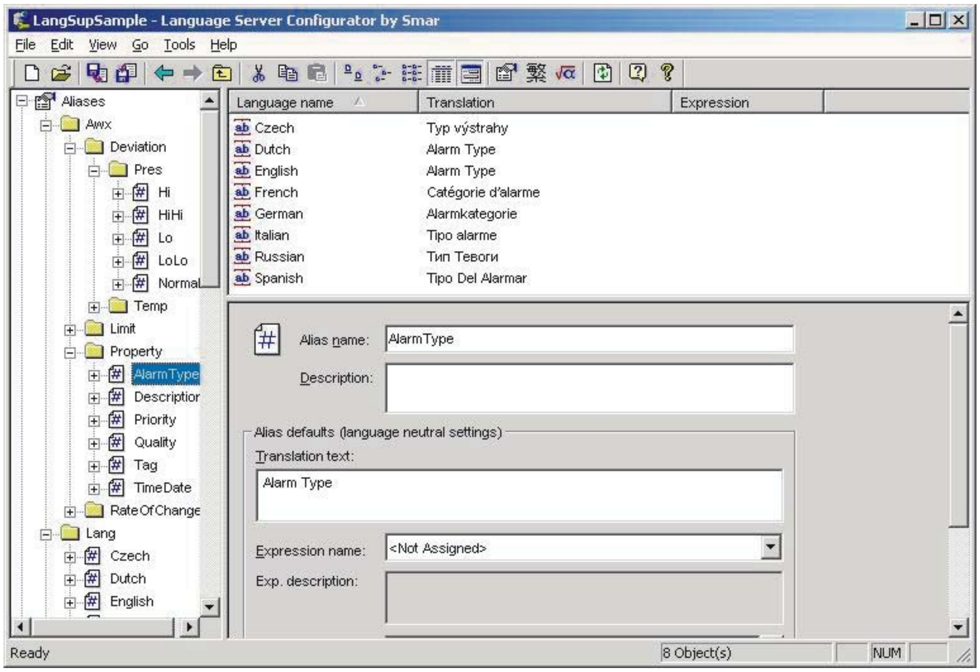
USER’S GUIDE

smar - Language Switching

FIRST IN FIELDBUS

Language Switching



Language name	Translation	Expression
Czech	Typ výstrahy	
Dutch	Alarm Type	
English	Alarm Type	
French	Catégorie d’alarme	
German	Alarmkategorie	
Italian	Tipo alarme	
Russian	Тип Тевоги	
Spanish	Tipo Del Alarmar	

Alias name: AlarmType

Description:

Alias defaults (language neutral settings)

Translation text: Alarm Type

Expression name: <Not Assigned>

Exp. description:

JUN / 04

Language Switching

VERSION 7.1

FOUNDATION

P V I E W L S W M E

smar

www.smar.com

Specifications and information are subject to change without notice.
Up-to-date address information is available on our website.

web: www.smar.com/contactus.asp

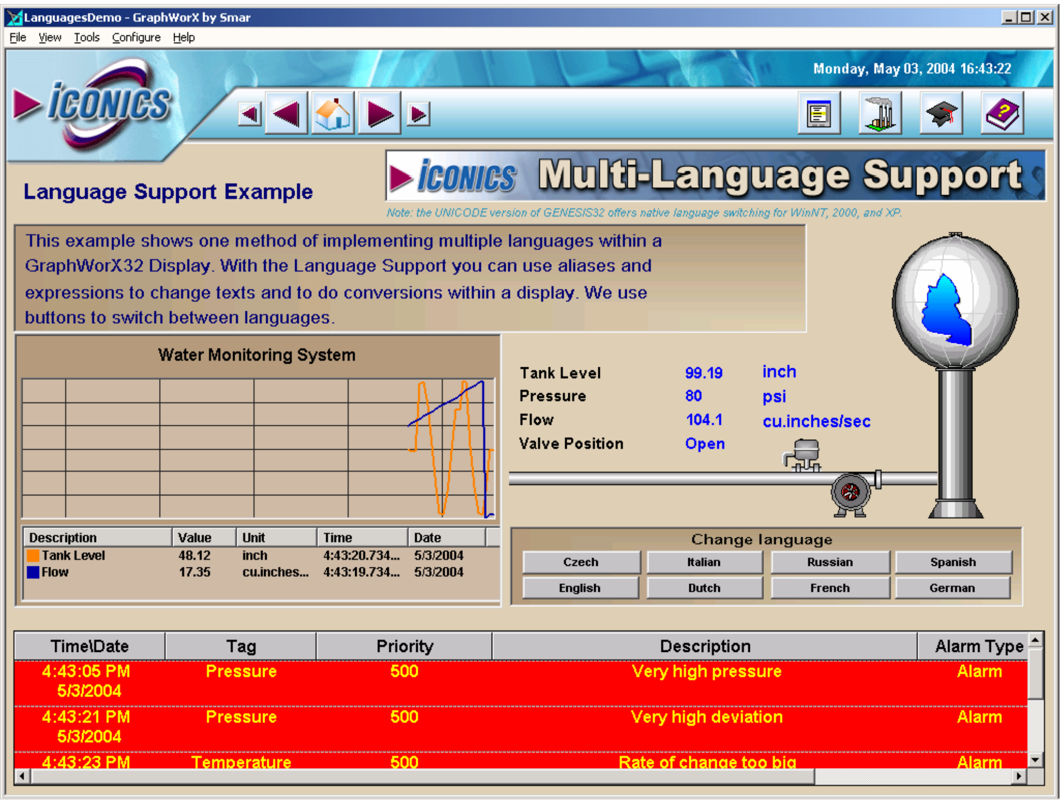
Index

Introduction to Language-Aliasing Support

ProcessView includes language-aliasing support provided by the Language Configurator. The Unicode version of the ProcessView supports language aliasing with automatic range and value scaling, as well as unit conversions, formatting, and font selection.

Sample Language Configuration

The Unicode version of ProcessView installs a sample Language Server configuration under the "Languages" folder of the ProcessView product installation "tree." In addition, a sample display "languagesDemo.gdf," which includes a sample Viewer configuration, is provided under the "ProcViewDemo" folder. The figure below shows this sample during runtime mode. Notice that you can switch between languages in this display by simply clicking the buttons in the **Change Language** field.



Note: the UNICODE version of GENESIS32 offers native language switching for WinNT, 2000, and XP.

This example shows one method of implementing multiple languages within a GraphWorX32 Display. With the Language Support you can use aliases and expressions to change texts and to do conversions within a display. We use buttons to switch between languages.

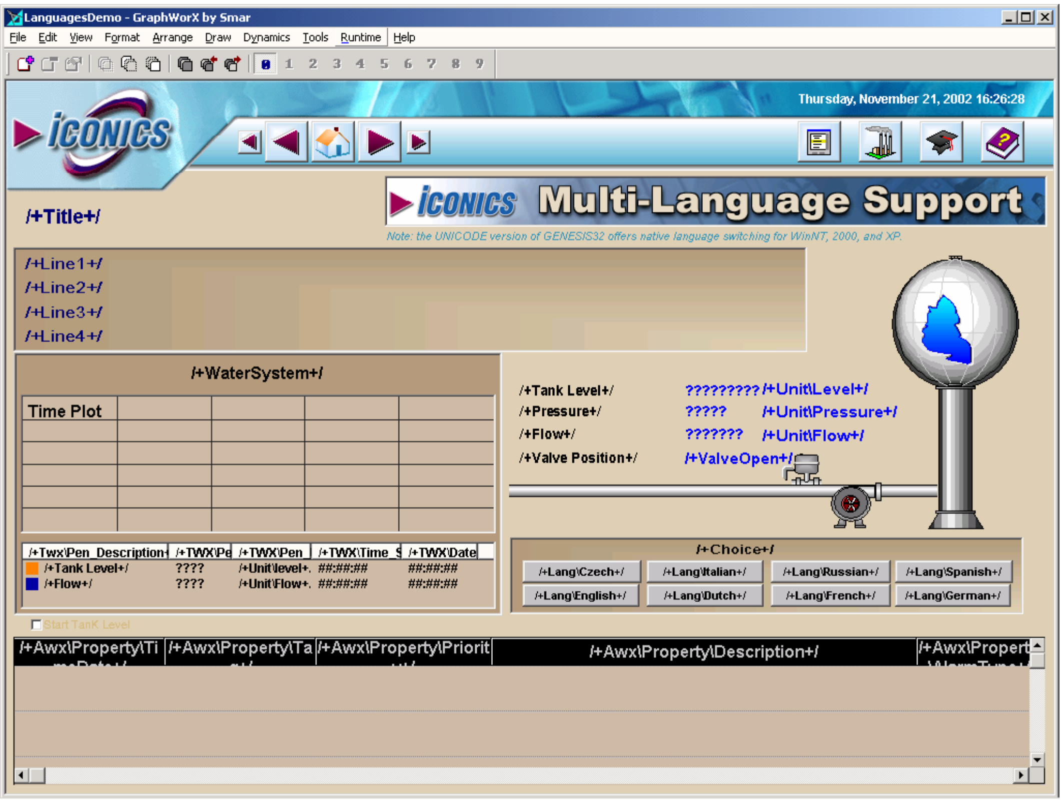
Description	Value	Unit	Time	Date
🟧 Tank Level	48.12	inch	4:43:20.734...	5/3/2004
🟦 Flow	17.35	cu.inches...	4:43:19.734...	5/3/2004

Tank Level	99.19	inch
Pressure	80	psi
Flow	104.1	cu.inches/sec
Valve Position	Open	

Time\Date	Tag	Priority	Description	Alarm Type
4:43:05 PM 5/3/2004	Pressure	500	Very high pressure	Alarm
4:43:21 PM 5/3/2004	Pressure	500	Very high deviation	Alarm
4:43:23 PM	Temperature	500	Rate of change too big	Alarm

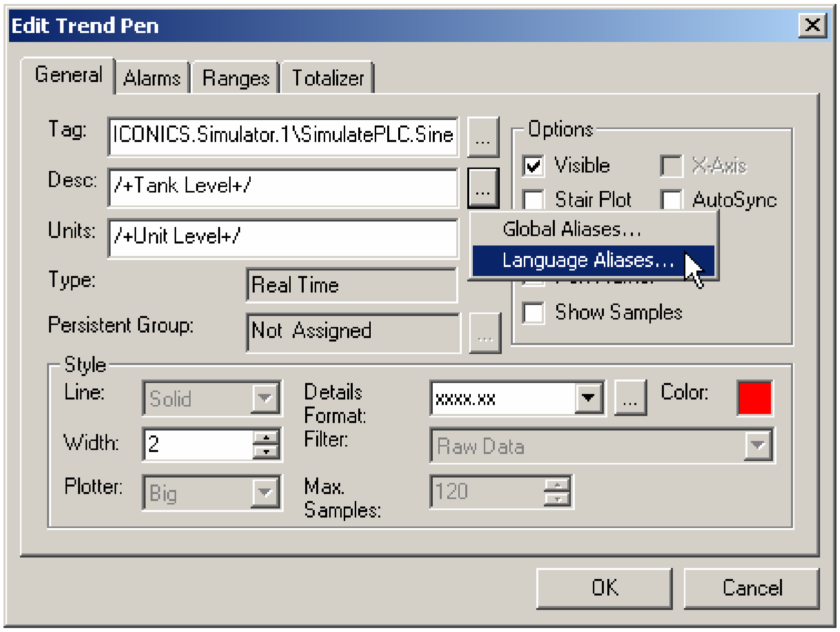
Language-Aliasing Demo in Runtime Mode

The figure below shows this same language-aliasing demo in configuration mode. This display incorporates ActiveX objects from GraphWorX, TrendWorX, and AlarmWorX.


LanguagesDemo - GraphWorX by Smar
File Edit View Format Arrange Draw Dynamics Tools Runtime Help
Thursday, November 21, 2002 16:26:28
ICONICS
ICONICS Multi-Language Support
Note: the UNICODE version of GENESIS32 offers native language switching for WinNT, 2000, and XP.
/+Title+/
/+Line1+/
/+Line2+/
/+Line3+/
/+Line4+/
/+WaterSystem+/
Time Plot
/+Twx\Pen_Description
/+TWX\Pe
/+TWX\Pen
/+TWX\Time_S
/+TWX\Date
/+Tank Level+/ ???? /+Unit\level+. ##:##:## ##:##:##
/+Flow+/ ???? /+Unit\Flow+. ##:##:## ##:##:##
/+Tank Level+/ ????????? /+Unit\Level+/
/+Pressure+/ ????? /+Unit\Pressure+/
/+Flow+/ ??????? /+Unit\Flow+/
/+Valve Position+/ /+ValveOpen+/
/+Choice+/
/+Lang\Czech+/
/+Lang\Italian+/
/+Lang\Russian+/
/+Lang\Spanish+/
/+Lang\English+/
/+Lang\Dutch+/
/+Lang\French+/
/+Lang\German+/
Start Tank Level
/+Awx\Property\Ti
/+Awx\Property\Ta
/+Awx\Property\Priorit
/+Awx\Property\Description+/
/+Awx\Propert

Language-Aliasing Demo in Configuration Mode

For example, if you click on the "Water System" time plot trend in the language-aliasing configuration shown above, you can view the properties of the trend in the TrendWorX Viewer Active X. The **Edit Trend Pen** dialog box, shown below, displays the language-aliasing configuration of the "Tank Level" pen in the time plot trend for the water-monitoring system.


Edit Trend Pen

General | Alarms | Ranges | Totalizer

Tag: ICONICS.Simulator.1\SimulatePLC.Sine

Desc: /+Tank Level+/

Units: /+Unit Level+/

Type: Real Time

Persistent Group: Not Assigned

Options: ☑ Visible □ X-Axis □ Stair Plot □ AutoSync □ Show Samples

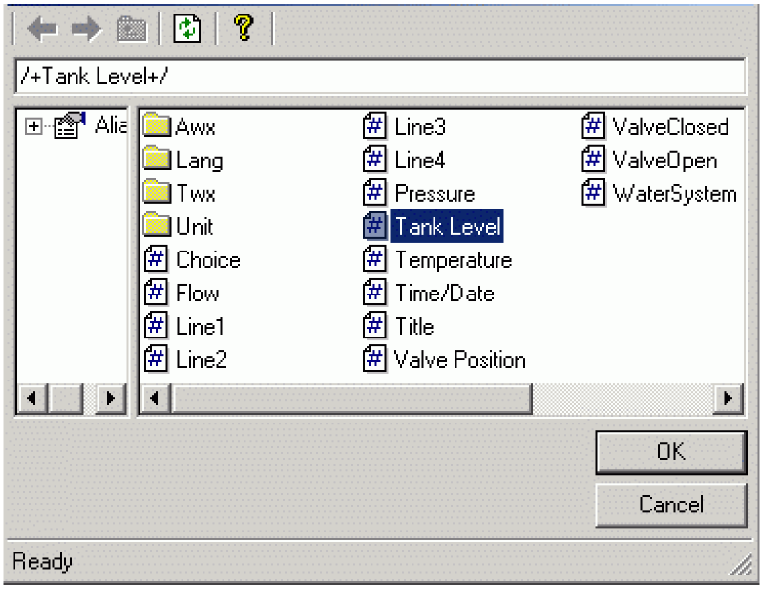
Style — Line: Solid; Width: 2; Plotter: Big; Details Format: xxxx.xx; Filter: Raw Data; Max. Samples: 120; Color: 🟥

Language Aliasing Pen Configuration

When specifying a language alias, you can also select a language alias from the Language Alias Browser, which includes all language aliases in the language database. This eliminates the need to manually type in the alias name. Click the **...** button and select **Language Aliases,** as shown in the figure above.

The **Language Alias Browser** opens, as shown in the figure below. The browser includes all languages aliases in the language database. All language aliases that are configured in the

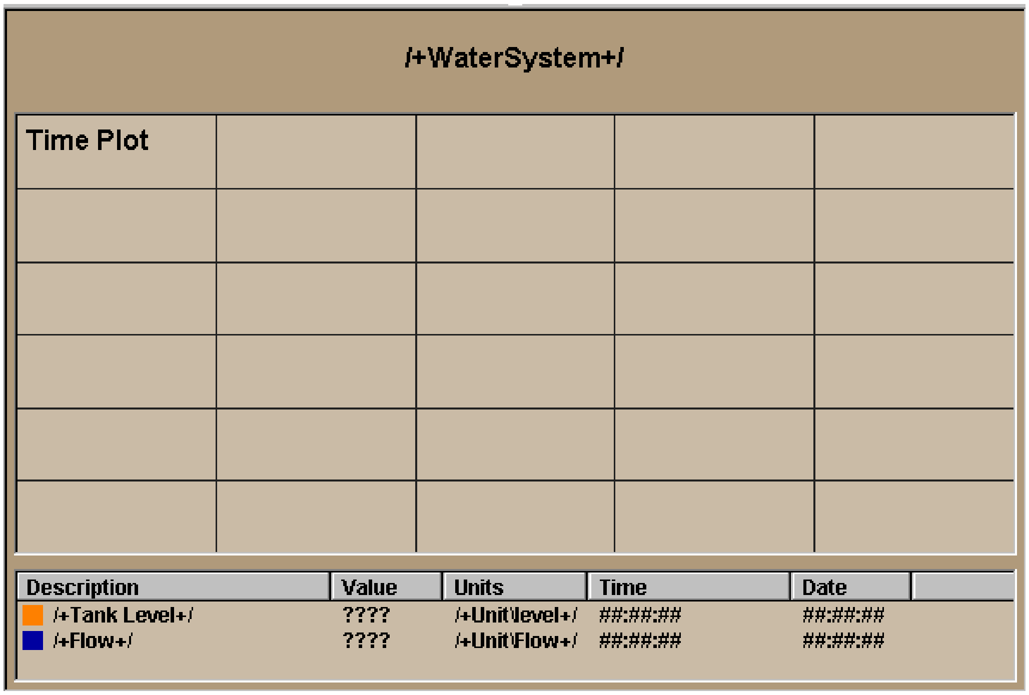
Language Configurator are conveniently available to choose from inside the browser. The tree control of the Language Configurator is mimicked in the tree control of the Language Alias Browser. Select a language alias by double-clicking the alias name. The alias name appears at the top of the browser, which automatically adds the /+ and +/ delimiters to the alias name. Click the **OK** button.



/+Tank Level+/

Selecting an Alias From the Language Alias Browser

Delimiters. For the example above, the trend display appears as shown in the figure below. Note that the strings within a "/+" and "+/" delimiter pair define a language-aliased string. During animation (runtime) mode, the TrendWorX Viewer will interface to the Language Server, and it will try to resolve the language-aliased strings. If successful, it will display the translated strings in the trend screen. In addition, if the **Units** language alias string has an equation defined in the Language Configurator, it will autoscale the pen values based on the scaling equation supplied by the Language Server.



Description	Value	Units	Time	Date
🟧 /+Tank Level+/	????	/+Unit\level+/	##:##:##	##:##:##
🟦 /+Flow+/	????	/+Unit\Flow+/	##:##:##	##:##:##

Language-Aliased Trend Display

System Requirements

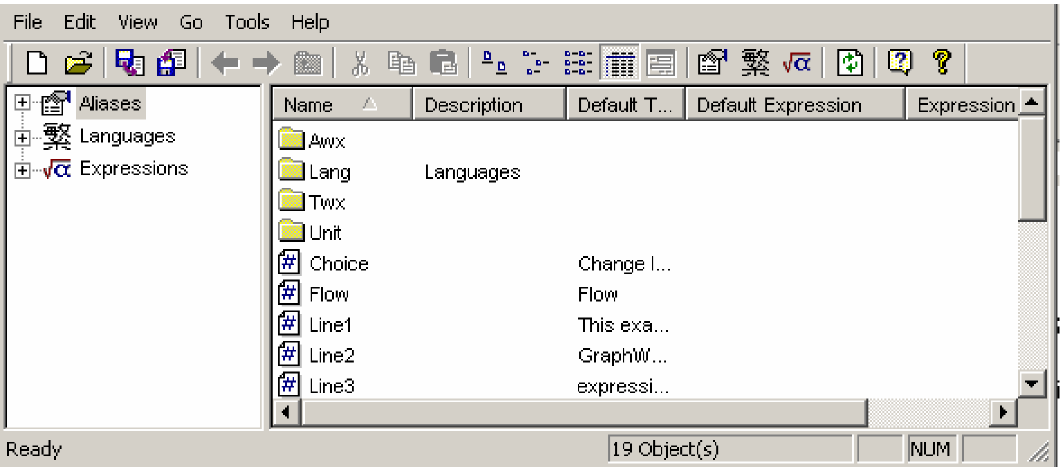
The Unicode version language aliasing is independent of the resource .dll selection, which relates mostly to the text in the dialog boxes and other user interface elements. The Unicode version of PROCESSVIEW is available only for Windows 2000, Windows XP, Windows NT, and .NET Server.

Starting the Smar Language Configurator

To start the Language Aliasing Configurator:

1. From the Windows **Start** menu, select **Programs > Smar ProcessView > Languages > Language Configurator.**

2. This opens the Language Aliasing Configurator, as shown in the figure below. The screen consists of a split window with a tree control view in the left-hand pane and a configuration view in the right-hand pane. The left-hand pane contains a "tree view" that lists the **Aliases, Languages,** and **Expressions** that are configured for language aliasing. Notice that the Language Configurator provides several aliases, languages, and expressions by default. These default settings are configured for the language-aliasing demo. The right-hand pane displays information about specific configurations and enables you to configure new language-aliasing settings. The Configurator provides a standard format for the configuration database, as well as a sample (default) Language Aliasing configuration project. The Configurator also includes a toolbar and menus with many command functions.



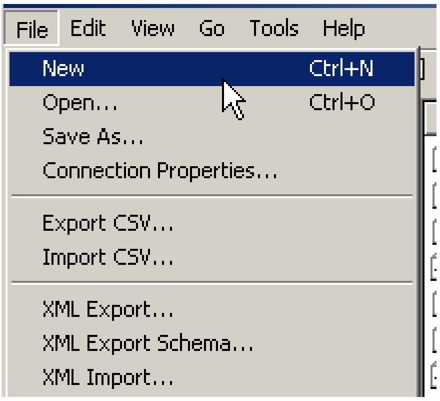
Name	Description	Default T...	Default Expression	Expression
Awx				
Lang	Languages			
Twx				
Unit				
Choice		Change l...		
Flow		Flow		
Line1		This exa...		
Line2		GraphW...		
Line3		expressi...		

Language Configurator Screen

Creating Configuration Databases

The Configurator provides a Configuration Database Wizard for creating new Microsoft Access and SQL Server configuration databases. To create a new configuration database in the Configurator:

1. Select **New** from the **File** menu, as shown in the figure below**.**



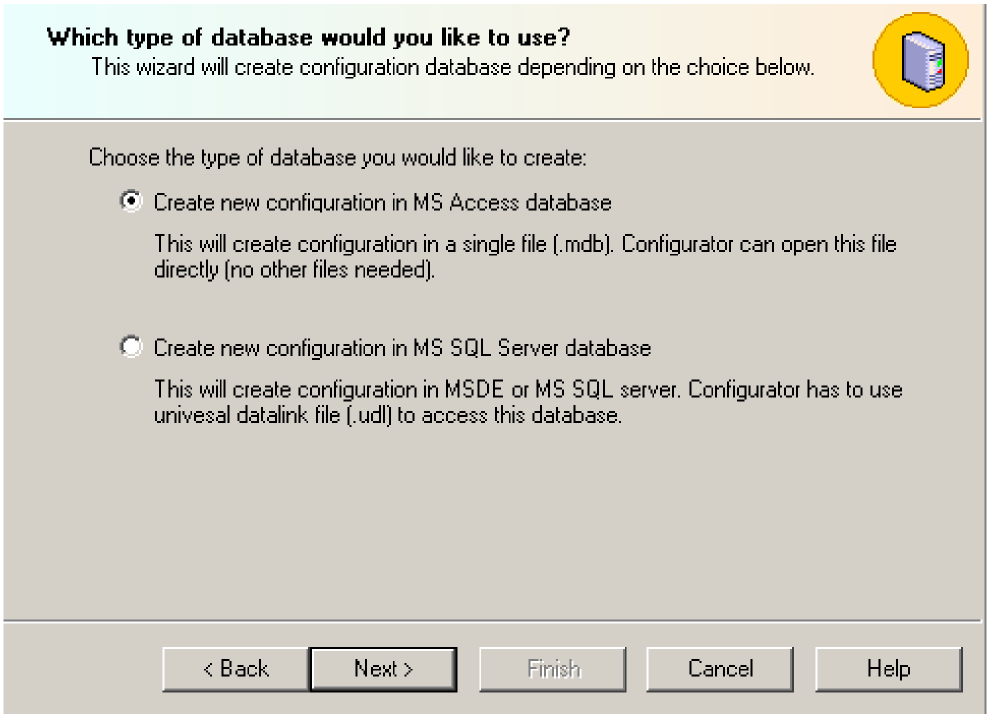
Creating a Configuration Database

2. The introduction screen for the Configuration Database Wizard appears. Click the **Next** button to continue.

3. You have two options for creating your new database, as shown in the figure below:

- **Create a new Microsoft Access configuration database:** For a Microsoft Access database, the Configurator uses a single .mdb file
- **Create a Microsoft SQL Server database:** The Configurator uses Universal Data Link (.udl) files to connect to a Microsoft SQL Server or MSDE database.

Select the database type you want to create, and then click **Next.**



Which type of database would you like to use?

This wizard will create configuration database depending on the choice below.

Choose the type of database you would like to create:

● Create new configuration in MS Access database

This will create configuration in a single file (.mdb). Configurator can open this file directly (no other files needed).

○ Create new configuration in MS SQL Server database

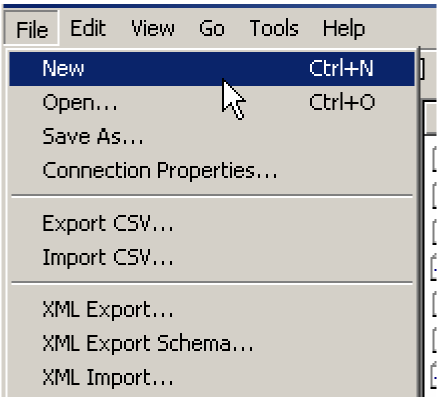
This will create configuration in MSDE or MS SQL server. Configurator has to use univesal datalink file (.udl) to access this database.

Choosing the Type of Database To Create

Creating a Microsoft Access Configuration Database

To create a new Microsoft Access configuration database in the Configurator:

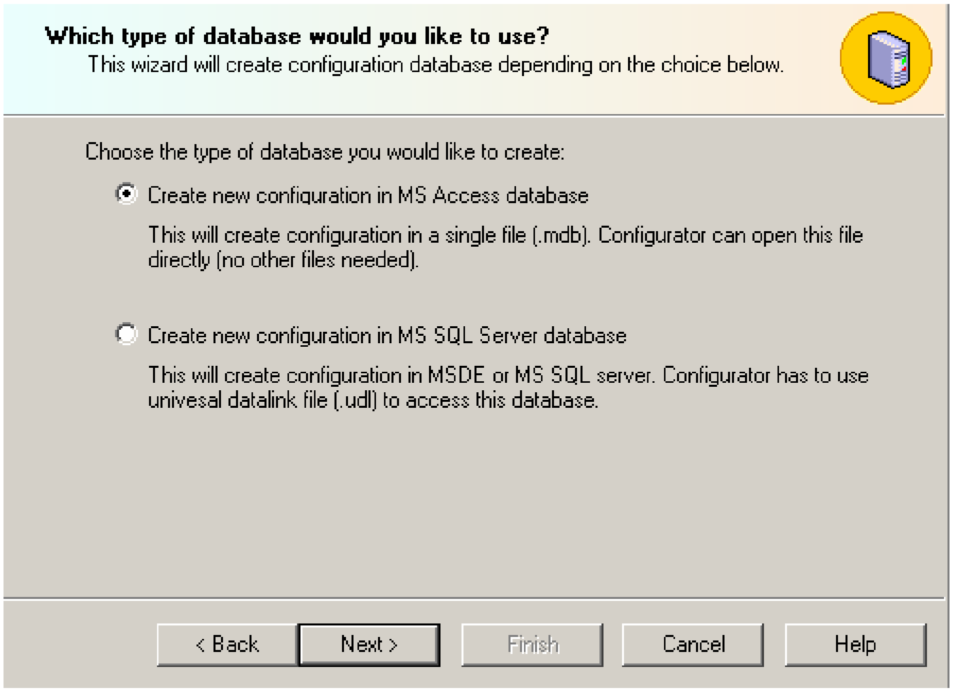
1. Select **New** from the **File** menu, as shown in the figure below**.**



Creating a Configuration Database

2. The introduction screen for the Configuration Database Wizard appears. Click the Next button to continue.

3. You have two options for creating your new database, as shown in the figure below. Select Create new configuration in MS Access database. Click the Next button to continue.



Which type of database would you like to use?
This wizard will create configuration database depending on the choice below.

Choose the type of database you would like to create:

● Create new configuration in MS Access database

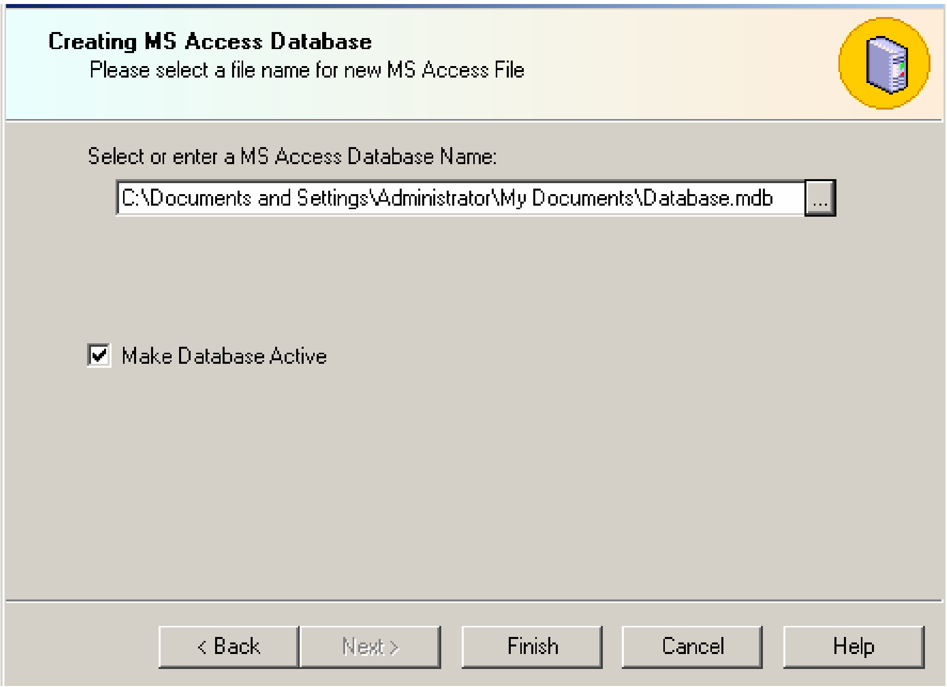
This will create configuration in a single file (.mdb). Configurator can open this file directly (no other files needed).

○ Create new configuration in MS SQL Server database

This will create configuration in MSDE or MS SQL server. Configurator has to use univesal datalink file (.udl) to access this database.

Creating a New Microsoft Access Database

4. Specify the directory path and file name for the new database, as shown in the figure below. Click the **...** button to browse for a directory. If you want this new database to be the active configuration database, check **Make Database Active.**



Creating MS Access Database
Please select a file name for new MS Access File

Select or enter a MS Access Database Name:

C:\Documents and Settings\Administrator\My Documents\Database.mdb

☑ Make Database Active

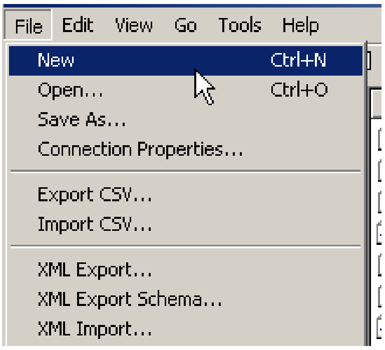
Naming the New Access Database

5. Click the **Finish** button. The new database is created and opened in the Configurator.

Creating a Microsoft SQL Server Configuration Database

To create a new Microsoft SQL Server configuration database in the Configurator:

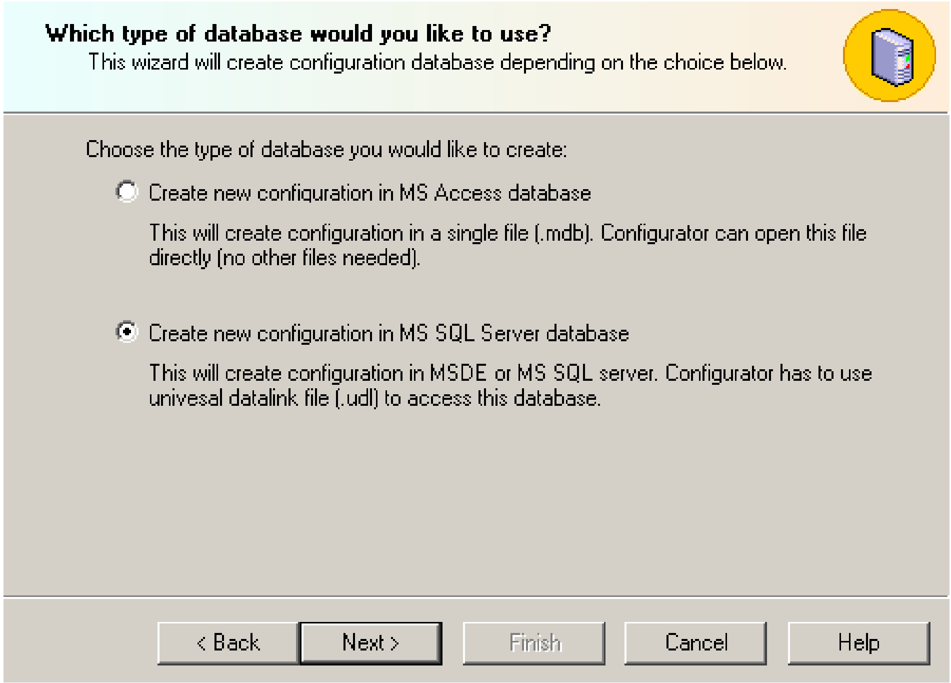
6. Select **New** from the **File** menu, as shown in the figure below**.**



Creating a Configuration Database

7. The introduction screen for the Configuration Database Wizard appears. Click the **Next** button to continue.

8. You have two options for creating your new database, as shown in the figure below. Select **Create new configuration in MS SQL Server database.** Click the **Next** button to continue.



Which type of database would you like to use?

This wizard will create configuration database depending on the choice below.

Choose the type of database you would like to create:

○ Create new configuration in MS Access database

This will create configuration in a single file (.mdb). Configurator can open this file directly (no other files needed).

● Create new configuration in MS SQL Server database

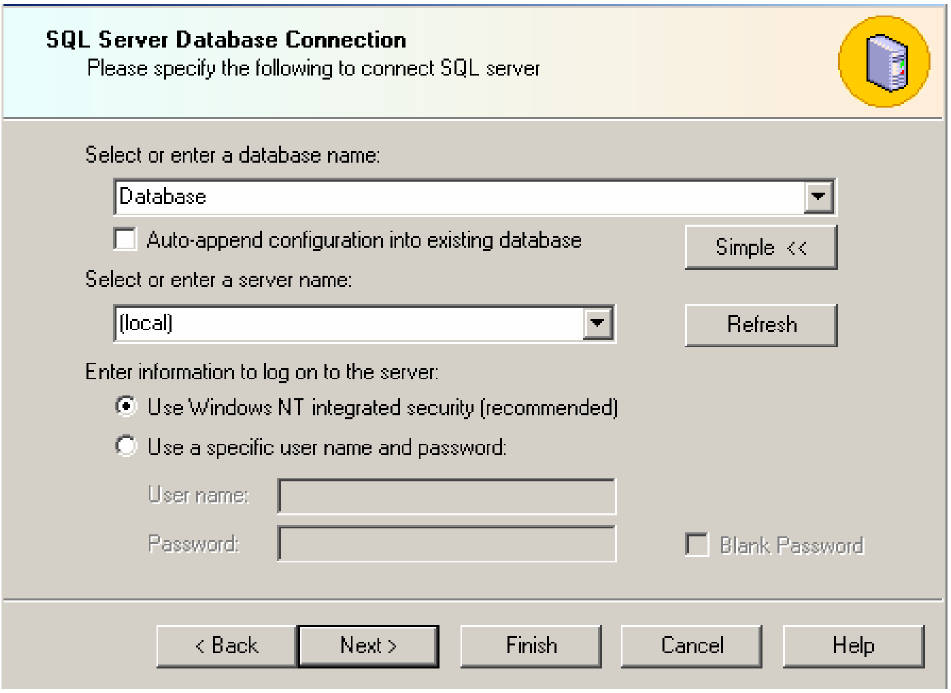
This will create configuration in MSDE or MS SQL server. Configurator has to use univesal datalink file (.udl) to access this database.

Creating a New Microsoft SQL Server Database

9. To connect to a SQL Server database, either select an existing database from the **Database Name** drop-down list, or a type a new name to create a new database, as shown in the figure below. If you choose to use an existing database, you have the option of adding (auto-appending) the configuration to the existing database. (For more information about the auto-append function, please see "Adding the Configuration to an Existing Database.")

10. In the **SQL Server Name** field, select the local SQL Server on which to create the database. If necessary, enter a user name and password to log on to the SQL Server. (It is recommended that you use Windows NT integrated security.)

11. **Note:** Usually you have only one instance of SQL Server running on the local node. In this case, the drop-down list under **SQL Server Name** has only one option: "(local)." However, it is possible to run multiple SQL Server instances on the local node, in which case the **SQL Server** field lists all those SQL Server instances: "(local)" for the default instance and "node_name/instance_name" for all others. The drop-down list may show SQL Server instances on other nodes as well.



SQL Server Database Connection
Please specify the following to connect SQL server

Select or enter a database name:
Database

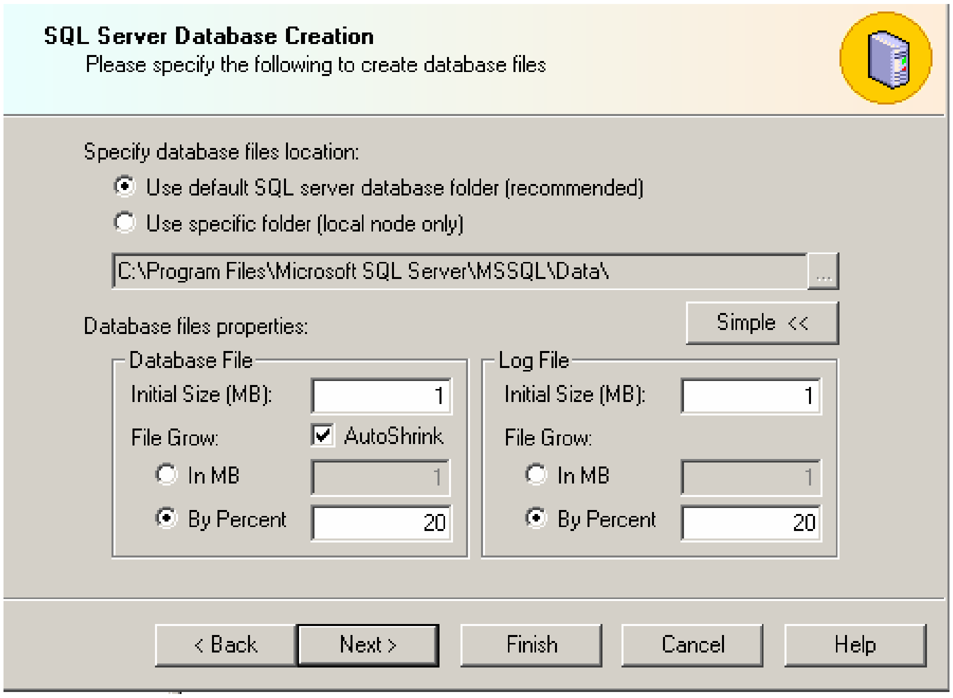
□ Auto-append configuration into existing database

Select or enter a server name:
(local)

Enter information to log on to the server:
● Use Windows NT integrated security (recommended)
○ Use a specific user name and password:
User name:
Password:
□ Blank Password

Connecting to a SQL Server Database

12. Specify a directory path location in which to create the database, as shown in the figure below. You can either use the default SQL Server database folder, or you can click the **...** button and browse for a specific folder.



SQL Server Database Creation
Please specify the following to create database files

Specify database files location:
● Use default SQL server database folder (recommended)
○ Use specific folder (local node only)
C:\Program Files\Microsoft SQL Server\MSSQL\Data\

Database files properties:

Database File
Initial Size (MB): 1
File Grow: ☑ AutoShrink
○ In MB 1
● By Percent 20

Log File
Initial Size (MB): 1
File Grow:
○ In MB 1
● By Percent 20

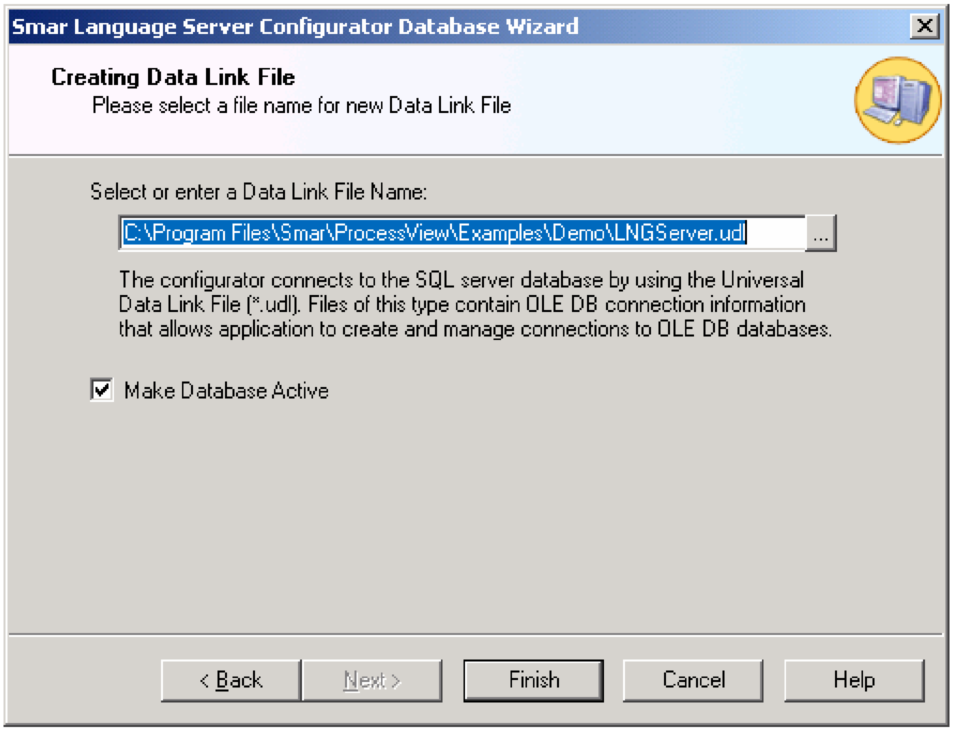
Specifying the Database Location and Properties

13. Under the **Database Properties** section, specify an initial size for the database, which should be as large as possible. You can also specify a **Database Growth** option (in megabytes) or as a percentage of the total size. MSDE servers are capable of growing the database on the fly to store more data. However, if this operation is performed frequently, the overall system performance may decrease. Choosing an initially large database size and a corresponding database growth option can drastically improve system performance.

14. Under the **Log File Properties** section, you can also modify the settings for the database transaction log file. Specify a **Log File Growth** option (in megabytes) or as a percentage of the total size. Again, a sufficient initial size setting can greatly improve performance. The default options should be adequate for most applications with a small to medium size load.

15. Click the **Next** button to create the new SQL Server database.

16. The Configurator uses Universal Data Link (.udl) files to connect to the Microsoft SQL Server database. These .udl files contain OLE database connection information that allows the Configurator to create and manage connections to OLE databases. Enter a name and directory path for the new .udl file in the **Data Link File Name** field, as shown in the figure below. You can browse for a directory by clicking the **...** button to the right of this field. Click the **Next** button to continue. If you want this new database to be the active configuration database, check **Make Database Active.**



Smar Language Server Configurator Database Wizard

Creating Data Link File

Please select a file name for new Data Link File

Select or enter a Data Link File Name:

C:\Program Files\Smar\ProcessView\Examples\Demo\LNGServer.udl

The configurator connects to the SQL server database by using the Universal Data Link File (*.udl). Files of this type contain OLE DB connection information that allows application to create and manage connections to OLE DB databases.

☑ Make Database Active

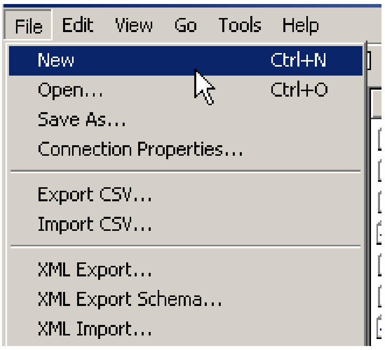
Creating a Universal Data Link File

17. Click the **Finish** button. The new database is created and opened in the Configurator.

Adding a Configuration Database to an Existing SQL Server Database

The Configuration Database Wizard also gives you the option to add the configuration database structure to an existing Microsoft SQL Server database:

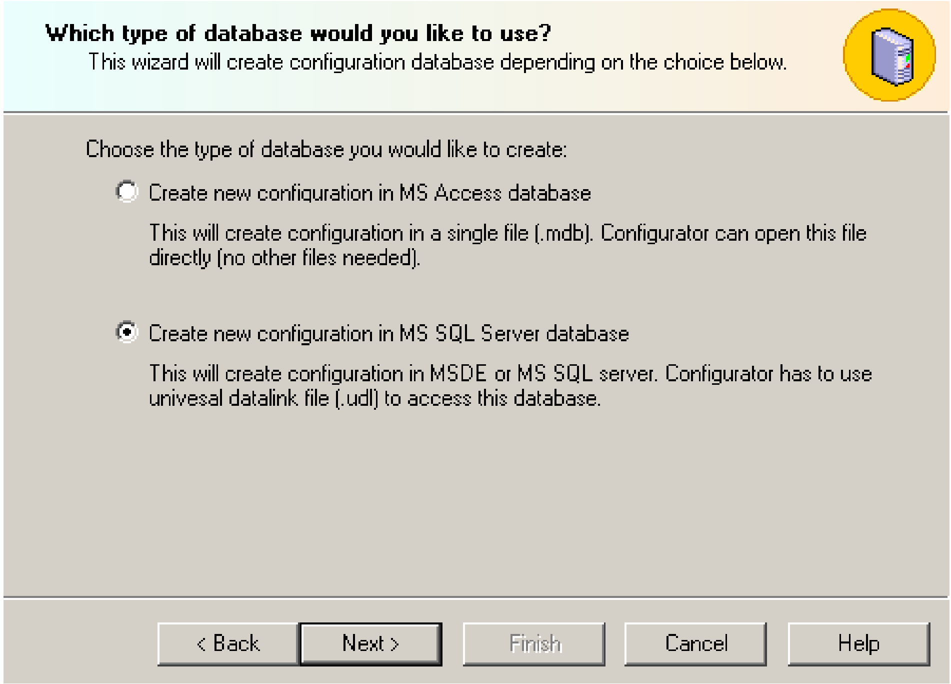
1. Select **New** from the **File** menu, as shown in the figure below.



Creating a Configuration Database

2. The introduction screen for the Configuration Database Wizard appears. Click the **Next** button to continue.

3. You have two options for creating your new database, as shown in the figure below. Select **Create new configuration in MS SQL Server database.** Click the **Next** button to continue.



Which type of database would you like to use?
This wizard will create configuration database depending on the choice below.

Choose the type of database you would like to create:

○ Create new configuration in MS Access database

This will create configuration in a single file (.mdb). Configurator can open this file directly (no other files needed).

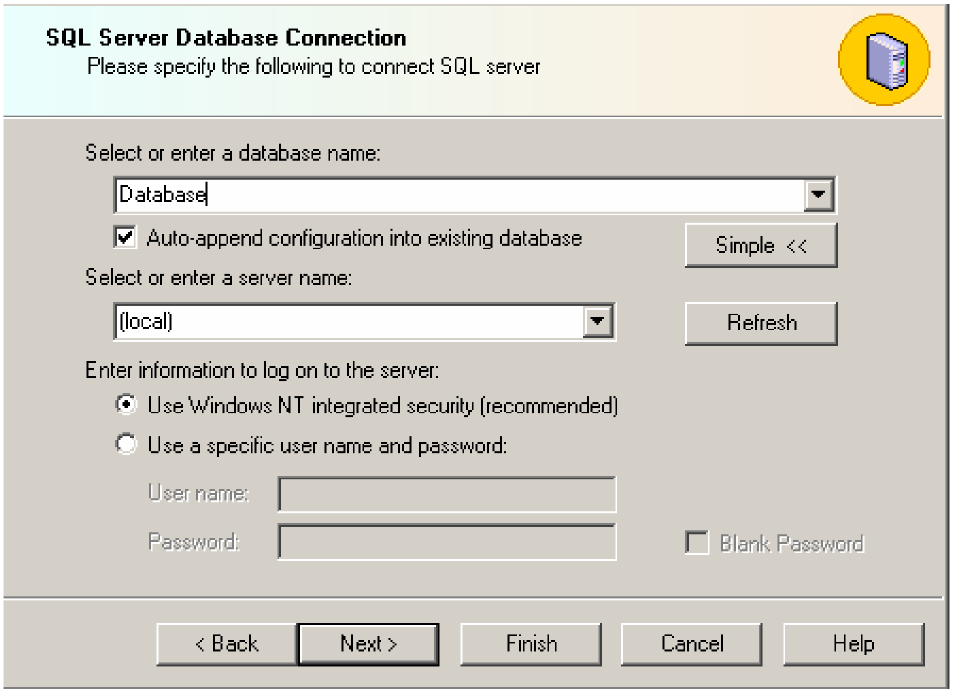
● Create new configuration in MS SQL Server database

This will create configuration in MSDE or MS SQL server. Configurator has to use univesal datalink file (.udl) to access this database.

Creating a Microsoft SQL Server Database

4. To connect to a SQL Server database, select the existing database from the drop-down list, as shown in the figure below. Check the **Auto-append configuration into existing database** check box. In the **SQL Server** field, select the local SQL Server on which to create the database. Type a name for the database in the **Database Name** field. If necessary, enter a user name and password to log on to the SQL Server. (**Note:** It is recommended that you use Windows NT integrated security.) Click the **Next** button to continue.

5. **Note:** Usually you have only one instance of SQL Server running on the local node. In this case, the drop-down list under **SQL Server Name** has only one option: "(local)." However, it is possible to run multiple SQL Server instances on the local node, in which case the **SQL Server** field lists all those SQL Server instances: "(local)" for the default instance and "node_name/instance_name" for all others. The drop-down list may show SQL Server instances on other nodes as well.



SQL Server Database Connection
Please specify the following to connect SQL server

Select or enter a database name:
Database

☑ Auto-append configuration into existing database

Select or enter a server name:
(local)

Enter information to log on to the server:
● Use Windows NT integrated security (recommended)
○ Use a specific user name and password:

User name:

Password:

☐ Blank Password

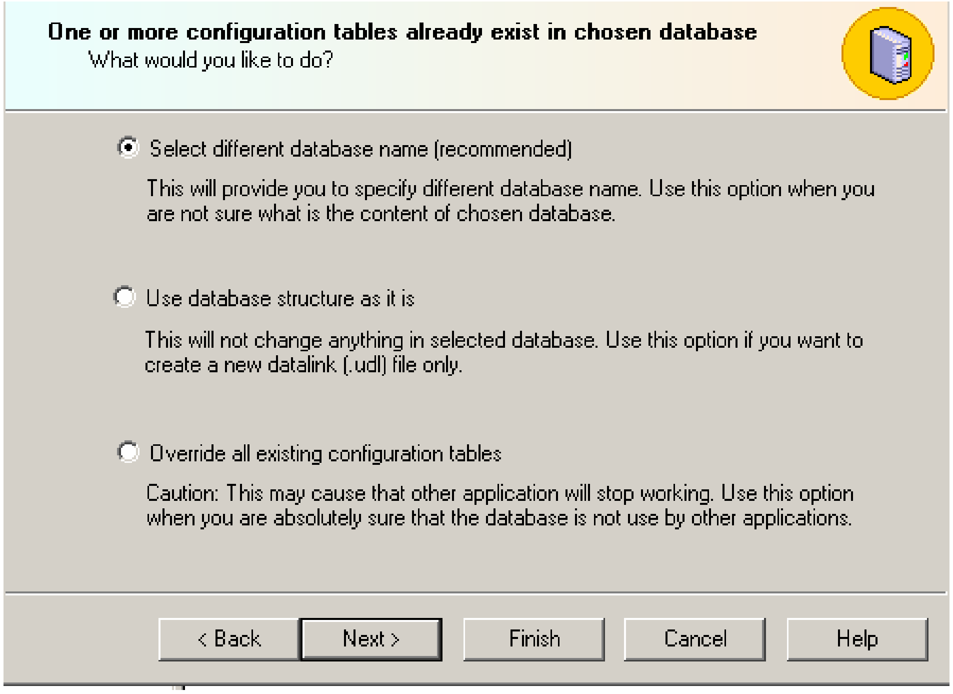
Connecting to an Existing SQL Server Database

6. If the existing database already has configuration tables, you have the following options, as shown in the figure below:

- **Select a different database name (recommended):** This option allows you to rename the database without affecting the existing database as well as create a new Data Link (.udl) file.
- **Use the existing database structure:** This option preserves the content of the existing database and allows you to create a new Universal Data Link (.udl) file.
- **Override all existing configuration tables:** This option expunges the content of the existing database and allows you to create a new Universal Data Link (.udl) file.

Note: Overriding the database may disable other applications that also use the database.

Click the **Next** button to continue.



One or more configuration tables already exist in chosen database
What would you like to do?

● Select different database name (recommended)

This will provide you to specify different database name. Use this option when you are not sure what is the content of chosen database.

○ Use database structure as it is

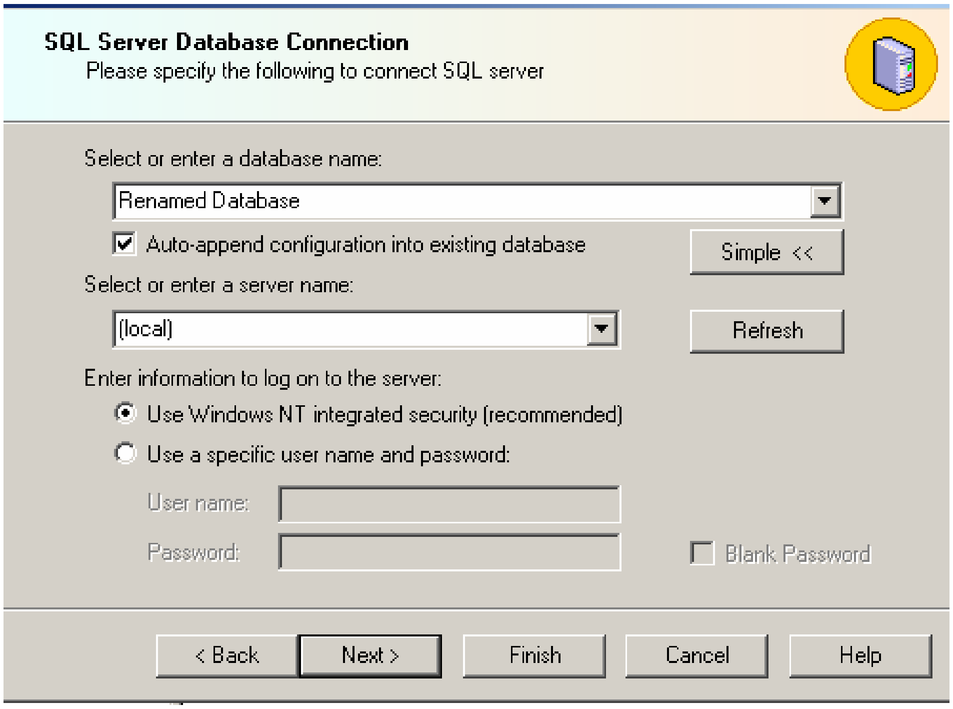
This will not change anything in selected database. Use this option if you want to create a new datalink (.udl) file only.

○ Override all existing configuration tables

Caution: This may cause that other application will stop working. Use this option when you are absolutely sure that the database is not use by other applications.

Adding the Configuration to an Existing SQL Server Database

7. If you chose **Select a different database name (recommended),** you are directed back to the SQL Server Database Connection dialog box, as shown in the figure below. Enter a new name for the database, and then click **Next.**



SQL Server Database Connection
Please specify the following to connect SQL server

Select or enter a database name: Renamed Database

☑ Auto-append configuration into existing database

Select or enter a server name: (local)

Enter information to log on to the server:

● Use Windows NT integrated security (recommended)

○ Use a specific user name and password:

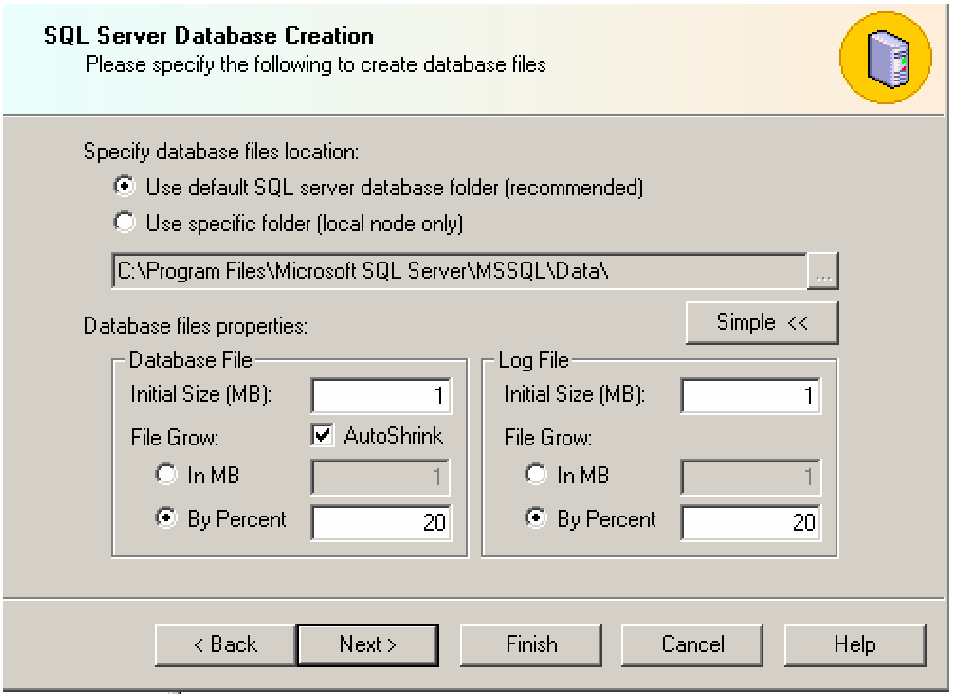
User name:

Password:

☐ Blank Password

Renaming the Existing SQL Server Database

8. Specify a directory path location in which to create the database, as shown in the figure below. You can either use the default SQL Server database folder, or you can click the ... button and browse for a specific folder.



SQL Server Database Creation
Please specify the following to create database files

Specify database files location:

● Use default SQL server database folder (recommended)

○ Use specific folder (local node only)

C:\Program Files\Microsoft SQL Server\MSSQL\Data\

Database files properties:

Database File
Initial Size (MB): 1
File Grow: ☑ AutoShrink
○ In MB 1
● By Percent 20

Log File
Initial Size (MB): 1
File Grow:
○ In MB 1
● By Percent 20

Specifying the Database Location and Properties

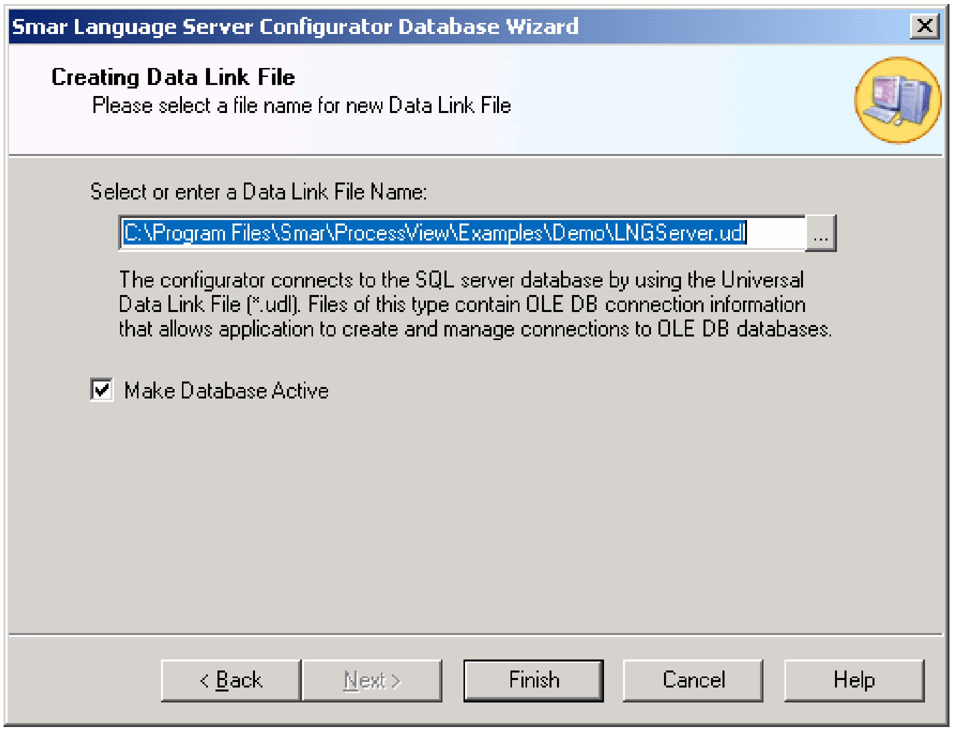
9. Under the **Database Properties** section, specify an initial size for the database, which should be as large as possible. You can also specify a **Database Growth** option (in megabytes) or as a percentage of the total size. MSDE servers are capable of growing the database on the fly to store more data. However, if this operation is performed frequently, the overall system performance may decrease. Choosing an initially large database size and a corresponding database growth option can drastically improve system performance.

10. Under the **Log File Properties** section, you can also modify the settings for the database transaction log file. Specify a **Log File Growth** option (in megabytes) or as a percentage of the total size. Again, a sufficient initial size setting can greatly improve performance. The default options should be adequate for most applications with a small to medium size load.

11. Click the **Next** button to create the new SQL Server database.

12. The Configurator uses Universal Data Link (.udl) files to connect to the Microsoft SQL Server database. These .udl files contain OLE database connection information that allows the Configurator to create and manage connections to OLE databases. Enter a name and directory path for the new .udl file in the **Data Link File Name** field, as shown in the figure below. You can browse for a directory by clicking the **…** button to the right of this field. Click the **Next** button to continue. If you want this new database to be the active configuration database, check **Make Database Active.**

13. Note: If you chose to **Use the existing database structure** or to **Override all existing configuration tables,** you will still need to create a new Data Link file.


Smar Language Server Configurator Database Wizard

Creating Data Link File

Please select a file name for new Data Link File

Select or enter a Data Link File Name:

C:\Program Files\Smar\ProcessView\Examples\Demo\LNGServer.udl

The configurator connects to the SQL server database by using the Universal Data Link File (*.udl). Files of this type contain OLE DB connection information that allows application to create and manage connections to OLE DB databases.

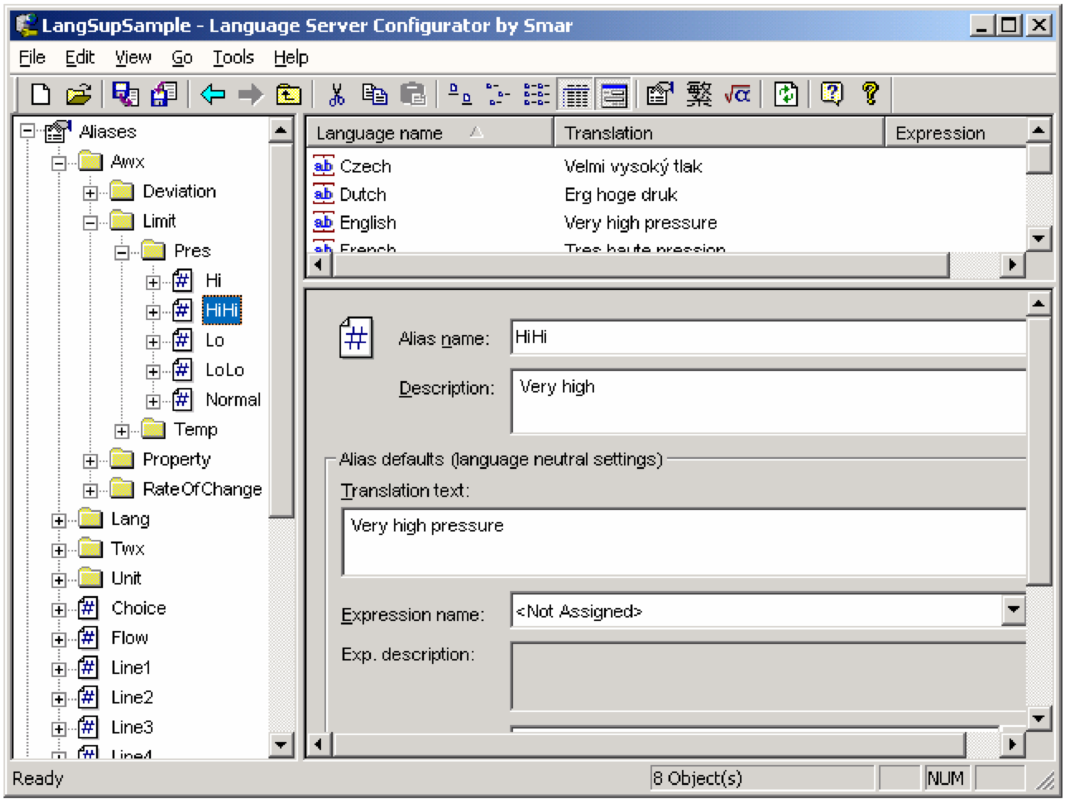
☑ Make Database Active

Creating a Universal Data Link File

14. Click the **Finish** button. The new database is created and opened in the Configurator.

Aliases Level

The **Aliases** level of the tree view contains various folders and aliases. The Language Configurator provides many aliases and folders by default. When you click on an alias, the properties of the alias are shown in the right-hand pane of the screen. For example, when you click the **HiHi** alias, as shown in the figure below, the alias name and description are displayed in right-hand pane. The translation text and the translation languages are also shown, as well as any expressions and formats associated with the alias. You can add a new alias to your configuration database for each item you wish to convert. For more information, see the **Adding a New Alias** section. For information on how to configure a language translation for each language under each alias you wish to convert see the **Configuring a Language Translation** section.



Language name	Translation	Expression
Czech	Velmi vysoký tlak	
Dutch	Erg hoge druk	
English	Very high pressure	
French	Tres haute pression	

Alias name: HiHi

Description: Very high

Alias defaults (language neutral settings)

Translation text: Very high pressure

Expression name: <Not Assigned>

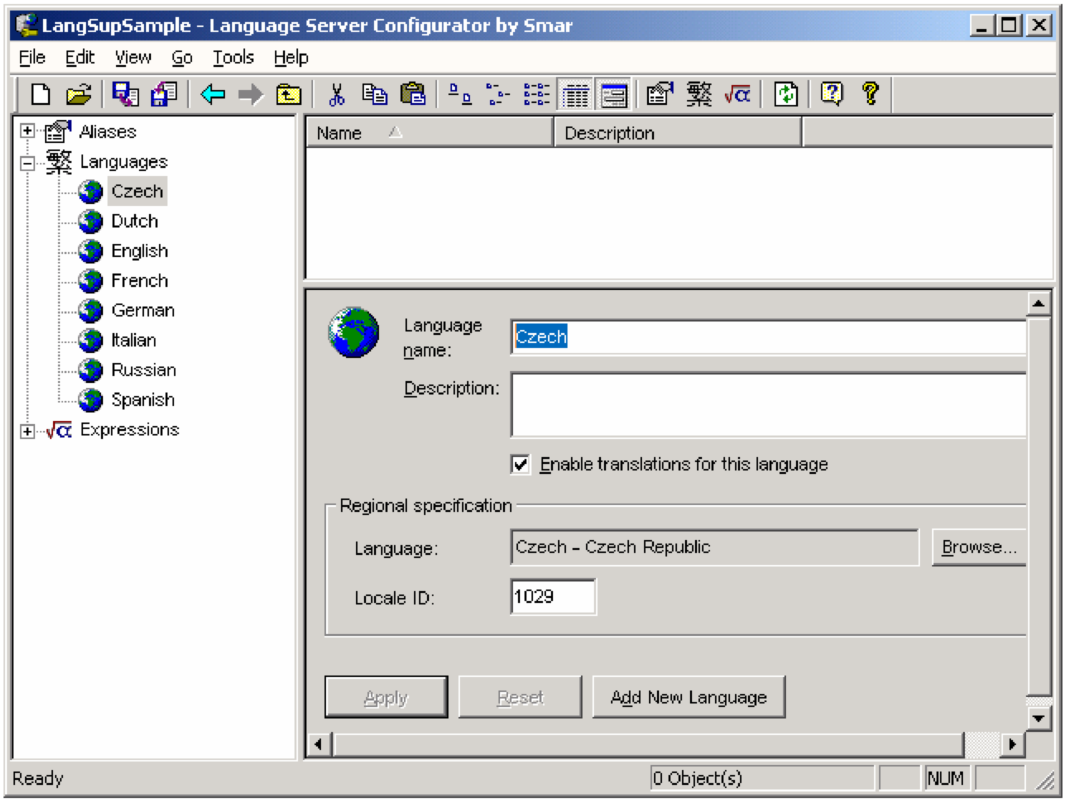
Exp. description:

Alias Properties

Languages Level

The **Languages** level of the tree view shows all languages in the configuration database. The Language Configurator provides Czech, Dutch, English, French, German, Italian, Russian, and Spanish as default languages. When you click on a language, the language name and description are displayed in the right-hand pane, as shown in the figure below. You can include additional comments in the **Description** field. The **Enable translations for this language** check box is checked by default. If you prefer a regional language, you can search for a **Regional Specification** of a particular language by clicking the **Browse** button. The **Locale ID** that corresponds with the selected language is also displayed.

You can add a new language to your configuration database. For more information, see the **Adding a New Language** section. You can also add language subsets, which allow you to create an additional configuration to be used with a language. For more information about language subsets, see the **Creating Language Subsets** section.



Language name: Czech

Description:

☑ Enable translations for this language

Regional specification

Language: Czech - Czech Republic

Locale ID: 1029

Ready | 0 Object(s) | NUM

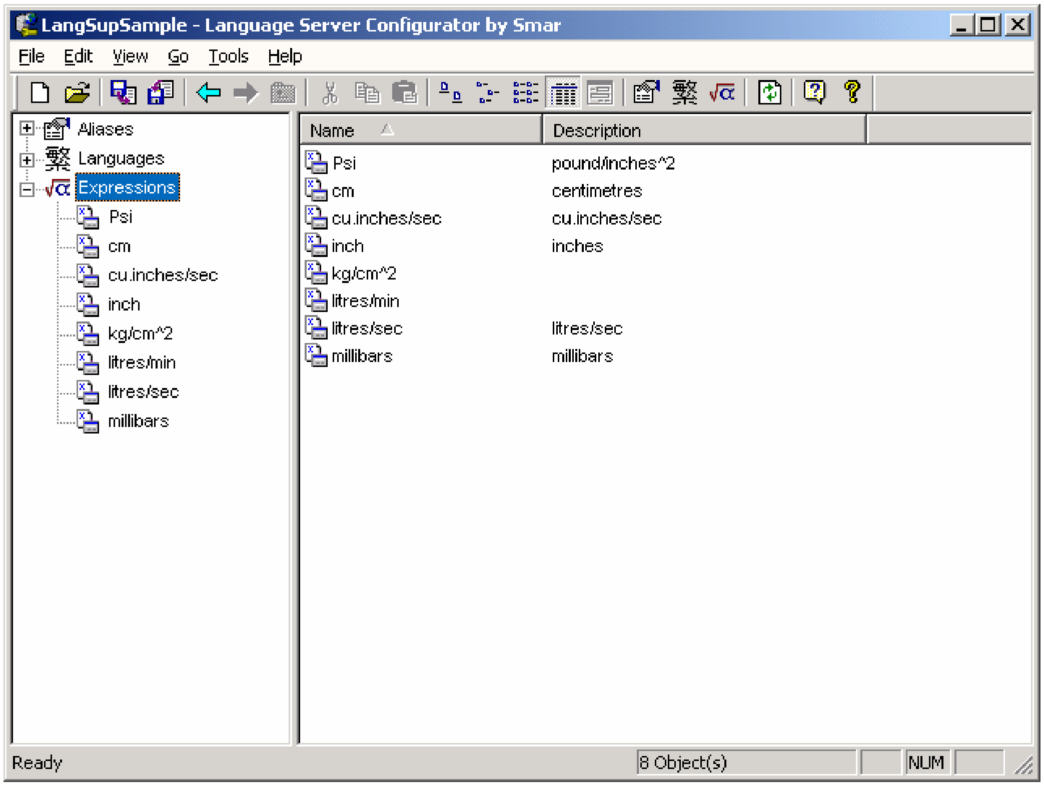
Language Properties

Expressions Level

The **Expressions** level of the tree view shows all expressions in the configuration database. The Language Configurator provides the following units as default expressions.

- Psi
- cm
- cu.inches/sec
- inch
- kg/cm^2
- liters/min
- liters/sec
- millibars

When you click on an expression, the expression name and description are displayed in the right-hand pane, as shown in the figure below. You can include additional comments in the **Description** field. The read and write expression parameters are also shown. You can modify the expression by clicking the **Exp** button, which opens the **Edit Expression** dialog box. You can also add new expressions to the configuration database. For more information, see the **Adding a New Expression** section.



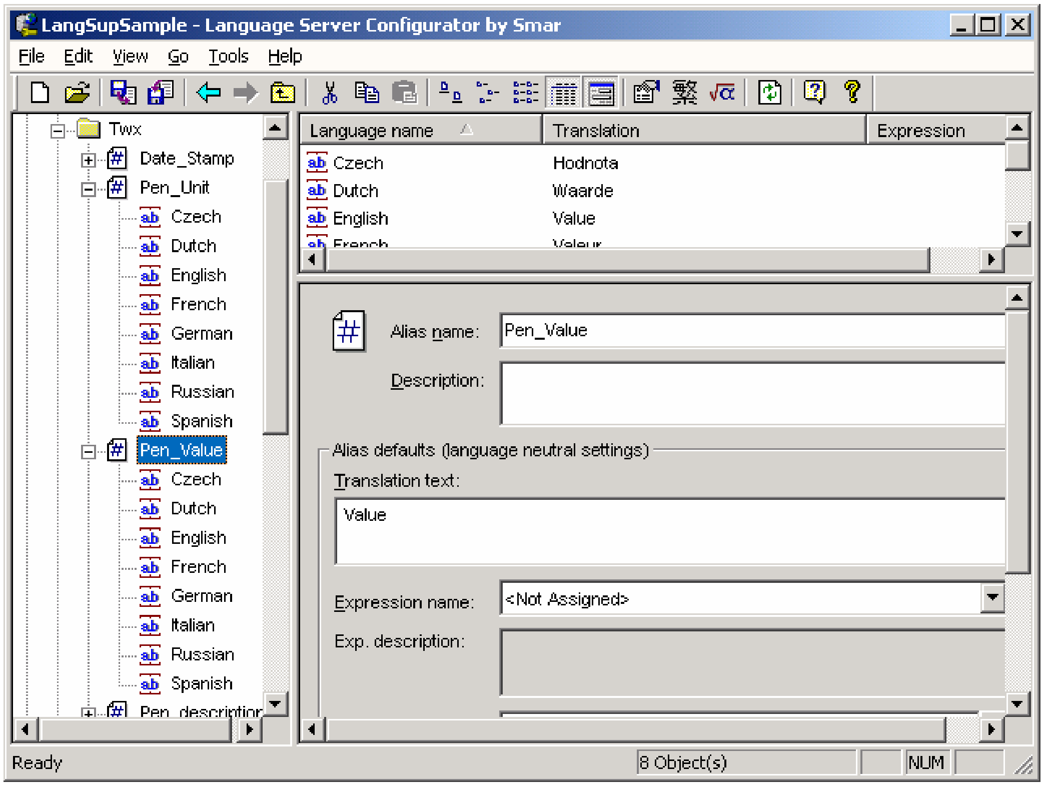
Name	Description
Psi	pound/inches^2
cm	centimetres
cu.inches/sec	cu.inches/sec
inch	inches
kg/cm^2	
litres/min	
litres/sec	litres/sec
millibars	millibars

Ready 8 Object(s) NUM

Expression Properties

Icons

As you can see in the figure below, you will encounter various icons in the tree view of the Language Configurator screen.



Language name	Translation	Expression
Czech	Hodnota	
Dutch	Waarde	
English	Value	
French	Valeur	

Alias name: Pen_Value

Description:

Alias defaults (language neutral settings)

Translation text:

Value

Expression name: <Not Assigned>

Exp. description:

Ready 8 Object(s) NUM

Icons

When a language translation for an alias is properly defined, the alias is indicated by the blue "pound sign" icon shown at left.

The red "pound sign" icon shown at left indicates that all languages for an alias are translated but that the alias contains different kinds of translations. For example, one language might have an associated text translation, while another language might have an associated expression. However, the red "pound sign" icon can also indicate that something is wrong with one of the translations, so you should check the translations.

Translated languages for each alias are indicated by the icon shown at left.

Translated language subsets for each alias are indicated by the icon shown at left.

Translated languages that contain an expression are indicated by the icon shown at left.

Translated language subsets that contain an expression are indicated by the icon shown at left.

Translated languages that contain an expression but no text are indicated by the icon shown at left.

Translated language subsets that contain an expression but no text are indicated by the icon shown at left.

When a language translation is defined and the translation text is missing, or at least one of the languages is not translated, then the alias is indicated by the "red exclamation point" icon shown at left.

A language that is not properly translated is indicated by the "blue question mark" icon shown at left.

A language subset that is not properly translated is indicated by the "green question mark" icon shown at left.

In the Languages level of the tree, languages are indicated by the "globe" icon shown at left.

In the Languages level of the tree, language subsets are indicated by the icon shown at left.

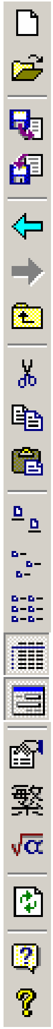
In the Expressions level of the tree, expressions are indicated by the icon shown at left.

Toolbars

The Language Configurator contains two toolbars that provide easy access to command functions. The **Standard** toolbar contains commands that enable you to modify the Language Configurator screen. The **Data-Manipulation** toolbar contains commands that affect language configuration.

Standard Toolbar

To display the **Standard** toolbar of the Language Configurator, shown below, select **Toolbars > Standard Buttons** from the **View** menu. The **Standard** toolbar of the Language Configurator, shown below, contains the following command buttons.



New: Creates a new configuration database.

Open: Opens an existing configuration database.

Import Data: Imports configuration data from a text file (.txt) or a Microsoft Excel file (.csv).

Export Data: Exports configuration data to a text file (.txt) or a Microsoft Excel file (.csv).

Back: Moves the cursor back to the previously selected item in the tree control.

Next: Moves the cursor to the next item in the tree control.

Up One Level: Moves up one level in the tree control.

Cut: Deletes current selection, sending it to the clipboard.

Copy: Copies the current selection to the clipboard.

Paste: Pastes the current contents of the clipboard.

Large Icons: Displays items as large icons.

Small Icons: Displays items as small icons.

List: Displays items as a list.

Details: Displays items as a list with details.

Dialog view: Displays additional configuration options.

Aliases: Moves the cursor to the **Aliases** tree control.

Languages: Moves the cursor to the **Languages** tree control.

Expressions: Moves the cursor to the **Expressions** tree control.

Global Refresh: Refreshes the data for the entire Configurator screen.

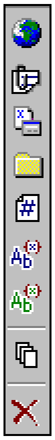
About: Displays information about the application.

Help Topics: Launches online Help for the application.

Most of the commands on the **Standard** toolbar are also available in the menus. For more information about the functions of these commands, refer to the **Menus** section.

Data-Manipulation Toolbar

To display the **Data-Manipulation** toolbar of the Language Configurator, shown below, select **Toolbars** > **Data Manipulation Buttons** from the **View** menu. The **Data-Manipulation** toolbar contains the following command buttons.



New 'Regional Specification': Creates a new language item.

New 'Subset': Creates a new language subset item.

New 'Expression': Creates a new expression item.

New 'Folder': Creates a new folder.

New 'Alias': Creates a new alias.

New 'Translation (Languages)': Creates a new translation language for the selected alias.

New 'Translation (Language Subsets)': Creates a new translation language subset for the selected alias.

Multiply: Creates multiple copies of the selected item.

Delete: Deletes the selected item.

Menus

The Language Configurator contains the following menus:

- **File menu**
- **Edit menu**
- **View menu**
- **Go menu**
- **Tools menu**
- **Help menu**

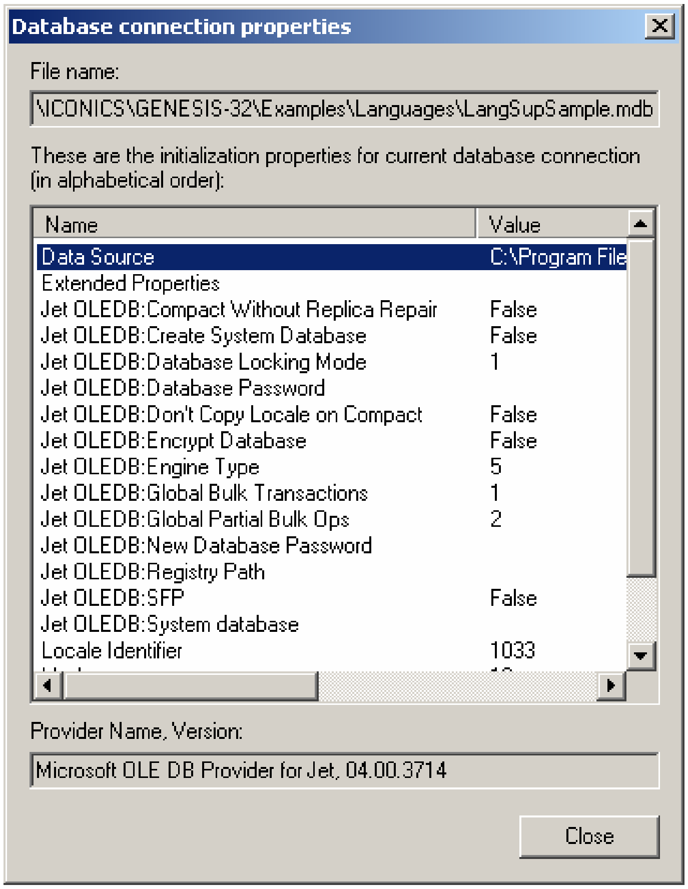
File Menu

The **File** menu contains the following commands:

Command	Shortcut Keys	Function
New	CTRL+N	Creates a new configuration database.
Open	CTRL+O	Opens a Microsoft Access (.mdb) or Microsoft Data Link (.udl) file, which allows you to connect to any OLE database source, such as a SQL database.
Save As		Saves the current database under a different name as a Microsoft Access (.mdb) or Microsoft Data Link (.udl) file.
Connection Properties		Displays the current database connection properties (see below).
Export Dictionary		Exports aliases and translations from your database to a text file (.txt) or a Microsoft Access file (.mdb). You can specify the delimiters and what to export.
Import Dictionary		Imports aliases and translations into your configuration database from a text file (.txt) or a Microsoft Access file (.mdb). You can then specify the delimiters and choose from the import settings.
Export Data		Exports configuration data from your database to a text file (.txt) or a Microsoft Excel file (.csv). You can specify the delimiters and what to export.
Import Data		Imports data into your configuration database from a text file (.txt) or a Microsoft Excel file (.csv). You can then specify the delimiters and choose from the import settings.
XML Export		Exports configuration data to an XML file.
XML Export Schema		Exports configuration data to an XML Schema file.
XML Import		Imports configuration data from an XML file.
XML Validate		This feature does not import an XML data file, but it will try to validate its structure using stored XML schema. Once it passes this validation, the XML file is acceptable for import by the Configurator.
Make Active		Makes the current database active for use by the global aliasing engine. If this command is not available, then the current database is already the active database.
Exit		Closes the applications.

Database Connection Properties

Selecting **Connection Properties** from the **File** menu opens the **Database Connection Properties** dialog box, shown below, which lists the initialization properties for the current database connection.



File name:

\ICONICS\GENESIS-32\Examples\Languages\LangSupSample.mdb

These are the initialization properties for current database connection (in alphabetical order):

Name	Value
Data Source	C:\Program File
Extended Properties	
Jet OLEDB:Compact Without Replica Repair	False
Jet OLEDB:Create System Database	False
Jet OLEDB:Database Locking Mode	1
Jet OLEDB:Database Password	
Jet OLEDB:Don't Copy Locale on Compact	False
Jet OLEDB:Encrypt Database	False
Jet OLEDB:Engine Type	5
Jet OLEDB:Global Bulk Transactions	1
Jet OLEDB:Global Partial Bulk Ops	2
Jet OLEDB:New Database Password	
Jet OLEDB:Registry Path	
Jet OLEDB:SFP	False
Jet OLEDB:System database	
Locale Identifier	1033

Provider Name, Version:

Microsoft OLE DB Provider for Jet, 04.00.3714

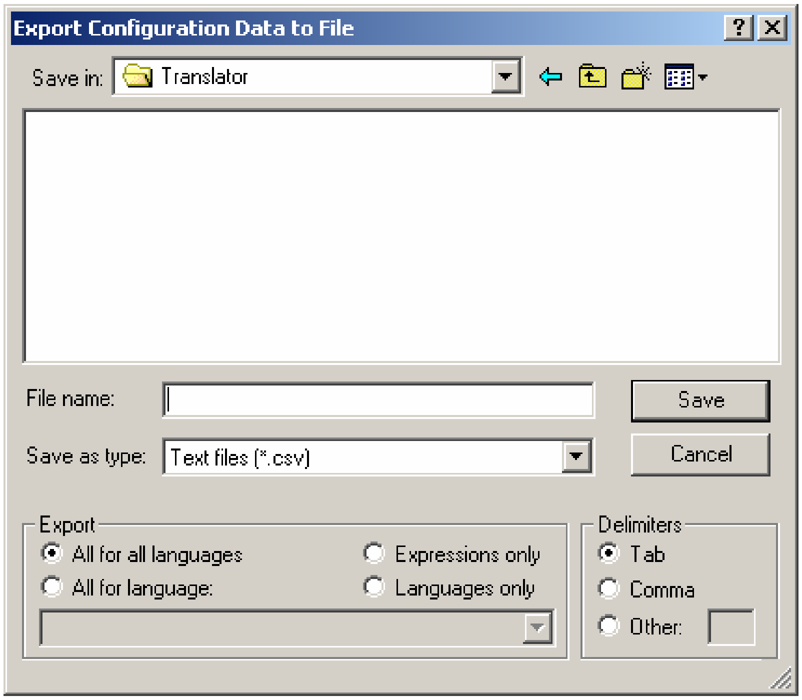
Database Connection Properties Dialog Box

Exporting Configuration Data

The Configurator offers the flexibility of exporting data from your configuration database to a text (.txt) file or a Microsoft Excel (.csv) file. To export data, select **Export Data** from the **File** menu. This opens the **Export Configuration Data to File** dialog box, shown below. You can then specify the delimiters and what to export:

- All for all languages
- All for language
- Expressions only
- Languages only

If you select **All for language,** the list of languages will become available at the bottom of the dialog box.



Save in: Translator

File name:

Save as type: Text files (*.csv)

Export
● All for all languages
○ All for language:
○ Expressions only
○ Languages only

Delimiters
● Tab
○ Comma
○ Other:

Export Configuration Data to File Dialog Box

Text file format. Choose the folder to which you want to export the data from your database. When Text files (*csv) is selected in the **Save As Type** field. When you click **Save,** the data will be saved to a Unicode text file. Microsoft Excel is one of the many formats that enable you to edit the file. In the newly created file, the data are grouped into languages, expressions, and aliases. Unless you specify delimiters in the **Export Configuration Data to File** dialog box, the file will use **Tabs** as delimiters by default. Each group contains headings and columns that provide information about each item, such as descriptions and associated translations and expressions. It also provides the "tree" pathway for each item.

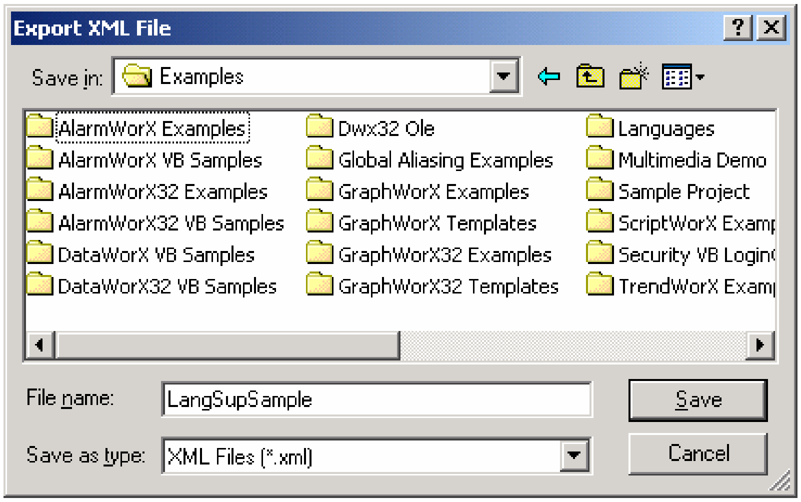
Exporting Data to an XML File

The Configurator also allows you to export data from your configuration database to an XML file. The XML export/import functionality was mainly developed for Windows platforms that do not support databases (e.g. Windows CE and Windows Embedded). XML has the following advantages over the CSV import/export function:

- XML has a standardized format, unlike the text/CSV format, which uses various delimiters (e.g. TAB instead of commas, strings could not accept all characters, etc.)
- XML is language-independent, whereas CSV converts date/time, floats, and currency fields according to local settings in Windows. For example, using CSV, you cannot export data on German Windows and import it on English windows without making changes
- Windows has an installed automation object that has the capability to work with XML. Thus, programmers can create/modify their configurations outside the Configurator using Visual Basic, if desired.
- XML supports schemas. A **schema** is a special XML file that specifies the data structure of an XML data file.

To export data, select **XML Export** from the **File** menu. This opens the **Export XML File** dialog box, as shown in the figure below. Give the file a name, and then choose the directory to which you want to export the data from your database. Click **Save.**

Note: You can also export configuration data to an XML Scheme file by selecting **XML Export** from the **File** menu.


Export XML File

Save in: Examples

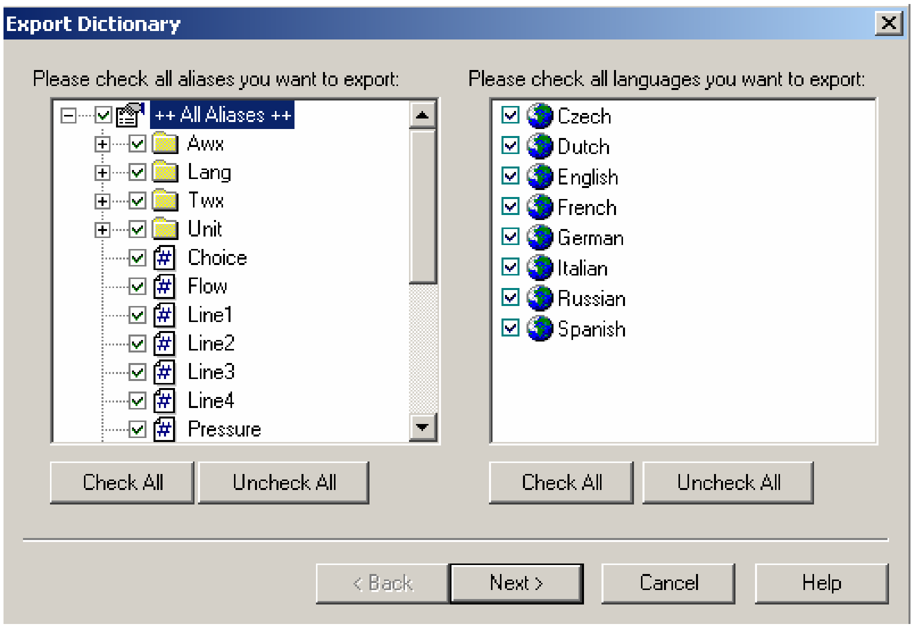
File name: LangSupSample

Save as type: XML Files (*.xml)

Exporting Configuration Data to an XML File

Exporting Aliases and Translations to a Dictionary File

The Configurator gives you the option of exporting aliases and their language translations from your configuration database to a text (.txt) file or a Microsoft Excel (.csv) file "dictionary." To export data to a dictionary, select **Export Dictionary** from the **File** menu. This opens the **Export Dictionary** dialog box, shown below. In the left pane, check the aliases that you want to include in the dictionary. In the right pane, select the language translations for the aliases. Click the **Next** button.


Export Dictionary

Please check all aliases you want to export:

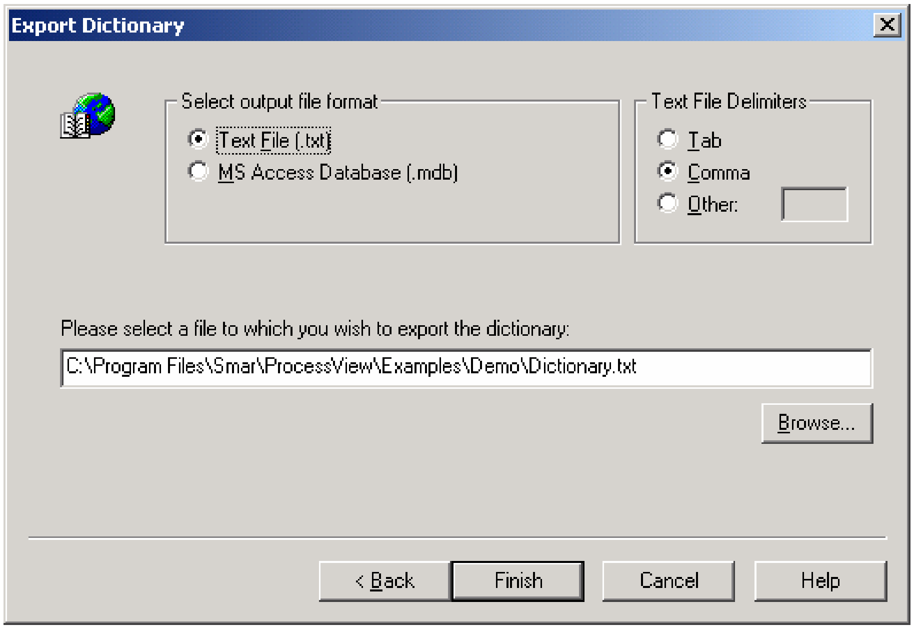
- ☑ ++ All Aliases ++
  - ☑ Awx
  - ☑ Lang
  - ☑ Twx
  - ☑ Unit
  - ☑ Choice
  - ☑ Flow
  - ☑ Line1
  - ☑ Line2
  - ☑ Line3
  - ☑ Line4
  - ☑ Pressure

Please check all languages you want to export:

- ☑ Czech
- ☑ Dutch
- ☑ English
- ☑ French
- ☑ German
- ☑ Italian
- ☑ Russian
- ☑ Spanish

Choosing Aliases and Language Translations to Export

Select the output file format (text file or Microsoft Access file), and then click the **Browse** button to specify a file name for the new dictionary file. You can then specify the delimiters for export. Click **Finish** to generate the dictionary.



Export Dictionary

Select output file format

● Text File (.txt)

○ MS Access Database (.mdb)

Text File Delimiters

○ Tab

● Comma

○ Other:

Please select a file to which you wish to export the dictionary:

C:\Program Files\Smar\ProcessView\Examples\Demo\Dictionary.txt

Choosing the File and Delimiters for Export

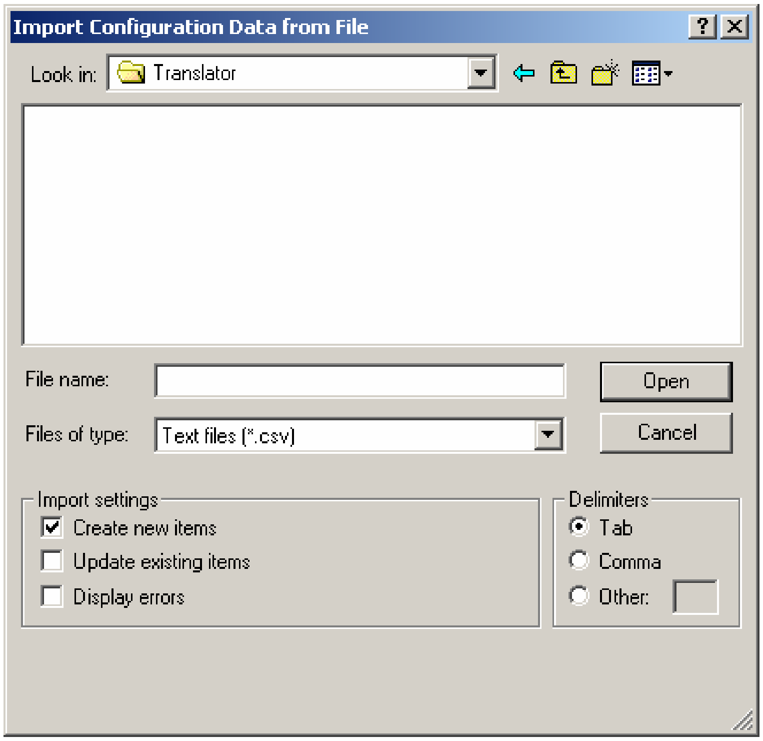
Importing Configuration Data

The Configurator offers the flexibility of importing data from a text (.txt) file or a Microsoft Excel (.csv) file to your configuration database. To import data, select **Import Data** from the **File** menu. This opens the **Import Configuration Data From File** dialog box, shown below. You can then specify the delimiters and choose from the following import settings:

- **Create new items.** When the import file contains items that are not yet in the configuration database, then it will create them. Otherwise it will skip these items.
- **Update existing items.** When the import file contains items that are in the configuration database, then it will update them using data from the import file. Otherwise it will skip these items.

Note: Either **Create new items** or **Update existing items** must be selected. Otherwise there is nothing to import.

- **Display errors.** When this item is checked, the Configurator will show a dialog box if an error occurs, and then will ask you if you want to proceed with the import. When it is not checked, it will skip all items where an error occurred.



Import Configuration Data from File

Look in: Translator

File name:

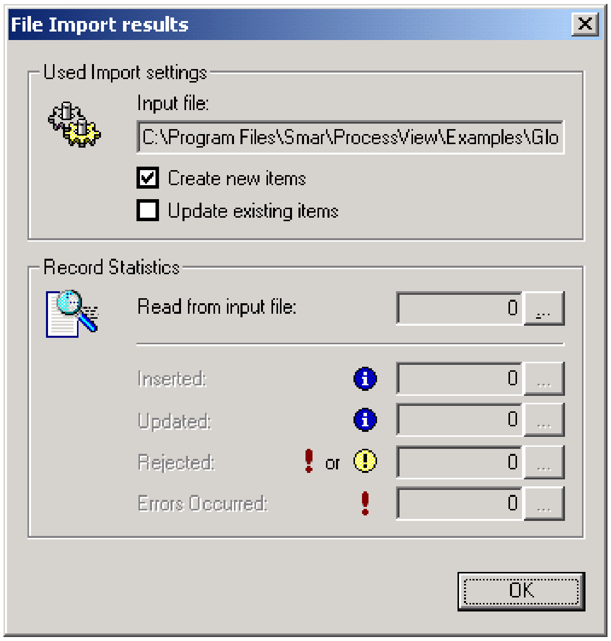
Files of type: Text files (*.csv)

Import settings
☑ Create new items
□ Update existing items
□ Display errors

Delimiters
● Tab
○ Comma
○ Other:

Import Configuration Data From File Dialog Box

When you have selected a file to import, click **Open.** When the import is completed, the **File Import Results** dialog box will open, as shown below. This shows the import settings, including the input file name. It also provides a summary of the import, including how many items were inserted, updated, or rejected, and shows how many errors occurred.



File Import results

Used Import settings

Input file: C:\Program Files\Smar\ProcessView\Examples\Glo

☑ Create new items
□ Update existing items

Record Statistics

Read from input file: 0

Inserted: 0

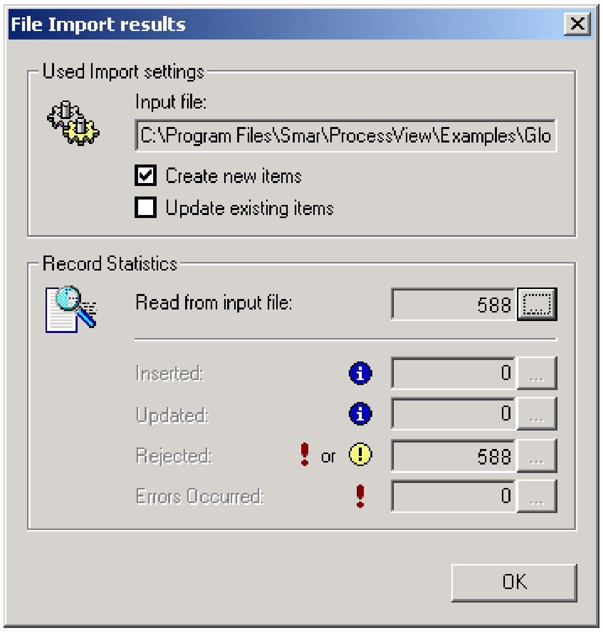
Updated: 0

Rejected: ! or ! 0

Errors Occurred: ! 0

File Import Results Dialog Box

Click the **...** button next to the right of each field to get the details view of the import results, as shown below. This view shows the specific items that were inserted, updated, or rejected, as well as a description of any errors that occurred.



File Import results

Used Import settings

Input file: C:\Program Files\Smar\ProcessView\Examples\Glo

☑ Create new items

□ Update existing items

Record Statistics

Read from input file: 588

Inserted: 0

Updated: 0

Rejected: ! or ! 588

Errors Occurred: ! 0

File Import Results: Details View

Importing Aliases and Translations From a Dictionary File

The Configurator also gives you the option of importing aliases and their language translations from a text (.txt) file or a Microsoft Excel (.csv) file "dictionary" to your configuration database. To import data from a dictionary, select **Import Dictionary** from the **File** menu. This is similar to the **Import Data** function described above.

Importing Data From an XML File

The Configurator allows you to import data from your configuration database to an XML file. The XML export/import functionality was mainly developed for Windows platforms that do not support databases (e.g. Windows CE and Windows Embedded). XML has the following advantages over the CSV import/export function:

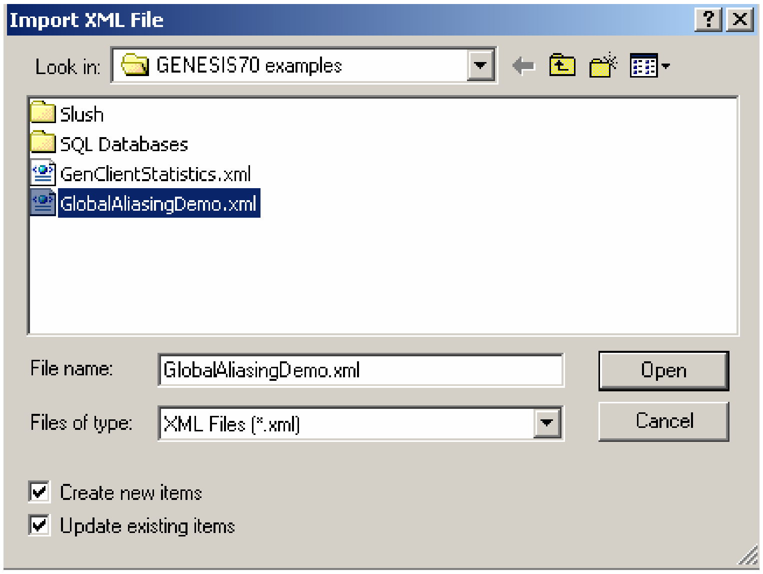
- XML has a standardized format, unlike the text/CSV format, which uses various delimiters (e.g. TAB instead of commas, strings could not accept all characters, etc.)
- XML is language-independent, whereas CSV converts date/time, floats, and currency fields according to local settings in Windows. For example, using CSV, you cannot export data on German Windows and import it on English windows without making changes
- Windows has an installed automation object that has the capability to work with XML. Thus, programmers can create/modify their configurations outside the Configurator using Visual Basic, if desired.
- XML supports schemas. A **schema** is a special XML file that specifies the data structure of an XML data file.

To import data, select **XML Import** from the **File** menu. This opens the **Import XML File** dialog box, as shown in the figure below. Give the file a name, and then choose the directory from which you want to import the data. You can then specify the delimiters and choose from the following import settings. Click **Open.**

- **Create new items.** When the import file contains items that are not yet in the configuration database, then it creates them. Otherwise it skips these items.
- **Update existing items.** When the import file contains items that are in the configuration database, then it updates them using data from the import file. Otherwise it skips these items.

Note: Either **Create new items** or **Update existing items** must be selected. Otherwise there is nothing to import.

Note: Selecting **XML Validate** from the **File** menu does not import an XML data file, but it will try to validate its structure using stored XML schema. Once it passes this validation, the XML file is acceptable for import by the Configurator.



File name: GlobalAliasingDemo.xml

Files of type: XML Files (*.xml)

☑ Create new items

☑ Update existing items

Importing Configuration Data From an XML File

Activating the Database

Once your configuration is complete, you need to make sure that it is the active database. The database that is currently active is the one that the server uses. To make the current database active, select **Make Active…** from the **File** menu. If the **Make Active…** selection is grayed out, then the current database is already the active database.

A dialog box appears showing both the current active database and database that is currently being edited. To set the edited database as the active database, click the **Yes** button.

Edit Menu

The **Edit** menu contains the following commands:

Command	*Shortcut Keys*	*Function*
New - Folder	CTRL+F	In the **Aliases** level of the tree in the left-hand pane of the Configurator screen, this command creates a new alias folder.
New - Alias	CTRL+I	In the **Aliases** level of the tree in the left-hand pane of the Configurator screen, this command creates a new alias.
New - Translation (Languages)	CTRL+T	Creates a new translation language for the selected alias.
New - Translation (Language Subsets)	CTRL+W	Creates a new translation language subset for the selected alias.
New - Language	CTRL+L	Creates a new language item.
New - Subset	CTRL+U	Creates a new language subset item.
New - Expression	CTRL+E	Creates a new expression item.
Rename	CTRL+R	Renames the selected item.
Multiply	CTRL+M	Opens the **Multiply Item** dialog box (see below), which allows you to multiply an item in the tree control.
Delete	CTRL+DEL	Deletes the selected object.
Cut	CTRL+X	Cuts the selected object from the current view and places it on the clipboard.
Copy	CTRL+C	Copies the selected object to the clipboard.
Paste	CTRL+V	Pastes the last object placed on the clipboard.
Select All	CTRL+A	Selects all objects in a list. The selection is shown in the upper-right-hand section of the Viewer.

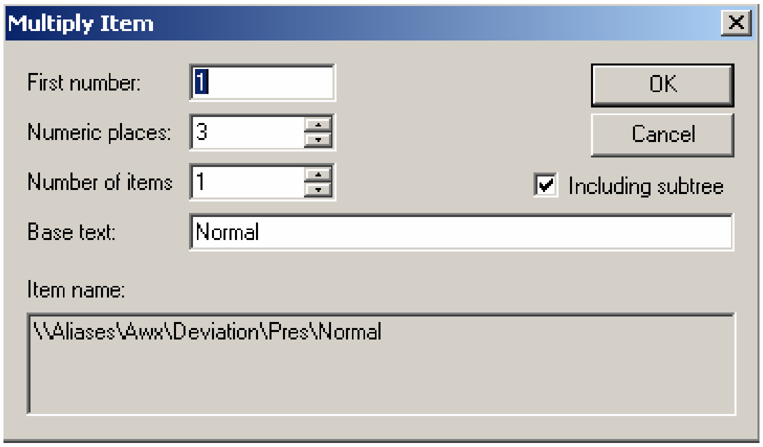
Command	Shortcut Keys	Function
Invert Selection		Unselects all selected items and selects all unselected items in a list in the upper-right-hand section of the viewer.

Multiplying Items

The Configurator allows you to multiply items in the tree control. Multiplication provides a simple way of developing configurations where there are many similar items in a given category. To multiply an item:

1. Select the item in the tree control that you wish to multiply.

2. Either right-click the item and select **Multiply** from the pop-up menu, or select **Multiply** from the **Edit** menu. This opens the **Multiply Item** dialog box, shown below.



First number: 1

Numeric places: 3

Number of items 1

☑ Including subtree

Base text: Normal

Item name:
\\Aliases\Awx\Deviation\Pres\Normal

Multiply Item Dialog Box

3. When the items are multiplied, they are all given a base name followed by a number. The default base text is the name of the item selected for multiplication. To modify the base text, change the **Base Text** field appropriately.

4. In the **First Number** field, specify the number to appear next to the first multiplied item.

5. In the **Number of Items** field, specify how many items you wish to create.

6. In the **Numeric Places** field, specify the minimum length of each number to append. Values that take up less space than the specified amount of numeric places will have zeros before the number.

7. If you want to multiply all subfolders as well, check the **Including Subtree** check box.

8. Click the **OK** button to do the multiplication. The example configuration shown in the **Multiply Item** dialog box above creates three new OPC Data folders with the following names:

- Normal001
- Normal002
- Normal003

All subfolders will also be multiplied.

View Menu

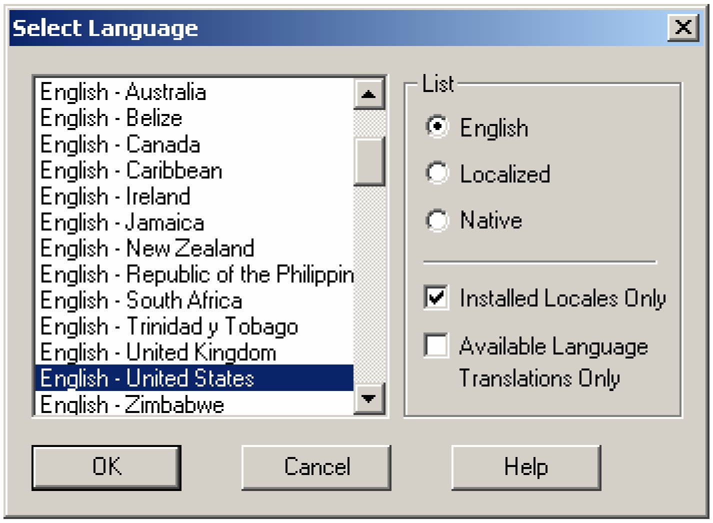
The **View** menu contains the following commands:

Command	Shortcut Keys	Function
Toolbars - Standard Buttons		Displays the **Standard toolbar.**
Toolbars - Data-Manipulation Buttons		Displays the **Data-Manipulation toolbar.**
Status Bar		Toggles the status bar.
Large Icons	F7	Displays items as large icons.
Small Icons	F8	Displays items as small icons.
List	F9	Displays items as a list.
Details	F10	Displays items as a list along with detailed information about the configuration of each item.
Dialog View	F11	Toggles the configuration window (right-hand pane).
Sort By		Displays a list of options for sorting the columns in the right-hand pane of the screen. The options listed depend on the level within the view.
Show/Hide Columns		Displays a list of options that you can choose to show or hide in the view.
Select Language		Opens the **Select Language** dialog box (see below). Choose the language you wish to use for your system (Unicode version only) and click **OK.** For navigation purposes, use the buttons and check boxes in the **List** section.
Global Refresh	F5	Refreshes the data for the entire Configurator screen.
Subtree Refresh	CTRL+F5	Refreshes only the data contained in the currently selected subtree.

Selecting Languages

The **Select Language** function on the **View** menu allows you to choose which language to use in your display. Choosing **Select Language** from the **View** menu opens the **Select Language** dialog box, shown in the figure below.

Note: A language resource .dll is required for language switching.



Select Language Dialog Box

Define the parameters listed in the table below. Then click **OK** to return to the work area.

Select Language Parameters

Parameter	Description
List	Lists available languages. Depending on which item you have selected, the view on the left will change. If **English** is checked, the languages will appear as their English name. If **Localized** is checked, the languages will appear with the native country in parentheses (for languages with several dialects only). When **Native** is checked, the languages are displayed the way they would be written in that language.
Installed Locales Only	If this is checked, local languages appear in the box.
Available Language Translations Only	Checking this box allows you to choose from available language translations only.

Sorting Items

The **Sort By** and **Show/Hide Columns** commands in the **View** menu are specific to each type of item available and allow you to change how each item and its attributes are displayed.

For **Aliases,** the following options are available:

Sort by:

- Name
- Description
- Default Translation
- Default Expression
- Expression Format

Show/hide columns:

- Description
- Default Translation
- Default Expression
- Expression Format

For a **specific alias,** the following options are available:

Sort by:

- Language Name
- Translation
- Expression
- Format
- Font Name

Show/hide columns:

- Translation
- Expression
- Format
- Font Name

For **Languages,** the following options are available:

Sort by:

- Name
- Description
- LCID
- Language Name
- Activated

Show/hide columns:

- Description
- LCID
- Language Name
- Activated

For a **specific language,** the following options are available:

Sort by:

- Name
- Description

Show/hide columns:

- Description

For **Expressions,** the following options are available:

Sort by:

- Name
- Description
- Read Expression
- Write Expression

Show/hide columns:

- Description
- Read Expression
- Write Expression

Go Menu

The **Go** menu contains the following commands:

Command	Shortcut Keys	Function
Back	CTRL+ALT+Left Arrow	Moves the cursor back to the previously selected item in the tree control.
Forward	CTRL+ALT+Right Arrow	Moves the cursor forward to the previously selected item in the tree control.
Up One Level		Moves the cursor up one level in the tree control.
Next Item	ALT+Down Arrow	Moves the cursor to the next item down in the tree control.
Previous Item	ALT+Up Arrow	Moves the cursor to the next item up in the tree view.
Expand Item	ALT+Left Arrow	Expands an item that contains a submenu.
Collapse Item	ALT+Right Arrow	Collapses an item that contains a submenu.
Page Up	ALT+PgUp	Moves the cursor up to the first item in the tree.
Page Down	ALT+PgDown	Moves the cursor down to the last visible item in the tree.
Home	ALT+Home	Moves the cursor up to the first item in the tree.
End	ALT+End	Moves the cursor down to the last visible item in the tree.
Previous Language Translation	CTRL+Up Arrow	When a translation is selected and the previous alias is available in the current level, this function moves the cursor to the translation of the same language under the previous language alias.
Next Language Translation	CTRL+Down Arrow	When a translation is selected and the next alias is available in the current level, this function moves the cursor to the translation of the same language under the next language alias.
Aliases	F2	Moves the cursor to the **Aliases** item in the tree.
Languages	F3	Moves the cursor to the **Languages** item in the tree.
Expressions	F4	Moves the cursor to the **Expressions** item in the tree.
Next Pane	F6	Moves the cursor to the next pane.
Previous Pane	SHIFT+F6	Moves the cursor to the last pane used.

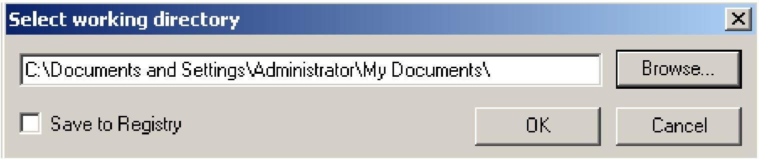
Tools Menu

The **Tools** menu contains the following commands:

Command	Function
Set Working Directory	Sets a working directory for all files relating to the saved configuration.
Options	Launches the **Options** dialog box, which is described below in detail.
Compact/Repair MS Access Database	Opens the **Compact/Repair MS Access Database** dialog box, which is described below in detail.

Setting the Working Directory

Selecting **Set Working Directory** from the **Tools** menu opens the **Set Working Directory** dialog box, shown below, which enables you to configure a custom directory in which all application configuration files will be stored and retrieved. Click **Browse** to select the directory.



C:\Documents and Settings\Administrator\My Documents\

□ Save to Registry

Set Working Directory Dialog Box

Options

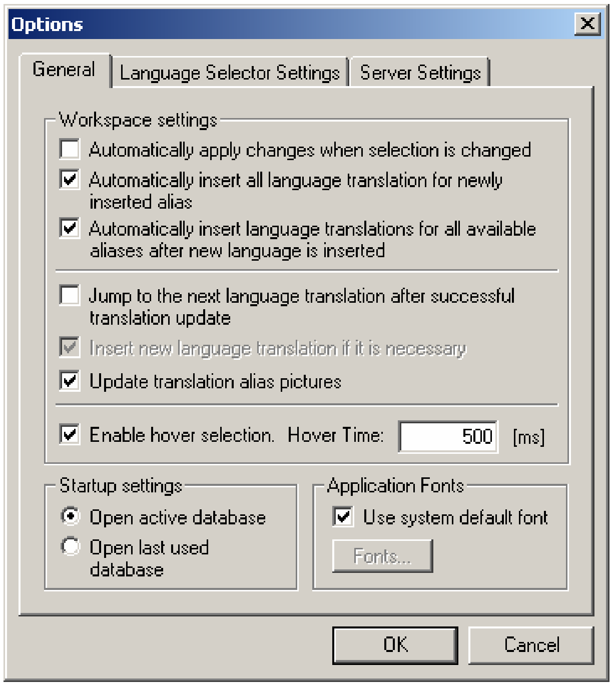
Selecting **Options** from the **Tools** menu opens the **Options** dialog box, which contains the following tabs:

- General
- Language selector settings
- Server settings

General Tab

The **General** tab of the **Options** dialog box, shown below, contains the following fields:

- Workspace settings
- Startup settings
- Application fonts



Workspace settings

□ Automatically apply changes when selection is changed

☑ Automatically insert all language translation for newly inserted alias

☑ Automatically insert language translations for all available aliases after new language is inserted

□ Jump to the next language translation after successful translation update

☑ Insert new language translation if it is necessary

☑ Update translation alias pictures

☑ Enable hover selection. Hover Time: 500 [ms]

Startup settings

● Open active database

○ Open last used database

Application Fonts

☑ Use system default font

Options Dialog Box: General Tab

Workspace settings. This field contains the following options:

- **Automatically apply changes when selection is changed:** This allows changes to be saved each time you switch between dialog boxes. When this option is checked, you need not click the **Apply** button, nor will you be shown a message asking if you would like to apply changes.
- **Automatically insert all language translations for newly inserted alias:** This adds all language translations when creating a new alias. You will still need to add any language subset translations manually.
- **Automatically insert language translations for all available aliases after new language is inserted:** After you create a new language, it will automatically be added to all aliases.
- **Jump to the next language translation after successful translation update:** This automatically opens the configuration dialog box for the next language translation when configuration for the current language translation is finished.
- **Insert new language translation if it is necessary:** This will take effect when **Automatically insert language translations….** is checked. It inserts new translation record under the next alias (if it missing).
- **Update translation alias pictures (icons):** Each alias can have more than one icon associated with it. These icons are updated according to translation states. For example, when a language translation is defined and the translation text is missing, or at least one of the languages is not translated, then the alias icon is red exclamation point (**!**). Because this alias icon update feature tests all the translation items for every alias to get the status, it can slow down browsing in the Configurator. Thus, this option allows you to switch off these updates. When this option is unchecked, each alias is indicated only by the blue pound sign.
- **Enable hover selection:** This allows you to highlight an item by moving the mouse pointer over that item and keeping it there for a specified amount of time (in milliseconds).

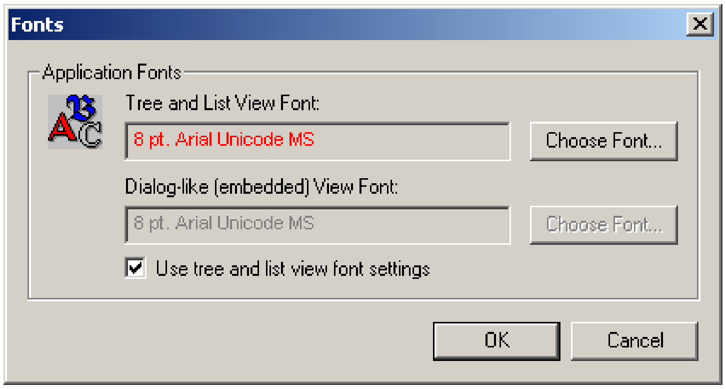
Startup settings. This field contains the following options:

- **Open active database:** This opens the database that is currently active when starting the Language Configurator.
- **Open last-used database:** This opens the last database that was opened when starting the Language Configurator.

Application fonts. This field contains the following options:

- **Use system default font:** When this option is checked, the system default font is used. To select a different font, uncheck this box to enable the **Fonts** button.
- **Fonts button:** This button allows you to choose an application font other than the system default font. When you click **Fonts,** the fonts dialog box opens, as shown below. If the font name is red, the font either has not been installed or has been improperly installed and therefore cannot be used. By clicking **Choose Font,** you can select different fonts (or font sizes) for the tree and list views as well as dialog boxes.

Note: To enhance language-aliasing performance, select the Microsoft Arial Unicode font, which contains all Unicode characters. When you begin configuring in GraphWorX, you should first select **Application Preferences** from the **Format** menu in GraphWorX. Click the **Font** button in the **Grid** tab of the **Application Preferences** dialog box and select the Arial Unicode MS font. The same applies to all ActiveX components, such as TrendWorX and AlarmWorX. The Arial Unicode MS font must be selected separately within the Properties dialog boxes of each ActiveX component.



Tree and List View Font: 8 pt. Arial Unicode MS

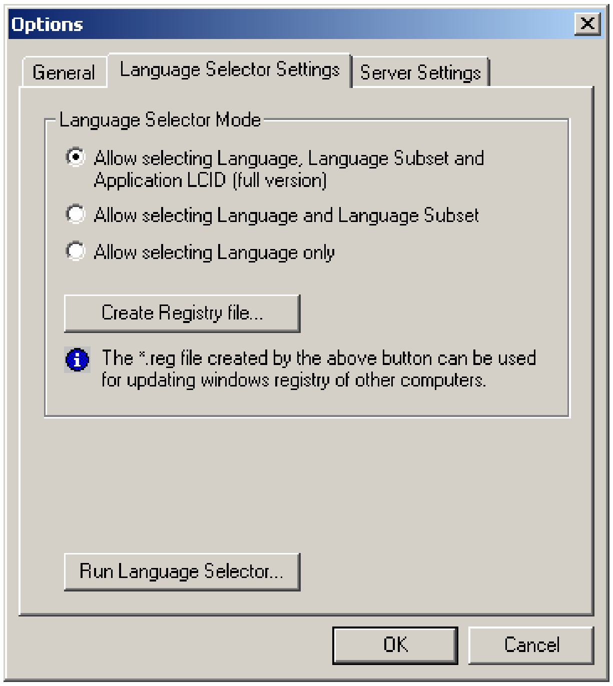
Dialog-like (embedded) View Font: 8 pt. Arial Unicode MS

☑ Use tree and list view font settings

Fonts Dialog Box

Language Selector Settings Tab

The **Language Selector Settings** tab of the **Options** dialog box, shown below, allows you to set the **Language Selector** mode.



Language Selector Mode

● Allow selecting Language, Language Subset and Application LCID (full version)

○ Allow selecting Language and Language Subset

○ Allow selecting Language only

The *.reg file created by the above button can be used for updating windows registry of other computers.

Options Dialog Box: Language Selector Settings Tab

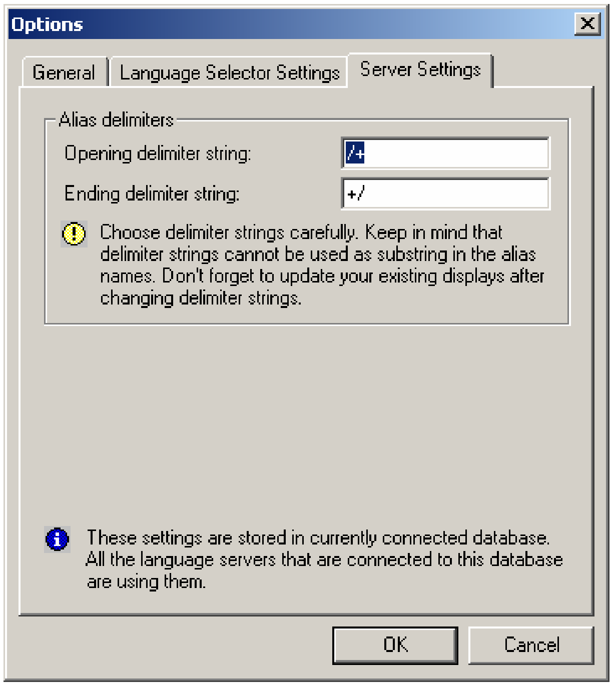
The **Language Selector Mode** field contains the following options for the **Language Selector** dialog box.

- **Allow selecting Language, Language Subset, and Application LCID:** Enables all fields in the Language Selector dialog box.
- **Allow selecting Language and Language Subset:** Enables only the **Language** and **Language Subset** fields in the **Language Selector** dialog box.
- **Allow selecting Language:** Enables only the **Language** field in the Language Selector dialog box.
- **Create Registry File:** Clicking this button enables you create to create a .reg file according to the current selector mode, which can be used in updating the Windows registry of other computers.

Once you have selected the mode, you can open the **Language Selector** dialog box by clicking **Run Language Selector.**

Server Settings Tab

The **Server Settings** tab of the **Options** dialog box, shown below, contains options for alias delimiters, which are the symbols that used to mark your display configuration as an alias.



Opening delimiter string: /+

Ending delimiter string: +/

Choose delimiter strings carefully. Keep in mind that delimiter strings cannot be used as substring in the alias names. Don't forget to update your existing displays after changing delimiter strings.

These settings are stored in currently connected database. All the language servers that are connected to this database are using them.

Options Dialog Box: Server Settings Tab

Opening delimiter string: Enter the character(s) you wish to use as the opening delimiter.

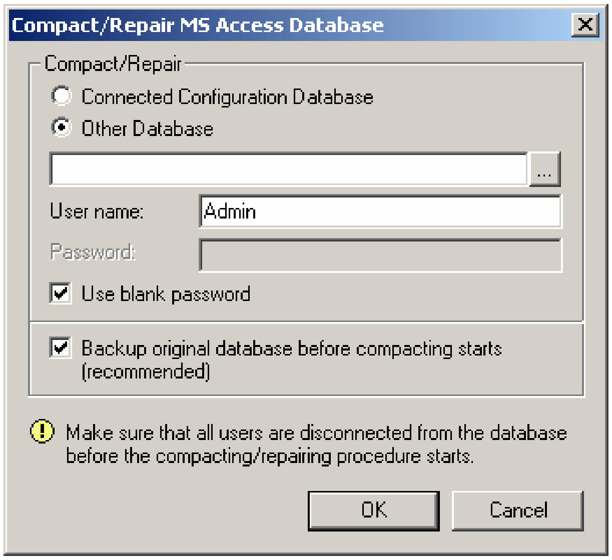
Ending delimiter string: Enter the character(s) you wish to use as the ending delimiter.

Note: Choose delimiter strings carefully because delimiter strings cannot be used as substrings in the alias names. Do not forget to update your existing displays after changing delimiter strings.

Compacting and Repairing MS Access Databases

You can compact Microsoft Access databases, which can be either configuration databases or historical databases, using the **Compact/Repair MS Access Database** dialog box, shown in the figure below. To open this dialog box, select **Compact/Repair MS Access Database** from the **Tools** menu. Microsoft Access–based databases are subject to database fragmentation over time, and the support for the database will compact the target database, reclaim unused space, and drastically improve database performance.

Note: It is critical that no users or client applications are connected to the database at the time of compacting and that, if the **Backup Original Database** option is selected, there is plenty of available hard disk space.



Compact/Repair MS Access Database

Compact/Repair

○ Connected Configuration Database

● Other Database

User name: Admin

Password:

☑ Use blank password

☑ Backup original database before compacting starts (recommended)

Make sure that all users are disconnected from the database before the compacting/repairing procedure starts.

Compact/Repair MS Access Database Dialog Box

Help Menu

The **Help** menu contains the following commands:

Command	Shortcut Key	Function
Help Topics	F1	Launches the online Help for the Configurator.
About Application		Launches the **SMAR About Box,** which contains information about the product version number, copyright, and available disk space. It also contains information about how to contact SMAR.

Basic Steps for Language Configuration

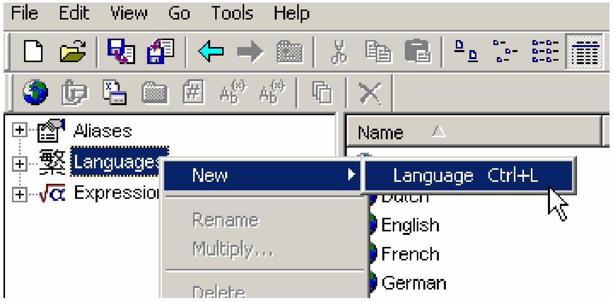
The following procedure describes the basic steps for creating a language database.

1. Add all languages you would like to use.
2. Add any expressions you would like to use.
3. Create aliases for all text and expressions you would like to convert.
4. Make the current database active.

Adding a New Language

The Language Configurator provides Czech, Dutch, English, French, German, Italian, Russian, and Spanish as default languages. To add a new language to your configuration database:

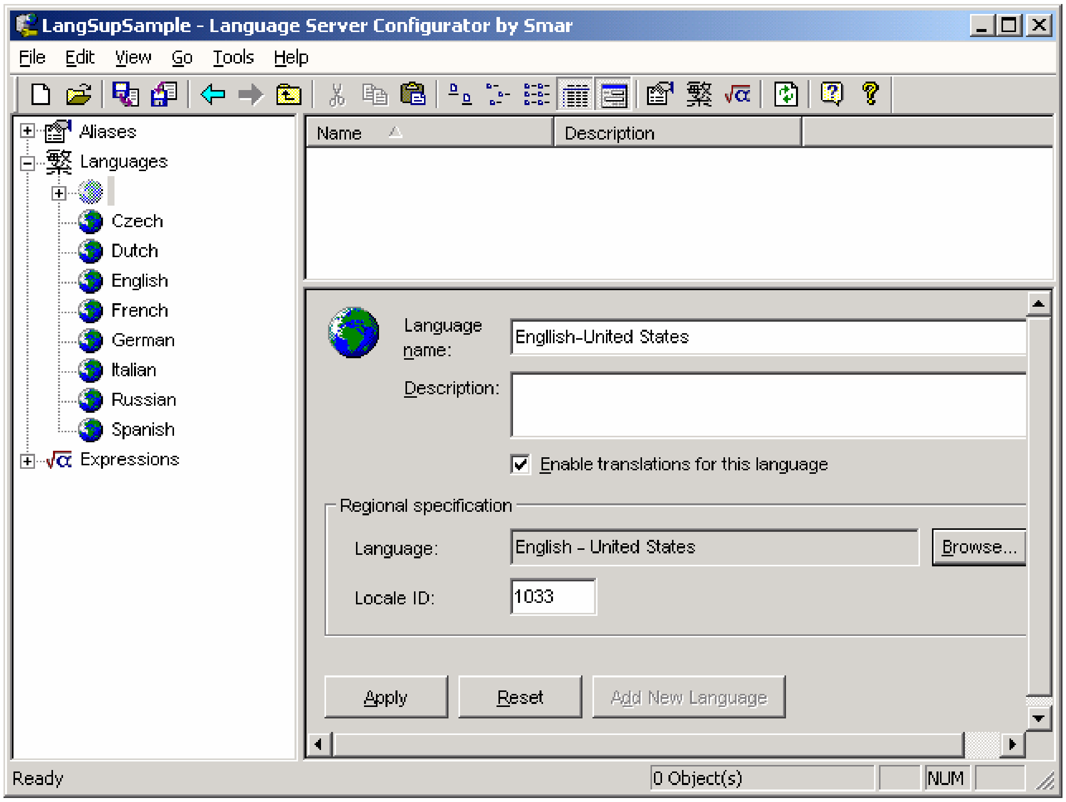
1. Click the **New 'Regional Specification'** button on the **Data-Manipulation toolbar.** Or right-click **Languages** on the tree view of the left-hand pane of the Configurator screen, select **New > Language** from the pop-up menu, as shown in the figure below. You can also press the shortcut keys **CTRL+L.**



Adding a New Language

2. This opens the **Language Configuration** dialog box in the right-hand pane of the Configurator screen, as shown in the figure below. Click **Browse** and select the language you would like to add. The language you have selected will automatically be displayed in the **Language Name** field. You may change this name if you wish. The **Locale ID** that corresponds with the selected language will also be displayed automatically. You can include additional comments in the **Description** field. Make sure that the **Enable Translations for This Language** check box is checked.

3. Click **Apply**.



Language name: Engllish-United States

Description:

☑ Enable translations for this language

Language: English - United States

Locale ID: 1033

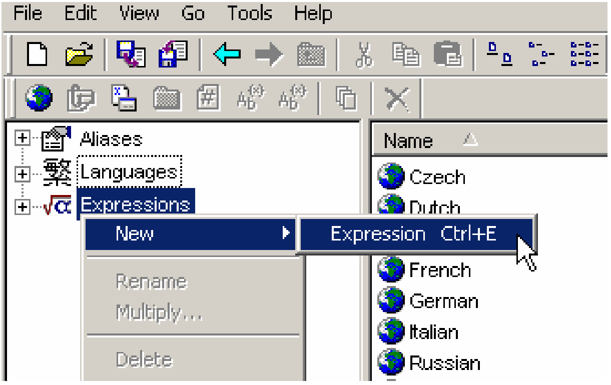
0 Object(s)

Language Properties

Adding a New Expression

Expressions enable you to make unit conversions as well as number-formatting conversions. To add a new expression to your configuration database:

1. Click the **New 'Expression'** button on the **Data-Manipulation toolbar.** Or right-click **Expressions** on the tree view of the left-hand pane of the Configurator screen, select **New > Expression** from the pop-up menu, as shown in the figure below. You can also press the shortcut keys **CTRL+E.**



Adding a New Expression

2. This opens the **Expression Configuration** dialog box in the right-hand pane of the Configurator screen, as shown in the figure below. Enter a name for this expression in the **Expression Name** field. You can include additional comments in the **Description** field.
3. Add the **Read** and **Write** Expressions in the **Expression Parameters** field. To use the **Expression Editor,** click **Exp** to open the **Edit Expression** dialog box. For example, to have an expression that converts inches to centimeters you would type the following:

Read Expression: x={{x}}*2.54

Write Expression: x={{x}}/2.54

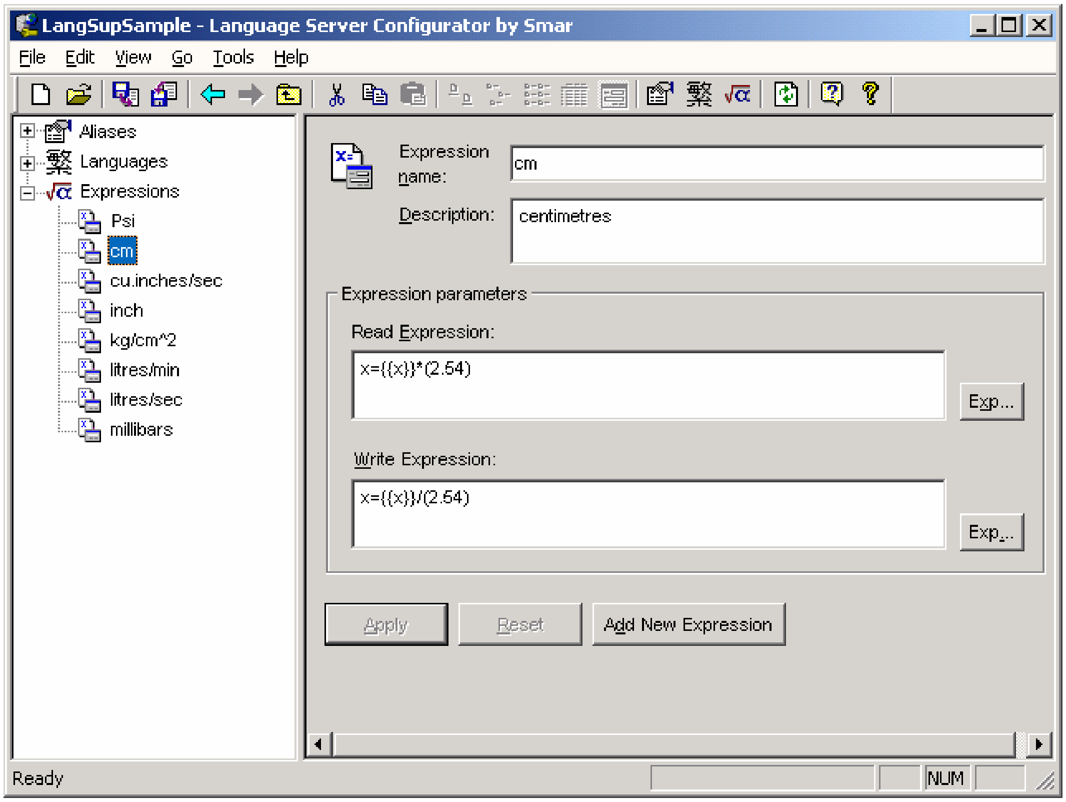
If *inch* were the default unit, you would type the following for inches:

Read Expression: x={{x}}

Write Expression: x={{x}}

4. Click **Apply.**

Note: For more information about expressions, please see the **Expression Editor** documentation**.**



Expression name: cm

Description: centimetres

Expression parameters

Read Expression: x={{x}}*(2.54)

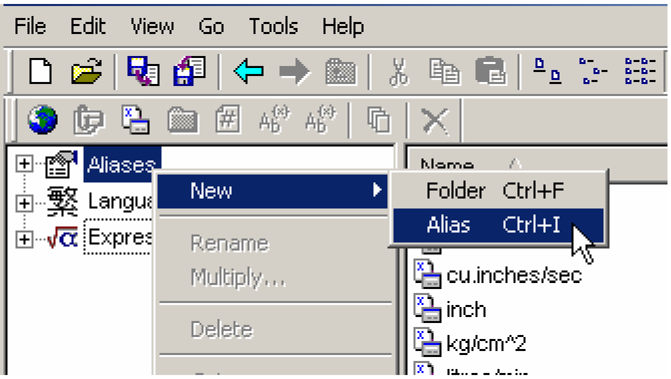
Write Expression: x={{x}}/(2.54)

Expression Properties

Adding a New Alias

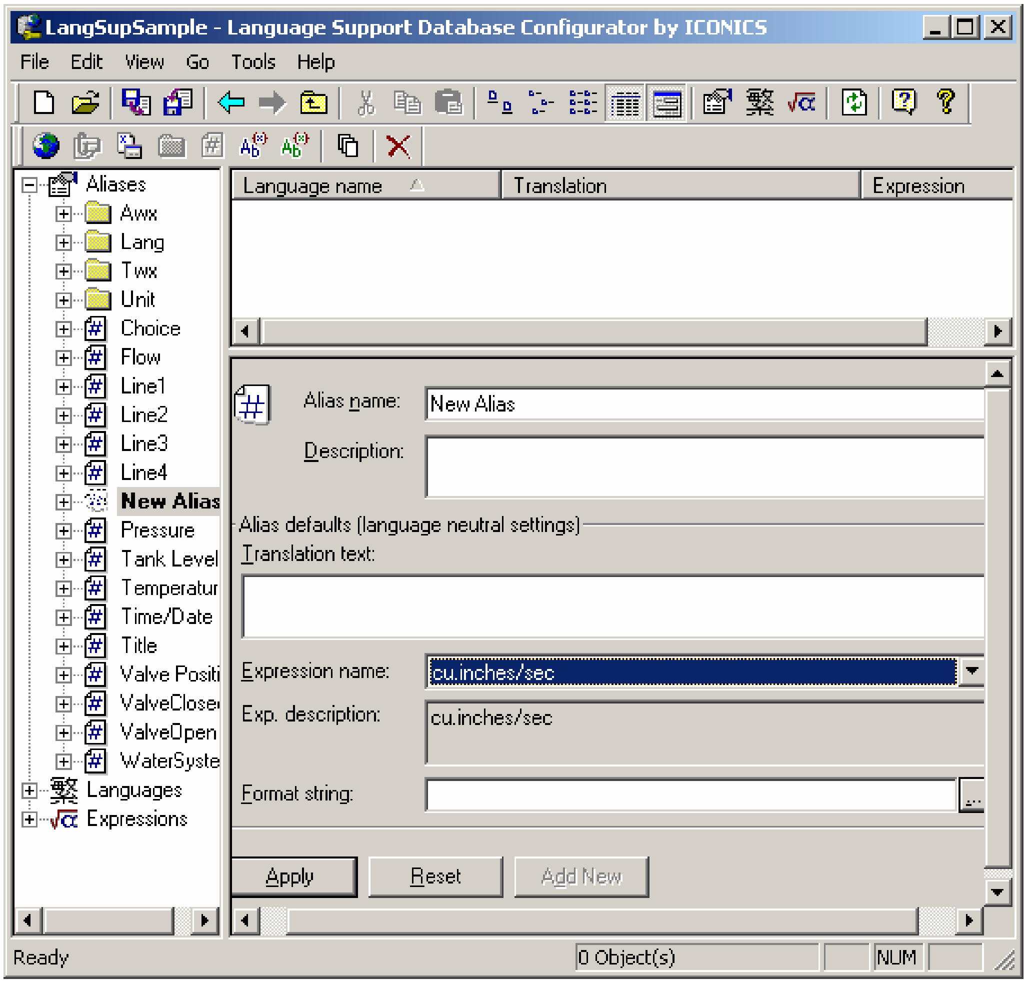
You must add a new Alias for every item you wish to convert. To add a new Alias to your configuration database, use the following procedure:

1. Click the **New 'Alias'** button on the **Data-Manipulation toolbar.** Or right-click **Aliases** on the tree view of the left-hand pane of the Configurator screen, select **New > Alias** from the pop-up menu, as shown in the figure below. You can also press the shortcut keys **CTRL+I.**



Adding a New Alias

2. This opens the **Alias Configuration** dialog box in the right-hand pane of the Configurator screen, as shown in the figure below. Enter a name for the alias in the **Alias Name** field. This is the string that must appear between the alias delimiters in configurations for displays. Note that you can copy the string from your display and paste it into this field. You can include additional comments in the **Description** field.
3. Enter the default information for the new alias in the **Alias Defaults** field.
   - **Translation text.** Enter the text that will appear as the default text.
   - **Expression name.** From the list of expressions, choose the expression that will be the default expression. The list will include all expressions you have created.
   - **Exp. description.** This field will automatically include any additional comments you added in the **Description** when you created the expression.
   - **Format string.** Click the button at right to open the **Format String Editor** dialog box, which enables you to choose how to format the string.
4. Click **Apply.**
5. You must now add a new language translation for each language or language subset you wish to use with this Alias. For more information, see **Configuring a Language Translation.**



Alias name: New Alias

Description:

Translation text:

Expression name: cu.inches/sec

Exp. description: cu.inches/sec

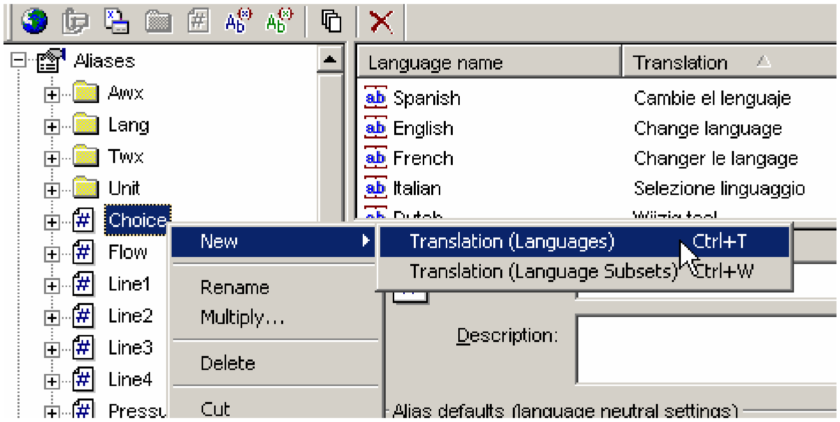
Format string:

Alias Properties

Configuring a Language Translation

You must configure a language translation for each language under each alias you wish to convert. To configure a language translation, use the following procedure:

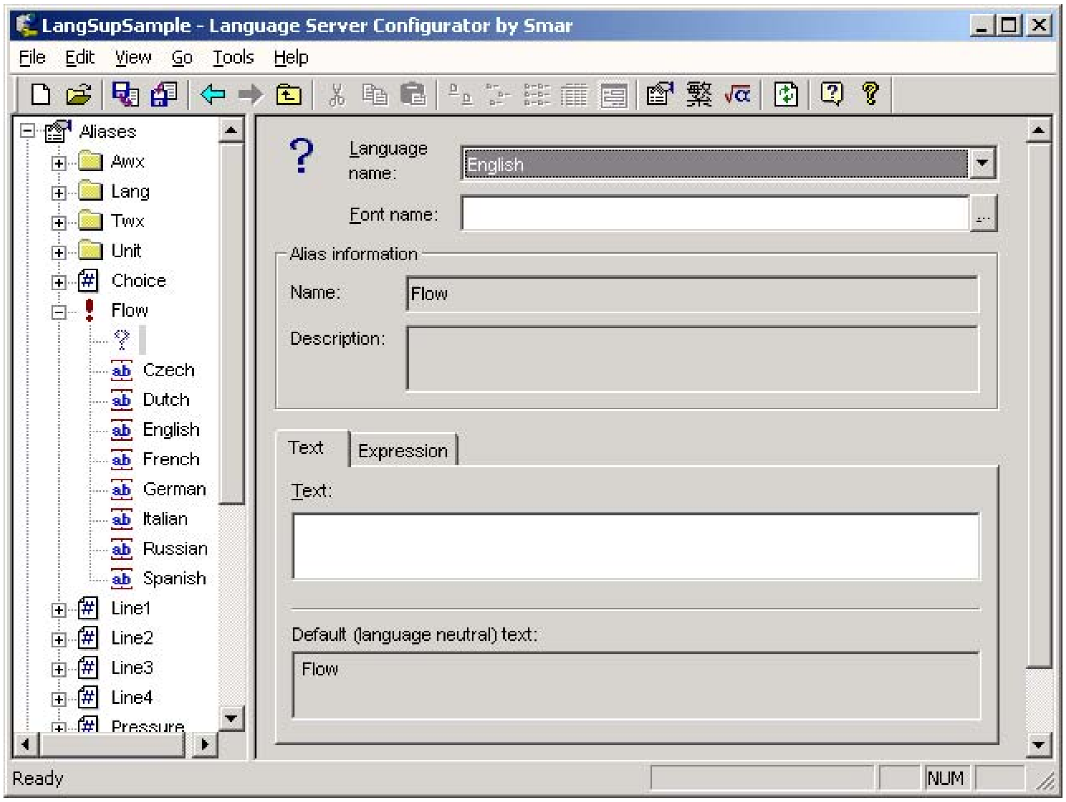
1. Select the alias name, and then click the **New 'Translation (Languages)'** button on the **Data-Manipulation toolbar.** Or right-click the alias name on the tree view of the left-hand pane of the Configurator screen, select **New > Translation (Languages)** from the pop-up menu, as shown in the figure below. You can also press the shortcut keys **CTRL+T.**


Aliases
Awx
Lang
Twx
Unit
Choice
Flow
Line1
Line2
Line3
Line4
Pressu
New
Rename
Multiply…
Delete
Cut
Translation (Languages) Ctrl+T
Translation (Language Subsets) Ctrl+W
Description:
Alias defaults (language neutral settings)

Language name	Translation
Spanish	Cambie el lenguaje
English	Change language
French	Changer le langage
Italian	Selezione linguaggio
Dutch	Wijzig taal

Creating a New Language Translation

2. This opens the **Language Translation** dialog box in the right-hand pane of the Configurator screen, as shown in the figure below. Choose a language from the list in the **Language Name** field, which will include all languages you have added.
3. If you wish to apply a font other than the default, click the button to the right of the **Font Name** field. This opens the **Select Font** dialog box, which enables you to choose a font.
4. The **Alias Information** field will display the name of the alias for which you are creating translation and any description you added to that alias.



Language name: English

Font name:

Alias information

Name: Flow

Description:

Text:

Default (language neutral) text: Flow

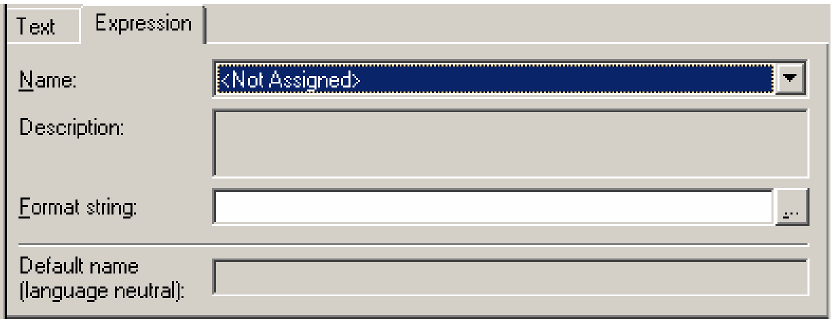
Configuring a Language Translation

Text Translation

For a text translation, enter the appropriate text in the language you are configuring In the **Text** field of the **Text** tab, shown above. This is what will appear when the language is converted. If you set up a default text under your alias configuration, it will appear in the **Default Text** field. Click **Apply** when you are finished.

Expression Translation

For an expression translation, choose from the list of expression names in the **Name** field of the **Expression** tab, shown in the figure below. The list will include all expressions you have created. If you added a description with the expression, it will appear in the **Description** field. The **Format String** field allows you to set the format for the string to be displayed. If you set up a default expression under your alias configuration, it will appear in the **Default Name** field. Click **Apply** when you are finished.



Text | Expression

Name: <Not Assigned>

Description:

Format string:

Default name (language neutral):

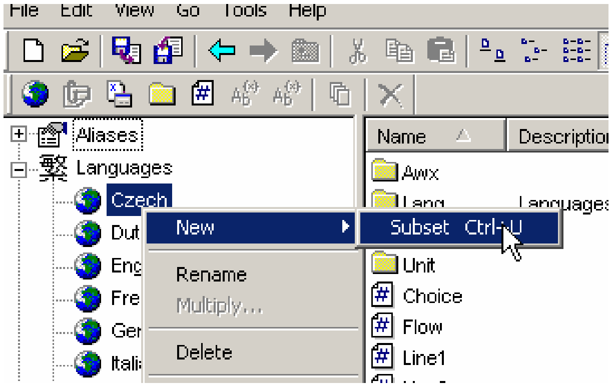
Expression Translation

Note: If you will be applying a language subset, see **Creating Language Subsets.**

Creating Language Subsets

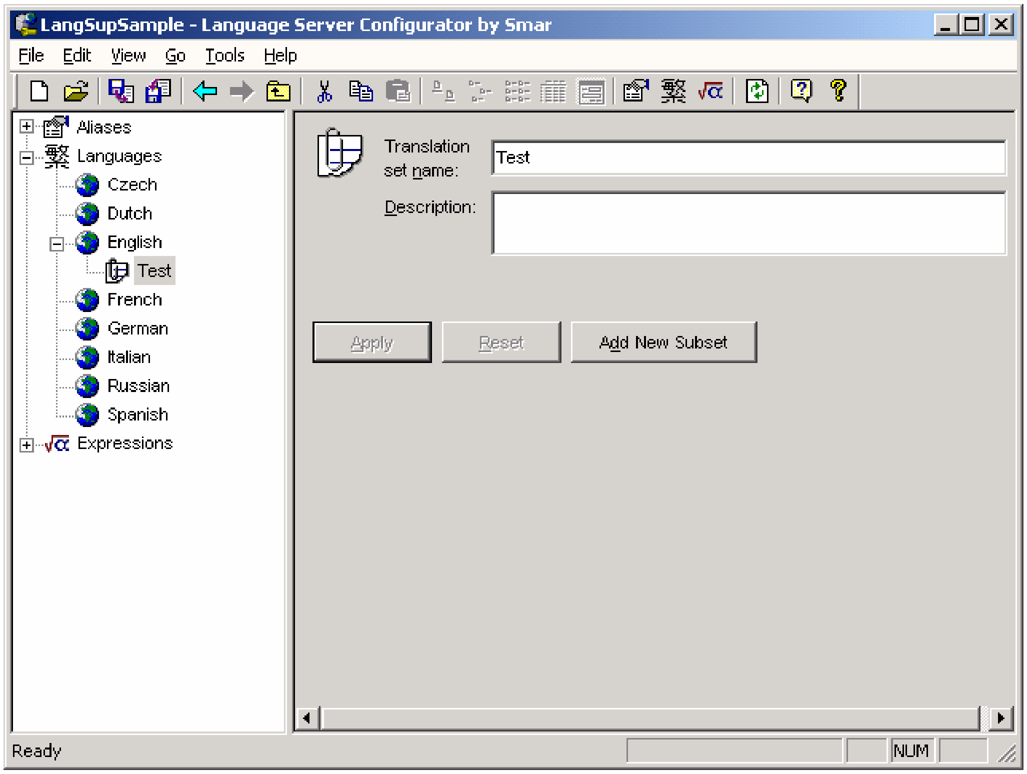
Language subsets allow you to create an additional configuration to be used with a language. You can apply a language with a translation other than the default for that language. The purpose of language subsets is to provide different measurement systems for a selected language translation. For example, if you have added the language "English," you can apply metric conversions as well as US standard conversions to that language. To create a language subset, use the following procedure:

1. Select the language, and then click the **New 'Subsets'** button on the **Data-Manipulation toolbar.** Or right-click the language on the tree view of the left-hand pane of the Configurator screen, select **New > Subset** from the pop-up menu, as shown in the figure below. You can also press the shortcut keys **CTRL+U.**



Creating a New Language Subset

2. This opens the **Language Subsets** dialog box in the right-hand pane of the Configurator screen, as shown in the figure below. Enter a name in the **Translation Set Name** field. You can include additional comments in the **Description** field.
3. Click **Apply.**



Translation set name: Test

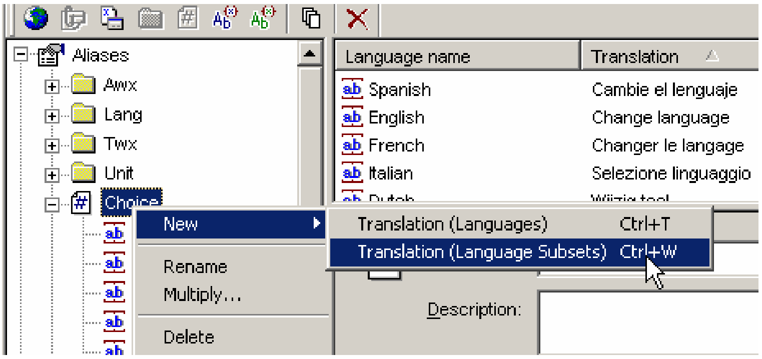
Description:

Configuring a Language Subset

Applying a Language Subset to an Alias

To apply a language subset to an alias, use the following procedure:

1. Select the alias name, and then click the **New 'Translation (Language Subsets)'** button on the **Data-Manipulation toolbar.** Or right-click the alias name on the tree view of the left-hand pane of the Configurator screen, select **New > Translation (Language Subsets)** from the pop-up menu, as shown in the figure below. You can also press the shortcut keys **CTRL+W.**

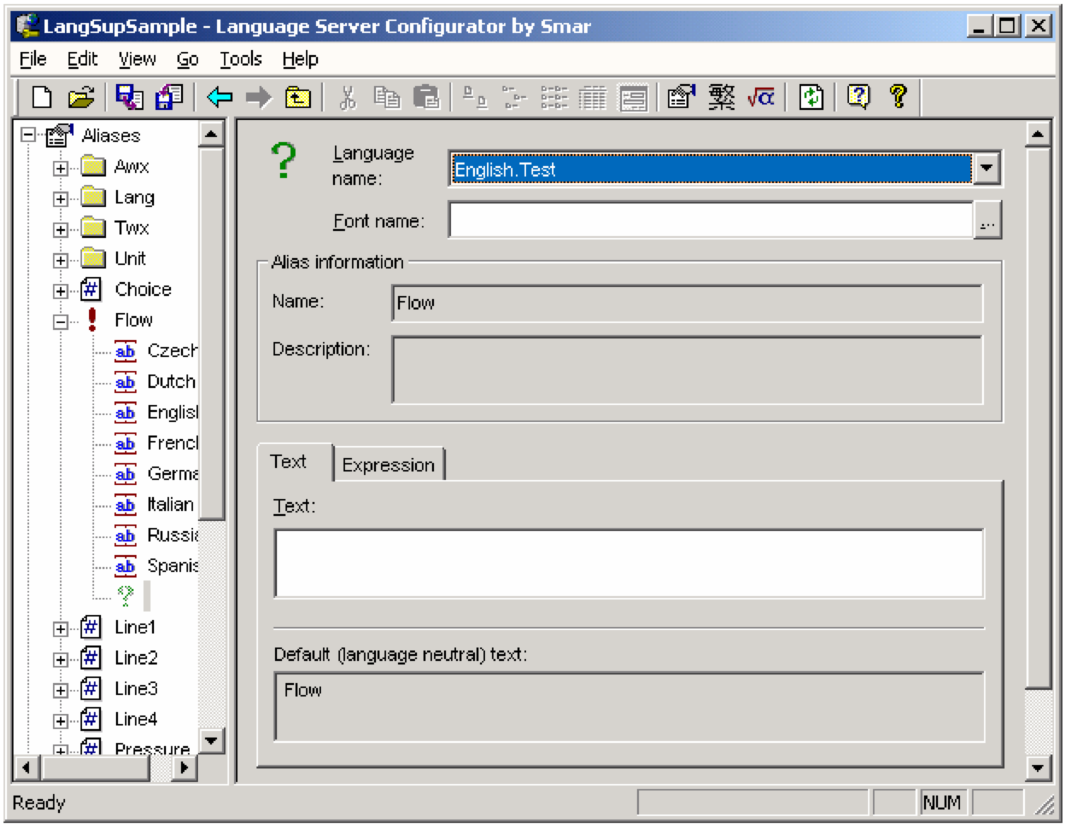

Aliases
Awx
Lang
Twx
Unit
Choice
New
Rename
Multiply...
Delete
Translation (Languages) Ctrl+T
Translation (Language Subsets) Ctrl+W

Language name	Translation
Spanish	Cambie el lenguaje
English	Change language
French	Changer le langage
Italian	Selezione linguaggio
Dutch	Wijzig taal

Description:

Applying a Language Subset to an Alias

2. This opens the **Language Subsets Translation** dialog box in the right-hand pane of the Configurator screen, as shown in the figure below. In the **Language Name** field, choose from the list of language names, which will include all language subsets you have created.
3. If you wish to apply a font other than the default, click the button to the right of the **Font Name** field. This opens the **Select Font** dialog box, which enables you to choose a font.
4. The **Alias Information** field will display the name of the alias for which you are creating translation and any description you added to that alias.



Language name: English.Test

Font name:

Alias information

Name: Flow

Description:

Text:

Default (language neutral) text: Flow

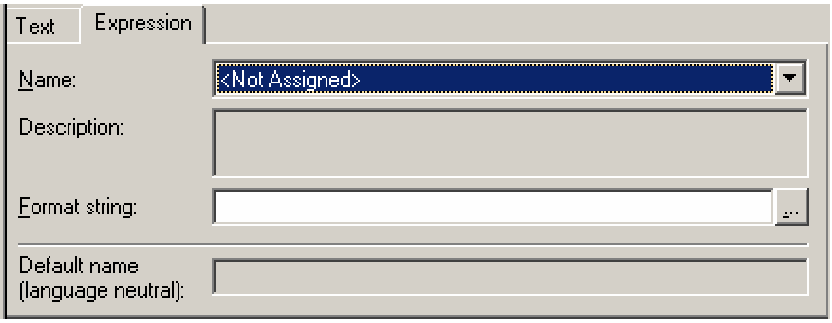
Applying a Language Subset to an Alias

Text Translation

For a text translation, enter the appropriate text in the language you are configuring In the **Text** field of the **Text** tab, shown above. This is what will appear when the language is converted. If you set up a default text under your alias configuration, it will appear in the **Default Text** field. Click **Apply** when you are finished.

Expression Translation

For an expression translation, choose from the list of expression names in the **Name** field of the **Expression** tab, shown in the figure below. The list will include all expressions you have created. If you added a description with the expression, it will appear in the **Description** field. The **Format String** field allows you to set the format for the string to be displayed. If you set up a default expression under your alias configuration, it will appear in the **Default Name** field. Click **Apply** when you are finished.



Name: <Not Assigned>

Description:

Format string:

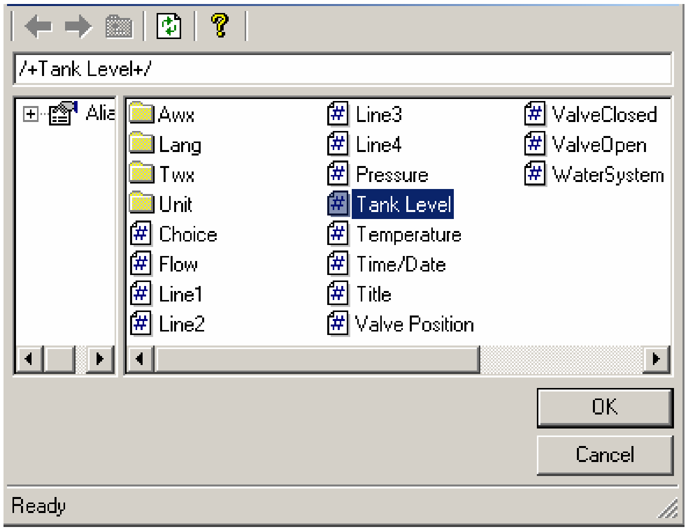
Default name (language neutral):

Expression Translation

Using the Language Alias Browser

Throughout the ProcessView client applications, such as GraphWorX, TrendWorX, and AlarmWorX, you can use language aliases in certain data fields (e.g. process points and trend pens). To select language aliases, you can open the **Language Alias Browser,** shown in the figure below. The Language Alias Browser includes all language aliases in the active language aliasing configuration database. This eliminates the need to manually type in the alias names. All language aliases that are configured in the Language Aliasing Configurator are conveniently available to choose from inside the browser. The tree control of the Language Aliasing Configurator is mimicked in the tree control of the Language Alias Browser.

Language aliases use the following syntax: **/+language_alias_name+/.** Select a global alias by double-clicking the alias name (e.g. "Valve Position" in the figure below). The alias name /+Valve Position+/ appears at the top of the browser, which automatically adds the /+ and +/ delimiters to the alias name. Click the **OK** button.



/+Tank Level+/

Selecting an Alias From the Language Alias Browser

Language Server

The Language COM Server is the heart of the Language-Switching utility. It services requests from the connected client programs. The main purpose of the language server is to find translation parameters for given alias names. The language server configuration data (translation parameters, alias names, and supported languages) are stored in a database (Microsoft Access, MSDE or Microsoft SQL Server).

The language server is designed as an out-of-process COM object that supports two kinds of interfaces: Custom interface and Dual interface.

OLE Automation Objects and Interfaces

Type Library

VB uses the LangServer Type Library to define the following interfaces. Make sure that (in Visual Basic) "Project | References" has "SMAR Language Tool 1.0 Type Library" checked.

LangServer Object

Description	A language server client creates the LangServer automation object. The LangServer object can be used to obtain language translations and general information about a language server.
Syntax	LangServer
Example	Dim ALangServer As LangServer Set ALangServer = New LangServer

Summary of Properties

AliasStartDelimiter	ConfigurationDSrc	LoggingEnabled
AliasStopDelimiter	LoggerDSrc	VariableText

Summary of Methods

GetTranslationText	GetTranslationRecordset	GetLocaleID

LangServer Properties

AliasStartDelimiter

Description	(Read-only) Returns the opening delimiter string for translation alias. The LangServer object uses this string to identify the beginning of the alias string in an input string.
Syntax	AliasStartDelimiter As String
Remarks	This property, as well as other properties, is stored in the language-configuration database. An error occurs if the server has not been able to connect to the configuration database.
Example	Dim AnAliasStartDelimiter As String AnAliasStartDelimiter = AnLangServer.AliasStartDelimiter

AliasStopDelimiter

Description	(Read-only) Returns the ending delimiter string for translation alias. The LangServer object uses this string to find out the end of the alias string in an input string.
Syntax	AliasStopDelimiter As String
Remarks	This property, as well as other properties, is stored in the language-configuration database. An error occurs if the server has not been able to connect to the configuration database.
Example	Dim AnAliasStopDelimiter As String AnAliasStopDelimiter = AnLangServer.AliasStopDelimiter

ConfigurationDSrc

Description	(Read-only) Returns the file path that LangServer object is using to connect to the configuration database. It can be the file path to a MS Access database file (extension .mdb) or to a MS data link file (extension .udl). Data link file is a text file that contains an OLE DB connection string.
Syntax	ConfigurationDSrc As String
Remarks	This property is stored in the computer registry database. No errors should occur.
Example	Dim LangServerConfigurationDataSource As String LangServerConfigurationDataSource = AnLangServer.ConfigurationDSrc

LoggerDSrc

Description	(Read-only) Returns the file path that the LangServer object is using to connect to the message log database. It can be the file path to a MS Access database file (extension .mdb) or to a MS data link file (extension .udl). Data link file is a text file that contains an OLE DB connection string.
Syntax	LoggerDSrc As String
Remarks	This property is stored in the computer registry database. No errors should occur.
Example	Dim LangServerLoggerDataSource As String LangServerConfigurationDataSource = AnLangServer.LoggerDSrc

LoggingEnabled

Description	(Read-only) When TRUE, then the LangServer object logs information and warning messages to the database specified in the LoggerDSrc property. If FALSE, then the server does not log any message to the logger database.
Syntax	LoggingEnabled As Boolean
Remarks	This property, as well as other properties, is stored in the language-configuration database. An error occurs if the server has not been able to connect to the configuration database.
Example	Dim ALoggingEnabled As Boolean ALoggingEnabled = ALangServer.LoggingEnabled

VariableText

Description	(Read-only) Returns the expression variable identification string. This string identifies the input value in the expression string. The language server client should replace all occurrences of this string in an expression string with "real" input value and then evaluate it using the expression engine.
Syntax	VariableText As String
Remarks	This property value is hard-coded in the LangServer object. In future versions, it will be stored in the configuration database.
Example	Dim AVariableText As String AVariableText = ALangServer.VariableText

LangServer Methods

GetTranslationText

Description	Returns the text translation(s) for a specified AliasID string.
Syntax	GetTranslationText (AliasID As String, Language As String, LanguageSubset As String, ClientIdentification As String, ErrMessageLocaleID As Long, FontFaceName As String) As String

Part	Description
AliasID	The AliasID string contains alias name(s) for which the caller wants to read the text translation(s). For the AliasID string "any text *Alias1* any other text *Alias2* etc…" the result will be "any text *TextTranslation1* any other text *TextTranslation2* etc…".
Language	The Language name for which the caller wants to read the translation(s).
LanguageSubset	The Language subset name for which the caller wants to read the translation(s). It can be an empty string if the caller wants to read translation for default language subset.
ClientIdentification	The LangServer object uses the ClientIdentification string to identify message records in the logger database. It should contain a unique client identification (including e.g. display name) and node information.
ErrMessageLocaleID	The ErrMessageLocaleID parameter is the locale ID that the LangServer object will use to create an error message description (method IErrorInfo::GetDescription(…..)).
FontFaceName	The FontFaceName returns font face name associated with the first alias string in AliasID parameter. The client should use the font face name to display translated text. If the FontFaceName is an empty string, then the client uses its default settings.

Remarks	This method can fail from many reasons. Three basic reasons are: **1.** The LangServer object has not been able to connect the configuration database. **2.** AliasID string contains invalid delimiters (opening delimiter was found but not ending, etc.). **3.** Language or Language Subset string does not exist in the configuration database.

Example

```
Private Sub cmdSomeButton_Click()
        On Error GoTo L_ERROR
        ' Read translation text for two aliases with names  "Alias1" and "Alias2";
        ' delimiters are `` and "

        Dim ALangServer As LangServer                   ' Declare Alias Server
        Set ALangServer = New LangServer                ' Create an instance
        If ALangServer Is Nothing Then                  ' Check for failure
                MsgBox "Failed to create an instance of 'LangServer'", vbOKOnly, "SMAR LangServer"
                Exit Sub
        End If

        Dim FontName As String
        Dim Text As String

        ' Get translation for language "English - United States" and default language subset; error
        ' messages in English
        strText = ALangServer.GetTranslationText("``Alias1" ``Alias2"", "English - United States", "", "My
identification", 1033, strFontName)

        ' Some code to show the results (strText, strFontName)….
        Set ALangServer = Nothing
        Exit Sub

L_ERROR:                                                ' Handle errors
        LblTranslText.Caption = ""
        MsgBox "Failed to obtain alias: " & Err.Description, vbOKOnly, "SMAR AliasServer"
End Sub
```

GetTranslationRecordset

Description	Returns the LangServerRecVarDescriptor object that contains all translation parameters (text translation, read/write expressions, etc.) for a specified AliasID string.
Syntax	GetTranslationRecordset (AliasID As String, Language As String, LanguageSubset As String, ClientIdentification As String, ErrMessageLocaleID As Long, Descriptor As LangServerRecVarDescriptor)

Part	***Description***
AliasID	The AliasID string has to contain alias name for which the caller wants to read the translation parameters. This method returns translation parameters for the first found alias name only (compare with GetTranslationText method).
Language	The Language name for which the caller wants to read the translation parameters.
LanguageSubset	The Language subset name for which the caller wants to read the translation parameters. It can be an empty string if the caller wants to read translation parameters for default language subset.
ClientIdentification	The LangServer object is using the ClientIdentification string to identify message records in the logger database. It should contain unique client identification (including e.g. display name) and node information.
ErrMessageLocaleID	The ErrMessageLocaleID parameter is the locale ID that the LangServer object will use to create an error message description (method IErrorInfo::GetDescription(…..)).
LangServerRecVarDescriptor	The GetTranslationRecordset method is using this object to return all translation parameters (see chapter LangServerRecVarDescriptor for details).

Remarks	This method can fail from many reasons. Four basic reasons are: **1.** The LangServer object has not been able to connect the configuration database. **2.** AliasID string contains invalid delimiters (opening delimiter was found but not ending, etc.). **3.** AliasID string does not contain any translation alias name. **4.** Language or Language Subset string does not exist in the configuration database.

Example

```
Private Sub cmdSomeButton_Click()
        On Error GoTo L_ERROR
        ' Read translation parameters for alias with name  "Alias1"; delimiters
        ' are `` and ''

        Dim ALangServer As LangServer                    ' Declare Alias Server
        Set ALangServer = New LangServer                 ' Create an instance
        If ALangServer Is Nothing Then                   ' Check for failure
                MsgBox "Failed to create an instance of 'LangServer'", vbOKOnly, "SMAR LangServer"
                Exit Sub
        End If

        Dim RecDescriptor As LangServerRecVarDescriptor          ' Declare descriptor object

        ' Get translation parameters for language "English - United States" and
        ' default language subset
        Call ALangServer.GetTranslationRecordset ("``Alias1''", "English - United States", "", "My identification",
1033, RecDescriptor)

        ' Add some code to show the results from RecDescriptor:
        ' RecDescriptor.TranslationText, RecDescriptor.ExpressionFormat, etc.

        Set ALangServer = Nothing
        Exit Sub

L_ERROR:                                                 ' Handle errors
        LblTranslText.Caption = ""
        MsgBox "Failed to obtain alias: " & Err.Description, vbOKOnly, "SMAR AliasServer"
End Sub
```

GetLocaleID

Description	Returns the locale ID associated with specified Language name.
Syntax	GetLocaleID (Language As String, ErrMessageLocaleID As Long) As Long

Part	*Description*
Language	The Language name for which the caller wants to read the locale ID.
ErrMessageLocaleID	The ErrMessageLocaleID parameter is the locale ID that the LangServer object will use to create an error message description (method IErrorInfo::GetDescription(…..)).

Remarks	This method can fail from many reasons. Two basic reasons are: **1.** The LangServer object has not been able to connect the configuration database. **2.** Language string does not exist in configuration database,

Example

```
Private Sub cmdSomeButton_Click()
        On Error GoTo L_ERROR

        Dim ALangServer As LangServer                    ' Declare Alias Server
        Set ALangServer = New LangServer                 ' Create an instance
        If ALangServer Is Nothing Then                   ' Check for failure
                MsgBox "Failed to create an instance of 'LangServer'", vbOKOnly, "SMAR LangServer"
                Exit Sub
        End If
        Dim LanguageLCID As Long

        ' Get LCID for language "English - United States"
        LanguageLCID  = ALangServer.GetLocaleID ("English - United States", 1033)

        ' Add some code to show the result

        Set ALangServer = Nothing
        Exit Sub

L_ERROR:                                                 ' Handle errors
        LblTranslText.Caption = ""
        MsgBox "Failed to obtain alias: " & Err.Description, vbOKOnly, "SMAR AliasServer"
End Sub
```

LangServerRecVarDescriptor Object

Description	The LangServerRecVarDescriptor object is a collection of translation properties. All supported properties are

	read-only. The LangServer object creates the LangServerRecVarDescriptor object and initializes it with translation parameters (see GetTranslationRecordset method in LangServer object). The client then can read all these properties to get translation parameters.
Syntax	LangServerRecVarDescriptor
Example	Dim ADescriptor As LangServerRecVarDescriptor

Summary of Properties

TranslationText	ExpressionFormat	ReadExpression
WriteExpression	FontFaceName	

LangServerRecVarDescriptor Properties

TranslationText

Description	(Read-only) Returns the translation text for a specified alias.
Syntax	TranslationText As String
Remarks	The TranslationText value is created similarly to the result of GetTranslationText method (LangServer object) but it returns translation text for the first found alias only.
Example	Dim ATranslationText As String ATranslationText = ADescriptor.TranslationText

ExpressionFormat

Description	(Read-only) Returns the expression format string associated with a specified alias. This format string should be used to format numeric (expression results) or date values. If this string is empty then no formatting should be done. The format string is using following syntax: 1. Numeric formats: "TYPE=<*type_name*> FORMAT=<*format_string*>" where *type_name* can be one of following: FLOAT, DOUBLE, BOOL, BYTE, WORD, DWORD, CHAR, SHORT, LONG and *format_string* is the string that uses "x" characters to specify numeric or decimal places, "." character specifies decimal point. 2. Date-time formats: "TYPE=<DATETIME> TIME=<*time_format_string*> DATE=<*date_format_string*>". DATE and TIME keywords are optional; if the formats string contains both then their order specifies the order of time and date values in the result.
Syntax	ExpressionFormat As String
Example	Dim AnExpressionFormat As String AnExpressionFormat = ADescriptor.ExpressionFormat

ReadExpression

Description	(Read-only) Returns the read expression string associated with specified alias. The client uses this expression to convert read value for a specified language (or language subset) translation. If the ReadExpression contains an empty string, then no conversion is necessary.
Syntax	ReadExpression As String
Remarks	The expression string contains special substring(s) defined by *VariableText* property in LangServer object. This substring should be replaced by input value, and then this expression string can be evaluated using the expression engine .dll.
Example	Dim AReadExpression As String AReadExpression = ADescriptor.ReadExpression

WriteExpression

Description	(Read-only) Returns the write expression string associated with a specified alias. The client uses this expression to convert a written value for a specified language (or language subset) translation. If the ReadExpression contains an empty string, then no conversion is necessary.
Syntax	WriteExpression As String
Remarks	The expression string contains special substring(s) defined by *VariableText* property in LangServer object. This substring should be replaced by the input value, and then this expression string can be evaluated using the expression engine .dll.
Example	Dim AWriteExpression As String AWriteExpression = ADescriptor.WriteExpression

FontFaceName

Description	(Read-only) Returns the font face name associated with a specified alias. The client should use the font face name to display translated text. If the FontFaceName is an empty string, then the client uses its default settings.
Syntax	FontFaceName As String
Example	Dim AFontFaceName As String AFontFaceName = ADescriptor.FontFaceName