

**1C:Enterprise 8
Localization and
internationalization issues**



Table of contents

Introduction.....	3
Localization of the platform.....	4
Localization support by functionality of platform.....	4
International dates and numbers support for most European languages	4
Two variants of the embedded language.....	5
Internationalization support by the embedded language.....	5
NStr function.....	5
PeriodPresentation function	6
NumberInWords function.....	6
Format function.....	7
Getting a template in a language specified.....	8
The platform interface	8
Localization of the platform into other languages.....	9
Localization of the platform interface	10
Localization of the installation wizard	10
The platform localization process management.....	10
Localization of business applications.....	12
Interface of multilingual applied solutions.....	12
Languages.....	12
Localization of interface texts	12
Creation of a business application localized for a certain country	14



Introduction

This document describes all matters concerning localization of 1C:Enterprise 8 product (including the platform and business applications).

The main goal of localization process is to provide a business application capable of working in the local conditions of a particular country, including language, economical and business aspects.

Localization of 1C:Enterprise 8 platform includes: translation of all texts in menus, messages, dialogs, forms, reports, and the help system; also, modification of business logic for meeting local conditions. Each of the items is complex and may include several iterations to get them localized properly.

Any business application that a user deals with includes GUI text coming from both the applied solution and from the platform. For example, the main menu line includes items provided by the platform (such as 'File', 'Edit', etc.) and those provided by the business application ('Documents', 'Accounts', 'Customers' or others, as defined in the solution). Besides, this is a business application that implements application business logic, which is also a subject for consideration.

Therefore, in order to get the software localized two big parts of it should be considered: the platform side and applied solution side.



Localization of the platform

Localization support by functionality of platform

The localization support of 1C:Enterprise 8 platform allows using various interface languages when developing an applied solution, also at user's work. Besides, there are means for international data/time/numbers representation and written form for numbers and periods (spelling).

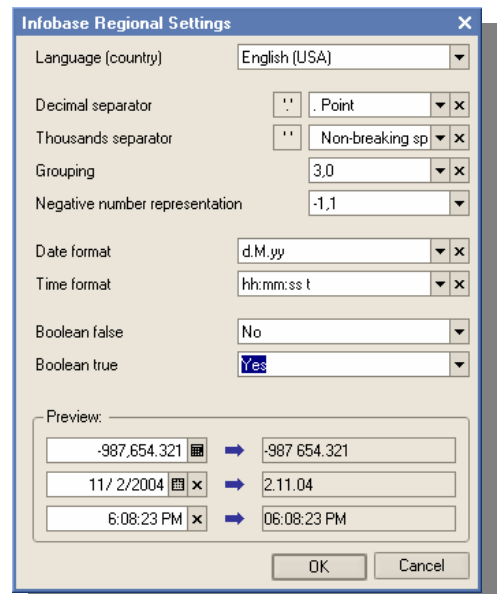
All applied solution configuration texts are stored in a database in **UNICODE** format. This format allows using characters from different languages at a time. So a user can enter data in various languages, for example if a product item description should include some text in the language of the manufacturer. At that the system allows performing filtration and sorting across such texts.

International dates and numbers support for most European languages

There is a capability to adjust a particular applied solution to use locales specific for a particular country. At that, regional settings used in a database may be set by an administrator, other than the default system settings (see the picture).

If none of the properties is set, the locales are defined by the operating system settings.

'Language (country)' field defines a set of predefined locales for a certain country. Selection of a country provides proper formatting of dates, time, numbers and strings collator functionality for the country. At present, the following locales are supported:



Afrikaans (South Africa)	Spanish (Colombia)	Croatian (Croatia)
Belorussian (Belorussia)	Spanish (Costa Rica)	Hungarian (Hungary)
Bulgarian (Bulgaria)	Spanish (Dominican Rep.)	Indonesian (Indonesia)
Catalan (Spain)	Spanish (Ecuador)	Icelandic (Island)
Czech (Czech Republic)	Spanish (Spain)	Italian (Switzerland)
Danish (Denmark)	Spanish (Spain, traditional sort)	Italian (Italy)
German (Austria)	Spanish (Guatemala)	Kazakh (Kazakhstan)
German (Switzerland)	Spanish (Honduras)	Lithuanian (Lithuania)
German (Germany)	Spanish (Mexico)	Lettish (Latvia)
German (Liechtenstein)	Spanish (Nicaragua)	Dutch (Belgium)
German (Luxemburg)	Spanish (Panama)	Dutch (Netherlands)
Greek (Greece)	Spanish (Peru)	Norwegian (Bokmal)(Norway)
English (Australia)	Spanish (Puerto Rico)	Norwegian (Nynorsk)(Norway)
English (Belize)	Spanish (Paraguay)	Polish (Poland)
English (Canada)	Spanish (El Salvador)	Portuguese (Brazil)
English (Great Britain)	Spanish (Uruguay)	Portuguese (Portugal)
English (Ireland)	Spanish (Venezuela)	Roumanian (Rumania)
English (Jamaica)	Estonian (Estonia)	Russian (Russia)
English (New Zeland)	Basque (Spain)	Slovak (Slovakia)
English (Philippines)		Slovenian (Slovenia)

English (Trinidad and Tobago)	Finnish (Finland)	Albanian (Albania)
English (USA)	Faroese (Faroe Islands)	Serbian (Cyrillic alphabet)
English (South Africa)	French (Belgium)	(Serbia)
English (Zimbabwe)	French (Canada)	Serbian (Roman alphabet)
Spanish (Argentina)	French (Switzerland)	(Serbia)
Spanish (Bolivia)	French (France)	Swedish (Finland)
Spanish (Chile)	French (Luxemburg)	Swedish (Sweden)
	French (Monaco)	Turkish (Turkey)
		Ukrainian (Ukraine)

Bottom pane of the window shows representation of formats according to a locale selected.

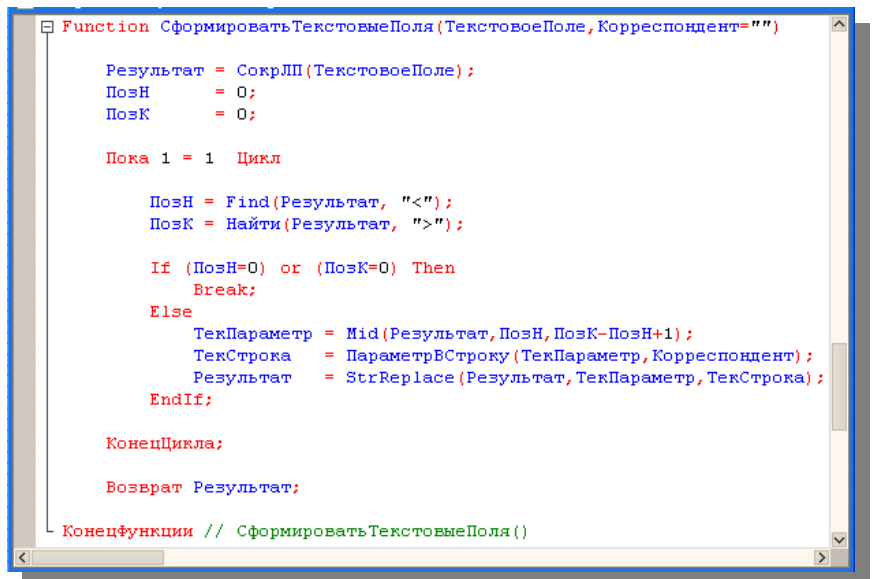
If no specific formats for date/time/numbers are specified, default settings for a defined country are used.

Two variants of the embedded language

All embedded language operators have Russian and English variants, both are accepted by the system equally without any special settings.

No additional modifications during localization process is required in the embedded language.

Syntax Assistant contains English syntax and synonyms for all constructions of the embedded language.



```

Function СформироватьТекстовыеПоля(ТекстовоеПоле, Корреспондент="")
    Результат = СокрЛП(ТекстовоеПоле);
    ПозН = 0;
    ПозК = 0;

    Пока 1 = 1 Цикл

        ПозН = Find(Результат, "<");
        ПозК = Найти(Результат, ">");

        If (ПозН=0) or (ПозК=0) Then
            Break;
        Else
            ТекПараметр = Mid(Результат, ПозН, ПозК-ПозН+1);
            ТекСтрока = ПараметрВСтроку(ТекПараметр, Корреспондент);
            Результат = StrReplace(Результат, ТекПараметр, ТекСтрока);
        EndIf;

    КонечЦикла;

    Возврат Результат;

КонечФункции // СформироватьТекстовыеПоля()

```

Internationalization support by the embedded language

The embedded language itself provides a number of means for applied solutions to be multi-lingual.

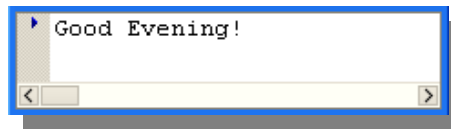
There are functions in the language that return text strings according with the language settings specified. So an applied solution can generate texts in languages according to user's preferences.

NStr function

The function returns a string in the user's language or the main language of the applied solution, taking it from a predefined set of strings in different languages. For example, the string

```
Message(NStr("ru = ""Добрый вечер!""; en = ""Good Evening!""", "en"));
```

results in the following:

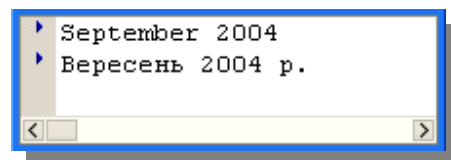


PeriodPresentation function

This function allows building a period description, defined as start date and end date in one of the supported languages. Execution of the code:

```
CurrentDate = CurrentDate();  
BegOfPeriod = BegOfMonth(CurrentDate);  
EndOfPeriod = EndOfMonth(CurrentDate);  
Message(PeriodPresentation(BegOfPeriod, EndOfPeriod, "L = en_US"));  
Message(PeriodPresentation(BegOfPeriod, EndOfPeriod, "L = uk_UA"));
```

results in the following:



At present, this functionality is provided for the following languages:

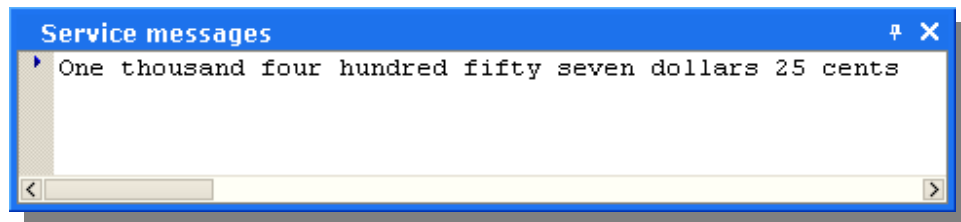
- English
- Estonian
- Byelorussian
- Bulgarian
- Finnish
- German
- Georgian
- Kazakh
- Latvian
- Lithuanian
- Romanian
- Russian
- Ukrainian
- Vietnamese

NumberInWords function

This function generates a written representation of a number in one of the supported languages, according to a formatting string. Here is an example:

```
Message(NumberInWords(1457.25, "L=en_US", "dollar, dollars, cent, cents, 2"));
```

Result of this function looks like the following:



At present, the number spelling functionality is provided for the following languages:

- English
- Estonian
- Byelorussian
- Bulgarian
- Finnish
- German
- Georgian
- Kazakh
- Latvian
- Lithuanian
- Romanian
- Russian
- Ukrainian
- Vietnamese

Format function

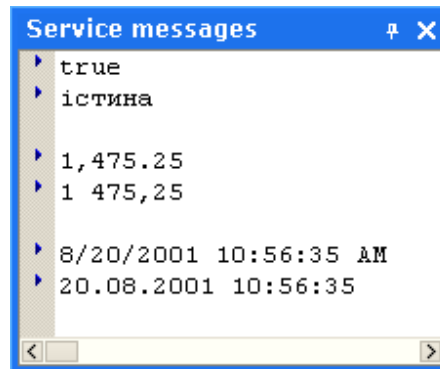
This function formats values representation for convenient reading. The representation may reflect standard format of a selected language or country. For example, boolean, numeric and date type values:

```
Message(Format(True, "L=en_US"));
Message(Format(True, "L=uk-UA) + Chars.CR);

Message(Format(1475.25, "L=en_US"));
Message(Format(1475.25, "L=uk-UA) + Chars.CR);

Message(Format('20010520105635', "L=en_US"));
Message(Format('20010520105635', "L=uk-UA) + Chars.CR);
```

will result in the following representation in English and Ukrainian:



With this function one can use a values representation format from one of 80+ locales.

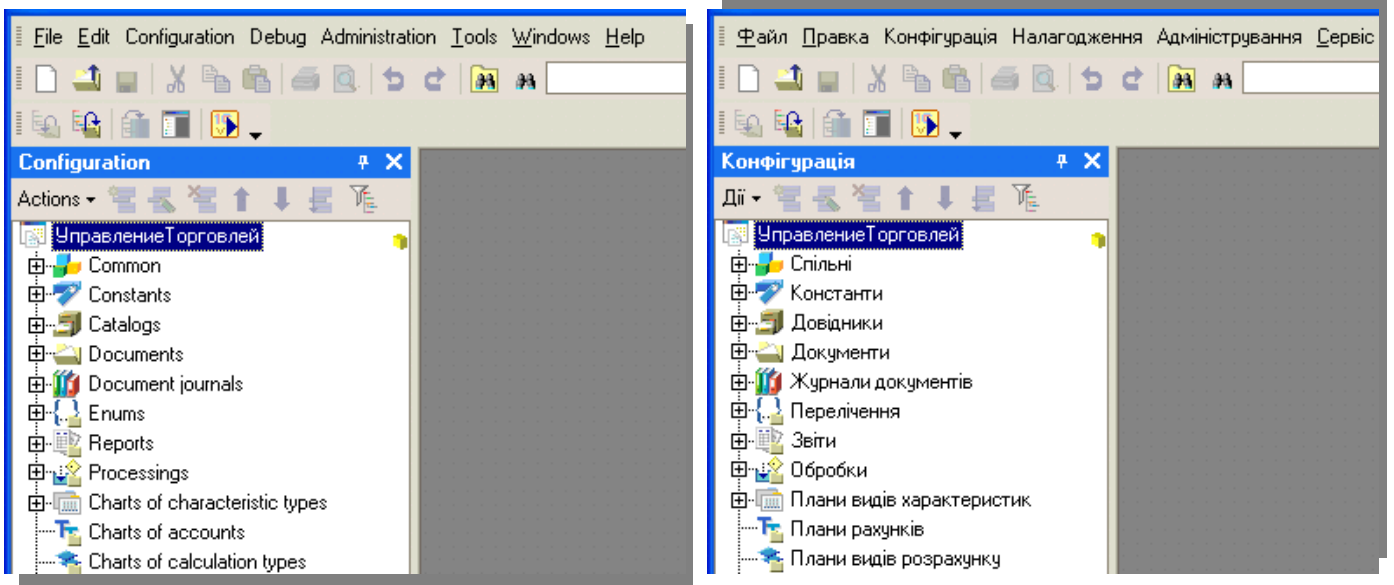
Getting a template in a language specified

When a developer creates output report forms from templates of text and spreadsheet documents, he can specify a language for the template; the language may differ from the one used by current business application user.

The platform interface

When developing an applied solution one can choose GUI language. At present, the platform interface includes the following versions:

- English
- Bulgarian
- German
- Georgian
- Kazakh
- Latvian
- Lithuanian
- Romanian
- Russian
- Ukrainian
- Vietnamese



Localization of the platform into other languages

Process of localization of the platform into other languages includes the following parts:

- Localization of date/time/number formats according to formats of the target locales. They must be formatted according with the standards accepted in the particular country (region).
- Localization of string collator functionality. Strings in reports and forms must be sorted properly.
- Localization of spelling totals and periods functionality.

These three localization parts are always done by 1C developers on the basis of formalized information on wording algorithms for spelling numbers, provided by the company partner. The partner can only check if they work properly setting a required [Infobase Regional Settings](#), after a test release of localized platform.

However, before going on with the localization process, localizer must make sure there is a proper locale code in Database Regional Settings window. Presence of a proper code means, that existing platform functionality supports formats and sorting order according to those accepted in the target country.

ATTENTION! If there is no proper country in the Country list of the window, one has to try to find another country with proper format settings, then – to check sorting order for that country. Results of these checks, together with detailed explanations of the check procedures done, should be immediately sent to 1C before going on further with localization process!



Localization of the platform interface

The platform interface must be readable in standard user conditions without special (3rd party's) software support.

This work is usually done by 1C Company's partners and implies mostly translation work.

For this work to be done there is a resource database (in 1C:Enterprise 8 format) that is used by our partners.

They unload resources from the database into external files and then use special localization tools (also provided) to localize them.

For localization purpose, the database and the localization tools can be sent to our partners by e-mail. It's required the partner to have 1C:Enterprise 8 platform, to open the resource database.

Localizable resources include:

- Pictures. Can be edited with any graphic editor.
- Templates – are localized with 'hlpedit' utility.
- Help - if localized with 'hlpedit' utility, too. When translating, it's suggested to go through the tree in 'Files' tab, translating each file included into a particular .hbk file. Going through "Contents" cannot guarantee all help files will be translated.
- Strings – translated directly in the database itself, without saving to external files. Can be accessed through 'Operations - Catalog – Strings' menu items.
- Forms – are localized with 'FormDes' utility.

Help resources can also be exported into .xml format, edited with 3rd party software and then imported back into resource files.

After translation of resources, they are uploaded back into the database and in its turn it's sent back to 1C.

If these points are completed and checked for proper functioning, the platform is considered localized.

Localization of the installation wizard

The installer generates several messages during installation process. To be represented in a proper language, these messages should be localized as well. Since there may be several localization processes going at a time, for different locales, these strings are sent separately by 1C, as the localization process goes.

The platform localization process management

1C Company provides all relevant materials and the platform localization technology to its partners (contractor). The partner performs the localization and sends a resulted database back to 1C. 1C builds in the localized resources with the main release and produces a localized version of the platform.



One should also take into account, that localization, like development, is an on-going process. Help and forms are updated as the development moves on, and localizer is responsible for keeping localized version up-to-date, even after a format software release. All contract terms with localizer are detailed in a separate Localization Agreement, signed between a localizing entity (a company) and 1C.

Localization of business applications

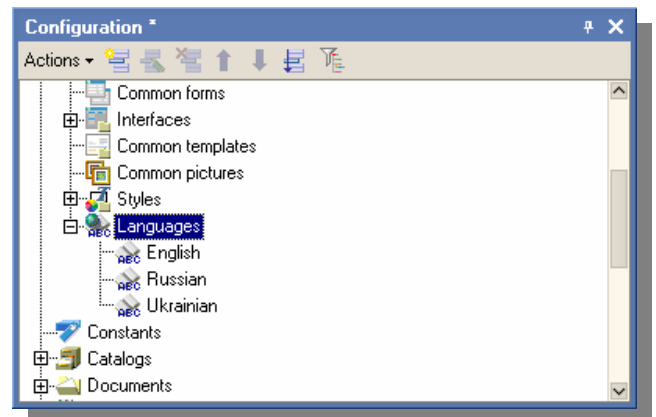
Business applications (applied solutions) have their part of interface and business logic in the software product.

Interface of multilingual applied solutions

Using a single localized platform, one can create multilingual applied solutions due to the possibility to use additional localized interfaces. After installation of the following localized interface items are available:

- system menu,
- dialogs,
- messages,
- help system.

Selection of one or other localized interface implies no restrictions to operations on an information base, therefore several users with different interfaces can operate with one the same database at a time. It's also possible to run several instances of 1C:Enterprise with different interfaces on one PC.



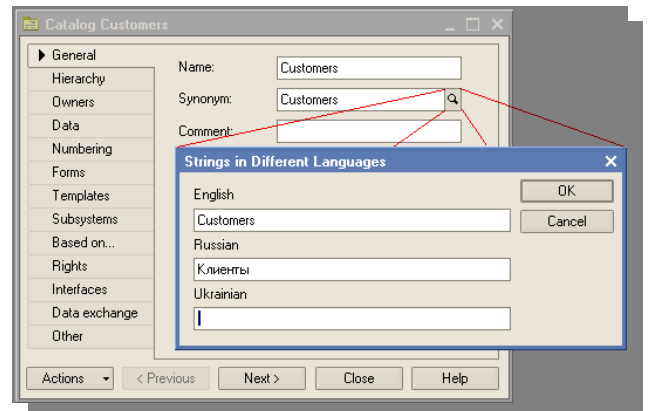
Additional applied solution interface languages

Languages

An applied solution can include several different interface languages.

Since all strings are stored in a database in UNICODE format, a developer is allowed to enter several variants of spelling in different languages for one string.

Besides, if a multi-lingual applied solution is developed, one can quickly switch from one language into other. This is very useful when developing forms: one can preview a form in English first, and then – in Russian.



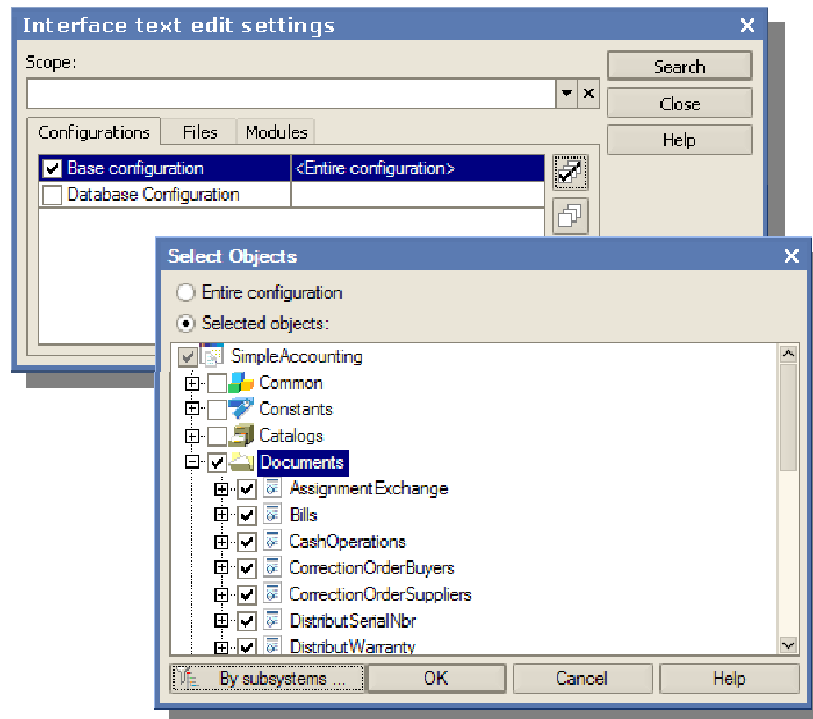
Localization of interface texts

Localization of interface texts is one of the most difficult tasks and it implies looking for places to enter texts in other languages. Therefore, localization of applied solutions' interface implies forming string values in required languages, for them to be displayed in user interface.



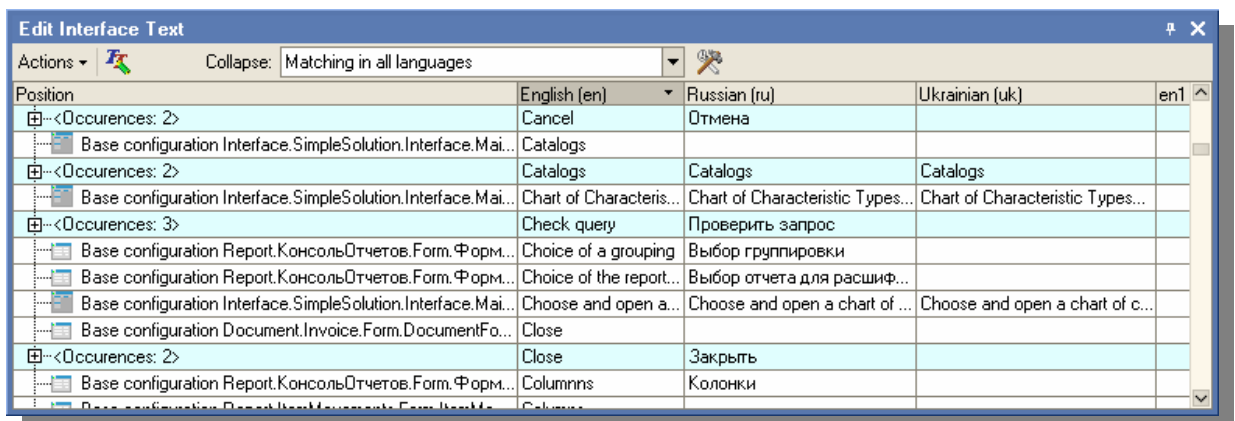
There is a mean for rapid applied solution interface localization in 1C:Enterprise platform. It allows locating, grouping and translating all occurrences of a string in an applied solution configuration, replacing it or just adding a translation into the configuration “in one click”.

First, one specifies an applied solutions to look at:



This search can be performed for an applied solution entirely, across external files, opened documents or through particular applied solution’s parts.

Result of this search is represented as a list items, containing locations of the text found and its meaning in different languages.



By mouse click one can jump to the applied solution item containing the string, to edit it; the string values can be also edited right in this list, too.

As a result of defining required values for all strings in each language, the applied solution is localized for using with interface in these languages. If any string is not localized, it shows nothing (empty string) at corresponding text fields.



Creation of a business application localized for a certain country

In order for a business application to be applicable in a certain country business conditions its business logic should be localized for these conditions, too.

Providing 1C:Enterprise business applications for other countries (such as Ukraine or Lithuania), 1C Company's partners there have a good experience in creation of business applications for local markets. At present, there are localized versions of applied solutions for Ukraine, Kazakhstan, Byelorussia and Baltic States, more are expected to come.

Some parts of business logic are moved to localized applied solutions without modifications, as there are no special requirements (like CRM, for example), while other parts require adaptation and there are examples of such adaptations.

'To localize' here doesn't means 'to translate into other language'. In fact, it means 'to adapt for operation in other locale's conditions'.

It's not possible immediately to translate business logic in modules (i.e. lines of the code implementing application business logic).

One of the advantages of the product is its ability to be easily customized, even by end-users. Therefore, we cannot allow incorrect lines of the code (or comments) to be shown in the designer window. Furthermore, there were attempts to translate business logic of some business applications. Results were far from being good.

In other words, adaptation includes not only translation of the interface, but also re-working of the business logic. It's closer to development process, than to 'translation'.

Some parts of an applied solution's business logic can be localized easier, while some parts are more difficult than others.

1C usually provides a real business application as an example, on development 'stage', for its partners. Our experience shows that adaptation of a business application takes 50% of the time spent for initial development of it (i.e. from scratch). For example, for an applied solution like Trade Management it takes approximately 6 men/months work (usually 2 – 3 developers work for 3 - 2 months).

From point of view of adaptation of a business application for a certain economic conditions, generally, modules responsible for the application's business logic are usually re-written from scratch, since economical conditions and standards are different in different countries.

In case with 1C:Enterprise business applications, most business logic modules need only some modifications, due to special means provided by the platform and 1C:Enterprise Designer. For example, there is Payroll Calculation module that can be adjusted by standard visual tools for almost any economic requirements. So, a developer does not need to re-write the whole module, he just needs to 'adjust' it. In cases like this, the core of the code implementing the business logic is generated by the platform, according to the interface language selected (i.e. English operators, with relevant comments inserted, etc.).

Boxed business applications from 1C company, including accounting business logic, are developed in close cooperation with PricewaterhouseCoopers and provide methodological basis for arranging accounting according with IAS (International Accounting Standards). So the mentioned above is applicable to many areas of creation of business logic of applied solutions.