

# **FastCube.Net User's Manual**

# Contents

<b>Description of the FastCube Library</b>	<b>3</b>
The Grid	3
Setting up the Grid Structure	3
Totals	5
Drill Up/Down	10
Drill Through	11
Rotation	12
Filtering	14
Sorting	17
Grouping	18
Formatting, highlighting	21
Data Formatting	21
Conditional highlighting	22
Continuous highlighting	24
<b>Expressions</b>	<b>28</b>
Expression editor	28
Using .Net functions	29
Reference to slice data	30
Reference to slice elements	30
Script	31
General information	31
Using .Net objects	31
Data export	32
Export to XLS	33
Export to ODS	34
Export to HTML	34
Export to XML	35
Toolbar	35
The Chart	36
The Chart Toolbar	37
<b>Example of FastCube Use</b>	<b>38</b>
Example "Demography"	38

# Description of the FastCube Library

## The Grid

The Grid is an interactive tool for the presentation and analysis of data. There are multi-level headers in the left and top parts of the grid. The 'headers' are filled with the dimension values; there are both horizontal and vertical 'headers'. The central part of the grid displays the values of the measures.

			Price	Amount	Work price
Grand total			16 179,00	27	3 410,00
Consumption	service center	Air filter	400,00	1	70,00
		Fuel filter	1 700,00	1	700,00
		Reducer oil	550,00	1	320,00
		Total	2 650,00	3	1 090,00
	shop	Antifreeze	50,00	10	
		Motor oil	1 079,00	1	
		Oil filter	150,00	1	0,00
		Transmission oil	1 970,00	1	320,00
		Total	3 249,00	13	320,00
	Total			5 899,00	16
Documents	auto market	contract of purchase	800,00	1	
	GAI	registration certificate	1 000,00	1	
	insurance agent	insurance	2 500,00	1	

The grid consists of several regions, identified by numbers above:

1. the grid title
2. the filter region : dimensions included in this region can be used for filtering the grid data
3. the vertical dimension headers : dimensions included in this region form the grid's vertical header
4. the horizontal dimension headers : dimensions included in this region form the grid's horizontal header
5. the vertical header of the grid
6. the horizontal header of the grid
7. data region
8. the list of fields : the fields required for processing are dragged from the drop-down list : fields not required are dragged back to this list
9. list of Top-N filters : filters can be modified through the context menu for this field
10. the region showing aggregates for the selected cells : can be modified through the context menu

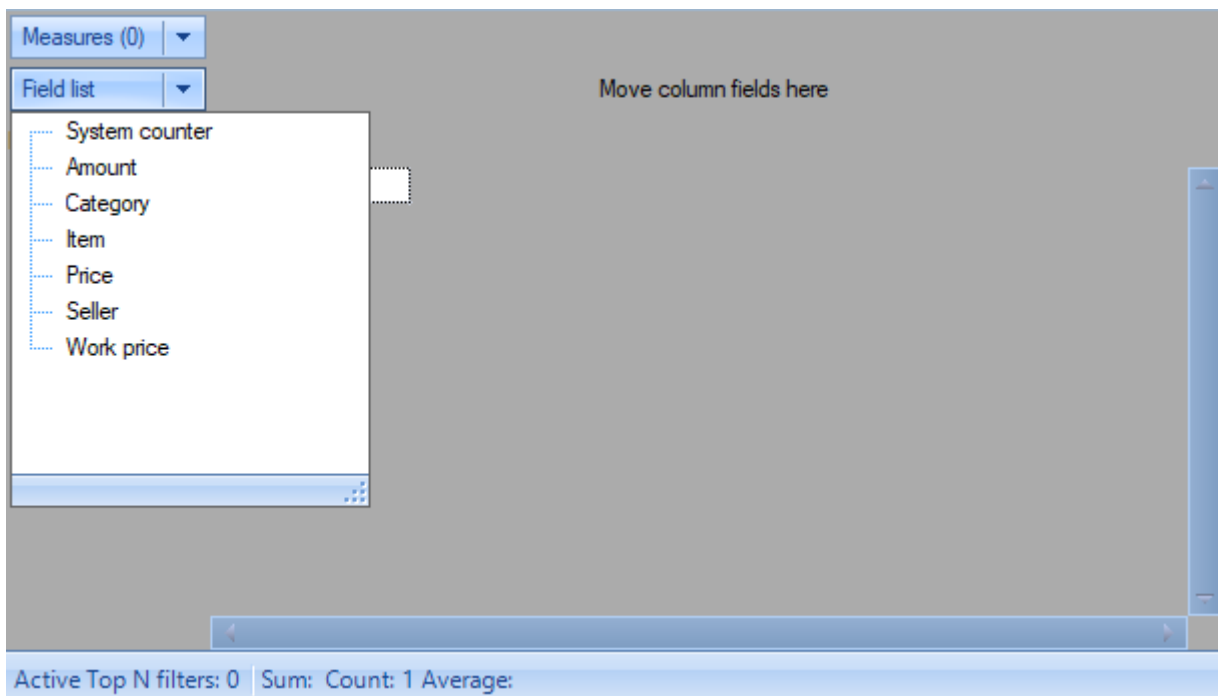
## Setting up the Grid Structure

The grid structure may be defined in several different ways: by code, by loading from a cube or scheme file or by manual setup by the user. This paragraph deals only with manual setup of the grid structure.

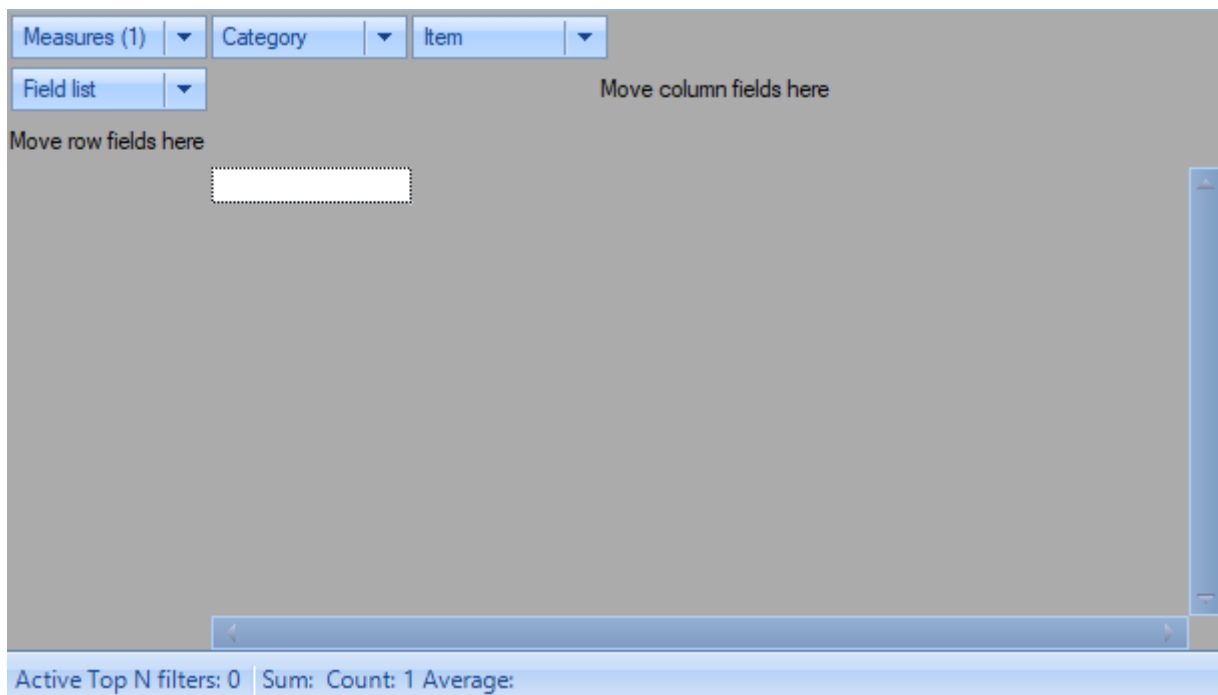
Dimensions and measures are added to the grid by dragging from the drop-down list (shown below) or by using the Field List dialog. They are added to one of the following regions:

- the filters region

- the row fields region
- the column fields region
- or the measure fields region (the data region)



Let's add the fields "Category" and "Item" to the filters region, and drag the "Price" field to the data region. The grid now looks like this:



The number in brackets in the "Measures" dimension indicates the number of measures defined for this grid. Here the number of measures is 1, since we dragged the "Price" field to the data region. The "Measures" dimension is in fact not a simple dimension. Unlike all the others, "Measures" contains measures, and not elements of the database row as its values. This implementation of the measure list as a pseudo-dimensions allows a consistent approach in the user interface - dragging to any dimension region and value filter.

No data is yet displayed in the grid above as the grid only displays data when it contains at

least one measure and the "Measures" dimension is located either in the row fields region or in the column fields region.

For example, if the "Category" and "Item" dimensions are dragged into the row fields region and the "Measures" dimension into the column fields region, then the grid will look like this:

Move filter fields here

Field list		Measures (1)
Category	Item	Price
Grand total		16 179,00
Consumption	Total	5 899,00
	Air filter	400,00
	Antifreeze	50,00
	Fuel filter	1 700,00
	Motor oil	1 079,00
	Oil filter	150,00
	Reducer oil	550,00
	Transmission oil	1 970,00
Documents	Total	4 300,00
	contract of purchase	800,00

Active Top N filters: 0 | Sum: Count: 1 Average:

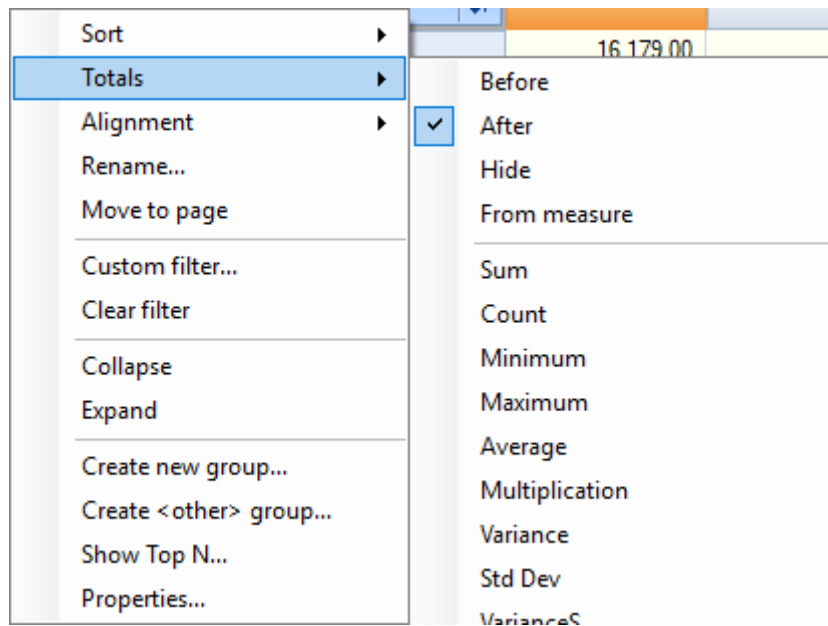
## Totals

As well as dimension values the grid can also show Total values. Total value is an aggregated measure value over a group of dimension values. A Total value is calculated using the default aggregate function for the measure.

The various levels of Total are shown in this partial grid:

Grand total		16 179,00	27	3 410,00
Consumption	service center	Air filter	400,00	70,00
		Fuel filter	1 700,00	700,00
		Reducer oil	550,00	320,00
		Total	2 650,00	1 090,00
		Antifreeze	50,00	
		Motor oil	1 079,00	
		Oil filter	150,00	0,00
		Transmission oil	1 970,00	320,00
		Total	3 249,00	320,00
	Total		5 899,00	1 410,00

Use the dimension context menu to change the position of Totals (before or after), to hide Totals and to manage the aggregate functions.

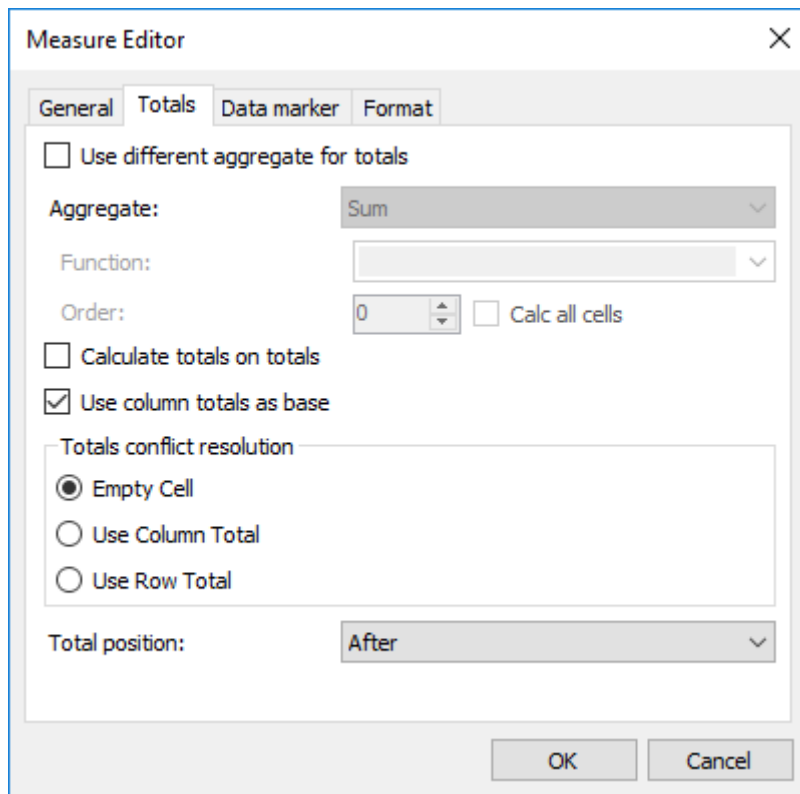


If in the example above we turn off the Total for the "Seller" dimension and also the Grand Total, set the Total position for "Category" dimension to "Before" and add an aggregate "Count" to it, we achieve this final grid:

Category	Seller	Item	Price	Amount	Work price
Consumption	Total		5 899,00	16	1 410,00
	Total Count		7,00	7	7,00
	service center	Air filter	400,00	1	70,00
		Fuel filter	1 700,00	1	700,00
		Reducer oil	550,00	1	320,00
	shop	Antifreeze	50,00	10	
		Motor oil	1 079,00	1	
		Oil filter	150,00	1	0,00
		Transmission oil	1 970,00	1	320,00

The context menu item "From measure" sets Totals to show a value defined on the "Total position" parameter for the measure. This item only has a value for the axis which contains the "Measures" field and is ignored for other axes.

A measure's Total parameters can be modified in the measure editor:



A Total value is calculated using the default aggregate function for the measure. The aggregate function to be used can be changed or can even be implemented using a script, if required.

The Total calculation can be made either on the source dataset values or on total values of a child level. This can be set using the "Calc totals on to totals" checkbox. If this option is checked then the axis from which to take the Total value (either columns or rows) must be set. This can be set using "Use column totals as base".

Sometimes a situation can arise where, according to the dimensions parameters, each axis requires a different aggregate function with a different values for particular cells. So there is a conflict of which aggregate function to use. The "Totals conflict resolution" group-box has options for solving the conflict. A conflicting cell will not be calculated by default.

The "Total position" parameter defines how to show the Total for the axis with the "Measures" field for those dimensions which have the Totals position set to "From measure".

The example below shows the dimension "Manager" Totals set to "From measure", the "Amount" measure Totals set to "Before", the "Cost" measure Totals set to "Hide" and the other measures Totals set to "After". It can be seen that the "Cost" Total is not shown for the "Manager" dimension:

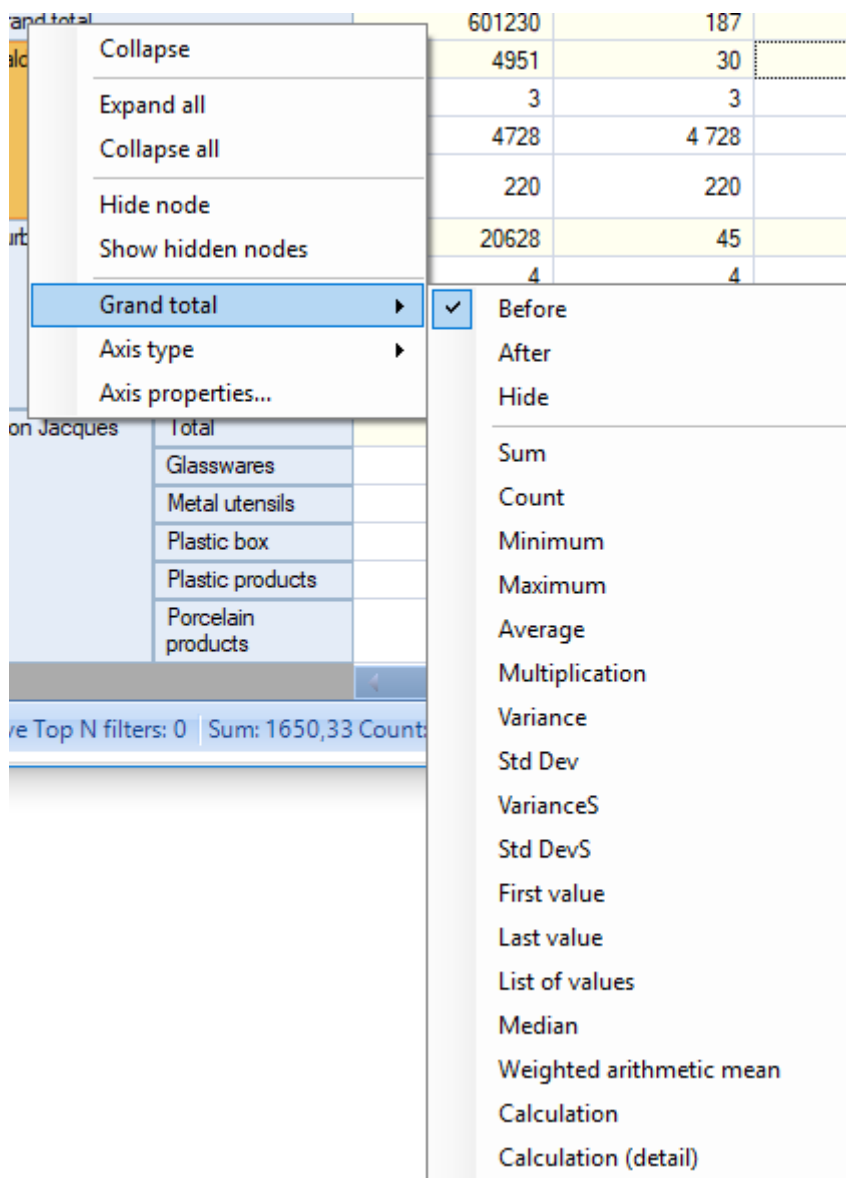
Customer		Field list	
Manager	Product	Measures (4)	
Amount		601230	
Cost		11 706 187,56	
Paid		5 219 178,52	
Balance		6 487 009,04	
Baldwin Janet	Amount	4951	
	Metal utensils	Amount	3
		Cost	643,00
		Paid	643,00
		Balance	0,00
	Plastic products	Amount	4728
		Cost	240 260,35
		Paid	6 453,60
		Balance	233 806,75
	Porcelain products	Amount	220
		Cost	6 416,30
		Paid	1 257,60
		Balance	5 158,70
	Paid	8 354,20	
	Balance	238 965,45	
Burbank Kevin	Amount	20628	
	Glasswares	Amount	4
		Cost	483,25

The next example shows the outcome when Total aggregate functions are changed and Total values are calculated on child totals. All three measures are built using the main aggregate function "Sum" by field "Amount". The Total for measure "Amount" is not customized. For measures "Amount1" and "Amount2" the Totals aggregate function is changed to "Average" and the measure "Amount2" uses the "Calculate totals on totals" option:



Field list		Measures (3)		
Manager	Product	Amount	Amount 1	Amount 2
Grand total		601230	187	54 657
Baldwin Janet	Total	4951	30	1 650
	Metal utensils	3	3	3
	Plastic products	4728	4 728	4 728
	Porcelain products	220	220	220
Burbank Kevin	Total	20628	45	5 157
	Glasswares	4	4	4
	Metal utensils	737	737	737
	Plastic products	19302	19 302	19 302
	Silver utensils	585	585	585
Glou Jacques	Total	322977	1 846	53 830
	Glasswares	3087	3 087	3 087
	Metal utensils	119	119	119
	Plastic box	289740	289 740	289 740
	Plastic products	1316	1 316	1 316
	Porcelain products	28711	28 711	28 711

Grand totals can only be managed through the context menu:

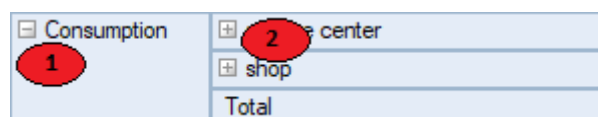


## Drill Up/Down

Grid data can be collapsed (drilled up) to exclude the values of the dimensions located below from processing or can be expanded (drilled down). These operations are performed using the "-" and "+" buttons located on both headers of the grid.

To group the data by the value of a certain dimension, you need to click the "-" button of the corresponding dimension. And conversely, to refine the data click the "+" button. These buttons are identified as '1' and '2' below:

1. drill up button
2. drill down button



Drilling up or drilling down changes the structure of the grid header and the data region. The two illustrations below show how the grid looks after drilling down on "service center" and then after drilling up on "Consumption":

Consumption	service center	Air filter	400,00	1	70,00
		Fuel filter	1 700,00	1	700,00
		Reducer oil	550,00	1	320,00
		Total	2 650,00	3	1 090,00
	shop	3 249,00	13	320,00	
Total		5 899,00	16	1 410,00	
Consumption		5 899,00	16	1 410,00	

Only the main Total shows when a dimension item is collapsed.

It is possible to collapse the grand totals. The result depends on whether or not the "Measures" field is placed in the collapsed axis. An axis without measures collapses with the hiding of all the cells except the grand total cell. An axis with measures can collapse grand totals for each measure independently. The collapse of measure grand totals hides all the cells of that measure. Collapsing the grand totals of all the measures hides all the cells except the grand total cells for each measure.

In this example the grand totals for measure "Amount" and "Cost" are collapsed:

Manager	Product	Measures (4)	
Amount			601230
Cost			11 706 187,56
Paid			5 219 178,52
Balance			6 487 009,04
Baldwin Janet	Paid		8 354,20
	Balance		238 965,45
	Metal utensils	Paid	643,00
		Balance	0,00
	Plastic products	Paid	6 453,60
		Balance	233 806,75
	Porcelain products	Paid	1 257,60
		Balance	5 158,70
Burbank Kevin	Paid		536 548,98
	Balance		885 426,09
	Glasswares	Paid	483,25
		Balance	0,00

And here the grand totals for all the measures are collapsed:

Manager	Product	
Amount		601230
Cost		11 706 187,56
Paid		5 219 178,52
Balance		6 487 009,04

## Drill Through

Double-clicking on a data cell opens a window showing a Detail Table which contains the source data rows used for calculation of the selected cell:

Amount	Category	Item	Price	Seller	Work price
1	Consumption	Air filter	400.00	service center	70.00
1	Consumption	Fuel filter	1,700.00	service center	700.00
1	Consumption	Reducer oil	550.00	service center	320.00

This Detail Table can be manipulated by:

- changing the visibility and size of the columns
- selecting data cells for copying to the clipboard
- exporting data to the same format as the main grid

The status bar shows the current row and total row count.

## Rotation

The grid layout can easily be changed by moving the dimensions between the three regions: filter region, row dimensions region and column dimensions region. In OLAP terms, this operation is called rotation, because it corresponds to rotating a multi-dimensional data array. Data rotation enables the same information to be analysed from different perspectives.

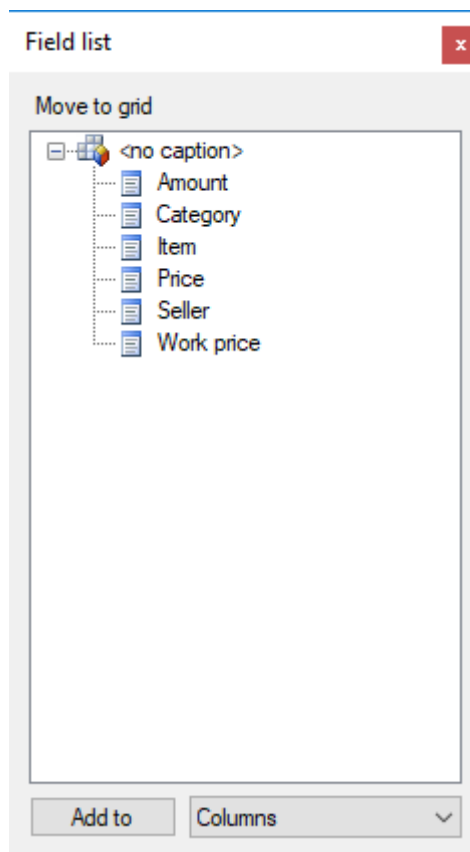
To rotate data in the grid the various dimensions must be dragged with the mouse from one region to another. As the mouse is dragged over the grid a special pointer shows where the dimension will go if the button mouse is released at that moment (see grid below):

Move filter fields here						
Field list			Measures (3)			
Category	Seller	Item	Price	Amount	Work price	
Grand total			16 179,00	27	3 410,00	
Consumption	service center	Air filter	400,00	1	70,00	
		Fuel filter	1 700,00	1	700,00	
		Reducer oil	550,00	1	320,00	
		Total	2 650,00	3	1 090,00	
	shop	Antifreeze	50,00	10		
		Motor oil	1 079,00	1		
		Oil filter	150,00	1	0,00	
		Transmission oil	1 970,00	1	320,00	
		Total	3 249,00	13	320,00	
	Total			5 899,00	16	1 410,00
Documents	auto market	contract of purchase	800,00	1		
	GAI	registration certificate	1 000,00	1		
	insurance agent	insurance	2 500,00	1		
	Total			4 300,00	3	0,00

This is the resultant grid after the "Item" dimension has been moved from the row dimensions region to the column dimensions region:

Move filter fields here							
Field list	Category	Seller	Item	Measures (3)	Air filter		
			Price	Amount	Work price	Price	Amor
Grand total			16 179,00	27	3 410,00	400,00	
Consumption	service center		2 650,00	3	1 090,00	400,00	
	shop		3 249,00	13	320,00		
	Total		5 899,00	16	1 410,00	400,00	
Documents	auto market		800,00	1			
	GAI		1 000,00	1			
	insurance agent		2 500,00	1			
	Total		4 300,00	3	0,00		
Goods	prev owner		1 100,00	4	500,00		
	shop		4 880,00	4	1 500,00		
	Total		5 980,00	8	2 000,00		

The move can also be made using the field list dialog of the grid:



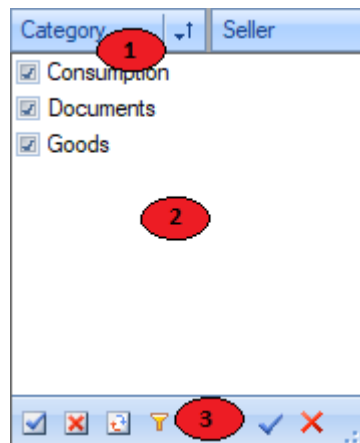
As well as partial rotation it is also possible to perform total rotation (grid transposition). This operation moves all the row dimensions to the column region and all the column dimensions to the row region. Unlike partial rotation, grid transposition does not require the recalculation of data cells and as a result is instantaneous. Transpose a grid using the 'Transpose' button on the toolbar. Our transposed grid looks like this:

Move filter fields here					
Field list	Category	Seller	Item		
Measures (3)	Grand total	Consumption			
		service center			
		Air filter	Fuel filter	Reducer oil	Total
	Price	16 179,00	400,00	1 700,00	550,00
Amount	27	1	1	1	3
Work price	3 410,00	70,00	700,00	320,00	1 090,00

## Filtering

Data filtering is performed by excluding certain dimension values from the analysis. Please note that it does not matter in which region of the grid this dimension is located (filter region, row dimensions region or column dimensions region). Filtered values are not only hidden in the grid header but are also excluded from measures and Totals calculations.

Values of dimensions can be filtered using the drop-down list which is opened with a mouse-click on the dimension filtering button ('1' below):



Key to the grid above:

1. dimension filtering button
2. list of dimension values
3. drop-down list toolbar

Checked values in the drop-down list are visible in the grid. If un-checked, all the corresponding rows of the source data are excluded from grid calculations. Values are toggled either by clicking on its checkbox or by selecting the value and pressing the spacebar on the keyboard. Holding the Ctrl button down while clicking a checkbox turns off all the values except the clicked one.

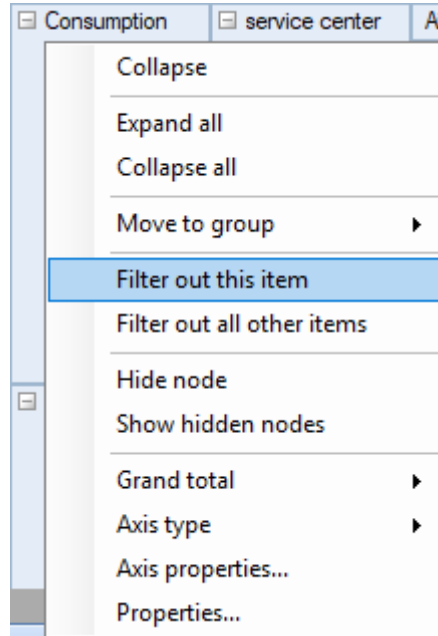
Depending on the grid settings, changes may be applied automatically or by the Apply button.

The toolbar may contain the following commands:

Icon	Action
<input checked="" type="checkbox"/>	Mark all values as visible
<input checked="" type="checkbox"/>	Mark all values as hidden
	Toggle value visibility
	Show only the values that are not filtered by other dimensions

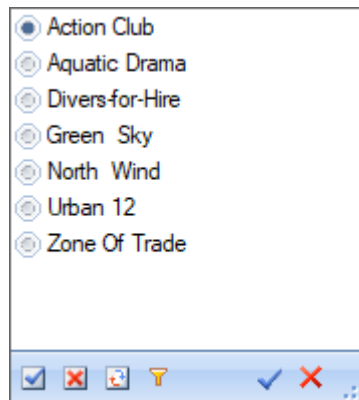
Icon	Action
✓	Apply changes and close list
✗	Cancel changes and close list

Filters can also be applied through the grid context menu:

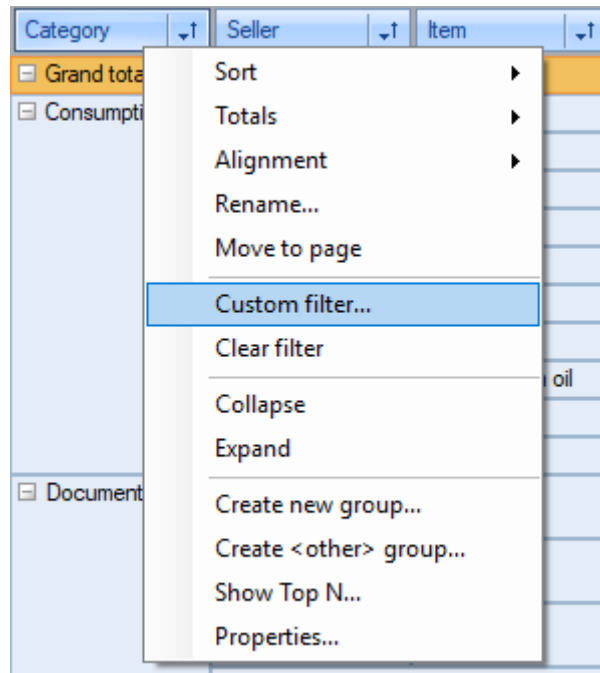


The Filter drop-down list can also be configured as radio buttons instead of checkboxes.

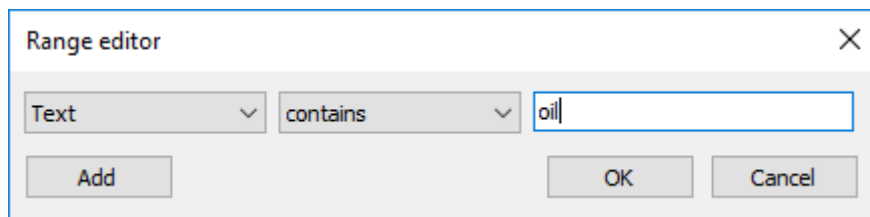
Radio buttons allow the selection of only one value at a time as visible:



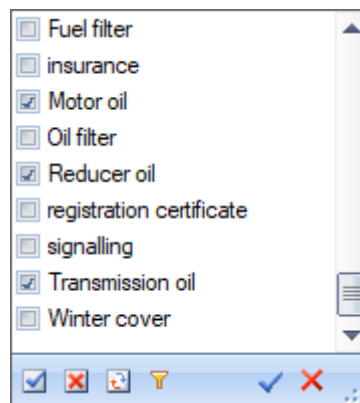
The Dimension values Range Editor offers conditional filtering of values. It is opened from the dimension context menu item "Custom filter...":



For example, the "Product" dimension can be set to filter out all values except where "Text contains 'utensils' substring":

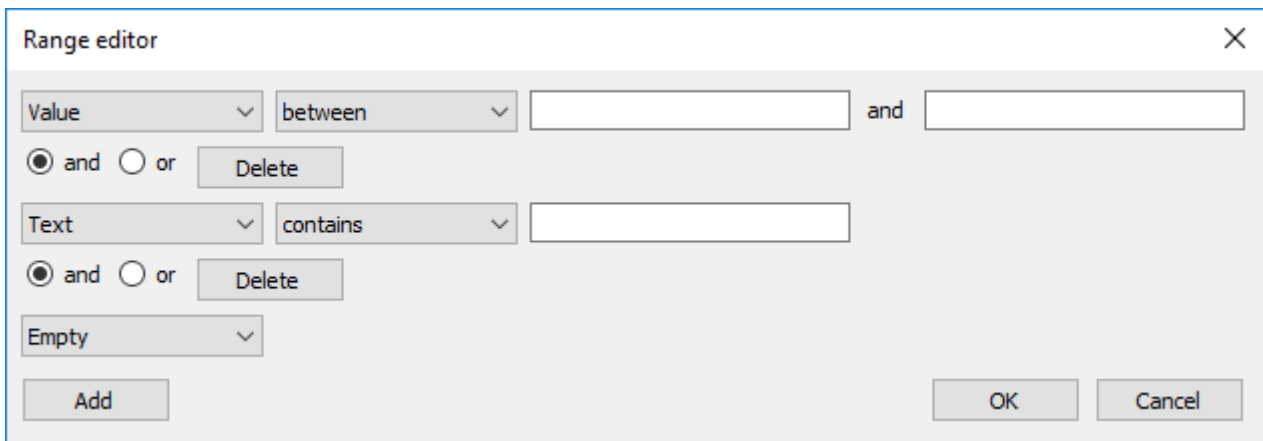


This is equivalent to:



The Range Editor can create multiple, simultaneous conditions of varying type:





Filtering affects measures and the calculation of Totals, but the grid can also be set to simply hide particular nodes in the headers. In this case the hidden nodes do not affect measures and the calculation of Totals. Node hiding can only be set using the context menu.

## Sorting

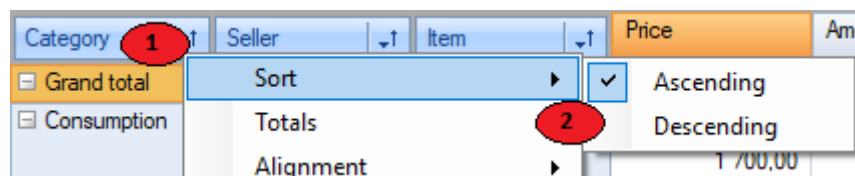
All the data in the grid is displayed in sorted order, irrespective of the original order in the source table. The sort order is either ascending or descending. Numeric values and date/times are sorted in their natural order, ascending or descending. Strings are sorted alphabetically, ascending or descending.

Every dimension in the grid can have its own sort order. For example, rows for the "Category" dimension can have ascending order and those for the "Seller" dimension descending order. The default sort order is ascending for all dimensions.

The sort order for a selected dimension is changed either by:

1. double-clicking on the dimension title
2. or selecting the required order from the dimension context menu (numbered '2' in diagram below)

The current dimension sort order is marked with a special symbol (numbered '1' in diagram below):



Apart from the sort order, it is also possible to set the sort type separately for lines and columns. There are three types of sorting:

1. by axes values - the default sort type
2. by measure Totals - data is sorted by the Total values for a measure - if there are several measures then only the values of the measure associated with the selected cell are included in the sort
3. by active row/column - sorting is similar to the previous type, but instead of the Total values of the row/column the values of the current row/column (selected in the grid) are used

The sort type is selected through the toolbar for both row dimensions and column dimensions.

## Grouping

FastCube can collect several dimension values into a group. Grouping is a two stage process - group creation followed by population with values.

Grouping features:

- unlimited number of dimension groups
- empty groups allowed
- empty groups not shown on axes
- groups must have unique names inside a dimension
- a dimension value cannot belong to more than one group
- inclusion of a dimension value in one group automatically excludes it from all other groups
- a dimension value can be excluded from grouping
- a system group "Others" can be created, which includes all values not belonging to any other group
- group deletion automatically excludes all the values it contains
- group creation, deletion, renaming, value inclusion and exclusion can be performed while the cube is active
- a filter window shows all groups and their members : group filtering state depends on member states
- moving a dimension from one region to another does not reset groups
- dimension groups are saved in cube files together with the dimension members

The axis shows dimensions having groups at two levels: group level and member level. The group level contains dimension group names and also dimension values that do not belong to any group. The member level contains the values which belong to the groups. A group can be in a collapsed state, in which case the group members are not shown. If all groups are collapsed then the member level is not shown.

Beware! any operation involving a group, including collapse/expansion, causes measure recalculation because of the change to the axis.

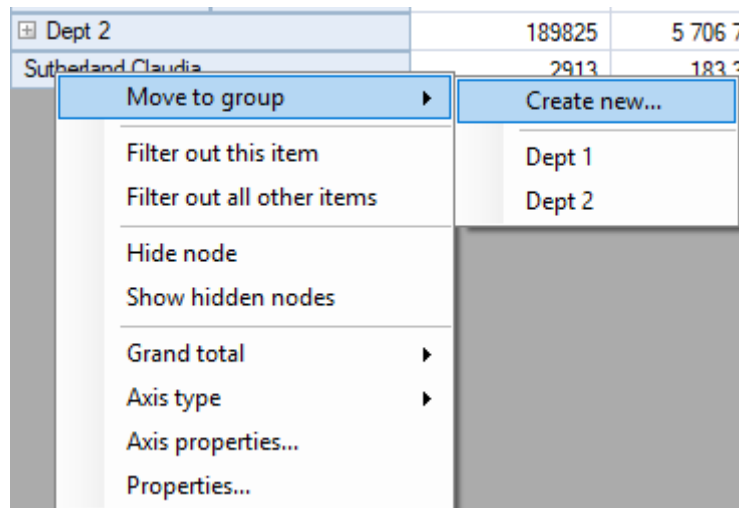
An example of a grid where the "Manager" dimension has no grouping:

Product	Customer			
Field list	Measures (4)			
Manager	Amount	Cost	Paid	Balance
Grand total	601230	11 706 187,56	5 219 178,52	6 487 009,04
Baldwin Janet	4951	247 319,65	8 354,20	238 965,45
Burbank Kevin	20628	1 421 975,07	536 548,98	885 426,09
Glon Jacques	322977	3 422 430,60	71 131,63	3 351 298,97
Guckenheimer Scott Jr	59936	724 337,69	21 410,20	702 927,49
Montgomery Jennifer	6957	364 227,36	1 592,20	362 635,16
Nordstrom Carol	314	15 550,20	15 503,20	47,00
Parker Robert	10490	218 139,22	188 610,63	29 528,59
Phong Leslie	68100	1 816 894,39	1 574 939,34	241 955,05
Reeves Roger	26879	998 331,20	843 409,90	154 921,30
Steadman Walter	77085	2 293 621,88	1 903 866,24	389 755,64
Sutherland Claudia	2913	183 360,30	53 812,00	129 548,30

The same grid with the dimension "Manager" having two groups, "Dept 1" and "Dept 2", each manager having been moved to the correct Department group:

Product		Customer					
Field list		Measures (4)					
Manager	Manager(...)	Amount	Cost	Paid	Balance		
Grand total		601230	11 706 187,56	5 219 178,52	6 487 009,04		
Dept 1	Total	408492	5 816 063,01	637 445,01	5 178 618,00		
	Baldwin Janet	4951	247 319,65	8 354,20	238 965,45		
	Burbank Kevin	20628	1 421 975,07	536 548,98	885 426,09		
	Glon Jacques	322977	3 422 430,60	71 131,63	3 351 298,97		
	Guckenheimer Scott Jr	59936	724 337,69	21 410,20	702 927,49		
Dept 2		189825	5 706 764,25	4 527 921,51	1 178 842,74		
Sutherland Claudia		2913	183 360,30	53 812,00	129 548,30		

Note that group "Dept 1" is expanded and group "Dept 2" is collapsed. Also, manager "Sutherland Claudia" does not belong to any group since she works in "Dept 3". Let's create a group for her. First open the context menu by right-clicking on value "Sutherland Claudia". The "Move to group" options are : add to a previously created group, or start a new group. Let's choose "Create new...":



And enter the department name:

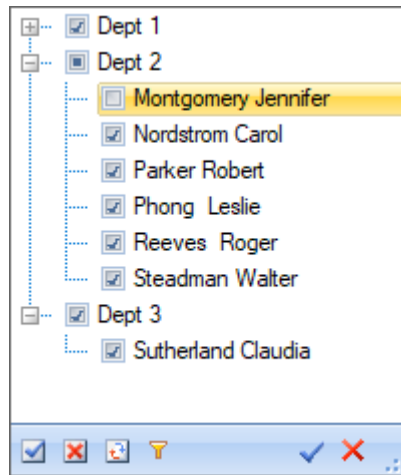
Create a new group ✕

Enter a new group name:

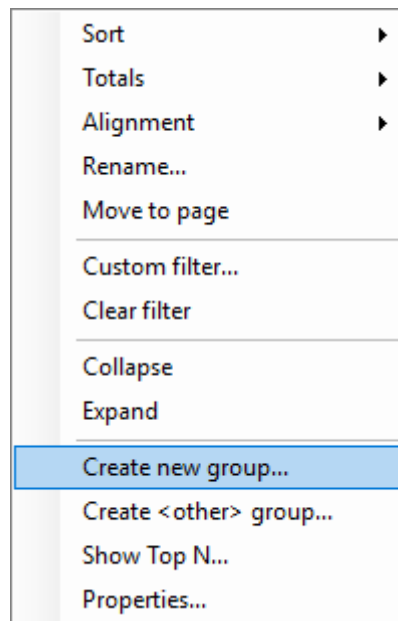
The new group is created and "Sutherland Claudia" is moved to it:

Manager	Amount	Cost	Paid	Balance
Grand total	601230	11 706 187,56	5 219 178,52	6 487 009,04
Dept 1	408492	5 816 063,01	637 445,01	5 178 618,00
Dept 2	189825	5 706 764,25	4 527 921,51	1 178 842,74
Dept 3	2913	183 360,30	53 812,00	129 548,30

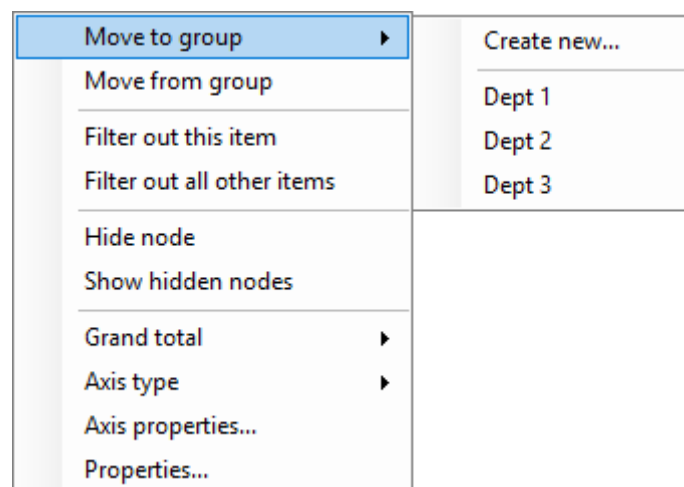
The filter drop-down list for the "Manager" dimension now looks like this:



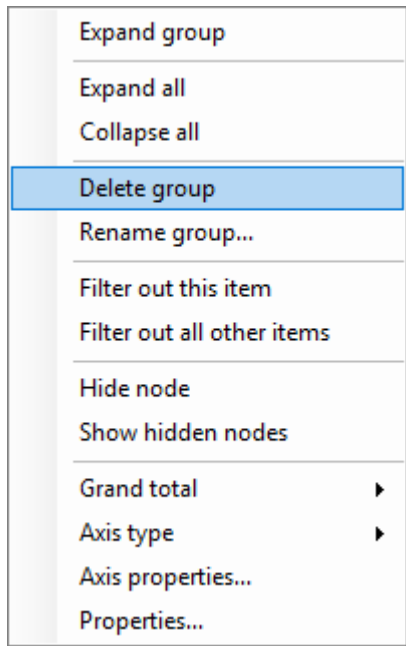
And the context menu for the dimension is:



While the context menu for a dimension member is:



And the context menu for the dimension group is:



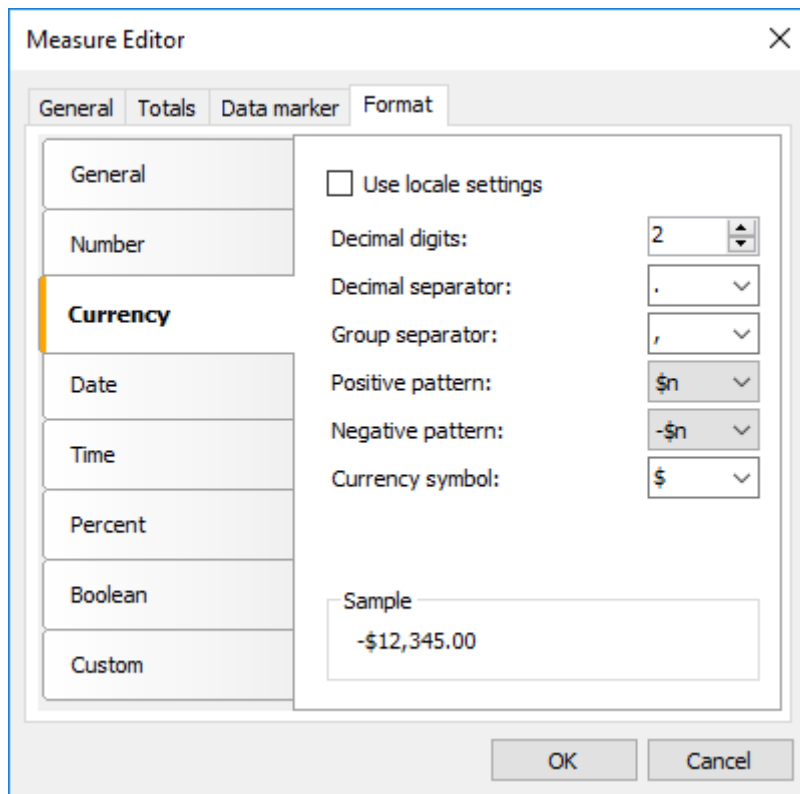
## Formatting, highlighting

### Data Formatting

By default a grid formats data according to the field type and the precision. For numeric measures the grid uses a number format having the same precision as the database field. But sometimes one or more measures need to be presented differently. For example, let's change the format of the "Price" measure to currency:

Category	Seller	Item	Price	Amount	Work price
Grand total			16 179,00	27	3 410,00
Consumption	service center	Air filter	400,00	1	70,00
		Fuel filter	1 700,00	1	700,00
		Reducer oil	550,00	1	320,00
		Total	2 650,00	3	1 090,00

The Format Editor is opened either from the cell's context menu or through the toolbar:



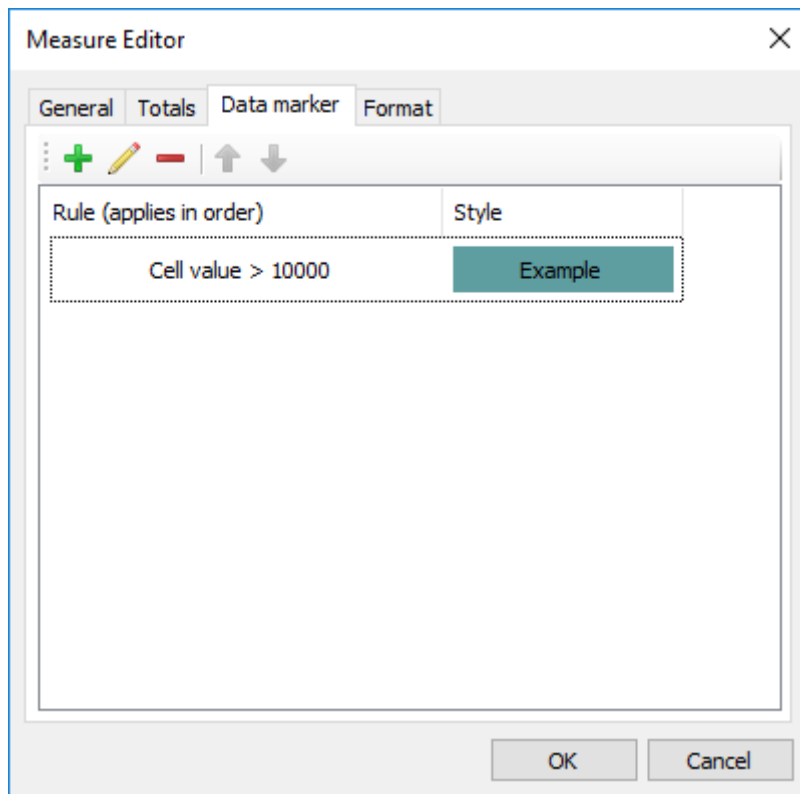
This editor lists the format categories on the left with the corresponding formats on the right. The format string, group and decimal separators can be altered if necessary.

Category	Seller	Item	Price	Amount	Work price
Grand total			\$16,179.00	27	3 410,00
Consumption	service center	Air filter	\$400.00	1	70,00
		Fuel filter	\$1,700.00	1	700,00
		Reducer oil	\$550.00	1	320,00
		Total	\$2,650.00	3	1 090,00

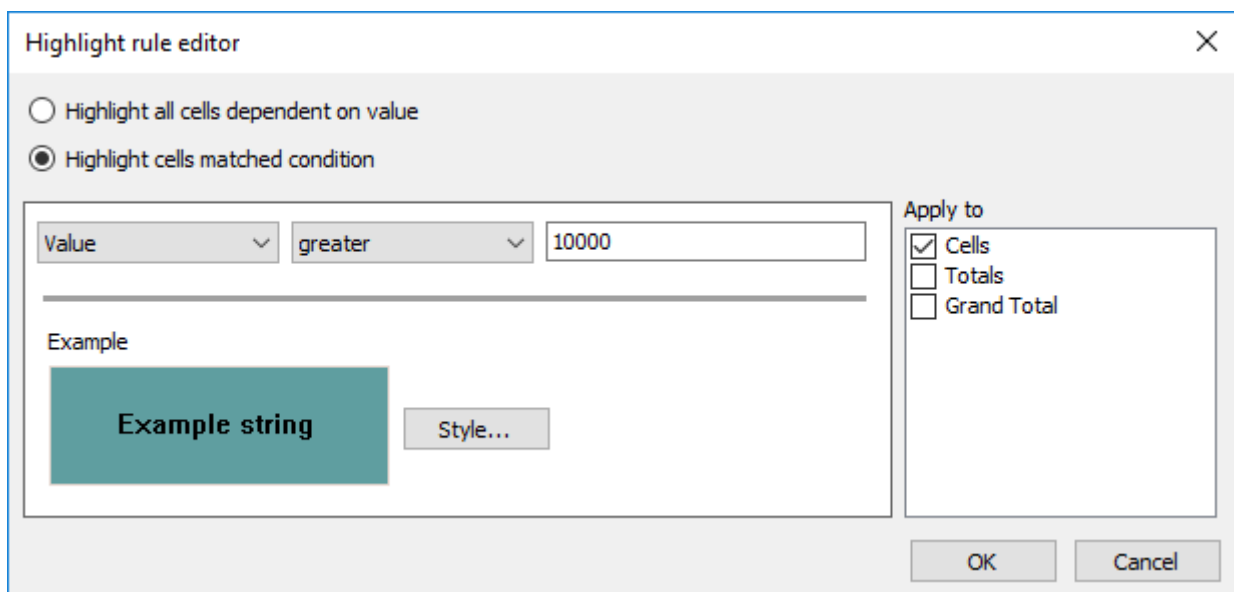
## Conditional highlighting

The grid allows custom drawing styles to be set up for particular measure values. For example, numbers inside or outside a particular range may need to be highlighted.

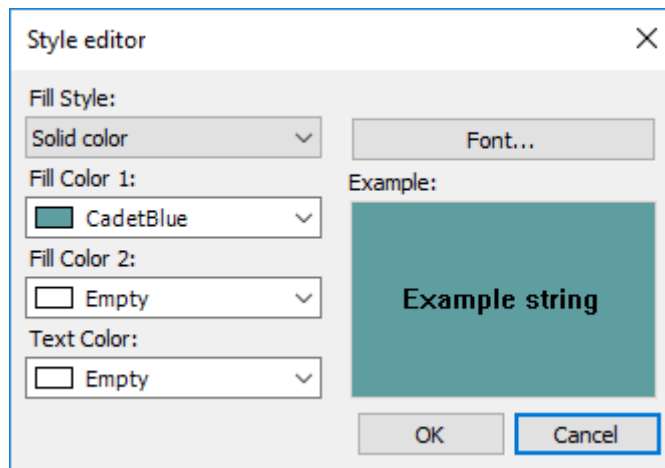
Before highlighting measure values a new highlight rule must be created in the measure editor dialog:



Highlight rules can process measure data as value, text or data, or as NULL. Allowable conditions depend on the type of the measure's data. So, for example, value processing allows "greater" and "lower" conditions while text processing allows "contains" and "not contains" conditions, which look for a substring within a text string:



The "Style..." button opens the drawing style editor:



The style editor has options for the background fill style and color, and text style and color. Solid fill style only requires a "Fill Color 1" value, but gradient fills also require a "Fill Color 2" value.

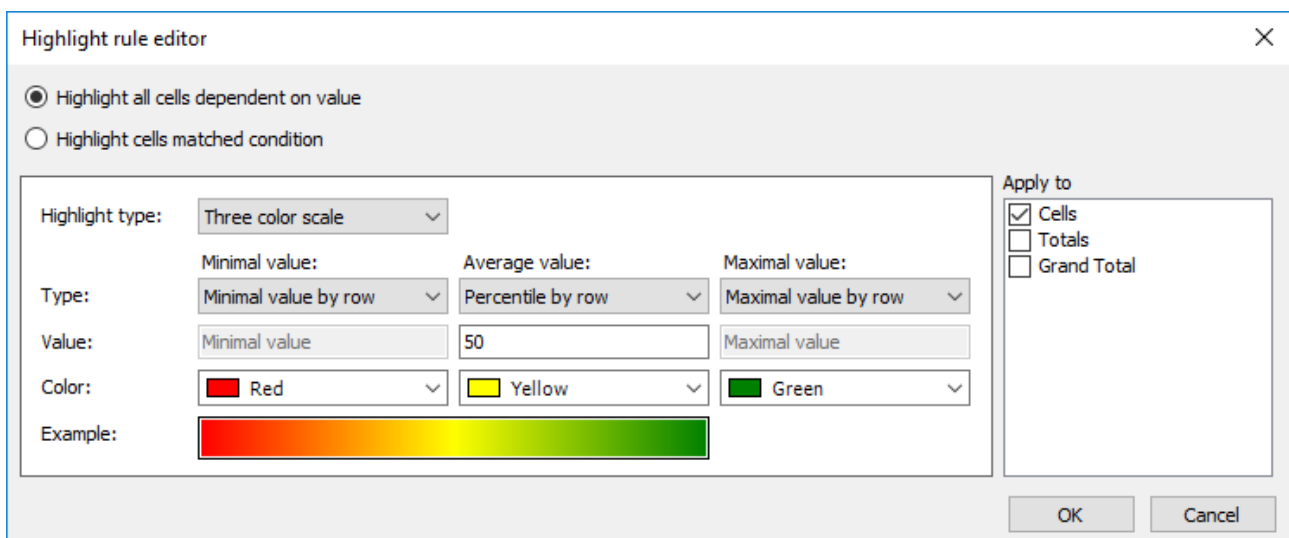
Here is an example after applying different conditional styles to some measure cells:

Paid
5 219 178,52
1 592,20
188 610,63
<b>1 574 939,34</b>
843 409,90

## Continuous highlighting

As well as the conditional cell highlighting described above, FastCube.Net can also apply four different types of highlight known as continuous highlighting, which applies highlighting to all cells dependent on their values. Each type of continuous highlighting is described below.

1. Two color scale and
2. Three color scale



Color scale highlighting fills each cell's background with a color calculated from the given color gradient scale. The editor dialog sets the values for the extreme points and their corresponding colors, and also for one intermediate point for the three color scale. The values



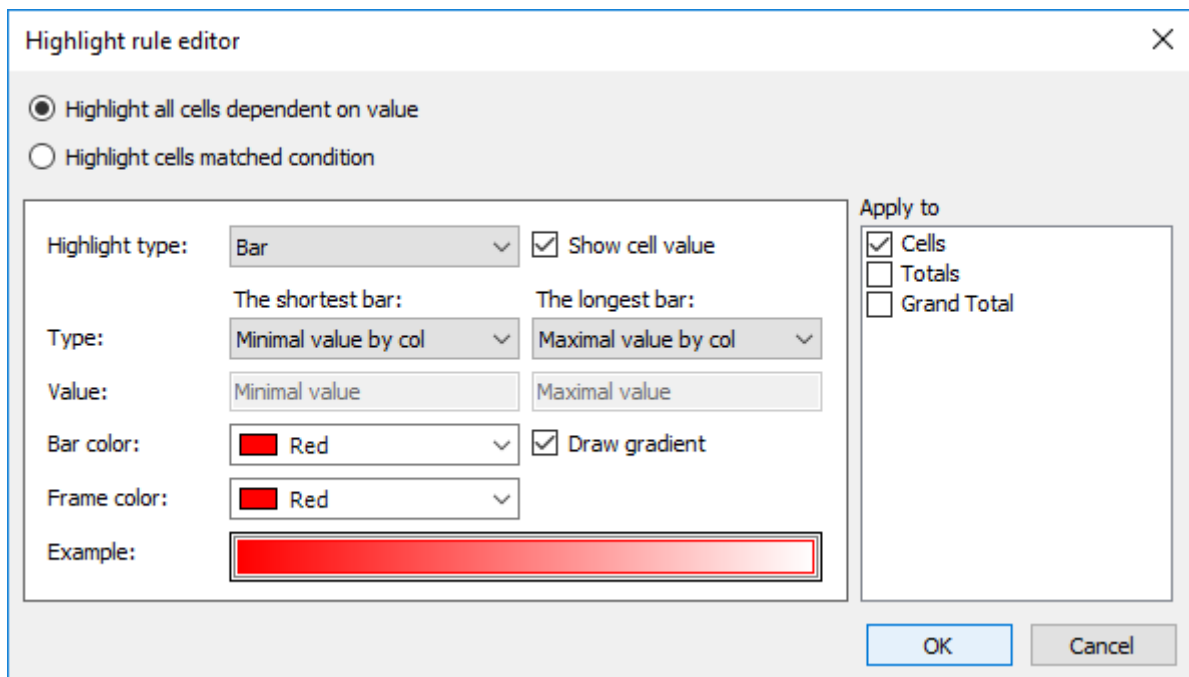
for the points can be set either as absolute numbers ("Number" option) or as one of the following relative options:

- Minimum by row/column - minimal measure value in row/column
- Percent by row/column - value field sets the percent relative to minimal and maximal measure values in row/column
- Percentile by row/column - value field sets the percentile relative to measure values in row/column

An example of two color and three color scale highlighting:

Price	Work price
16 179,00	3 410,00
400,00	70,00
1 700,00	700,00
550,00	320,00
2 650,00	1 090,00
50,00	
1 079,00	
150,00	0,00
1 970,00	320,00

### 3. Bar



This type of highlighting draws a colored bar within the measure cell. The length of the bar depends on the values set under the Shortest bar and the Longest bar properties. The property values, as previously, can be set with either the absolute or the relative option. The bar can be drawn either in a solid color or with a gradient color to white. The frame color for the bar is set independently. The cell value can also be hidden for this type of highlighting, leaving only the bar visible.

An example of bar highlighting with and without visible cell values:

Price	Work price
16 179,00	3 410,00
400,00	70,00
1 700,00	700,00
550,00	320,00
2 650,00	1 090,00
50,00	
1 079,00	
150,00	0,00
1 970,00	320,00
3 249,00	320,00

#### 4. Icon set

Highlight rule editor ✕

Highlight all cells dependent on value  
 Highlight cells matched condition

Highlight type: Icon set  Show cell value

Icon set: ↓ ↘ → ↙ ↑ Reverse order

Icon:	Value:	Type:
<span style="border: 1px solid gray; padding: 2px; display: inline-block; background-color: #90c190; color: white; font-size: 12px;">↑</span>	if value <span style="margin-left: 5px;">&gt;=</span> <span style="border: 1px solid gray; padding: 2px;">80</span>	Percent by row <span style="float: right;">▼</span>
<span style="border: 1px solid gray; padding: 2px; display: inline-block; background-color: #f1c40f; color: white; font-size: 12px;">↘</span>	if < 80 and <span style="margin-left: 5px;">&gt;=</span> <span style="border: 1px solid gray; padding: 2px;">60</span>	Percent by row <span style="float: right;">▼</span>
<span style="border: 1px solid gray; padding: 2px; display: inline-block; background-color: #f1c40f; color: white; font-size: 12px;">→</span>	if < 60 and <span style="margin-left: 5px;">&gt;=</span> <span style="border: 1px solid gray; padding: 2px;">40</span>	Percent by row <span style="float: right;">▼</span>
<span style="border: 1px solid gray; padding: 2px; display: inline-block; background-color: #f1c40f; color: white; font-size: 12px;">↙</span>	if < 40 and <span style="margin-left: 5px;">&gt;=</span> <span style="border: 1px solid gray; padding: 2px;">20</span>	Percent by row <span style="float: right;">▼</span>
<span style="border: 1px solid gray; padding: 2px; display: inline-block; background-color: #e74c3c; color: white; font-size: 12px;">↓</span>	if < 20	

Example: ↓ ↘ → ↙ ↑

Apply to

Cells

Totals

Grand Total

OK
Cancel

In Icon set highlighting an icon related to the value inside the measure cell is drawn to the left of the cell's value. The editor dialog has options for sets of 3, 4 or 5 icons. Each icon used requires a value range to be set. The cell value can also be hidden for this type of highlighting, leaving only the icon visible.

An example of icon set highlighting with and without visible cell values:

Price	Work price
16 179,00	3 410,00
 400,00	 70,00
 1 700,00	 700,00
 550,00	 320,00
2 650,00	1 090,00
 50,00	
 1 079,00	
 150,00	 0,00
 1 970,00	 320,00
3 249,00	320,00

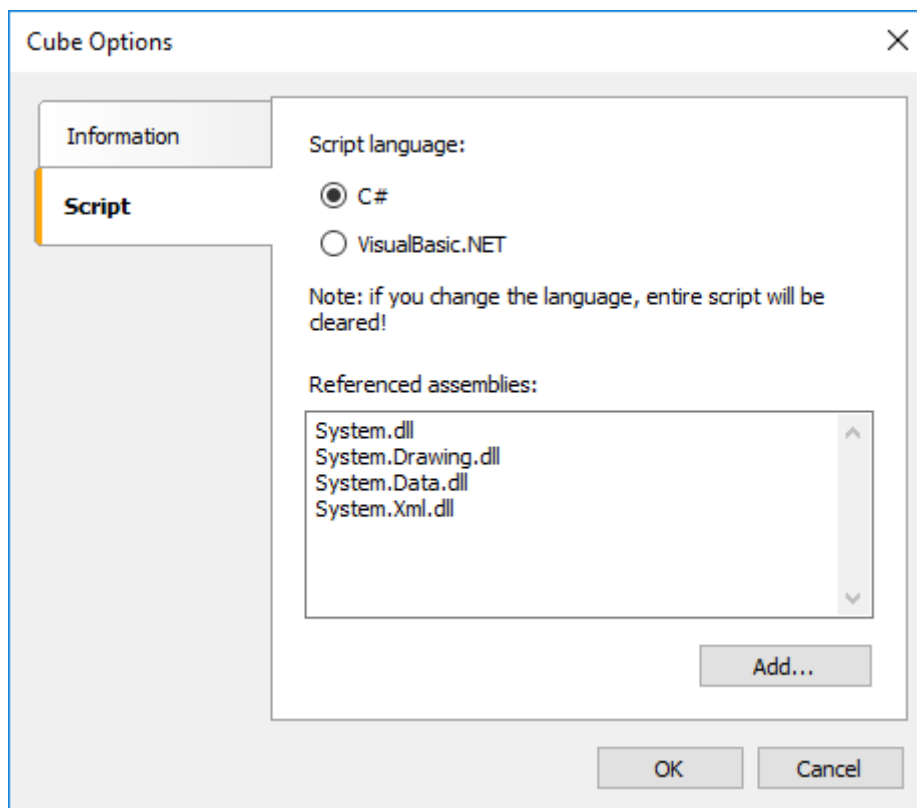
## Expressions

FastCube allows to use expressions for Measures, Totals calculation and for Measure filtering.

An expression is a code in C# or VB.Net language, which returns any value. For example:

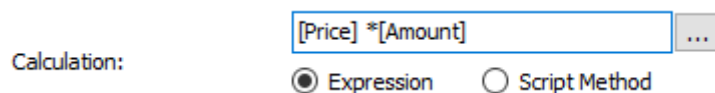
2 + 2

An expression should be written in a language chosen as a script in a cube. By default, it is C#. You can change the language by the button "Cube options" and by choosing the "Script" element in a window.

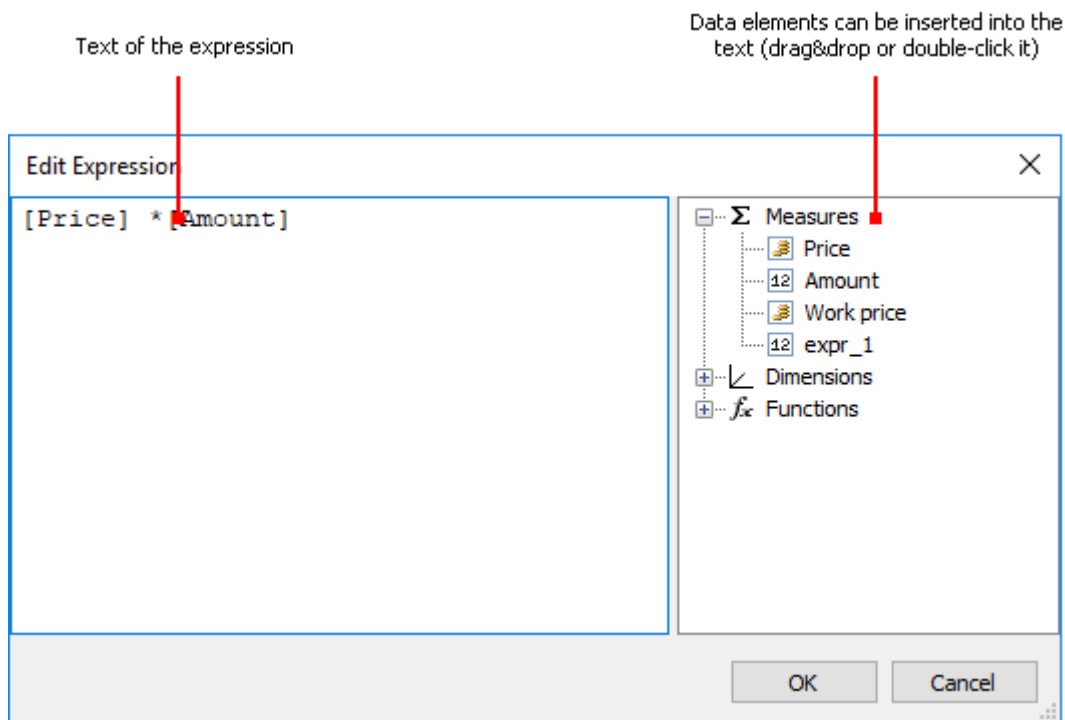


## Expression editor

To write an expression quickly, use the expression editor. It can be invoked in such places of FastCube UI, where you can type an expression:



The expression editor presents a window where you can write an expression and insert some elements into it:



## Using .Net functions

You can use any .Net objects in expressions. The following example demonstrates Max function use:

```
Math.Max(5, 10)
```

By default a report uses the following .Net assemblies:

```
System.dll
System.Drawing.dll
System.Windows.Forms.dll
System.Data.dll
System.Xml.dll
```

You have access to all .Net objects declared in these assemblies. If you need to have access to another assembly, add its name in report assemblies list. You can do it in the "Report|Options..." menu, by choosing the "Script" element in a window:

For example, if you want to use a function in your report which was declared in your application, add application assembly (.exe or .dll) in a report assemblies list. After that you can call the function by using namespace of your application. For example, if the following function is defined in application:

```
namespace Demo
{
    public static class MyFunctions
    {
        public static string Func1()
        {
            return "Hello!";
        }
    }
}
```

You can use it in your report in the following way:

```
Demo.MyFunctions.Func1()
```

If you add the "using Demo" line at the top of the report's script, it will allow you to shorten the syntax:

```
MyFunctions.Func1()
```

To refer to the function or variable that was defined in a script, just use its name:

```
myPrivateVariableThatIHaveDeclaredInScript  
MyScriptFunction()
```

You can use in an expression only those functions which return a value.

## Reference to slice data

Apart from standard language elements, you can use the following slice elements in expressions:

- measures;
- dimensions.

Any of these elements can be used in an expression, by including it in square brackets. For example:

```
[Price] + 1
```

A variable "Page", which returns Price measure value, is used in the expression. It is enclosed in square brackets.

## Reference to slice elements

For reference to measures and dimensions, the following format is used:

```
[MeasureName]
```

Let us look at the following example of using measures in an expression:

```
[Price] * [Amount]
```

Here it should be noted that: every measure and dimension has a definite data type. How element can be used in an expression depends on its type. For instance, in above mentioned example, Price is double and Amount is integer and that is why they can be used in such a way. In the following example, we will try to use "Item" dimension of string type, which will lead to an error:

```
[Amount] + " " + [Item]
```

The error occurs because, you never mix strings with numbers. For this, you need to convert the number into a string:

```
[Amount].ToString() + " " + [Item]
```

In this case, we refer to the "Amount" measure as if it is an integer variable. And it is so. We know that all expressions are compiled. All non-standard things (like referring measure) from a compiler's point of view are converted into another type, which is understandable to a compiler. So, the last expression will be turned into the following form:

```
((int)Slice.Measures.GetItemByName("Amount").CurrentValue).ToString() + " " +  
((string)Slice.Dimensions.GetItemByName("Item").CurrentValue)
```

As seen, FastCube changes reference to measures and dimensions in the following way:

```
[Amount] --> ((int)Slice.Measures.GetItemByName("Amount").CurrentValue)  
[Item] --> ((string)Slice.Dimensions.GetItemByName("Item").CurrentValue)
```

That is, we can use data column in an expressions as if it is a variable having a definite type.

For example, the following expression will return the first symbol of an employee's name:

```
[Employees.FirstName].Substring(0, 1)
```

## Script

Script is a higher-level programming language, which works inside the FastCube library. Script can be written in one of the following .Net languages:

- C#
- VisualBasic.Net

Using the script, you can do the following:

- perform measure calculation and totals calculation, which cannot be done via regular means of the FastCube engine;
- perform dimension values calculation;

In order to see the cube's script, press the "Code" button.

Script language can be set in the "Cube options" window (click on appropriate button on the toolbar). This is supposed to be done before writing various handlers, because when changing the language, the existing script gets deleted.

## General information

In the script, you can:

- add your variables, methods and properties to the main script class;
- create slice/cube elements events handler;
- add new classes to the script, if needed. A class can be added either before the SliceScript main class or after it.

You cannot:

- delete, rename or change the visibility area of the SliceScript main class;
- rename a namespace in which the main class is located.

When cube/slice builds, the following occurs:

- FastCube adds Slice and Cube variables into the script. This is done before compiling the script and allows you to refer them;
- a script is compiled, if it is not empty;
- the script class is initialized;
- performs further cube/slice building.

## Using .Net objects

In a script, you can use any .Net objects, which are defined in the following assemblies:

```
System.dll  
System.Drawing.dll  
System.Windows.Forms.dll  
System.Data.dll  
System.Xml.dll
```

Apart from that, you can use any object, defined in the FastCube assembly. If you need access to another assembly, add it to the list of assemblies. This can be done by "Cube options" button, by choosing the "Script" tab:

For example, if you want to use a function in your cube which was declared in your

application, add application assembly (.exe or .dll) in a cube assemblies list. After that you can call the function by using namespace of your application. For example, if the following function is defined in application:

```
namespace Demo
{
    publicstaticclass MyFunctions
    {
        publicstaticstring Func1()
        {
            return"Hello!";
        }
    }
}
```

Calling it in the script can be done in the following way:

```
string hello = Demo.MyFunctions.Func1();
```

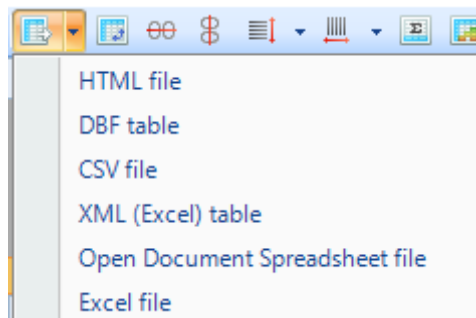
If you add the "using Demo" line at the top of the cube's script, it will allow you to shorten the syntax:

```
string hello = MyFunctions.Func1();
```

## Data export

FastCube.Net can export grid data to a number of formats.

Presently export is supported in five formats: Excel, ODS, XML, HTML and DBF. Export in each format is initiated from buttons on the toolbar:

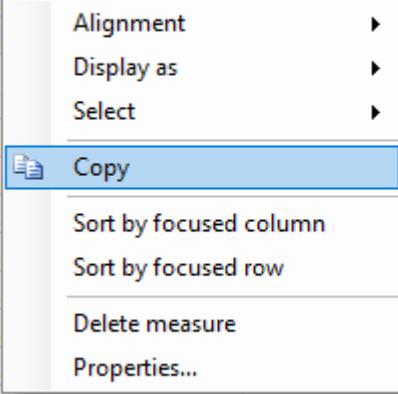


Exporting uses either the grid data or the source cube data, depending on the export format chosen. So, for example, Excel export uses the dimensions and measures in the grid, while DBF export is from the source cube data.

In addition to exporting data, FastCube can copy data from a selected region to the clipboard. A region in the grid can be selected either using the mouse or by pressing the Shift+arrow keys on the keyboard. Copying to the clipboard is done using the usual 'Ctrl+C' or 'Ctrl+Insert' combinations or using the context menu:



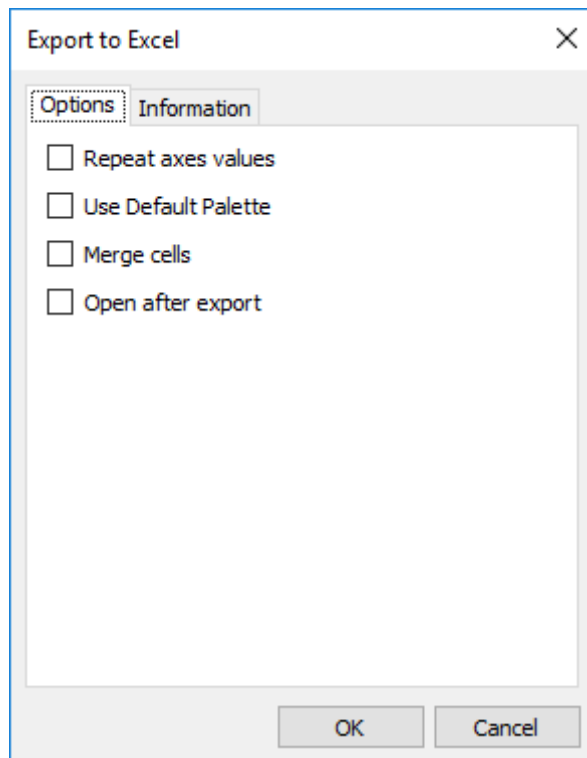
4951	247 319,65	8 354,20	238 965,45
3	643,00	643,00	0,00
4728	240 260		75
220	6 416		70
20628	1 421 975		09
4	483		00
737	212 175		10
19302	867 704		99
585	341 612		00
322977	3 422 430		97
3087	15 295		00
119	1 601,85	1 333,85	268,00



## Export to XLS

Excel is a Microsoft application for working with electronic spreadsheets. It is included in Microsoft Office. XLS files can also be opened with OpenOffice and LibreOffice.

When exporting to Excel format the output file settings are made in the export dialog:



Export settings:

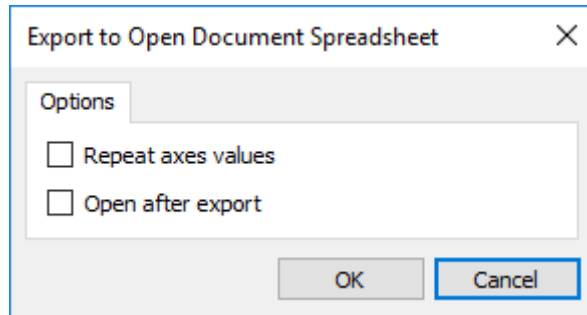
Setting	Action
Repeat axes values	Grouped values in grid headers are repeated for each row/column
Use default palette	Use standard Excel colors
Merge cells	Merge cells for ranges
Open after export	Open file immediately after export (requires Excel, OpenOffice or LibreOffice)

## Export to ODS

OpenDocument Format (ODF : OASIS Open Document Format for Office Application) was designed by OASIS and based on the XML format used in OpenOffice.

FastCube.Net supports export to the table type OpenDocument format (.ods). These files can be opened in OpenOffice or LibreOffice.

When exporting to ODF format the output file settings are made in the export dialog:



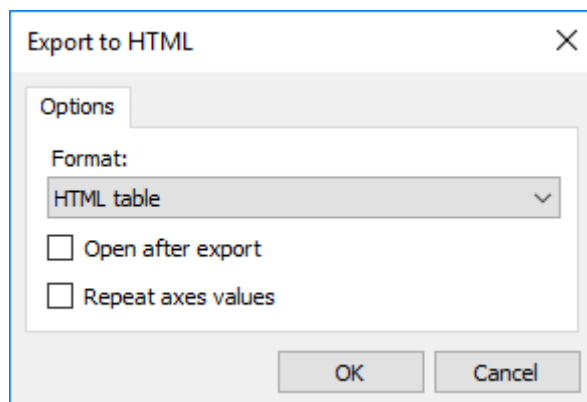
Export settings:

Setting	Action
Repeat axes values	Grouped values in grid headers are repeated for each row/column
Open after export	Open file immediately after export

## Export to HTML

HTML (Hypertext Markup Language) is regarded as the standard language for documents on the Internet. It is used for creating relatively simple but well designed documents. HTML supports hypertext linking as well as simple document layout.

When exporting to HTML format the output file settings are made in the export dialog:



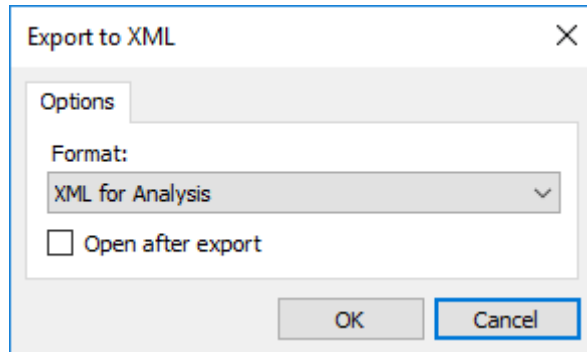
Export settings:

Setting	Action
HTML Format	Possible values: <i>HTML table</i> - export to HTML table (table tag); <i>Excel Worksheet HTML</i> - export to HTML format with Excel extensions
Repeat axes values	Grouped values in grid headers are repeated for each row/column
Open after export	Open file immediately after export (requires Excel, OpenOffice or LibreOffice)

## Export to XML

XML (Extensible Markup Language) is an extendable markup language. XML is intended for structured data storage and also for information exchange between different programs.

When exporting to XML format the output file settings are made in the export dialog:



Export parameters:

Setting	Action
XML format	See bellow
Open after export	Open file immediately after export (requires Excel, OpenOffice or LibreOffice)








XML formats:

Format	Description
Rowset	export cube data in the format compatible with ADO Rowset
Dataset	export cube data in the format compatible with .Net Dataset
DataPacket	export cube data in the format compatible with ClientDataset of Embarcadero Delphy
XML for Analysis	export slice data in XML for Analysis format

## Toolbar

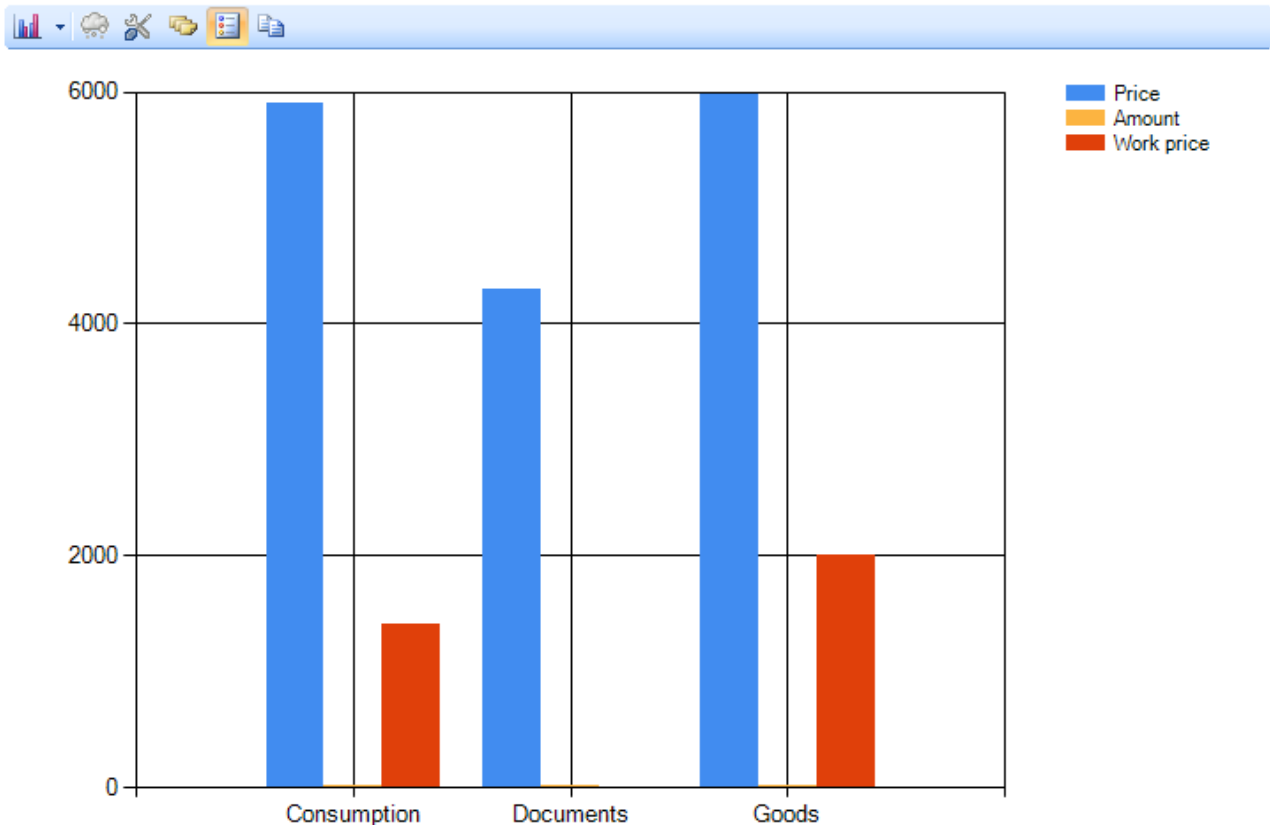
The grid toolbar contains the following set of buttons:

Icon	Command
	Save cube or scheme to file. Cube files have an mdc extension - they contain source data array, filters and formatting settings. Scheme files have an mds extension - as well as formatting settings they can also contain filters
	Load cube or scheme file. Also allows loading of an additional data array to the current cube (e.g. data from another time period)
	Clear formatting scheme
	Export data to external formats drop-down menu shows list of available formats
	Transpose grid
	Hide dimension values having only zero or empty data in rows
	Hide dimension values having only zero or empty data in columns

Icon	Command
	Select sort mode by rows
	Select sort mode by columns
	Open measure editor dialog
	Open measure highlight options
	Open measure format options
	Open field list window
	Show information about grid structure, data loading elapsed time, creation of headers and calculation of measures

## The Chart

The data in a grid may be presented as a cumulative 2D chart. The MsChart component is used for display of the chart. Depending on the settings, one of the major dimensions of the rows/columns is used for the chart series and the opposing dimension (i.e. the major dimension in the columns/rows) is used to supply the values in each of the series.



If the source grid does not contain row or column dimensions, an internal representation of a grid with row and column dimensions will be built inside the chart. The chart properties dialog is opened from the toolbar:

Chart properties
✕

Auto apply changes

Chart data source properties

Series and categories from dimensions

Use as series:

Dimensions from rows

Dimensions from columns

Use as categories:

Dimensions from rows

Dimensions from columns

Field count for Series:

Field count for Categories:

Measure index:







Skip null points

Data type for base axis:

The main property is the source of data for the series and categories. Possible sources are: the dimensions and the measures in the grid. The dimension region, number of dimensions for the names of categories/series and the measure index need to be selected appropriate to the selected source.

## The Chart Toolbar

The chart toolbar contains the following set of buttons:

Icon	Action
	Select type of chart series
	"Freeze" chart. Chart will not update automatically
	Chart properties editor
	Display labels on values of series
	Display chart legend
	Copy to clipboard

## Example of FastCube Use

This chapter describes examples of using FastCube.Net.

The examples can be seen in the demo program 'Advanced'.

### Example "Demography"

Let's show an example of using a multi-dimensional table holding demographic data for a number of countries. The source data was taken from the free service of the U.S. Census Bureau.

After some insignificant processing the data was loaded in the cube, so allowing the data to be displayed in a convenient way.

After loading the data into FastCube, start with the empty grid below (see 'Advanced' demo, 'Life Examples', 'Demography'):

Clicking on the "Field list" element, show this drop-down list:

What can be done with this data? Let's create several examples.

Example 1: Population by country

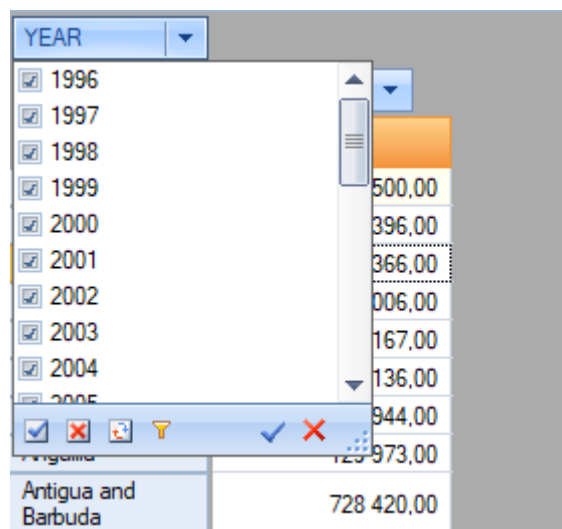
Drag the "Country" field to the row region and the "People" field to the data region, then move the measures field that is already located in the filters region into the columns region:

These steps give the grid below:

It is easy to see that the data in the grid is incorrect. The reason for this is that here the "People" measure sums the values for all years. So we need to specify the required year. To do this, we must drag the "Year" field into the filters region.

When a field is added to the filters region of the grid, an element appears to enable data filtering through a drop-down list.

Let's click on the drop-down list of the "Year" field:



As shown, the grid contains data for 11 years. But we only need the data for 2006 - this can be achieved in several ways:

1. by un-checking every year not needed, one by one (i.e. from 1996 to 2005)
2. by un-checking just 2006 and then clicking the "Inverse" button
3. by clicking the "Clear all" button and then clicking on 2006
4. by clicking on 2006 while holding down the 'Ctrl' key on the keyboard

After these changes to the "Year" dimension the grid looks like this:

Note that the title 'Year' is displayed in italic and the drop-down icon shows a funnel, which means that a filter is applied to it.

Also note that the values of the "Country" dimension are listed in the alphabetical order. For our analysis it would be more useful to list the countries in descending order of population. To do this, we need to sort the data by the value of the measure.

Click on any data cell in the grid, in the column which will be used for sorting the data, and select the option "Sort by focused column" in the toolbar:

This results in the grid displaying the data sorted by the value of the "People" measure (population), but in ascending order. Note that the column used for the sort is marked with a special symbol (diamond):

To change the sort order to descending, double-click on the "Country" dimension (or do the same on the drop-down dimension menu).

Adding Percentage and Rank columns to the measure values, shows that China takes the first place by population (20% of the total population) and the United States only takes third place (4% of the total population).

#### Example 2: The dynamics of population growth

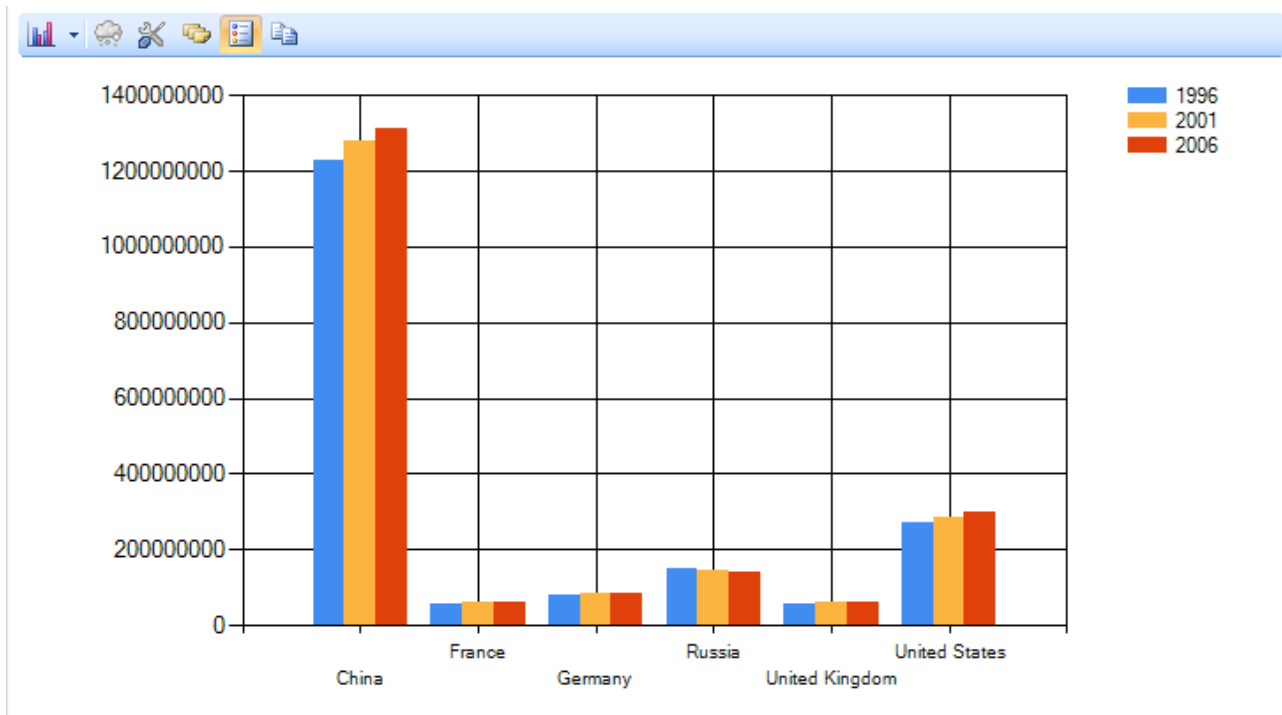
This example is based on the grid created above.

To achieve our objective we need to move the "Year" dimension from the filters region to the columns region, before the measures field, and add the values '1996' and '2001' to the years filter. As a result, the grid will show the population size in 5-year intervals between 1996 and 2006. Further, we can filter the country dimension to show just six countries: China, France, Germany, Russia, United Kingdom and the USA:

Now the first column shows the sum of the other columns and does not make any sense in this context, so we can hide the column. This is done through the context menu for the People item - 'Grand total - Hide':

The final grid looks like this:

This grid is useful for presenting the raw demographic data, but a far greater impact can be made when presented as a chart:



The chart clearly shows the relative sizes of the populations.

But to reveal which populations are increasing in size and which are falling, the grid data can be emphasized using the highlight tools:

### Example 3: Population by sex and age

In this example we will try to answer the following questions:

1. children of which gender are born more frequently?
2. does this gender ratio change with age?

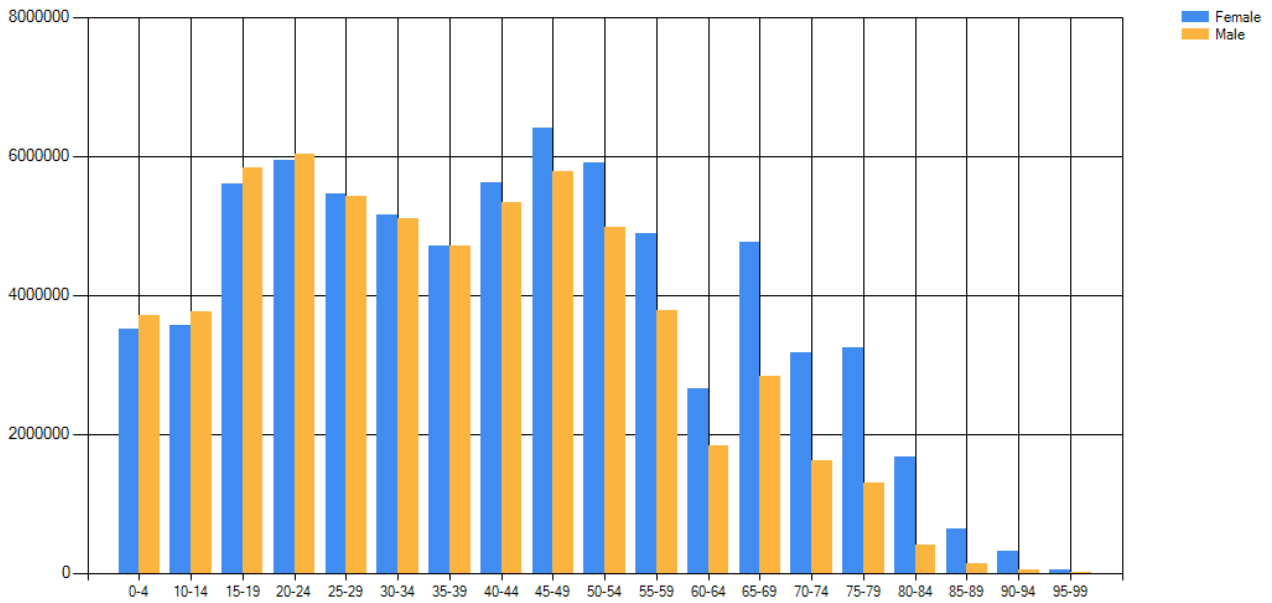
To answer these questions we will make the following changes to the grid:

1. drag the "Age\_group" field from the Field list into the row region
2. drag the "Year" and "Country" fields into the filters region
3. drag the "Sex" field from the Field list and the measures field to the measure fields region
4. exclude the values of '100+' and '5-9' from the values of the "Age\_group" dimension
5. leave only 2006 in the "Year" dimension
6. leave only Russia in the "Country" dimension

These steps produce the following grid:

Just as previously, the raw data is less informative than a chart (though the figures may be important for printed reports), so let's switch to the chart page:





A chart makes it easy to answer our initial questions:

1. more boys are born than girls
2. however, from the age group 40-44 more women survive than men

If you have not been able to reproduce these examples, you can open the ready-made examples from the Advanced demo, "Life Examples", "Demography".