

Описание библиотеки CmpDynamicText

1. Основные сведения

Системная библиотека **CmpDynamicText** используется для работы со списками текстов. В частности, она может пригодиться в двух случаях:

- при необходимости считывать записи из списков текстов в программу;
- при необходимости обновления списков текстов в контроллере «на лету».

Ниже описаны некоторые функции библиотеки.

Для каждой функции приведен пример работы с ней. В примерах используется список текстов **MyTextList**:

ID	По умолчанию	Russian	English
0	nebo	небо	sky
1	more	море	sea
2	oblaka	облака	clouds

Рис. 1. Содержимое списка текстов **MyTextList**

Список текстов содержит три языка – дефолтный (транслит), русский и английский.

Определить, какой из них будет выбран в проекте после его загрузки в контроллер, можно в **Менеджере визуализации**:

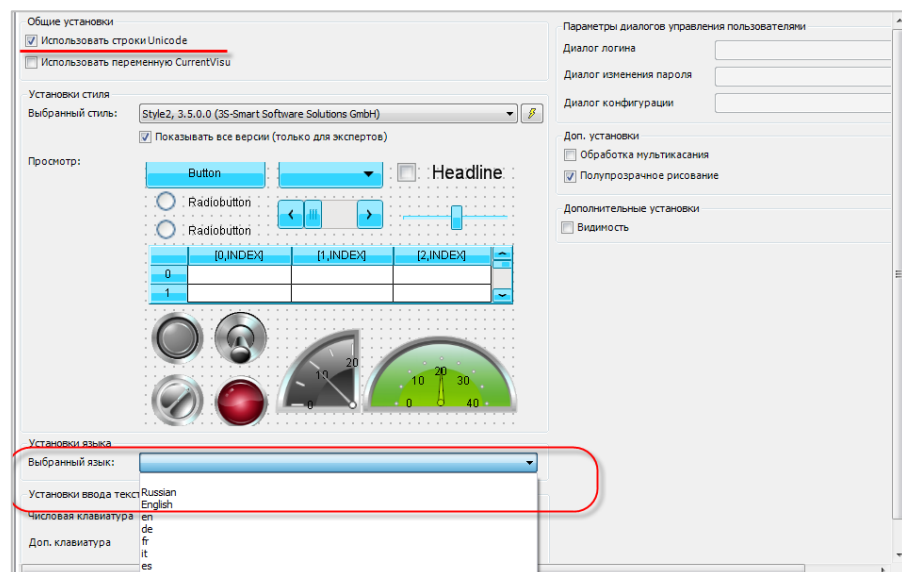


Рис. 2. Выбор языка проекта в установках **Менеджера визуализации**

Предположим, выбран дефолтный язык (пустая строка). Тогда после загрузки проекта в элементах, где настроены динамические тексты, отобразятся тексты на дефолтном языке (предварительно еще раз посмотрите на рис. 1 и рис. 2):

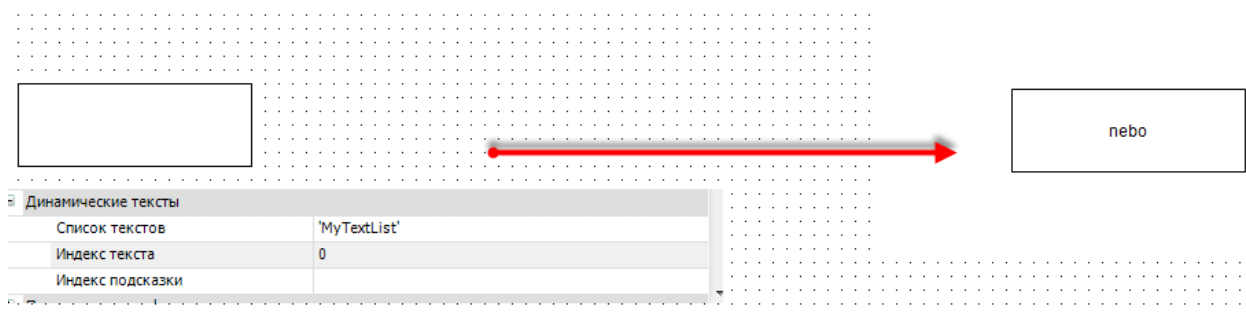


Рис. 3. Отображение динамических текстов

Вполне вероятно, что потребуется менять язык в процессе работы проекта. Это можно сделать одним из нескольких способов:

1. привязать к кнопке действие Изменить язык:

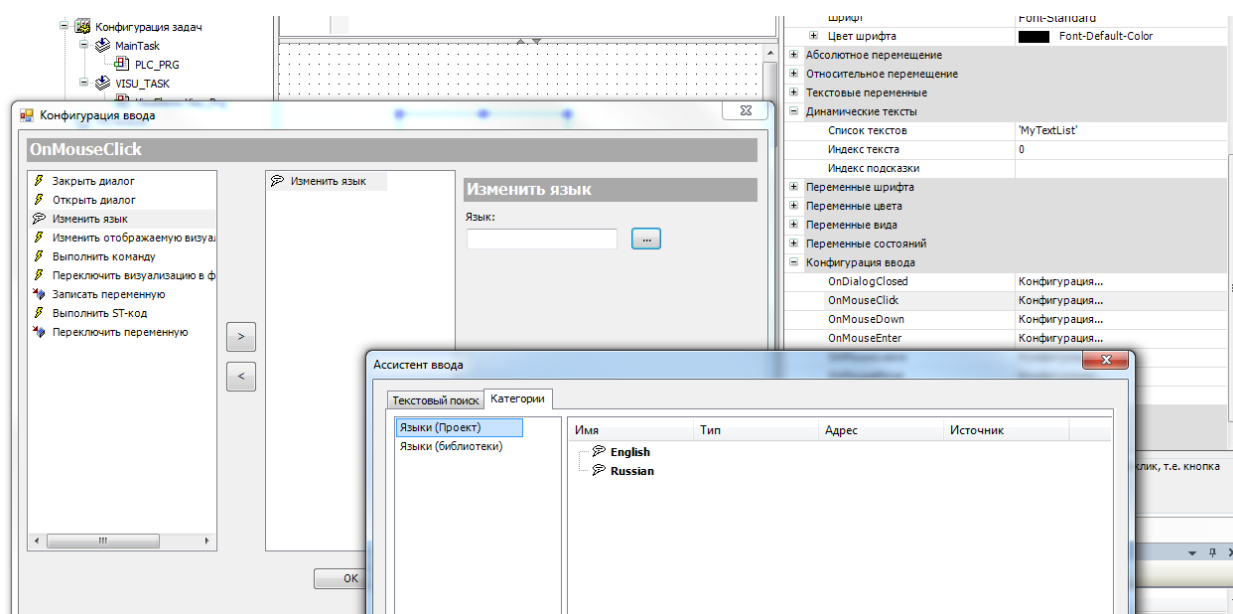


Рис. 4. Изменение языка проекта

Правда, таким образом нельзя переключить язык на дефолтный.

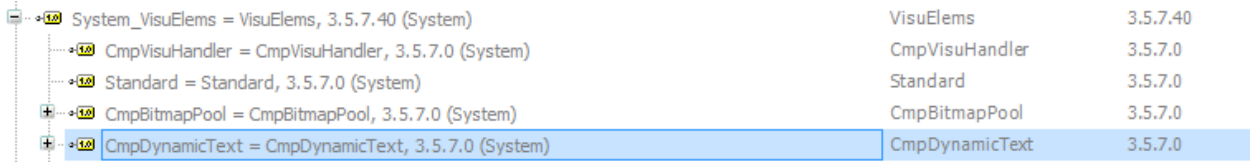
2. Менять язык из программы с помощью системной переменной VisuElems.CURRENTLANGUAGE:

```
1 | VisuElems.CURRENTLANGUAGE:='russian';
```

Рис. 5. Использование системной переменной VisuElems.CURRENTLANGUAGE

2. Добавление библиотеки в проект

Вообще, если в проекте есть визуализация, то библиотека **CmpDynamicText** в нем уже присутствует – в недрах библиотеки **VisuElems**:



System_VisuElems = VisuElems, 3.5.7.40 (System)	VisuElems	3.5.7.40
CmpVisuHandler = CmpVisuHandler, 3.5.7.0 (System)	CmpVisuHandler	3.5.7.0
Standard = Standard, 3.5.7.0 (System)	Standard	3.5.7.0
CmpBitmapPool = CmpBitmapPool, 3.5.7.0 (System)	CmpBitmapPool	3.5.7.0
CmpDynamicText = CmpDynamicText, 3.5.7.0 (System)	CmpDynamicText	3.5.7.0

Рис. 6. Встроенные библиотеки для **VisuElems**

Поэтому можно вызывать функции библиотеки следующим образом:

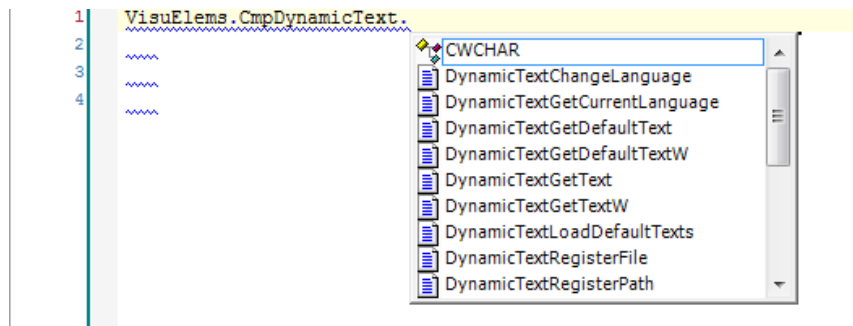


Рис. 7. Использование функций встроенной библиотеки

Но можно, конечно, добавить библиотеку в проект и в явном виде – тогда не придется каждый раз указывать полный путь.

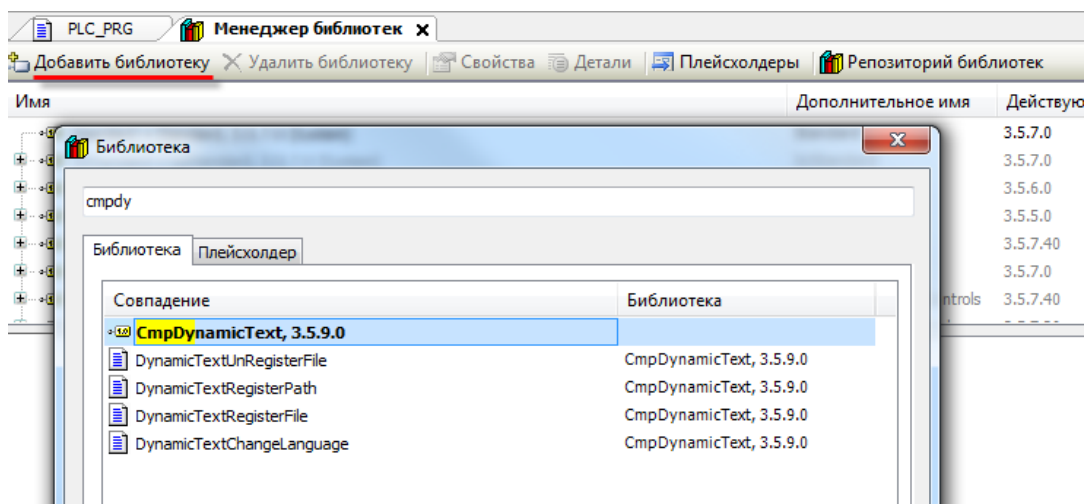


Рис. 8. Добавление библиотеки в проект

Версия библиотеки не должна превышать версию используемого таргет-файла.

3. Описание функций библиотеки

3.1. DynamicTextGetDefaultText

Используется для считывания записи из заданного списка текстов по заданному **ID** в том случае, если текущий язык проекта – дефолтный.

	Название	Тип	Описание
вход	pstTextList	POINTER TO STRING	Имя списка текстов.
вход	pstTextIndex	POINTER TO STRING	ID записи списка текстов.
выход	(имя функции)	POINTER TO STRING	Считанный текст.

The screenshot displays a table of variables and a corresponding code snippet. The table, titled "Device.Application.PLC_PRG", lists the following variables:

Выражение	Тип	Значение	Подготовленное ...	Адрес
sGetCurrentLanguage	STRING	"		
sTextList	STRING	'mytextlist'		
sVar1	STRING	'nebo'		
sVar2	STRING	'more'		
sVar3	STRING	'oblaka'		

Below the table, the code snippet shows the following lines:

```
1 sGetCurrentLanguage := VisuElems.CURRENTLANGUAGE ;  
2  
3 sVar1 := DynamicTextGetDefaultText (ADR(sTextList), ADR('0'))^ ;  
4 sVar2 := DynamicTextGetDefaultText (ADR(sTextList), ADR('1'))^ ;  
5 sVar3 := DynamicTextGetDefaultText (ADR(sTextList), ADR('2'))^ ;  
6
```

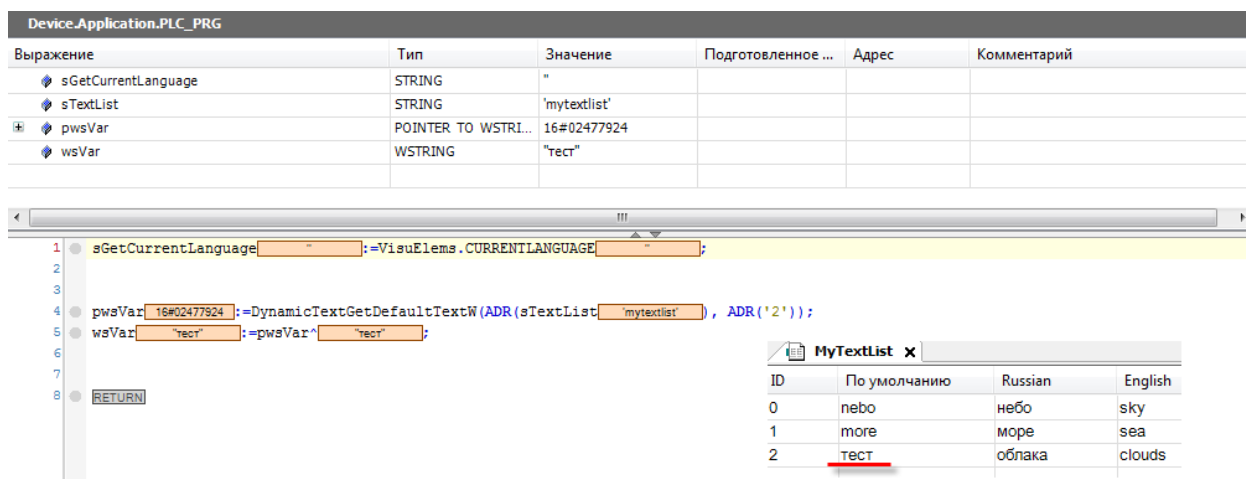
Рис. 9. Пример работы с функцией **DynamicTextGetDefaultText**

Следует заметить, что в режиме отладки невозможно присвоить переменной **VisuElems.CURRENTLANGUAGE** пустое значение (""). В этом случае для установки дефолтного языка достаточно ввести в нее что угодно, кроме названия языка, существующего в проекте.

3.2. DynamicTextGetDefaultTextW

Работает аналогично [DynamicTextGetDefaultText](#), но используется для **Юникода** – т.е., в частности, для кириллического текста. Следует помнить, что для работы с кириллицей используется тип **WSTRING** и устанавливается галочка **Использовать строки Unicode** в **Менеджере визуализации**.

Немного отредактируем исходный список текстов, чтобы показать, как использовать эту функцию:



The screenshot shows a PLC editor window titled "Device.Application.PLC_PRG". It contains a table of variables and a list of code lines. The variable table is as follows:

Выражение	Тип	Значение	Подготовленное ...	Адрес	Комментарий
sGetCurrentLanguage	STRING	"			
sTextList	STRING	'mytextlist'			
pwsVar	POINTER TO WSTR...	16#02477924			
wsVar	WSTRING	"тест"			

The code lines are:

```
1 sGetCurrentLanguage := VisuElems.CURRENTLANGUAGE;  
2  
3  
4 pwsVar[16#02477924] := DynamicTextGetDefaultTextW(ADR(sTextList), ADR('2'));  
5 wsVar["тест"] := pwsVar["тест"];  
6  
7  
8 RETURN
```

To the right, a table titled "MyTextList" is shown:

ID	По умолчанию	Russian	English
0	nebo	небо	sky
1	more	море	sea
2	тест	облака	clouds

Рис. 10. Пример работы с функцией **DynamicTextGetDefaultTextW**

3.3. DynamicTextGetText

Используется для считывания записи из заданного списка текстов по заданному ID в том случае, если текущий язык проекта – любой, кроме дефолтного.

	Название	Тип	Описание
вход	pstTextList	POINTER TO STRING	Имя списка текстов.
вход	pstTextIndex	POINTER TO STRING	ID записи списка текстов.
выход	(имя функции)	POINTER TO STRING	Считанный текст.

The screenshot displays a software interface for a PLC program. At the top, a tab labeled 'PLC_PRG' is visible. Below it, the title bar reads 'Device.Application.PLC_PRG'. A table lists the following variables:

Выражение	Тип	Значение	Подготовленное ...
sGetCurrentLanguage	STRING	'english'	
sTextList	STRING	'mytextlist'	
sVar1	STRING	'sky'	
sVar2	STRING	'sea'	
sVar3	STRING	'clouds'	

Below the table, a code editor shows the following Ladder Logic (LAD) code:

```
1 sGetCurrentLanguage 'english' := VisuElems.CURRENTLANGUAGE 'english';
2
3 sVar1 'sky' := DynamicTextGetText (ADR (sTextList 'mytextlist'), ADR ('0')) ^;
4 sVar2 'sea' := DynamicTextGetText (ADR (sTextList 'mytextlist'), ADR ('1')) ^;
5 sVar3 'clouds' := DynamicTextGetText (ADR (sTextList 'mytextlist'), ADR ('2')) ^;
6
```

Рис. 11. Пример работы с функцией **DynamicTextGetText**

3.4. DynamicTextGetTextW

Работает аналогично [DynamicTextGetText](#), но используется для Юникода – т.е., в частности, для кириллического текста. Следует помнить, что для работы с кириллицей используется тип **WSTRING** и устанавливается галочка **Использовать строки Unicode** в Менеджере визуализации.

Device.Application.PLC_PRG			
Выражение	Тип	Значение	Подготовленное ...
sGetCurrentLanguage	STRING	'russian'	
sTextList	STRING	'mytextlist'	
pwsVar1	POINTER TO WSTRI...	16#06C91210	
pwsVar2	POINTER TO WSTRI...	16#06C9121A	
pwsVar3	POINTER TO WSTRI...	16#06C91224	
wsVar1	WSTRING	"небо"	
wsVar2	WSTRING	"море"	
wsVar3	WSTRING	"облака"	


```
1  sGetCurrentLanguage 'russian' :=VisuElems.CURRENTLANGUAGE 'russian';
2
3  pwsVar1 16#06C91210 :=DynamicTextGetTextW(ADR(sTextList 'mytextlist'), ADR('0'));
4  pwsVar2 16#06C9121A :=DynamicTextGetTextW(ADR(sTextList 'mytextlist'), ADR('1'));
5  pwsVar3 16#06C91224 :=DynamicTextGetTextW(ADR(sTextList 'mytextlist'), ADR('2'));
6
7  wsVar1 "небо" :=pwsVar1^ "небо";
8  wsVar2 "море" :=pwsVar2^ "море";
9  wsVar3 "облака" :=pwsVar3^ "облака";RETURN
```

Рис. 12. Пример работы с функцией **DynamicTextGetTextW**

3.5. DynamicTextReloadTexts

Эта функция подгружает записи из списков текстов в проект – что позволяет менять списки текстов в процессе работы программы. Легче показать это на примере. Пусть все будет настроено следующим образом:

The screenshot shows the SIMATIC Manager interface with the variable declaration table and the ladder logic for the `DynamicTextReloadTexts` function.

Выражение	Тип	Значение	Подготовленное ...	Адрес	Комментарий
xReload	BOOL	FALSE			
sGetCurrentLanguage	STRING	'russian'			
sTextList	STRING	'mytextlist'			
pwsVar1	POINTER TO WSTRI...	16#06C91210			
pwsVar2	POINTER TO WSTRI...	16#06C9121A			
pwsVar3	POINTER TO WSTRI...	16#06C91224			
wsVar1	WSTRING	"небо"			
wsVar2	WSTRING	"море"			
wsVar3	WSTRING	"облака"			


```

1 sGetCurrentLanguage := VisuElems.CURRENTLANGUAGE;
2
3 pwsVar1 := DynamicTextGetTextW(ADR(sTextList), ADR('0'));
4 pwsVar2 := DynamicTextGetTextW(ADR(sTextList), ADR('1'));
5 pwsVar3 := DynamicTextGetTextW(ADR(sTextList), ADR('2'));
6
7 wsVar1 := pwsVar1;
8 wsVar2 := pwsVar2;
9 wsVar3 := pwsVar3;
10
11 IF xReload THEN
12   DynamicTextReloadTexts();
13 END_IF;
  
```

ID	По умолчанию	Russian	English
0	nebo	небо	sky
1	more	море	sea
2	oblaka	облака	clouds

Рис. 13. Пример работы с функцией `DynamicTextReloadTexts`

Отредактируем список текстов. В СПК он лежит в папке `/mnt/ufs/root/codesys_wrk/visu`.

```

mytextlist — Блокнот
Файл  Правка  Формат  Вид  Справка
Id; Default; Russian; English
0; небо; a тут; sky
1; море; новые; sea
2; облака; тексты; clouds
;;;
  
```

Рис. 14. Отредактированный список текстов

Теперь в программе присвоим значение **TRUE** переменной **xReload**:

Device.Application.PLC_PRG				
Выражение	Тип	Значение	Подготовленное ...	A
xReload	BOOL	TRUE		
sGetCurrentLanguage	STRING	'russian'		
sTextList	STRING	'mytextlist'		
+ pwsVar1	POINTER TO WSTRI...	16#06C91210		
+ pwsVar2	POINTER TO WSTRI...	16#06C9121C		
+ pwsVar3	POINTER TO WSTRI...	16#06C91228		
wsVar1	WSTRING	"а тут"		
wsVar2	WSTRING	"новые"		
wsVar3	WSTRING	"тексты"		


```
1 sGetCurrentLanguage "russian" :=VisuElems.CURRENTLANGUAGE "russian";
2
3 pwsVar1 16#06C91210 :=DynamicTextGetTextW(ADR(sTextList "mytextlist"), ADR('0'));
4 pwsVar2 16#06C9121C :=DynamicTextGetTextW(ADR(sTextList "mytextlist"), ADR('1'));
5 pwsVar3 16#06C91228 :=DynamicTextGetTextW(ADR(sTextList "mytextlist"), ADR('2'));
6
7 wsVar1 "а тут" :=pwsVar1^ "а тут";
8 wsVar2 "новые" :=pwsVar2^ "новые";
9 wsVar3 "тексты" :=pwsVar3^ "тексты";
10
11 IF xReload TRUE THEN
12     DynamicTextReloadTexts();
13 END_IF RETURN
```

Рис. 15. Пример работы с функцией **DynamicTextGetDefaultText**

Как можно заметить – тексты изменились.

Можно забирать списки текстов с **USB-накопителя** или **SD-карты** (с помощью библиотеки **CAA File** или **СmpSysExec**), перезаписывать ими текущие, а потом с помощью данной функции обновлять списки текстов в проекте (они также автоматически обновятся и в визуализации).