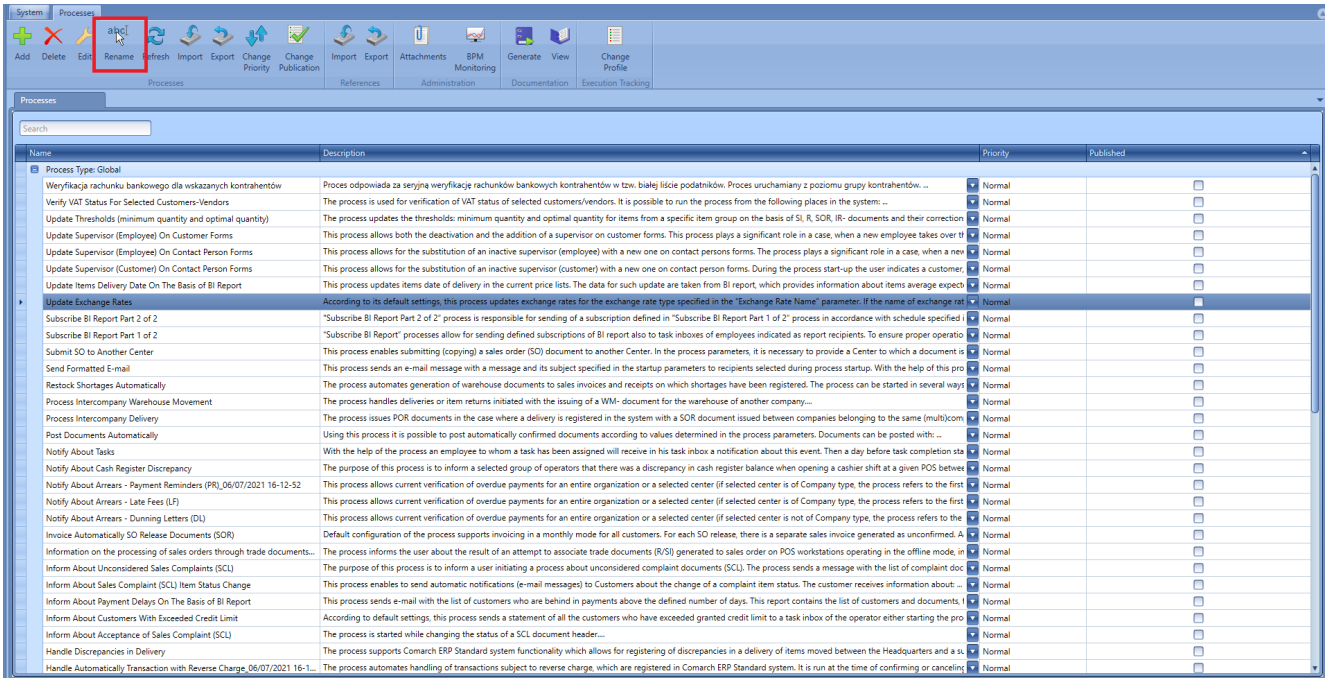
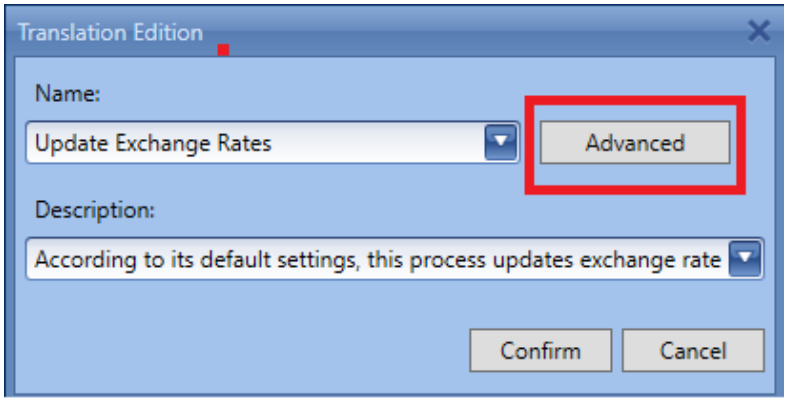
Working in different languages

Translating processes

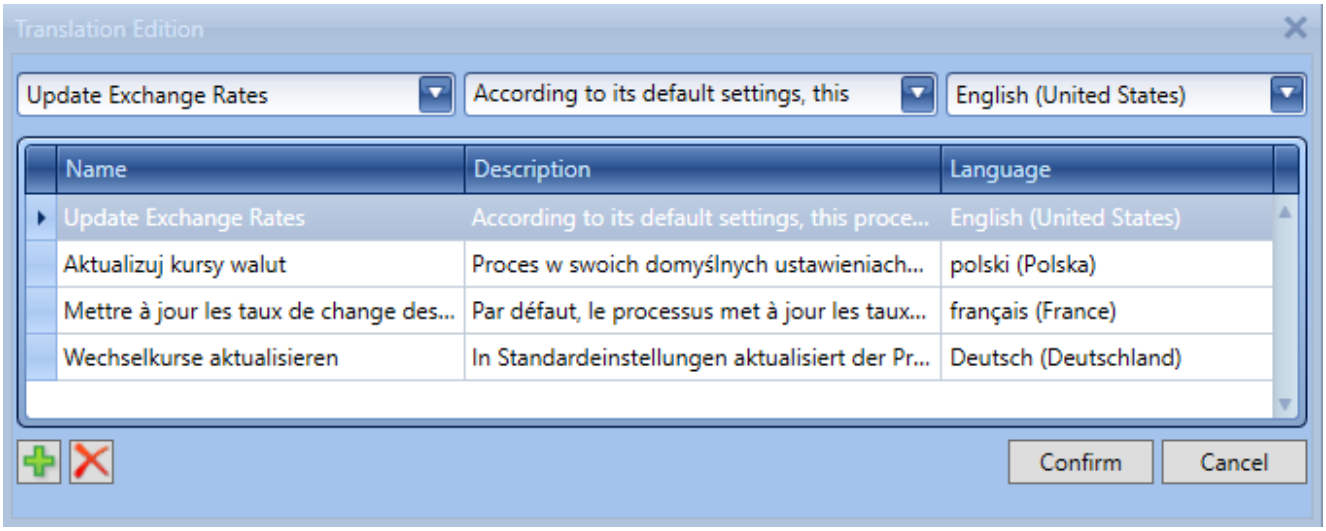
To add/edit the name and description of a process in another language, it is necessary to mark such process and select [**Rename**], and next [**Advanced**] button.



Changing name and translations of a process



Editing process name

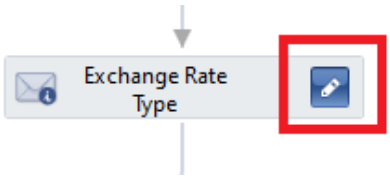


Editing process translations

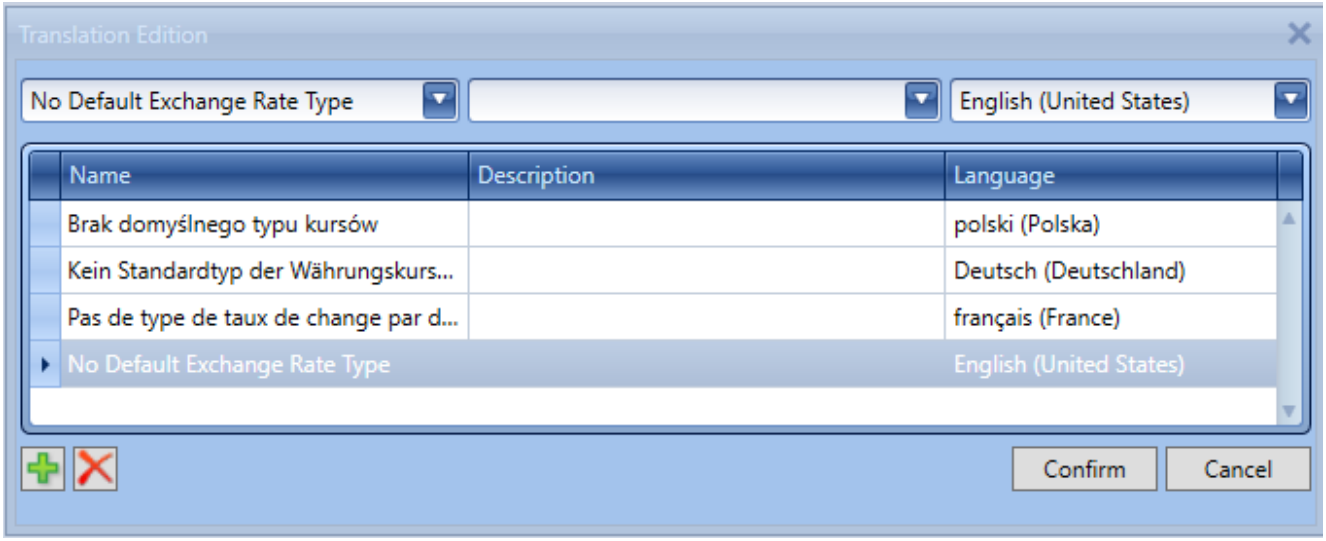
To add translation to a process, it is necessary to select the plus button, choose the language and complete the name field. To edit a translation, it is necessary to mark a given entry and modify relevant fields. After entering the modifications, click on [**Confirm**] button.

Translating activities

To add/edit translations for activities in the BPM editor, it is necessary to click on the pencil icon placed on the element. These translations are defined in the same way as process translations.



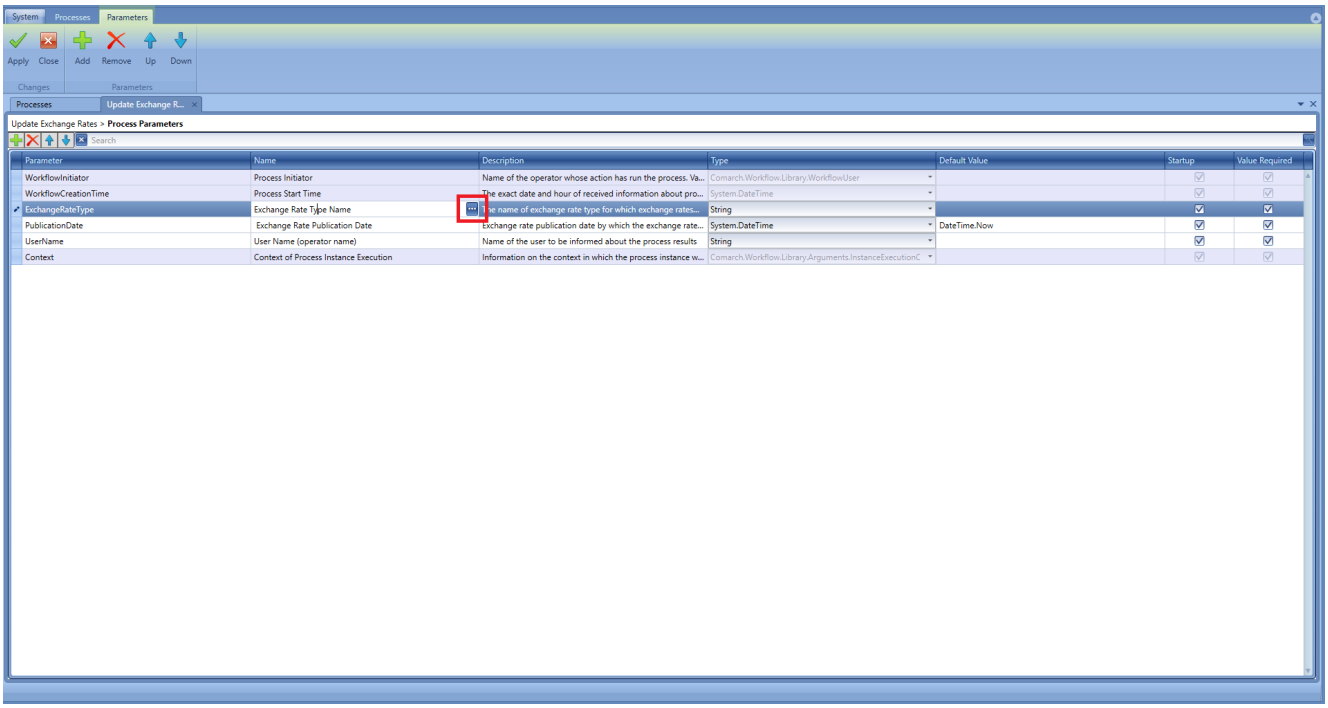
Translating activities



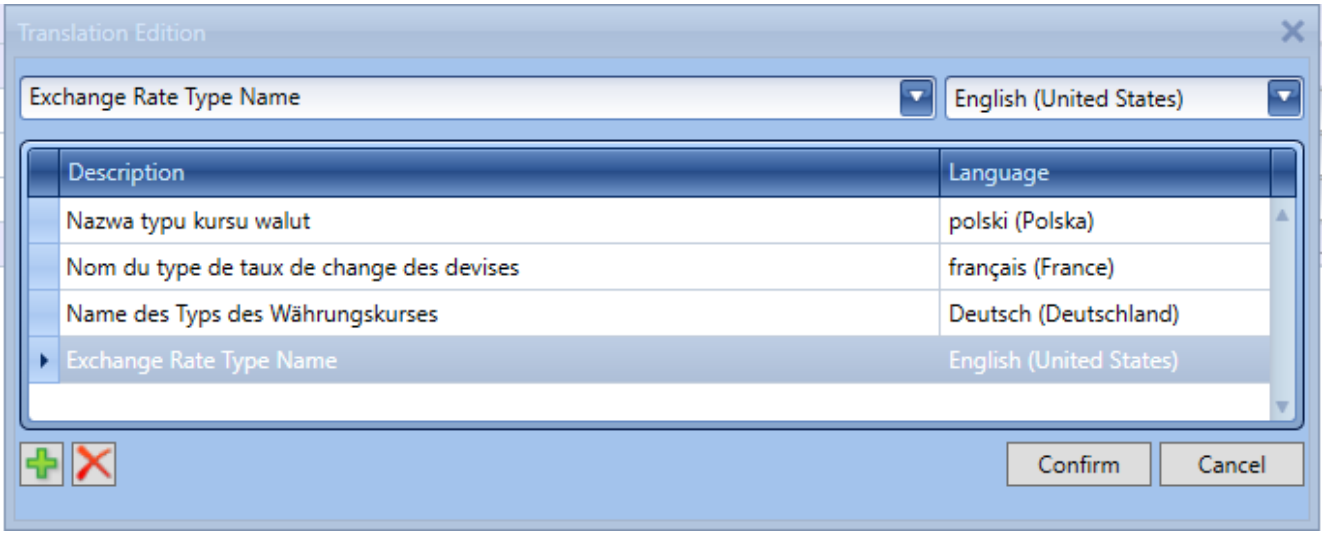
Editing activities translations

Translating parameters

To add/edit translations of <<parameters>>, it is necessary to select […] button during the edition of the filed to which the translation is to be added. It can be done both for the name and description.



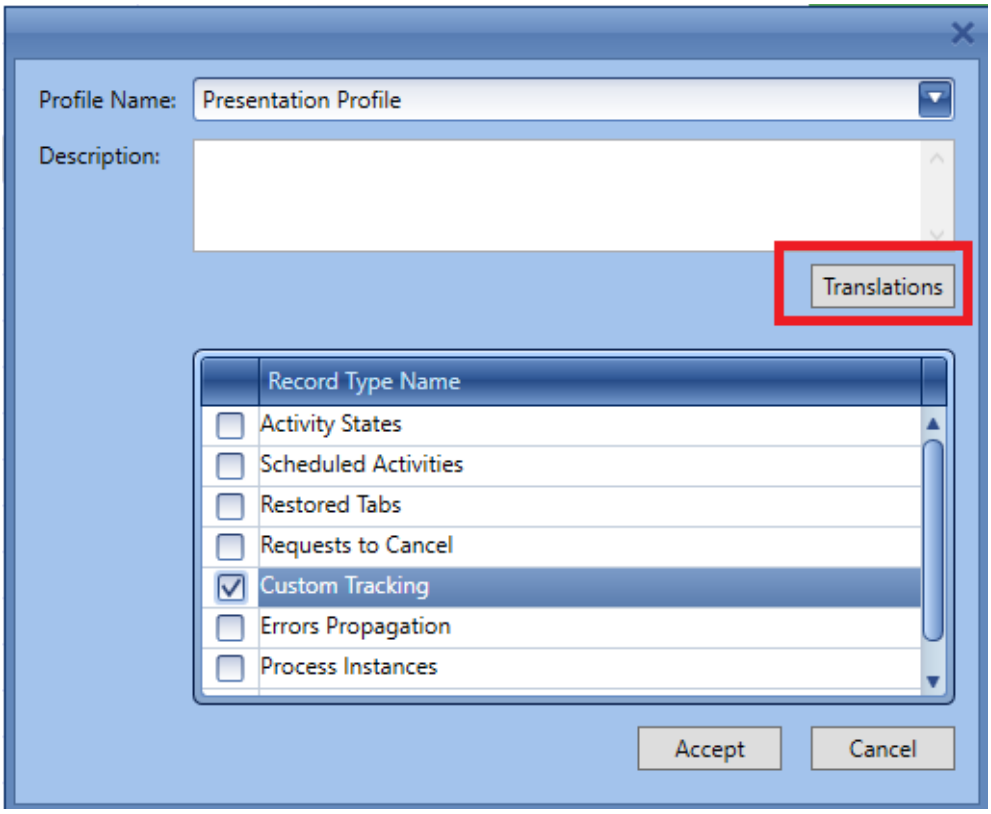
Translating parameter



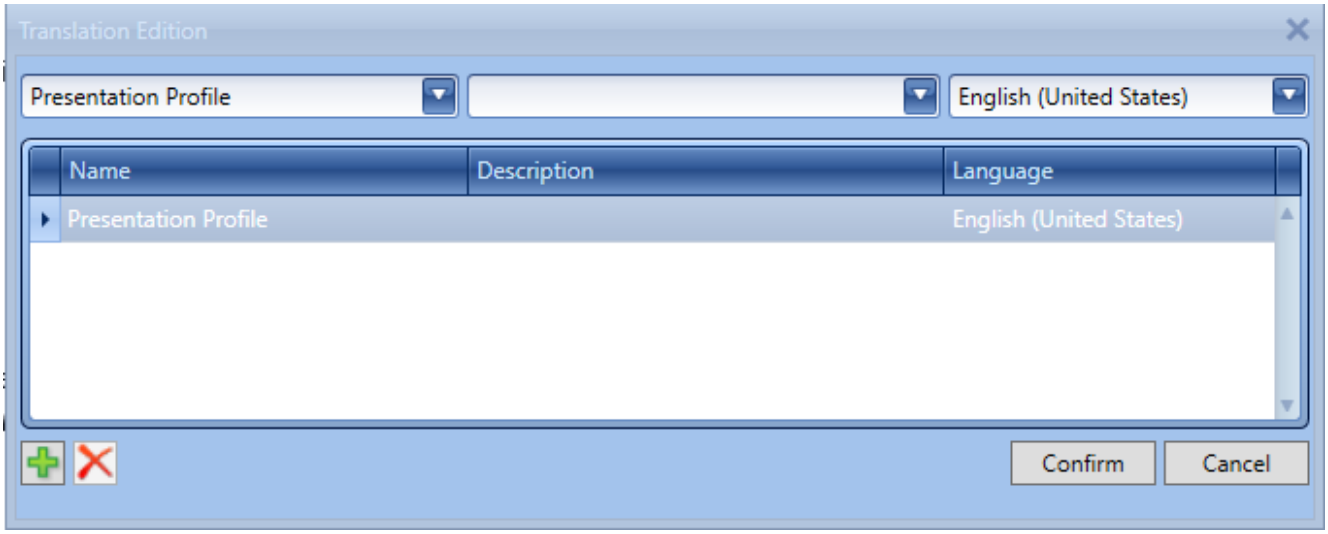
Editing parameter translations

Translating tracking profiles

To add/edit translations of <<tracking profiles>>, it is necessary to select [**Translations**] button when editing or adding new profile.



Translating tracking profile



Editing tracking profile translations

Good practices

Below, there is a list of good practices, that lis suggesions, thanks to which the work with processes can turn out easier and processes more effective.

Readibility

- Using flowchart as a main avctivity
- It is neccessary to group activities with the use of the activity *Sequence/Flowchart* activities with the same scope (e.g. adding a SI item or setting a customer/vendor) and name and localize them in a proper way.

Multiculturalism

- Names of variables and input/output arguments should

have English names or be abbreviations of English names
- Default names shoud be changed and <<translations>> should be added to them.
- Descriptions of parameters should contain information about the use of a given parameter.
- A process should have name and added desription.
- The names of exported .wdf files should be in English.

Handling of errors

- It is necessary to complete fields left empty, e.g. e-mail address, to prevent eocurring of exception *NullReferenceException*
- It is necessary to use *TryCatch* activity, if it’s possible, in order to catch busness exceptions. Information about errors can be sent to the task inbox

Configuration possibilities

- It is necessary to perform parametrization of process operation, e.g. by adding the possibility of selecting a center with permissions of which the process is to be performed.
- Oermaters of SQL queries should be associated with variables or arguments to facilitate modifciation of process working.

Performance

- When preparing processes, it is necessary to remember that they must work on production databases. A test database used for verification of processes should correspond to the production database in termos of the quantity of data, so that already in the production phase it is possible to identify performance problems.
- A user should pay attention to the method of determining

scopes for variables. In case a given variable is used e.g. only inside a given sequence, its scope should not be set.
- To improve memory performance, it is possible to set unused variables to *null* in the further part of a process. When a process is waiting for decisions, the data regarding it is saved in the datatabse. The less data is in the database, the less cache memory and memory in the database is occupied.

Integration with BI

In the Comarch ERP Standard BPM system, it is possible to use reports created in Comarch Business Intelligence. Before proceeding to the BPM configuration, it is necessary to:

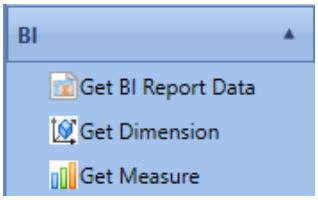
- Install Comarch Business Intelligence module
- Create BI databases for the company database (DW, LOG, META and REPO) and transform the company database.

BPM configuration in terms of BI

To add business activities associated with BI, in the references of a given process, it is necessary to select two assemblies:

- BI.Integration.dll
- BI.Integration.Interfaces.dll

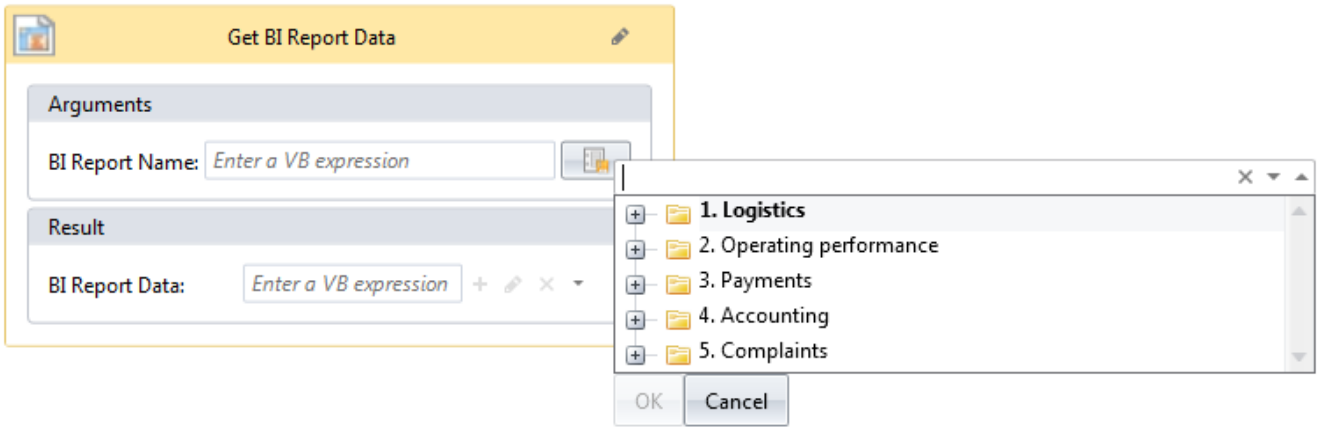
Detailed description of the process of importing of assemblies can be found in article *References*.



BI activities

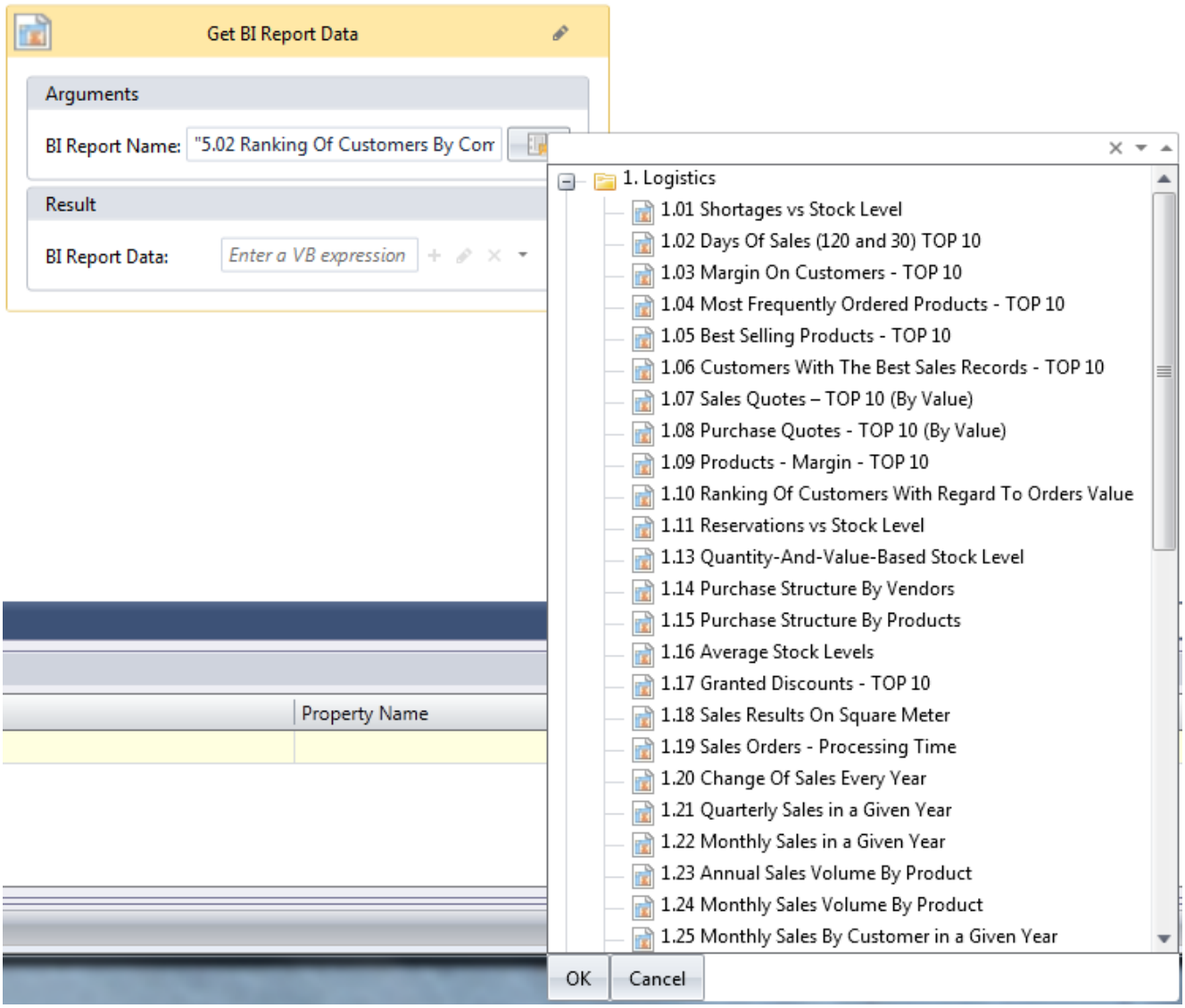
Retrieving data from a report

Retrieving data on the basis of a report is performed by selecting an appropriate report in the definition of the activity *Get BI Report Data.* Additionally, when indicating a report, a list of BI reports is retrieved from BI. Below, there is an example of opening a list of OLAP reports grouped by an analytical cube.



Selecting a BI report from the level of activity

After expanding each group, a list of BI reports available in a given area, is presented. After placing the mouse cursor over a selected report, the user can preview more details regarding it. A report description contains the following data: *Created On, Cube Name, Dimensions, Measures* and *Filters.*



BI report details from the level of an activity

Standard processes using BI

In the system, there are three standard processes which are based on BI reports:

- Inform about Payment Delays On The Basis of BI Report
- Update Items Delivery Date On The Basis of BI Report
- Generate a Series of Questionnaires for Top Vendors on the Basis of BI Report

To use the above-mentioned process, it is not necessary to add new the assemblies, as in case of creating individual processes. After importing the processes, the assemblies will be added automatically.

Apart from the above processes, there are two associated with each other processes which send BI subscription

- Subscribe BI Report Part 1 of 2
- Subscribe BI Report Part 2 of 2

To ensure proper operation of sending reports from to the task inbox, the following conditions must be fulfilled:

- Both BI processes must be imported and published
- BPM environment must be properly configured (*BPM configuration*)
- When creating a BI report subscription, it is necessary to set an appropriate subscription type.

Starting BPM processes from the level of BI

It is possible to run BPM processes from the level of the list of items and customers/vendors in BI reports. To do so, it is necessary to add the following contexts in the process:

- Items in BI Reports
- Customers/Vendors in BI Reports

More information regarding contexts can be found in article *ERP Context*.

Basic elements of Visual Basic .NET syntax

When creating BPM processes, it is possible to use VB language syntax. For instance, initial values can be assigned to parameters in such way or operations on numbers or texts inside processes can be performed.

Operators

Visual Basic supports the following types of operators:

Operation Name	VB.NET	C#	T-SQL
A less than B	A < B	A < B	A < B
A less than or equal to B	A <= B	A <= B	A <= B
A equal to B	A = B	A == B	A = B
A greater than B	A > B	A > B	A > B
A greater or equal to B	A >= B	A >= B	A >= B
A different from B	A <> B	A != B	A <> B
A or B	A OrElse B	A | | B	A OR B
A and B	A AndAlso B	A & & B	A AND B
Assigning decimal value 10,1 to A	A = 10.1D	A = 10.1m	A = 10.1
Assigning floating-point value 10,1 to A	A = 10.1R	A = 10.1	A = 10.1
Negation of A	Not A	!A	-
Combining strings	”a” & b.ToString()	”a” + b.ToString()	‘a’ + b
The first element of T table	T(0)	T [0]	-
Creating an object of nullable integer type	New Nullable(Of Int32)	New Nullable()	-

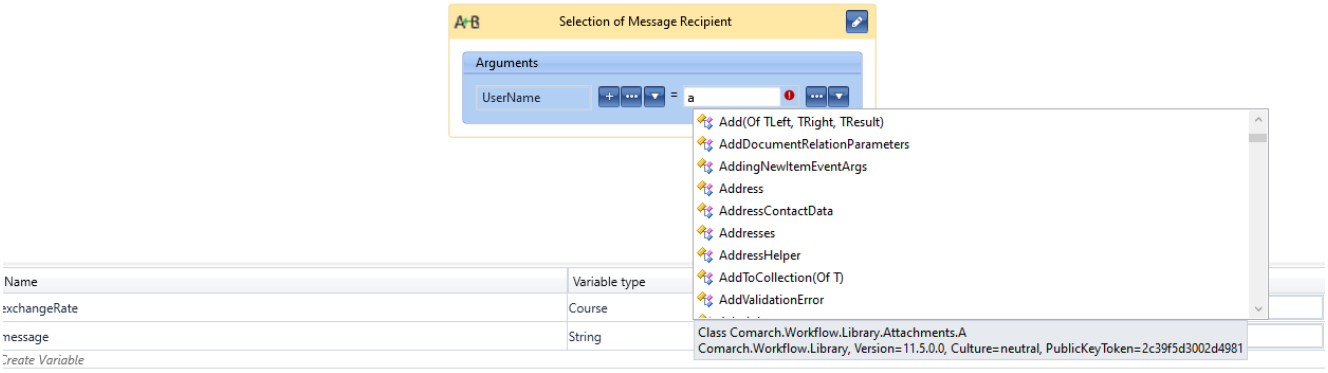
Operation Name	VB.NET	C#	T-SQL
Invoking a method with SalesInvoice generic type	ObjectFactory.Create(Of SalesInvoice)()	ObjectFactory. Create()	-
Converting an object object into type Type	DirectCast(object, Type) CType(object, Type)	(Type)Object	-
Null value	Nothing	null	NULL
A is not a null	A IsNot Nothing	A !=null	A IS NOT NULL
A is a null	A is Nothing	A == M	
A is of *Type* type	TypeOF A Is Type	A Is Type	-
Creating a directory of String types key and Decimal values and initiating it with”KOS” key of value 2.0D	New Dictionary(Of String, Decimal)() From {{"KOS", 2.0D}}	new Dictionary() {{"KOS", 2.0m}}	-
Creating an object of DocumentEventParams type and initiating its property DocumentId with value 169	New DocumentEventParams() With {.DocumentId = 169}	new DocumentEventParams() {DocumentId = 169}	-
Assigning a text value: Text to variable A	A ="Text"	A ="Text"	SET @A = 'Text'
Assigning a text value in inverted commas: “Text” to variable A	A ="""Text"""	A ="\"Text\""	SET @A = '"Text"'

Operation Name	VB.NET	C#	T-SQL
Using a conditional operator to assign a text value to a variable of string type A on the basis of the value B (if B equals to Nothing, A will equal to String.Empty, otherwise, A will equal to B)	A = If(B Is Nothing, String.Empty, B)	A = B == null ? String.Empty : B	-
Creating an anonymous method	IEnumerable.Find(Function(c) c.Code.Equals("PL"))	IEnumerable.Find(c => c.Code == "PL")	

Note
It is necessary to remember about correct types when comparing. For example, expression “73” <“9” returns *True*, because the first character in the first expression is classified higher than in the other one. If first characters are equal, then the next character from both expressions is compared, etc.

Intellisense mechanism

Comarch ERP Standard BPM has its own Intellisense mechanism. Intellisense is a form of automatic completion. At the same time, it is designed for documenting and disambiguating names of variables, functions and methods. Using the mechanism is a convenient way to get access to descriprions of functions, and partially to the list of their parameters.



Intellisense mechanism