

## Chart FX properties, events and methods

### Properties

[About](#)

[Adm](#)

[Angles3D](#)

[AutoIncrement](#)

[BarHorzGap](#)

[BGap](#)

[BkColor](#)

[Chart3D](#)

[ChartStatus](#)

[ChartType](#)

[Color](#)

[Const](#)

[ConstType](#)

[CustTool](#)

[Decimals](#)

[DecimalsNum](#)

[Edit](#)

[FixedBkColor](#)

[FixedColor](#)

[FixedGap](#)

[FixedStyle](#)

[FixedWidth](#)

[FixLeg](#)

[Fonts](#)

[Grid](#)

[hCtlWnd](#)

[hFont](#)

[hText](#)

[IniValue](#)

[ItemColor](#)

[ItemStyle](#)

[ItemWidth](#)

[KeyLeg](#)

[KeySer](#)

[LeftGap](#)

[Legend](#)

[LegendWidth](#)

[LineBkColor](#)

[LineColor](#)  
[LineStyle](#)  
[LineWidth](#)  
[NSeries](#)  
[NValues](#)  
[PaletteBar](#)  
[Pattern](#)  
[PatternBar](#)  
[PixFactor](#)  
[PointType](#)  
[ReturnValue](#)  
[RGap](#)  
[RGB2DBk](#)  
[RGB3DBK](#)  
[RGBBarHorz](#)  
[RGBBk](#)  
[RGBFont](#)  
[Scheme](#)  
[SerLeg](#)  
[ShowDialog](#)  
[ShowStatus](#)  
[Stacked](#)  
[StatusText](#)  
[Style](#)  
[ThisBkColor](#)  
[ThisColor](#)  
[ThisPoint](#)  
[ThisSerie](#)  
[ThisValue](#)  
[Title](#)  
[ToolBar](#)  
[TopGap](#)  
[Type](#)  
[Value](#)  
[VertGridGap](#)  
[View3D](#)  
[ViewRot3D](#)  
[WallWidth](#)  
[XLegType](#)  
[XValue](#)

[YLeg](#)

## **Events**

[ChangeColor](#)

[ChangeFont](#)

[ChangePalette](#)

[ChangePattern](#)

[ChangePattPal](#)

[ChangeString](#)

[ChangeType](#)

[ChangeValue](#)

[Destroy](#)

[GetLegend](#)

[GotFocus](#)

[LButtonDbIClk](#)

[LostFocus](#)

[Menu](#)

[RButtonDown](#)

[ReadFile](#)

[ReadTemplate](#)

[UserScroll](#)

## **Methods**

[CloseData](#)

[CopyBitmap](#)

[CopyData](#)

[DbIClk](#)

[ExportFile](#)

[ImportFile](#)

[Language](#)

[OpenDataEx](#)

[Paint](#)

[PrintIt](#)

[ReadTemplate](#)

[RigClk](#)

[Scroll](#)

[SetStatusItem](#)

[SetStripe](#)

[WriteTemplate](#)

---

ChartFX is a trademark of Software FX, Inc.

Visual dBASE is a registered Trademark of Borland International.

Other products are trademarks or registered trademarks of their respective companies.

## Constants and structures

Constant/structure	Used in
<a href="#">Chart Types</a>	<a href="#">Type Property</a>
<a href="#">Chart Status Constants</a>	<a href="#">ChartStatus Property</a>
<a href="#">Chart Styles</a>	<a href="#">Style Property</a>
<a href="#">chart_OpenData Constants</a>	Index in <a href="#">OpenDataEx Method</a>
<a href="#">Adm Constants</a>	Index in <a href="#">Adm Property</a>
<a href="#">Click Styles</a>	Index in <a href="#">DblClk</a> and <a href="#">RigClk</a> methods
<a href="#">Color Schemes</a>	<a href="#">Scheme Property</a>
<a href="#">Decimal Items</a>	Index in <a href="#">DecimalsNum Property</a>
<a href="#">Fonts List</a>	<a href="#">Font Property</a>
<a href="#">Font Types</a>	Index in <a href="#">Font Property</a>
<a href="#">Grid Styles</a>	<a href="#">Grid Property</a>
<a href="#">High-Low constants</a>	<a href="#">ThisSerie Property</a>
<a href="#">Line Styles</a>	<a href="#">LineStyle</a> and <a href="#">FixedStyle</a> Properties
<a href="#">Point Types</a>	<a href="#">PointType Property</a>
<a href="#">Stack Styles</a>	<a href="#">Stacked Property</a>
<a href="#">Status Items Styles</a>	<a href="#">Stacked Property</a>
<a href="#">Titles Styles</a>	dwStyle field in <a href="#">CHART_STITEM struct</a>
<a href="#">Tools Constants</a>	<a href="#">CustTool Property</a>

## About

### Description

This property displays the Chart FX version information dialog.

## Adm

### Description

A float property that sets administration values of the chart.

### Remarks

Available index values you can use with this property:

Index	What the value represents
<b>CSA_MIN</b>	Minimum value used in Y-axis
<b>CSA_MAX</b>	Maximum value used in Y-axis
<b>CSA_GAP</b>	Gap used in Y-axis. This value is calculated automatically each time you create a chart. If you want your Y-axis to put thick at a set interval, you would specify it with this index.
<b>CSA_SCALE</b>	Used scale. This constant is very useful when the values used in Y-axis are too big (i.e. 10.000.000,00) in this case you can use a 1.000.000 scale and the values on the Y-axis will be divided by this scale.
<b>CSA_YLEGGAP</b>	Equivalent units of a y legend text.
<b>CSA_PIXXVALUE</b>	Equivalent unit representation of Y-axis in pixels.
<b>CSA_XMIN</b>	Minimum value used in X-axis (Scatter)
<b>CSA_XMAX</b>	Maximum value used in X-axis (Scatter)
<b>CSA_XGAP</b>	Gap used in X-axis (Scatter)

### Data Type

Float (single)

## Angles3D

### Description

A Long value that sets or returns the angles used to draw the chart in 3DView mode. The X-angle is in the low word and the Y-angle is in the high word. The default value is 0,0.

### Remarks

This property is only used when drawing the chart in 3DView mode (when [View3D Property](#) is set to TRUE).

This function is provided to the end user through the rotation dialog.

At design time the programmer can use the [ViewRot3D Property](#) to set the initial values for these angles.

### Data Type

Long

## AutoIncrement

### Description

A boolean property that sets the autoincrement mode, when this value is TRUE, the library will increment [ThisPoint](#) and [ThisSerie](#) each time you set one of the following properties: [ThisBkColor](#), [ThisColor](#), [ThisValue](#)

### Remarks

This property resets the ThisPoint and ThisSerie properties to 0 each time the maximum value is reached.

The library will try to increment ThisSerie first, if it is not possible (the maximum value was reached) it will increment ThisPoint (only if needed).

### Data Type

Boolean



## **BarHorzGap**

### **Description**

An integer value (16 bits) that sets or returns the width or gap of the legend (Text) for a horizontal bar chart. The default value is 50.

### **Remarks**

This property affects only horizontal bar charts and is measured in device units (Pixels).

### **Data Type**

Integer

## BottomGap

### Description

An integer value (16 bits) that sets or returns the gap between the bottom border of the chart and the border of the Chart FX Component. This gap is used to draw titles. The default value is 40.

### Remarks

This value is measured in device units (Pixels). The end user can modify this distance manually from the chart window by pointing the mouse near each border and dragging it to the desired position. to stop the user from changing this distance please refer to [Style Property](#) with **CS\_RESIZEABLE** code.

### Data Type

Integer

## **BkColor**

### **Description**

A color (Long) value that sets or returns the background color that will be used to paint the markers corresponding to the series supplied as the index of the property.

### **Remarks**

Before using this property the **COD\_COLORS** communication channel must be opened with the [OpenDataEx Method](#).

### **Data Type**

Color (Long)

## **Chart3D**

### **Description**

A boolean (Integer) value (16 bits) that sets or returns TRUE if the chart is in 3D and FALSE if it is in 2D.

### **Remarks**

This function is provided to the end user through the menu.

### **Data Type**

Integer

## ChartStatus

### Description

This property returns a long value that specifies what kind of changes the user has made.

### Remarks

See [Chart Status Constants](#) for returned values.

### Data Type

Long

## ChartType

### Description

An Integer value (16 bits) that sets or returns the gallery type of the chart. (BAR, AREA, etc.). This property is a subset of the [Type Property](#).

### Remarks

This function is provided to the end user through the menu.

## CloseData

### Description

By calling this method you close the communications channel opened with the **OpenDataEx Method**, It's extremely important that you close all the opened channels.

The value assigned to this property is not used (must be set to zero).

### Remarks

See [OpenDataEx Method](#) explanation.

## Color

### Description

A color (Long) value that sets or returns the color that will be used to paint the markers corresponding to the series supplied as the index of the property.

### Remarks

Before using this property the **COD\_COLORS** communication channel must be opened with the [OpenDataEx Method](#).

### Data Type

Color (Long)



## Const

### Description

A float property that sets or returns the value of a constant in the chart, the number of the constant to set (get) is represented by the index supplied.

### Remarks

Before using this property the **COD\_CONSTANTS** communication channel must be opened with the [OpenDataEx Method](#).

### Data Type

Float (Single)

## ConstType

### Description

An Integer value (16 bits) that sets or returns the type of the constant values and legends. This property affects how these elements are presented in a chart.

### Remarks

This property affect constants setted with the Const Property.

The value must be a combination of the following flags:

**CC\_HIDETEXT**                      Hide the constants text.

**CC\_HIDE**                              Hide the constants.

### Data Type

Integer

## **CopyBitmap**

### **Description**

Allows the programmer to copy the chart to the clipboard (As a bitmap).

The value assigned to this property is not used (must be set to zero).

### **Remarks**

This function is provided to the end user through the menu if permitted by the programmer by using the **CS\_COPY** style.

## **CopyData**

### **Description**

Allows the programmer to copy the charted data to the clipboard.

The value assigned to this property is not used (must be set to zero).

### **Remarks**

This function is provided to the end user through the menu if permitted by the programmer by using the **CS\_COPY** style.

## **CustTool**

### **Description**

A Long value (32 bits) that sets or returns the visible buttons in the ToolBar.

### **Remarks**

Setting of this property must contain a Bitwise **OR** of [Tool Constants](#)

### **Data Type**

Long

## DbIClk

### Description

S the response of the library when the user makes a double click on a marker. The index will specify the type of response and the value must always be zero (0) except in the case of **CHART\_MENUCLK** index where the value is the handle of the popup menu. The default value is **CHART\_BALLOONCLK**.

### Remarks

See [Click Styles](#) for index supported values. Note that the library will **always** generate the DbIClk event.

***If you pass a menu to the library you must remember to destroy it when you are finished with it.***

## **Decimals**

### **Description**

An integer value (16 bits) that sets or returns the number of decimals used to show numbers in the chart. The default value is 2.

### **Remarks**

This function is provided to the end user through the options dialog.

### **Data Type**

Integer

## DecimalsNum

### Description

An integer value (16 bits) that sets or returns the number of decimals used to show specific elements in the chart. The index specifies the item to change the number of decimals.

### Remarks

See [Decimal Items](#) for index supported values. This function is provided to the end user through the options dialog.

### Data Type

Integer



## Edit

### Description

This is a read-only property that holds a window handle when the user is trying to change a legend or value in the DataEditor.

### Remarks

This property must only be used in response of a notification of the [ChangeString Event](#) or the [ChangeValue Event](#).

### Data Type

Integer

## ExportFile

### Description

Allows the programmer to create a file with all the attributes and data of the current chart.

### Remarks

**sFileName** must be a valid file name including path. This function is provided to the end user through the menu if permitted by the programmer by using the **CS\_EXPORTFILE** style.

The file containing the saved chart is proprietary and you can only display it by using the [ImportFile Method](#) or from the chart menu.

## **FixedBkColor**

### **Description**

A Color (Long) value that sets or returns the background color used to paint the constant lines. The default value is white (RGB(255,255,255))

### **Remarks**

This property is used when [FixedStyle Property](#) is not equal to **CHART\_SOLID**.

### **Data Type**

Long

## **FixedColor**

### **Description**

A Color (Long) value that sets or returns the foreground color used to paint the constant lines. The default value is red (RGB(255,0,0))

### **Data Type**

Long

## **FixedGap**

### **Description**

An Integer value (16 bits) that sets or returns the minimum width of each unit in the X-axis where 0 means a default value chosen by the library. The default value is 0.

### **Remarks**

This property is measured in device units (Pixels).

### **Data Type**

Integer

## FixedStyle

### Description

An Integer value (16 bits) that sets or returns the style used to paint the constants lines. The default value is **CHART\_DOT**.

### Remarks

See [Line Styles](#) for supported values.

### Data Type

Integer

## **FixedWidth**

### **Description**

An Integer value (16 bits) that sets or returns the width used to paint the constants lines. The default value is 1.

### **Data Type**

Integer

## FixLeg

### Description

A string value that sets or returns the text of the Constant lines, the index supplied specifies the position of the legend.

### Remarks

Normally you will supply as many legends as the number of constants of the chart. This is the same number that you supplied for the [OpenDataEx Method](#) with the **COD\_CONSTANTS** code.

See also [Const Property](#).

### Data Type

String



## Font

### Description

A Long value (32 bits) that sets the font used to draw different texts in a chart. The index of this property represents the text to change and the value represents the new font. The value must always be a combination (bitwise OR) of Font Styles.

### Remarks

This function is provided to the end user through the menu. See [Font Types](#) for index supported values. See [Font Styles](#) for property supported values.

***You must take care of giving a TRUE-TYPE font when setting the left or right title.***

### Data Type

Long

## Grid

### Description

An integer value (16 bits) that sets or returns the type of grid. The default value is **CHART\_NONEGRID**.

### Remarks

This property does not affect Pie Charts. This function is provided to end user through the menu.

See [Grid Styles](#) for supported values.

### Data Type

Integer

## **hCtlWnd**

### **Description**

This property returns the Window handle of the control, this value is exactly the same as the **hWnd** property and is provided for Gupta SQLWindows 4.1 compatibility.

### **Data Type**

Window handle (16 bits)

## hFont

### Description

An Integer value (16 bits) that sets the font used to draw different text elements in a chart. The index of this property represents the text to change and the value represents the new font. The value must always be a valid font handle (HFONT)

### Remarks

This function is provided to the end user through the menu. See [Font Types](#) for index supported values. You must take care of destroying the font passed to this property.

### Data Type

Integer

## HText

### Description

A string value that sets or returns the text that Chart FX will use in the dialog or balloon generated by the last double click or right click of the mouse. For example, when the user double clicks with the left mouse button a point in any of the series of the chart a default text containing the Series Name-Legend-point value is automatically generated, if you want to modify this default text, use this property in conjunction with the [DblClk](#) or [RigClk](#) events.

### Data Type

String

## ImportFile

### Description

Allows the programmer to read a previously saved file with all the attributes and data into the current chart, this file must be created with the **ExportFile Method** or through the menu.

### Remarks

**sFileName** must be a valid file name including path. This function is provided to the end user through the menu if permitted by the programmer by using **CS\_IMPORTFILE** style.

## IniValue

### Description

A float property that sets or returns the initial value of a point in the chart, the point to set/get is represented by the index and the series is given by the current value of the [ThisSerie Property](#).

### Remarks

Before using this property the **COD\_INIVALUES** communication channel must be opened with the [OpenDataEx Method](#). The **IniValue** property is supported by bar charts only.

### Data Type

Float (Single)

## **ItemColor**

### **Description**

A Color (Long) value that sets or returns the foreground color used to paint the item specified in the index.

### **Data Type**

Long



## ItemStyle

### Description

An integer value that sets or returns the style used to paint the item specified in the index.

### Remarks

See [Line Styles](#) for supported values.

### Data Type

Integer

## **ItemWidth**

### **Description**

An integer value that sets or returns the width used to paint the item specified in the index.

### **Data Type**

Integer

## **KeyLeg**

### **Description**

A string value that sets or returns the text of the X Legend Keys, the index supplied specifies the position of the legend.

### **Remarks**

Normally you will supply as many legends as the number of points of the chart. The library will always draw this Key Legends assuming they are short enough.

### **Data Type**

String

## **KeySer**

### **Description**

A string value that sets or returns the text of the Series Legend Keys, the index supplied specifies the position of the legend.

### **Remarks**

Normally you will supply as many legends as the number of series of the chart. The library will always draw this Key Legends assuming they are short enough.

This property is supported by horizontal bar charts only.

### **Data Type**

String

## Language

### Description

This method is used to change the current language used by the library. This language is represented by a resource DLL that holds the dialogs, strings and menus needed by Chart FX.

### Remarks

**sLangDLL** is the name of the resource DLL (including path). To support other languages, the package provides the necessary files (RC, DEF, DLG, C) to translate the information and build it to obtain a DLL (Dynamic Link Library) which is the kind of file that this message handles.

For more information on this topic please refer to **International Support** on previous sections of this manual

Loading this DLL follows the Windows standard search for this type of file, therefore you should place the new DLL containing the language support, in your PATH, WINDOWS or SYSTEM directory.

## **Legend**

### **Description**

A string value that sets or returns the text of the X Legend, the index supplied specifies the position of the legend.

### **Remarks**

Normally you will supply as many legends as the number of points of the chart.

### **Data Type**

String

## LegendWidth

### Description

An integer value (16 bits) that sets or returns the width of the legend window. The default value is 100.

### Remarks

This value is measured in device units (Pixels). The end user can modify this distance manually from the chart window by pointing the mouse near the border of the legend and dragging it to the desired position. This property has no effect if the legend window is hidden.

### Data Type

Integer

DesignTime



## LeftGap

### Description

An integer value (16 bits) that sets or returns the gap between the left border of the chart and the left border of the Chart FX Component. This gap is used to draw titles. The default value is 40.

### Remarks

This value is measured in device units (Pixels). The end user can modify this distance manually from the chart window by pointing the mouse near each border and dragging it to the desired position. to stop the user from changing this distance please refer to [Style Property](#) with **CS\_RESIZEABLE** code.

### Data Type

Integer



## LineBkColor

### Description

A Color (Long) value that sets or returns the background color used to paint the lines in a 2D line chart. The default value is white (RGB(255,255,255))

### Remarks

This property is used when [LineStyle Property](#) is not equal to **CHART\_SOLID**.

### Data Type

Long

## **LineColor**

### **Description**

A Color (Long) value that sets or returns the foreground color used to paint the lines in a 2D line chart. The default value is black (RGB(0,0,0))

### **Data Type**

Long

## LineStyle

### Description

An Integer value (16 bits) that sets or returns the style used to paint the lines in a 2D line chart.

### Remarks

See [Line Styles](#) for supported values.

### Data Type

Integer

## **LineWidth**

### **Description**

An Integer value (16 bits) that sets or returns the width used to paint the lines in a 2D line chart. The default value is 1.

### **Data Type**

Integer

## **NSeries**

### **Description**

An integer value (16 bits) that sets the number of series the chart will have. The default value is 2.

### **Remarks**

At runtime the programmer must use the [OpenDataEx Method](#) to set the number of series and points.

### **Data Type**

Integer

## **NValues**

### **Description**

An integer value (16 bits) that sets the number of points the chart will have. The default value is 4.

### **Remarks**

At runtime the programmer must use the [OpenDataEx Method](#) to set the number of series and points.

### **Data Type**

Integer

## OpenDataEx

### Description

Opens a communication channel to send data to the chart object, the index represents the type of channel to be opened and the value represents the number of items.

### Remarks

Once the data is filled, you must use **CloseData Method** in order to close the communication channel. The indexes that can be used with this property and the meaning of these settings are:

<b>nChannel</b>	<b>What the value represents</b>	<b>Related Property</b>
<b>COD_VALUES</b>	MAKELONG(nSeries,nPoints)	<a href="#"><u>Value</u></a>
<b>COD_CONSTANTS</b>	nConstants	<a href="#"><u>Const</u></a>
<b>COD_COLORS</b>	nColors	<a href="#"><u>Color</u></a>
<b>COD_STRIPES</b>	nStripes	<a href="#"><u>chart_SetStripe</u></a> function
<b>COD_INIVALUES</b>	MAKELONG(nSeries,nPoints)	<a href="#"><u>XValue</u></a>
<b>COD_XVALUES</b>	MAKELONG(nSeries,nPoints)	<a href="#"><u>IniValue</u></a>
<b>COD_STATUSITEMS</b>	nStatusItems	<a href="#"><u>chart_SetStatusItem</u></a> function

## **PaletteBar**

### **Description**

A boolean (Integer) value (16 bits) that determines whether the PaletteBar is visible or hidden.

### **Remarks**

This function is provided to the end user through the menu.

### **Data Type**

Integer



## Pattern

### Description

An integer value (16 bits) that sets the pattern used to paint the serie supplied as an index of the property. This value must be less than 16 and represents the index of the pattern in the current PatternBar.

### Remarks

This property affects charts with pattern schemes only (See [Scheme Property](#))

### Data Type

Integer

## **PatternBar**

### **Description**

A boolean (Integer) value (16 bits) that determines whether the PatternBar is visible or hidden.

### **Remarks**

This function is provided to the end user through the menu.

### **Data Type**

Integer

## PixFactor

### Description

An Integer value (16 bits) that sets or returns the equivalent unit representation of Y-axis in pixels. This means, that you can change the factor in pixels for each unit in the Y-axis, to accomplish vertical scroll bars in the chart window. The default value is 0.

### Remarks

This property is used in conjunction with the [Adm Property](#) (**CSA\_PIXXVALUE** code) to modify the pixel factor of the unit in Y-axis.

A value of zero (0) means the library will choose an appropriate PixFactor for the chart to fit the current height.

### Data Type

Integer

## PointType

### Description

An integer value (16 bits) that sets or returns the type of point used to paint markers in line, mark, spline and scatter charts. The default value is **CHART\_RECTMK**.

### Remarks

This function is provided to the end user through the options dialog.

See [Point Types](#) for supported values.

### Data Type

Integer

## **PrintIt Method**

### **Description**

This method allows the programmer to print the chart.

The value assigned to this property is not used (must be set to zero).

### **Remarks**

This function is provided to the end user through the menu if permitted by the programmer by using the **CS\_PRINTABLE** style.

## ReadTemplate

### Description

Allows the programmer to read a previously saved file with all the visual attributes (Colors, borders, etc.) into the current chart (without including data).

### Remarks

**sFileName** must be a valid file name including path. This function is provided to the end user through the menu if permitted by the programmer by using the **CS\_TEMPLATE** style.

## **ReturnValue**

### **Description**

This property sets the value that will be returned in the current event notification. ReturnValue is provided for Gupta SQLWindows 4.1 compatibility.

### **Remarks**

See [Returning values](#) for a detailed description of how to return values in other development tools.

### **Data Type**

Integer

## RightGap

### Description

An integer value (16 bits) that sets or returns the gap between the right border of the chart and the border of the Chart FX Component. This gap is used to draw titles. The default value is 40.

### Remarks

This value is measured in device units (Pixels). The end user can modify this distance manually from the chart window by pointing the mouse near each border and dragging it to the desired position. to stop the user from changing this distance please refer to [Style Property](#) with **CS\_RESIZEABLE** code.

### Data Type

Integer



## **RGB2DBk**

### **Description**

A Color (Long) value that sets or returns the color for the 2D charts background. The default value is light gray (RGB(192,192,192)).

### **Remarks**

This function is provided to the end user by allowing him to drag & drop a color from the PaletteBar to any part of this background.

### **Data Type**

Long

## **RGB3DBK**

### **Description**

A Color (Long) value that sets or returns the color for 3D charts background. The default value is white (RGB(255,255,255)).

### **Remarks**

This function is provided to the end user by allowing him to drag & drop a color from the PaletteBar to any part of this background. This property is used by 3D charts only.

### **Data Type**

Long

## **RGBBarHorz**

### **Description**

A Color (Long) value that sets or returns the color of the X legend background of a horizontal Bar Chart. The default value is cyan (RGB(0,255,255))

### **Remarks**

This function is provided to the end user by allowing him to drag & drop a color from the PaletteBar to any part of the area of this Legend.

### **Data Type**

Long

## **RGBBk**

### **Description**

A Color (Long) value that sets or returns the color for the background surrounding the chart. The default value is light gray (RGB(192,192,192))

### **Remarks**

This function is provided to the end user by allowing him to drag & drop a color from the PaletteBar to any part of this background.

### **Data Type**

Long

## **RGBFont**

### **Description**

A Color (Long) value that sets the color of the font used to draw different text elements in a chart. The index of this property represents the text to change and the value represents the new color.

### **Remarks**

This function is provided to the end user through the menu. See [Font Types](#) for index supported values.

At design time the user can change the colors through the Chart FX Component properties.

### **Data Type**

Long

## RigClk

### Description

An Integer value (16 bits) that sets the response of the library when the user makes a right click on a marker. The index will specify the type of response and the value must always be zero (0) except in the case of **CHART\_MENUCLK** index where the value is the handle of the popup menu. The default value is **CHART\_BALLOONCLK**.

### Remarks

See [Click Styles](#) for index supported values. Note that the library will **always** generate the RigClk event.

***If you pass a menu to the library you must remember to destroy it when you are finished with it.***

## Scheme

### Description

An integer value (16 bits) that sets or returns the scheme used to paint the markers. The default value is **CHART\_CSSOLID**.

### Remarks

This function is provided to end user through the options dialog.

See [Color Schemes](#) for supported values.

### Data Type

Integer

## Scroll

### Description

This method is used to modify the scroll position of the chart. The index specifies the scroll message (see wParam in WM\_HSCROLL) and the value represents the scroll additional information (see lParam in WM\_HSCROLL).

### Remarks

You can use this property to "synchronize" two charts in conjunction with the **UserScroll** Event



## **SerLeg**

### **Description**

A string value that sets or returns the text of the Series, the index supplied specifies the position of the legend.

### **Remarks**

Normally you will supply as many legends as the number of series of the chart.

### **Data Type**

String

## ShowDialog

### Description

This property is used to show any of the user interface dialogs provided by the library. The index value specifies the dialog to show.

### Remarks

nDialog	IExtra	Meaning
CDIALOG_EXPORTFILE	Not Used (*)	Export File Dialog
CDIALOG_IMPORTFILE	Not Used (*)	Import File Dialog
CDIALOG_WRITETEMPLATE	Not Used (*)	Write Template Dialog
CDIALOG_READTEMPLATE	Not Used (*)	Read Template Dialog
CDIALOG_PAGESETUP	Not Used (*)	Page Setup Dialog
CDIALOG_ABOUT	Not Used (*)	About Dialog
CDIALOG_OPTIONS	Not Used (*)	Options Dialog
CDIALOG_EDITTTITLES	Not Used (*)	Edit Titles Dialog
CDIALOG_FONTS	<a href="#">Font Types</a>	Change Font Dialog
CDIALOG_ROTATE	Not Used (*)	Rotate Dialog

*\* Not used values must be set to zero.*

## ShowStatus

### Description

A boolean (Integer) value (16 bits) that determines whether the StatusBar is visible or hidden.

### Remarks

This function is provided to the end user through the menu if the statusBar has been previously created by the programmer. See also **chapter 5 of the programmer's guide**.

### Data Type

Integer

## Stacked

### Description

An integer value (16 bits) that sets or returns the type of stack used to draw area and bar charts. The default value is **CHART\_NOSTACKED**.

### Remarks

This property affects Bar and Area charts only. This function is provided to the end user through the options dialog.

See [Stack Styles](#) for supported values.

### Data Type

Integer

## StatusText

### Description

A string value that sets or returns the text of an existent status item. The index supplied represent the code (ID) of the item.

### Remarks

This property must be used only after you create the status bar with the [SetStatusItem Method](#)

### Data Type

String

## Style

### Description

A long value (32 bits) that sets or returns the style of the chart. This style refers to what the end user can do in the chart window, thus permitting to change from one type of chart to another, modify 3D View, Rotation, etc.

### Remarks

Setting of this property must contain a bitwise **OR** of [Chart Styles constants](#)

### Data Type

Long

## TopGap

### Description

An integer value (16 bits) that sets or returns the gap between the top border of the chart and the border of the Chart FX Component. This gap is used to draw titles. The default value is 40.

### Remarks

This value is measured in device units (Pixels). The end user can modify this distance manually from the chart window by pointing the mouse near each border and dragging it to the desired position. to stop the user from changing this distance please refer to [Style Property](#) with **CS\_RESIZEABLE** code.

### Data Type

Integer

## **ThisBkColor**

### **Description**

A color (Long) value that sets the background color that will be used to paint the markers corresponding to the series indicated by the [ThisSerie Property](#).

### **Remarks**

When "Each Bar" Type is set, this property uses the point indicated by the [ThisPoint Property](#).

At runtime you can modify (get) the colors by using the [BkColor Property](#)

### **Data Type**

Color (Long)



## **ThisColor**

### **Description**

A color (Long) value that sets the color that will be used to paint the markers corresponding to the series indicated by the [ThisSerie Property](#).

### **Remarks**

When "Each Bar" Type is set, this property uses the point indicated by the [ThisPoint Property](#).

At runtime you can modify (get) the colors by using the [Color Property](#)

### **Data Type**

Color (Long)

## **ThisPoint**

### **Description**

An Integer value (16 bits) that sets or returns the actual point.

### **Remarks**

The properties that use **ThisPoint** are: [ThisColor](#), [ThisBkColor](#).

### **Data Type**

Integer

## **ThisSerie**

### **Description**

An Integer value (16 bits) that sets or returns the actual serie.

### **Remarks**

The properties that use **ThisSerie** are: [Value](#), [XValue](#), [IniValue](#), [ThisColor](#), [ThisBkColor](#).

For High-Low-Close and Open-High-Low-Close charts see [HLC and OHCL constants](#).

### **Data Type**

Integer

## **ThisValue**

### **Description**

A float property (Single Type in Borland Delphi) that sets the value of a point in the chart, the point to set (get) is represented by the [ThisPoint Property](#) and the serie is given by the [ThisSerie Property](#).

### **Remarks**

Before using this property the **COD\_VALUES** communication channel must be opened with the [OpenDataEx Method](#).

This property works in conjunction with the [AutoIncrement Property](#).

### **Data Type**

Float (Single)

## **Title**

### **Description**

A string value that sets or returns the titles of the chart. The type of title to set is specified through the index supplied.

### **Remarks**

This function is provided to the end user through the menu. See [Title Types](#) for index supported values.

### **Data Type**

String

## **ToolBar**

### **Description**

A boolean (Integer) value (16 bits) that determines whether the ToolBar is visible or hidden.

### **Remarks**

This function is provided to the end user through the menu.

### **Data Type**

Integer

## Type

### Description

A long value (32 bits) that sets or returns the type of the chart, this type includes gallery type as well as other visual elements in the chart window. The default value is **LINE | CT\_SHOWPOINTS**.

### Remarks

Setting of this property must contain a bitwise **OR** of [Chart Types constants](#)

### Data Type

Long

## Value

### Description

A float property (Single Type in Borland Delphi) that sets or returns the value of a point in the chart, the point to set (get) is represented by the index and the serie is given by the [ThisSerie Property](#).

### Remarks

Before using this property the **COD\_VALUES** communication channel must be opened with the [OpenDataEx Method](#).

### Data Type

Float (Single)



## **View3D**

### **Description**

A Boolean (Integer) value that sets or returns View mode of the chart in 3D. The default value is FALSE.

### **Remarks**

Removing the 3D View (FALSE) sets the axes at right angles independent of chart rotation. Setting the 3D View (TRUE) show axes in perspective.

This function is provided to the end user through the rotation dialog.

At design time the programmer must use the [ViewRot3D Property](#)

### **Data Type**

Integer

## **VertGridGap**

### **Description**

An Integer value (16 bits) that sets or returns the vertical grid gap. This is the distance (measured in X points) between 2 lines of the vertical grid

### **Remarks**

The value must be greater than 0

### **Data Type**

Integer

## ViewRot3D

### Description

This property shows a dialog that allows the programmer to set/reset the 3D View and the angles used to draw the chart.

### Remarks

At runtime the programmer must use the [View3D](#) and [Angles3D](#) properties

## **WallWidth**

### **Description**

An integer value (16 bits) that sets or returns the wall's width of a 3D Chart. The default value is 8.

### **Remarks**

This property does not affect 2D Charts and is measured in device units (Pixels).

### **Data Type**

Integer

## WriteTemplate

### Description

Allows the programmer to create a template file with all the visual attributes (Colors, borders, etc) of the current chart without including data.

### Remarks

**sFileName** must be a valid file name including path. This function is provided to the end user through the menu if permitted by the programmer by using the **CS\_TEMPLATE** style.

## XLegType

### Description

An Integer value (16 bits) that sets or returns the type of the X Legend text. This property affects how this legend is presented in a chart.

### Remarks

The value must be a combination of the following flags:

<b>CL_NOTCLIPPED</b>	Chart FX will not clip the X legends (Its programmers responsibility to assure that the legends dont overlap each other).
<b>CL_NOTCHANGECOLOR</b>	Chart FX will not change the color of the legends that dont fit in the available space (the default behavior is to draw that legends in RED)
<b>CL_HIDE</b>	Chart FX will not draw X Legend.
<b>CL_GETLEGEND</b>	Chart FX will send an event every time it needs to draw a legend in the Y axis. You can use the <b>HTEXT Property</b> to get and set the string to draw. This event ( <b>GetLegend</b> ) will also be sent for the X axis in a scatter chart.

### Data Type

Integer

## XValue

### Description

A float property that sets or returns the X-axis value of a point in the chart, the point to set (get) is represented by the index and the serie is given by the [ThisSerie Property](#).

### Remarks

Before using this property the **COD\_XVALUES** communication channel must be opened with the [OpenDataEx Method](#). This property is supported by scatter charts only.

### Data Type

Float (Single)

## YLeg

### Description

A string value that sets or returns the text of the Y Legend, the index supplied specifies the position of the legend.

### Remarks

This property must be used in conjunction with [Adm Property](#) with **CSA\_YLEGGAP** code. The value supplied with Adm property is the gap the library will use between the legends.

### Data Type

String



## SetStatusItem

**long SetStatusItem**(int *nIndex*, **BOOL** *bHaveText*, **UINT** *ID*, **BOOL** *bFramed*, **int** *nWidth*, **int** *nMin*, **int** *nDesp*, **DWORD** *dwStyle*)

**SetStatusItem** is a method that allows the programmer to pass items that will appear in the status bar (if created) of the window containing the chart. This function must be accessed after calling the **OpenDataEx** method with the COD\_STATUSITEMS constant.

Parameter	Name	Description
<b>int</b> <i>nIndex</i>	Index of the item	
<b>BOOL</b>	<i>bHaveText</i>	Specifies if that item will contain text.
<b>UINT</b>	<i>ID</i>	ID number of the item
<b>BOOL</b>	<i>bFramed</i>	Specifies if that item will have a frame surrounding it.
<b>int</b> <i>nWidth</i>	Specifies the width in device units (pixels)	
<b>int</b> <i>nMin</i>	Specifies the minimum tracking width of the item in device units (pixels), in case the user resize the window containing the chart.	
<b>int</b> <i>nDesp</i>	Distance from the last item in device units (pixels)	
<b>DWORD</b>	<i>dwStyle</i>	Specifies the style of the text in the item: <b>STITEM_SLEFT</b> = Left alignment <b>STITEM_SCENTER</b> = Centered <b>STITEM_SRIGHT</b> = Right alignment

### Comments

This function has no effect without a chart\_OpenData function call.

Although there is a way to create and change status bar items in the window containing the chart using chart\_Send function and messages, this function is provided to support compatibility with other development tools.

### See Also

[OpenDataEx Method](#).

## SetStripe

**long SetStripe(int *nIndex*, double *fBegin*, double *fEnd*, DWORD *dwColor*)**

**SetStripe** is a method that allows the programmer to pass two values in which a color frame is to be displayed in the background of the chart. This function is very useful when you want to denote a specific area in the chart. This function must be accessed after calling the `chart_OpenData` function with the `COD_STRIPES` constant.

Parameter	Name	Description
<b>int</b>	<i>nIndex</i>	Index of the stripe
<b>double</b>	<i>fBegin</i>	Numeric value corresponding to the beginning of the stripe
<b>double</b>	<i>fEnd</i>	Numeric value corresponding to the ending of the stripe
<b>DWORD</b>	<i>dwColor</i>	RGB color of the stripe

### Comments

This function has no effect without a `chart_OpenData` function call.

This function can apply to any type of charts (except pie charts).

### Sample

To set a blue stripe from 10 to 20 of a chart :

```
Chart1.OpenDataEx(COD_STRIPES,1,0);
```

```
Chart1.SetStripe(0,10,20,RGB(0,0,255));
```

```
Chart1.CloseData(COD_STRIPES);
```

### See Also

[OpenDataEx](#)

## Paint

**long Paint(HWND *hChart*, HDC *hDC*, int *nLeft*, int *nTop*, int *nRight*, int *nBottom*, BOOL *bPrint*, LONG *lReserved*)**

**Paint** is a method that allows the programmer to draw a chart in any device context. This function is very useful when you want to print charts and others objects in the same page or more than one chart in a page.

Parameter	Name	Description
<b>HDC</b>	<i>hDC</i>	Device context where the chart is going to be drawn.
<b>int</b> <i>nLeft</i>		x-coordinate of the upper-left corner of the bounding rectangle.
<b>int</b> <i>nTop</i>		y-coordinate of the upper-left corner of the bounding rectangle.
<b>int</b> <i>nRight</i>		x-coordinate of the bottom-right corner of the bounding rectangle.
<b>int</b> <i>nBottom</i>		y-coordinate of the bottom-right corner of the bounding rectangle.
<b>BOOL</b>	<i>bPrint</i>	Indicates whether the chart is going to be shown as in the printer or as in the screen (in the printer the background color is not shown).
<b>LONG</b>	<i>lReserved</i>	Reserved. Must be set to 0.

### Return Value

None.

### Comments

The bounding rectangle must be in device units.

### Sample

```
{ To print two charts into a 3000 x 3000 printer }  
Chart1.Paint(Printer.Handle, 0, 0, 1500, 3000, 1, 0);  
Chart1.Print(Printer.Handle, 1500, 0, 3000, 3000, 1, 0);
```

## ChangeColor

### Description

This event occurs when the user changed a color (using the drag&drop technique).

### Parameters

Name	Type	Description
nType	Integer	Color code (See remarks)
nIndex	Integer	Index of the changed color.

### Remarks

nType Parameter	Description and nIndex Parameter
CCC_SERIE	Color of a series has changed, nIndex represents the series
CCC_SERIEBK	Back Color of a series has changed, nIndex represents the series
CCC_ONE	Color of markers has changed, nIndex <b>Not used</b> .
CCC_ONEBK	Back Color of markers has changed, nIndex <b>Not used</b> .
CCC_BARHORZ	Color the horizontal bar text has changed, nIndex <b>Not used</b> .
CCC_BKGND	Background of the chart has changed, nIndex <b>Not used</b> .
CCC_2DBK	Background of the 2D chart has changed, nIndex <b>Not used</b> .
CCC_3DBK	Background of the 3D chart has changed, nIndex <b>Not used</b> .

### Return Value

The return value is not used (**Must NOT be set**)

## ChangeFont

### Description

This event occurs when the user changed a font.

### Parameters

Name	Type	Description
nIndex	Integer	Index of the changed font.

### Remarks

See [Font Types](#) for possible values of nIndex.

### Return Value

The return value is not used (***Must NOT be set***)

## ChangePalette

### Description

This event occurs when the user changed a color in the PaletteBar.

### Parameters

Name	Type	Description
nIndex	Integer	Index of the changed color.

### Return Value

The return value is not used (***Must NOT be set***)

## ChangePattern

### Description

This event occurs when the user changed a pattern (using Drag&Drop feature).

### Parameters

Name	Type	Description
nType	Integer	Pattern code (See remarks)
nIndex	Integer	Index of the changed pattern.

### Remarks

nType Parameter	Description and nIndex Parameter
CCP_SERIE	Pattern of a series has changed, nIndex represents the series
CCP_ONE	Pattern of markers has changed, nIndex <b><i>Not used.</i></b>

### Return Value

The return value is not used (***Must NOT be set***)

## ChangePattPal

### Description

This event occurs when the user changed a pattern in the PatternBar.

### Parameters

Name	Type	Description
nIndex	Integer	Index of the changed pattern.

### Return Value

The return value is not used (***Must NOT be set***)



## ChangeString

### Description

This event occurs when the user is trying to change a string (legend) in the DataEditor.

### Parameters

Name	Type	Description
nType	Integer	Legend code (See remarks)
nIndex	Integer	Index of the legend to is about to be changed.

### Remarks

This event is only generated if the style of the chart contains the **CS\_EDITABLE** flag.

nType Parameter	Description and nIndex Parameter
<b>CCS_LEGEND</b>	Changing X Legend, nIndex represents the point.
<b>CCS_SERLEGEND</b>	Changing Series Legend, nIndex represents the series.

### Return Value

The return value is NULL (default) if Chart FX must accept the user changes.

See [Returning values](#) for a detailed description of how to return values in an event.

## ChangeType

### Description

This event occurs when the user is trying to change a value in the DataEditor.

### Parameters

Name	Type	Description
nType	Integer	New Type.

### Remarks

This event is generated when the gallery type of the chart is changed.

### Return Value

The return value is not used (***Must NOT be set***)

## ChangeValue

### Description

This event occurs when the user is trying to change a value in the DataEditor.

### Parameters

Name	Type	Description
dValue	Double	New value to be set.
nSerie	Integer	Clicked serie.
nPoint	Integer	Clicked point.

### Remarks

This event is only generated if the style of the chart contains the **CS\_EDITABLE** flag.

### Return Value

The return value is NULL (default) if Chart FX must accept the user changes.

See [Returning values](#) for a detailed description of how to return values in an event.

## Destroy

### Description

This event occurs when the chart is about to be destroyed

### Parameters

None.

### Remarks

Please note that if the chart is being destroyed due to destruction of its parent, this event will not be generated, instead you must use the Destroy event of the parent window.

### Return Value

The return value is not used (***Must NOT be set***)

## GetLegend

### Description

This event occurs when Chart FX is about to draw a legend in the Y axis or the X axis of a scatter chart. In order to get/set the string to show you must use the **HText Property**.

### Parameters

Name	Type	Description
bYText	Integer	TRUE if drawing the Y axis.

### Remarks

This event is only generated if the **XLegType Property** contains the **CL\_GETLEGEND** flag..

### Return Value

If the return value is different from 0 Chart FX will draw the legend contained in the **Htext**.

See [Returning values](#) for a detailed description of how to return values in an event.

## GotFocus

### Description

This event occurs when the chart gains the focus

### Parameters

None.

### Return Value

The return value is not used (***Must NOT be set***)

## LButtonDbIClk

### Description

This event occurs when the user makes a double click on any portion of the chart.

### Parameters

Name	Type	Description
X	Integer	X screen coordinate.
Y	Integer	Y screen coordinate.
nSerie	Integer	Clicked serie.
nPoint	Integer	Clicked point.

### Remarks

nSerie and nPoint are both set to **0xFFFF** if the user has not double clicked on any marker.

### Return Value

The return value is NULL (default) if Chart FX must process this event according to the chart settings (See [DbIClk Method](#)), if the return value is not null Chart FX does nothing.

See [Returning values](#) for a detailed description of how to return values in an event.

## LostFocus

### Description

This event occurs when the chart loses the focus

### Parameters

None.

### Return Value

The return value is not used (***Must NOT be set***)



## Menu

### Description

This event occurs when the user selects a menu item in a programmer defined popup menu displayed by Chart FX. This menu can be shown in response of a double click (See [DblClk Method](#)) or a right click of the mouse (See [RigClk method](#)).

### Parameters

Name	Type	Description
wParam	Integer	ID of the menu item.
nSerie	Integer	Clicked serie.
nPoint	Integer	Clicked point.

### Remarks

This event is generated only if the user makes a click over a marker.

### Return Value

The return value is not used (***Must NOT be set***)

## RButtonDown

### Description

This event occurs when the user makes a right click on any portion of the chart.

### Parameters

Name	Type	Description
X	Integer	X screen coordinate.
Y	Integer	Y screen coordinate.
nSerie	Integer	Clicked serie.
nPoint	Integer	Clicked point.

### Remarks

nSerie and nPoint are both set to **-1** if the user has not right clicked on any marker.

### Return Value

The return value is NULL (default) if Chart FX must process this event according to the chart settings (See [RigClk Method](#)), if the return value is not null Chart FX does nothing.

See [Returning values](#) for a detailed description of how to return values in an event.

## **ReadFile**

### **Description**

This event occurs when the user read a file (ImportFile menu option).

### **Parameters**

None.

### **Return Value**

The return value is not used (***Must NOT be set***)

## ReadTemplate

### Description

This event occurs when the user read a template (ReadTemplate menu option).

### Parameters

None.

### Return Value

The return value is not used (***Must NOT be set***)

## UserScroll

### Description

This event occurs when the user scroll through the chart.

### Parameters

Name	Type	Description
wMsg	Integer	Scroll type (as wParam in WM_HSCROLL)
wParam	Integer	Info (as Loword(IParam) in WM_HSCROLL)

### Remarks

This event can be used to "synchronize" two charts in conjunction with the **Scroll** Property.

### Return Value

The return value is not used (***Must NOT be set***)

## Chart types

LINE	CT_3D	CT_SHOWZERO	CT_PIEVALUES
BAR	CT_HORZ	CT_EACHBAR	
SPLINE	CT_TOOL	CT_CLUSTER	
MARK	CT_PALETTE	CT_SHOWDATA	
PIE	CT_PATTERN	CT_DLGRAY	
AREA	CT_MENU	CT_SCATTERLINE	
PARETO	CT_LEGEND	CT_COLORLINE	
SCATTER	CT_TOGETHER	CT_NOAREALINE	
HILOW	CT_POINTS	CT_NOBORDERS	

### Comments:

The first set of constants specify the type of chart you want to make or change (see [Type](#)).

Constant	Description
LINE	Line Chart
BAR	Bar Chart (Including Horizontal, and stacked charts)
SPLINE	Curve-fitting Chart
MARK	Point Chart
PIE	Pie Chart
AREA	Area Chart (Including stacked charts)
PARETO	Pareto Chart (Statistical Chart. Special)
SCATTER	Scatter Chart
HILOW	Hi-Low Close Chart

The second set of constants are used to specify various aspects of the chart design. These constants can be combined using a bitwise **OR**.

Constant	Description
CT_3D	To specify that the graph will be created or modified with 3D effect, if you include this constant the chart will be created as a 3D chart (if supported).
CT_HORZ	This constant works only with Bar Charts, and if it is included the bar chart will be created as a horizontal bar chart.
CT_TOOL	This constant specifies that the toolbar will be shown in the window containing the chart. This gives the end user access to the tools provided by the toolbar
CT_PALETTE	This constant will turn on the Palette Bar, providing the end user the ability to change the colors of several objects in the chart, such as: series, background, etc.
CT_PATTERN	This constant will turn on the Pattern Bar, providing the end user the ability to change patterns used in the series of the chart.
CT_MENU	This constant will turn on the menu of the chart, that provides the end user access to several options to modify the aspect of the chart.

<b>CT_LEGEND</b>	This constant will turn on the legend window in the chart.
<b>CT_TOGETHER</b>	This constant will cause a Bar Chart to display joined bars.
<b>CT_POINTS</b>	This constant will show the points on a Line or Spline Chart.
<b>CT_SHOWZERO</b>	This constant will cause a chart to set the starting point at zero. For example, if you have a bar chart with a minimum value of -50 and turn on this constant the starting point will be zero and you will have bars that go up or down, depending on their value.
<b>CT_EACHBAR</b>	This constant is used to specify that a chart with a single series will have distinct colors at each data marker. i.e. Each bar will have different colors.
<b>CT_CLUSTER</b>	This constant turn on the cluster options in which each data series is in its own row. To turn on this constant the CT_3D constant must be turned on.
<b>CT_SHOWDATA</b>	This constant turns on the Data Editor (When this options is enabled the chart will not be visible).
<b>CT_DLGGRAY</b>	This constant will cause the dialogs to be shown with a gray background, to provide support for applications that also use gray backgrounds. This keeps the graphics library consistent with the rest of the client application.
<b>CT_SCATTERLINE</b>	This constant specifies that a line must be drawn between the points of a Scatter chart.
<b>CT_COLORLINE</b>	This constant specifies that the lines of a 2D Line Chart must be drawn using colours (the default behavior is to draw black lines)
<b>CT_NOAREALINE</b>	This constant specifies that the vertical lines of an Area Chart will not be drawn.
<b>CT_NOBORDERS</b>	This constant turns off the borders in bar charts.
<b>CT_PIEVALUES</b>	This constant specifies that the values must be painted in the pie chart (instead of painting the percentages).

## Chart status constants

Flag	If the user has changed...
CHART_GSVALUES	Values
CHART_GSLEGENDS	X or Series Legend
CHART_GSCOLORS	Colors
CHART_GSPATTERNS	Patterns
CHART_GSPALETTE	Colors Palette
CHART_GSPATTPAL	Pattern Palette
CHART_GSREADTEMP	The user has read a template
CHART_GSREADFILE	The user has read a file
CHART_GSGALLERY	The gallery type
CHART_GSOPTIONS	The options dialog



## Chart styles table

<b>CS_CHLINE</b>	<b>CS_SCROLLABLE</b>	<b>CS_EDITABLE</b>
<b>CS_CHBAR</b>	<b>CS_MINMAX</b>	<b>CS_FILEEXPORT</b>
<b>CS_CHSPLINE</b>	<b>CS_3D</b>	<b>CS_FILEIMPORT</b>
<b>CS_CHMARK</b>	<b>CS_HORZ</b>	<b>CS_TEMPLATE</b>
<b>CS_CHPIE</b>	<b>CS_TOGETHER</b>	<b>CS_PRINTABLE</b>
<b>CS_CHAREA</b>	<b>CS_STACKED</b>	<b>CS_3DVIEW</b>
<b>CS_CHPARETO</b>	<b>CS_SHOWPOINT</b>	<b>CS_RESIZEABLE</b>
<b>CS_CHSCATTER</b>	<b>CS_SCALE</b>	<b>CS_COPY</b>
<b>CS_CHHILOW</b>	<b>CS_TITLES</b>	<b>CS_ALL</b>
<b>CS_CHDEFAULT</b>	<b>CS_GRID</b>	

With these constants you can restrict the access to several functions provided to the end user by the menu or toolbar. Include the following constants in a bitwise OR, and you will activate that feature for the end user.

<b>Constant</b>	<b>Description</b>
<b>CS_CHLINE</b>	This will permit the end user to change to a Line chart, if another type of chart is being displayed. This feature can be accessed from the toolbar or from the menu.
<b>CS_CHBAR</b>	This will permit the end user to change to a Bar chart, if another type of chart is being displayed. This feature can be accessed from the toolbar or from the menu.
<b>CS_CHSPLINE</b>	This will permit the end user to change to a Curve-fitting chart, if another type of chart is being displayed. This feature can be accessed from the toolbar or from the menu.
<b>CS_CHMARK</b>	This will permit the end user to change to a Points chart, if another type of chart is being displayed. This feature can be accessed from the toolbar or from the menu.
<b>CS_CHPIE</b>	This will permit the end user to change to a Pie chart, if another type of chart is being displayed. This feature can be accessed from the toolbar or from the menu.
<b>CS_CHAREA</b>	This will permit the end user to change to an Area chart, if another type of chart is being displayed. This feature can be accessed from the toolbar or from the menu.
<b>CS_CHPARETO</b>	This will permit the end user to change to a Pareto chart (if supported), if another type of chart is being displayed. This feature can be accessed from the menu only
<b>CS_CHSCATTER</b>	This will permit the end user to change to a Scatter chart (if supported), if another type of chart is being displayed. This feature can be accessed from the menu only
<b>CS_CHHILOW</b>	This will permit the end user to change to a Hi-Low-Close chart (if supported), if another type of chart is being displayed. This feature can be accessed from the menu only.
<b>CS_MINMAX</b>	This will permit the end user to change the minimum or maximum value being used in the chart from the options dialog.

<b>CS_3D</b>	This will permit the end user to switch to 3D view any type of chart that is being displayed in 2D, this function can be accessed from the toolbar or from the menu.
<b>CS_HORZ</b>	This will permit the end user to change the aspect of a bar chart to be displayed in horizontal bars. Since this type of chart belongs to the family of standard bar chart, you can restrict the end user from changing to a horizontal bar chart. This option can be accessed from the toolbar or the options dialog.
<b>CS_TOGETHER</b>	By default all the bar charts are created with a gap between each bar. the user can remove this gap from the options dialog if you permit this feature with this constant.
<b>CS_STACKED</b>	This will permit the end user to change the aspect of a chart change to stack type. Since this type of chart belongs to the family of the standard bar (area) chart, you can restrict the user from changing to a stacked bar (area) chart. This option can be accessed from the options dialog
<b>CS_SHOWPOINT</b>	This will permit the end user to show or hide the points in a line, spline or similar charts. This function can be accessed from the options dialog.
<b>CS_SCALE</b>	This will permit the end user to change scale used on the Y-axis from the options dialog.
<b>CS_TITLES</b>	This will permit the end user to change the text being assigned to the different titles supported (Top, Bottom, Right or Left). This function is provided in the options dialog.
<b>CS_FONTS</b>	This will permit the end user to change the fonts used in any of the titles supported. This function is provided in the menu.
<b>CS_EDITABLE</b>	This will permit the end user to change the values actually being graphed. This function is provided in the Data Editor.
<b>CS_FILEEXPORT</b>	This will permit the end user to export (save) the current to a file. This function is provided in the toolbar or menu.
<b>CS_FILEIMPORT</b>	This will permit the end user to import (open) a previously saved chart. This function is provided in the toolbar or menu.
<b>CS_SCROLLABLE</b>	This will permit the end user to scroll if the current chart will not fit in the open window.
<b>CS_PRINTABLE</b>	This will permit the end user to print the contents of the chart window. This function is provided in the toolbar and in the menu.
<b>CS_3DVIEW</b>	This will permit the end user to modify the 3D View by accesing the 3D dialog where the user can rotate the view around the x or Y-axis.
<b>CS_GRID</b>	This will permit the end user to modify the actual grids (Vertical, Horizontal or None) being displayed in the chart. This function is provided in the toolbar and in the menu.
<b>CS_RESIZEABLE</b>	This will permit the end user to modify the internal borders of the chart. Note that if the chart is being displayed in a child window, the end user can resize the graph inside that window, but not the window itself.
<b>CS_TEMPLATE</b>	This will permit the end user to operate (Save or apply) templates to a chart. This function is provided in the menu.
<b>CS_COPY</b>	This will permit the end user to copy bitmap or data of actual chart to the clipboard, functionality provided in the toolbar and menu.
<b>CS_ALL</b>	This will permit the end user to access all the functions explained above.

## **chart\_Get Constants**

<b>Constant</b>	<b>To get ...</b>
<b>CHART_GVALUES</b>	plotted values
<b>CHART_GXVALUES</b>	X-axis values (in scatter charts only)
<b>CHART_GINIVALUES</b>	Initial values (in bar charts only)

## **chart\_OpenData Constants**

<b>Constant</b>	<b>Open the communication channel to supply ...</b>
<b>COD_VALUES</b>	Plotted values
<b>COD_CONSTANTS</b>	Constants
<b>COD_COLORS</b>	Colors
<b>COD_STRIPES</b>	Stripes
<b>COD_INIVALUES</b>	Initial values (for bar charts only)
<b>COD_XVALUES</b>	X-axis values (for scatter charts only)
<b>COD_STATUSITEMS</b>	StatusBar items

## **chart\_SetAdm Constants**

<b>Constant</b>	<b>To change the ...</b>
<b>CSA_MIN</b>	Minimum value used on the Y-axis
<b>CSA_MAX</b>	Maximum value used on the Y-axis
<b>CSA_GAP</b>	Y-axis values Gap.
<b>CSA_SCALE</b>	Used scale.
<b>CSA_YLEYGAP</b>	Equivalent units for a Y-axis legend text.
<b>CSA_PIXXVALUE</b>	Equivalent unit representation of Y-axis in pixels.
<b>CSA_XMIN</b>	Minimum value used on the X-axis (scatter).
<b>CSA_XMAX</b>	Maximum value used on the X-axis (scatter).

### **Click styles**

**CHART\_BALLOONCLK**

Displays a Balloon Help

**CHART\_DIALOGCLK**

Displays a predefined Dialog

**CHART\_NONECLK**

Nothing

**CHART\_MENUCLK**

Displays a Menu (This value is only supported at runtime)

## **Color modes**

**CHART\_FGROUND**

To set foreground color

**CHART\_BGROUND**

To set background color (used in pattern schemes)

## **Color schemes**

**CHART\_CSSOLID**

Use solid colors (BkColor is not used)

**CHART\_CSBWPATTERN**

Use BW Patterns (useful for monochrome devices)

**CHART\_CSPATTERN**

Use color patterns



## Decimal items

Constant	Used to change number of decimals used in ...
CD_ALL	All the items.
CD_VALUES	Point Values (Ballon, Dialog and Data-Editor)
CD_YLEG	Y Legend Values.
CD_XLEG	X Legend Values.

## Fonts

CF_BOLD	CF_FSCRIPT	CF_SCRIPT
CF_ITALIC	CF_FDECORATIVE	CF_SYMBOL
CF_UNDERLINE	CF_ARIAL	CF_TIMES
CF_STRIKEOUT	CF_COURIER	CF_TIMESNR
CF_FDONTCARE	CF_COURIERNEW	CF_WINGDINGS
CF_FROMAN	CF_HELVETICA	
CF_FSWISS	CF_MODERN	
CF_FMODERN	CF_ROMAN	

### **/\* Font Effects Supported\*/**

Constant	Description
CF_BOLD	Specifies whether the font is Bold
CF_ITALIC	Specifies whether the font is Italic
CF_UNDERLINE	Specifies whether the font is Underline
CF_STRIKEOUT	Specifies whether the font is Strikeout

### **/\* Font Families Supported\*/**

Note: The font family is used in case the font specified is not found (or not installed) in Windows environment, by specifying this family Windows can create a similar font that you want to set.

To obtain more information about font families please refer to Windows SDK help file.

Constant	Description
CF_FDONTCARE	No family given
CF_FROMAN	Specifies whether the family is Roman
CF_FSWISS	Specifies whether the family is Swiss
CF_FMODERN	Specifies whether the family is Modern
CF_FSCRIPT	Specifies whether the family is Script
CF_FDECORATIVE	Specifies whether the family is Decorative

### **/\* Font Typefaces Supported\*/**

Constant	Description
CF_ARIAL	Specifies whether the Typeface is Arial
CF_COURIER	Specifies whether the Typeface is Courier
CF_COURIERNEW	Specifies whether the Typeface is Courier New
CF_HELVETICA	Specifies whether the Typeface is Helvetica
CF_MODERN	Specifies whether the Typeface is Modern
CF_ROMAN	Specifies whether the Typeface is Roman
CF_SCRIPT	Specifies whether the Typeface is Script
CF_SYMBOL	Specifies whether the Typeface is Symbol
CF_TIMES	Specifies whether the Typeface is Times

**CF\_TIMESNEW**

Specifies whether the Typeface is Times New Roman

**CF\_WINGDINGS**

Specifies whether the Typeface is WingDings

## Font types

<b>CHART_LEFTFT</b>	Left Title
<b>CHART_RIGHTFT</b>	Right Title
<b>CHART_TOPFT</b>	Top Title
<b>CHART_BOTTOMFT</b>	Bottom Title
<b>CHART_XLEGFT</b>	X Legend
<b>CHART_YLEGFT</b>	Y Legend
<b>CHART_FIXEDFT</b>	Constants
<b>CHART_LEGENDFT</b>	Legend

## **Grid styles**

<b>CHART_NONEGRID</b>	Remove both grids
<b>CHART_HORZGRID</b>	Set horizontal grid
<b>CHART_VERTGRID</b>	Set vertical grid
<b>CHART_BOTHGRID</b>	Set both grids

## High-low constants

Constant	Description
HLC_HIGH	High Series for a High-Low-Close chart
HLC_LOW	Low Series for a High-Low-Close chart
HLC_CLOSE	Close Series for a High-Low-Close chart
OHLC_OPEN	Open Series for an Open-High-Low-Close chart
OHLC_HIGH	High Series for an Open-High-Low-Close chart
OHLC_LOW	Low Series for an Open-High-Low-Close chart
OHLC_CLOSE	Close Series for an Open-High-Low-Close chart

## Line styles

Constant	Style Description
CHART_SOLID	Solid Pen
CHART_DASH	Dashed Pen
CHART_DOT	Dotted Pen
CHART_DASHDOT	Dash-Dotted Pen
CHART_DASHDOTDOT	Dash-Dot-Dotted Pen

## Point types

<b>CHART_RECTMK</b>	Displays a Rectangle
<b>CHART_CIRCLEMK</b>	Displays a Circle
<b>CHART_TRIANGLEMK</b>	Displays a Triangle
<b>CHART_MARBLEMK</b>	Displays a Marble
<b>CHART_CUBEMK</b>	Displays a Cube (in 2D displays a rectangle)
<b>CHART_MANYMK</b>	Displays different type of markers
<b>CHART_NONEMK</b>	No points



## **Stack styles**

<b>CHART_NOSTACKED</b>	Remove stacked option.
<b>CHART_STACKED</b>	Set stacked option
<b>CHART_STACKED100</b>	Set stacked 100% option

## **Status Items styles**

<b>CHART_STLEFT</b>	Left Alignment
<b>CHART_STCENTER</b>	Center
<b>CHART_STRIGHT</b>	Right Alignment

## **Titles styles**

<b>CHART_LEFTTIT</b>	Left Title
<b>CHART_RIGHTTIT</b>	Right Title
<b>CHART_TOPTIT</b>	Top Title
<b>CHART_BOTTOMTIT</b>	Bottom Title

## Tools constants

Flag	Grouped in
CST_IMPORT	CST_FILE, CST_FILEEDIT
CST_EXPORT	CST_FILE, CST_FILEEDIT
CST_COPYDATA	CST_COPY, CST_FILEEDIT
CST_COPYBITMAP	CST_COPY, CST_FILEEDIT
CST_PRINT	CST_FILEEDIT
CST_AREA	CST_GALLERY
CST_BAR	CST_GALLERY
CST_BARHORZ	CST_GALLERY
CST_LINE	CST_GALLERY
CST_MARK	CST_GALLERY
CST_PIE	CST_GALLERY
CST_SPLINE	CST_GALLERY
CST_PARETO	CST_GALLERYEXT
CST_SCATTER	CST_GALLERYEXT
CST_HILOW	CST_GALLERYEXT
CST_3D	CST_VIEW
CST_ROTATE	CST_VIEW
CST_CLUSTER	CST_VIEW
CST_LEGEND	CST_LEGGRID
CST_VGRID	CST_LEGGRID
CST_HGRID	CST_LEGGRID
CST_TITLES	CST_OTHER
CST_FONTS	CST_OTHER
CST_TOOLS	CST_OTHER
CST_OPTIONS	CST_OTHER
CST_SPACE1	
CST_SPACE2	
CST_SPACE3	
CST_SPACE4	

## CHART\_STITEM struct

```
typedef struct tagSTITEM {  
    LPSTR sText;           // Initial Text, NULL means never will have text  
    UINT wIdm;             // Text ID, needed to change text  
    BOOL bFrame;           // draw frame surrounding  
    int nWidth;            // Initial width  
    int nMin;              // Minimum width  
    int nDesp;             // Offset in pixels respect previous item  
    DWORD dwStyle;         // style of text  
} CHART_STITEM;
```

## Returning values

To return values from certain events, Chart FX uses a special parameter called ***nRes***. This parameter is initialized with null before calling the event handler. You can also return a different value by assigning the new value to the parameter.

