Data Import for SQL Server User's Manual

© 1999-2014 EMS Database Management Solutions, Ltd.

Data Import for SQL Server
User's Manual

© 1999-2014 EMS Database Management Solutions, Ltd.

Document generated on: 05/06/2014

Table of Contents

Part

I

1   Welcome to EMS Data Import utility!

EMS Data Import for SQL Server is a powerful program to import your data quickly from MS Excel/Excel 2007, MS Word 2007, MS Access, DBF, TXT, CSV, Open Document, HTML files to SQL Server tables. It provides adjustable import parameters, including source data formats for all the fields and destination data formats for the selected fields, commit options, number of records to skip, etc. Data Import for SQL Server includes a wizard which allows you to set all the import options for different files visually, and a command-line utility to execute import in one-touch.

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

Key features:
- Import from most popular formats: MS Excel 97-2007, MS Access, XML, DBF, TXT, CSV, MS Word 2007, Open Document Format and HTML
- Importing data to one or several tables or views from different databases
- Automatically creates table structure
- Special batch insert commands allow to import data at the maximum possible speed
- A number of import modes - Insert All, Insert New, Insert or Update, and others
- UNICODE support
- Adjustable import parameters for each source file
- Saving all import parameters set on current wizard session
- Command-line utility to import data using the configuration file
- Powerful visual options module
- User-friendly localisable wizard interface

Product information:

Homepage:	http://www.sqlmanager.net/en/products/mssql/dataimport
Support Ticket System:	http://www.sqlmanager.net/support
Register online at:	http://www.sqlmanager.net/en/products/mssql/dataimport/buy

1.1 What's new

Version	Release date
Data Import for SQL Server 3.7	June 09, 2014

What's new in EMS Data Import?

- Import rate is increased due to use of multithreading.
- [XML](#) 29. Added the possibility of manual and automated identification of the type of the imported XML-file.
- Now [log](#) 64 contains information about the string number where the error occurred.
- When importing from several files their [logs](#) 64 are located at different tabs, while the main log contains general information about these processes.
- [DBF](#) 37. Wrong identification of Integer type fields. Fixed now.
- "Keep identity" option did not work. Fixed now.
- "*yyyy-mm-ddThh:nn:ss*" format was recognized incorrectly. Fixed now.
- [HTML](#) 45. When importing from HTML file with the links the application froze. Fixed now.
- Other improvements and bugfixes.

See also:

Version history

1.2   System requirements

System requirements for Data Import for SQL Server

- 300-megahertz (MHz) 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, Windows® 2008 Server, Microsoft® Windows Vista, Microsoft® Windows 7, Microsoft® Windows 8, Microsoft® Windows 8.1
- 64MB RAM or more; 128MB or more recommended
- 20MB 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 SQL Server™
- Supported MS SQL Server versions: from 7.0 up to 2012

1.3   Installation

If you are **installing Data Import for SQL Server for the first time** on your PC:
- download the Data Import 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 *MsImportSetup.exe* from the local directory and follow the instructions of the installation wizard;
- after the installation process is completed, find the Data Import shortcut in the corresponding group of Windows Start menu.

[Installation FAQs](#)

If you want to **upgrade an installed copy of Data Import for SQL Server** to the latest version:
- download the **Data Import 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*;
- close Data Import application if it is running;
- run *MsImportSetup.exe* from the local directory and follow the instructions of the installation wizard.

See also:

[System requirements](#)
8

1.4   How to buy Data Import

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 online 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

To obtain **MORE INFORMATION** on this product, visit us at http://sqlmanager.net/en/products/mssql/dataimport

Product distribution	
EMS Data Import for SQL Server (Business license) + 1-Year Maintenance*	
EMS Data Import for SQL Server (Business license) + 2-Year Maintenance*	
EMS Data Import for SQL Server (Business license) + 3-Year Maintenance*	
EMS Data Import for SQL Server (Non-commercial license) + 1-Year Maintenance*	Buy Now!
EMS Data Import for SQL Server (Non-commercial license) + 2-Year Maintenance*	
EMS Data Import for SQL Server (Non-commercial license) + 3-Year Maintenance*	
EMS Data Import for SQL Server (Trial version)	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
- Access to personalized sqlmanager.net account
- online, speed-through maintenance renewal
- Advanced and exclusive notification of software promotions

· "Maintenance Owner ONLY" product promotions

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 online, speed-through Maintenance Reinstatement/Renewal Interface. After reinitiating/renewal you will receive a confirmation e-mail with all the necessary information.
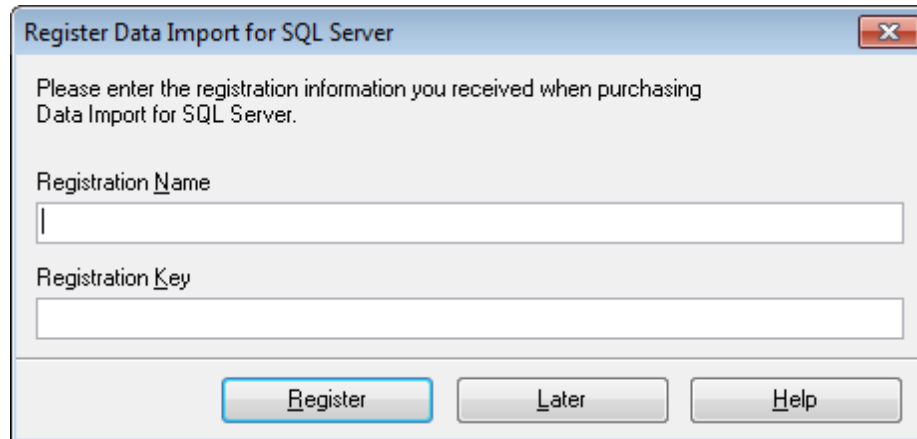

See also:

How to register Data Import 12

1.5   How to register Data Import

To **register** your newly purchased copy of EMS Data Import for SQL Server, perform the following:

- 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 at the startup page 24.

See also:
How to buy Data Import 10

1.6   EMS Data Import FAQ

Please read this page attentively if you have questions about Data Import for SQL Server.

Table of contents

Question/answer list

Q: What is EMS Data Import for SQL Server?
A: EMS Data Import for SQL Server is a powerful tool to import your data quickly from MS Access, MS Excel, DBF, TXT, CSV, XML, MS Excel 2007, MS Word 2007, HTML, ODF files to SQL Server tables. It provides adjustable import parameters, including source data formats for all the fields and destination data formats for the selected fields, commit options, number of records to skip, etc. Data Import includes a wizard which allows you to set all the import options for different files visually, and a command-line utility to execute import in one-touch.

Q: What do I need to start working with EMS Data Import for SQL Server?
A: First of all, you must have a possibility to connect to some local or remote SQL Server to work with Data Import. You can download SQL Server from http://www.microsoft. com/downloads/. Besides, you need your workstation to satisfy the system requirements 8 for Data Import for SQL Server.

Q: What is the difference between the Import feature of SQL Manager for SQL Server and the Data Import for SQL Server utility?
A: First of all, Data Import for SQL Server works faster as it is a considerably lighter application. Besides, it provides additional features for query building, namely:
- import data to several tables simultaneously;
- import data to tables selected from different databases;
- the command-line version of the utility to import data using the configuration (template) file with all import settings;
- automatically creates table structure.

Q: Are there any limitations implied in the trial version as compared with the full one?
A: Actually the trial version of the utility only allows you to import 20% of records into each table. With this limitation, you can test all the features implemented in Data Import for SQL Server within the 30-day trial period.

 Note: There is a 100 records (instead of 20%) limitation when importing from CSV, TXT or MS Access files.

Q: What is the easiest way to configure the template files for the Data Import console application?
A: You can configure the template files for each table visually using the Data Import Wizard application. Set the required options and select the Tools | Save Template menu item. All the options will be saved to the template file which can be used by the console application.

Q: I work behind a firewall (router). Can EMS Data Import connect to a remote SQL Server server?
A: Yes, but to do this, you should open the port where SQL Server works (1433 by default) on your firewall and on the firewall of the remote network.

Q: How to connect to a named instance of SQL Server in EMS Data Import for SQL Server?
A: To connect to a named instance of SQL Server 2005 just enter instance name in after host name and back slash symbol in 'Host name' field at the Setting Connection Properties step. (For example: my_server\my_instance.)
  To connect to a named instance of SQL Server 2000 you need:
  - Run SQL Server Client Network Utility (WINDOWS\\SYSTEM32\\cliconfig.exe) on client computer where Data Import for SQL Server runs.
  - Go to the second (Alias) tab of the Utility.
  - Add a new server alias with Server Name in form <server_name>\\<instanse_name> and TCP/IP as Network library.
  - Use the server alias name as host name in Data Import.

Q: When connecting to SQL Server I get an error "SQL Server does not exist or access denied". What is wrong?
A: Probably, your SQL Server is installed as a named instance (See Q. "How to connect to a named instance of SQL Server in EMS SQL Manager for SQL Server? 14 ").
  Besides, the following factors can be the reasons for connection problems:
  - You are trying to connect to SQL Server beyond your local network, through firewall (See Q. "I work behind a firewall (router). Can Data Import connect to a remote SQL Server? 14 "
  - Wrong login/password. Either you use SQL Server Authentication, but this authentication type is forbidden on the server, or you use Windows Authentication to connect to the server, but the login that corresponds to your domain name is absent on SQL Server.
  - MDAC works incorrectly. Try to reinstall Microsoft Data Access Components.
  - SQL Server works on a non-standard (other than 1433) port (See Q. "I can access

my server via an alternative port. I have searched for a setting in the program to alter the connection port, but have not yet been able to locate this. How would I go about altering this port? 15".
- There are no enabled client connection protocols. Or all enabled client connection protocols are disabled on server side. Use SQL Server Client Network Utility (WINDOWS\\SYSTEM32\\cliconfig.exe) for connection to SQL Server 2000 or lower or SQL Server Configuration Manager to enable appropriate protocols.

Q: I can access my server via an alternative port. I have searched for a setting in EMS Data Import 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 on alternative port do the following:
  - Run SQL Server Configuration Manager on client computer where Data Import runs.
  - Expand the SQL Native Client Configuration section.
  - Select the Aliases section.
  - Add a new server alias and set server name, port number and TCP/IP as Network library.
  - Use the server alias name as host name in Data Import utility.
  To connect to SQL Server 2000 on alternative port do the following:
  - Run SQL Server Client Network Utility (WINDOWS\\SYSTEM32\\cliconfg.exe) on client computer where Data Import runs utility.
  - Go to the second (Alias) tab of the Utility.
  - Add a new server alias and set server name, port number and TCP/IP as Network library.
  - Use the server alias name as host name in Data Import utility.

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

1.7   Other EMS Products

Quick navigation

MySQL [16]   Microsoft SQL [17]   PostgreSQL [18]   InterBase / FireBird [18]   Oracle [19]   IBM DB2 [20]   Tools & components [21]

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 20 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 [16]

Microsoft SQL

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!

EMS SQL Backup for SQL Server
Perform backup and restore, log shipping and many other regular maintenance tasks on the whole set of SQL Servers in your company.

SQL Administrator for SQL Server
Perform administrative tasks in the fastest, easiest and most efficient way. Manage maintenance tasks, monitor their performance schedule, frequency and the last execution result.

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 20 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 16

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 20 most popular data formats, including MS Access, MS Excel, MS Word, PDF, HTML and more

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 16

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 20 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.

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 16

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 20 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.

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

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 20 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 ¹⁶

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 Export .NET
Advanced Data Export .NET is a component suite for Microsoft Visual Studio .NET 2003, 2005, 2008 and 2010 that will allow you to save your data in the most popular data formats for the future viewing, modification, printing or web publication. You can export data into MS Access, MS Excel, MS Word (RTF), PDF, TXT, DBF, CSV and more! There will be no need to waste your time on tiresome data conversion - Advanced Data Export will do the task quickly and will give the result in the desired format.

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 ¹⁶

Part II

2 Wizard application

Data Import for SQL Server wizard application provides easy-to-use wizard interface to set all data import parameters visually.

Working with wizard application [24]
Using configuration files [66]
Setting program preferences [68]
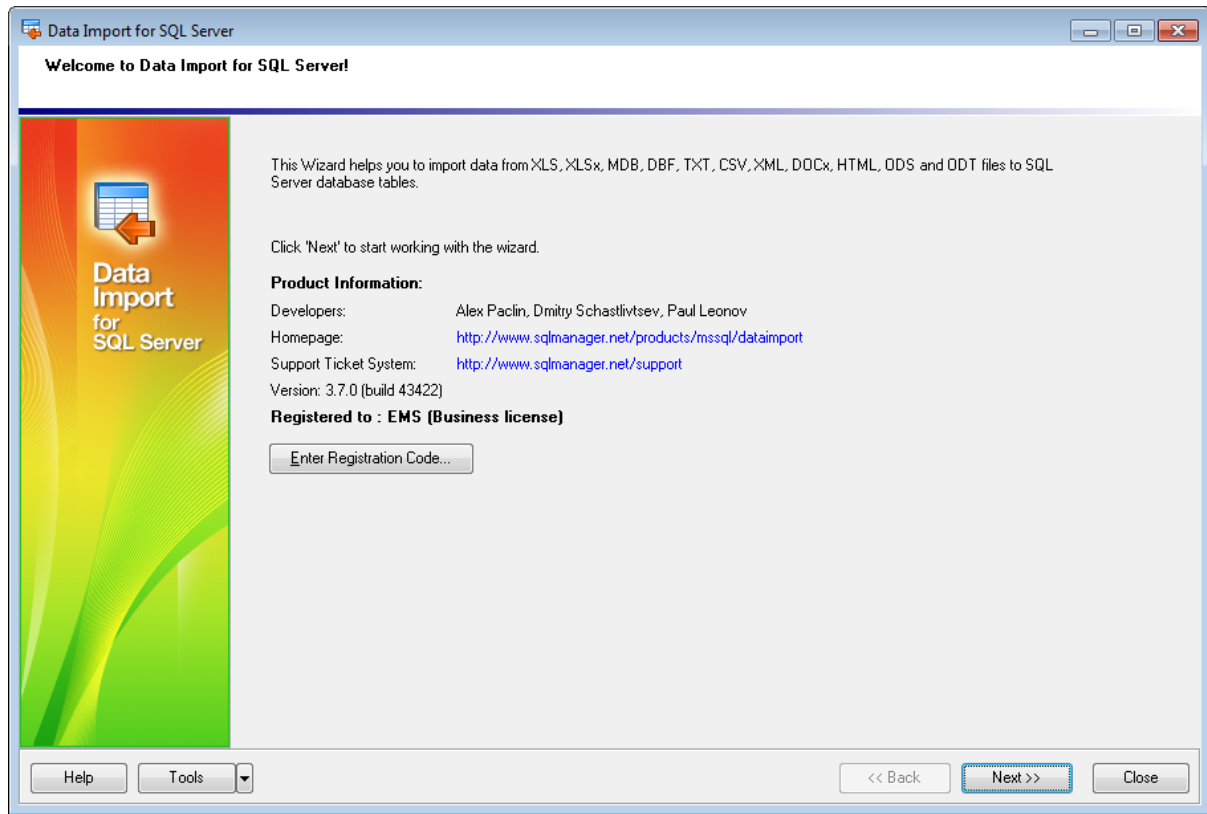
See also:
Console Application [76]

2.1   Working with wizard application

Follow the steps of the wizard to import data to SQL Server tables for your needs.

See also:

2.1.1   Getting started

This is how Data Import for SQL Server **wizard application** looks when you first start it.

This page allows you to view registration information. If you have not registered Data Import for SQL Server yet, you can do it by pressing the **Register** button and entering your registration information 12.
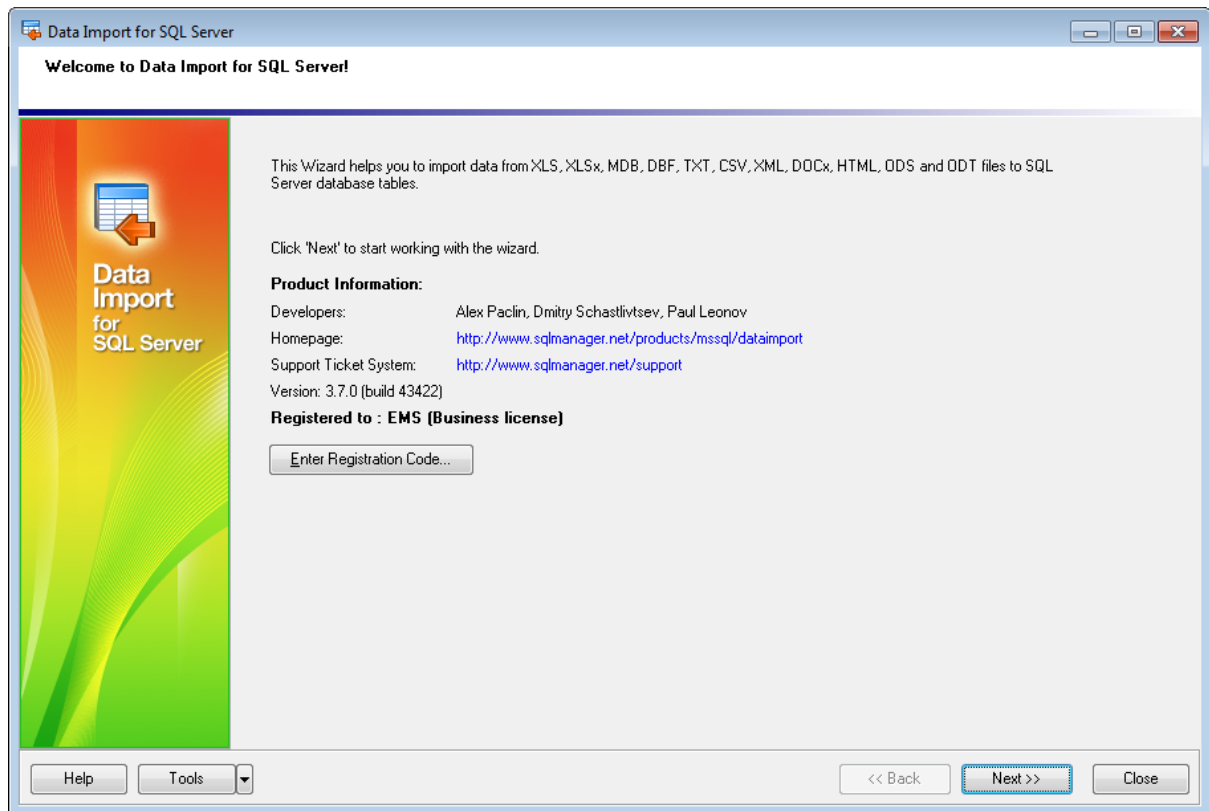
Press the **Next** button to proceed to <u>Setting connection properties</u> 25 .

See also:
<u>How to buy Data Import</u> 10

2.1.2   Step 1 - Setting connection properties

At this step you should specify necessary settings to establish **connection** to the target SQL Server database.

Connection settings

For connection you should enter SQL Server host name in the **Host** field or select the needed host from the drop-down list. This list is formed using *SQL-DMO* components function - "ListAvailableSQLServers". If *SQL-DMO* components are not installed the "NetServerEnum" function from *netapi32.dll* is used. If neither SQL-DMO components, nor netapi32.dll are installed, the **Host** drop-down list will be empty when the program is launched for the first time, and filled with host names already registered in the program, during further launches.

Please note that if Microsoft® SQL Server™ is installed as a named instance, you should specify 'Remote' as the connection type and enter the name, or IP address of your

machine and the instance name in the 'Host' field in the following format: computer_name\sqlserver_instance_name or ip_address\sqlserver_instance_name (e.g. "MYCOMPUTER\SQLEXPRESS", "127.0.0.1\SQLEXPRESS").

Authentication Type
Specify the type of SQL Server authentication to be used for the connection: *SQL Server* or *Windows* 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: **Login** and **Password**.

For your convenience the **Connection timeout** option is implemented: set this option to optimize the performance of the utility upon connection to your instance of SQL Server. Use the **Command timeout** option to set time available for command execution.
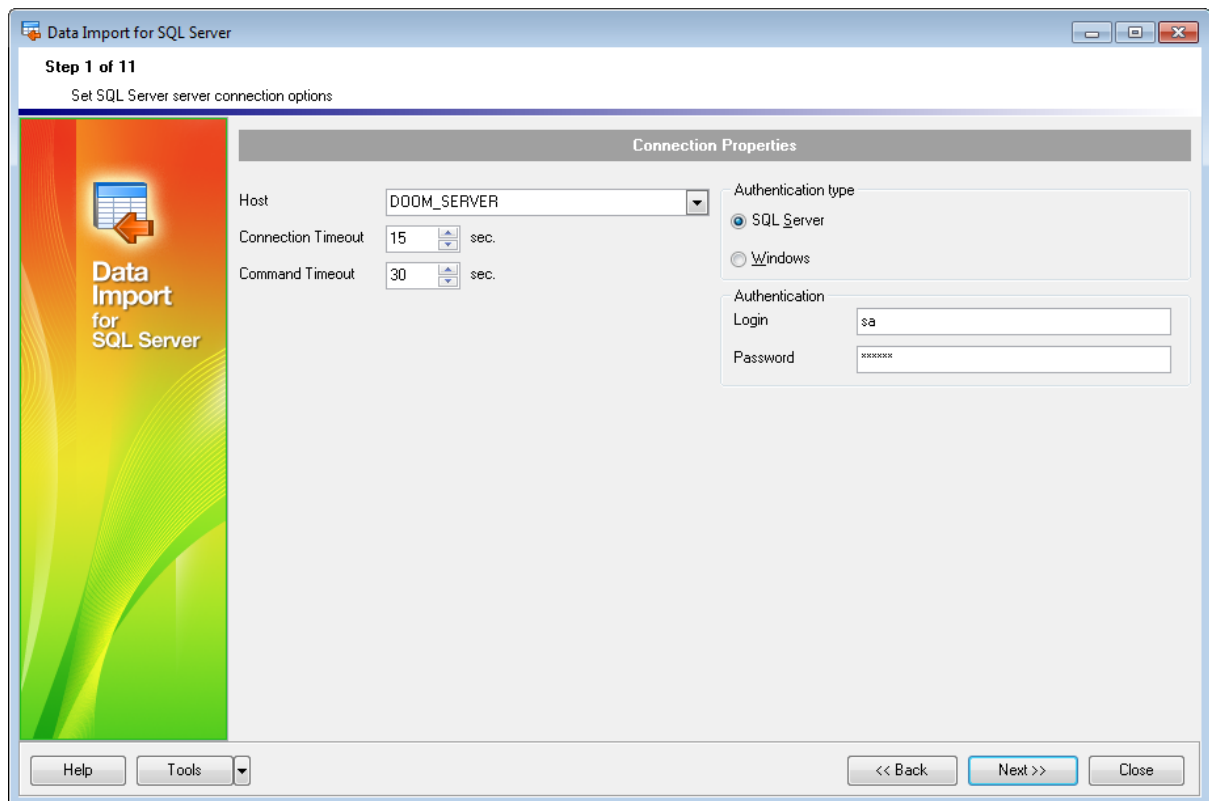
Please note that you need to have sufficient privileges to be able to write to the destination database on SQL Server.

If you get any difficulties connecting to the database, it can be useful to look through FAQs 13.

When you are done, press the **Next** button to proceed to the next step 27.

2.1.3 Step 2 - Selecting files to import

At this step you should **select source file(s) to be imported**.

Click the 🗒 **Add File** button to select the source file name using the **Open file...** dialog.
Repeat this operation to add more source files (if necessary).
You can choose among the following types of the source data file:
- *MS Excel*
- *MS Excel 2007*
- *MS Word 2007*
- *MS Access*
- *DBF*
- *TXT*
- *CSV*
- *HTML*
- *XML Data Packet*
- *XML MS Access*
- *XML Doc*
- *ODS*
- *ODT*

The open file dialog allows you to set a filter on the source file format.
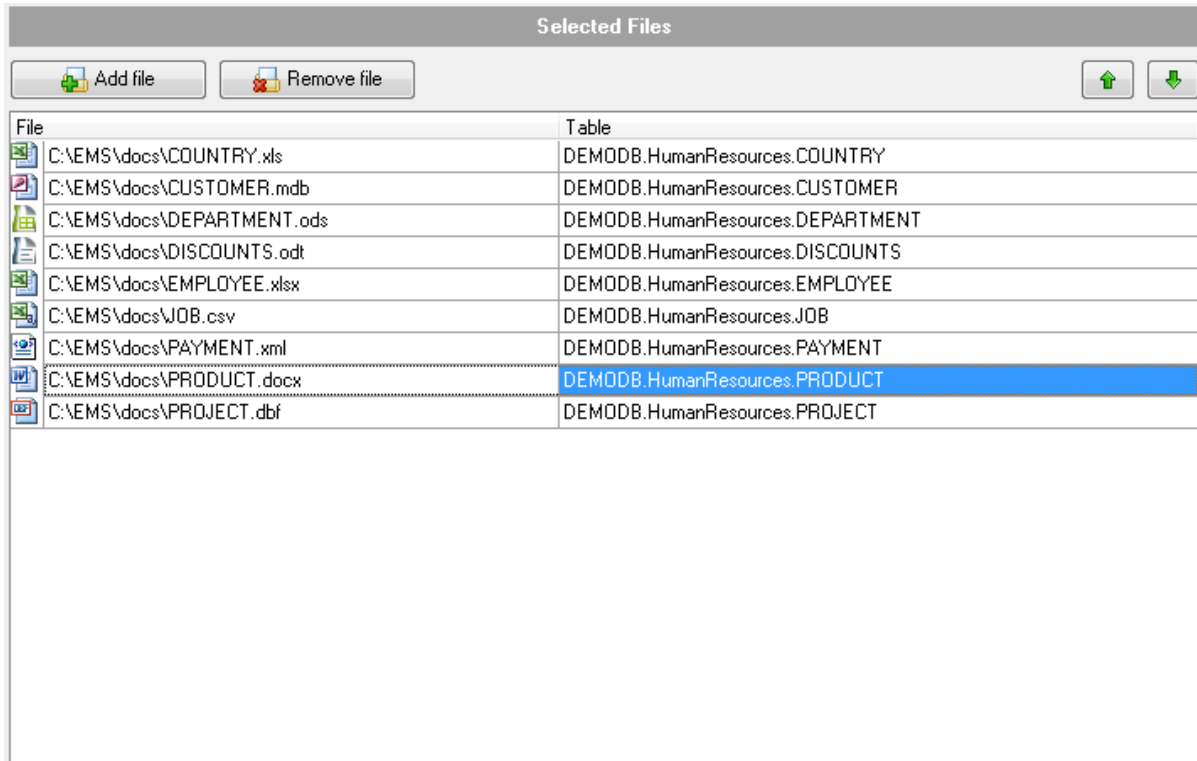
If you have selected the file of a format that is not supported the **Select Import Type** dialog appears.
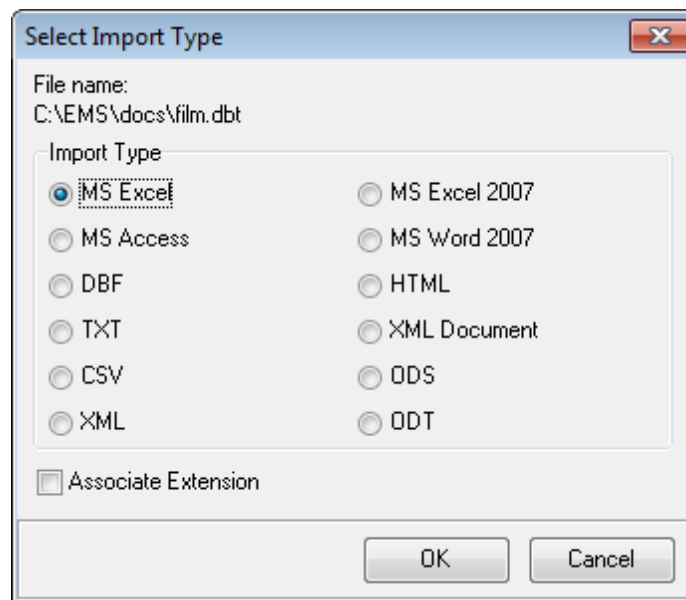
Here you can select which import type for supported file formats should be applied to import the selected file.

To delete a file from the list, select it and click the 🗒 **Remove File** button.

You can also change the order of the source files in the list using the **Up** [↑] and the **Down** [↓] buttons.

File	Table
C:\EMS\docs\COUNTRY.xls	DEMODB.HumanResources.COUNTRY
C:\EMS\docs\CUSTOMER.mdb	DEMODB.HumanResources.CUSTOMER
C:\EMS\docs\DEPARTMENT.ods	DEMODB.HumanResources.DEPARTMENT
C:\EMS\docs\DISCOUNTS.odt	DEMODB.HumanResources.DISCOUNTS
C:\EMS\docs\EMPLOYEE.xlsx	DEMODB.HumanResources.EMPLOYEE
C:\EMS\docs\JOB.csv	DEMODB.HumanResources.JOB
C:\EMS\docs\PAYMENT.xml	DEMODB.HumanResources.PAYMENT
C:\EMS\docs\PRODUCT.docx	DEMODB.HumanResources.PRODUCT
C:\EMS\docs\PROJECT.dbf	DEMODB.HumanResources.PROJECT

Note: For the spreadsheet files: if you need to import data from different sheets into different tables you are to include that multi-sheet spreadsheet file into the list several times (equal to the number of sheets to be imported to different tables).

When a file is selected the **Select Table** dialog appears:
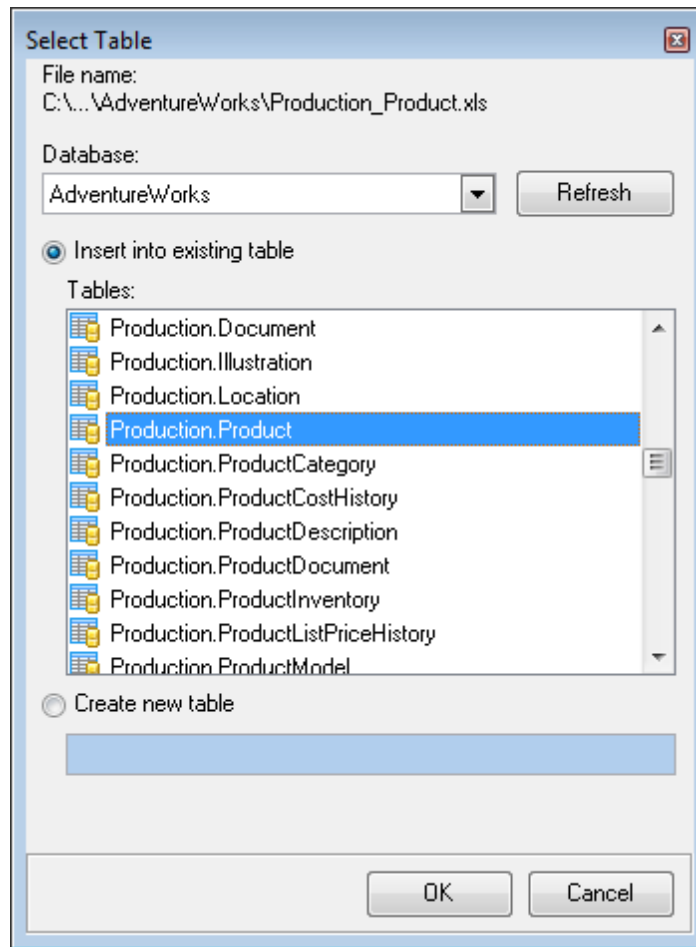
File name field displays the selected file name and its location.
Database field allows you to specify the target database.

⦿ **Insert into existing table**
Select this option to import data to the existing table. **Tables** area contains the list of tables in the selected database. You can refresh the list by clicking the **Refresh** button.

⦿ **Create new table**
Use this option to create import destination table.

To change the target SQL Server table that has been already assigned to a source data file, select the table in the list and press the ellipsis ⬚ button to call the **Select Table** dialog again.

When you are done, press the **Next** button to proceed to the <u>next step</u> 30 .

2.1.4 Step 3 - Setting XML file type

This step appears if you have selected **.xml* file for importing data from.
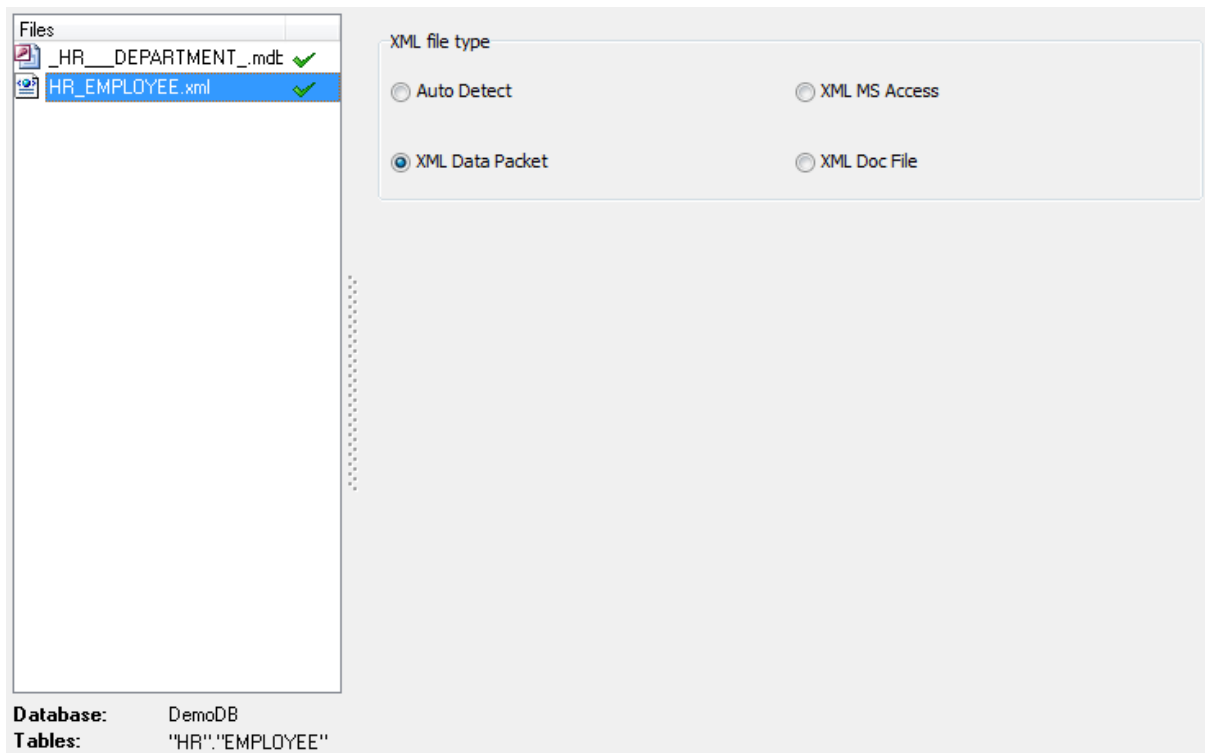
XML file type
Select the type of *.xml* file from which you are importing data:

◉ *Auto Detect* – file type will be detected automatically whether it is XML MS Access or XML data packet file;

◉ *XML MS Access* - select this option if your *.xml* file has the same structure as files exported from MS Access;

◉ *XML Data Packet* - select this option if your *.xml* file has data representation format used by Embarcadero;

◉ *XML Doc File* - select this option to map fields of a Generic XML document⌐40¬ manually.

2.1.5   Step 4 - Mapping fields

This step of the wizard allows you **to set correspondence** between columns of the source file and fields of the target SQL Server table, according to the source data format.

- MS Excel mapping⌐31¬
- MS Excel 2007 mapping⌐32¬
- MS Word 2007 mapping⌐35¬
- MS Access mapping⌐36¬
- DBF mapping⌐37¬
- XML mapping⌐39¬
- XML Document mapping⌐40¬
- TXT mapping⌐42¬
- CSV mapping⌐43¬
- HTML Document⌐45¬
- ODS mapping⌐46¬
- ODT mapping⌐48¬

To get more information about the file formats, see the <u>Supported file formats</u> 82 page.

When you are done, press the **Next** button to proceed to the <u>next step</u> 49.

2.1.5.1   MS Excel

Select the needed source file from the list. Then 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 file and the destination SQL Server table have the same order of columns, you can use the [icon] **Auto Fill Columns** mode (default) to set correspondence between them automatically. If source file and destination SQL Server table have different order of columns but identical names you can use the [icon] **Auto Fill by Captions** mode to set the correspondence based on name's identity. You can also use the [icon] **Auto fill Rows** to set the correspondence between destination table fields and source file rows.

If new table creation was selected at the <u>previous step</u> 27, then fields needed for the import procedure will be created automatically.
You can <u>manage destination table fields</u> 83 if needed. Use the corresponded context menu item to **Add**/**Edit**/**Drop** field.
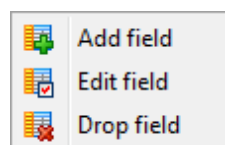
Note: If table was properly created or already exist, it will be marked with a tick ✔. If an error occur during table creation, this table will be marked with a cross ✖.

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.
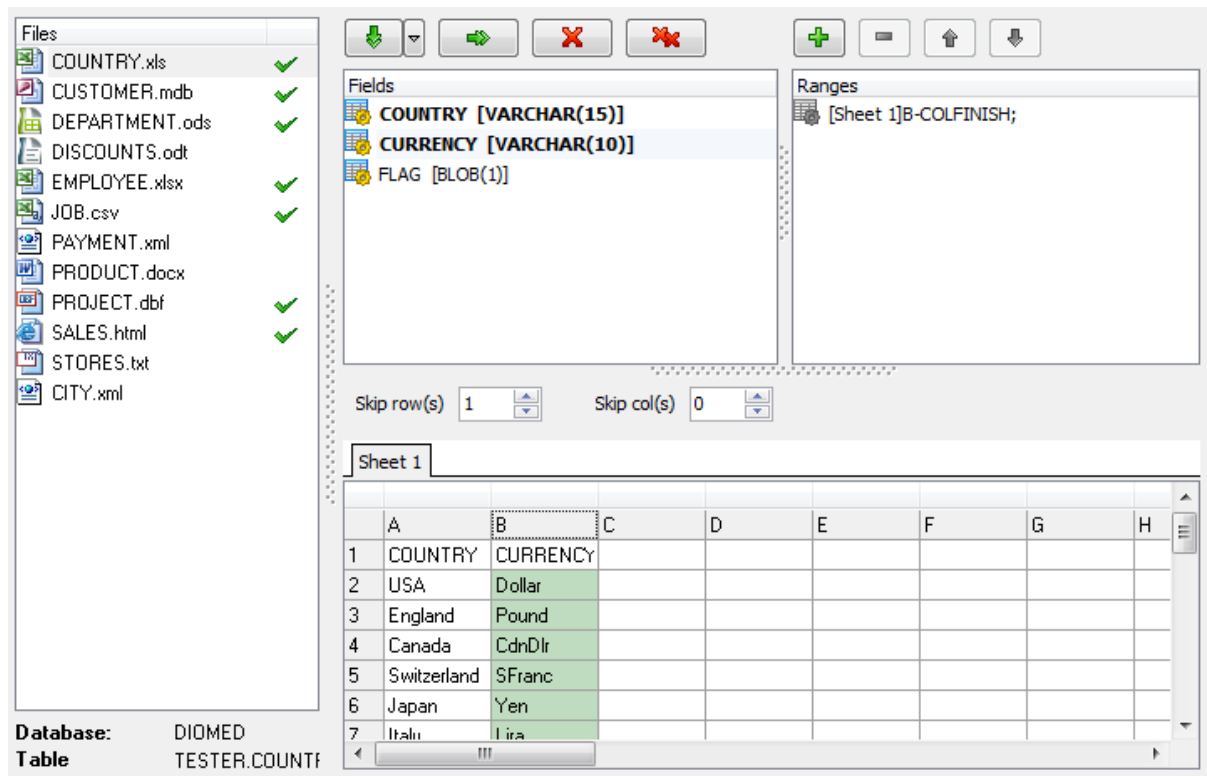
If your spreadsheet file contains several sheets, you are able to set different mapping for each sheet.

To clear ranges for a field, select the field in the **Fields** list and press the [X] **Clear Field Ranges** button.

To clear all ranges specified for the target table fields, press the [X] **Clear All ranges** button.

To set a range of data|85 to be imported from the file, use the [+] **Add range** button.

To remove a range, use the [—] **Delete range** button.

Using the [↑] **Move Up** and the [↓] **Move Down** buttons you can change the order of ranges applied to data.

Click the **Next** button to proceed to the Setting base data formats|49 step of the wizard.

2.1.5.2 MS Excel 2007

Select the needed source file from the list. Then 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;
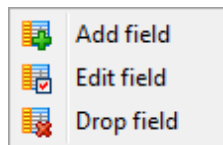* repeat the operation for all the fields you need to be included in the import process.

If the source file and the destination SQL Server table have the same order of columns, you can use the [icon] **Auto Fill Columns** mode (default) to set correspondence between them automatically. If source file and destination SQL Server table have different order of columns but identical names you can use the [icon] **Auto Fill by Captions** mode to set the correspondence based on name's identity.

Note: The number of the column for which correspondence was set is displayed at the **Col.** control. You can use it for setting the fields correspondence as well.

If new table creation was selected at the previous step 27, then fields needed for the import procedure will be created automatically.
You can manage destination table fields 83 if needed. Use the corresponded context menu item to **Add**/**Edit**/**Drop** field.
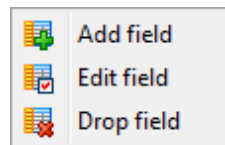
If necessary, you can choose to **skip** a defined number of the source file rows or columns using the **Skip lines** or **Skip cols** spin-edit.

Note: If table was properly created or already exist, it will be marked with a tick ✔. If an error occur during table creation, this table will be marked with a cross ✖.

To remove a correspondence, select the field in the **Fields** list and press the [X] **Clear Field Ranges** button.

To remove all correspondences, press the [X] **Clear All Ranges** button.

If your spreadsheet file contains several sheets, you are able to set different mapping for each sheet.

To clear ranges for a field, select the field in the **Fields** list and press the **Clear Field Ranges** [X] button.

To clear all ranges specified for the target table fields, press the **Clear All ranges** [X] button.

To set a range of data 85 to be imported from the file, use the **Add range** [+] button.

To remove a range, use the **Delete range** [-] button.

Using the **Move Up** [↑] and the **Move Down** [↓] buttons you can change the order of ranges applied to data.

Click the **Next** button to proceed to the Setting base data formats 49 step of the wizard.

2.1.5.3 MS Word 2007

Specify ranges in the grid for the target and source fields:
- select a field of the target SQL Server table in the **New fields** list;
- proceed to the **Table** 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, you can use the ⬇️▾ **Auto Fill Columns** mode (default) to set correspondence between them automatically. If source file and destination SQL Server table have different order of columns but identical names you can use the ⬇️▾ **Auto Fill by Captions** mode to set the correspondence based on name's identity.

The number of the column which correspondence is set for are displayed at the **Grid Column** column. You can use it for setting the fields correspondence as well.

If new table creation was selected at the previous step 27, then fields needed for the import procedure will be created automatically.
You can manage destination table fields 83 if needed. Use the corresponded context menu item to **Add**/**Edit**/**Drop** field.

If necessary, you can choose to **skip** a defined number of the source file rows using the **Skip lines** spin-edit.

Note: If table was properly created or already exist, it will be marked with a tick ✔. If an error occur during table creation, this table will be marked with a cross ✖.

To remove a correspondence, select the field in the **New fields** list and press the [X] **Clear Field Ranges** button.

To remove all correspondences, press the [X] **Clear All Ranges** button.

Click the **Next** button to proceed to the Setting base data formats [49] step of the wizard.
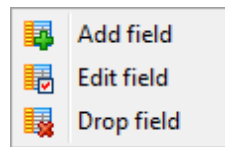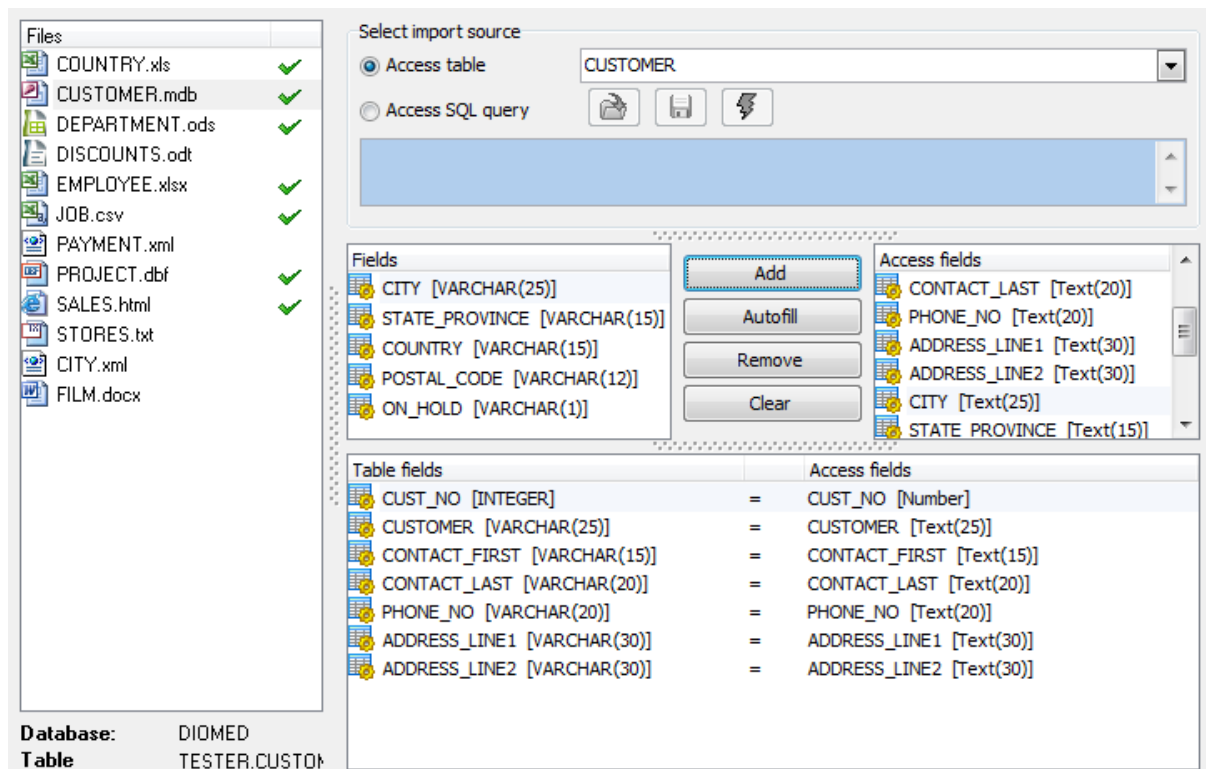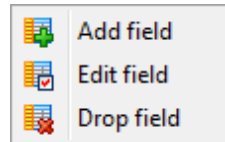
2.1.5.4 MS Access

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 **Table fields** list;
- select the corresponding field of the source MS Access table in the **Access 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.

If new table creation was selected at the previous step [27], then fields needed for the import procedure will be created automatically.
You can manage destination table fields [83] if needed. Use the corresponded context menu item to **Add/Edit/Drop** field.

Note: If table was properly created or already exist, it will be marked with a tick ✔. If an error occur during table creation, this table will be marked with a cross ✖.

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.

Use the **Autofill** button to set correspondence between the source and target fields automatically on the basis of their order.

To remove a correspondence, select a 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 step of the wizard.

2.1.5.5 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 **Table Fields** list;

- select the corresponding column of the source DBF table in the **DBF 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.

If new table creation was selected at the <u>previous step</u> [27], then fields needed for the import procedure will be created automatically.

You can <u>manage destination table fields</u> [83] if needed. Use the corresponded context menu item to **Add**/**Edit**/**Drop** field.

Note: If table was properly created or already exist, it will be marked with a tick ✔. If an error occur during table creation, this table will be marked with a cross ✖.

Use the **Autofill** button to set correspondence between the source and target fields automatically on the basis of their order.

Charset
Use this field to specify the source file character set.

☑ **Skip deleted rows**
Use the option to exclude records marked as deleted in source DBF file.

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 Setting base data formats⌐49⌐ step of the wizard.
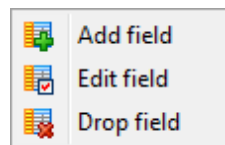
2.1.5.6   XML

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 **Table Fields** list;
- select the corresponding column of the source XML table in the **XML 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.

If new table creation was selected at the previous step⌐27⌐, then fields needed for the import procedure will be created automatically.
You can manage destination table fields⌐83⌐ if needed. Use the corresponded context menu item to **Add**/**Edit**/**Drop** field.
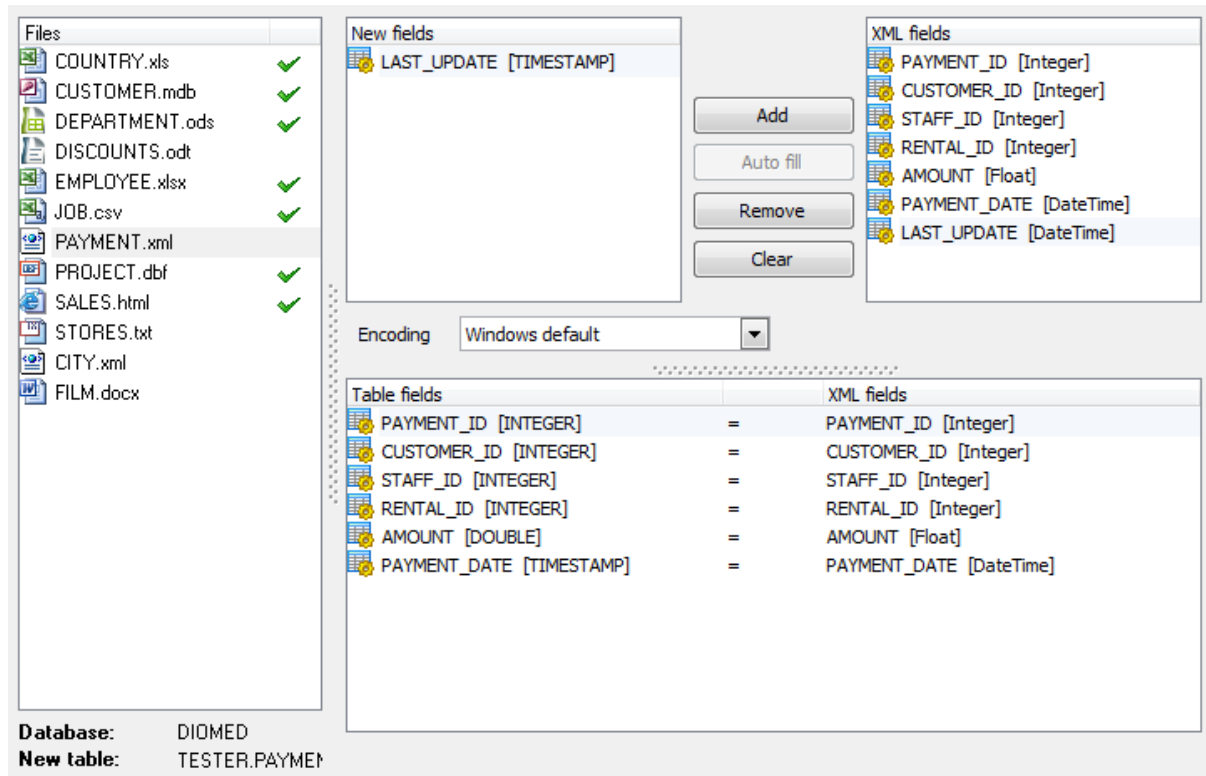

Note: If table was properly created or already exist, it will be marked with a tick ✔. If an error occur during table creation, this table will be marked with a cross ✖.

Use the **Autofill** button to set correspondence between the source and target fields automatically on the basis of their order.

Encoding
Use this field to select the file encoding.

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 Setting base data formats [49] step of the wizard.

2.1.5.7 XML Document

In order to set mapping of a Generic XML document, you should first select the desired XML Document in the **Files** list, then press the **Build Tree** button to get the tree-like structure of source document in the area located to the right from the **New fields** list. Select a node in the tree to get its relative path or type the path manually in the editor below (the path must be specified in the XPath format). Upon pressing the **Fill grid** button the grid gets filled with *Sub Nodes Text* or *Attributes* values of the selected node.

Note: if the source XML document contains huge amount of data, building the tree may take a long time.

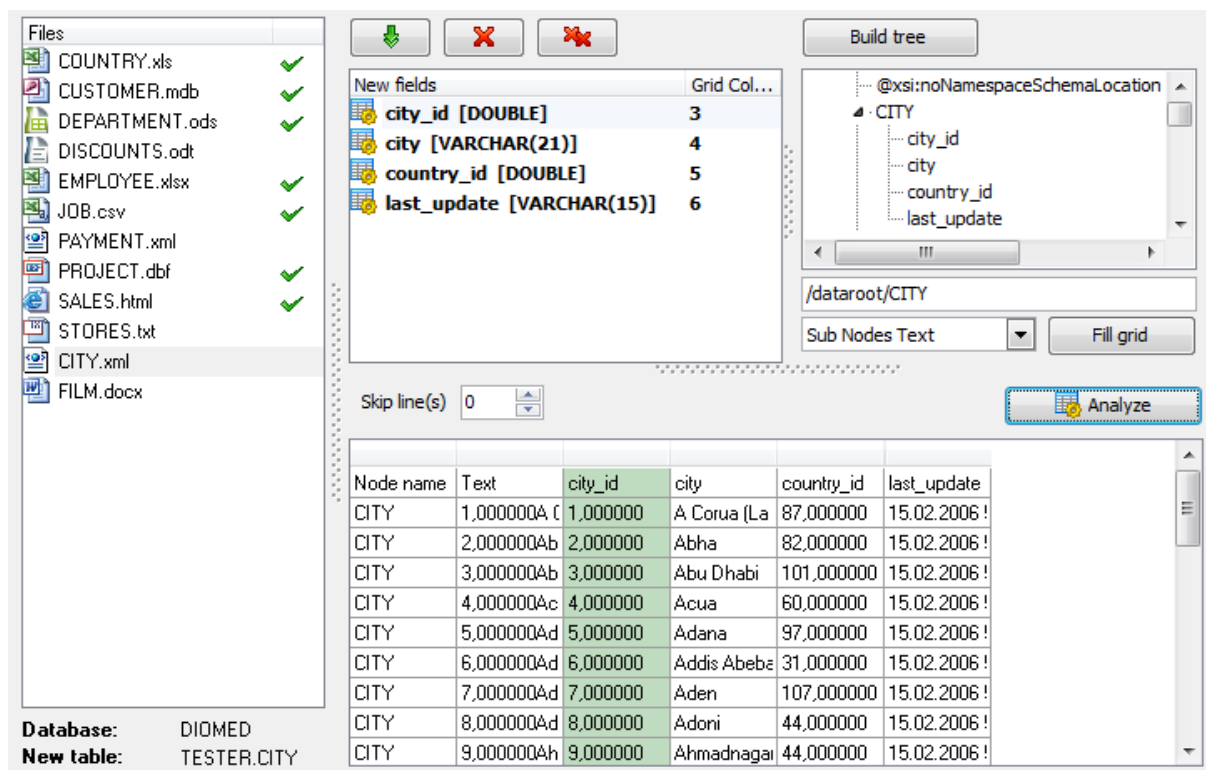Set correspondence between the source XML file columns and the target SQL Server table fields:
- select a field of the target SQL Server table in the **Fields** list;
- proceed to the source grid viewer area: click a column to assign the column to the selected target table field;
- the selected column of the source file gets gray highlight;
- repeat the operation for all the fields you need to be included in the import process.

You can use the [↓] **Autofill** 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).

When you import data to a newly created table, the **Analyze** button appears. Use this button for automatic field creation.

The number of the column which correspondence is set for are displayed at the **Grid Column** column. You can use it for setting the fields correspondence as well.

If necessary, you can choose to **skip** a defined number of the source file lines using the **Skip Lines** spin-edit.

To remove all correspondences, press the [×] **Clear All Ranges** button.

If you choose to import data to the newly created table, at this step you need to create necessary fields. Use the corresponded context menu item to [manage fields 83].

Note: If table was properly created or already exist, it will be marked with a tick ✔. If an error occurs during table creation, this table will be marked with a cross ✖.

Click the **Next** button to proceed to the Setting base data formats 49 step of the wizard.

2.1.5.8   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.

The **Fields** list also allows you to view the following values:
P represents the starting point of your selection;
S displays the width of the selected area.
You can change these parameters manually or by moving the slider in the grid.

If necessary, you can choose to **skip** a defined number of the source file lines using the **Skip lines** spin-edit.
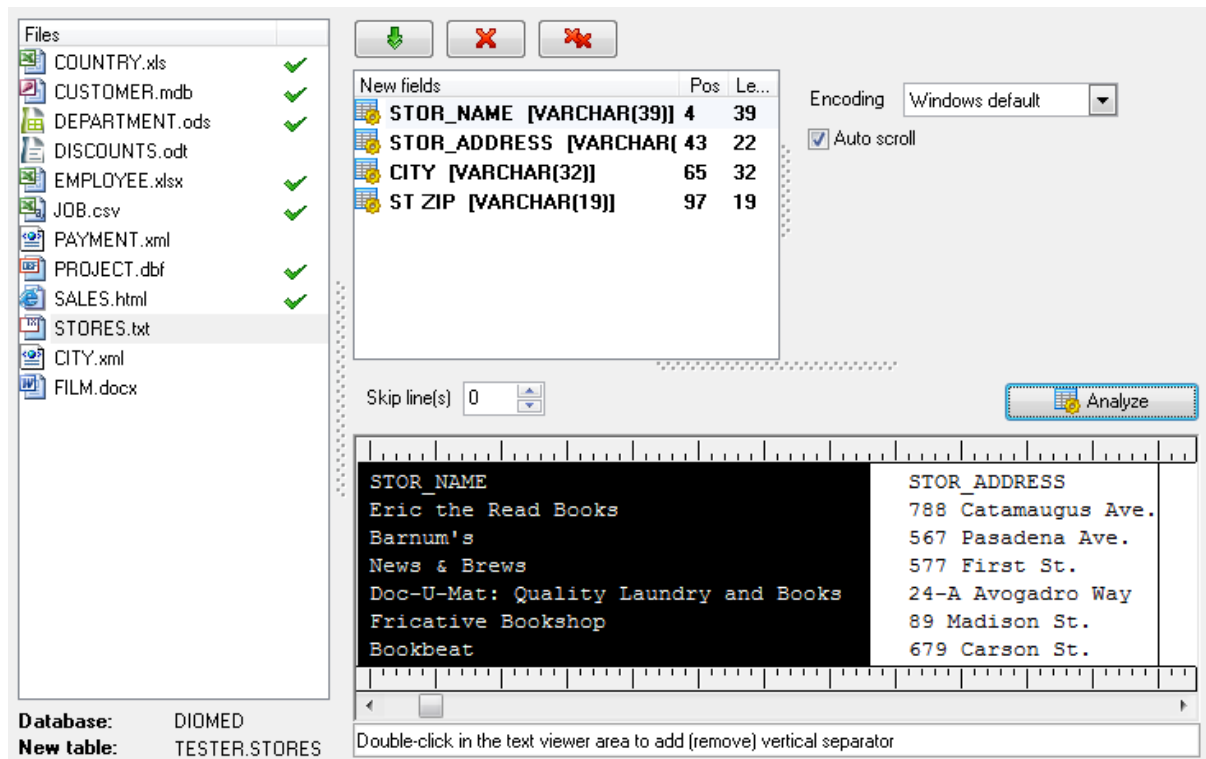If the source text file and the destination SQL Server table have the same order of columns, you can use the [↓] **Autofill Columns** button to set correspondence between them automatically.

The source file character set can be defined at the **Encoding** field.

☑ **Autoscroll**
This option scrolls the document automatically when you switch to the next field for mapping.

When you import data to a newly created table, the **Analyze** button appears. Use this button for automatic field creation.

To remove all correspondences, press the [🗙🗙] **Clear All Ranges** button.

If you choose to import data to the newly created table, at this step you need to create necessary fields. Use the corresponded context menu item to [manage fields](83).

Note: If table was properly created or already exist, it will be marked with a tick ✔. If an error occur during table creation, this table will be marked with a cross ✖.

Click the **Next** button to proceed to the [Setting base data formats](49) step of the wizard.

2.1.5.9   CSV

Set correspondence between the target table fields and the source CSV file columns:
- select the **Delimiter** and **Quote** characters for CSV files using the corresponding drop-down lists of the **CSV Parameters** group;
- 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.
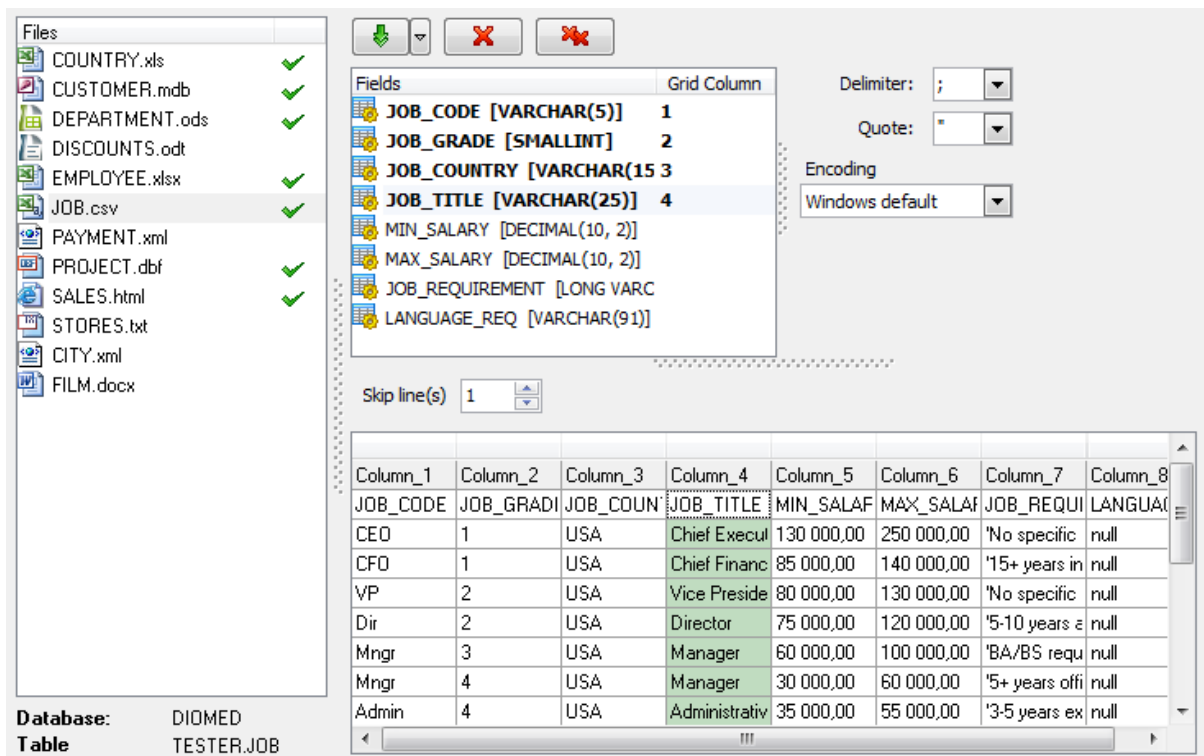
The source file character set can be defined at the **Encoding** field.

If the source file and the destination SQL Server table have the same order of columns, you can use the ⬇️ **Auto Fill Columns** mode (default) to set correspondence between them automatically. If source file and destination SQL Server table have different order of columns but identical names you can use the ⬇️ **Auto Fill by Captions** mode to set the correspondence based on name's identity.

When you import data to a newly created table, the **Analyze** button appears. Use this button for automatic field creation.

If necessary, you can choose to **skip** a defined number of the source file rows using the **Skip lines** spin-edit.

The number of the column which correspondence is set for are displayed at the **Grid Column** column. You can use it for setting the fields correspondence as well.
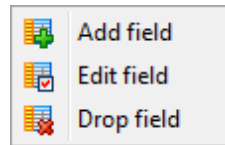

To remove all correspondences, press the ❌ **Clear All Ranges** button.

If you choose to import data to the newly created table, at this step you need to create necessary fields. Use the corresponded context menu item to <u>manage fields</u>[83].

Note: If table was properly created or already exist, it will be marked with a tick ✔. If an error occur during table creation, this table will be marked with a cross ✖.

Click the **Next** button to proceed to the Setting base data formats 49 step of the wizard.

2.1.5.10 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 **New 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 file and the destination SQL Server table have the same order of columns, you can use the **Auto Fill Columns** mode (default) to set correspondence between them automatically. If source file and destination SQL Server table have different order of columns but identical names you can use the **Auto Fill by Captions** mode to set the correspondence based on name's identity.

The number of the column which correspondence is set for are displayed at the **Grid Column** column. You can use it for setting the fields correspondence as well.

If new table creation was selected at the previous step 27, then fields needed for the import procedure will be created automatically.
You can manage destination table fields 83 if needed. Use the corresponded context menu item to **Add**/**Edit**/**Drop** field.

If necessary, you can choose to **skip** a defined number of the source file rows using the **Skip lines** spin-edit.

Note: If table was properly created or already exist, it will be marked with a tick ✔. If an error occur during table creation, this table will be marked with a cross ✖.
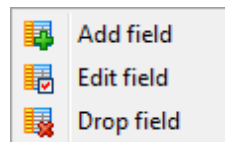
To remove all correspondences, press the [⚔] **Clear All Ranges** button.

Click the **Next** button to proceed to the [Setting base data formats](49) step of the wizard.

2.1.5.11 ODS

The **OpenDocument** format is used by Mobile Office, as well as other well-known desktop applications, such as OpenOffice, StarOffice and KOffice. **ODS** stands for **OpenDocument Spreadsheet** (*.ods*).

Specify ranges in the grid for the target and source fields:
- select a field of the target SQL Server table in the **New 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, you can use the [⬇] **Auto Fill Columns** mode (default) to set correspondence between them automatically. If source file and destination SQL Server table have different order of columns but identical names you can use the [⬇] **Auto Fill by Captions** mode to set the correspondence based on name's identity.

The number of the column which correspondence is set for are displayed at the **Grid Column** column. You can use it for setting the fields correspondence as well.
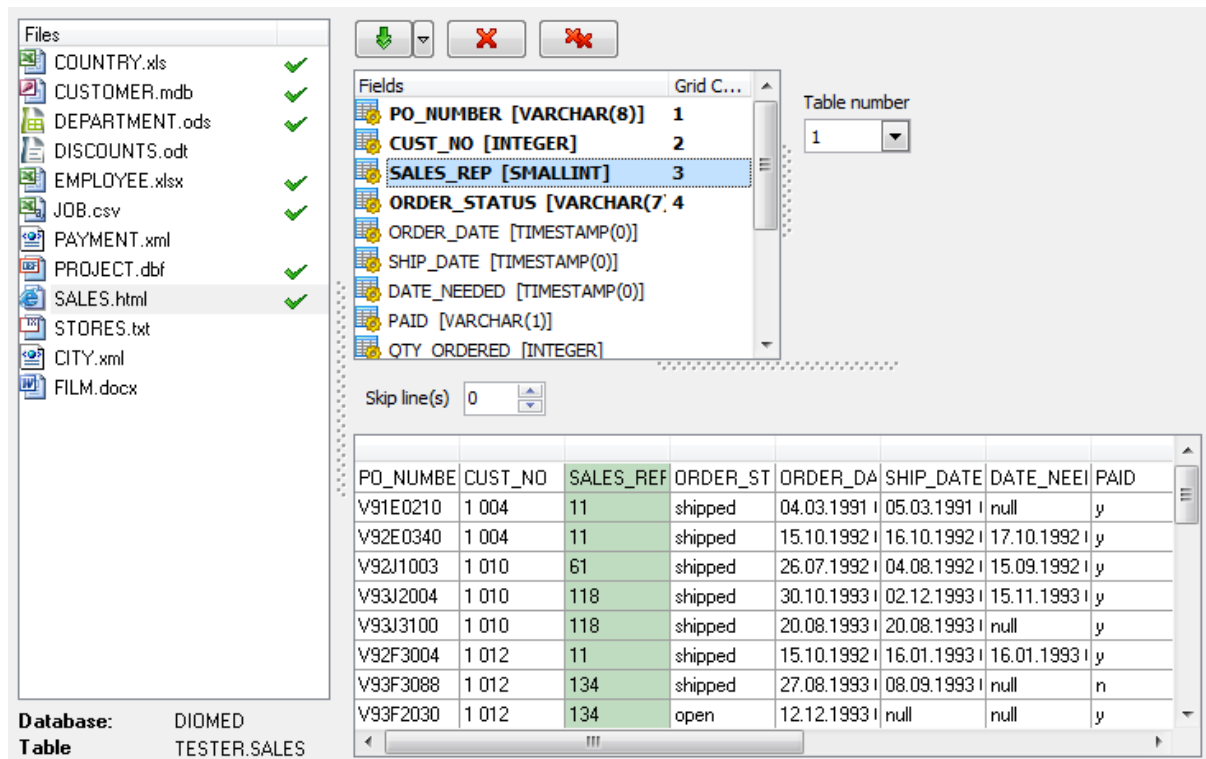
If new table creation was selected at the previous step 27, then fields needed for the import procedure will be created automatically.
You can manage destination table fields 83 if needed. Use the corresponded context menu item to **Add/Edit/Drop** field.

If necessary, you can choose to **skip** a defined number of the source file rows using the **Skip lines** spin-edit.

Note: If table was properly created or already exist, it will be marked with a tick ✔. If an error occur during table creation, this table will be marked with a cross ✖.

To remove a correspondence, select the field in the **New fields** list and press the ✖ **Clear Field Ranges** button.

To remove all correspondences, press the ✖ **Clear All Ranges** button.

If your OpenDocument Spreadsheet file contains several sheets, you are able to set different mapping for each sheet.

Click the **Next** button to proceed to the Setting base data formats 49 step of the wizard.

2.1.5.12 ODT

The **OpenDocument** format is used by Mobile Office, as well as other well known desktop applications such as OpenOffice, StarOffice and KOffice. **ODT** stands for **OpenDocument Text** (*.odt*).

Specify ranges in the grid for the target and source fields:
- select a field of the target SQL Server table in the **New fields** list;
- proceed to the **Table** 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, you can use the ![icon] **Auto Fill Columns** mode (default) to set correspondence between them automatically. If source file and destination SQL Server table have different order of columns but identical names you can use the ![icon] **Auto Fill by Captions** mode to set the correspondence based on name's identity.

The number of the column which correspondence is set for are displayed at the **Grid Column** column. You can use it for setting the fields correspondence as well.

If new table creation was selected at the previous step 27, then fields needed for the import procedure will be created automatically.
You can manage destination table fields 83 if needed. Use the corresponded context menu item to **Add**/**Edit**/**Drop** field.

![context menu showing Add field, Edit field, Drop field]

If necessary, you can choose to **skip** a defined number of the source file rows using the **Skip lines** spin-edit.

Note: If table was properly created or already exist, it will be marked with a tick ✔. If an error occur during table creation, this table will be marked with a cross ✖.

To remove a correspondence, select the field in the **Fields** list and press the [X] **Clear Field Ranges** button.

To remove all correspondences, press the [XX] **Clear All Ranges** button.

If your OpenDocument Text file contains several tables, you are able to set different mapping for each table.

Click the **Next** button to proceed to the <u>Setting base data formats</u> step of the wizard.

2.1.6   Step 5 - Setting base data formats

This step of the wizard provides a number of options for setting **base formats** for each source data file. The specified format should match the source data representation.

Separators
Decimal, *Thousand*, *Date*, *Time*

Decimal
Set a character to delimit the decimal parts of the imported numbers.
Thousand
Set a character to separate the digit groups in the imported numbers.
Date
Set a character to separate the year, month and day parts of date values.
Time
Set a character to separate the hour, minute and second parts of time values.

Constants

Boolean True
Set one or more variants of TRUE value representation in the imported table, e.g. 'Yes' or '+'. Use a new line for each additional variant.

Boolean False
Set one or more variants of FALSE value representation in the imported table, e.g. 'No' or '-'. Use a new line for each additional variant.

Null Values
Set one or more variants of NULL value representation in the imported table, e.g. 'Null'.

Use a new line for each additional variant.

Date/Time formats
Use these edit fields to set the date and time formats: *Date, Time.*

For more information refer to the Format specifiers 51 page.

☑ **Skip this step**
Check this option to skip the current step in the future. To edit the list of skipped steps, use the **Skipped Steps** group available in the General 68 section of the Preferences 68 dialog.

When you are done, press the **Next** button to proceed to the next step 54.

2.1.6.1 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
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).

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.

2.1.7   Step 6 - Settings data formats for each field

This step of the wizard allows you to set **formats** for each imported field separately, in case additional formatting is required.

Select a field in the list and use the **Field Customization** group to adjust format options that will be applied to this field only: *generator value*, *generator step*, *constant value*, *NULL value*, *default value*, *function*, *script*, *left/right quotation*, *quotation action*, *character case*, *character set*.

Generator value
Use this field to set the initial value of the autoincrement field.

Generator step
Set the step of the autoincrement field. If it is *0* or if the field is of *Identity* one, then the value of the generator will be ignored.

Constant value
Use this field to set a constant value for the field.

Null value
Set the value which will be considered as NULL to set the default value.

Default value
Set the default value of the NULL field.
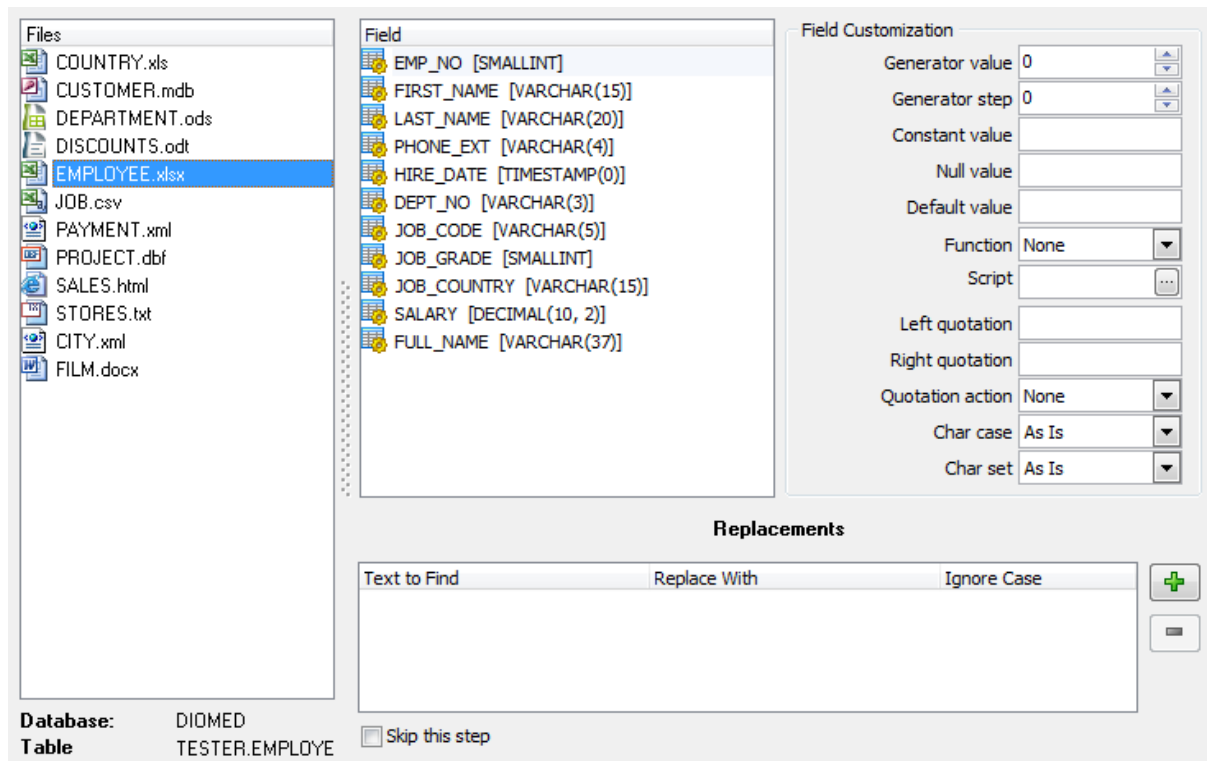
Function
Use the drop-down list to select one of the function return values that can be set in the field: *Date, Time, Date&Time, Long File Name, Short File Name*.

Script
Press the ⬚ button to add/edit the script in the **Script Editor**.
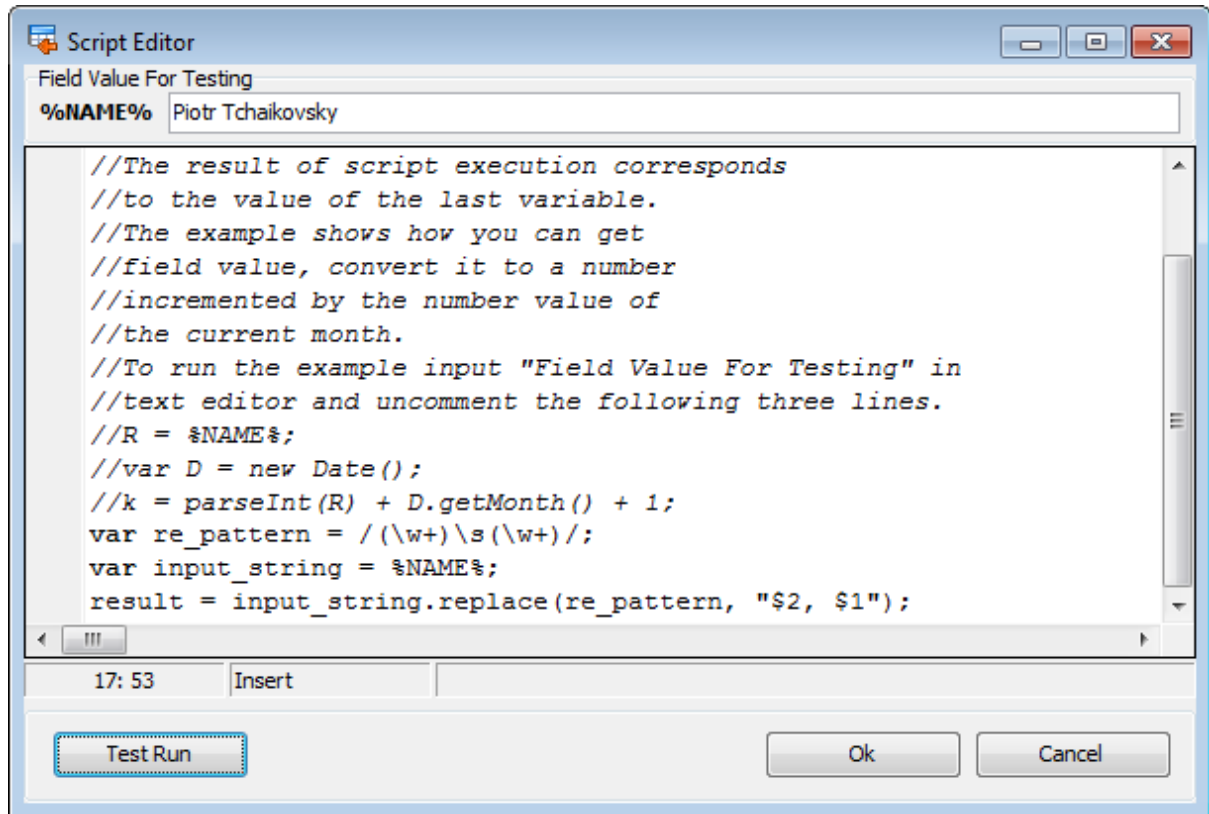
Use the **Script Editor** window to enter the script for the field processing. Here you can use all features of the MS JScript.

The picture above shows the example of the Jscript use for the field *Name* processing. The entered code converts the *'XXXX YYYY'* string to the *'YYYY, XXXX'* one, for example 'Piotr Tchaikovsky' -> 'Tchaikovsky, Piotr'.
If you want to test your code and see the result of its execution then enter the test field value in the **Field Value For Testing** box and click the **Test Run** button.
If the execution fails you see the message box with the description of an occurred mistake.
If the execution succeeds you see the message box with the result as it is shown on the picture below.
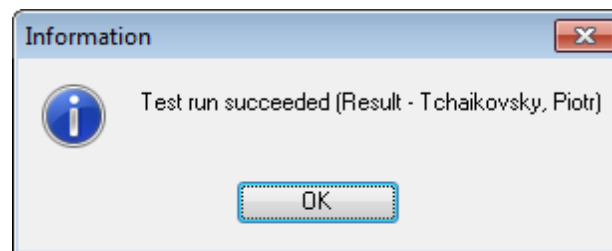
For more information about JScript, see http://msdn.microsoft.com/en-us/library/x85xxsf4. aspx.

Quotation

Left quotation
Set a character or a number of characters to denote quoting in the imported string.

Right quotation
Set a character or a number of characters to denote unquoting in the imported string.

Quotation action
You can select the *Add* item to add quotation marks to the imported string, the *Remove* item to remove all the quotation marks from the imported string, or the *None* item to save the original quotation marks.

String conversion

Char case
Set the case of the imported string. *As Is* keeps the original string unchanged, *Upper* sets the whole string to the upper case, *Lower* sets the whole string to the lower case, *UpperFirst* sets the first character of the string to the upper case, *UpperFirstWord* sets the first character of each word to the upper case.

Char set
Set the char set of the imported string to *ANSI* or *OEM*. *As Is* saves the original character set of the string.

The **Replacements** area allows you to set the text you need to be replaced during data import into the selected field. Press the **Add Replacement...** button to specify a new replacement options using the **Add Replacement** dialog.

Define the text to replace and the value to replace with in the appeared dialog window. Check the **Ignore Case** option to make replacement case-insensitive.
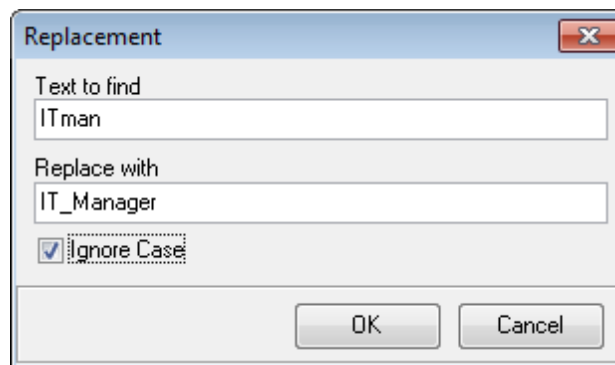
To remove a replacement, select it in the list and click the **Delete Replacement...** button.

☑ **Skip this step**
Check this option to skip the current step in the future. To edit the list of skipped steps, use the **Skipped Steps** group available in the General 68 section of the Preferences 68 dialog.

When you are done, press the **Next** button to proceed to the next step 58.

2.1.8   Step 7 - Specifying 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:
- **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 Data Import utility:

DB Table		Source file	
ID	**DATA**	**A**	**B**
1	a	1	c
2	b	2	d
4	f	3	e

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).

Insert mode	Insert all		Insert new		Update	
Result	**ID**	**DATA**	**ID**	**DATA**	**ID**	**DATA**
	1	c	1	a	1	c
	1	a	2	b	2	d
	2	b	3	e	4	f
	2	d	4	f		
	3	e				
	4	f				

Insert mode	Update or insert		Delete		Delete or insert	
Result	ID	DATA	ID	DATA	ID	DATA
	1	c	4	f	3	e
	2	d			4	f
	3	e				
	4	f				

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 at the <u>Selecting key columns</u> ⁶⁰ step of the wizard.
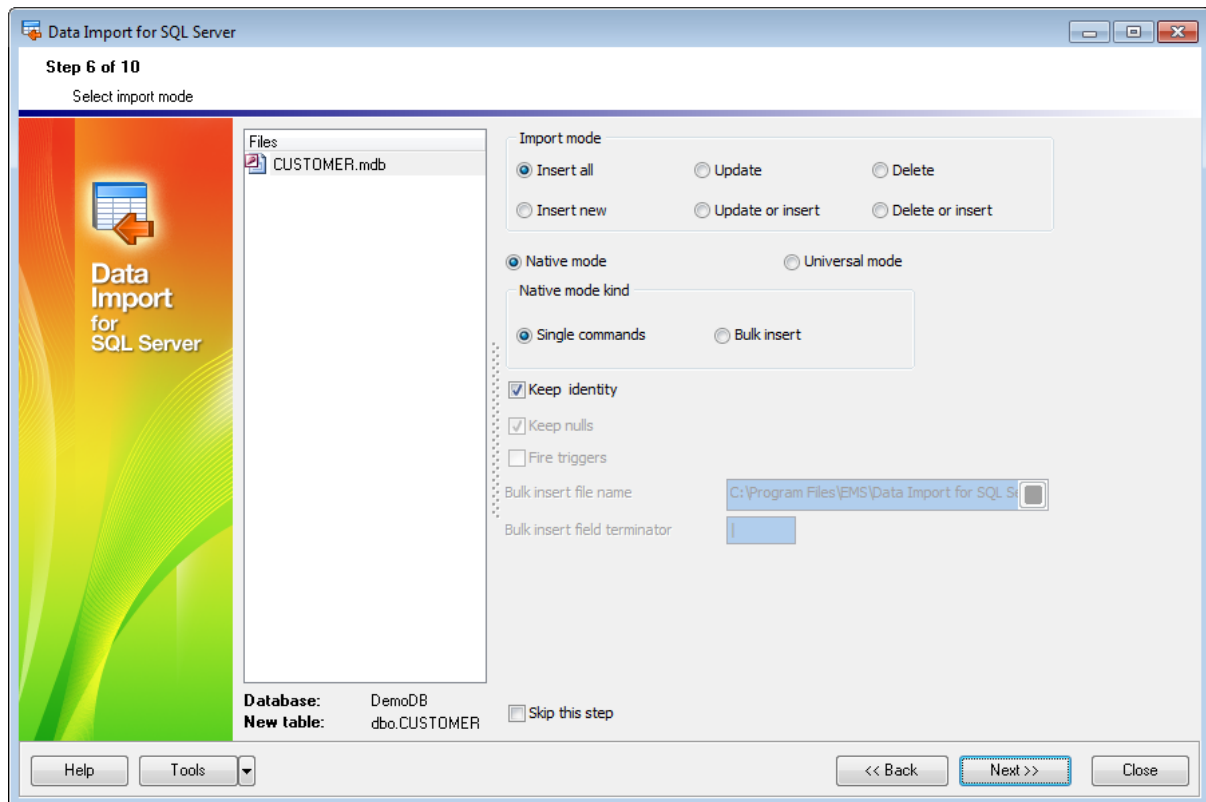
If the *Update* value is selected for Import Mode, then you need to specify the Primary Key field(s) that will serve for data identification of your source file with the data of the target table. The source file must contain the column(s) that will correspond to the Primary Key field of the target table. If the target table contains a record in which the value of the primary key field coincides with the value of the corresponding column of the source table, the data stored in this table record will be updated. You can specify the field(s) which will be used for identification at <u>Step 7</u> ⁶⁰ of the wizard. This field(s) will only serve for identification and will not be imported.

Native / Universal mode

The *Native* mode uses the *Single Commands* method that 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 *Native* mode your data can be imported dozen (!) times faster as compared to the *Universal* mode which is used for backward compatibility. For the *Universal* mode **Cursor location** needs to be defined: either ◉ **Server** or ◉ **Client**.

☑ **Keep Identity**

This option specifies that the values for an identity column are present in the file being imported.

☑ **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

Specify the field terminator to be used for char and varchar data files.

☑ **Truncate Long Strings**

If this option is enabled the long strings are truncated. This option is available only in the **Universal** mode.

☑ **Skip this step**

Check this option to skip the current step in the future. To edit the list of skipped steps, use the **Skipped Steps** group available in the <u>General</u>⌐68⌐ section of the <u>Preferences</u>⌐68⌐ dialog.

When you are done, press the **Next** button to proceed to the <u>next step</u>⌐60⌐.

2.1.9   Step 8 - Selecting key columns

This step of the wizard allows you to select the fields of the table to be used as the **key fields** for the import process.

This step is not available if the ◉ *Insert* all option has been selected on the <u>previous step</u> ⌐58⌐.

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.

✓ Skip this step
Check this option to skip the current step in the future. To edit the list of skipped steps, use the **Skipped Steps** group available in the General[68] section of the Preferences[68] dialog.

When you are done, press the **Next** button to proceed to the next step[61].

2.1.10 Step 9 - Setting common options

Using this step of the wizard you can set **final import options**.

Commit

✓ Commit when done
Check this option to commit the transaction after all records are imported.

Commit after each ... records
These control allows you to define the number of records in each block to be supplemented with the COMMIT statement.

✓ Rollback on error
Check this option to rollback the transaction if an error occurred.

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.

☑ **Import empty rows**
If this option is checked then empty records are imported.
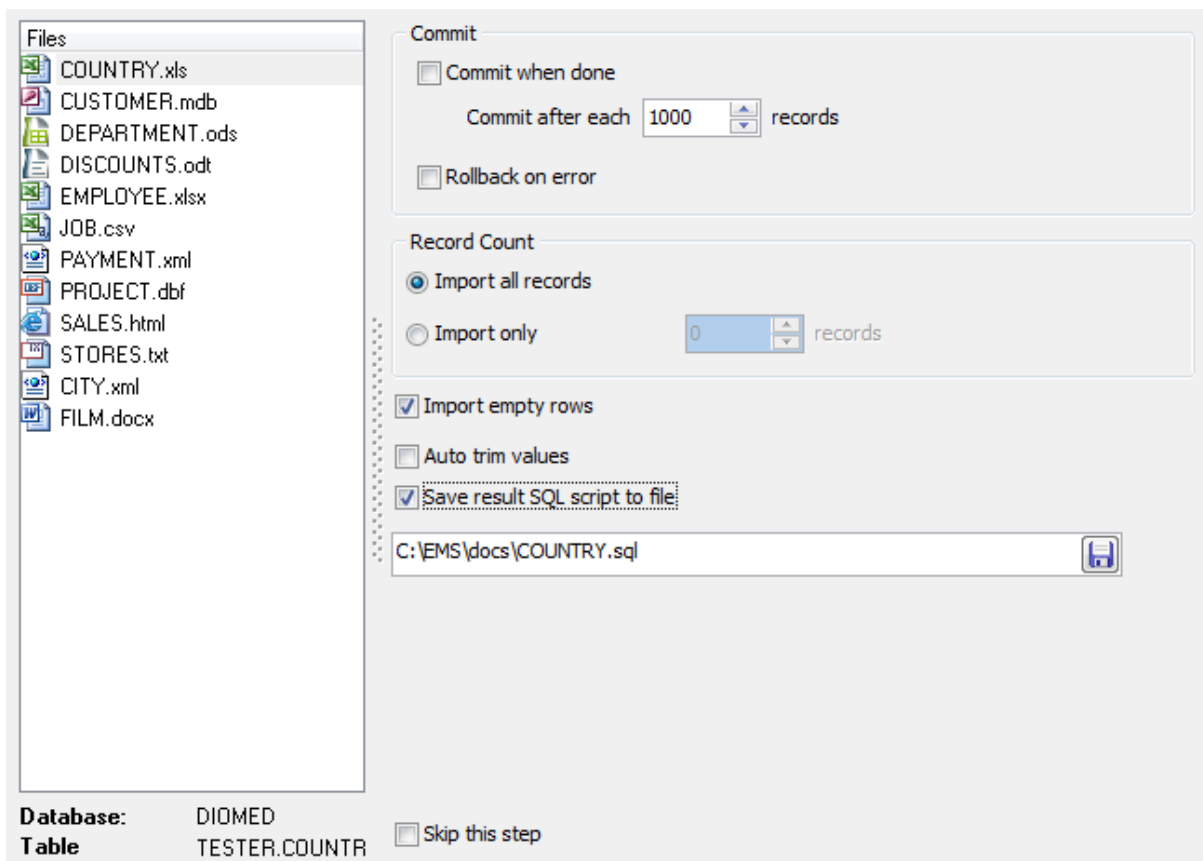
☑ **Auto trim values**
Check this function to remove all leading and trailing white-space characters from all imported records.

☑ **Save Result SQL Script to File**
This option allows you to save the result SQL script of the import operation to an external *.sql* file on your disk drive.

Specify the script file name and select its location using the 🖫 button which calls the **Save As...** dialog.

☑ **Skip this step**
Check this option to skip the current step in the future. To edit the list of skipped steps, use the **Skipped Steps** group available in the General 68 section of the Preferences 68 dialog.

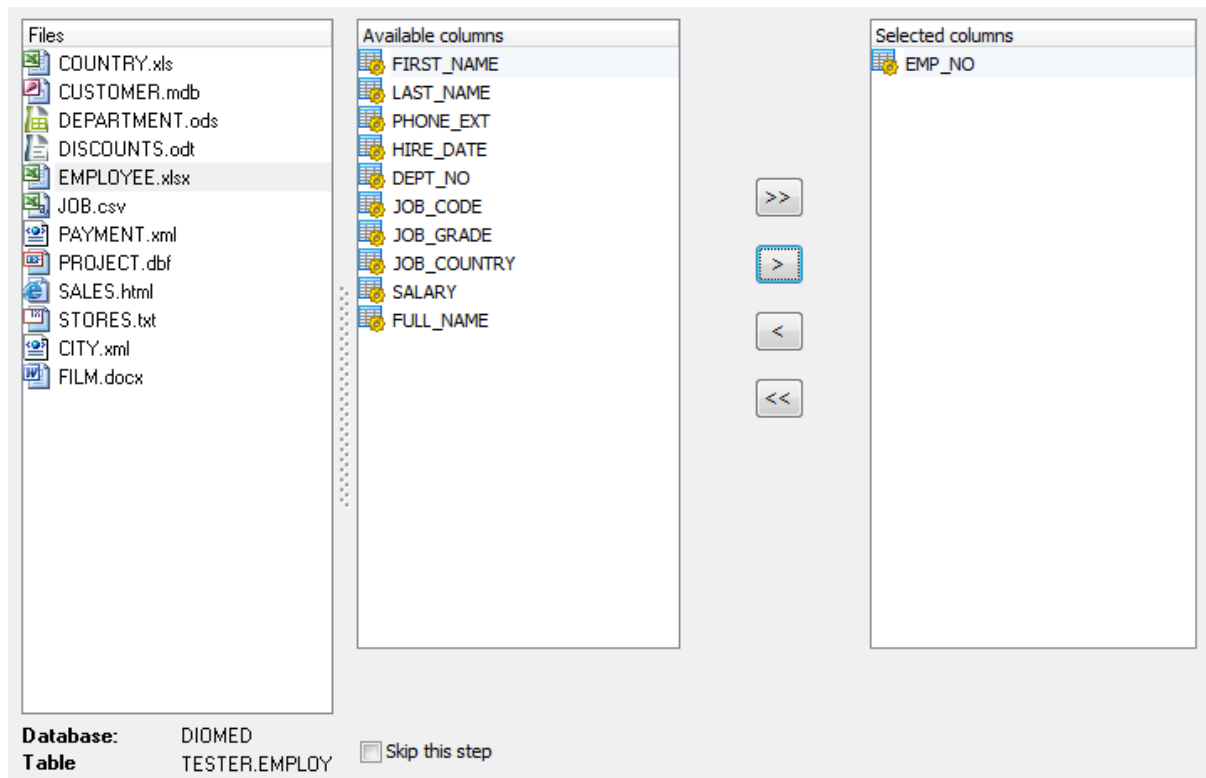When you are done, press the **Next** button to proceed to the <u>next step</u> 63.

2.1.11 Step 10 - Defining scripts

This step of the wizard allows you to define scripts to be **executed before and after import process** for each database.

Select a database to define the script for. Type the text of the script to be executed before the import operation in the **Before Import Script** area, the script to be executed after the import operation in the **After Import Script** area.
Select another database and add scripts for it, if necessary.

You can also save and load *Before Import* and *After Import* scripts using the corresponding **Save Script...** and **Load Script...** buttons.

☑ **Skip this step**
Check this option to skip the current step in the future. To edit the list of skipped steps, use the **Skipped Steps** group available in the <u>General</u> 68 section of the <u>Preferences</u> 68 dialog.

When you are done, press the **Next** button to proceed to the .

2.1.12   Step 11 - Start of data import process

This step of the wizard is intended to inform you that all import options have been set, and you can start the import process. If everything is correct, press the **Import** button to start the process. If you want to change something, you can return to any of the wizard steps using the **Back** button.

☑ **Continue if before import⌐63⌐ script is executed with errors**
Check this option to ignore errors in *Before Script* execution during import.

☑ **Close progress when done**
If this option is selected, the child window indicating the import progress is closed automatically when the import process is completed.

☑ **Parallel execution**
Check this option to perform several files import process simultaneously.
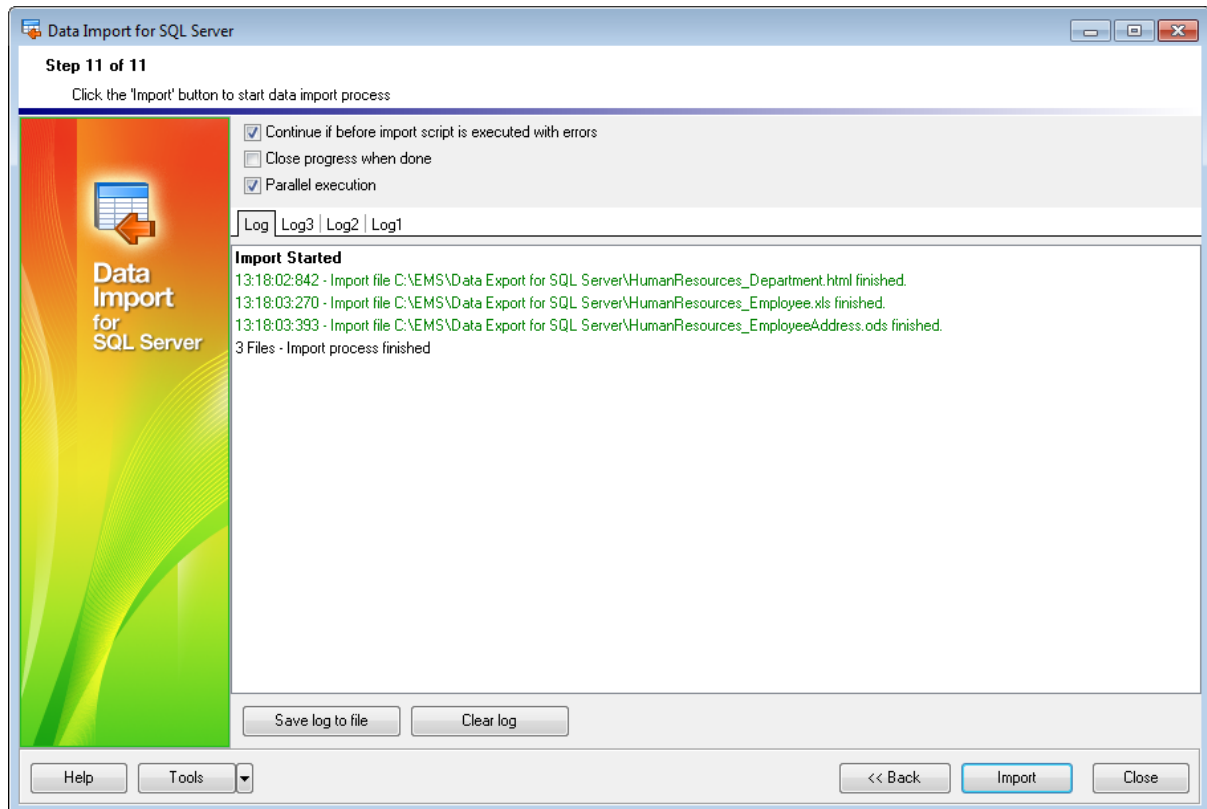
Save log to file
This button calls the **Save file** dialog which allows you to save the on-screen log to a file.

Clear log

Pressing this button clears the on-screen log area, removing all messages.

If necessary, you can [save a template](#) 66 for future use.

Click the **Finish** button to start the import process.

2.2   Using data import configuration files

Data Import for SQL Server allows you to store its configuration settings in external *.itm* template files if you need to perform the data import process repeatedly.

You can load previously saved configuration settings to the application wizard 23 if you need to make some changes before data comparison, or you can run it with the console application 76 for quicker data import.

Data Import templates are saved/loaded within the **Save template options**/**Load template** dialog. To open this dialog, press the **Tools** button and select the **Save template**/**Load template** popup menu item.
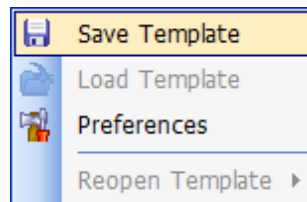
File name

When saving template, specify the template file name and select its location using the 🖫 button which calls the **Save As...** dialog.
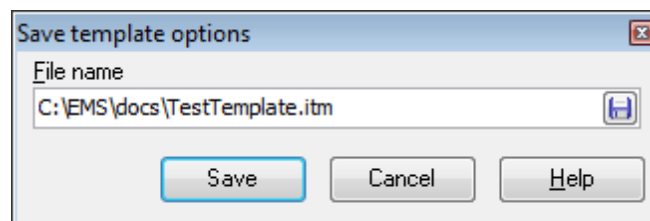
If you need to repeat data importing process with the same or similar settings later, it is reasonable to save all the settings you entered on the Start of data import process 64 step of the Wizard.

Please note that loading a template is only available at the Getting started 24 and the Setting connection properties 25 steps of the Wizard.

If necessary, you can **reopen a template** at any step of the wizard using the corresponding popup menu item of the **Tools** menu.
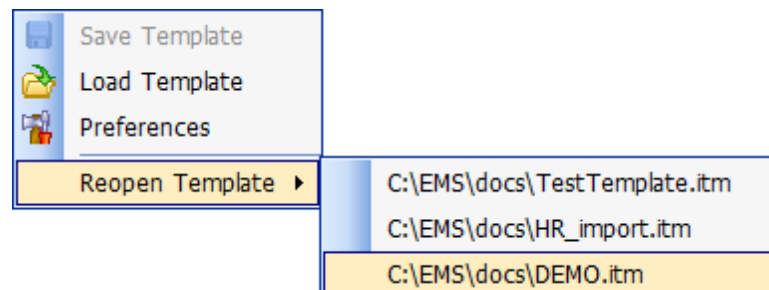
See also:

Working with wizard application 24
Setting program preferences 68

2.3   Setting program preferences

Data Import for SQL Server provides full customization of the program interface by setting various options within the **Preferences** dialog. This chapter is intended to inform you how to use all these options.

Setting general options [68]
These options define general behavior of Data Import for SQL Server.

Selecting program language [70]
This page allows you to select a language to be applied for your copy of Data Import for SQL Server.

Setting data font [71]
Allows you to adjust font settings for the data displayed in the application.

Setting Excel options [72]
Allows you to adjust Excel data representation settings.

Defining interface style [73]
This branch contains several pages with a number of options allowing you to customize the application interface style according to your liking.
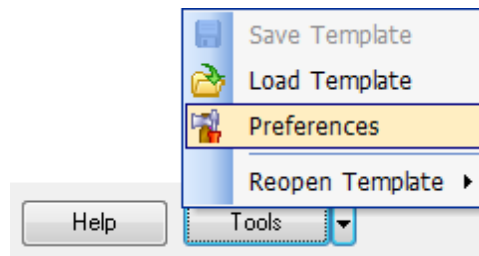
See also:
Working with wizard application [24]
Using data import configuration files
[66]

2.3.1   General

General

☑ **Confirmation on exit**
Enables/disables confirmation upon exiting the program.

☑ **Skip intermediate steps**
This option determines the behavior of the *Next* and *Back* buttons. When this option is disabled, clicking the *Next* button leads to sequential passing through the list of imported objects. When this option is enabled, the steps checked in the **Skipped steps** group will

be skipped.

☑ **Key columns reminder**
Use this option to enable/disable prompts to define key columns.
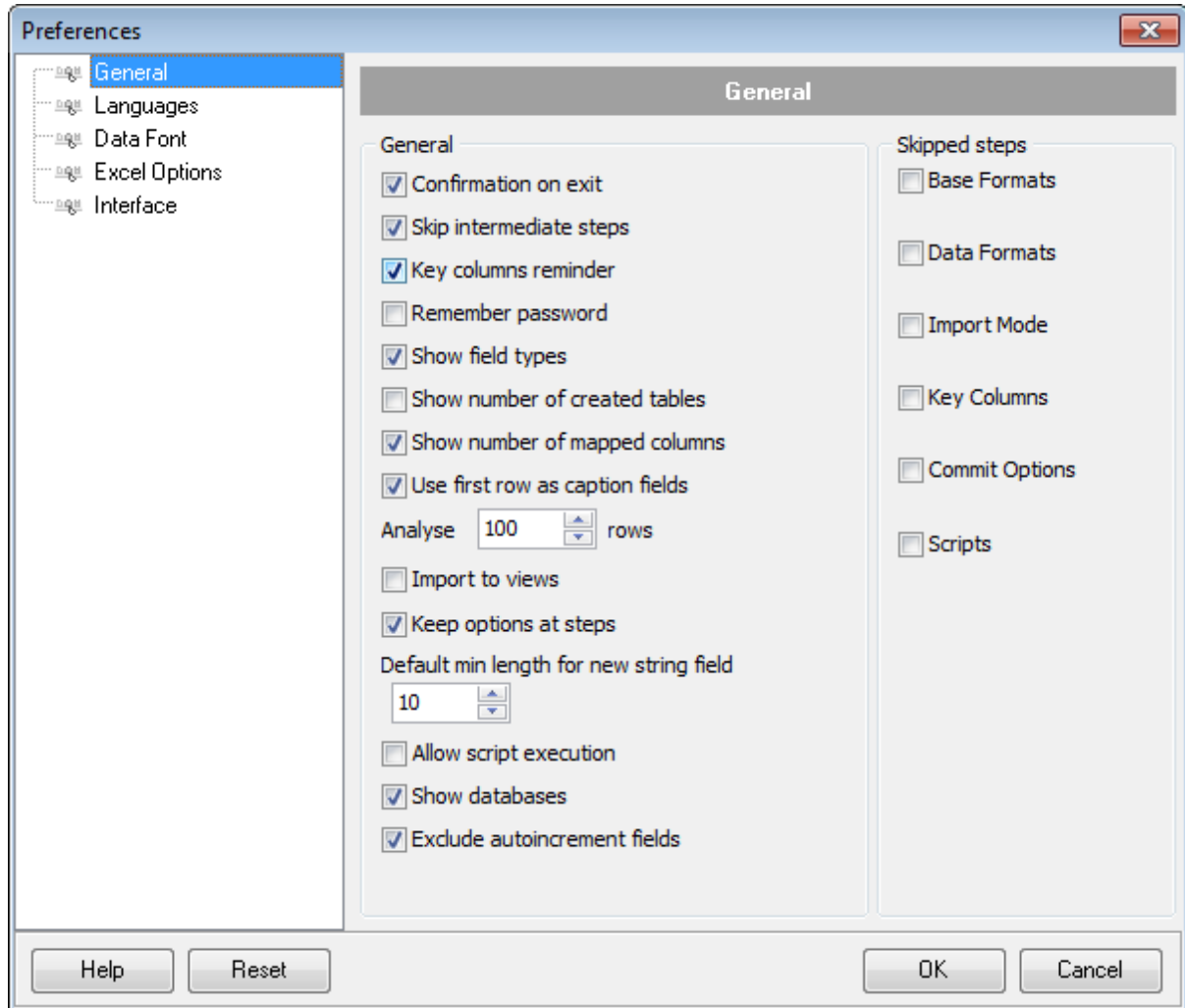
☑ **Remember password**
Setting this option allows you to save passwords used for access to the databases automatically upon closing the application. Please note that checking this option saves the latest password used for connection to the database.

☑ **Show field types**
Hides/displays field type description in the field list.

☑ **Show number of created tables**
Use this option to allow/disallow appearance of the correspondent information window.

☑ **Show number of mapped columns**
Displays the number of mapped fields on proceeding to Step 3 30.

☑ **Use first row as caption fields**
Enable this option to use first rows as captions when mapping fields.

Analyse *N* rows
Define the amount of rows used for source data analysis before table creation. This analysis allows the program to define the destination table field type.

☑ **Import to Views**
By setting this option on you can import data to views - they will appear in the list of available tables at the Step 2 27 of Wizard Application 23.

☑ **Keep options at steps**
If this option is checked all your import settings will remain the same the next time the program is started.

☑ **Allow script execution**
If this option is checked then the script entered on the Step 5 54 is always executed.

☑ **Show databases**
This option enables/disables the 'Select Database' drop-down list in 'Select Table' dialog at Step 2 27 of the wizard. If you check this option, you can select the database from the list, otherwise you must type the database name manually.

☑ **Exclude autoincrement fields**
If this option is checked then autoincrement fields will not be mapped on auto mapping 30.

Skipped Steps
Use this option group to define the Wizard application 23 steps to be skipped.

See also:
Selecting program language
70
Setting data font 71
Setting Excel options 72
Defining interface style 73

2.3.2 Languages

The **Languages** page is provided for managing Data Import localization files.

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.

The table lists all the languages available for localization and the corresponding *.lng* files.
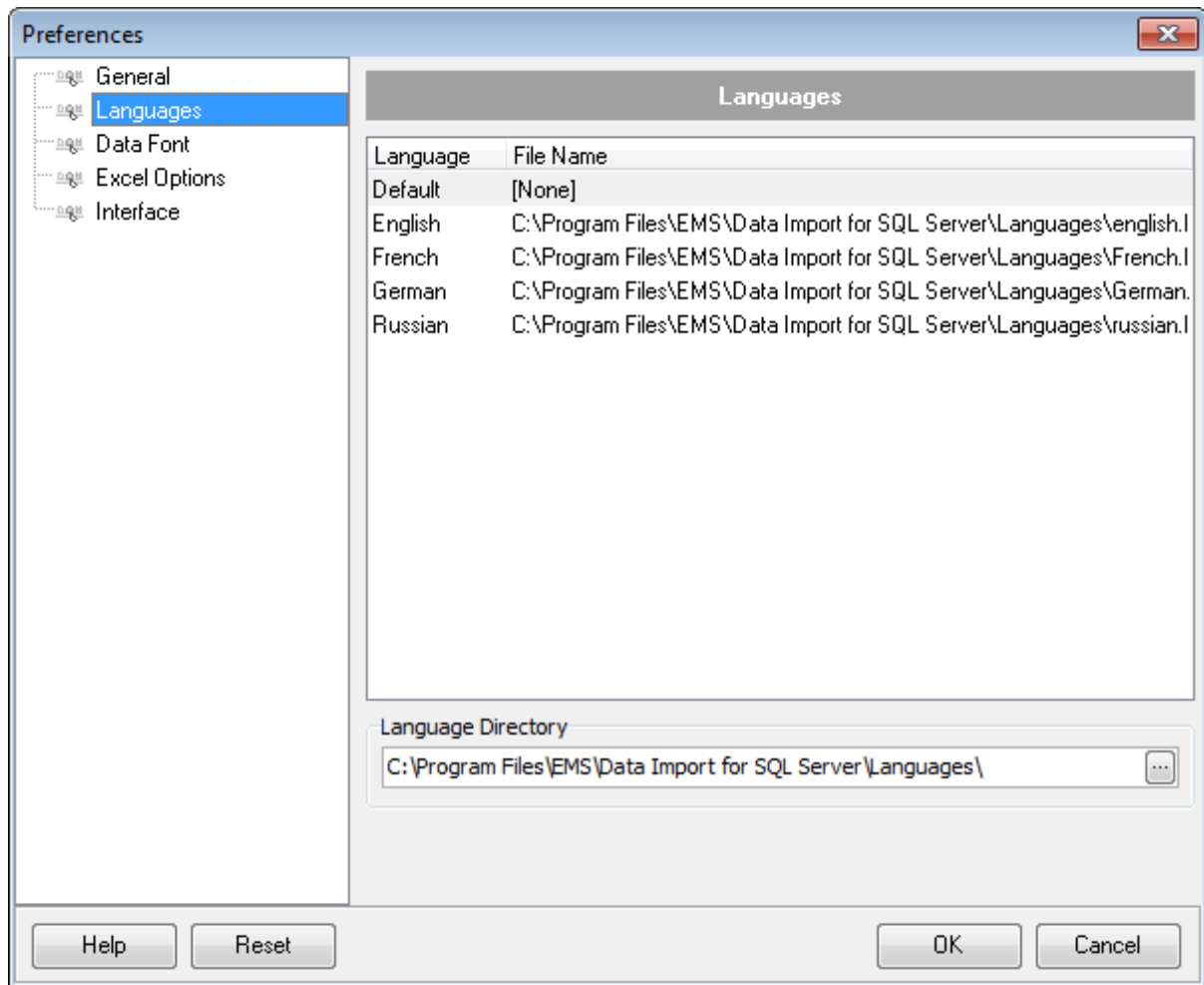
Language Directory
Use the ellipsis ⬚ button to specify the directory where the *.lng* files are stored by default.

See also:

2.3.3   Data Font

The **Data Fonts** page is provided for setting up the fonts for displaying data in Data Import for SQL Server.

Set the appropriate **font** and the preferable **charset** using the corresponding drop-down lists.
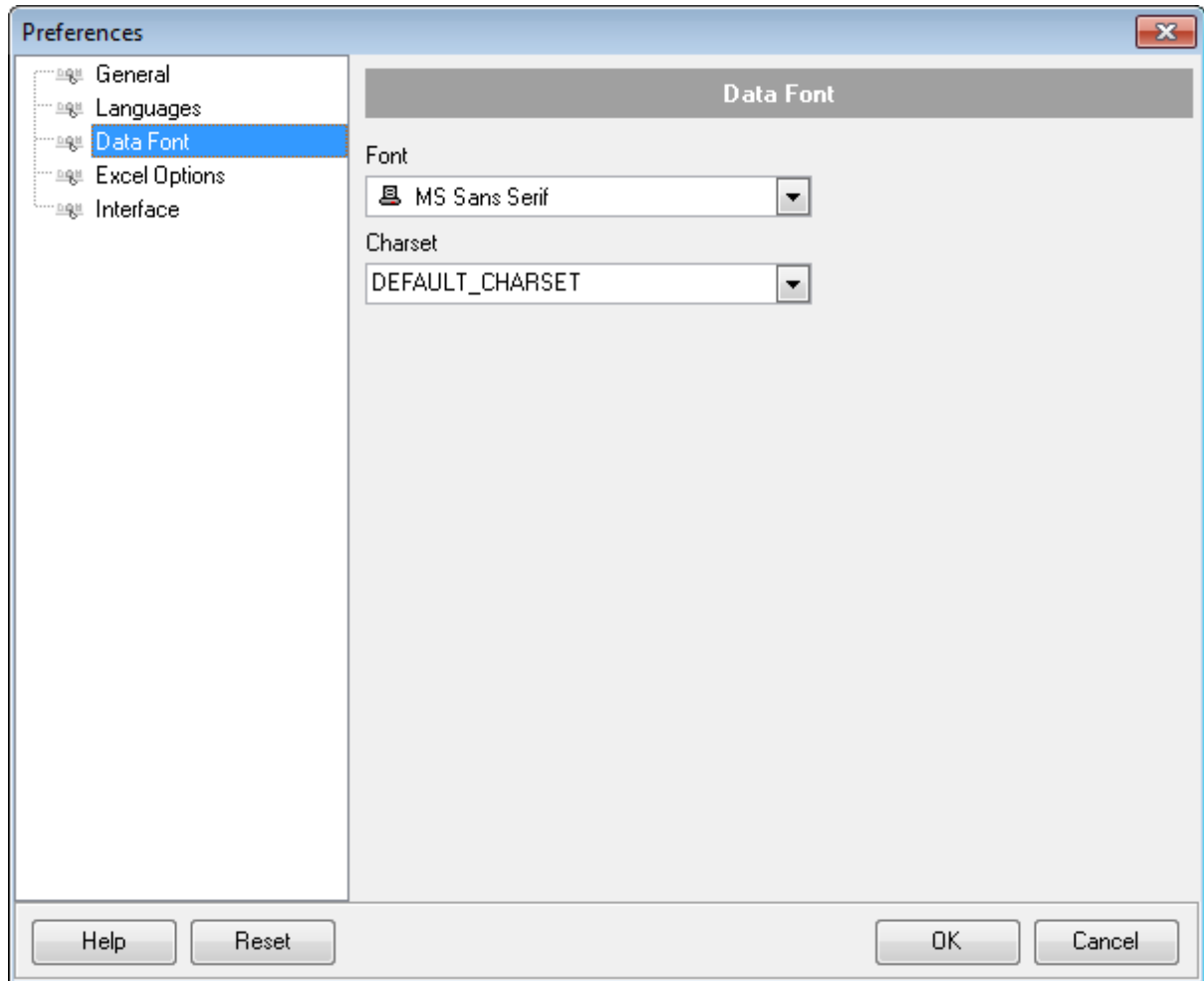
Note that neither font nor character set changes can be applied to TXT 42 files.

See also:

2.3.4   Excel Options

The **Excel Options** page allows you to set the **Max Col Width** parameter which defines the maximum column width for Excel data grid.

See also:

2.3.5 Interface

This page allows you to customize the application interface style to your liking.

Use the **Scheme name** drop-down list to select an interface scheme according to your liking: *Classic*, *Office XP* style, *Windows XP* native style, etc.
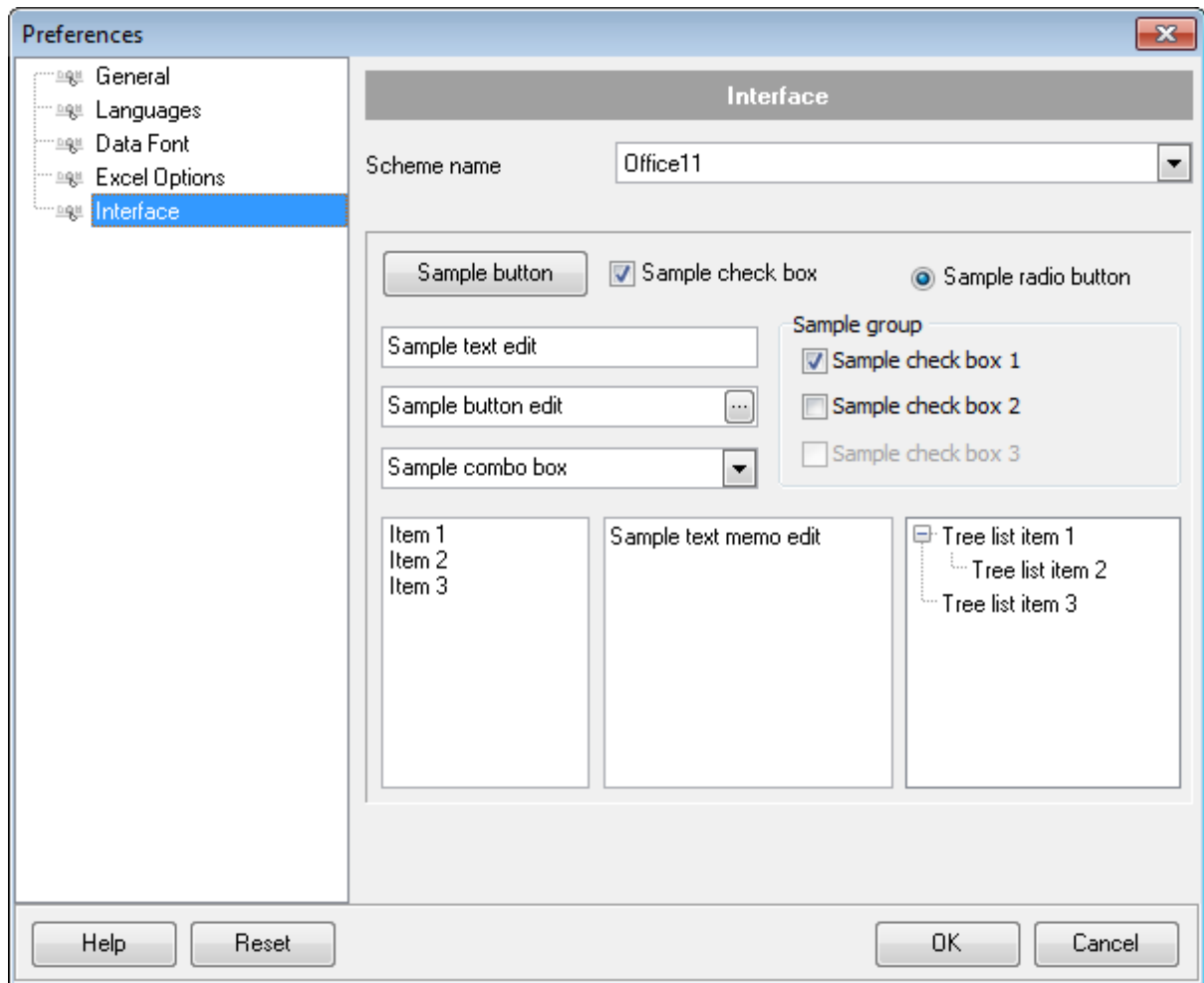
For your convenience the previews illustrating the changes are displayed in the **Sample Group** area.

See also:

Part III

3 Console application

Additionally to **the GUI version** which is implemented in the form of a wizard application, the installation package of Data Import for SQL Server includes **the console version** which is intended for being run from Windows command line with a template file name used as the execution parameter.

```
C:\Program Files\EMS\Data Import for SQL Server>MSImportC.exe_
```

Data Import for SQL Server command line utility is intended for quick and powerful data import to SQL Server tables.

See also:

3.1   Working with console application

All the import options are set in **template** (**.itm*) files. A template can be also used in the **Console version** of Data Import for SQL Server.

To create a template file, follow the instructions below:
- start Data Import Application wizard 23 ;
- set all the required options in all steps of the wizard;
- test the import process at the last step;
- save all import options in the template 66 .

Console application also generates the **.log* file that contains information about the import operation completion.

The easiest way to start Data Import for SQL Server console application is to double-click the generated **.itm* template. The other way is to enter the command line and type the appropriate command.

Usage:

<path to Data Import for SQL Server console application>\MsImportC.exe *TemplateFile [-datafile=<source file name>] [-B]*

datafile
Indicates the source file for data import.

TemplateFile
Stands for the **.itm* template file to be used as the console version execution parameter

[-B]
Use this parameter in the command line to run the console version of Data Import for SQL Server in the background mode

Example:

"C:\Program Files\EMS\Data Import for SQL Server\MsImportC.exe" "C:\EMS\DataImport\Template1.itm" -B

Note: The following exit codes can be returned by Data Import for SQL Server to the operating system after performing the latest task:
0 - successful completion;
1 - error(s) occurred during task performing;
2 - fatal error occurred. The task was not performed.

See also:
Working with wizard application 24
Configuration file format 78

3.2   Configuration file format

The configuration file is divided into several sections.

[Connection]
This section contains parameters for connecting to the server: *Host*, *Login*, *Password* (encrypted). These parameters are obligatory.

[PREFERENCES]
This section contains the general settings of the utility.

ImportToViews
1 - enabled, *0* - disabled

The following sections are unique for each imported file and contain the appropriate postfix (*FILE_00*, *FILE_01*, *FILE_02*, etc.)
In these sections the import options for each particular file are stored.

[FILE_00]
In the main file section the following parameters are used:
Database (destination database name), *Table* (destination table), *FileName* (the source file name)

ImportType
Indicates the format of the source file: *0* stands for MS Excel, *1* for MS Access, *2* - DBF, *3* - TXT, *4* - CSV, *5* - XML, *6* - MS Excel 2007, *7* - MS Word 2007, *8* - HTML, *9* - XML Document, *10* - ODS, *11* - ODT

The next section is specific to the file format and is named correspondingly, e.g. *[FILE_00_XLS]*. In this section the following specific file type options are set:

[FILE_00_XLS]
SkipRows, *SkipCols* - the number of rows and columns in the source file to be skipped on import.

[FILE_00_Access]
SourceType
This option indicates the Access source type - Table or query, *0* stands for table, and *1* for query.
TableName and *query* - these options set the source Access table name to import from if source type is *0* and the source query text if the source type is *1*.

[FILE_00_TXT]
SkipRows - the number of rows in the source file to be skipped on import.

[FILE_00_CSV]
SkipRows - the number of rows in the source file to be skipped on import.
CSVDelimiter, *CSVRightQuot*, *CSVLeftQuot* - these options are used on reading the source CSV file, they store options for delimiting columns and the quotation marks.

The next section is **[FILE_XX_MAP]**.
It is also specific to the file type, but is obligatory for all file types. In this section correspondence between the source file columns and the destination table fields is set.

If the source file is an *Excel* file, then the map is set in the following format:
<DB_FieldName>=CellIndex.
Use semicolon to separate multiple cells. You can define an Excel row or column as a range of cells, e.g.
Field1=A1;A4;A6-A9;B1-F1.

If the source file is an *Access* or *DBF* file, then the mapping is set in the following format:
<DB_FieldName>=<Source_Field_Name>.

If the source file is a *TXT* file, the mapping looks in the following way:
<DB_FieldName>=<Position>;<Size>.
This means that you should provide the starting position and the size of the source file column for each destination field.

To set mapping for a *CSV* file, provide a column number for each destination field, e.g.
Field1=1
Field2=4
etc.

The properties that are set in sections **[FILE_XX_BASE_FORMATS]** and **[FILE_XX_IMPORT_OPTIONS]** correspond to those set on the 'Base Formats' and the 'Options' tabs.

The values that stand for *BOOLEAN TRUE* and *BOOLEAN FALSE* values are set in sections **[FILE_XX_BOOLEAN_TRUE]** and **[FILE_XX_BOOLEAN_FALSE]**.

The *NULL* values are set in section **[FILE_XX_NULL_VALUES]**

Data formats that are set for each destination field separately are stored in sections named in the following way:
[FILE_XX_DATA_FORMATS_<FIELD_NAME>],
e.g. *FILE_00_DATA_FORMATS_FIELD1*.
The properties within these sections also correspond to the properties set on the 'Data Formats' tab of the Settings data formats 54 step for each field.

[FILE_00_IMPORT_OPTIONS]
This section contains parameters that define import options specified on Steps 6-9 of the wizard.

CommitAfterDone
1 - enabled, *0* - disabled

CommitRecCount
The number of records in each block to be supplemented with the COMMIT statement.

ImportRecCount
1 - enabled, *0* - disabled

ImportAllRows
1 - enabled, *0* - disabled

SaveResultSQL

1 - enabled, *0* - disabled

ResultSQLFile
The path to the result *.sql* file.

ImportKind
0 - Universal Mode, *1* - Native Mode

TruncateLongString
1 - enabled, *0* - disabled

RollbackIfError
1 - enabled, *0* - disabled

ImportMode
0 - Insert All, *1* - Insert New, *2* - Update, *3* - Update or Insert, *4* - Delete, *5* - Delete or Insert

ImportAddType
1 - enabled, *0* - disabled

AllowDuplicates
1 - enabled, *0* - disabled

NativeModeKind
0 - Single Commands, *1* - Batch insert

[#General#]
This section stores information about the product name and its major version.

See also:

Working with console application 77

Part IV

4   Appendix

4.1   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.

◉ **HTML**

Hyper Text Markup Language file format (*.html*, *.htm*), complete compatibility with HTML 4.0 specification.

◉ **Text file**

Plain text file format (*.txt*).

◉ **CSV file**

Comma-Separated Value file format (*.csv*).

Note: all the text formats including *Text file*, *CSV* are usually used as working or interchange formats.

◉ **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.

◉ **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.
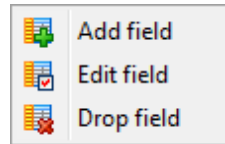
◉ **ODF Text**

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.
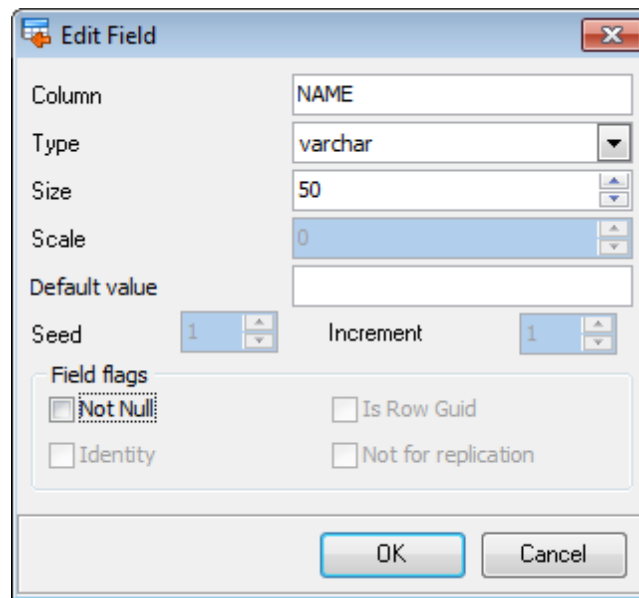
4.2 Add/Edit field

When you import data with destination table creation, you can manage this table's fields.

Use the context menu to Add/Edit/Drop a field.

Selecting **Add field** or **Edit field** context menu item opens the field editor that allows you to view/edit the properties of the destination table fields.

Use the **Column** edit box to set the field name. Note that the name of the field must be unique among all the field names in the table.

The **Type** tab defines the type of the field data.

Size
Defines the size of the field value.

Scale
For *numeric* and *decimal* types you need to define the number of decimal to the right of the decimal point.

Default value
Define this option value if you need inserted records to get specified value.

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.

☑ **Not NULL**
Check this option to prevent the entry of NULL or unknown values in column.

☑ **Is Row Guid**
This option specifies that the *ROWGUIDCOL* property is added to the column. *ROWGUIDCOL* indicates that the column is a row GUID column.

☑ **Identity**
This option specifies that the new column is an identity column.
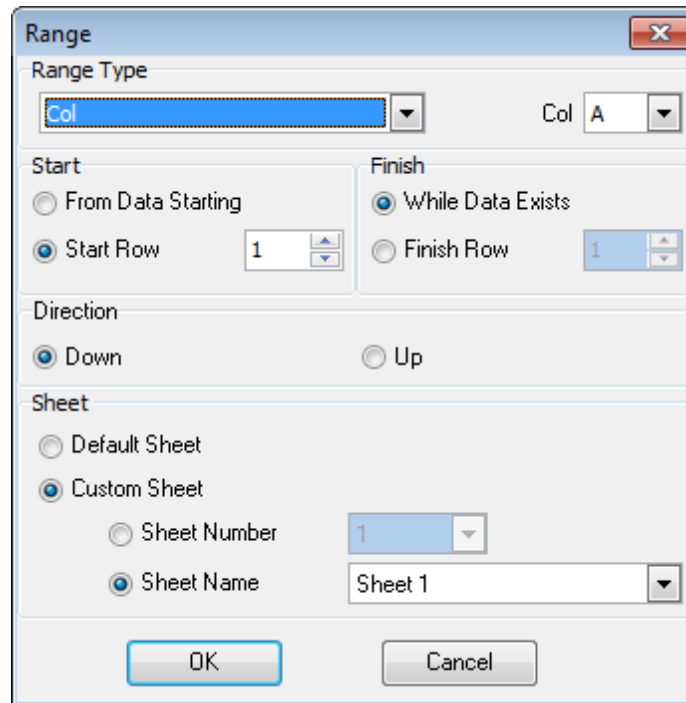
☑ **Not for replication**
This option allows you to specify that the Identity property should not be enabled when a replication agent modifies the table that contains the identity column.

4.3   Add/Edit Range

To set a range of data [85] to be imported from the file, use the **Add range** [+] button.

To remove a range, use the **Delete range** [▬] button.

To edit an existing range, double-click it in the **Ranges** list. The **Range** dialog allows you to set a number of options for the range being edited.

Select **Range Type** from the corresponding drop-down list. Then specify the *column*, *row* or *cell* number whose range is to be defined.
Start and **Finish** section allows you to set the range of the values to be imported from the specified row or column.
Use the ◉ **From Data Starting** and ◉ **While Data Exist** options to automatically define the range of the values.
If neither start row/column nor finish row/column was specified you need to set range direction: ◉ **Down** or ◉ **Up** for column range and ◉ **Right** or ◉ **Left** for row range.
Use **Sheet** section to define the location of the row/column for which the row was specified.
You can select either ◉ **Default sheet**, or ◉ **Custom Sheet**.
For custom sheet you need to select **Sheet Number** or **Sheet Name** from the appropriate drop-down list.

Note: If Cell was set as Range Type, the only thing you can specify at this dialog is its number.

Credits

Software Developers:

Alex Paclin

Alexey Butalov

Michael Kuzevanov

Alexey Saybel

Technical Writers:

Semyon Slobodenyuk

Dmitry Doni

Olga Ryabova

Cover Designer:

Tatyana Makurova

Translators:

Anna Shulkina

Serge Fominikh

Team Coordinators:

Alexey Butalov

Alexander Chelyadin

Roman Tkachenko