selected text or the entire script (no matter where the cursor is in the Editor area) depending upon the *Scope* options.

Mark search result with stack marker
The option toggles marking search results. If this option is selected, stack markers are set at all search positions - this makes it possible to jump from one marker (search result) to another within the text.

Click the **Show All** button to highlight every occurrence of the search string.

© 2008 EMS Database Management Solutions

14.8 Replace Text dialog

The **Replace Text** dialog is provided for searching and replacing text within the working area of SQL Manager editors.

Text to find
Enter a search string in this box. The Arrow-Down button which can be found next to the input box allows you to select any of the previously entered search strings.

Text to replace
This box allows you to enter a string to replace the search string. The Arrow-Down button which can be found next to the input box allows you to select any of the previously entered strings. To replace the search string with an empty string, leave this input box blank.

Options

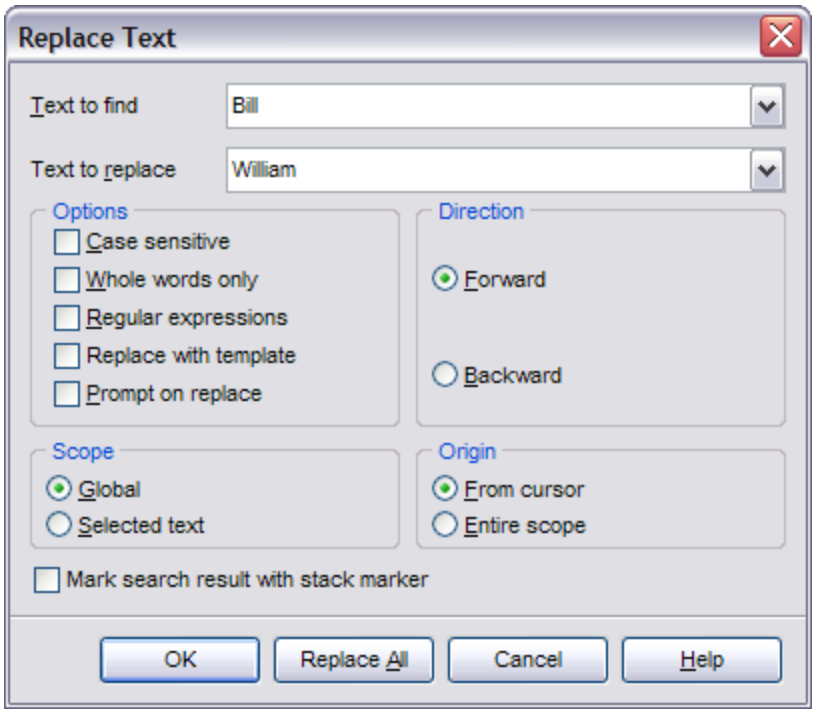
☑ **Case sensitive**
This option can be used to differentiate uppercase characters from lowercase ones during the search process.

☑ **Whole words only**
Use this option to search 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.

☑ **Prompt on replace**
Check this option if you wish to be prompted before replacing upon each occurrence of the search string. When this option is off, the search string is replaced automatically.

© 2008 EMS Database Management Solutions



Direction

Forward
Searches and replaces from the current position to the end of the working area.

Backward
Searches and replaces from the current position to the beginning of the working area.

Scope

Global
Searches and replaces within the entire working area, in the direction specified by the *Direction* setting.

Selected text
Searches and replaces only within the currently selected text, in the direction specified by the *Direction* setting. You can use the mouse or block commands to select a block of text.

Origin

From cursor
The search and replace process 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.

Entire scope
The search and replace process covers either the entire block of selected text or the

© 2008 EMS Database Management Solutions

entire script (no matter where the cursor is in the Editor area) depending upon the *Scope* options.

Mark search result with stack marker
The option toggles marking search results. If this option is selected, stack markers are set at all search positions - this makes it possible to jump from one marker (search result) to another within the text.

Click the **Replace All** button to replace every occurrence of the search string. If you have checked the **Prompt on replace** option, the confirmation dialog box appears upon each occurrence of the search string.

© 2008 EMS Database Management Solutions

14.9 Format specifiers

The following format specifiers are supported in the format string:

Float/Integer format

0
Digit place holder. If the value being formatted has a digit in the position where the '0' appears in the format string, then that digit is copied to the output string. Otherwise, a '0' is stored in that position in the output string.

#
Digit placeholder. If the value being formatted has a digit in the position where the '#' appears in the format string, then that digit is copied to the output string. Otherwise, nothing is stored in that position in the output string.

.
Decimal point. The first '.' character in the format string determines the location of the decimal separator in the formatted value; any additional '.' characters are ignored.

,
Thousand separator. If the format string contains one or more ',' characters, the output will have thousand separators inserted between each group of three digits to the left of the decimal point. The placement and number of ',' characters in the format string does not affect the output, except to indicate that thousand separators are wanted.

E+
Scientific notation. If any of the strings 'E+', 'E-', 'e+', or 'e-' are contained in the format string, the number is formatted using scientific notation. A group of up to four '0' characters can immediately follow the 'E+', 'E-', 'e+', or 'e-' to determine the minimum number of digits in the exponent. The 'E+' and 'e+' formats cause a plus sign to be output for positive exponents and a minus sign to be output for negative exponents. The 'E-' and 'e-' formats output a sign character only for negative exponents.

Date/Time format

c
Displays the date using the format using the Short Date Format, followed by the time using the Long Time Format. The time is not displayed if the date-time value indicates midnight precisely.

d
Displays the day as a number without a leading zero (1-31).

dd
Displays the day as a number with a leading zero (01-31).

ddd
Displays the day as an abbreviation (Sun-Sat) using the strings of the Short Day Names.

dddd

© 2008 EMS Database Management Solutions

Displays the day as a full name (Sunday-Saturday) using the strings of the Long Day Names.

ddddd
Displays the date using the Short Date Format.

dddddd
Displays the date using the Long Date Format.

e
Displays the year in the current period/era as a number without a leading zero (Japanese, Korean and Taiwanese locales only).

ee
Displays the year in the current period/era as a number with a leading zero (Japanese, Korean and Taiwanese locales only).

g
Displays the period/era as an abbreviation (Japanese and Taiwanese locales only).

gg
Displays the period/era as a full name. (Japanese and Taiwanese locales only).

m
Displays the month as a number without a leading zero (1-12). If the m specifier immediately follows an h or hh specifier, the minute rather than the month is displayed.

mm
Displays the month as a number with a leading zero (01-12). If the mm specifier immediately follows an h or hh specifier, the minute rather than the month is displayed.

mmm
Displays the month as an abbreviation (Jan-Dec) using the strings given of the Short Month Names.

mmmm
Displays the month as a full name (January-December) using the strings of the Long Month Names.

yy
Displays the year as a two-digit number (00-99).

yyyy
Displays the year as a four-digit number (0000-9999).

h
Displays the hour without a leading zero (0-23).

hh
Displays the hour with a leading zero (00-23).

n
Displays the minute without a leading zero (0-59).

© 2008 EMS Database Management Solutions

nn
Displays the minute with a leading zero (00-59).

s
Displays the second without a leading zero (0-59).

ss
Displays the second with a leading zero (00-59).

z
Displays the millisecond without a leading zero (0-999).

zzz
Displays the millisecond with a leading zero (000-999).

t
Displays the time using the Short Time Format.

tt
Displays the time using the Long Time Format.

am/pm
Uses the 12-hour clock for the preceding h or hh specifier, and displays 'am' for any hour before noon, and 'pm' for any hour after noon. The am/pm specifier can use lower, upper, or mixed case, and the result is displayed accordingly.

a/p
Uses the 12-hour clock for the preceding h or hh specifier, and displays 'a' for any hour before noon, and 'p' for any hour after noon. The a/p specifier can use lower, upper, or mixed case, and the result is displayed accordingly.

ampm
Uses the 12-hour clock for the preceding h or hh specifier, and displays the contents of the TimeAMString global variable for any hour before noon, and the contents of the TimePMString global variable for any hour after noon.

/
Displays the date separator character using the Date Separator.

:
Displays the time separator character using the Time Separator.

'xx'/"xx"
Characters enclosed in single or double quotes are displayed as-is, and do not affect formatting.

© 2008 EMS Database Management Solutions

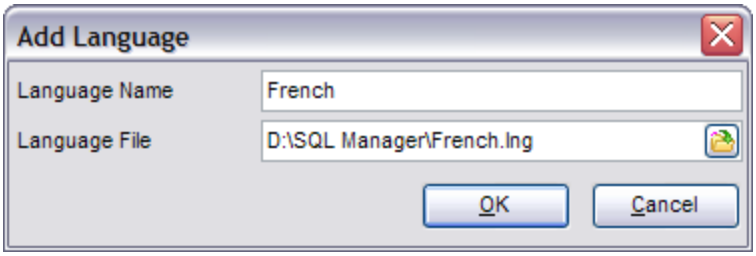
14.10 Language Info Editor

The **Language Info Editor** dialog allows you to set the language name and specify the corresponding **.lng* localization file. This dialog is opened when you add or edit a language (see Environment Options | Localization).

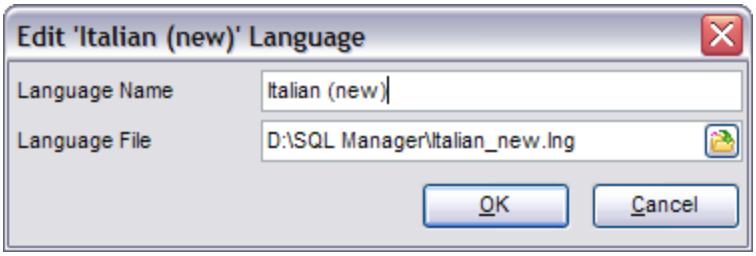
Language Name
The name of the language that is displayed in the Select Program Language dialog and within the **Available Languages** list of the Environment Options | Localization section.

Language File
The **.lng* file containing the translated string resources. See the *%program_directory%\Languages* folder to find already existing localization files.

Adding a language
The *Add language* dialog allows you to specify your own localization file and set the language name.



Editing a language
The *Edit language* dialog allows you to change the language name or select another localization file for the specified language.



© 2008 EMS Database Management Solutions

14.11 Using templates

For your convenience the ability to use templates is provided by SQL Manager 2008 for SQL Server. A template is a named collection of wizard options stored in a file.

Instead of performing a long chain of routine steps all the time you can save all the options of the wizard for future use as a template file. Select the **Templates | Save Template** drop-down menu item, specify the template file name and set an optional comment for the template file.
When starting the wizard next time, you can load the template by selecting the **Templates | Load Template** drop-down menu item.

Note that saving/loading of templates is possible at any step of the wizard.

© 2008 EMS Database Management Solutions

14.12 Supported file formats

MS Excel
The most popular e-table format used by Microsoft® Excel (*.xls). The result files are fully compatible with Microsoft® Excel versions 97-2000, 2003 and XP.

MS Access
File of Microsoft® Access format (*.mdb) with an ADO connection used.

MS Word
One of the most popular text processing formats used by Microsoft® Word (*.doc). The result files are fully compatible with Microsoft® Word versions 97-2000, 2003 and XP.

RTF
Rich Text Format (*.rtf) supported by many text processing programs (e.g. WordPad).

HTML
Hyper Text Markup Language file format (*.html, *.htm), complete compatibility with HTML 4.0 specification.

PDF
A standard format in electronic publishing (*.pdf).

Text file
Plain text file format (*.txt).

CSV file
Comma-Separated Value file format (*.csv).

DIF file
Data Interchange File (*.dif) format.

SYLK
Symbolic Links (*.slk) file format.

Note: all the text formats including *Text file*, *CSV*, *DIF*, *SYLK* are usually used as working or interchange formats.

LaTeX
A specific file format (*.tex) which is a popular (especially among mathematicians and physicists) macroextension of *TeX* pack developed by D.Knut.

XML
A markup language for documents containing structured information (*.xml).

DBF
Database file format (*.dbf)used by dBASE and a number of xBASE applications.

MS Excel 2007
The contemporary e-table format used by Microsoft® Excel 2007 (*.xlsx). The result files are fully compatible with Microsoft® Excel 2007.

© 2008 EMS Database Management Solutions

MS Word 2007
The contemporary text processing format used by Microsoft® Word 2007 (*.*docx*). The result files are fully compatible with Microsoft® Word 2007.

ODF Spreadsheets
OASIS Open Document Format for Office Applications - open document file format for spreadsheets (*.*ods*) used by a number of applications including OpenOffice.org and KOffice.

ODF text
OASIS Open Document Format for Office Applications - open document file format for word processing (*.*odt*) documents used by a number of applications including OpenOffice.org and KOffice.

© 2008 EMS Database Management Solutions

14.13 To-Do List

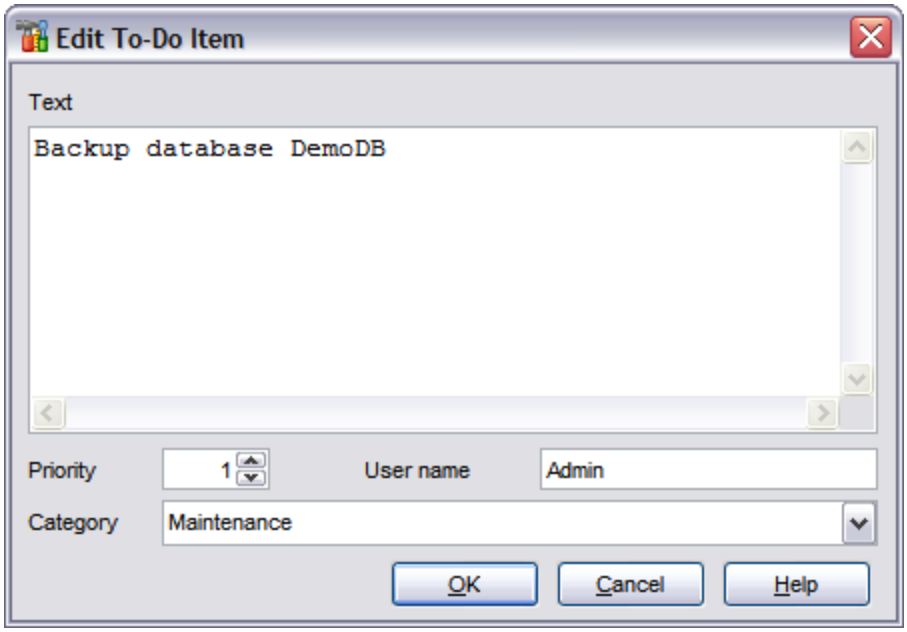
The **To-Do List** window allows you to make up a list of tasks for the database.

To call this window, select the **View | To-Do List** main menu item, or use the *Shift+Ctrl+T* shortcut.

The task list is displayed in a form of a grid. Its columns (*Action*, *Priority*, *User*, *Category*) correspond to the task parameters. Clicking the column caption to sort the task list by the current parameter or change the sorting direction.

Database
Select the database to apply the task list to. When switching between the databases you can view different task lists.

To add a task to this list, click the **Add Item** link of the Navigation bar, or select **Add Item** in the context menu. You can also use the *Ins* key for the same purpose. Define the task parameters and click **OK** to add the new task to the list.



Text
Optional text to describe the task.

Priority
Set a numeric value to indicate the priority of the task.

User Name
The database User name this task is applied to.

Category
Set a category for the task. Using categories may be useful for grouping tasks.

© 2008 EMS Database Management Solutions

To modify a task, select the task in the list and click the **Edit Item** link of the Navigation bar, or select **Edit Item** in the context menu. You can also use the *Enter* key for the same purpose.

To remove a task, select the task in the list and click the **Delete Item** link of the Navigation bar, or select **Delete Item** in the context menu. You can also use the *Del* key for the same purpose.

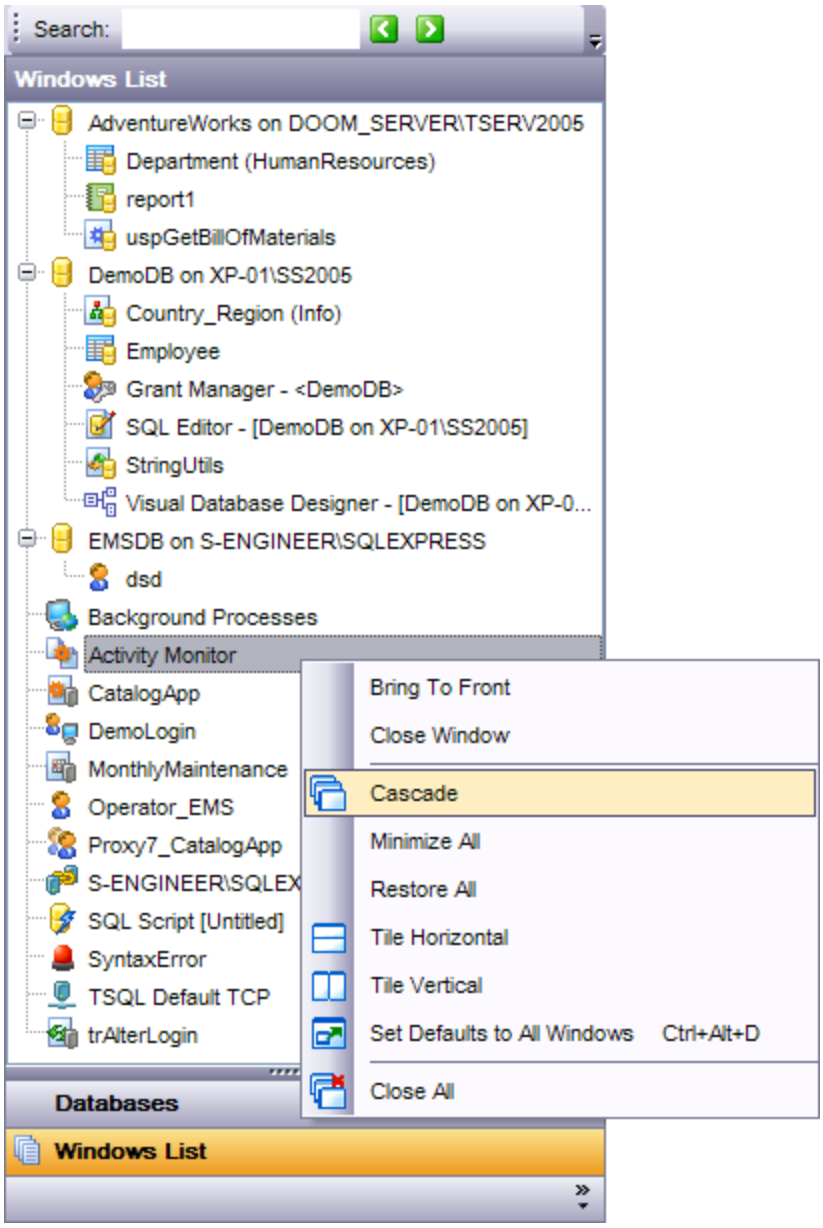
To remove all tasks from To-Do List, click the **Delete all** link of the Navigation bar, or select **Delete all** in the context menu. You can also use the *Ctrl+Del* shortcut for the same purpose.

© 2008 EMS Database Management Solutions

14.14 Window List

The **Window List** panel allows you to browse the list of windows that are currently opened within SQL Manager 2008 for SQL Server IDE.

To activate this panel as a DB Explorer tab, select the **Windows | Window List** main menu item, or use the *Ctrl+Alt+0* shortcut.



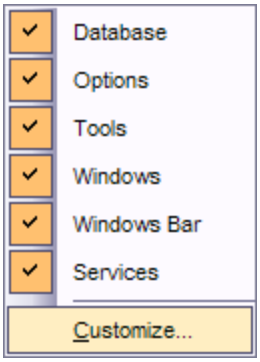
If necessary, you can right-click within the list area to call the **popup menu** which allows you to bring a window to foreground, close windows one by one or in groups, and to arrange the windows according to your preferences.

© 2008 EMS Database Management Solutions

14.15 Customize toolbars and menus

The **Customize** dialog allows you to create and personalize SQL Manager menus and toolbars.

To call this dialog, click the down-arrow button on the right side of any toolbar, then click **Add or Remove Buttons** and select **Customize...** from the drop-down menu. Alternatively, you can right-click any toolbar and select the **Customize...** popup menu item.



Toolbars

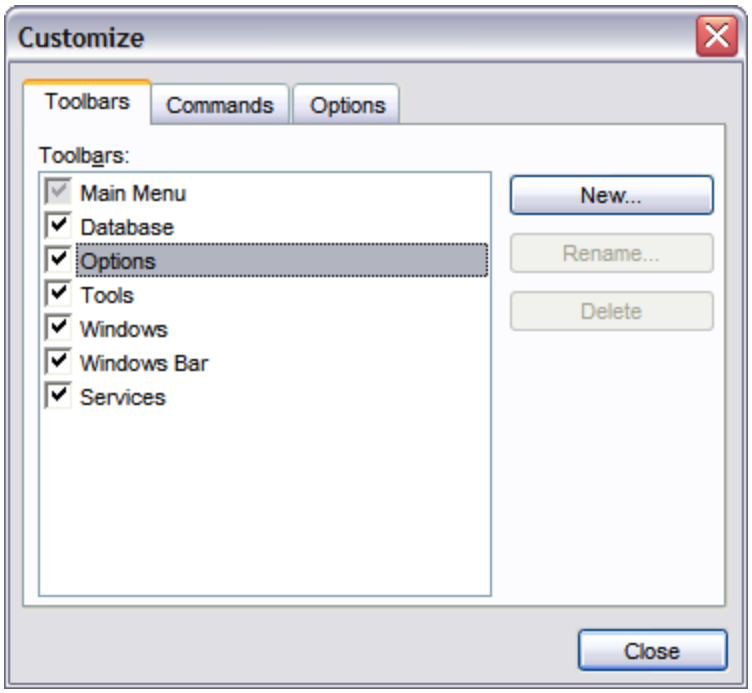
Toolbars
This list displays all currently existing toolbars of SQL Manager (both *default* and *user-defined* toolbars). Check/uncheck the box at a toolbar name to show/hide the toolbar.

New...
Use this button to add a new user-defined toolbar to the **Toolbars** list. Set a name for the newly created toolbar and dock it by dragging it to any permitted location within the application window.

Rename...
Use this button to rename the selected user-defined toolbar.

Delete
Use this button to delete the selected user-defined toolbar.

© 2008 EMS Database Management Solutions

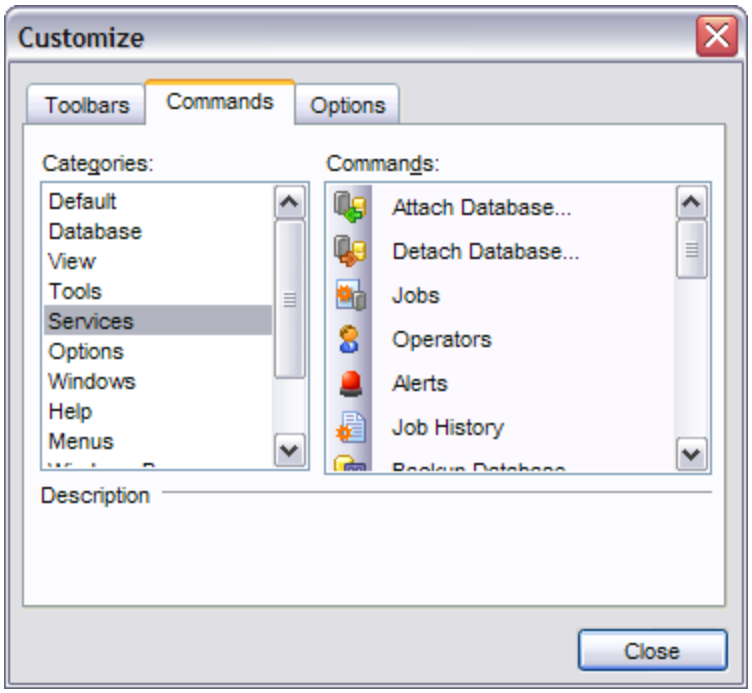


Commands

This tab allows you to browse the list of all commands available within the menus and toolbars of the application window. Selecting categories in the **Categories** list displays commands of the selected category (e.g. 'Database' or 'Tools') in the **Commands** list.

If necessary, you can pick a command and drag it to any toolbar to create a button for this command.

© 2008 EMS Database Management Solutions



Options

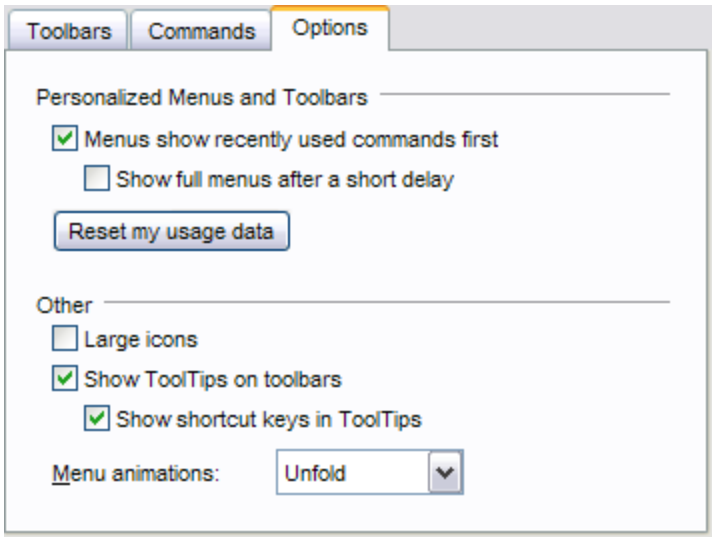
Personalized Menus and Toolbars

Menus show recently used commands first
This option determines whether the most frequently used items will be placed in menus at first position.
If this option is enabled, frequently used menu items are "promoted" and displayed higher on the list. Unused and infrequently used menu items are visually suppressed and appear "collapsed".

Show full menus after a short delay
This option is available only if the **Menus show recently used commands first** option is selected.
If this option is enabled, infrequently used menu items (if they appear "collapsed") will be automatically expanded after a delay upon setting mouse cursor (or upon selection with the *Up*/*Down* keys) on the bottom of the menu. Otherwise, the menu expands only after clicking its bottom-most button (or using the *Ctrl*+*Down* shortcut).

Reset my usage data
Resets the lists of recently used commands in the toolbars and menus.

© 2008 EMS Database Management Solutions



Other

Large icons
This option displays larger icons on the parent window toolbars.

Show ToolTips on toolbars
If this option is selected, ToolTips (hints) popup when the mouse cursor is positioned over a toolbar button.

Show shortcut keys in ToolTips
If this option is selected, the corresponding shortcuts are displayed in ToolTips (hints) for toolbar buttons.

Menu animations
Use the drop-down list to specify the menu animation effects:
None (no animation)
Random (random choice: *Unfold*, *Slide*, *Fade*)
Unfold (unfolding menus)
Slide (sliding menus)
Fade (menus fade in when appearing)

© 2008 EMS Database Management Solutions

14.16 SQL Manager shortcuts

Database management:

Shift+Ctrl+R	Register a SQL Server host using Register Host Wizard
Shift+Alt+R	Register a database using Register Database Wizard
Shift+Alt+U	Unregister the selected database
Shift+Ctrl+C	Connect to a database
Shift+Ctrl+D	Disconnect from a database

Database objects management:

Ctrl+N	Create a new object (the object type depends on the current selection)
Ctrl+O	Edit the selected object in its editor
Ctrl+R	Rename the selected object
Shift+Del	Drop the selected object
Ctrl+Shift+C	Collapse the current DB Explorer tree branch and switch selection to the parent tree node

SQL Manager tools:

F11	View/hide Database Explorer
Ctrl+F	Search for an item in the DB Explorer tree
Shift+Ctrl+T	Open the To-Do List window
F12	Show SQL Editor
Shift+F12	Open a new instance of SQL Editor
Shift+Ctrl+M	Open SQL Monitor
Shift+Ctrl+S	Open SQL Script Editor
Shift+Ctrl+L	Open Localization Editor
Ins	Add a new table subobject (the subobject type depends on the current tab selection)

SQL Editor and SQL Script (fixed and default):

F9	Execute query/script
Alt+F9	Execute selected only
Ctrl+Alt+F9	Execute under cursor
Ctrl+Alt+F2	Reset execution point (SQL Editor only)
Shift+Ctrl+<digit>	Toggle bookmark #*<digit>*
Ctrl+Z	Undo
Shift+Ctrl+Z	Redo
Ctrl+F	Search for text using the Find Text dialog
Ctrl+R	Replace text using the Replace Text dialog

© 2008 EMS Database Management Solutions

F3	Search next
Ctrl+I	Start incremental search
Alt+G	Go to line number (an input number dialog prompts for the number)
Ctrl+L	Load a script from an external file
Ctrl+S	Load the script to an external file
Shift+Ctrl+F	Format the SQL text using SQL Formatter
Alt+<symbol>	Switch to the query with *<&symbol>* in its name (SQL Editor only)

Working with windows and tabs:

Ctrl+Tab	Switch to the next tab
Ctrl+Alt-0	Open Window List
Ctrl+Alt+D	Set defaults to all windows
Ctrl+F6	Switch to the previous window
F6	Switch to the next window
Ctrl+W	Close the active window

© 2008 EMS Database Management Solutions

Credits

Software Developers:

Alexander Zhiltsov

Alexey Butalov

Dmitry Goldobin

Dmitry Schastlivtsev

Nicolay Sezganov

Vladimir Prischepa

Technical Writers:

Semyon Slobodenyuk

Dmitry Doni

Olga Ryabova

Cover Designer:

Tatyana Makurova

Translators:

Anna Shulkina

Serge Fominikh

Team Coordinators:

Alexander Zhiltsov

Alexander Chelyadin

Roman Tkachenko

© 2008 EMS Database Management Solutions