

What's New

Essential Studio WPF Edition 2013

Volume 1



Contents

WPF	3
All BI Products.....	3
MDX Query Binding with Drill-Up and Drill-Down Operations.....	3
Essential BI PivotGrid.....	3
Sorting.....	3
Add and Remove Filters without Expressions	4
Essential Calculate.....	5
Support for Essential Calculate for WPF in samples and source.....	5
Essential Chart	7
Axis Label and Series Positions.....	7
Fast Spline Chart.....	8
Essential Grouping	8
WPF samples with Essential Grouping enhancements	8
Essential Spreadsheet	12
Undo and Redo Support.....	12

WPF

■ All BI Products

MDX Query Binding with Drill-Up and Drill-Down Operations

In all business intelligence controls, users can perform drill-up and drill-down operations by using MDX queries to manipulate data at different levels.

	Australia	Canada	France	Germany	United Kingdom	United States	Total
Internet Sales Amount							
FY 2002	\$2,568,701.39	\$573,100.97					\$7,072,084.24
FY 2003	\$2,099,585.43	\$305,010.69					\$5,762,134.30
FY 2004	\$4,383,479.54	\$1,088,879.50					\$16,473,618.05
FY 2005	\$9,234.23	\$10,853.70					\$50,840.63
Total	\$9,061,000.58	\$1,977,844.86					\$29,358,677.22

```
SELECT NON EMPTY ({{Hierarchize({{DrilldownLevel({{Customer}.
[Customer Geography].[All Customers]]}}) * {{MEASURES}.
[Internet Sales Amount]}} ) dimension properties member_type ON
COLUMNS, NON EMPTY ({{Hierarchize({{DrilldownLevel({{Date}.
[Fiscal].[All Periods]]}})} ) dimension properties member_type ON
ROWS FROM [Adventure Works] CELL PROPERTIES VALUE,
FORMAT_STRING, FORMATTED_VALUE
```

Manipulating OLAP Data through MDX Queries

■ Essential BI PivotGrid

Sorting

The values in a column can be sorted in ascending or descending order when its column header is clicked. The following sorting options are supported:

- All—sorts all columns in the Pivot Grid.
- Total sorting—sorts only the total columns.
- Grand total Sorting—sorts only the grand total columns.
- Column sorting—sorts columns other than the total and the grand total columns.
- None—disables sorting for all columns.

The default sorting option for the control is None.

Drop Filter fields here		Canada		France	
Am...	Qua...	Amount	Quantity	Amount	Quantity
▼ Bike	FY 2005	\$23,668,800.00	138	\$24,978,300.00	148
	FY 2006	\$11,853,900.00	66	\$8,278,500.00	55
	FY 2007	\$5,744,100.00	33	\$3,469,500.00	21
	FY 2008	\$1,955,700.00	10	\$1,948,800.00	12
	FY 2009	\$371,100.00	4	\$888,000.00	5
Bike Total		\$43,593,600.00	251	\$39,563,100.00	241
▼ Bus	FY 2005	\$2,976,600.00	23	\$5,380,200.00	31
	FY 2006	\$2,809,800.00	15	\$2,451,000.00	15
	FY 2008	\$765,600.00	5	\$542,100.00	4
	FY 2007	\$318,300.00	4	\$1,073,100.00	6
	FY 2009			\$293,700.00	1
Bus Total		\$6,870,300.00	47	\$9,740,100.00	57

Sorting Options

- Enable Sorting on all columns
- Enable Column Sorting
- Enable Total Column Sorting
- Enable GrandTotal Column Sorting
- Disable Sorting

Pivot Grid with Sorting Enabled

Add and Remove Filters without Expressions

Users can add and remove filters programmatically via the following options:

- Add
- InsertAt
- Remove
- RemoveAt
- Clear

Users can also apply filters to the pivot grid by passing a dimension name as an argument without using expressions.

Providing support to programmatic level support for filtering

Add Product Insert State at the index '0' Clear the filter

Remove Product Remove the item at Position 1

Product ▾	State ▾						
A...	Q...	Country ▾ ▲					
Date ▾ ▲	Canada		France		Germa		
	Amount	Quantity	Amount	Quantity	Amount	Quantity	
FY 2005	\$38,827,200.00	229	\$39,792,600.00	235	\$35,8		
FY 2006	\$18,820,200.00	103	\$13,621,200.00	89	\$14,9		
FY 2007	\$8,828,400.00	51	\$6,151,800.00	37	\$7,8		
FY 2008	\$4,270,500.00	26	\$3,688,200.00	23	\$5,9		
FY 2009	\$1,123,200.00	8	\$2,495,400.00	14	\$1,9		
Grand Total	\$71,869,500.00	417	\$65,749,200.00	398	\$66,6		

PivotTable Field List

Choose Field to add to report

- Product
- Date
- Country
- State
- Quantity
- Amount

Drag Fields between areas below

Report Filter Column Label

Product x ▾ Country x ▾

State x ▾

Filters in a Pivot Grid

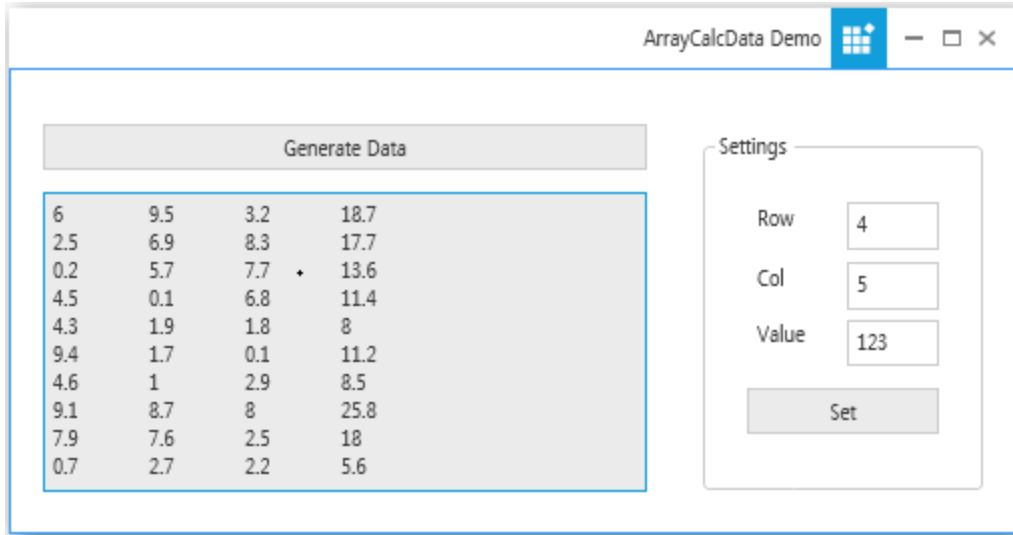
- Essential Calculate

Support for Essential Calculate for WPF in samples and source

Additional support for Essential Calculate for WPF adds the following samples and features to the control.

Array ICalcData Demo

Adds calculation support to a double array.



Working with CalcQuick Demo

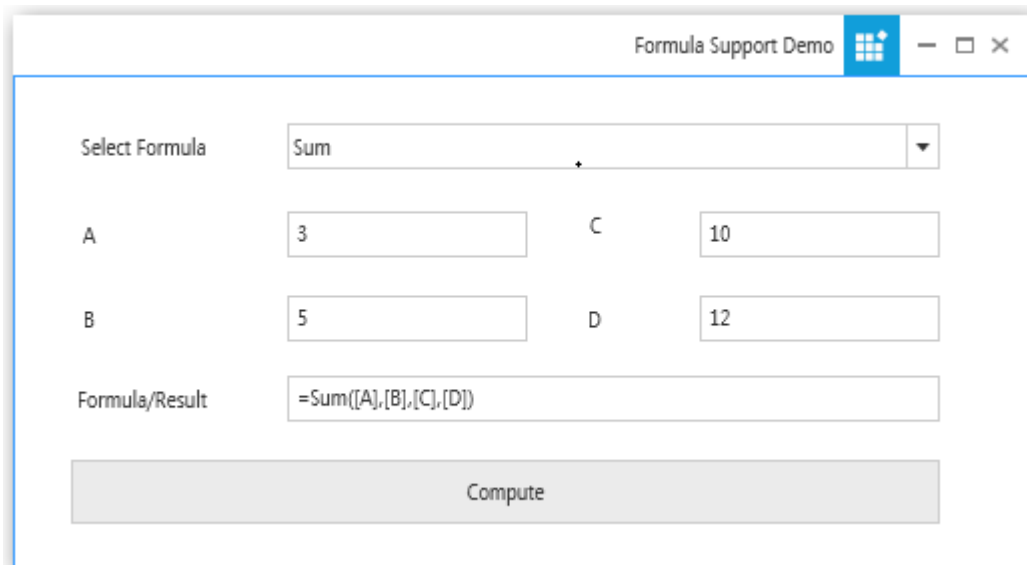
Illustrates several tasks associated with using CalcQuick objects.

XLS File Using XlsIO Demo

Illustrates the XlsIOCalcWorkbook class that allows Essential Calculate to work with XLS files.

Formula Support

Illustrates the formulas supported in CalcQuick.



■ Essential Chart

Axis Label and Series Positions

This feature allows users to set the range padding for the indexed axis, and specify a "between segment" mode that positions chart series and their labels between two axis lines. Users can optimize any series position based on their requirements.

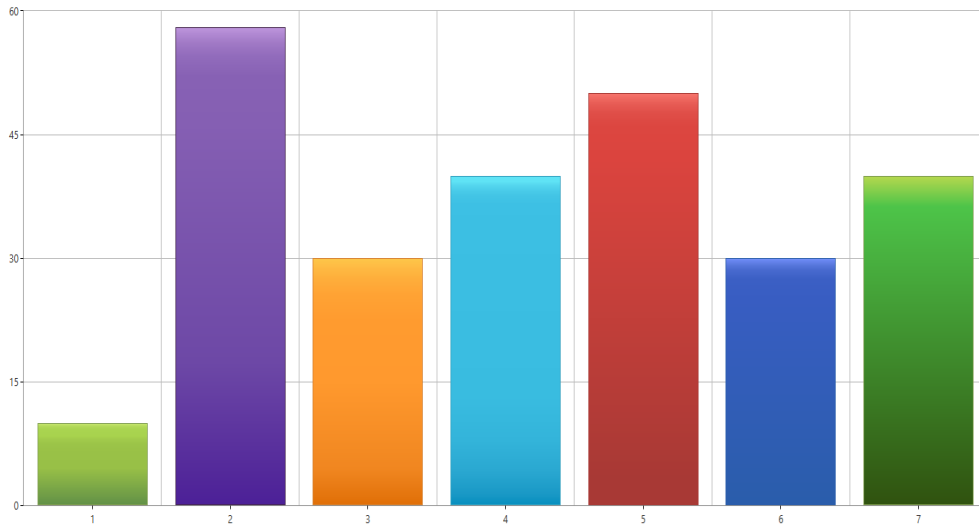


Chart Segments between Axis Lines

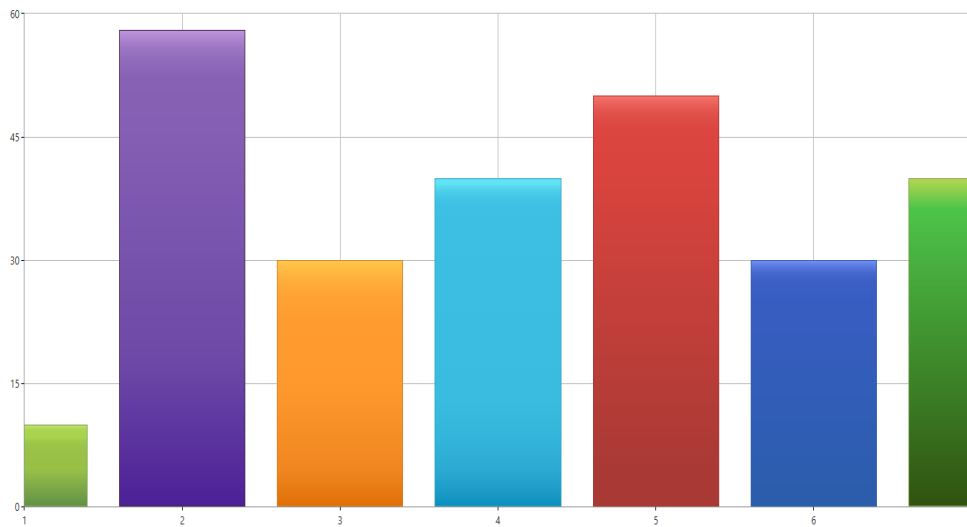
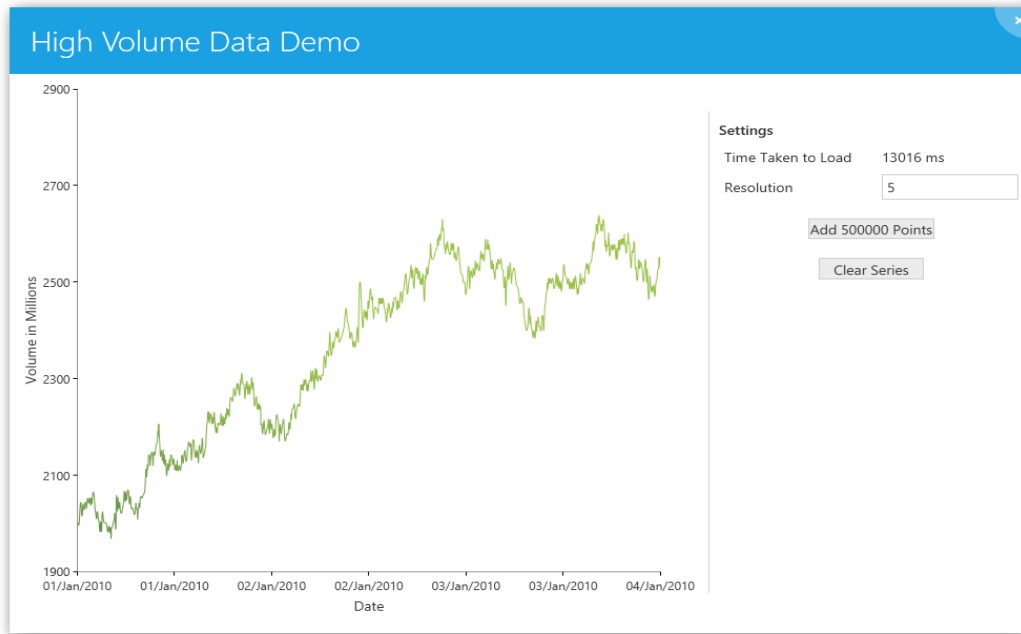


Chart Segments on Axis Lines

Fast Spline Chart

A fast spline chart enables users to load high-volume data and real-time data at much faster rates than a regular spline chart. It can load 500,000 points in 12,000 milliseconds.



Fast Spline Chart

- **Essential Grouping**

WPF samples with Essential Grouping enhancements

Some WPF samples have been enhanced with Essential Grouping capabilities. The improved samples are listed below.

Custom Summary Demo

The last two fields in this sample show custom counters. One counter automatically counts the total number of records, and the other counts the number of visible records using a filter set to [UnitPrice] <> 20. Changing the quantity in one record will automatically update all subsequent records.

Foreign-Key Reference Demo

This sample demonstrates setting up a relation between two tables using a foreign key.

Grouping Tutorial

This sample applies simple button handlers that can sort data based on a key, filter data based on a key, or add a summary.


The screenshot shows a window titled "Grouping Tutorial" with a standard Windows title bar. The interface is divided into several sections:

- Control Buttons:** A grid of buttons at the top left includes "Sorted on B using Default Sort", "Sorted on B using Custom Sort", "Group by property C", and "Group by properties C and D".
- Filter and Summary:** A "Filter" section contains a text input field with "D like d0*" and a "Summary" section with a dropdown menu set to "A" and a "Count" label.
- Clear and Display Buttons:** A vertical stack of buttons on the right includes "Clear Sorts/Filters/Groups", "Clear the Display Text", "Display Raw Data", "Reset Raw Data" (with a "10.00" spinner control to its left), "Iterate Thru DisplayElements", "Iterate Thru Elements", "Iterate Thru FilteredRecords", and "Iterate Thru Records".
- Expand/Collapse:** "ExpandAll" and "CollapseAll" buttons are located at the bottom right.
- Data Output:** A large text area on the left displays a list of operations: "Sorted On B using default Ascending", "GroupBy C", "GroupBy CD", "Sorted On B using custom Comparer", "GroupBy CD", "GroupBy C", "Sorted On B using custom Comparer", "Sorted On B using default Descending", "GroupBy C", "Sorted On B using default Descending", and "Sorted On B using custom Comparer".

Grouping Tutorial Demo

Grouping with Data Grid Demo

This sample displays statistics in labels based on values in the data grid. As you edit the items by inserting or deleting rows, the labels will automatically maintain proper values by using the grouping engine.

Grouping With Data Grid Demo 

A	B
476	968
589	622
687	914
476	534
368	819
445	435
461	126
850	839
429	131
421	268
275	663
239	190
308	528
614	873
518	35
900	392
546	350
899	116
731	960
426	918
458	981

Number of rows

Column	Maximum	Minimum	Total	Count
A	996	15	52498	100
B	995	14	51343	100

List-Item Reference Demo

This sample shows the RelationKind.ListItemReference relation.

Manual Relations Demo

This demo allows you to manually specify the relations in the grouping engine; however, the data set does not need any data relations.

Random Test Demo

This sample inserts random numbers into an IBindingList object that has been set as a data source of a grouping engine object. This means that the engine dynamically monitors the numbers and maintains the statistics as each new item is added to the collection.

Random Test Demo

Total number of random numbers	<input type="text" value="5338"/>	<input type="button" value="Start Test"/>
Count in bucket 3	<input type="text" value="27"/>	
Count in bucket 56	<input type="text" value="27"/>	
Average value of all random numbers	<input type="text" value="50.57"/>	
Median value of all random numbers	<input type="text" value="51"/>	
25 percentile level	<input type="text" value="26"/>	
75 percentile level	<input type="text" value="76"/>	

Number of seconds in test	<input type="text" value="1.00"/>
Refresh iteration	<input type="text" value="100.00"/>

Random Test Demo

Strong-Typed Collection Demo

This sample shows a hierarchical grid in which each parent record in the collection can have a collection of child nodes.

```

CaptionRow <CustomerCollection> 2-Items
RecordRow <AddNewRecord>: ID = , FirstName = , LastName =
RecordRow <Record>: ID = 536-45-1245, FirstName = Jo, LastName = Brown
CaptionRow <Level_0> 2-Items
RecordRow <Record>: ID = 537-45-1245, FirstName = Keith, LastName = Brown
CaptionRow <Level_1> 0-Items
SummarySection
RecordRow <Record>: ID = 247-12-5645, FirstName = Sven, LastName = Brown
CaptionRow <Level_1> 0-Items
SummarySection
SummarySection
RecordRow <Record>: ID = 246-12-5645, FirstName = Robert, LastName = Brown
CaptionRow <Level_0> 1-Items
RecordRow <Record>: ID = 538-45-1245, FirstName = Katie, LastName = Brown
CaptionRow <Level_1> 0-Items
SummarySection
SummarySection
SummarySection
Press any key to continue_

```

- Essential Spreadsheet

- Undo and Redo Support

- Changes made in a spreadsheet are tracked by two stacks: one for redo operations and one for undo operations. These stacks can be enabled or disabled by the user. Any number of changes can be grouped as a single transaction. Users can clear the stack when necessary.