Language Manager is a freely distributed utility meant for language files creating, translating and for creating of resource DLLs.

Language file is a file in a special format that contains strings to be translated (forms resources and string constants) for the concrete language. It has the **.xxx.lng** extension, where xxx is two- or three-character language abbreviation set by the user in the time of file creating.

The language files contain the translations you made. For the OnFly edition they also are the final product - they are distributed to end user together with EXE.

Native language file is a language file for application native language. It has the **.ntv.lng** extension. It is not the subject for translation, is renewed by Language Wizard (see Localizer menu in the Delphi IDE).

Native language file also serves as **project** for Language Manager.

The native language file is created by Language Wizard when you localize your Delphi project first time. All other language files you should create using Language Manager.

Each language file contain three section:
1. Forms - string and non-string (for example position and size of controls) items extracted from form controls used in your program.
2. Units - resource strings placed in resourcestring section and string table resources from .rc or .res files included into the project;
3. External - any other string items defined by user. This section can contain one or more groups of items each group contain one or more named string items. Developer can add new groups and items into External section and then access those items and their translations to other languages by several procedures described in LocOnFly API topic.
Read External items topic to learn how to add new external item or group or modify/delete existing ones.

See also:
Operating projects
Working with language files
Export and Import

Resource DLL – the dynamic link library (DLL) that contains application resources (namely forms and string constants) and doesn't contain any code. It is the final product of Localizer Standard. The application that uses Localizer OnFly automatically creates resource DLLs on the end user computer. The resource DLL is created with extension corresponding to two- or three-character language abbreviation set by user in the time of file creating.

Language Manager allows you to create resource DLLs for the languages you translate to. You need to have access to the application executable file (EXE, BPL, DLL) to make this option available.

See also:
Creating resource DLL

The **repository** is a multilingual dictionary that is filled in by the user (translator) and is served for (semi) automatic translating. This tool is especially convenient for the translators that translates several applications with the similar interface elements.

You may change the repository currently used by Language Manager. It allows you to have different repositories for different purposes.

The repository database can be imported from/exported to the XML file compatible with the one exported by the Delphi Language Repository (see Borland's Integrated Translation Environment).
You may also import MS Glossary files to the repository.

See also:
Using the repository
Filling the repository
Repository form
Repository export/import
Options

You can translate third-party components resources just once and then use the results any time. Thereto, some developers make the localized resources of their components available in some way.

Library is a collection of language files that contain the translation of the resources of some library (third-party, VCL, etc.). The library translation process is nearly the same as the usual project translation.

The Open Library dialog ("Libraries | Open Library…" menu item) allows you to open one of the registered libraries as the Language Manager project. The libraries that does not contain a native file are disabled.

To process component resources and create corresponding library, use the special tool – **Library Manager**. Created libraries can be stored anywhere in the separate folder. To register the library in the Language Manager create (if it is not created yet) the "HKEY_LOCAL_MACHINE\SOFTWARE\Korzh\Language Manager\Libraries" key in the Windows Registry. Then add the DWORD value with name set to the path of the library and the value set to 1. You can temporarily exclude the library from the library list by setting the value to 0.

The main goal of using libraries is to apply stored translations to the current project.

See also:
Apply library

External section contains string items defined by developer. It can be used to store (and then translate) any messages, captions, etc. which are not used as properties of some components and could not be placed into resources by some reasons.

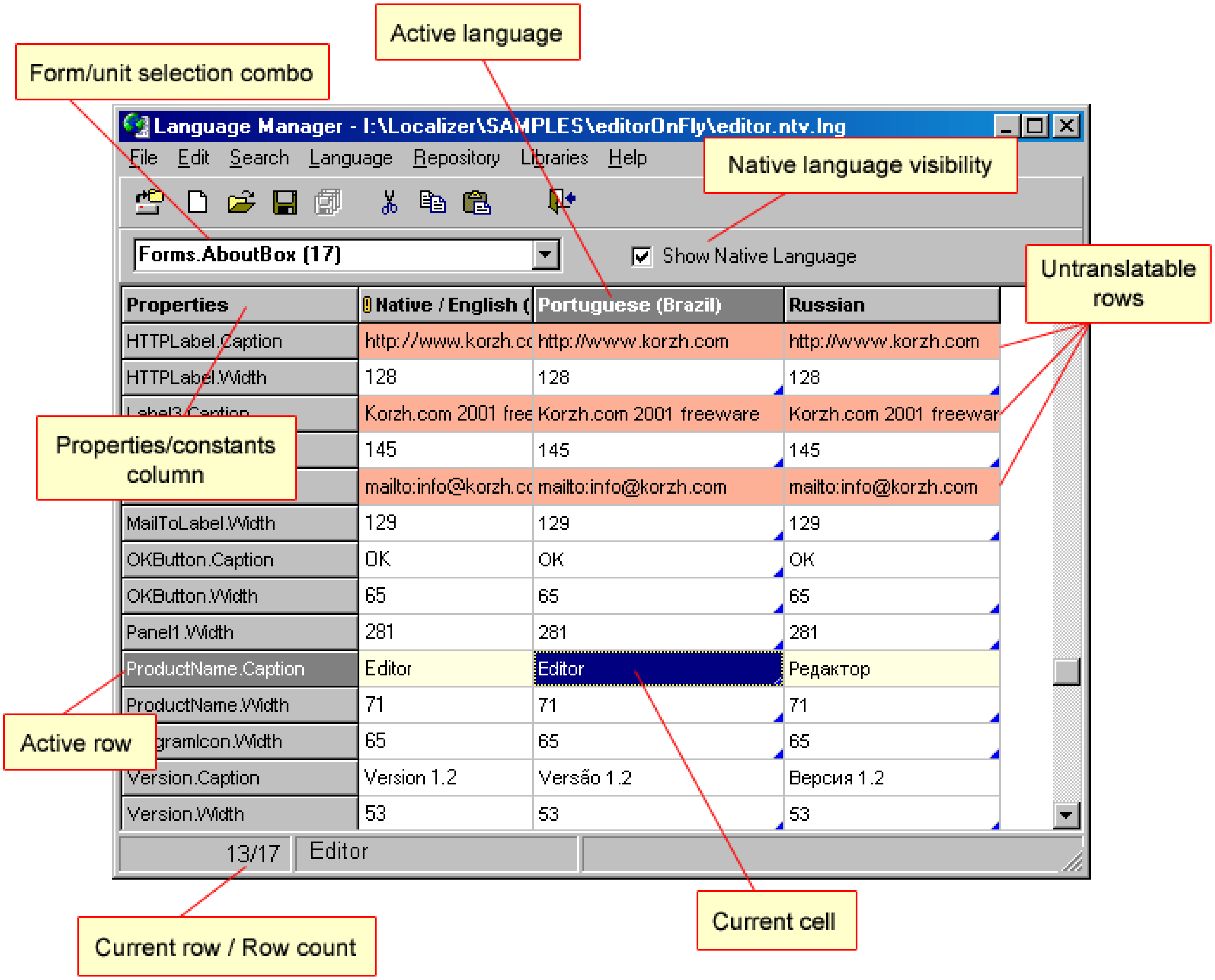
This section can contain one or more groups of items each group contain one or more named string items.

To add new item and group use **Edit | Add External Item** menu command or **Ctrl+Shift+X** shortcut. In the dialog that appear you should enter name of item group (or select it from the list of available groups), item name and its native value.

To edit some already existed external item you need to select it in the main grid (previously you possibly need to select External section at the top-right combo box and necessary group in this section) and then use **Edit | Edit External Item** menu command or **Ctrl+Shift+E** shortcut.

To delete some external item use **Edit | Delete External Item** command.

When the project is opened you see a grid that allows you to translate your language files. Each column but the fixed one corresponds to one language. You may change the order of columns by dragging and dropping them. Fixed (the most left) column contains properties / constants names.


Active language
Form/unit selection combo
Native language visibility
Untranslatable rows
Properties/constants column
Active row
Current cell
Current row / Row count
Language Manager - I:\Localizer\SAMPLES\editorOnFly\editor.ntv.lng
File Edit Search Language Repository Libraries Help

Forms.AboutBox (17)

☑ Show Native Language

Properties	Native / English (	Portuguese (Brazil)	Russian
HTTPLabel.Caption	http://www.korzh.cc	http://www.korzh.com	http://www.korzh.com
HTTPLabel.Width	128	128	128
Label3.Caption	Korzh.com 2001 free	Korzh.com 2001 freeware	Korzh.com 2001 freewar
	145	145	145
	mailto:info@korzh.cc	mailto:info@korzh.com	mailto:info@korzh.com
MailToLabel.Width	129	129	129
OKButton.Caption	OK	OK	OK
OKButton.Width	65	65	65
Panel1.Width	281	281	281
ProductName.Caption	Editor	Editor	Редактор
ProductName.Width	71	71	71
gramIcon.Width	65	65	65
Version.Caption	Version 1.2	Versão 1.2	Версия 1.2
Version.Width	53	53	53

13/17 Editor

The grid filled with properties/constants values is the main field of action of a translator.
Each column of the grid represents the language file to be translated. Each row represents the property or the constant to be translated.
The special mark * put on the right side of the column caption means that the language has been modified (some cells have changed their values).

Native language visibility check box

Unmark this check box if you want to hide the native language column. Mark it again to show the native language.

Properties/Constants column

The most left (fixed) column displays the names of the properties or constants. The properties/constants names includes form/unit name if the form/unit selection combo is set to display all forms/units (i.e. Form1.Label1.Caption is displayed if Forms is selected). Otherwise, the forms/units names are omitted (i.e. Label1.Caption is displayed if Form1 is selected).

Click on the cell in this column to see a hint with full property/constant name.

Active language

Active (current) language – the language of one of opened language files. Each open file corresponds to one column in the table. The active language column is marked out with darker header color. To make the column (language) active select one of it cells.
Operations concerning the language (somewhat file operations, font changing etc.) are performed under the active language.

Read-only columns

The columns marked with ! sign are read-only. It means that you will not be able to edit the cells in those columns.
The native language column is always marked as read-only to prevent the editing of native language.

Translated/untranslated

The ◢ sign in the cell right bottom corner indicates that its content was not translated yet (or its version is out-of-date). The state is changed to "translated" when you edit the cell.
If the native value was updated by the Language Wizard, the others state is changed to "untranslated". In this case the values that have never been translated are changed to the new native value.
The translator can also set the state manually.

Untranslatable items

The developer can mark some properties or constants as untranslatable. Such rows are marked out with color and do not allow to edit their cells. The program can automatically hide the untranslatable rows if the appropriate option is set.

The Language Manager installed separately (not in the Localizer package) does not allow to change the translatable/untranslatable status of the row.

The untranslatable properties and constants are not changed during language switching.

Verified cells

If the cell value is checked and is not to be changed in the future, the cell can be marked as "verified". Such cells are marked out with color and can not be edited (read only).

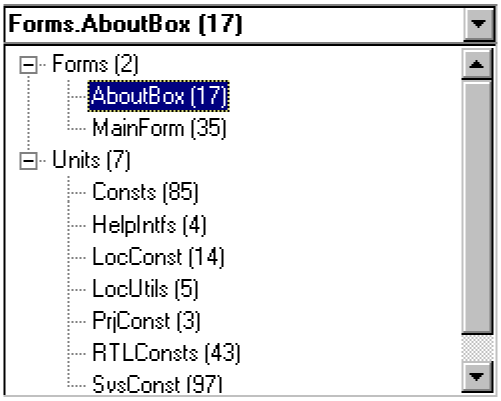
The cells marked as untranslated can not be marked as verified. When the cell state is changed to "untranslated" (due to native language updating), the "verified" mark is removed.

Filtering

To simplify the translator's work Language Manager give you the possibility to limit the count of properties and constants displayed depending on your needs.

Form/unit selection combo

This combo box allows to limit the set of displayed resources by one/all form(s) or by one/all unit(s) with constants. Choose it to cause appearance of the two-level tree that contains separately form names and unit names sorted alphabetically.



Forms.AboutBox (17)

Forms (2)
- AboutBox (17)
- MainForm (35)

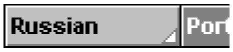
Units (7)
- Consts (85)
- HelpIntfs (4)
- LocConst (14)
- LocUtils (5)
- PrjConst (3)
- RTLConsts (43)
- SysConst (97)

Each form/unit name is accompanied by the number of items (in the brackets). The root nodes display the number of forms/units respectively.

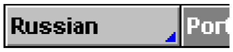
Translated/untranslated

Language Manager allows you to view only translated values, only untranslated or both (default). Make appropriate language active and choose one of subitems of "Language | Show" menu item.

Current mode is indicated with triangle on the column header.



– show translated values only.



– show untranslated values only.

If neither of columns have such indication then both translated and untranslated values are displayed.

Limitation can be applied to single column in a time.

Textual/non-textual

Language Manager allows you to view only textual properties, only non-textual properties (i.e. Width, Height etc.) or both (default). Make appropriate language active and choose one of subitems of "Language | Show" menu item.

This filter doesn't have effect on constants displaying.

Untranslatable items

Untranslatable items are marked out with color or hidden depending on "Show untranslatable items" option. Select "Language | Show | Untranslatable Only" menu item to display untranslatable items only.

Sorting

Language Manager allows you to view string values ordered ascending or descending. Values are ordered according to sorting rules of selected language.

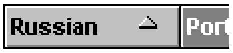
Original order

Properties are initially ordered by form name alphabetically. Inside of each form the order is the following:
- form properties are ordered alphabetically in the beginning;
- form components are ordered alphabetically. Properties of each component are ordered alphabetically too.

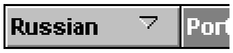
Constants are ordered by unit name alphabetically. Constants of each unit are ordered by constant name alphabetically.

Sorting the language

Current sort order is indicated with triangle on the column header.



– ascending sort order.



– descending sort order.

If neither of columns have such indication then values are displayed in the original order.

To sort values just click on column header or make appropriate language active and choose one of subitems of "Language | Sort" menu item.

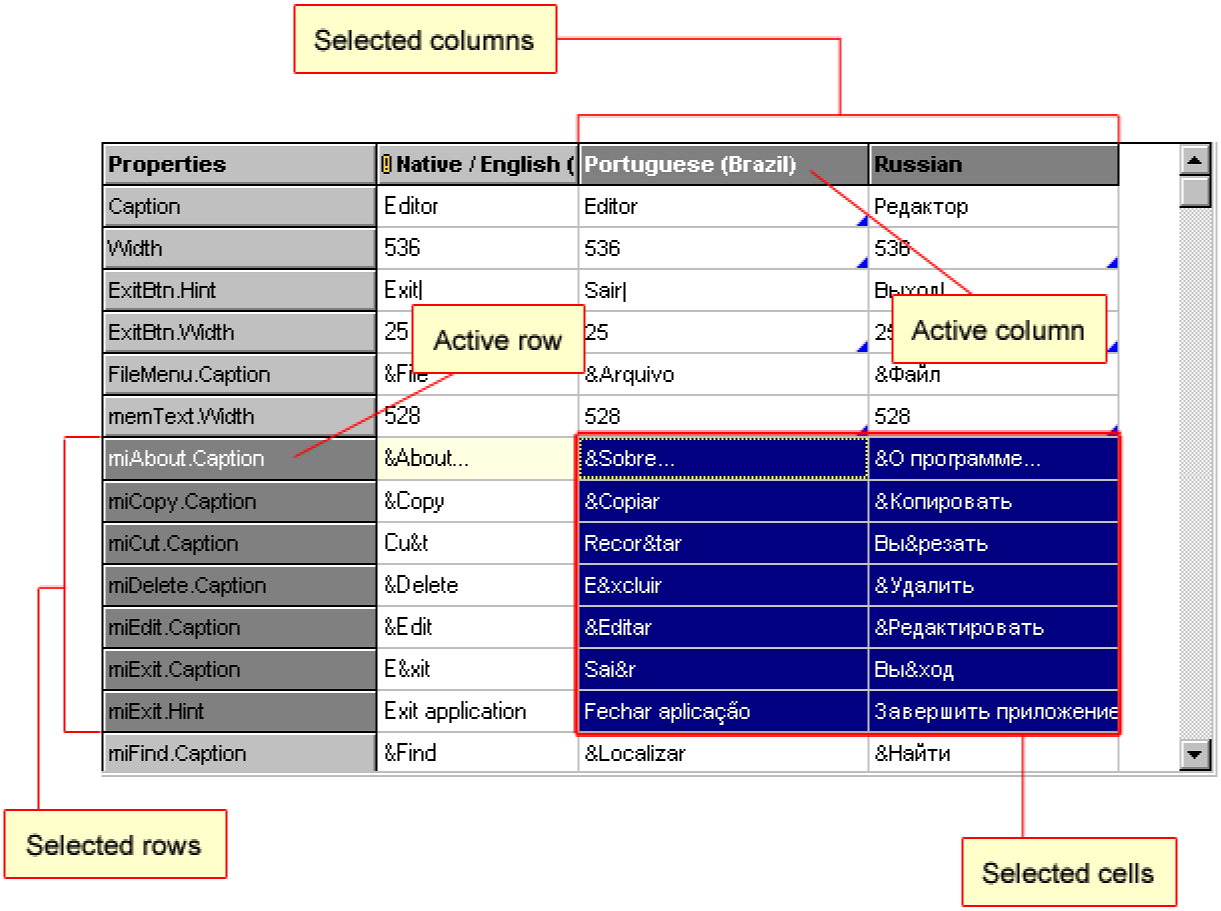
You can also sort the fixed column (properties/constants names) by pressing on fixed column header. In this case properties can be sorted by full name (i.e. "Form1.Edit1.Text") of just by property name ("Text") depending on “Compare properties by full name” option value.

Sorting can be applied to single column in a time.

Refreshing grid

When you change the values the grid does not reorder or filtered automatically. You need to refresh grid ("Edit | Refresh" menu item or F5 shortcut) to get the values ordered and filtered properly.

Starting from version 3.0 Language Manager allows to select and manipulate the range of cells. It is very useful if you need to perform the same action for numerous cells.



Properties	Native / English (	Portuguese (Brazil)	Russian
Caption	Editor	Editor	Редактор
Width	536	536	536
ExitBtn.Hint	Exit|	Sair|	Выход|
ExitBtn.Width	25	25	25
FileMenu.Caption	&File	&Arquivo	&Файл
memText.Width	528	528	528
miAbout.Caption	&About...	&Sobre...	&О программе...
miCopy.Caption	&Copy	&Copiar	&Копировать
miCut.Caption	Cu&t	Recor&tar	Вы&резать
miDelete.Caption	&Delete	E&xcluir	&Удалить
miEdit.Caption	&Edit	&Editar	&Редактировать
miExit.Caption	E&xit	Sai&r	Вы&ход
miExit.Hint	Exit application	Fechar aplicação	Завершить приложение
miFind.Caption	&Find	&Localizar	&Найти

Selected cells

The selected cells are marked with Navy color. Current cell is outlined with dotted line in addition. If you press **Enter** or **F2**, the selection is cleared and the current cell is gone to the edit mode.

Selected rows/columns

Language Manager performs some operations on entire rows or entire columns. The selected rows or columns are the scope of such actions.

How to select a range of cells?

The range of cells can be selected using mouse (press left mouse button and hold it while dragging the cursor) or keyboard (hold **Ctrl** button and use arrows buttons) or both (select cell, hold **Shift** button and click on the other cell).

Click on any cell or use arrows keys to change the selection.

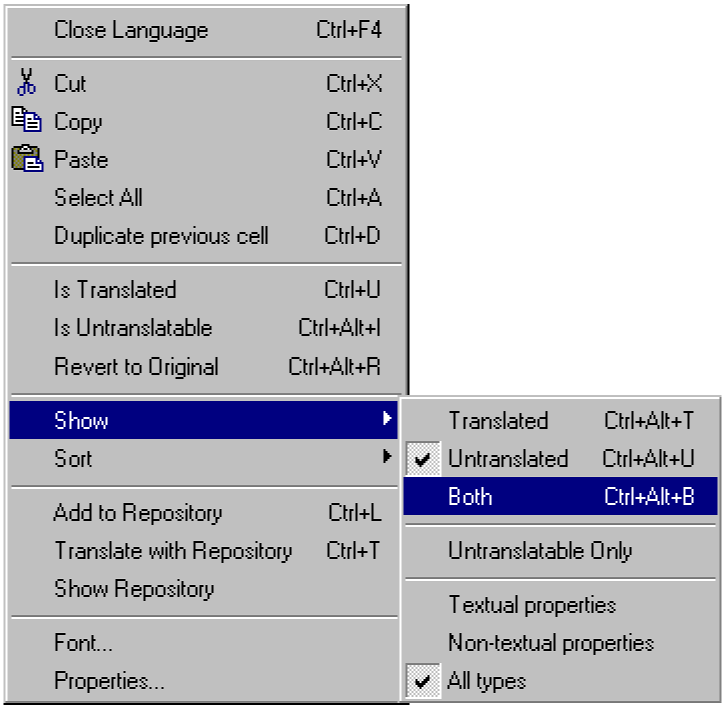
Why do some menu items become unavailable?

Language Manager automatically disables the operations that are not allowed on the selected range of cells. For example, some actions are performed on the strings of the single language (i.e. "Revert to original", repository actions etc.). So, if more than one column is selected then such action will not be available.

Clipboard operations (**Cut**, **Copy**, **Paste**) do not work on multiple cells selected.

A lot of menu items are duplicated in the context menu that is appeared by the right mouse button click.

If multiple cells are selected, you may right click on any selected cell to evoke the context menu.



The most of frequently used commands are also available through the keyboard shortcuts.

You can customize some of the features of Language Manager by selecting "Edit | Options…" menu item. A dialog box appears with the following customizable features:

The "General" tab

☑ **Try to translate automatically** – mark check box if you want Language Manager to search the same string as you have just translated each time you enter some new value into a cell. Language Manager tries to translate each of found cells unless the "Use replace toolbar" check box is marked.

☑ **Use replace toolbar** – mark check box if you want the toolbar to appear for managing of the automatic translation. The check box is disabled if "Try to translate automatically" check box is not marked.

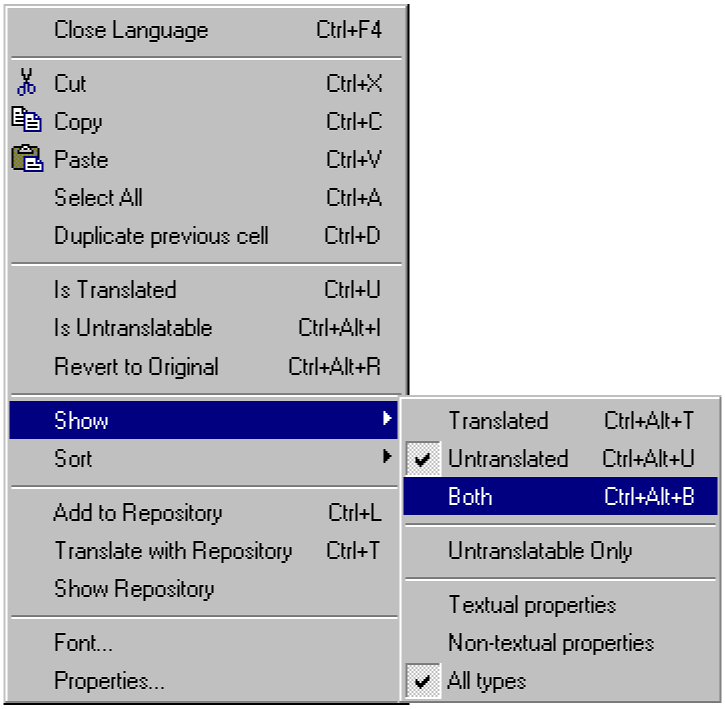
☑ **Use Multiline Cell Edit Dialog** – mark check box if you want to edit multiline cells with the special dialog. Otherwise they are edited with drop-down memo control.

☑ **Compare properties by full name** – sets the rule for sorting of properties in the fixed column. If checked, properties are sorted by full name (i.e. "Form1.Edit1.Text"), else – just by property name (i.e. "Text").

☑ **Show Untranslatable items** – mark check box if you want untranslatable items to be shown in the grid. The untranslatable items are marked out with color.

☑ **Move down automatically** – mark check box if you want to automatically jump to the next cell when you changed the current cell or its status (translated/untranslated, untranslatable etc.).

☑ **Refresh project on close** – mark check box if you want all language files to be refreshed when the project is closed. It is useful if you change "Untranslatable" status of some items.



RightToLeft for middle east – the middle east languages will be displayed using RightToLeft alignment. Moreover, you will be able to input text as right-to-left.

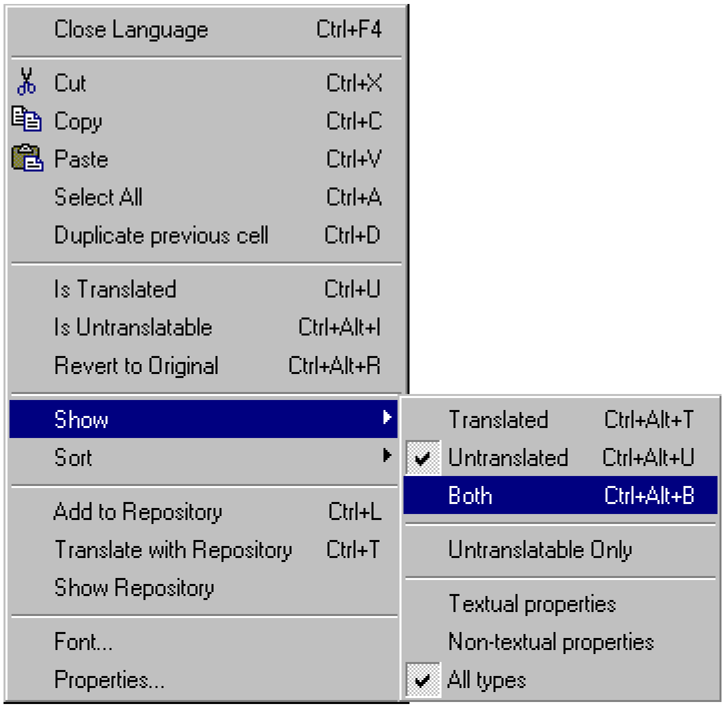
Grid Font – displays the grid font properties. Click "Change..." button to change the grid font.

The "Repository" tab

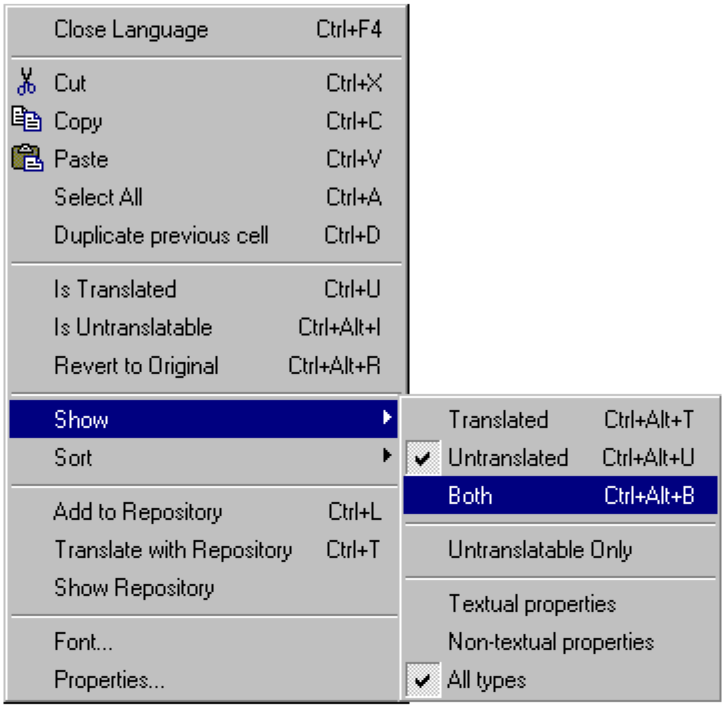
Repository path – you can select what repository database (the set of two files: <name>.lr and <name>.mem) should be used.

Translate – when the user choose "Repository | Translate language" menu item, this option defines which strings are to be translated:
◉ **All strings** – all strings are to be translated independently of their "translated" status.
◉ **Strings marked as untranslated** – program will try to translate the strings that are marked as untranslated.

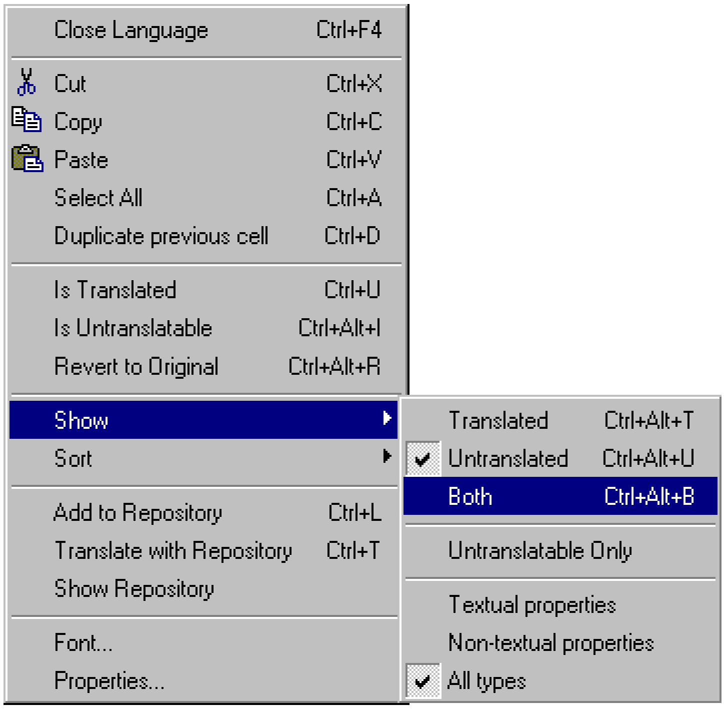
Adding an existing string – defines what happens when the user is trying to add to the Repository a translation of the string that already has some translations:
◉ **Append to Repository** – the translation (if it is unique) is automatically appended to the Repository.
◉ **Skip** – the new translation is ignored.
◉ **Ask user** – the special dialog appears, where the user can choose to append, to ignore or to replace some old translation with a new one.



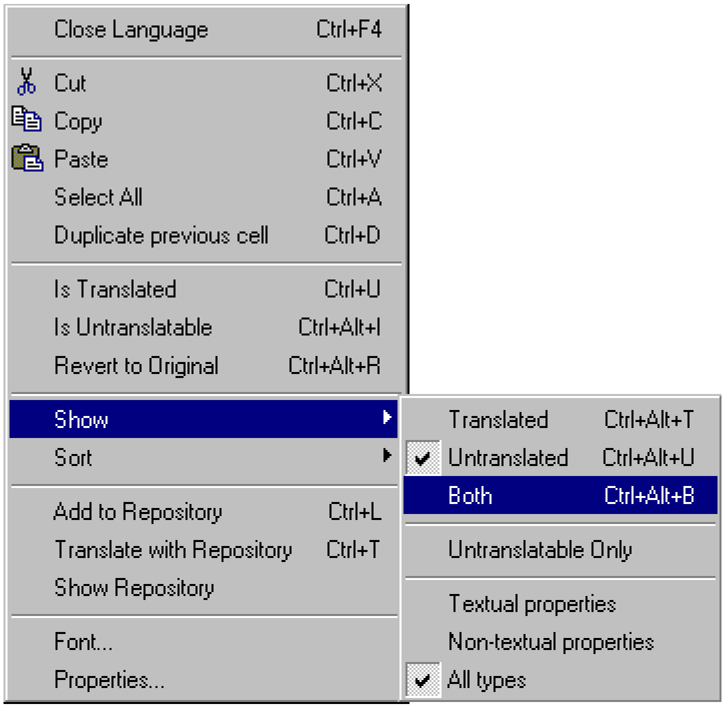
Ask user translating the cell – mark check box if you want the special dialog to appear when the string being translated has more than one translation in the Repository. Otherwise, the first found translation will be used.



Add strings that are the same as native – if marked, you will be able to add to the repository the translations that are the same as native string (i.e. the pair "Cancel" : "Cancel"). Otherwise such pairs will be omitted.



Automatically translate the language being created – mark check box if you want the "Repository | Translate language" action to be performed just after the new language is created.



Automatically refresh repository after the cell has been translated – mark check box if you want the program to automatically add the current cell to the Repository each time you have finished the cell editing.

The "Colors" tab

Active row color – the background color of the current row. Click on colored box to change it.

Untranslatable items color – the background color of the rows that are marked as untranslatable. Click on colored box to change it.

Verified items color – the background color of the rows that are marked as verified. Click on colored box to change it.

Default colors – click the button to set colors to their default values.

The "Language" tab

Select interface language – the Language Manager interface can be translated. Select one of the available languages from the list.

General shortcuts

Shortcuts for general operations allow you to:

F1	Display Language Manager help.

Ctrl + N	Create new language
Ctrl + O	Open language file
Ctrl + S	Save selected language(s)
Ctrl + F4	Close selected language(s)
Alt + X	Exit Language Manager

Cell editing shortcuts

Shortcuts for cell editing allow you to:

F2	Start editing selected cell. Cursor is set to the end of the text
Enter	Start editing selected cell. All text is selected.
Ctrl + Enter	Forcedly start editing selected cell as multiline. All text is selected.

Grid related shortcuts

These shortcuts allow you to:

Ctrl + X	Cut selected cell to clipboard
Ctrl + C	Copy selected cell to clipboard
Ctrl + V	Paste selected cell from clipboard
Ctrl + A	Select all cells in the selected columns
Ctrl + D	Copy the above cell content to the selected cell

Ctrl + U	Toggle translated/untranslated state of the selected cell(s)
Shift + Ctrl + V	Switch the "verified" status of the selected cell(s) on/off
Shift + Ctrl + I	Switch the "untranslatable" status of the selected row(s) on/off
Shift + Ctrl + R	Revert selected cell(s) value to original (native)

F5	Refresh grid

Ctrl + F	Display "Find" dialog to start searching
F3	Find next item
Alt + F3	Find previous item
Ctrl + R	Display "Replace" dialog to start replacing
Alt + G	Jump to specified line

Shift + Ctrl + T	Set filter to show items marked as translated in the selected language
Shift + Ctrl + U	Set filter to show items marked as untranslated in the selected language
Shift + Ctrl + B	Set filter to show both translated and untranslated items

Shift + Ctrl + A	Sort selected column ascending
Shift + Ctrl + D	Sort selected column descending
Shift + Ctrl + O	Display strings in their original order

Ctrl + L	Add selected cell(s) to the repository

Ctrl + T	Translate selected cell(s) using the repository

Repository form shortcuts

Shortcuts of repository form allow you to:

Ctrl + A	Select all cells in the selected columns
Ctrl + N	Add a new row
Ctrl + D	Delete selected row(s)

Language Manager uses native language files as projects.

The text file with **.leo** extension is used to save/restore project configuration. It is created when you open the project first time and is updated every time you close the project.

Open project

To open project select "File | Open Project…" menu item (the open dialog has file name extension filter set to **.ntv.lng**).
If there is no .leo file in the project folder, then the language files opening dialog should appear. Otherwise, Language Manager will automatically open language files that were open when you closed the project last time.

When the project is opened you can open only language files with the same file name located in the same folder.

Only one project can be opened in a time. When you open some project, the currently opened project is closed automatically.

You may also use some command line parameters to open project.

Reopen project

The six last opened projects are listed as "File | Reopen" menu subitems. You may select one of them to open listed project.

Close project

To close currently opened project select "File | Close project" menu item.

See also:
Working with language files

Creating new language file

When some project is opened, you may create a new language file.
Select "File | Create New Language…" menu item. The language file creating dialog should appear. Set appropriate parameters and press "OK" button:

Based On Language – indicates the language file that will be used as the source of resource strings when creating new file. This list includes all languages opened by the moment.

Language Identifier – the identifier of the new file language. For languages that have sublanguages: select the root node if you want to create file for any of the selected language locales. For example, if you choose "Spanish", the language file with **ES** abbreviation will be created and the generated DLL can be used for each of locales in **Spanish (Argentina) .. Spanish (Venezuela)** range.

English Name – the language name in English. Is set in by default when selecting the Language Identifier.

Localized Name – the name of the language in the language itself. Is set in by default when selecting the Language Identifier.

Abbreviation – the two- or three-character abbreviation of the language. Is used as a part of language file extension and as resource DLL file extension. Is set in when selecting the Language Identifier. You can't change this value.

All enumerated parameters but "English Name" and "Localized Name" are set once only and can not be changed later.

You won't be able to create a file if its abbreviation corresponds to the abbreviation of one of the already opened files.

Files saving

If language has been modified, the * sign is displayed in the right side of column header. Select "File | Save language" to save all selected languages. Or you can select "File | Save all" to save all opened files.

When you close the project and "Refresh project on close" option is set, all language files of the project are saved forcedly.

Opening and closing language files

To open the language file(s) select "File | Open Language…" menu item. The language files opening dialog should appear.

The "**Languages**" list contains only files that are located in the project folder and are not opened yet. The language files that have unsupported code page are disabled and the problem description is added.

Mark languages you want to open and press "Open" button.

Note: opened language files are protected from writing by another programs (including Language Wizard).

Select "File | Close language" to close all selected languages (except the native one).

When you try to close the modified file, Language Manager will ask you if you want to save the file being closed.

Language properties

Font

Select "Language | Font..." to change the font that is used for active language displaying and editing.

Properties dialog

Select "Language | Properties…" menu item to view language properties. The language file properties dialog appears:

Based On Language – indicates which language was used as a base during language file creating.

Language Identifier – the identifier of the file language.

English Name – the language name in English.

Localized Name – the name of the language in the language itself.

Abbreviation – the two- or three-character abbreviation of the language. Is used as a part of language file extension and as resource DLL file extension.

Version – consists of the language file version (corresponds to localization version in Language Wizard) and build (number of times the file was saved) separated by dot.

All enumerated parameters but "English Name" and "Localized Name" are read only and can not be changed.

See also:
Language files
Export & import
Options

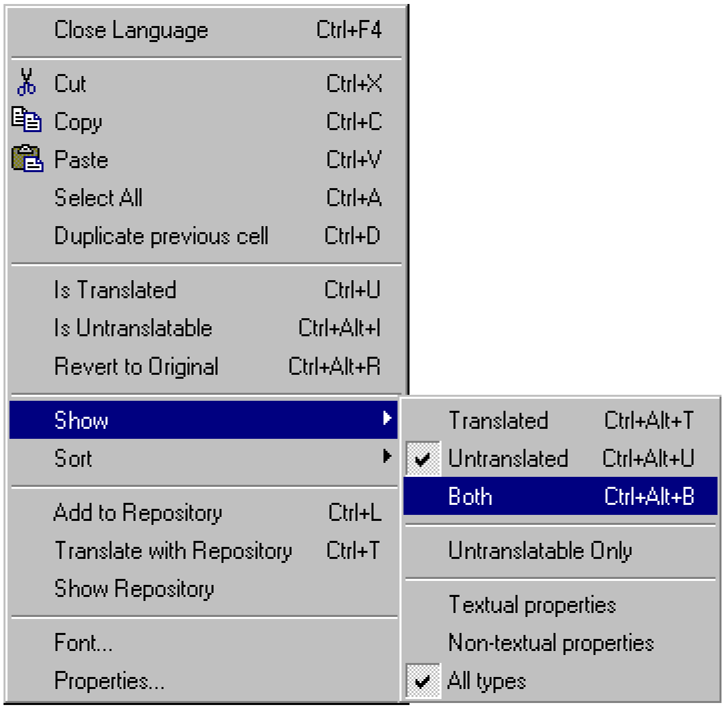
Export

You may export active language to text or XML file.
To export active language choose "File | Export to | Text…" or "File | Export to | XML…" menu item. The language files exporting dialog should appear.

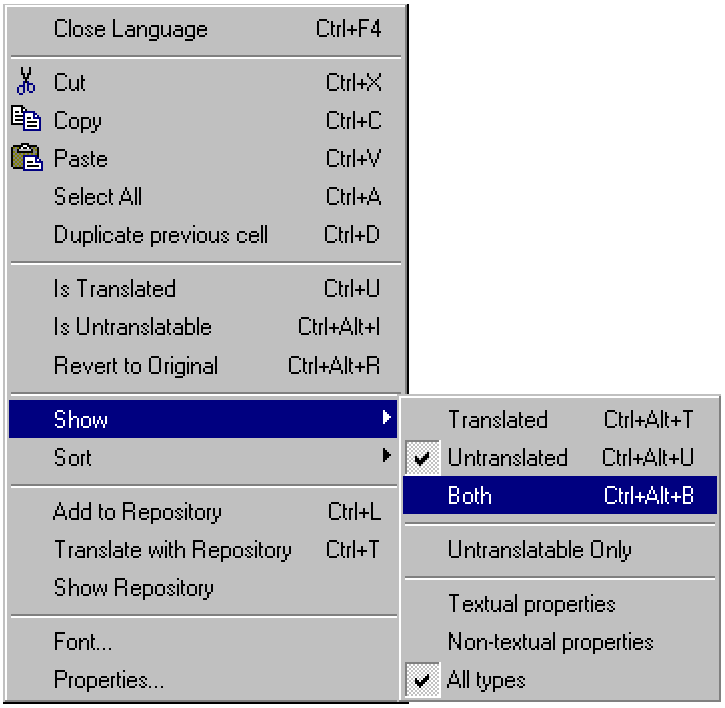
In the "**File name**" field indicate the name of text (or XML) file an active language should be exported to. Press Browse button to set file name using standard dialog.

The "**Delimiter**" field sets the symbol (comma or TAB symbol) that serves as delimiter of strings that contain carriage returns This field is hidden for exporting to XML.

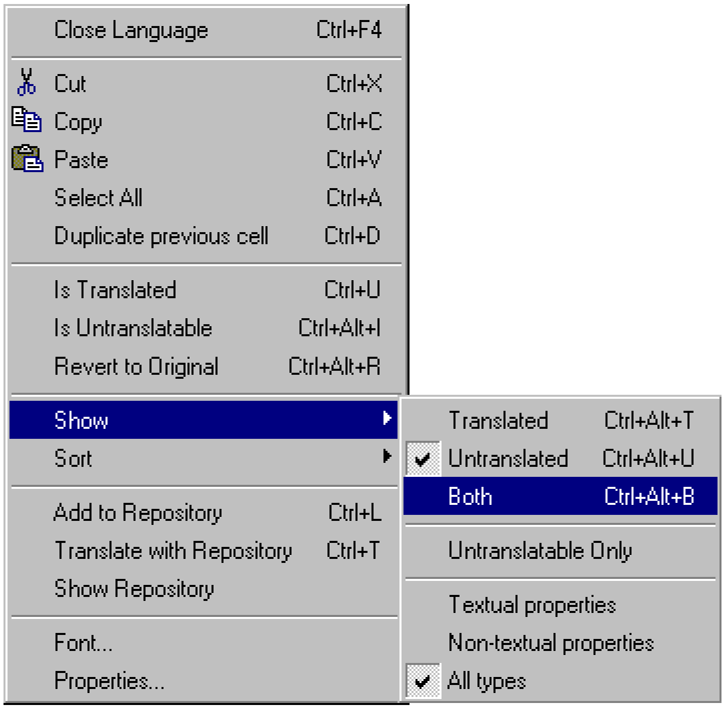
The "**Include**" field defines which identifiers are included into the target file together with properties/constants values.



"**IDs**" check box allows to include internal identifiers of properties and constants.

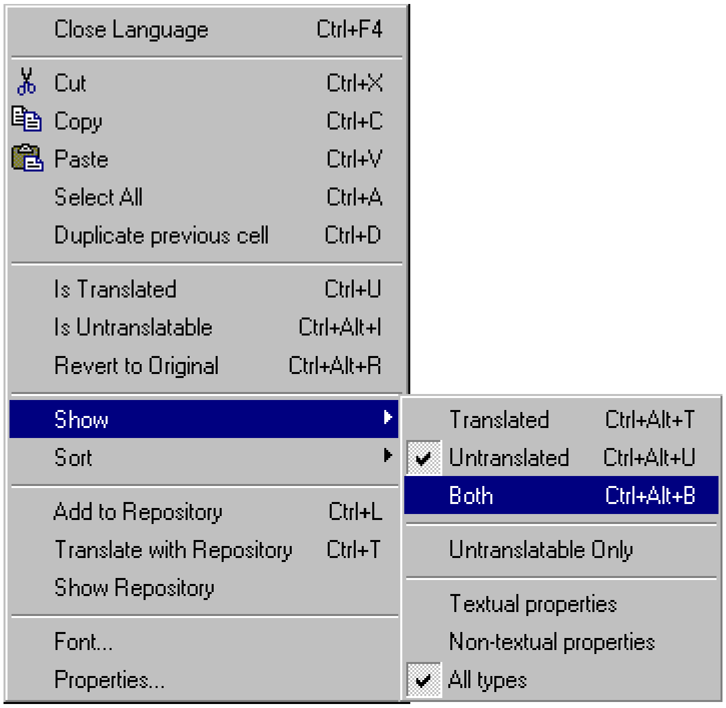


"**String IDs**" check box allows to include constant strings numbers as they appears in resource DLL.



"**Properties/constants names**" check box allows to include names of properties and constants (names include forms/units names, components names separated by dot). This check box is always checked for exporting to XML.

The "**Export**" field allows to select which items do you want to export – translated, untranslated or both.



Mark "**Export untranslatable items**" check box if you want untranslatable items to be exported.

Click "**More >>**" button if you want to exclude some forms or units from among exported ones or to change file encoding.

> "**Save as**" - select the encoding of the file from list. This filed is set to "UTF-8" and disabled for exporting to XML.
> "**Forms**" - lists all available forms. Mark only those of them that you want to be exported.
> "**Units**" - lists all available units. Mark only those of them that you want to be exported.

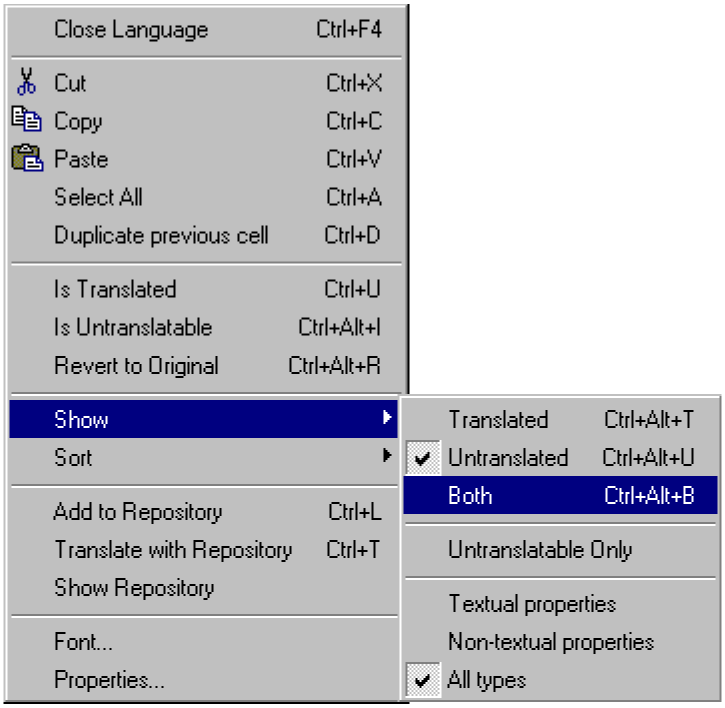
Import

You may import values to the active language from text or XML that was exported by the Language Manager before or from another language file .

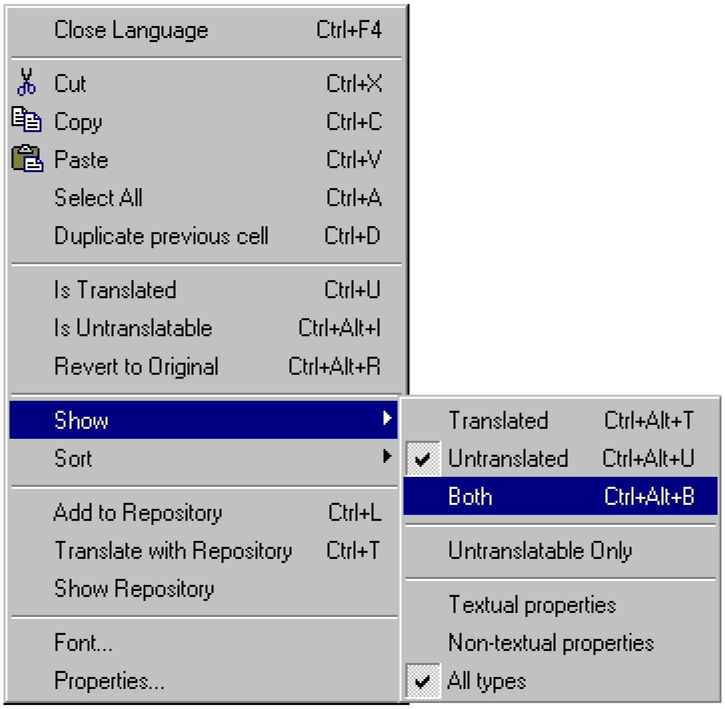
To import file select **File | Import from | Text…** or **File | Import from | XML…** or **File | Import from | Language file…**menu item. The importing dialog should appear.

NOTE: when currently selected column in the main grid is native language file column then you can import only External items. In such case **File | Import** menu item is changed to **File | Import external items from**. Only external items will be imported from the selected textual or XML file.

In the "**File name**" field indicate the name of the file you want to import from. To do it press Browse button and select file name in the standard dialog. You can not input file name manually.



"**Change translated items**" - mark check box if you want the items marked as translated to be replaced with imported values.



"**Import untranslated items**" - this item is enabled only for language file importing. Mark it if you want the untranslated items of imported language file to be imported.

The "**Synchronize by**" field indicates which parameter (the internal identifier or the property/constant name) should be used for determining of imported strings position in the active language.

Click "**More >>**" button if you want to exclude some forms or units from among imported ones.

"**Forms**" - lists all available forms. Mark only those of them that you want to be imported.
"**Units**" - lists all available units. Mark only those of them that you want to be imported.

Select the cell to be translated and change its value. Repeat this process for all strings or integers that are to be translated.

Set "Try to translate automatically" option if you want Language Manager to search the same string as you have just translated each time you enter some new value into a cell. The "Replace" toolbar may appear if such string was found.
This option doesn't work when you change a non-textual value. Moreover, the program doesn't allow you to enter wrong chars into the integer cells.

The blue triangle in the cell right bottom corner indicates that its content was not translated yet. To change its state change the cell contents or use "Edit | Is translated" menu item.

The cell can be single- or multiline depending on property type and value.

The transition into edit mode is realized with <F2>, <Enter>, mouse click or click on "…" button (if it appears in the right side of the cell).
Exit from edit mode:

- confirming changes – <Enter>, proceed to another cell or <Ctrl><Enter> (just for multiline cells)
- discarding changes – <ESC>

Besides, you can edit any cell with multi-line editor just by pressing <Ctrl><Enter> on it. In this case the cell pretends to be multiline.

If "Use Multiline Cell Edit Dialog" option is set then a special dialog appears for multiline cell editing.

If you want to revert the string value to the value it has when the language file has been created, select "Edit | Revert to original" menu item.

Above mentioned actions don't work for the cells that can not be edited.

You may translate current cell or the current language using the repository.
The strings that belong to some third-party library can be translated with "Apply libraries" tool.

Set "Move down automatically" option if you want to automatically jump to the next cell when you changed the current cell or its status (translated/untranslated, untranslatable etc.).

To view the native value of the cell being editing you can select "Edit | Show native value form" menu item. The floating form with the native value will be displayed.

Just for developers:
You may make the current row untranslatable by selecting of "Edit | Is Untranslatable" menu item.
Depending on "Show untranslatable items" option the row will be hidden or marked out with a color.

See also:
Filtering and Sorting
Using the Repository
Apply libraries
Options

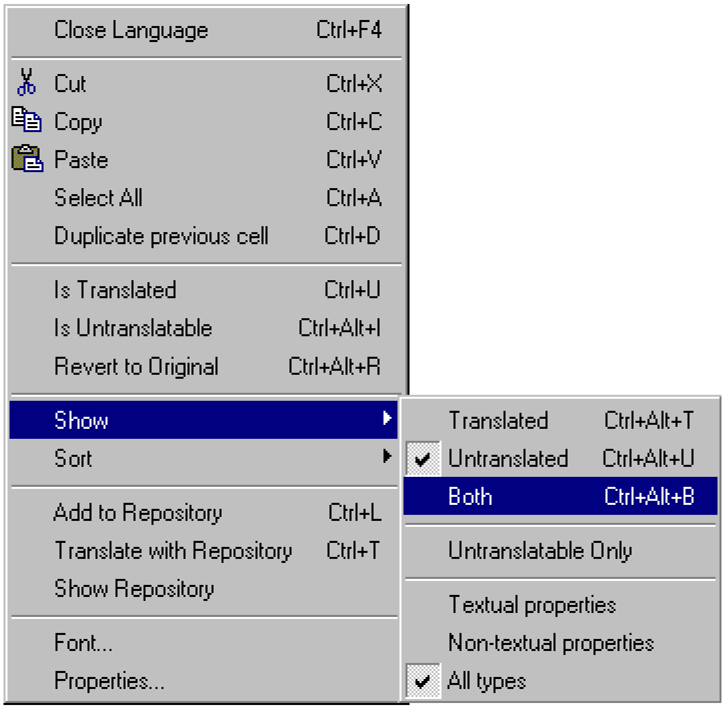
The Language Manager gives you a tool for searching and replacing strings in the active language. The searching parameters may be set using the special dialogs.

Search/replace commands are not available if more then one column is selected.

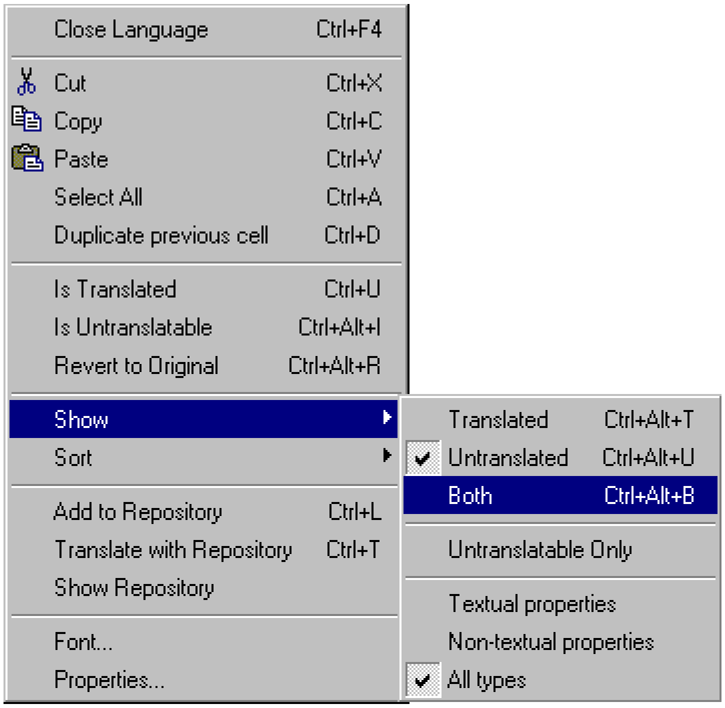
Searching

Set the proper active language and select "Search | Find…" menu item to start searching. The "**Find**" dialog appears.

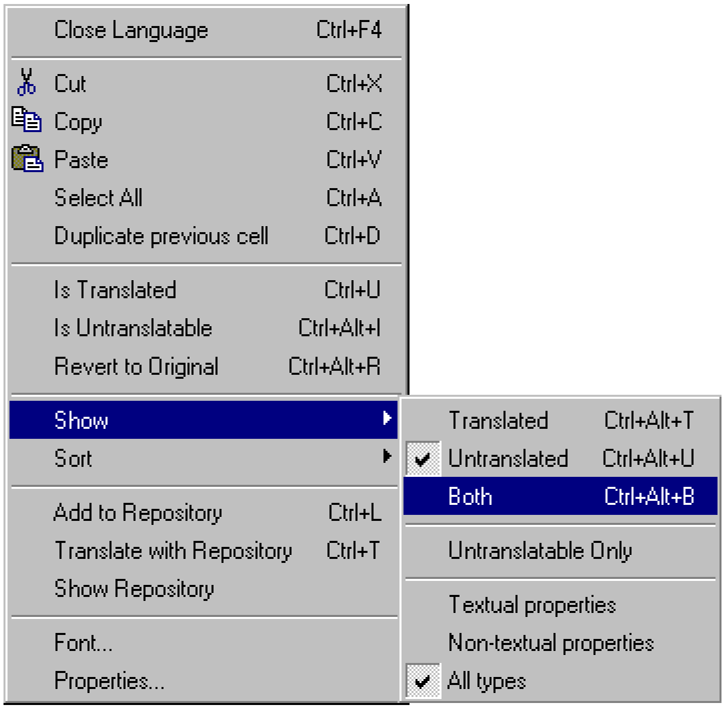
Enter the text you wish to search for into **Find what** field. This field font corresponds to active language font.



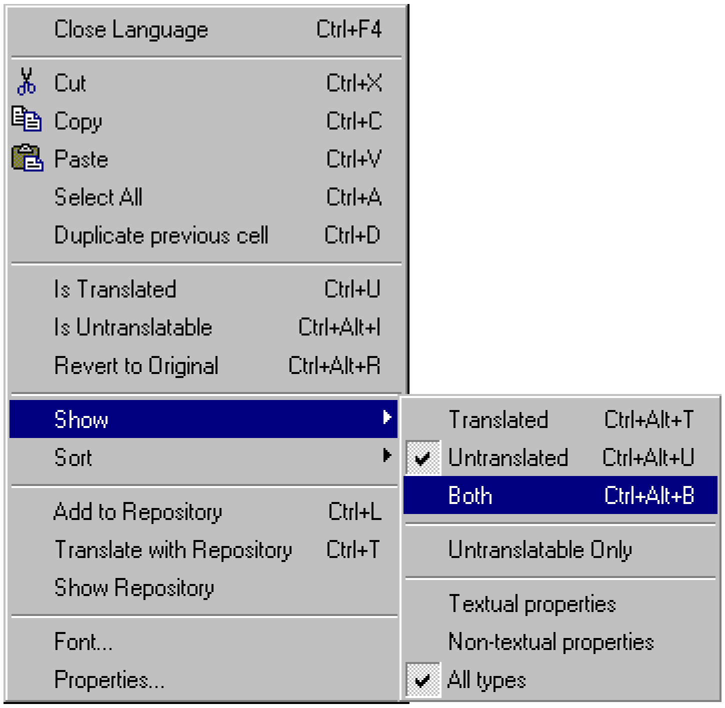
Match whole word only – mark to search for whole words only. Words are delimited with spaces, punctuation marks, brackets and arithmetic.



Match whole cell only – mark to search for whole cells only. Cancels "Match whole word only" field.



Match case – mark to perform case sensitive search. Otherwise the search is case insensitive.



Search cyclically – mark if you want to search cyclically (search from the beginning when you have reached the end).

Direction – mark appropriate radio button to set search direction ◉ **Down** (toward the last row) or ◉ **Up** (toward the first row).

Find Next / Find Prev

"Search | Find Next" command repeats the previous **Find** command using all parameters set in the "Find" dialog including the direction.
"Search | Find Prev" command does the same but in the in the opposite direction.

Replacing

Set the proper active language and select "Search | Replace…" menu item to start searching and replacing. The "**Replace**" dialog appears.

Enter into **Replace with** field text that should replace found text.
These field font corresponds to active language font.

Other parameters in the dialog are the same as in "**Find**" dialog.

Press "**Replace All**" button to close dialog and replace all text occurrences according to dialog parameters.

Press "**Start**" button to perform operated replacing. Special toolbar with the following buttons appears if "Find what" text was found:

Find Next – searches for next "Find what" text occurrence.
Find Prev – searches for previous "Find what" text occurrence.
Replace – replaces last found text with "Replace with" text and searches for next "Find what" text occurrence.

Replace All – closes dialog and replaces all text occurrences according to Replace dialog parameters.

Go to Line Number

If you know the number of the row you want to jump to, it may be easily done by selecting "Search | Go to Line Number..." menu item. Input the row number in the appeared dialog end press "OK" button.

Creating resource DLL

Language Manager allows you to create resource DLLs for the languages you translate to. To make this option available you need to select PE (portable executable) file that is used for DLLs generating. Choose "File | Select PE file..." menu item and select your application file (EXE, BPL or DLL) in the appeared dialog.

Select the column(s) you want to create resource DLLs for. Then select "File | Create DLL" menu item. If the native language is among the selected languages, it is ignored.

If "File | Create DLL" menu item is present but disabled then the reason is one of the following:

- You are trying to create DLL for native language;
- The selected PE file does not exist or is not valid localized PE file;
- The opened project is a library.

Note: You may use command line parameter "**-dll**" to create resource DLLs in a batch mode.

Synchronizing resource DLLs

You can also synchronize (recreate) all resource DLLs you created before. Select "File | Synchronize DLLs" menu item when some project is opened to do it.

If there are modified but not saved files in your open project then saving changes query should appear. Press "Cancel" to cancel synchronizing.

Note: if you decide not to save files then unsaved changes will not come into refreshed resource DLLs. The strings will have the same values as they have in the grid.

Error messages

If there were errors during synchronizing then synchronization dialog should appear. Each string represents the name of one of processed DLLs and synchronization result. "**Ok**" mark says that synchronization was successful. Other possible messages are described below:

- "**Invalid language file**" – program can not open the existing language file. Most probably, its format doesn't correspond to language files format.
- "**Is native**" – DLL contains the native mark (once it was created as native), though it has not **.ntv** extension.
- "**Can not delete file**" – program can not delete current file for its refreshing. Possibly, this file is opened by another program or has "read only" attribute.
- "**Can not create DLL**" – there was an error during language DLL creating (this process includes native DLL loading, language file loading, language DLL saving).
- "**Unknown Error**" – there is an error not described above. If error repeats and you can not find the cause then let support personnel know about it.

Note: You may use command line parameter "**-syncdlls**" or "**-syncall**" to synchronize resource DLLs in a batch mode.

The **repository** is a multilingual dictionary that is filled in by the user (translator) and is served for (semi) automatic translating. This tool is especially convenient for the translators that translates several applications with the similar interface elements.

The repository structure

You may think about the repository as about the table, where the columns are the languages, and the rows are the translations of the same string. The repository form displays this structure visually.

Using the repository

Selecting the repository

You may have several different repositories for different purposes. Select "Repository | Open repository..." menu item to choose the repository (**.lr** file) you want to work with.

The path to currently selected repository is stored in the Registry. If there is no information or there are no repository files in the mentioned folder, then Language Manager creates a new empty repository in the "<LocalizerRootFolder>\Repository" folder.

Note: If you need to copy or move the repository to another folder, note that it consists of two files - **.lr** and **.mem**.

Translating cells using the repository

The main goal of the repository is to simplify the translation process.

To translate the selected cells with a value stored in the repository choose "Repository | Translate cell" menu item. For each selected cell the program searches the repository for the translations of the native value in the grid row. If only one translation was found Language Manager changes the cell value. Otherwise, if there are several suitable translations, the special dialog "Translate with repository" appears to let user select which one of translations to use.

> **How does it search the repository?**
>
> First, the Language Manager tries to find in the repository the language that corresponds to native language. If it's not found, then LM searchs for its "next-of-kin" (i.e. "English" for "English (Canada)" or "English (United States)" for "English").
> Then we find in the repository the language that corresponds to active language. The procedure is the same.
> The found native language is searched for the native string - first case sensitive, then, if not found, insensitive.
> All nonempty translations of the found entries of native string are the result of the searching.

Use "Repository | Translate language" menu item to translate all the values of the current language with a values stored in the Repository. The program searches the Repository for the translations of the native language values in the corresponding rows. Actually, it does the same as discribed above for each string of the language file.

There is a "Repository" tab in the "Options" dialog that allows you to manage the repository behaviour. For example, you can choose if to translate all strings or just untranslated ones; to ask user which translation to use or just automatically put the first found translation etc.

Packing the repository

When you add/delete rows and languages to/from repository, it is possible that it will contain some empty rows and some duplicate rows. To delete superfluous rows select "Repository | Pack" menu item.

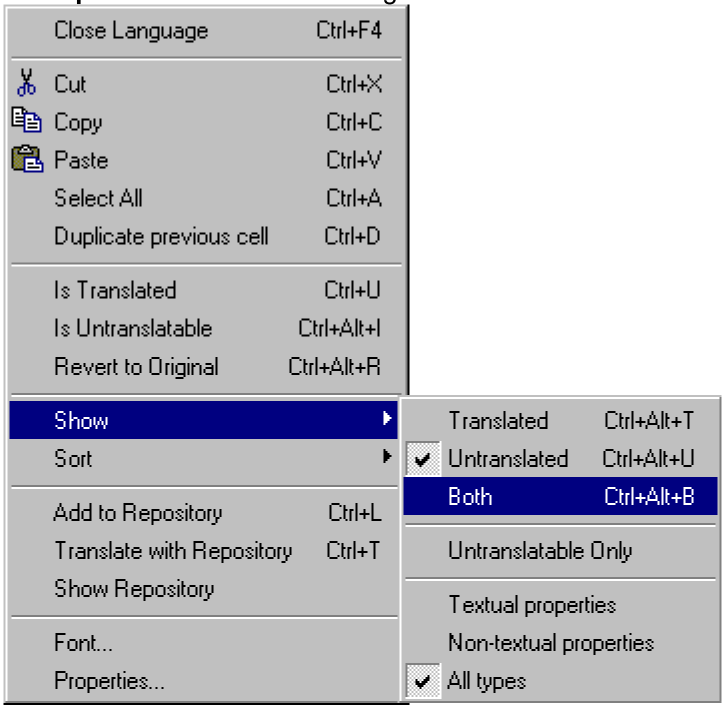
See also:
Repository
Filling the repository
Repository form
Repository export & import
Options

To add the values of the selected cells (or, rather, the pairs "native value - selected value") into the repository select "Repository | Add cell" menu item. The cells values will be added as a translations of the native language values.

If native value being added already has the translation(s) in the repository, Language Manager behaviour is defined by the options set. The possible variants are the following:

◉ **Append to Repository** – the translation (if it is unique) is automatically appended to the Repository.

◉ **Skip** – the new translation is ignored.



Ask user – the "Refresh repository" dialog appears, where the user can choose to append, to ignore or to replace some old translation with a new one.

To add all translated values of the active language to the repository select "Repository | Refresh by language" menu item. The cells values will be added as a translations of the native language values in the corresponding rows.
Note: Only strings marked as translated are added.

The other way to fill the repository is to import some XML or MS Glossary file.

You can also fill the repository manually, using the repository form.

Note: Only textual properties can be added to the repository.

See also:
Using the repository
Repository form
Repository export & import
Options

Repository form

Select "Repository | Show Repository" menu item to show the modal form that allows you to edit the repository. Each column of the grid represents one language. You can edit grid cells as in the main Language Manager form. The grid also supports multiselection.

Show / hide columns

The toolbar in the top of the form allows you to show / hide separate languages (columns). Push the corresponding button in to make language visible. Push it out to hide the language. You can press "All" button to make all availale languages visible.

Context menu

Use context menu to add or delete repository items or languages.

Add record - adds an empty record to the repository. A new row appears in the end of grid.

Delete record(s) - deletes all selected records from the repository. The confirmation dialog appears before the deleting.

Add language - adds a new language to the repository (and a new column to the grid). The "Add new language" dialog appears to let you choose the language you want to add.

Delete language(s) - deletes all selected languages from the repository. The confirmation dialog appears before the deleting.

Properties... - you may change the language represented by the active column. Select this menu item and choose another language in the dialog appeared. You can't choose the language that is already exist in the repository.

Exporting to XML file

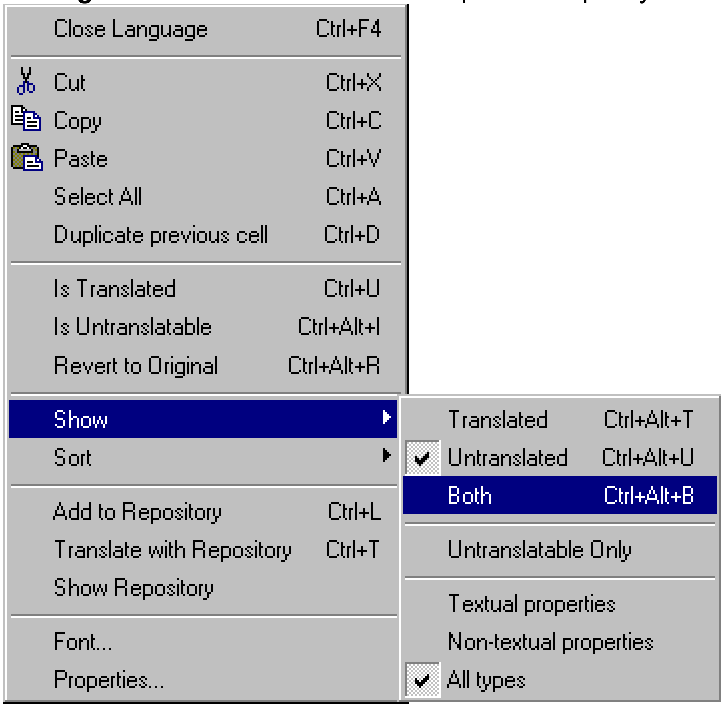
The repository may be exported to the XML file. The XML file format is compatible with the Borland's one, so you can use files exported by the Language Manager in the Delphi Language Repository.

Select "Repository | Export XML…" menu item and set proper parameters in appeared dialog:

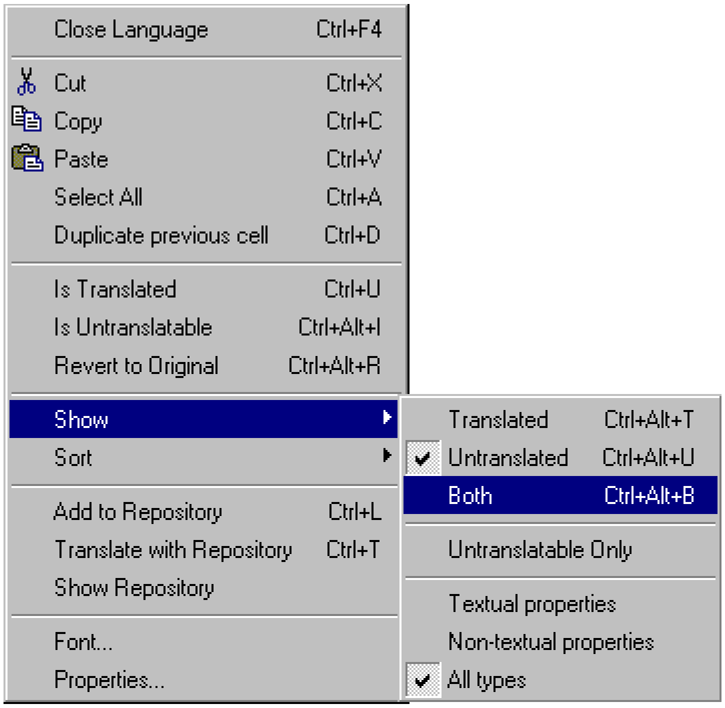
"Base locale" – select the language that will be used as native when you try to import XML file into Delphi Language Repository. This language is mentioned as native in the XML file.

"Languages to be exported" – displays all the languages the repository contain. Mark the languages you wish to be exported. The language chosen in the "Base locale" field is marked by default.

"Strings format" – select one of two options to specify how should the strings be written to the file.



"C - style" – the strings are quoted as "string", the special characters are encoded through \ sign (\n, \" etc.).



"DFN (Delphi) - style" – the strings are quoted as 'string', the special characters are encoded through # sign (#13, #10, #39 etc.).

"File name" – select or enter the file name to export to.

Importing XML file

You may import the translations from the XML file to the Repository. The XML file format is compatible with the Borland's one, so you can import files exported by the Delphi Language Repository.

Select "Repository | Import XML…" menu item and select the file to import from in appeared dialog.

After you selected the file the "OK" button may stay disabled. It means that selected file has wrong format.

If some imported items already have translations in the repository, the "Refresh repository" dialog appears and ask you what to do with such items. See "Filling the Repository" topic for details.

Importing MS Glossary

The Microsoft Glossaries are the text files that contain a two-languages dictionary. Language Manager allows you to import MS Glossaries to the repository.

Select "Repository | Import MS Glossary..." menu item and set proper parameters in appeared dialog:

"File name" – select or enter the file name to import from.

"Language 1", **"Language 2"** – since MS Glossary format does not provide the information about the languages it includes, you need to select the languages manually. The languages you select will be used (and created if don't exist) to import to.

Applying of the libraries is initiated by "Libraries | Apply Libraries…" menu item.

This process consists in the substitution of the language files strings to corresponding strings of the libraries files according to synchronization rules. Synchronization rules (by unit/constant name, by string ID) are set during creating of the library (using the **Library Manager** tool). Libraries selected by the user are applied to all open language files in the project (except the native file).

At first, the dialog of selecting of the libraries that are to be applied appears.
Mark the libraries that you want to apply and press OK button.

The applying process is displayed on the special progress-indicator.

In the case if there were any errors during working the "Apply Libraries" dialog appears. It contains the results of applying of the libraries to the each open language in the project (except the native).
For example, "Report Builder ….. Language file doesn't exist" line in the Russian language section says that the file ReportBuilder.RUS was not found in the Report Builder library folder (External\ReportBuilder).

You can cancel introduced changes – just close the language file without saving changes.

You can pass parameters into Language Manager to perform some specific actions. Possible parameters are described below.

langmngr.exe ***filename.ntv.lng***
Language Manager tries to open *filename.ntv.lng* project after starting.

langmngr.exe -last
Language Manager tries to open last opened project after starting.

langmngr.exe -lib ***filename.ntv.lng***
Language Manager tries to open *filename.ntv.lng* library after starting.

langmngr.exe -synclangfiles ***filename.ntv.lng***
Language Manager synchronizes (reopens and saves) all language files found in the *filename.ntv.lng* project folder. User interface is not shown. Language Manager terminates when synchronization is complete.

langmngr.exe -syncdlls -pefile:***pefilename.ext filename.ntv.lng***
Language Manager recreates all resource DLLs found in the *filename.ntv.lng* project folder. The action is the same as "Synchronize DLLs" when *filename.ntv.lng* project is open. User interface is not shown. Language Manager terminates when synchronization is complete.
The **-pefile:** parameter indicates the PE file (EXE, BPL, DLL) that is used to generate resource DLLs. If this parameter is omitted then Language Manager tries to use *filename.exe* file.

langmngr.exe -syncall -pefile:***pefilename.ext filename.ntv.lng***
Language Manager performs two operations described above - ***synclangfiles*** and ***syncdlls*** - sequentially. User interface is not shown. Language Manager terminates when synchronization is complete.
The **-pefile:** parameter indicates the PE file (EXE, BPL, DLL) that is used to generate resource DLLs. If this parameter is omitted then Language Manager tries to use *filename.exe* file.

langmngr.exe -dll -pefile:***pefilename.ext filename.*.lng***
Language Manager creates the resource DLL for the mentioned language file. User interface is not shown. Language Manager terminates when creation is complete. The files *filename.ntv.lng* and *filename.ntv* must exist in the same folder.
The **-pefile:** parameter indicates the PE file (EXE, BPL, DLL) that is used to generate resource DLL. If this parameter is omitted then Language Manager tries to use *filename.exe* file.

langmngr.exe -delatom
Language Manager allows only one copy of application to be launched. The special global atom is added on application start and is deleted on close. But if Language Manager is aborted unexpectedly, the atom is not deleted and the next time application will not start. To start Language Manager after abnormal termination use *-delatom* parameter.
This parameter can be combined with any other.

langmngr.exe -debug
This parameter is used to show some additional information and to make a few options available. I.e. the internal properties and constants ID will be displayed, the project and repository repairing menu items will become available.

Localizer restricts the use of unregistered copies.

The limitations differ slightly for Standard and OnFly editions.

Localizer Standard: if the project is localized using the unregistered copy of Language Wizard then only 30 properties on each form and 20 constants of each unit are translated in the resource DLLs created by Language Manager. In contrast to normal mode the strings marked as untranslated are not put to the DLL and are not counted.
Besides, it displays the reminder every time you call "Refresh Language Files" operation.

To get Localizer Standard registered you need to apply a special registration key that is send to you after purchasing.

Localizer OnFly: includes all the limitations of Standard edition. Moreover, if you use unregistered LocOnFly unit in your application, it will not start without Delphi IDE launched.

To get Localizer OnFly registered you need to apply the key as described in Standard section. In addition, you need to overwrite some Localizer trial version units by the corresponding units from the full version which can be downloaded from our site using the login and password you get after purchasing.

1. All rights to Language Manager (mentioned below as the Product) are the property of Korzh.com (mentioned below as the Author).
2. The Product may be freely distributed provided that the distribution package is not modified. No natural person or body corporate may charge a fee for the distribution of Product without written authorization from the copyright holder.
3. The Product is distributed "AS IS". The Author expressly disclaims any warranty for the Product. The Author is not liable for any damage, caused by use or inability to use the Product.
4. The Author states that each part and component of the Product were (or will be) developed by the Author or its employees or are used with permission of the original authors and that no part of the Product were (and will be) borrowed without appropriate permission.
5. This Agreement can be broken only by mutual consent (agreement) of the Author and User.

XML parsing - the destructor.de XML Parser (http://www.destructor.de/xmlparser/index.htm)

Code samples:
David Hildingsson
Ralf Steinhaeusser

-- **Korzh.com** --

E-mail: locsupport@korzh.com
Homepage: http://www.korzh.com
Product page: http://www.korzh.com/delphi/localizer