wk *syspro*

FlexSpooler ProgramSuite 2.0

XLanguage V2.6

Additions/Changes compared to XLanguage V2.5

Solution Partner

PLM

SIEMENS

Velocity Series

Additions/Changes to XLanguage V2.6 compared to V2.5

Optional support for native Solid Edge tables

The texts of Solid Edge tables may now optionally be translated too. Table titles, column headers and cell data texts are full text searched against all languages available in the XLanguage database. Also place holder texts are supported! For better performance the language columns of the database table TextBlocks should now be indexed.

The translation of Solid Edge tables is controlled by the newly introduced configuration setting *TranslateTables=0|1* in the configuration file "XLanguage.ini".

To insert text from the Xlanguage PlaceTextBlock dialog into a selected Solid Edge table title, column header or data cell just press the Ctrl- or Shift-Key while selecting a text in the list box of the PlaceTextBlock dialog, which causes the text to be copied onto the Windows clipboard. Then paste the text from the Windows clipboard by pressing Ctrl-V into the selected table area of the opened table properties dialog of ST4/ST5 or the directly per double click selected table cell with ST6. If a place holder text is selected, the EditPlaceHolder dialog will be opened first to allow comfortable editing of the place holders before the (edited) text is copied onto the Windows clipboard.

All Solid Edge table formatting options are supported. Attention should be payed to select the right font (i.e. Arial Unicode MS).

Due to the internal organization of tables by Solid Edge and restricted API support table contents gets translated on all sheets they appear and not only on the currently activated sheet.

Optional control of the XLanguage text box border line color by the category of its text block

The border line color (yellow by default) may now be controlled by the category assigned to the text in the database. The mapping between category IDs and border line colors is controlled by the newly introduced configuration setting *TextBlockBorderLineColorsByCategory* in the configuration file "XLanguage.ini". The value of this setting is a comma separated list of hexadecimal coded ARGB-colors where the first entered color corresponds to the text category ID 1, the 2nd to category ID 2 and so forth.

Toggling ShowTextBlockID setting by pressing the Ctrl-Key while clicking on the Translate button in the XLanguage TranslateTextBlocks dialog

For mostly temporary usage, the display of all XLanguage texts prepended by their database ID enclosed in brackets (ShowTextBlockID configuration setting) may now also be interactively controlled by pressing the Ctrl-Key while clicking on the Translate button in the Xlanguage TranslateTextBlocks dialog. The size of the text boxes is not recomputed which may lead to wrapped and/or overlapping text boxes. To recompute the maximum required size of all text boxes, switch on from the XLanguage main dialog the option to display the text box border lines while pressing the Ctrl- or Shift-Key.

The list box of the XLanguage PlaceTextBlock dialog now also displays the database ID of the texts if the setting ShowTextBlockID is activated

If the ShowTextBlockID option is set either by the corresponding configuration setting or interactively as described above, the text blocks presented in the list box of the XLanguage PlaceTextBlock dialog will be prepended by their database ID enclosed in brackets for the sake of easier navigation to certain IDs. But text block IDs still can't be used as search criterion.

The list box of the XLanguage PlaceTextBlock dialog now got the input focus instead of the language combo box

When the XLanguage PlaceTextBlock dialog gets opened the input focus is now set to the list box allowing immediately scrolling through the text blocks by the use of the mouse wheel or the up-/down- arrow keys of the keyboard.

When clicking the Translate button in the XLanguage TranslateTextBlocks Dialog no longer a confirmation dialog appears

There is now no longer a confirmation required by the user when pressing the Translate button of the XLanguage TranslateTextBlocks dialog.

Usage note

While placing a text block via XLanguage onto the Solid Edge drawing sheet the process must not be interrupted by the user by pressing the ESC key! Since this will cause Solid Edge to break the communication of Solid Edge mouse events with the external XLanguage process. XLanguage will then stay in invisible state since the awaited events for Solid Edge mouse moves and/or a left/right mouse click never get sent by Solid Edge. If such a situation happened, the still running XLanguage process first must be terminated via the Windows Task Manager dialog before trying to restart XLanguage.

Installation/Uninstallation

Use the so called xcopy method to install XLanguage: Just copy the folder containing XLanguage and its components to i.e. "C:\Program Files" onto each computer.

If you „install" to a network share it is no longer required to configure each computer's code access security policy (CAS policy) since .NET 4.0 by default now full trusts execution of .NET executables from within the intranet.
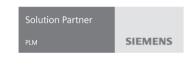
To uninstall XLanguage just remove the folder containing XLanguage from the system.

Requirements

XLanguage is verified to run on the configurations below. Updates to this list can be found in the online documentation on: http://www.wksyspro.de/en/support/support.html

Attention: XLanguage V2.6 requires at least the .NET 4.0 client profile being installed!

System/Application	Solid Edge V20 32/64-Bit	Solid Edge ST-ST6 32/64-Bit
Windows Vista 32/64-Bit	√	√
Windows 7 32/64-Bit	√*	√
Windows 8/8.1 32/64-Bit	–	√**

* Solid Edge is not officially released for this version of Windows.

** Solid Edge ST6

Same applies to the Windows Server versions : 2003, 2003/R2, 2008, 2008/R2, 2014, 2014/R2

Configuration

Some configuration of XLanguage can be done by editing the file "XLanguage.ini". The table below shows the contents of the configuration file along with the comments describing the meaning of the entries. In order since V2.3 to additionally support Microsoft SQL Server as database server there were some minor changes regarding the database access. XLanguage V2.3 and higher implemented a provider independant database access which allows to specify the provider to be used via the configuration. By specifying the so called .NET database provider invariant name and the corresponding provider specific .NET database connection string you are free to use any database supported by .NET.
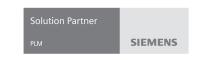
The configuration file contains 2 preconfigured pairs of provider invariant name and connection string. One pair configured for usage of a MDB file based database and the other one configured for usage of a Microsoft SQL Server based database.

[Options]
# Options section
# <Required>: Pair of .NET Database provider invariant name and connection string.
# Sample entries for Microsoft Access 2003 MDB database file
DbProviderInvariantName="System.Data.OleDb"
DbConnection="Provider=Microsoft.Jet.OLEDB.4.0; Data Source=|DataDirectory|\XLanguage.mdb"
# Sample entries for Microsoft SQL Server
;DbProviderInvariantName="System.Data.SqlClient"
;DbConnection="Data Source=P5Q\SQLEXPRESS;Initial Catalog=XLanguageSQL;Integrated Security=True"
# <Required>: (Unicode) font name used by the GUI, Solid Edge and the Access Database.
# Caution: Case sensitive!
Font=Arial Unicode MS
;Font=Solid Edge ISO
<Required>: GUI Font size in points.
Decimals must match your Windows region settings for numbers.
;GuiFontSize=8,25
GuiFontSize=8.25
# <Required>: Legacy option to precede each text block with its text block ID enclosed by []
#   (0=No, 1=Yes).
ShowTextBlockID=0
# <Required>: Legacy option to scan the text of a textbox/callout for its text block ID (0=No, 1=Yes).
# This option is required only to process text blocks produced by the
# program 'SeTextBlock' of version V1.0.0.161 or earlier.
ScanTextForTextBlockID=0
# <Required>: Name of the dynamic attribute set assigned to XLanguage textboxes.

```
# Caution: Case sensitive!
AttributeSetName=SeTextBlock
# <Required>: Name of the attribute storing the text block ID of an XLanguage textbox/callout.
# Caution: Case sensitive!
AttributeName=ID
# <Required>: Option to enhance standard textboxes/callouts by XLanguage text block attributes (0=No, 1=Yes).
# Enhancement of a textbox/callout happens only, if its text exactly matches an existing text block
# (not case sensitive, any language)in the XLanguage database.
# Its text size keeps unchanged, its font is set to the font defined above and in case of a textbox
# its maximum boundary gets recomputed to fit all current languages.
# Caution: Setting this option may slow down the speed of processing depending on the
# number of standard textboxes/callouts to scan against all text blocks in the database!
# As result of changing the font of a textbox/callout, its width might increase also.
EnhanceStdTextBoxes=0
# <Optional>: Whether Solid Edge table contents should be processed too (0=No, 1=Yes).
TranslateTables=0
<Optional>: Comma separated list of text block border line colors
#          (hexadecimal format for Alpha, Blue, Green and Red as &HABGR) by category ID.
#          The 1st entry defines the color to be used for text block category id 1, the 2nd color for category 2 etc.
#          The 1st entry is also taken as default for category ids higher than specified.
#          Hexadecimal color code examples: Yellow=&H0000FFFF=&HFFFF, Red=&H000000FF=&HFF
TextBlockBorderLineColorsByCategory=&HFFFF,&HFF
# <Optional>: Default language for the graphical user interface (GUI).
# Caution: Case sensitive!
GUILanguage=German
# <Optional>: Default language to translate text blocks to
# Caution: Case sensitive!
TextBlockLanguage=English
# <Optional>: Option to show GUI buttons under Windows XP in XP visual style (0=No, 1=Yes).
# If 'No', flat popup style will be used (recommended for Windows 2000 or classic style).
XPButtonStyle=1
```
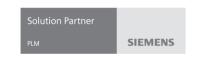
Please refer to the standard Microsoft .NET documentation on all available options of the connection string formats depending on the database provider chosen.

Appendix

If you intend to change XLanguage from using a shared MDB database file to a Microsoft SQL Server based database you may migrate from a copy of your existing MDB database file easily to a SQL Server database by opening the MDB file in the Microsoft Access application and select from the menu *Database Tools* to transfer all tables of the current database to a new SQL Server database on your SQL Server.

Next set the values of *DbProviderInviariantName* and *DbConnection* in the XLanguage configuration file accordingly. The access to the SQL Server database is fully controlled by the available options for the connection string. So you may either use Integrated Security or SQL Server login. Inside SQL Server map the Integrated Security logins or explicit SQL Server logins to the XLanguage database and make them member of the database role *db_datareader.*