

MultiBase Cosmos

Notes to version 5.4

BASE100

BASE 100, S.A.
www.base100.com

Contents

1. IMPLEMENTATIONS.....	3
2. IMPROVEMENTS.....	4
3. ENVIRONMENT VARIABLES.....	5
4. SIMPLECONTROL CLASS METHODS	6
5. TTCHARTDLL FUNCTIONS.....	7
6. BUG FIXES.....	12
ANNEX. COSMOS RUNTIME IN TABLETS.....	14
ADD COLUMNS TO THE GROUPING AND SORTING BAR	14
REMOVE COLUMNS FROM THE GROUPING AND SORTING BAR	15
HORIZONTAL SCROLL IN LIST BOX CONTROLS	15
ZOOM	16

1. Implementations

- We have made the necessary modifications to Cosmos so it can run on Windows 8 Tablets (see *Annex* at the end of the document).
- We have added a new function to the SetComputedColumn method of the SimpleControl class that allows to round a decimal number to the specified number of decimal places.

2. Improvements

- The Interactive SQL displays the line where the cursor is placed.
- There has been made a number of implementations in the TTCHARTDLL library. From this version you can create grouped bar, stacked bar and percentage bar to allow for comparisons between multiple data series. You can add markers to the line charts.

The plot area can be transparent.

- The length of the second parameter of the PutEnv method have been expanded.
- Added the possibility of scrolling with the mouse wheel in Box-type controls.

3. Environment Variables

- **AUTOSHOWONSCREENKEYBOARD.** When the Cosmos runtime is executed on a Tablet with Windows OS and an edit field receives the focus the runtime displays a virtual keyboard. To not display the virtual keyboard is necessary to define this environment variable.

Allowed values:

YES or TRUE The keyboard will be displayed (default behavior).

NO or FALSE The keyboard will not be displayed.

To prevent virtual keyboard onscreen on personal computers will be added to the cosmos.ini file or project environment file the environment variable AUTOSHOWONSCREENKEYBOARD=FALSE.

- **ONSCREENKEYBOARDPATH.** This environment variable allows to modify the keyboard displayed. The variable must specify the absolute path of the program that activates the virtual keyboard. The default keyboard is Taptip.exe.

4. SimpleControl Class Methods

- **SetListSpreadSheetNavigation.** This method allows to emulate the navigation in the cells of a List Control of Sql type or String type, similar to how you are navigating a spreadsheet.

Syntax:

```
SetListSpreadSheetNavigation(show as Boolean)
```

Parameters:

show	If the value of this parameter is TRUE, the navigation in the cells when you use the keyboard arrows will be similar to how you are navigating a spreadsheet. If the value is FALSE, the navigation in the cells will be the same as the previous versions. FALSE is the default value.
------	---

- **SetListRecursiveAggregate.** This method tells the runtime of Cosmos that the calculation of aggregate functions is not done recursively, ie, the result of these functions (TotalizeColumn method) only take into account the values of the nodes in the same level.

This method is only available in Tree List Box controls.

Syntax:

```
SetListRecursiveAggregate(isrecursive as Boolean)
```

Parameters:

isrecursive	Indicates whether the calculation of the aggregate functions will be recursive or not.
-------------	--

The available values are: FALSE and TRUE. TRUE is the default value.

5. TTCHARTDLL Functions

- **ChartLayerSetLineWidth.** This function sets the thickness of the line in line chart type and step line chart type.

Syntax:

```
ChartLayerSetLineWidth(hWindow as integer, nchartlayer as integer,  
lineWidth as decimal) return boolean
```

Parameters:

hWindow	User's Control window handle.
nchartlayer	Chart's layer's ID where the property will be changed.
lineWidth	Line thickness. The default value is 0.

Returns: TRUE if the operation was executed without errors.

- **ChartLayerSetDot.** This function allows to define individual data markers for all the elements of the data set or line type layer in two dimension's charts.

Syntax:

```
ChartLayerSetDot(hWindow as integer, nchartlayer as integer, dotSize  
as decimal, dotColor as integer, dotType as integer) return boolean
```

Parameters:

hWindow	User's Control window handle.
nchartlayer	Chart's layer's ID where the property will be changed.
dotSize	Marker size.
dotColor	Marker color.
dotType	Sets if the marker will be a circular dot or a square dot.

Returns: TRUE if the operation was executed without errors.

- **ChartElementSetDot.** This function allows to define individual data markers for one of the elements of the data set or line type layer in two dimension's charts.

These markers can be defined in two dimension's charts of line type.

Markers assigned to this function precedence over those defined in ChartLayerSetDot function.

Syntax:

```
ChartElementSetDot (hWindow as integer, nchartlayer as integer,  
elementid as integer, dotSize as decimal, dotColor as integer, dotType  
as Integer) return boolean
```

Parameters:

hWindow	User's Control window handle.
nchartlayer	Chart's layer's ID where the property will be changed.
elementid	Element's Id of the element where the marker will be displayed. The first element is 0.
dotSize	Marker's size. If the size is 0 the marker will not be added.
dotColor	Dot's color in RGB format.
dotType	Dot type (circle or square).

Returns: TRUE if the operation was executed without errors.

- **ChartLayerSetCurvedLine.** This function sets if the line type layer will be displayed as straight lines or curved lines.

Syntax:

```
ChartLayerSetCurvedLine(hWindow as integer, nchartlayer as integer,  
iscurved as boolean) return boolean
```

Parameters:

hWindow	User's Control window handle.
nchartlayer	Chart's layer's ID where the property will be changed.
iscurved	Sets if the line will be curved or straight. The allowed values are: TRUE (curved line) and FALSE (straight line).

Returns: TRUE if the operation was executed without errors.

- **ChartLayerIsCurvedLine.** This function allows to query if the line is curved or straight.

Syntax:

```
ChartLayerIsCurvedLine(hWindow as integer, nchartlayer as integer) re-  
turn boolean
```

Parameters:

hWindow	User's Control window handle.
---------	-------------------------------

`nchartlayer` Chart's layer's ID where the property will be changed.

Returns: TRUE if the line is curved. FALSE if the line is straight.

- **ChartHideLineOnNullLabel.** This function allows to set if all the vertical lines in the back panel will be displayed or only all those where has an assigned label (function `ChartSetXAxysLabels`).

Syntax:

```
ChartHideLineOnNullLabel(hWindow as integer, hideLine as boolean) re-
turn boolean
```

Parámetros:

`hWindow` User's Control window handle.

`hideLine` This parameters sets if the vertical lines without label will be displayed. The allowed values are: TRUE and FALSE.

Returns: TRUE if the operation was executed without errors.

- **ChartIsHiddenLineOnNullLabel.** This function allows to query if all the vertical lines in the back panel have been displayed or only those that has an assigned label (function `ChartSetXAxysLabels`).

Syntax:

```
ChartIsHiddenLineOnNullLabel(hWindow as integer) return boolean
```

Parameters:

`hWindow` User's Control window handle.

Returns: TRUE if the lines without label are hidden and FALSE if all the lines have been displayed.

- **ChartSetMultiBarType.** This function allows to set the display type of the different bar type layers.

Syntax:

```
ChartSetMultiBarType (hWindow as integer, multiBarType as smallint)
return boolean
```

Parameters:

`hWindow` User's Control window handle.

`multiBarType` Allowed values:

1. Grouped bars. This is the default display type. The different layers or data sets of bar type are displayed next to each other.

2. Stacked bars. The different layers or data sets of bar type are displayed on top of the other. Show the relationship of the individual elements with the whole.
3. Percentage bars. The bars are displayed as stack, but the length of every layer is proportional to the amount that is, the total being a constant value of 100%.

Returns: TRUE if the operation was executed without errors.

- **ChartLayerShiftToSecondaryYAxis.** This function display a data layer to a different scale to the major scale. The labels of the layers assigned to this second scale is displayed on the secondary vertical axis on the left side of the graph.

Syntax:

```
ChartLayerShiftToSecondaryYAxis (hWindow as integer, nchartlayer as integer, shift as boolean) return boolean
```

Parameters:

hWindow	User's Control window handle.
nchartlayer	Chart's layer's ID where the property will be changed.
shift	TRUE: The data layer will be displayed regarding the secondary scale and value labels will be displayed on the secondary axis. FALSE: The data layer will be displayed regarding the main scale and the values labels will be displayed on the main axis. This is the default value.

Returns: TRUE if the operation was executed without errors.

- **ChartSetShortYAxisLabel.** This function indicates whether the labels on the y-axis values appear abbreviated or complete.

Syntax:

```
ChartSetShortYAxisLabel(hWindow as integer, mainYAxis as boolean, doSet as boolean) return boolean
```

Parameters:

hWindow	User's Control window handle.
mainYAxis	Sets if the property will be set to the main axis or the secondary axis. If the value is TRUE the property will be set in the main axis, if the value is FALSE the property will be set in the secondary axis.
doSet	This parameter indicates if the label will be abbreviated or not. The allowed values are: TRUE, the label will be displayed abbreviated, and FALSE, the label will be displayed complete.

The abbreviations are: 1M for values over 1 million and 1K and 999K for values between 1.000 and 999.000.

Returns: TRUE if the operation was executed without errors.

- **ChartSetTitleScale.** This function allows to change the size of the title's text.

Syntax:

```
ChartSetTitleScale(hWindow as integer, scale as smallint) return boolean
```

Parameters:

hWindow	User's Control window handle.
scale	Scale of the text. The scale is inversely proportional to the size, that is, if you want to decrease the size, the value of this parameter should be increased. The default value is 2.

Returns: TRUE if the operation was executed without errors.

- **ChartSetYAxisColorLabel.** This function allows to change the labels's text color in the Y axis.

Syntax:

```
ChartSetYAxisColorLabel(hWindow as integer, primaryYAxis as boolean, textColor as integer) return boolean
```

Parameters:

hWindow	User's Control window handle.
primaryYAxis	If the value is TRUE, the property will be set to the labels of the main Y axis. If the value is FALSE, the property will be set to the labels of the secondary Y axis.
textColor	Color in RGB format.

Returns: TRUE if the operation was executed without errors.

6. Bug Fixes

- Cosrun. The method ShowListAsMultiColumnGroup not initialized values of parameters received.
- Cosrun. Sometimes when you run the command EditNew there was a general protection fault and abort the execution of the application.
- Cosrun. If a variable is defined in a report as a variable without decimal mask, when it acquires value, it had no decimal and not exported the data.
- Cosrun. The headers created with the GroupListColumns method are not exported.
- Cosrun. The runtime does not correctly draw a TabControl on which had executed the method SetTabVerticalPages when the Form object was invoked for the second time.
- Cosrun. A List Box control without Vertical Scroll property allowed to use the mouse wheel.
- Cosrun. When a List Box control had not assigned Vertical Scroll property but had been assigned a skin, when running the Form Object there was a general protection fault and aborts execution.
- Cosrun. In the export of "only data" in some reports with the PLAINDETAIL variable specified, the runtime did not properly assigns some of the labels to the columns in the spreadsheet.
- Cosrun. If the method LosingFocus was executed after executing the ShowListFilterBar method there was a general protection fault.
- Cosrun. The method EditListColumn not change the current row and column.
- Cosrun. The totals of the aggregate functions that are calculated with the method TotalizeColumn and that had not value are unaligned. Now it has added the character "-".
- Cosrun. If the method TotalizeColumn not indicate all aggregate functions shows blank lines.
- Cosrun. The method EditListColumn of the SimpleControl's class allows to edit an aggregate node in a tree list control.
- Ttchartdll. Numbers not properly displayed on the ordinate axis.
- Ttchartdll. Area type charts are not drawn correctly if there were two consecutive elements with the same value.
- Ttchartdll. In xy type graphs not showing horizontal lines of the plot area when all values are positive.
- Ttchartdll. If NewChartElement function received a null value when creating the element, there was a general protection fault.

- Ttchartdll. In 3D line type, sometimes the labels were covered by the line.
- Prnpag32.dll. In versions of Windows newest than XP only printed one copy of the report although the user selects more copies.

Annex. Cosmos runtime in Tablets

From this version it is possible to scroll on touch screens List Box controls, Grid, Multi Box and Edit Field.

We have implemented this possibility in the preview screen (Preview) and code editor of the development environment.

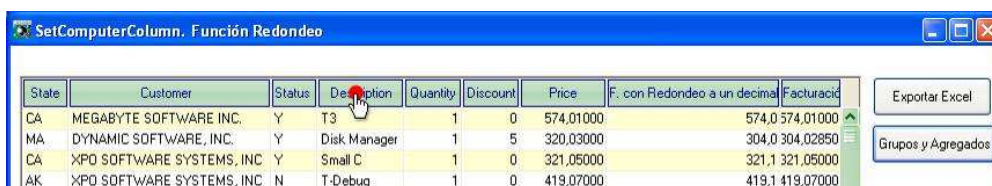
As mentioned above, when an edit field receives focus automatically displays an on-screen virtual keyboard.

To prevent to display onscreen virtual keyboard on personal computers¹ will need to be added to the project environment file or the cosmos.ini file the AUTOSHOWONSCREENKEYBOARD environment variable to FALSE (see section Environment Variables in this document).

Add columns to the grouping and sorting bar

In grouping methods (ShowMultiColumnGroupDlg) and sorting (ShowMultiColumnSortDlg) to add the desired column to the bar at the bottom of the screen will proceed as follows:

1º. Press on the title of the column you wish to add to the bar sorting and grouping:

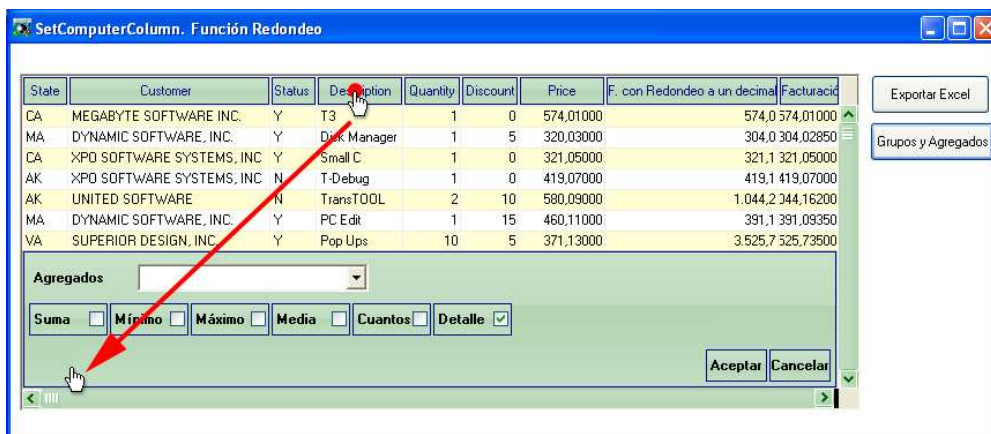


2º. Holding down, gently slide to the right or left to confirm the selection of the column:



3º. Then move the column heading, dragging it to the grouping and sorting bar and stop pushing:

¹ This could occur if the computer on which it is running Cosmos certain devices are installed (such as a digitizing tablet, a digital signature's reader, etc..).



4º. As a result, the selected column is displayed in the grouping and sorting bar:



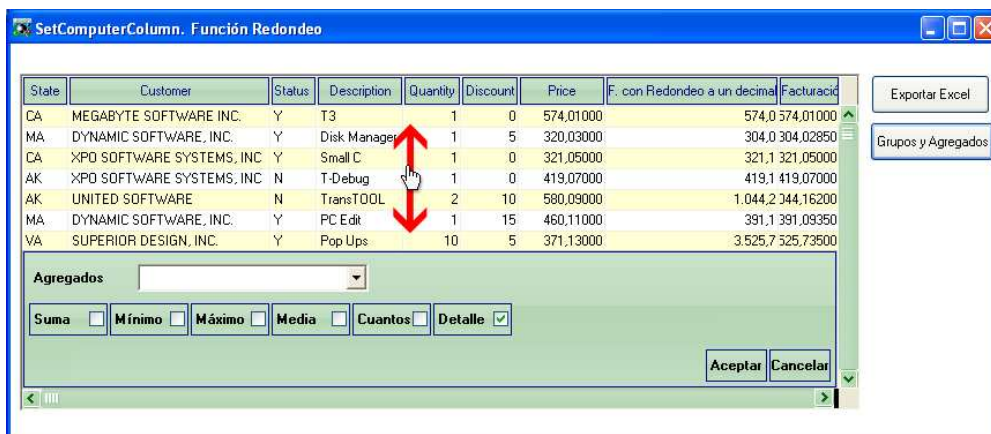
Remove columns from the grouping and sorting bar

- 1º. To delete a column from the grouping and sorting bar, will press on the column to remove from the bar and we will move slightly to the left or right.
- 2º. Then drag the selected column outside of the area occupied by the bar.
- 3º. The selected column will have disappeared from the grouping and sorting bar.

Horizontal scroll in List Box controls

To make the horizontal scroll in List Box controls will proceed according to the following steps:

- 1º. Select the list you want to perform on the scroll and pressing softly on it, move slightly your finger up or down:



2º. After this operation it will be possible to make the horizontal scroll by moving your finger across the screen to right or left:



Zoom

The reports preview screen allows performing zoom on it, either to enlarge or reduce.

To make the zoom you will need to place two fingers on the screen (usually thumb and index fingers) and bring them together to reduce or remove them to expand:

