Querix 4GL UI types

Elena Krivtsova, Olga Gusarenko

July 3, 2022

Contents

Chapter 1

4GL UI Entities Description

1.1 ui.AbstractBoolField

1.1.1 Brief description

It is an abstract UI element, which unites the concrete UI elements that can be in one of the two states: enabled (TRUE) or disabled (FALSE). The concrete UI elements that inherit their properties from the AbstractBoolField are ui.CheckBox .

1.1.2 Inheritance Diagram

1.1.3 Fields Description

Table 1.1: Fields description

Field Name	Type	Description	Field Accessors
AllowNewlines	BOOLEAN	This property specifies whether the Enter key will be used to move to another form element at runtime (if the value is FALSE), or it will create a newline symbol inside the current field (if the value is TRUE). It is typically applied for the ui.TextArea element.	SetAllowNewlines, GetAllowNewlines
Image	ui.Image	It is an image that can be applied to other UI elements, e.g. to a button.	SetImage, GetImage

Continued on next page

Table 1.1 – *Continued from previous page*

Field Name	Type	Description	Field Accessors
IsChecked	BOOLEAN	The UI element that has such field can be either in checked state (TRUE) or unchecked state (FALSE). UI elements like check boxes or radio buttons typically contain such field. Every time the element is clicked, the state is flipped.	SetIsChecked, GetIsChecked
OnCheck	ui.EventHandler	The OnCheck field defines the event which will be triggered if the IsChecked field of the UI element is changed to TRUE.	SetOnCheck, GetOnCheck
OnUncheck	ui.EventHandler	The OnUncheck field defines the event which will be triggered if the IsChecked field of the UI element is changed to FALSE.	SetOnUncheck, GetOnUncheck
Title	String	This is the inscription attached to the UI element. Usually this is the text of all sorts of labels.	SetTitle, GetTitle

1.1.4 Static Methods Description

Table 1.2: Static methods description

Name	Parameters	Description
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

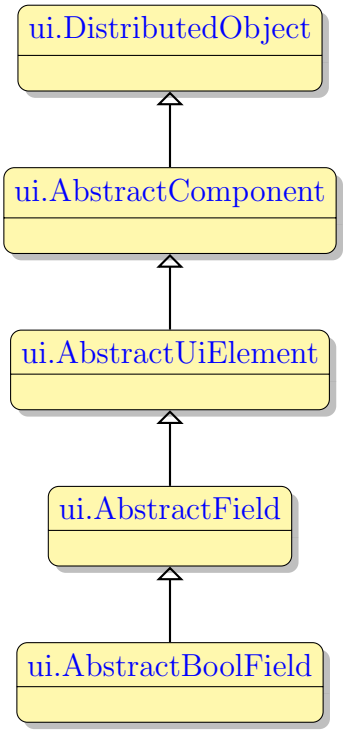

Figure 1.1: Inheritance Diagram of ui.AbstractBoolField

1.2 ui.AbstractComponent

1.2.1 Brief description

This is the common parent of all UI elements.

1.2.2 Inheritance Diagram

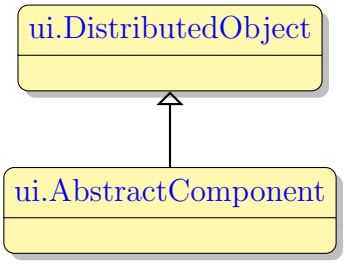

Figure 1.2: Inheritance Diagram of ui.AbstractComponent

1.2.3 Fields Description

Table 1.3: Fields description

Field Name	**Type**	**Description**	**Field Accessors**
Identifier	String	It is a unique name of a UI element by which it can be referenced.	SetIdentifier, GetIdentifier

1.3 ui.AbstractContainer

1.3.1 Brief description

This UI element represents an abstract container from which all the form containers their properties. This abstract UI element unites all form containers - elements that determine the form layout.

1.3.2 Inheritance Diagram

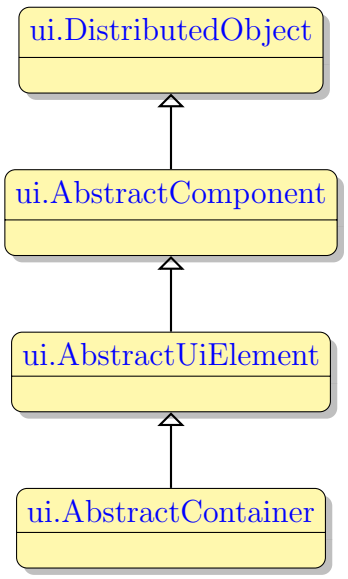

Figure 1.3: Inheritance Diagram of ui.AbstractContainer

1.3.3 Static Methods Description

Table 1.4: Static methods description

Name	Parameters	Description
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.4 ui.AbstractDataTable

1.4.1 Brief description

This UI element is used to display and edit data in a customized two-dimensional table of cells. The data in the cell therefore can be retrieved by specifying the row and column identifier of that cell in the table. AbstractDataTable UI element manages the overall appearance and behavior of the table, but does not have direct influence on the columns and rows.

1.4.2 Inheritance Diagram

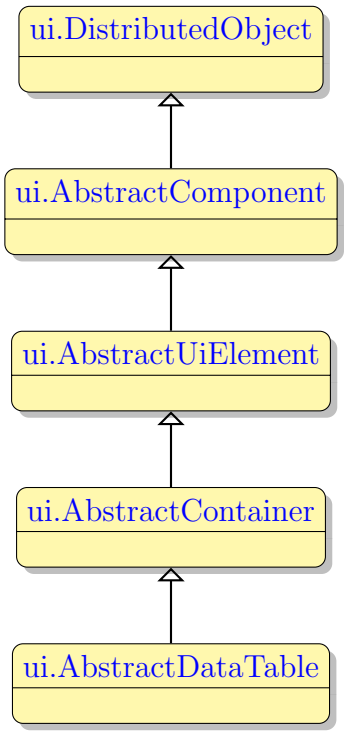

Figure 1.4: Inheritance Diagram of ui.AbstractDataTable

1.4.3 Fields Description

Table 1.5: Fields description

Field Name	Type	Description	Field Accessors

Continued on next page

Table 1.5 – *Continued from previous page*

Field Name	Type	Description	Field Accessors
BufferLength	Int	It defines the number of rows that will be loaded into the table at a time. It is only applicable if the buffering is enabled.	SetBufferLength, GetBufferLength
ColumnEdit	String	It should be assigned to the column containing the labels for the tree items. By default is is the first column of the table.	SetColumnEdit, GetColumnEdit
ColumnExpanded	String	It should be assigned to the column which indicates whether each tree element should be collapsed or expanded when the tree is first displayed at runtime. It is an optional column in the array that is used in the DISPLAY ARRAY for the tree container. In this column each row should have value 1, if the element on the row should be expanded, and 0 if it should be collapsed.	SetColumnExpanded, GetColumnExpanded
ColumnId	String	It specifies the identifier of the column that stores the id of the row. If a column's identifier is specified in here, the column becomes hidden.	SetColumnId, GetColumnId
ColumnImage	String	It should be assigned to the column which contains individual images for each tree element. It is an optional column in the array that is used in the DISPLAY ARRAY for the tree container. In this column each row should contain a BYTE value which will be displayed next to the tree element at runtime.	SetColumnImage, GetColumnImage

Continued on next page

Table 1.5 – *Continued from previous page*

Field Name	Type	Description	Field Accessors
ColumnIsNode	String	It should be assigned to the column which indicates the tree items that have children. It is an optional column in the array that is used in the DISPLAY ARRAY for the tree container. In this column each row should have value 1, if the element on the row has children and 0 if it does not. For the rows where 1 is set, the icons indicating that the element includes a sub-tree will be shown next to the element at runtime even if it does not factually have any children. The elements for which 0 is set will look as if they have no children even if they actually do.	SetColumnIsNode, GetColumnIsNode
ColumnParentId	String	It specifies the identifier of the column that stores the id of the parent tree element which serves as the root of the sub-tree to which each row belongs. If a column's identifier is specified in here, the column becomes hidden.	SetColumnParentId, GetColumnParentId
FirstRowNum	Int	No information	SetFirstRowNum, GetFirstRowNum
GridColor	ui.Color	The color of the grid lines that separate one table cell from the other cells.	SetGridColor, GetGridColor
ImageCollapsed	ui.ResourceId	It specifies the icon to be shown next to a collapsed tree element which has a sub-tree. Its priority is lower than that of the ImageColumn and it is ignored if both are used.	SetImageCollapsed, GetImageCollapsed
ImageExpanded	ui.ResourceId	It specifies the icon to be shown next to an expanded tree element which has a sub-tree. Its priority is lower than that of the ImageColumn and it is ignored at runtime if both are used.	SetImageExpanded, GetImageExpanded

Continued on next page

Table 1.5 – *Continued from previous page*

Field Name	Type	Description	Field Accessors
ImageLeaf	ui.ResourceId	It specifies the global icon for the tree elements that do not have the nested elements / sub-trees. Its priority is lower than ImageColumn and is ignored at runtime if ImageColumn is also set.	SetImageLeaf, GetImageLeaf
Indent	Int	It specified how far should the tree elements in each sub-tree be offset to the right. It is used if the AutoIndent is set to false.	SetIndent, GetIndent
IsMultiSelect	BOOLEAN	It enables or disables the possibility to select multiple rows of one table during DISPLAY ARRAY execution. The default value is FALSE - the multi-selection is turned off.	SetIsMultiSelect, GetIsMultiSelect
MultipleSelect	BOOLEAN	No information	SetMultipleSelect, GetMultipleSelect
OnCollapse	ui.EventHandler	It is the event that is triggered when the tree or sub-tree received the command to collapse (the user clicked on the collapse button).	SetOnCollapse, GetOnCollapse
OnExpand	ui.EventHandler	It is the event that is triggered when the tree or sub-tree received the command to unfold (the user clicked on the unfold button).	SetOnExpand, GetOnExpand
OnFillBuffer	ui.EventHandler	If the dialog is using the paged mode, this event is triggered every time a new page is loaded.	SetOnFillBuffer, GetOnFillBuffer
OnSelectionChanged	ui.EventHandler	It defines an event which must be triggered if the current row is changed or if a new row is selected or deselected, if the multiselect mode is on.	SetOnSelectionChanged, GetOnSelectionChanged
RowHeight	String	It defines the default height of a table row in pixels.	SetRowHeight, GetRowHeight

Continued on next page

Table 1.5 – *Continued from previous page*

Field Name	Type	Description	Field Accessors
ScrollBarMaxValue	Int	This field has effect only if the table is virtual. It defines the maximum number of rows that can be loaded and displayed to the table. The rows themselves would not be loaded unless the user scrolls and the client requests them to be loaded. The ScrollBarMaxValue defines the appearance of the vertical scrollbar of a table so that the scrollbar visually corresponds to the number of rows it potentially can scroll.	SetScrollBarMaxValue, GetScrollBarMaxValue
SelectedMany	BOOLEAN	No information	SetSelectedMany, GetSelectedMany
StartLoadedIndex	Int	This property defines the first of the loaded rows. When the table is just loaded and user did not scroll anywhere, its value is 0. After the user, for example, scrolled to the middle of the set of rows the StartLoadedIndex will be equal to the first row of the current buffered set of rows.	SetStartLoadedIndex, GetStartLoadedIndex
TableColumns	list of ui.TableColumn	A set of columns that belong to the same table.	SetTableColumns, GetTableColumns

1.4.4 Extra Methods Description

Table 1.6: Extra methods description

Name	Parameters	Description
Complete		Complete dynamically created control.

1.4.5 Static Methods Description

Table 1.7: Static methods description

Name	**Parameters**	**Description**
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.5 ui.AbstractField

1.5.1 Brief description

This UI element represents an abstract field from which all the form widgets inherit their properties. This abstract UI element unites all form fields - the form elements that can accept and display data - as opposed tp form containers - elements that determine the form layout.

1.5.2 Inheritance Diagram

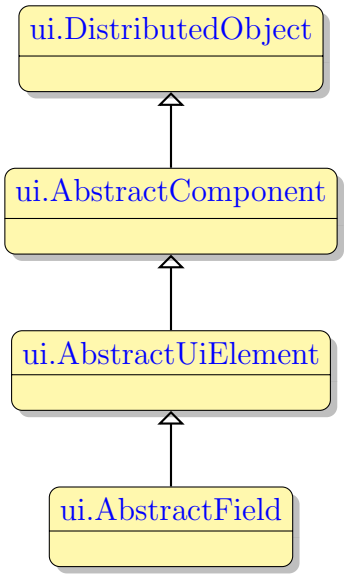

Figure 1.5: Inheritance Diagram of ui.AbstractField

1.5.3 Fields Description

Table 1.8: Fields description

Field Name	Type	Description	Field Accessors
AutoCompleteList	list of String	No information	SetAutoCompleteList, GetAutoCompleteList
InvokeAction	String	No information	SetInvokeAction, GetInvokeAction

Continued on next page

Table 1.8 – *Continued from previous page*

Field Name	Type	Description	Field Accessors
OnTouched	ui.EventHandler	No information	SetOnTouched, GetOnTouched
OnValueChanged	ui.EventHandler	This event is triggered when the value of the UI element changes. The value of the element is the value which will be recorded to the underlying variable when the input finishes.	SetOnValueChanged, GetOnValueChanged
ReadOnly	BOOLEAN	If enabled, it prevents the user from entering values into the field at runtime even if the field is included into the input routine.	SetReadOnly, GetReadOnly

1.5.4 Static Methods Description

Table 1.9: Static methods description

Name	Parameters	Description
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.6 ui.AbstractRangeField

1.6.1 Brief description

It is an abstract UI element, which unites the concrete UI elements which accept only the values included into the specified range. It is typically a range or numeric values, for example from 1 to 100. The concrete UI elements that inherit their properties from the AbstractRangeField are ui.Slider , ui.ProgressBar , ui.Spinner , and ui.ScrollBar .

1.6.2 Inheritance Diagram

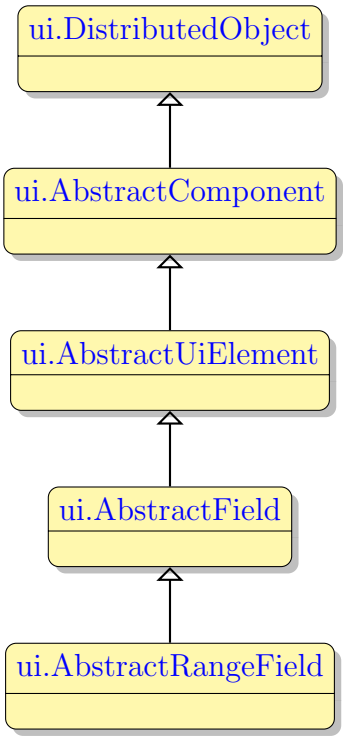

Figure 1.6: Inheritance Diagram of ui.AbstractRangeField

1.6.3 Fields Description

Table 1.10: Fields description

Field Name	Type	Description	Field Accessors

Continued on next page

Table 1.10 – *Continued from previous page*

Field Name	Type	Description	Field Accessors
CurrentValue	Int	The value that the UI element has at the moment, it must be within the range of accepted values.	SetCurrentValue, GetCurrentValue
MaxValue	Int	The maximum value in the range of values accepted by a UI element.	SetMaxValue, GetMaxValue
MinValue	Int	The minimum value in the range of values accepted by a UI element.	SetMinValue, GetMinValue

1.6.4 Static Methods Description

Table 1.11: Static methods description

Name	Parameters	Description
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.7 ui.AbstractStringField

1.7.1 Brief description

It is an abstract UI element, which unites the concrete UI elements that accept a character string as their value. Most of the concrete UI elements that are not containers inherit their properties from this element.

1.7.2 Inheritance Diagram

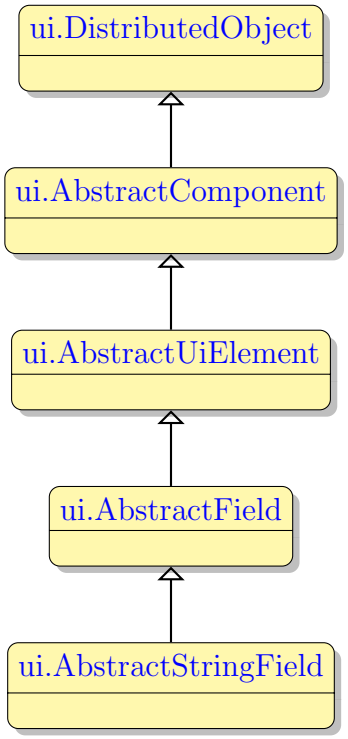

Figure 1.7: Inheritance Diagram of ui.AbstractStringField

1.7.3 Fields Description

Table 1.12: Fields description

Field Name	Type	Description	Field Accessors

Continued on next page

Table 1.12 – *Continued from previous page*

Field Name	Type	Description	Field Accessors
Text	String	This is the value of the UI element, typically of a text field or a combo box which is recorded to the variable linked to it after the input or which is displayed to it.	SetText, GetText

1.7.4 Static Methods Description

Table 1.13: Static methods description

Name	Parameters	Description
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.8 ui.AbstractTextField

1.8.1 Brief description

It is an abstract UI element, which unites a subset of ui.AbstractStringField elements with the exception of ui.TextArea , ui.ComboBox , and ui.Button . Typically it includes the UI elements which allow entering values, like normal text fields, and usually are only one line wide.

1.8.2 Inheritance Diagram

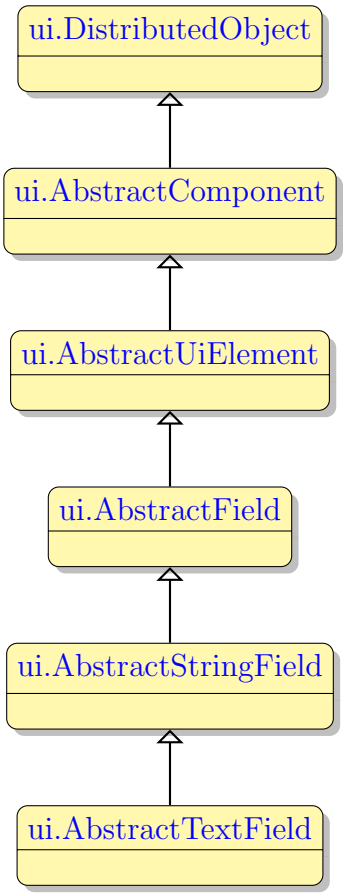

Figure 1.8: Inheritance Diagram of ui.AbstractTextField

1.8.3 Fields Description

Table 1.14: Fields description

Field Name	Type	Description	Field Accessors
Autonext	BOOLEAN	If enabled, moves the cursor to the next field during input automatically, when the MaxLength of the current field is met.	SetAutonext, GetAutonext
Editor	String	Specifies the program to be used for opening and editing the BYTE or TEXT value.	SetEditor, GetEditor
Format	String	It specifies the format pattern according to which the entered data should be formatted. Typically used for numeric values to specify the decimal point sign and location and the thousands separator.	SetFormat, GetFormat
IsPasswordMask	BOOLEAN	If enabled, it turns the entered value into a set of * signs to mask it. The value displayed to the field will also be masked with asterisks.	SetIsPasswordMask, GetIsPasswordMask
MaxLength	Int	It specifies the maximum length in bytes allowed for entering into the filed. Its value is normally taken from the data type and size of the variable linked to the field.	SetMaxLength, GetMaxLength
Required	BOOLEAN	No information	SetRequired, GetRequired
TextPicture	String	It formats the entered value by specifying that only letters or only numbers or both can be entered and by supplying delimiters. It is typically used for character values. E.g. if picture is AA-XX, the value may be ab-3c.	SetTextPicture, GetTextPicture
ToCase	String catalog for ToCase	This property specifies the case of a UI element. It can be applied to any UI element that allows entering text from keyboard. By default its value is None, meaning that the case of the letters does not change and remains as they were inputted.	SetToCase, GetToCase

1.8.4 Static Methods Description

Table 1.15: Static methods description

Name	**Parameters**	**Description**
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.9 ui.AbstractUiElement

1.9.1 Brief description

AbstractUiElement is the base class for UI widgets. It is a generic UI element that can accept user actions. Most of concrete UI elements must inherit the properties and action types from the AbstractUIElement.

1.9.2 Inheritance Diagram

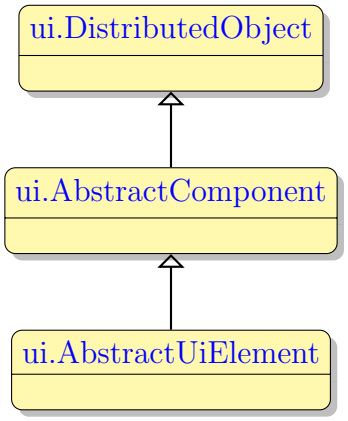

Figure 1.9: Inheritance Diagram of ui.AbstractUiElement

1.9.3 Fields Description

Table 1.16: Fields description

Field Name	Type	Description	Field Accessors
ClassNames	list of String	The name of a class that is applied to the UI element. There can be a customly created class or one of the default classes. The default classes depend on the 4GL attributes applied to the element by means of the 4GL code or form file and usually specify the colour or intensity attribute.	SetClassNames, GetClassNames

Continued on next page

Table 1.16 – *Continued from previous page*

Field Name	Type	Description	Field Accessors
Background	ui.Background	Background - defines the background type, color and other parameters.	SetBackground, GetBackground
ForeColor	ui.Color	ForeColor - foreground color of the control(used to draw text and/or control border)	SetForeColor, GetForeColor
Font	ui.Font	The font to be used for the UI element.	SetFont, GetFont
Location	ui.Location	The location of the UI element specified in pixels.	SetLocation, GetLocation
Size	ui.Size	The size of the UI element in pixels that.	SetSize, GetSize
PreferredSize	ui.Size	The size of the UI element in pixels that specified by the user that will override the size dynamically calculated at runtime.	SetPreferredSize, GetPreferredSize
MinSize	ui.Size	The minimum size of the UI element smaller than which an element cannot shrink when resized.	SetMinSize, GetMinSize
MaxSize	ui.Size	The maximum size of the UI element bigger than which an element cannot become when resized.	SetMaxSize, GetMaxSize
NotNull	BOOLEAN	If enabled, it forbids to save NULL values to the variable linked to the field.	SetNotNull, GetNotNull
Padding	ui.Thickness	The space between the contents of the UI element (e.g. text in a text field) and the border of this element.	SetPadding, GetPadding
Margin	ui.Thickness	The space between the border of the UI element and other UI elements surrounding it.	SetMargin, GetMargin
Cursor	String catalog for Cursor	The type of the cursor that should be applied when the mouse cursor is hovering above the UI Element.	SetCursor, GetCursor
Locale	ui.Locale	The custom locate of the UI element that may be different from the default locale of the application.	SetLocale, GetLocale

Continued on next page

Table 1.16 – *Continued from previous page*

Field Name	Type	Description	Field Accessors
Visible	BOOLEAN	If enabled, the UI element is visible at runtime. If disabled, it is hidden. The default value is TRUE.	SetVisible, GetVisible
Collapsed	BOOLEAN	No information	SetCollapsed, GetCollapsed
Enable	BOOLEAN	If set to TRUE (the default value), the UI element can be interacted with (e.g. button can be pressed, text can be entered into the field). If a UI element is disabled, it is grayed and inaccessible.	SetEnable, GetEnable
ContextMenu	ui.PopupMenu	It contains the information about the context to be displayed when the user right-clicks the UI element at runtime.	SetContextMenu, GetContextMenu
ToolTip	String	It specifies the text of the tooltip to be visible when the mouse hovers over the element at runtime. If its value is empty, the element will have no tooltip.	SetToolTip, GetToolTip
TabIndex	Int	It specifies the order of the UI elements located on a single form. This order can be used during input for cursor navigation.	SetTabIndex, GetTabIndex
ZOrder	Int	It specifies whcih element should be on top if two or more elements overlap. It should be applied only to elements whose container is the coordinate panel.	SetZOrder, GetZOrder
EnableBorder	BOOLEAN	If set to TRUE (the default value), shows the default 1 pixel border around UI elements. If disabled, the element will have no default border.	SetEnableBorder, GetEnableBorder
ScaleType	String catalog for ScaleType	It defines whether the element contents will be scaled, if the element is resized.	SetScaleType, GetScaleType
ElementBorder	ui.Border	Sets the custom border for a UI element.	SetElementBorder, GetElementBorder

Continued on next page

Table 1.16 – *Continued from previous page*

Field Name	Type	Description	Field Accessors
VerticalAlignment	String catalog for VerticalAlignment	Specifies the vertical alignment of the UI element inside its container.	SetVerticalAlignment, GetVerticalAlignment
HorizontalAlignment	String catalog for HorizontalAlignment	Specifies the horizontal alignment of the UI element inside its container.	SetHorizontalAlignment, GetHorizontalAlignment
OnKeyDown	ui.EventHandler	The event specified will be triggered, when the cursor is in the given UI element and any key on the keyboard is pressed down.	SetOnKeyDown, GetOnKeyDown
OnKeyUp	ui.EventHandler	The event specified will be triggered when the cursor is in the given UI element and the key on the keyboard previously pressed is released.	SetOnKeyUp, GetOnKeyUp
OnMouseDown	ui.EventHandler	The event specified will be triggered when left mouse button is clicked on the UI element.	SetOnMouseDown, GetOnMouseDown
OnMouseUp	ui.EventHandler	The event specified will be triggered when the left mouse button is released after it was clicked on the UI element.	SetOnMouseUp, GetOnMouseUp
OnMouseMove	ui.EventHandler	The event specified will be triggered when the mouse cursor is moved inside the UI element area.	SetOnMouseMove, GetOnMouseMove
OnMouseEnter	ui.EventHandler	The event specified will be triggered when the mouse cursor enters the UI element area.	SetOnMouseEnter, GetOnMouseEnter
OnMouseHover	ui.EventHandler	The event specified will be triggered when the mouse cursor enters the UI element area and remains them for a second. Triggered only once while the cursor is inside the element.	SetOnMouseHover, GetOnMouseHover
OnMouseExit	ui.EventHandler	The event specified will be triggered when the mouse cursor exits the UI element.	SetOnMouseExit, GetOnMouseExit
OnMouseWheel	ui.EventHandler	The event specified will be triggered when the mouse wheel is rotated while the cursor hovers over the UI element.	SetOnMouseWheel, GetOnMouseWheel

Continued on next page

Table 1.16 – *Continued from previous page*

Field Name	Type	Description	Field Accessors
OnMouseDoubleClick	ui.EventHandler	The event specified will be triggered when the the user double-clicks on the UI element.	SetOnMouseDoubleClick, GetOnMouse-DoubleClick
OnMouseClick	ui.EventHandler	The event specified will be triggered when the the user left-clicks on the UI element.	SetOnMouseClick, GetOn-MouseClick
OnMenuDetect	ui.EventHandler	This event is triggered when the user right-clicks the UI element to invoke context menu.	SetOnMenuDetect, GetOnMenuDe-tect
OnDragStart	ui.EventHandler	The event is triggered when the user clicks on an element, holds the mouse key and starts moving it away from its location.	SetOnDragStart, GetOnDragStart
OnDragEnter	ui.EventHandler	The event is triggered when the mouse cursor with the dragged item enters the visual boundaries of the UI element to which the item may be dropped.	SetOnDragEnter, GetOnDragEnter
OnDragOver	ui.EventHandler	The event is triggered when the mouse cursor with the item is dragged over a drop target. Typ-ically invoked after OnDragEnter event.	SetOnDragOver, GetOnDragOver
OnDragFinished	ui.EventHandler	Triggered after OnDragStart was in-voked and then OnDrop executed successfully or the drag and drop ac-tion was terminated.	SetOnDragFinished, GetOnDragFin-ished
OnDrop	ui.EventHandler	The event is triggered when the user releases the mouse button holding the dragged item over an area which allows the item to be dropped.	SetOnDrop, GetOnDrop
OnResize	ui.EventHandler	The event is triggered when the size of a UI element is changed.	SetOnResize, GetOnResize
OnSelection	ui.EventHandler	The event is triggered when a UI el-ement is selected by mouse cursor.	SetOnSelection, GetOnSelection

Continued on next page

Table 1.16 – *Continued from previous page*

Field Name	Type	Description	Field Accessors
OnFocusIn	ui.EventHandler	The event is triggered when the UI element becomes the current element, e.g. is when the cursor enters the field or when an element is selected.	SetOnFocusIn, GetOnFocusIn
OnFocusOut	ui.EventHandler	The event is triggered when the UI element stops being the current element, e.g. is when the cursor leaves the field or when an element is deselected.	SetOnFocusOut, GetOnFocusOut
TextAlignment	ui.TextAlignment	It specifies the alignment of the text withing the UI element. E.g. the placement of the text inside the label area or in a text field.	SetTextAlignment, GetTextAlignment
Wrapper	ui.Wrapper	It defines the wrapper to be applied to the element. A wrapper is typically a chart or a gauge applied to a table or a field.	SetWrapper, GetWrapper
ElementRole	String catalog for ElementRole	The role the UI element is executing at the moment. It depends on the 4GL code, thus a character string can be either a message, an error, a displayed string, etc.	SetElementRole, GetElementRole
IsProtected	BOOLEAN	If set to TRUE it prevents character strings displayed from 4GL to overlap with the UI elements. Such strings will be displayed below the UI elements where they are supposed to overlap.	SetIsProtected, GetIsProtected
Focusable	BOOLEAN	If set to TRUE, the UI element can acquire focus. All form widgets normally can acquire focus while elements that inherit their properties from ui.AbstractContainer should not be able to acquire focus.	SetFocusable, GetFocusable
HasFocus	BOOLEAN	It indicates that the UI element is selected in the moment and the 4GL cursor is located in it.	SetHasFocus, GetHasFocus

Continued on next page

Table 1.16 – *Continued from previous page*

Field Name	Type	Description	Field Accessors
BorderPanelItemLocation	String catalog for BorderPanelItemLocation	It is applicable only if the UI element is located inside the ui.BorderPanel container and indicates which part of the border panel the element occupies.	SetBorderPanelItemLocat GetBorderPanelItemLocation
GridItemLocation	ui.GridItemLocation	It is applicable only if the UI element is located inside the ui.GridPanel container and indicates which cell of the grid panel the element occupies.	SetGridItemLocation, GetGridItemLocation
AllowDrag	BOOLEAN	If set to TRUE indicates that the dragging items from the UI element is allowed to perform Drag and Drop activities.	SetAllowDrag, GetAllowDrag
AllowDrop	BOOLEAN	If set to TRUE indicates that the dropping items into the UI element is allowed to perform Drag and Drop activities.	SetAllowDrop, GetAllowDrop
TrackSizes	BOOLEAN	If set to true, the client tracks any resizing the element might undergo and sends the information about the changes to the server.	SetTrackSizes, GetTrackSizes
TrackLocation	BOOLEAN	If set to true, the client tracks any changes of the element location and sends the information about the changes to the server.	SetTrackLocation, GetTrackLocation
StyleClassName	String	The class that is applied to the UI element and depends on the conditional 4GL display attributes applied to the element in a form file. If an attribute is applied without the condition, the ClassName if used instead.	SetStyleClassName, GetStyleClassName
Target	String	No information	SetTarget, GetTarget
Comment	String	A character string with some sort of description.	SetComment, GetComment
FieldTable	String	No information	SetFieldTable, GetFieldTable

1.9.4 Extra Methods Description

Table 1.17: Extra methods description

Name	Parameters	Description
CreateClone		
GetContainer		Get link to the parent container.
SetContainer	ui.DistributedObject	Set link to the parent container. 'Container' will be set automatically when added to the parent container.

1.9.5 Static Methods Description

Table 1.18: Static methods description

Name	Parameters	Description
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.10 AppType

1.10.1 Brief description

This entity defines the application type from the point of view of MDI containers. An application can either be normal - not involved in MDI interface, or it can attain its role depending on its function in MDI.

1.10.2 Possible values

Table 1.19: Possible values

Value	Description
Child	An application launched inside an MDI container as one of its child applications.
Normal	A normal application is an independednt application that is launched outside the MDI interface.
Container	An application launched as an MDI container which can house other applications.

1.11 ui.Background

1.11.1 Brief description

This element determines the colour of the background of an element, the background image, if any, and its properties.

1.11.2 Fields Description

Table 1.20: Fields description

Field Name	Type	Description	Field Accessors
BackgroundImage	ui.ResourceId	A background image for the UI element.	SetBackgroundImage, GetBackgroundImage
BackgroundStyle	String catalog for BackgroundStyle	The position of the background image of the UI element.	SetBackgroundStyle, GetBackgroundStyle
FillColor	ui.Color	The color of the background of an element.	SetFillColor, GetFillColor
Location	ui.Location	The location of the UI element specified in pixels.	SetLocation, GetLocation
Size	ui.Size	The size of the UI element in pixels that.	SetSize, GetSize

1.12 ui.BackgroundServerEventHandler

1.12.1 Brief description

EMPTY.

1.12.2 Inheritance Diagram

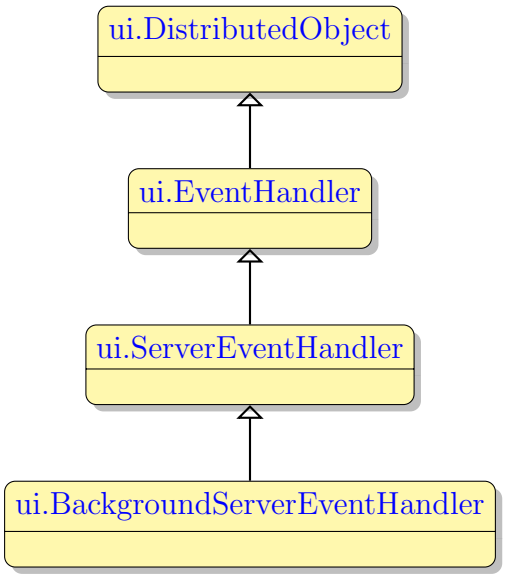

Figure 1.10: Inheritance Diagram of ui.BackgroundServerEventHandler

1.12.3 Fields Description

Table 1.21: Fields description

Field Name	Type	Description	Field Accessors
Delegate	BOOLEAN	No information	SetDelegate, GetDelegate
Selector	String	No information	SetSelector, GetSelector

1.12.4 Static Methods Description

Table 1.22: Static methods description

Name	**Parameters**	**Description**
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.

1.13 BackgroundStyle

1.13.1 Brief description

This element determines the position and arrangement of the background image of the UI element. It is not applicable if the background of an element does not have a background image specified.

1.13.2 Possible values

Table 1.23: Possible values

Value	Description
Default	The window size is the size with which it was opened or which was set after opening by 4GL or graphical theme means.
Normal	The background image is not changed, it retains its size, unless ui.Size is applied, and is placed in the top left colour, if the ui.Location is not set.
Stretched	The background image is stretched to fill whole UI element without preserving the aspect ratio. Its size and location cannot be changed.
Tiled	The background image retains its original size, but it is multiplied and used to cover the whole UI element area in a form of tiles. The size and location of the image cannot be changed.
Centered	The background image retains its original size and is placed in the center of the UI element. Its size and location cannot be changed.
Uniform	The background image is stretched to fill whole UI element while preserving the aspect ratio. Some margin will be added to the image. Its size and location cannot be changed.
UniformToFill	The background image is stretched to fill whole UI element while preserving the aspect ratio. No margin will be added to the image. Its size and location cannot be changed.

1.14 ui.BatchEventHandler

1.14.1 Brief description

This is an event handler which allows a UI element to have more than one event handler assigned to one event.

1.14.2 Inheritance Diagram

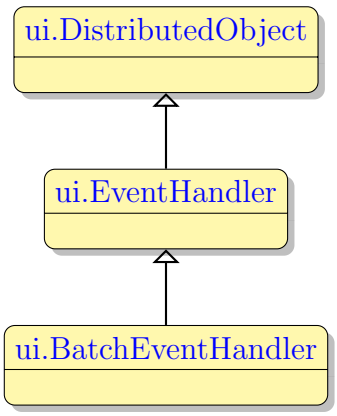

Figure 1.11: Inheritance Diagram of ui.BatchEventHandler

1.14.3 Fields Description

Table 1.24: Fields description

Field Name	Type	Description	Field Accessors
Handlers	list of ui.EventHandler	A set of event handlers assigned to a single event.	SetHandlers, GetHandlers

1.14.4 Static Methods Description

Table 1.25: Static methods description

Name	Parameters	Description

Continued on next page

Table 1.25 – *Continued from previous page*

Name	**Parameters**	**Description**
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.

1.15 ui.BevelBorder

1.15.1 Brief description

This UI element is used to apply a custom bevel border to any concrete UI element. The border can be lowered or raised, its thickness or colour can be changed.

1.15.2 Inheritance Diagram

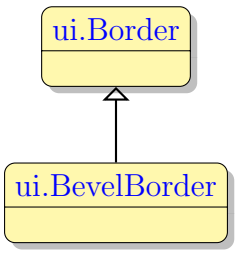

Figure 1.12: Inheritance Diagram of ui.BevelBorder

1.15.3 Fields Description

Table 1.26: Fields description

Field Name	Type	Description	Field Accessors
IsRaised	BOOLEAN	This property specifies whether custom the bevel or etched border should be raised or lowered.	SetIsRaised, GetIsRaised

1.15.4 Static Methods Description

Table 1.27: Static methods description

Name	Parameters	Description

Continued on next page

Table 1.27 – *Continued from previous page*

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.

1.16 ui.BlobViewer

1.16.1 Brief description

This UI element is used to display and edit BYTE or TEXT values e.g a text or a picture.

1.16.2 Inheritance Diagram

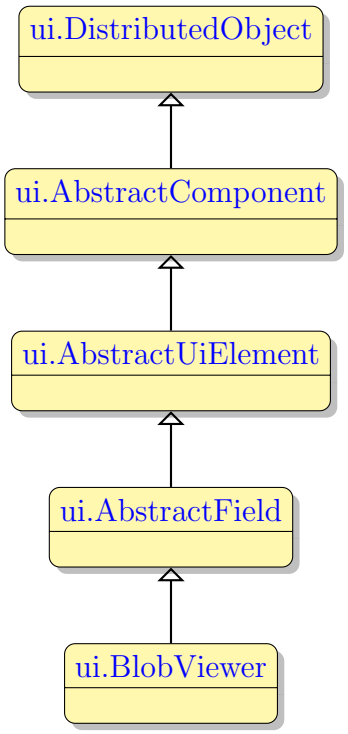

Figure 1.13: Inheritance Diagram of ui.BlobViewer

1.16.3 Fields Description

Table 1.28: Fields description

Field Name	Type	Description	Field Accessors
Editor	String	Specifies the program to be used for opening and editing the BYTE or TEXT value.	SetEditor, GetEditor

Continued on next page

Table 1.28 – *Continued from previous page*

Field Name	Type	Description	Field Accessors
IsTouched	BOOLEAN	It indicates whether the BLOB data in the blob viewed was modified by the user at runtime.	SetIsTouched, GetIsTouched

1.16.4 Static Methods Description

Table 1.29: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.17 ui.BlockingServerEventHandler

1.17.1 Brief description

EMPTY.

1.17.2 Inheritance Diagram

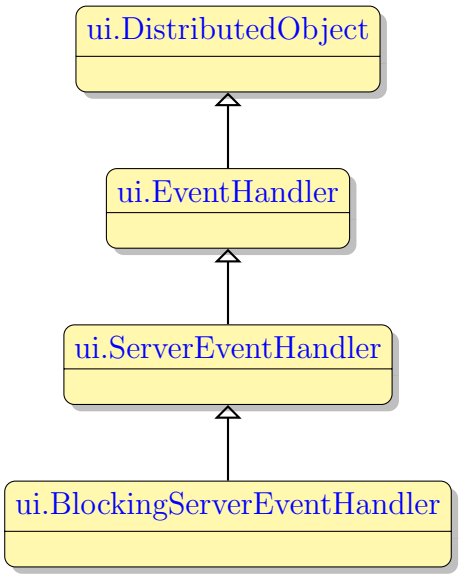

Figure 1.14: Inheritance Diagram of ui.BlockingServerEventHandler

1.17.3 Static Methods Description

Table 1.30: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.

1.18 ui.Border

1.18.1 Brief description

It defines the properties of a custom border around a concrete UI element. The properties border can be applied to one of the three border types: ui.BevelBorder , ui.EtchedBorder , and ui.LineBorder .

1.18.2 Fields Description

Table 1.31: Fields description

Field Name	Type	Description	Field Accessors
BorderBrush	ui.Color	It specifies the colour of the border. Typically applied to ui.LineBorder .	SetBorderBrush, GetBorderBrush
CornerRadius	ui.CornerRadius	The radius of a corner of a custom border around the UI element. It is used to make the border corners rounded.	SetCornerRadius, GetCornerRadius
Thickness	ui.Thickness	It defines the thickness of a border, or the space left empty for a margin or padding in pixels.	SetThickness, GetThickness

1.19 ui.BorderPanel

1.19.1 Brief description

It is a concrete UI element - a container for arranging the layout of other UI elements. Other UI elements can be located either alongside the panel borders or in its center, thus this panel can incorporate up to 5 elements - 1 for each side and 1 in the center. The elements are stretched by default, one element can take up more than one position cell. The position of an element inside the Border panel (that is which of the) is defined by the ui.BorderPanelItemLocation property of this element.

1.19.2 Inheritance Diagram

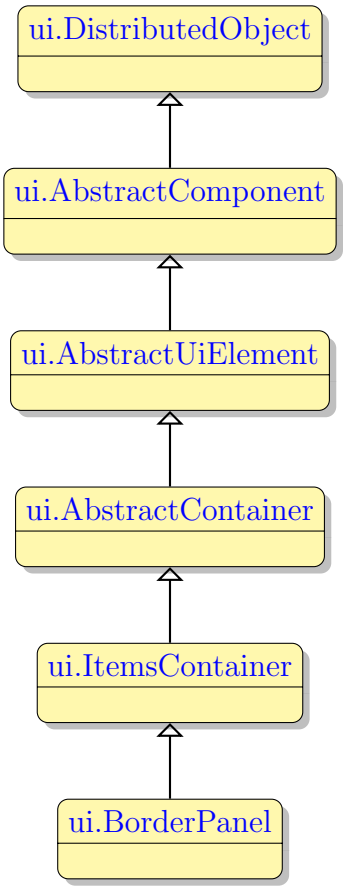

Figure 1.15: Inheritance Diagram of ui.BorderPanel

1.19.3 Static Methods Description

Table 1.32: Static methods description

Name	**Parameters**	**Description**
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.20 BorderPanelItemLocation

1.20.1 Brief description

This property is applicable only if the UI element is located inside the ui.BorderPanel container. It indicates which part of the border panel the element occupies. A Border panel can have 5 positions that elements can take. One element can take several adjacent positions at once. They cannot overlap.

1.20.2 Possible values

Table 1.33: Possible values

Value	Description
Center	The element is located without adjoining to any of the 4 borders of the container, in the space in the middle of the panel.
Left	The element is located adjoined to the left side of the border panel.
Right	The element is located adjoined to the right side of the border panel.
Top	The element is located adjoined to the top border of the border panel.
Bottom	The element is located adjoined to the bottom border of the border panel.

1.21 ui.Browser

1.21.1 Brief description

It is a concrete UI element that encompasses a built-in web browser with a somewhat limited functionality. It is used to display web pages, but can also work as a file explorer, display contents of files (e.g. text or image files), etc.

1.21.2 Inheritance Diagram

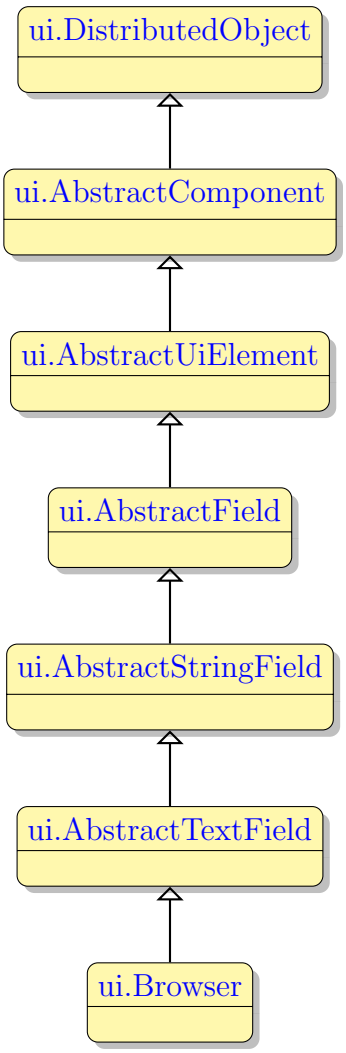

Figure 1.16: Inheritance Diagram of ui.Browser

1.21.3 Static Methods Description

Table 1.34: Static methods description

Name	**Parameters**	**Description**
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.22 ui.Button

1.22.1 Brief description

It is a clickable concrete UI element in a form of a button that is typically used to trigger various events when it is pressed and/or released. It can have a text label or an image on it.

1.22.2 Inheritance Diagram

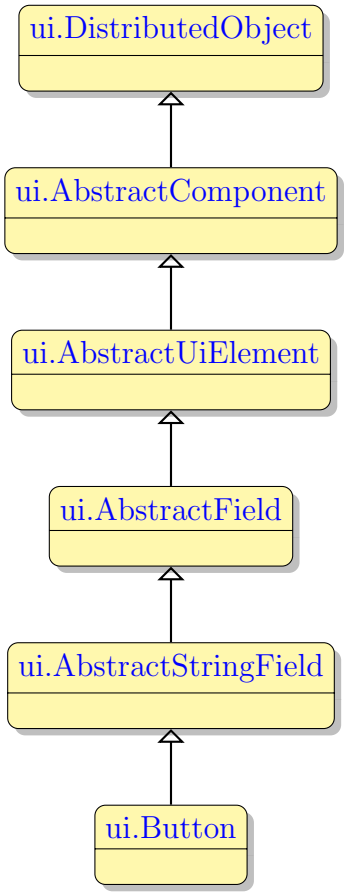

Figure 1.17: Inheritance Diagram of ui.Button

1.22.3 Fields Description

Table 1.35: Fields description

Field Name	Type	Description	Field Accessors
AllowNewlines	BOOLEAN	This property specifies whether the Enter key will be used to move to another form element at runtime (if the value is FALSE), or it will create a newline symbol inside the current field (if the value is TRUE). It is typically applied for the ui.TextArea element.	SetAllowNewlines, GetAllowNewlines
Image	ui.Image	It specifies the icon that should be displayed to the button instead of the inscription. The button is resized to the size of the icon applied.	SetImage, GetImage
IsPressed	BOOLEAN	It tracks the state of the button and its value changes every time the button is pressed or released. It is applicable only to toggle buttons.	SetIsPressed, GetIsPressed
IsToggleButton	BOOLEAN	Determines that the button should be released automatically after it was pressed if set to FALSE (the default value). If set to TRUE - the button is treated as a toggle button which does not get released automatically. Once it was clicked it remains pressed and can only be released with another click.	SetIsToggleButton, GetIsToggleButton
OnInvoke	ui.EventHandler	The event which is triggered when the UI element is invoked. It can be invoked by mouse click, by pressing Enter, or in some cases Space, when the cursor is in the element.	SetOnInvoke, GetOnInvoke

1.22.4 Static Methods Description

Table 1.36: Static methods description

Name	**Parameters**	**Description**
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.23 ui.Calendar

1.23.1 Brief description

It is a concrete UI element that serves for displaying and inputting dates and has a drop-down lookup calendar for graphical date selection.

1.23.2 Inheritance Diagram

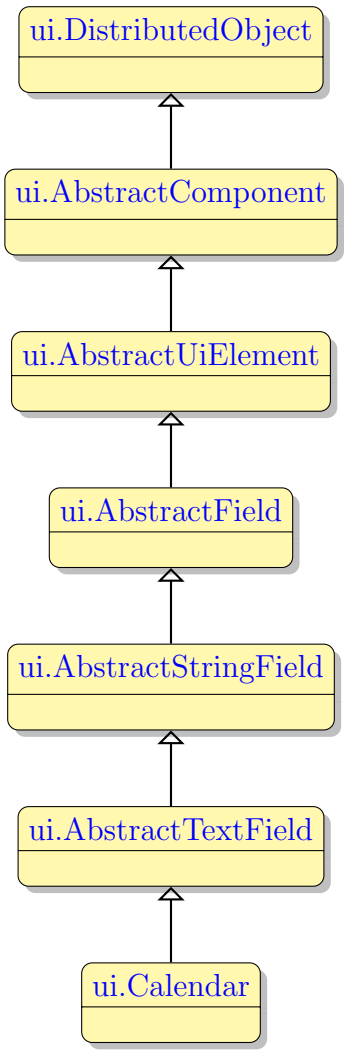

Figure 1.18: Inheritance Diagram of ui.Calendar

1.23.3 Fields Description

Table 1.37: Fields description

Field Name	Type	Description	Field Accessors
DateValue	String	A date value in format "yyyy-mm-dd".	SetDateValue, GetDateValue
HelperText	String	No information	SetHelperText, GetHelperText
LabelText	String	No information	SetLabelText, GetLabelText
OnSelectDate	ui.EventHandler	This event is triggered when the value of the Calendar changes. The value of the element is the value which will be recorded to the underlying variable when the input finishes. (Don't use it. It's for internal usage.)	SetOnSelectDate, GetOnSelectDate
PlaceholderText	String	No information	SetPlaceholderText, GetPlaceholder-Text
SystemDate	BOOLEAN	It's set if 'DateValue is in format "yyyy-mm-dd".	SetSystemDate, GetSystemDate

1.23.4 Static Methods Description

Table 1.38: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.24 ui.Canvas

1.24.1 Brief description

It is a concrete UI element that serves as a container for SVG images and allows interactions with such images.

1.24.2 Inheritance Diagram

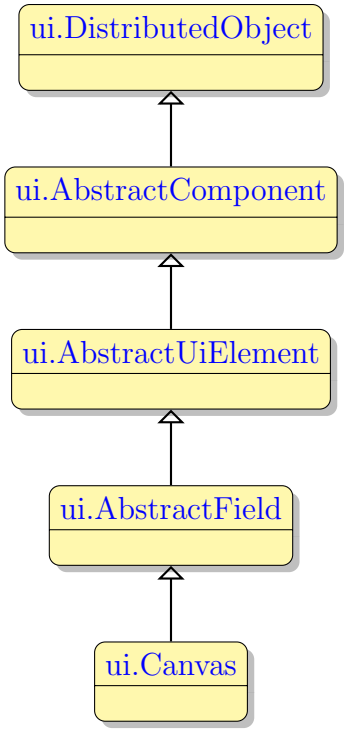

Figure 1.19: Inheritance Diagram of ui.Canvas

1.24.3 Fields Description

Table 1.39: Fields description

Field Name	Type	Description	Field Accessors
Image	ui.Image	It specifies the SVG image that should be displayed to the canvas area.	SetImage, GetImage

1.24.4 Static Methods Description

Table 1.40: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.25 ui.CheckBox

1.25.1 Brief description

It is a concrete UI element that consists of a single check box and a label attached to it. It can be in only one of 2 states at a time - either checked or unchecked. Changing of the state can either change the value that will be written to the underlying variable, or trigger an event handler.

1.25.2 Inheritance Diagram

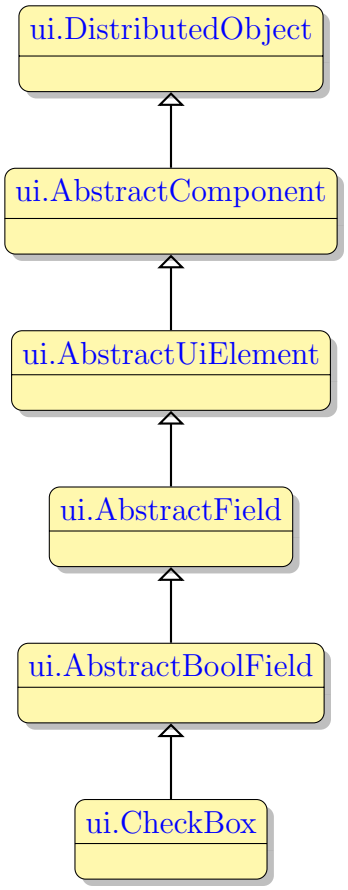

Figure 1.20: Inheritance Diagram of ui.CheckBox

1.25.3 Fields Description

Table 1.41: Fields description

Field Name	Type	Description	Field Accessors
Required	BOOLEAN	No information	SetRequired, GetRequired

1.25.4 Extra Methods Description

Table 1.42: Extra methods description

Name	Parameters	Description
GetCheckedValue		Returns the value of a CheckBox when checked
GetUncheckedValue		Returns the value of a CheckBox when unchecked
SetCheckedValue	Type value	Assigns the value to a CheckBox when checked
SetUncheckedValue	Type value	Assigns the value to a CheckBox when unchecked

1.25.5 Static Methods Description

Table 1.43: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.26 ui.ClientSideExecEventHandler

1.26.1 Brief description

No information

1.26.2 Inheritance Diagram

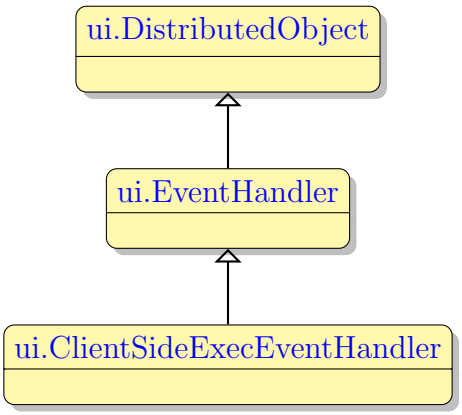

Figure 1.21: Inheritance Diagram of ui.ClientSideExecEventHandler

1.26.3 Fields Description

Table 1.44: Fields description

Field Name	Type	Description	Field Accessors
ExecCommand	String	No information	SetExecCommand, GetExecCommand
ExecParam	String	No information	SetExecParam, GetExecParam

1.26.4 Static Methods Description

Table 1.45: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.

1.27 ui.Color

1.27.1 Brief description

It is the root element to all color properties that can be applied to any UI element.

1.28 ui.ComboBox

1.28.1 Brief description

It is a concrete UI element that has a form of a text field with a drop-down list. It can be restricted to accepting only values from this drop-down list, or it can be set to accept values from the list and the custom values entered by the user. Only one item from the drop-down combobox list can be selected at a time.

1.28.2 Inheritance Diagram

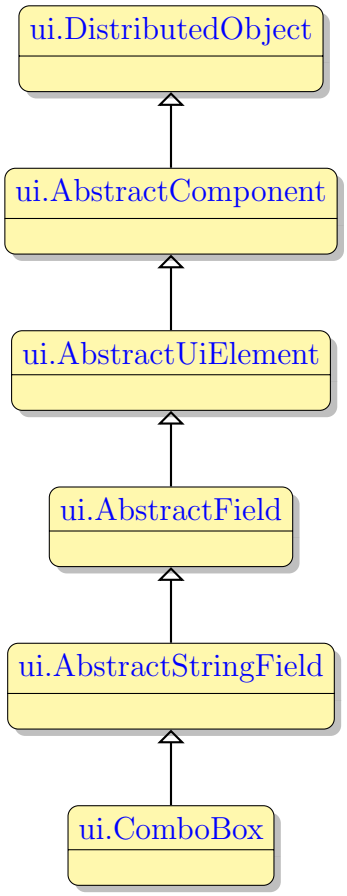

Figure 1.22: Inheritance Diagram of ui.ComboBox

1.28.3 Fields Description

Table 1.46: Fields description

Field Name	Type	Description	Field Accessors
Autonext	BOOLEAN	If enabled, moves the cursor to the next field during input automatically, when the MaxLength of the current field is met.	SetAutonext, GetAutonext
Editable	BOOLEAN	It indicates that the combo box accepts values that are not in its drop-down list.	SetEditable, GetEditable
HelperText	String	No information	SetHelperText, GetHelperText
LabelText	String	No information	SetLabelText, GetLabelText
MaxLength	Int	It specifies the maximum length in bytes allowed for entering into the filed. Its value is normally taken from the data type and size of the variable linked to the field.	SetMaxLength, GetMaxLength
Required	BOOLEAN	No information	SetRequired, GetRequired
SelectedItem	Int	No information	SetSelectedItem, GetSelectedItem
ToCase	String catalog for ToCase	This property specifies the case of a UI element. It can be applied to any UI element that allows entering text from keyboard. By default its value is None, meaning that the case of the letters does not change and remains as they were inputted.	SetToCase, GetToCase

1.28.4 Extra Methods Description

The table below contains brief information regarding the subject. For more detailed description, please, refer to the Function Guide, ui.COMBOBOX Class section, included into the package.

Table 1.47: Extra methods description

Name	Parameters	Description
AddItem	Type name, String text	Adds a single ComboBoxItem with "text" for the Text property and "name" for the Value property
Clear		Removes all the ComboBox items from the dropdown list of the ComboBox specified
GetColumnName		Returns the column prefix of the specified form field
GetIndexOf	Type name	Returns the position of the first ComboBoxItem in the list matching the value of the Value property and an argument passed to the method. If Value is not set, then the Text property value and an argument matching takes place
GetItemCount		Returns the total number of the items currently available from the ComboBox dropdown list
GetItemName	Int index	Returns Value of the ComboboxItem specified by its index position as an argument
GetItemText	Int index	Returns (String) Text of ComboBoxItem specified by its index position as an argument
GetTableName		Returns the table prefix of the specified form field
GetTag		Returns the value specified in the TAG attribute in a form file for the referenced ComboBox widget (ComboBox identifier)
GetTextOf	Type name	Returns (String) Text of the first ComboBox item from the list matching the Value property and an argument passed to the method. If Value is not set, then the Text property value and an argument matching takes place
RemoveItem	Type name	Removes the first ComboBoxItem matching the Value property and an argument passed to the method. If Value is not set, then the Text property value and an argument matching takes place
Sort	Int direction	Sorts ComboboxItems specified by direction as an argument (0 = DES, 1 = ASC)

1.28.5 Static Methods Description

Table 1.48: Static methods description

Name	**Parameters**	**Description**
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.29 CompatibilityMode

1.29.1 Brief description

No information

1.29.2 Possible values

Table 1.49: Possible values

Value	Description
Lycia	Not described yet
Informix4GL	Not described yet
GBDS	Not described yet

1.30 ui.ComponentProperty

1.30.1 Brief description

This is the property of a ui.WebComponent UI element. Each property is defined by the HTML file that describes the web component.

—This type is represented as 4GL record—

1.30.2 Fields Description

Table 1.50: Fields description

Field Name	Type	Description	Field Accessors
PName	String	It specifies the name of a web component property.	
PValue	String	It specifies the value of a web component property.	

1.31 ui.CoordPanel

1.31.1 Brief description

This is a container the location of the elements inside which is determined by the coordinates of the component. The coordinates are stored in pixels and specify the ui.Location on the coord panel where the top left corner of the child element is placed.

1.31.2 Inheritance Diagram

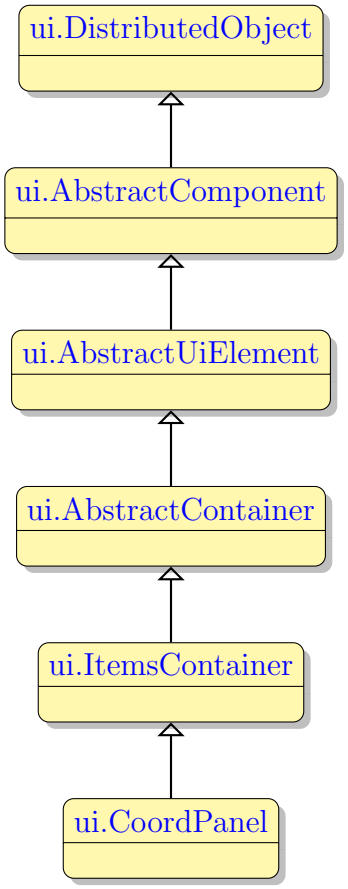

Figure 1.23: Inheritance Diagram of ui.CoordPanel

1.31.3 Static Methods Description

Table 1.51: Static methods description

Name	**Parameters**	**Description**
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.32 ui.CornerRadius

1.32.1 Brief description

This enum specifies the radius of a corner of a custom border around the UI element. It is used to make the border corners rounded. It can be applied only to ui.LineBorder border type. All four corners can have different corner radius.

—This type is represented as 4GL record—

1.32.2 Fields Description

Table 1.52: Fields description

Field Name	Type	Description	Field Accessors
BottomLeft	Float	The bottom left corner of the border frame.	
BottomRight	Float	The bottom right corner of the border frame.	
TopLeft	Float	The top left corner of the border frame.	
TopRight	Float	The top right corner of the border frame.	

1.33 Cursor

1.33.1 Brief description

It defines the animation the mouse cursor should have when hovering over the UI element for which this enum is specified. The cursor animation at runtime is selected on the basis of the cursors available for the system or for the browser, if the web client is used.

1.33.2 Possible values

Table 1.53: Possible values

Value	Description
Arrow	The default arrow cursor.
Cross	The cursor in a form of a cross.
IBeam	The cursor in a form of a vertical line.
SizeAll	The cursor in a form of a cross with arrows at all 4 ends.
SizeNESW	The cursor in a form of a diagonal line in direction from top right to bottom left with arrows on both ends .
SizeNS	The cursor in a form of a vertical line with arrows on both ends .
SizeNWSE	The cursor in a form of a diagonal line in direction from top left to bottom right with arrows on both ends .
SizeWE	The cursor in a form of a horizontal line with arrows on both ends .
UpArrow	The cursor in a form of a vertical line with an arrow pointing upwards .
WaitCursor	The default waiting cursor of the system (e.g. in Windows XP - glass clock, in Windows 7 - a blue ring).
Help	The default help cursor of the system (normally in a form of a question mark).
HSplit	The default cursor that appears when the mouse is positioned over a horizontal splitter bar.
VSplit	The default cursor that appears when the mouse is positioned over a vertical splitter bar.
Hand	The default hand cursor.

1.34 ui.CustomizedColor

1.34.1 Brief description

This enum defines a custom color in the RGB encoding plus the transparency.

1.34.2 Inheritance Diagram

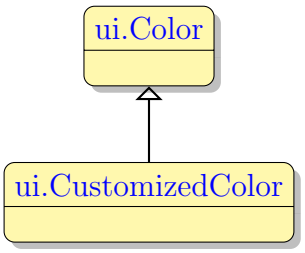

Figure 1.24: Inheritance Diagram of ui.CustomizedColor

1.34.3 Fields Description

Table 1.54: Fields description

Field Name	Type	Description	Field Accessors
Alpha	Int	The value of the transparency applied to the color. 0 - completely transparent. 255 - completely solid color.	SetAlpha, GetAlpha
BlueColor	Int	The value of the blue colour in the RGB color model (0-255).	SetBlueColor, GetBlueColor
GreenColor	Int	The value of the green colour in the RGB color model (0-255).	SetGreenColor, GetGreenColor
RedColor	Int	The value of the red colour in the RGB color model (0-255).	SetRedColor, GetRedColor

1.34.4 Static Methods Description

Table 1.55: Static methods description

Name	**Parameters**	**Description**
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.

1.35 ui.DefaultBorder

1.35.1 Brief description

No information

1.35.2 Inheritance Diagram

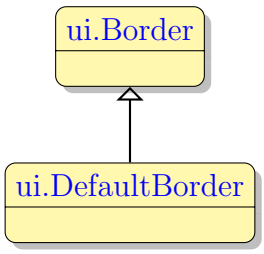

Figure 1.25: Inheritance Diagram of ui.DefaultBorder

1.35.3 Static Methods Description

Table 1.56: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.

1.36 ui.DefaultColor

1.36.1 Brief description

No information

1.36.2 Inheritance Diagram

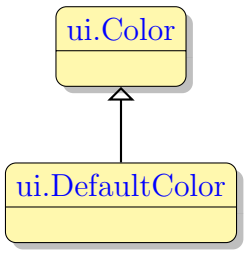

Figure 1.26: Inheritance Diagram of ui.DefaultColor

1.36.3 Static Methods Description

Table 1.57: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.

1.37 Direction

1.37.1 Brief description

This enum defines the direction of the text: left to right or right to left.

1.37.2 Possible values

Table 1.58: Possible values

Value	Description
LTR	The text is written and displayed in the direction from left to right.
RTL	The text is written and displayed in the direction from right to left.

1.38 ui.DistributedObject

1.38.1 Brief description

This is the root of the UI element hierarchy.

1.38.2 Extra Methods Description

Table 1.59: Extra methods description

Name	Parameters	Description

1.39 ui.ElementContainer

1.39.1 Brief description

This UI element unites all the containers which can contain exactly one element. The containers that derive from ElementContainer UI element can be logically opposed to containers derived from ui.ItemsContainer UI element that can contain any number of elements of any type. The elements that inherit their properties from ElementContainer can encompass such elements as ring menu area or any other container. They can also contain an element belonging to ui.AbstractFiled class, but only one such element.

1.39.2 Inheritance Diagram

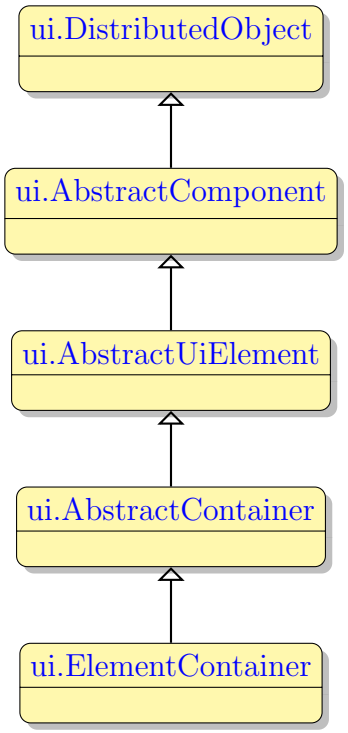

Figure 1.27: Inheritance Diagram of ui.ElementContainer

1.39.3 Fields Description

Table 1.60: Fields description

Field Name	Type	Description	Field Accessors
Content	ui.AbstractUiElement	It specifies the UI element that is located inside the ElementContainer.	SetContent, GetContent

1.39.4 Static Methods Description

Table 1.61: Static methods description

Name	Parameters	Description
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.40 ElementRole

1.40.1 Brief description

The role the UI element is executing at the moment. It depends on the 4GL code, thus a character string can be either a message, an error, a displayed string, etc.

1.40.2 Possible values

Table 1.62: Possible values

Value	Description
None	The UI element is not currently executing any of the predefined roles.
Query	The UI element takes part in a CONSTRUCT statement.
Display	The UI element takes part in a DISPLAY statement.
Input	The UI element takes part in a INPUT statement.
Message	The UI element is a result of the MESSAGE statement.
Error	The UI element is a result of the ERROR statement.
Comment	The UI element is a result of the COMMENT property of a widget is displayed.
Prompt	The UI element takes part in a PROMPT statement.
Form	The UI element is a form.
RingMenu	Not described yet

1.41 ui.EtchedBorder

1.41.1 Brief description

It sets a custom etched border around the UI element. The border can be raised and lowered, its colour can be changed.

1.41.2 Inheritance Diagram

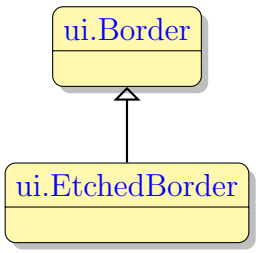

Figure 1.28: Inheritance Diagram of ui.EtchedBorder

1.41.3 Fields Description

Table 1.63: Fields description

Field Name	Type	Description	Field Accessors
IsRaised	BOOLEAN	This property specifies whether custom the bevel or etched border should be raised or lowered.	SetIsRaised, GetIsRaised

1.41.4 Static Methods Description

Table 1.64: Static methods description

Name	Parameters	Description

Continued on next page

Table 1.64 – *Continued from previous page*

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.

1.42 ui.EventHandler

1.42.1 Brief description

This is common class for all the specific event handler types.

1.42.2 Inheritance Diagram

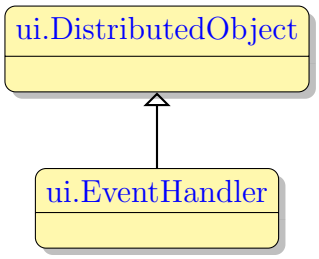

Figure 1.29: Inheritance Diagram of ui.EventHandler

1.43 ui.EventInfo

1.43.1 Brief description

It is an abstract UI entity which is the root class for the ui.KeyEvent . It is used to send the information to the server about the event triggered on the client side.

1.43.2 Fields Description

Table 1.65: Fields description

Field Name	Type	Description	Field Accessors
OriginalSource	ui.AbstractUiElement	No information	SetOriginalSource, GetOriginal-Source
Source	ui.AbstractUiElement	The information which the EventInfo sends to the server. It contains the information which element of the form triggered the event and other useful information about the event.	SetSource, Get-Source

1.44 ui.Font

1.44.1 Brief description

The font to be used for any text that is a part of the UI element - either label ot inputted text.

—This type is represented as 4GL record—

1.44.2 Fields Description

Table 1.66: Fields description

Field Name	Type	Description	Field Accessors
Family	list of String	This is the name of the font. E.g. Arial or Tahoma.	
Bold	BOOLEAN	It indicates whether the text should be bold.	
Italic	BOOLEAN	It indicates whether the text should be in italics.	
Underline	BOOLEAN	It indicates whether the text should be underlined.	
FontSize	Int	It specifies the font size.	

1.45 ui.FunctionFieldAbs

1.45.1 Brief description

This UI entity is a function field that is a combination of a text field and a button attached to it. It serves mainly for grouping the button element and the text field element in one object. The properties of the field and button are independent.

1.45.2 Inheritance Diagram

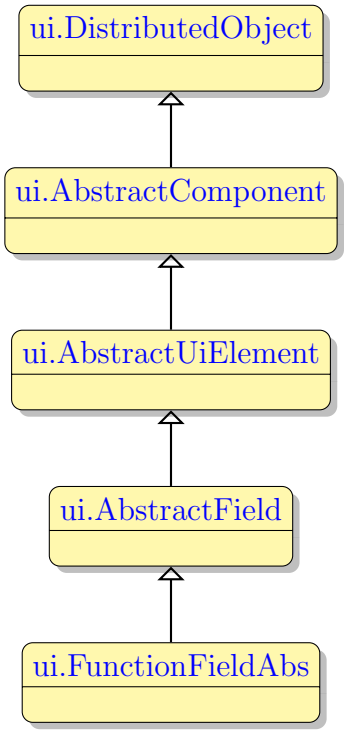

Figure 1.30: Inheritance Diagram of ui.FunctionFieldAbs

1.45.3 Fields Description

Table 1.67: Fields description

Field Name	Type	Description	Field Accessors
FunctionFieldButton	ui.Button	It is the button widget that is included into a function field.	SetFunctionFieldButton, GetFunction-FieldButton

Continued on next page

Table 1.67 – *Continued from previous page*

Field Name	Type	Description	Field Accessors
FunctionFieldText	ui.TextField	It is the text field widget that is included into a function field.	SetFunctionFieldText, GetFunctionFieldText

1.45.4 Static Methods Description

Table 1.68: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.46 ui.GridColumnDefinition

1.46.1 Brief description

This UI element defines the properties of a columns in a ui.GridPanel container and their properties.

—This type is represented as 4GL record—

1.46.2 Fields Description

Table 1.69: Fields description

Field Name	Type	Description	Field Accessors
GridLengthValue	String	The width of the grid column or row in the units specified by the length type.	
GridMinLength	String	This the minimum size of a grid column or row to which it can be resized.	
GridMaxLength	String	This the maximum size of a grid column or row to which it can be resized.	

1.47 ui.GridItemLocation

1.47.1 Brief description

This property defines the position of an element located within a ui.GridPanel in relation to this grid panel. The grid panel is divided into cells which are created by means of grid rows and columns. Each element placed inside the grid panel must occupy at least one cell. It can occupy more than one cell, but two elements cannot occupy one and the same cell. Each element inside a grid panel is located inside the cells, it cannot occupy half of a cell.

—This type is represented as 4GL record—

1.47.2 Fields Description

Table 1.70: Fields description

Field Name	Type	Description	Field Accessors
GridX	Int	It is the number of column in which the grid cell with the UI element is located. It is treated as the X coordinate of an element within the grid panel.	
GridY	Int	It is the number of row in which the grid cell with the UI element is located. It is treated as the Y coordinate of an element within the grid panel.	
GridWidth	Int	It specifies the number of horizontal cells that the element occupies. It cannot be less than 1.	
GridHeight	Int	It specifies the number of vertical cells that the element occupies. It cannot be less than 1.	

1.48 ui.GridLength

1.48.1 Brief description

This UI element defines the length of the grid columns and width of the rows. Thus it can define the size of the ui.GridPanel cells. The size can be absolute or relative. It can also define the length of the table columns.

—This type is represented as 4GL record—

1.48.2 Fields Description

Table 1.71: Fields description

Field Name	Type	Description	Field Accessors
GridLengthValue	String	The width of the grid column or row in the units specified by the length type.	
GridMinLength	String	This the minimum size of a grid column or row to which it can be resized.	
GridMaxLength	String	This the maximum size of a grid column or row to which it can be resized.	

1.49 ui.GridPanel

1.49.1 Brief description

It is a container that is used to arrange the layout of other UI elements placed inside. The elements inside the grid panel are placed inside the grid cells that are formed by the grid rows and columns. Each element must occupy at least 1 grid cell, two elements cannot occupy one and the same grid cell. The number of the grid cells can be defined by the user.

1.49.2 Inheritance Diagram

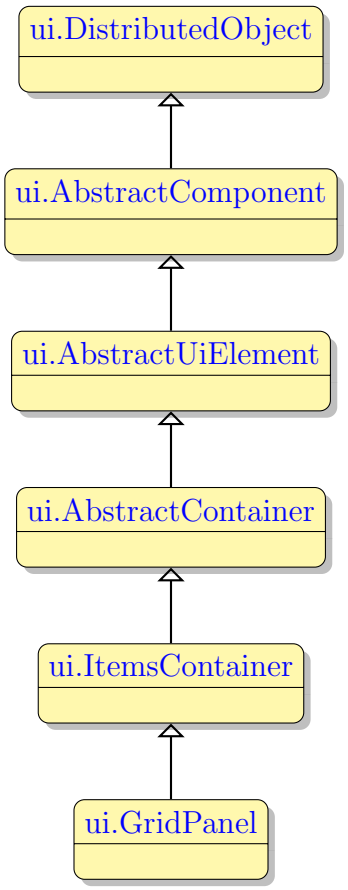

Figure 1.31: Inheritance Diagram of ui.GridPanel

1.49.3 Fields Description

Table 1.72: Fields description

Field Name	Type	Description	Field Accessors
GridColumnDefinitions	list of ui.GridColumnDefinition	This UI element defines the number of rows in a grid panel container and their properties.	SetGridColumnDefinition GetGridColumn-Definitions
GridRowDefinitions	list of ui.GridRowDefinition	This UI element defines the number of rows in a grid panel container and their properties.	SetGridRowDefinitions, GetGridRowDefi-nitions

1.49.4 Static Methods Description

Table 1.73: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.50 ui.GridRowDefinition

1.50.1 Brief description

This UI element defines the properties of a row in a ui.GridPanel container.

—This type is represented as 4GL record—

1.50.2 Fields Description

Table 1.74: Fields description

Field Name	Type	Description	Field Accessors
GridLengthValue	String	The width of the grid column or row in the units specified by the length type.	
GridMinLength	String	This the minimum size of a grid column or row to which it can be resized.	
GridMaxLength	String	This the maximum size of a grid column or row to which it can be resized.	

1.51 ui.GroupBox

1.51.1 Brief description

It is a container that groups the UI elements inside a visible border with an optional title at the top. It can contain only one other UI element. It can be another container or a form widget. Thus though it can encompass UI elements of the ui.AbstractField group, having only one element of this group in a container makes little sense. So it should include one of the other containers first.

1.51.2 Inheritance Diagram

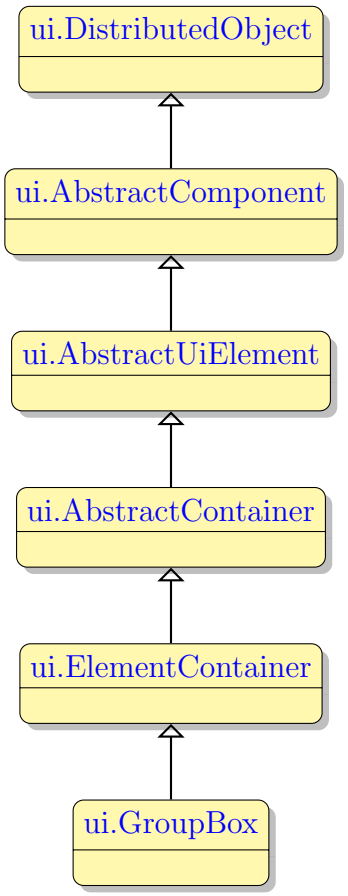

Figure 1.32: Inheritance Diagram of ui.GroupBox

1.51.3 Fields Description

Table 1.75: Fields description

Field Name	Type	Description	Field Accessors
Title	String	This is the inscription attached to the UI element. Usually this is the text of all sorts of labels.	SetTitle, GetTitle
TitleJustification	String catalog for TitleJustification	It specifies the horizontal alignment of the text of the title.	SetTitleJustification, GetTitleJustification

1.51.4 Static Methods Description

Table 1.76: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.52 HorizontalAlignment

1.52.1 Brief description

This enum specifies the horizontal alignment of a UI element inside a container. It is applicable to UI elements inside any container except coord panel. It defines to which border of the container (or container cell) - left or right - the element must adjoin.

1.52.2 Possible values

Table 1.77: Possible values

Value	Description
Default	The window size is the size with which it was opened or which was set after opening by 4GL or graphical theme means.
Stretch	The UI element will be stretched to fit the container (or container cell) without preserving the aspect ratio.
Left	The UI element will be aligned to the left side of the container (or container cell).
Center	The UI element will be equidistant from both sides.
Right	The UI element will be aligned to the right side of the container (or container cell).

1.53 HorizontalTextAlignment

1.53.1 Brief description

1.53.2 Possible values

Table 1.78: Possible values

Value	Description
Default	The window size is the size with which it was opened or which was set after opening by 4GL or graphical theme means.
Left	The UI element will be aligned to the left side of the container (or container cell).
Center	The UI element will be equidistant from both sides.
Right	The UI element will be aligned to the right side of the container (or container cell).

1.54 ui.Image

1.54.1 Brief description

It is an image that can be applied to other UI elements, e.g. to a button.

1.54.2 Fields Description

Table 1.79: Fields description

Field Name	Type	Description	Field Accessors
ImagePosition	String catalog for ImagePosition	No information	SetImagePosition, GetImagePosition
ImageScaling	String catalog for ImageScaling	It specifies whether the image should be scaled to fit the UI element it is applied to.	SetImageScaling, GetImageScaling
ImageUrl	ui.ResourceId	It specifies the URI of an image file. The image should be located on the application server and inside the folder into which the application is deployed. The URL should begin with: qx://application/... .	SetImageUrl, GetImageUrl
Size	ui.Size	The size of the UI element in pixels that.	SetSize, GetSize

1.55 ImagePosition

1.55.1 Brief description

No information

1.55.2 Possible values

Table 1.80: Possible values

Value	Description
Left	The UI element will be aligned to the left side of the container (or container cell).
Right	The UI element will be aligned to the right side of the container (or container cell).
Top	The UI element will be aligned to the top of the container (or container cell).
Bottom	The UI element will be aligned to the bottom of the container (or container cell).

1.56 ImageScaling

1.56.1 Brief description

It specifies whether the image should be scaled (resized) to fit the UI element it is applied to. The scaling preserves the aspect ratio of an image, so in case the image is scaled by the larger side of the UI element, a part of it might be cut off.

1.56.2 Possible values

Table 1.81: Possible values

Value	Description
None	The property is not applied and the default behaviour is used.
Horizontal	The image will be scaled to fit the width of the UI element.
Vertical	The image will be scaled to fit the height of the UI element.
Both	The image will be scaled to fit the smallest dimension (either height or width) of the UI element.

1.57 ui.ItemsContainer

1.57.1 Brief description

The containers that can contain any number of UI elements inherit their properties from the ItemsContainer UI element. These are the containers that can contain any number of form fields and other containers, as opposed to the containers belonging to ui.ElementContainer class.

1.57.2 Inheritance Diagram

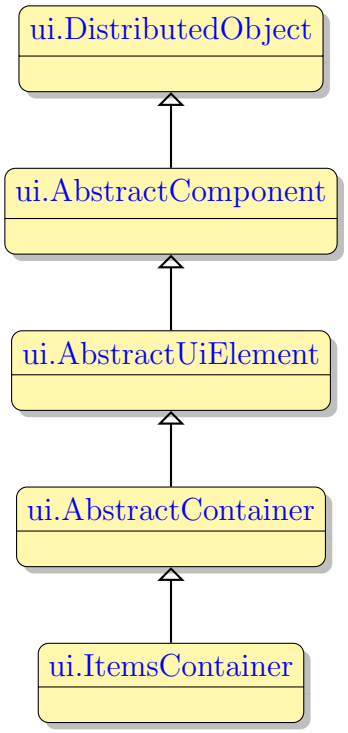

Figure 1.33: Inheritance Diagram of ui.ItemsContainer

1.57.3 Fields Description

Table 1.82: Fields description

Field Name	Type	Description	Field Accessors
Items	list of ui.AbstractUiElement	A set of UI elements that are placed inside the container.	SetItems, GetItems

1.57.4 Static Methods Description

Table 1.83: Static methods description

Name	Parameters	Description
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.58 ui.KeyEvent

1.58.1 Brief description

It is an event that is triggered when the specified key on the keyboard is pressed. This event is sent to the Application server on the keypress.

1.58.2 Inheritance Diagram

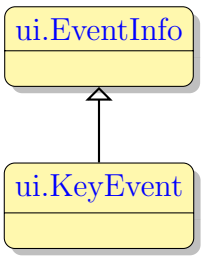

Figure 1.34: Inheritance Diagram of ui.KeyEvent

1.58.3 Fields Description

Table 1.84: Fields description

Field Name	Type	Description	Field Accessors
AltModifier	BOOLEAN	It indicates whether the Alt key should be held down when the key is pressed.	SetAltModifier, GetAltModifier
ControlModifier	BOOLEAN	It indicates whether the Ctrl key should be held down when the key is pressed.	SetControlModifier, GetControlModifier
KeyValue	String	The name of the key pressed. The key name is the name written on the key, e.g. F12 or A.	SetKeyValue, GetKeyValue
ShiftModifier	BOOLEAN	It indicates whether the Shift key should be held down when the key is pressed.	SetShiftModifier, GetShiftModifier

Continued on next page

Table 1.84 – *Continued from previous page*

Field Name	Type	Description	Field Accessors
VirtualKeyValue	String	The code of the key pressed. E.g. there are two keys with key name 5 on a standard keyboard, one of them on the numpad. Their codes will be different, though the key names are the same.	SetVirtualKeyValue, GetVirtualKeyValue

1.58.4 Static Methods Description

Table 1.85: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.

1.59 ui.Label

1.59.1 Brief description

It is a concrete UI element that has the form of a label with some text, image or both. The label is not an interactive widget and cannot be used for input, but the information displayed by it can be changed dynamically.

1.59.2 Inheritance Diagram

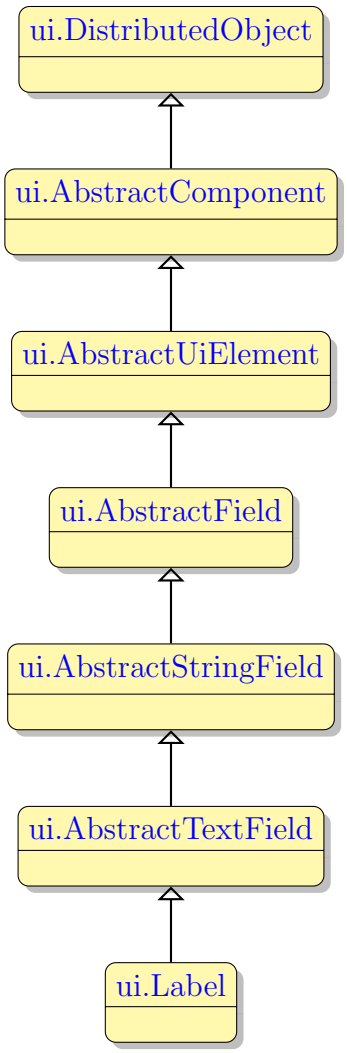

Figure 1.35: Inheritance Diagram of ui.Label

1.59.3 Fields Description

Table 1.86: Fields description

Field Name	Type	Description	Field Accessors
AllowNewlines	BOOLEAN	This property specifies whether the Enter key will be used to move to another form element at runtime (if the value is FALSE), or it will create a newline symbol inside the current field (if the value is TRUE). It is typically applied for the ui.TextArea element.	SetAllowNewlines, GetAllowNewlines
Image	ui.Image	The image that is displayed to a label.	SetImage, GetImage
IsDynamic	BOOLEAN	It specifies whether the information displayed by the label can be changed dynamically by means of the DISPLAY TO statement.	SetIsDynamic, GetIsDynamic
LinkedTo	ui.LinkedTo	No information	SetLinkedTo, GetLinkedTo
OnInvoke	ui.EventHandler	The event which is triggered when the UI element is invoked. It can be invoked by mouse click, by pressing Enter, or in some cases Space, when the cursor is in the element.	SetOnInvoke, GetOnInvoke

1.59.4 Static Methods Description

Table 1.87: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.60 ui.LineBorder

1.60.1 Brief description

This UI element is used to apply a custom line border to any concrete UI element. A line border is just a line of the defined thickness and colour that surrounds the element. The line border allows the ui.CornerRadius to be set to round the corners.

1.60.2 Inheritance Diagram

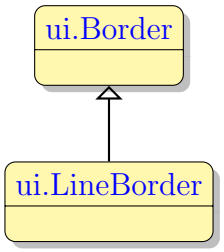

Figure 1.36: Inheritance Diagram of ui.LineBorder

1.60.3 Static Methods Description

Table 1.88: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.

1.61 ui.LinkedTo

1.61.1 Brief description

No information

1.61.2 Fields Description

Table 1.89: Fields description

Field Name	Type	Description	Field Accessors
LinkedToElement	ui.AbstractUiElement	No information	SetLinkedToElement, GetLinkedToElement
TableRowPos	ui.TableRowPos	No information	SetTableRowPos, GetTableRowPos

1.62 ui.ListBox

1.62.1 Brief description

It is a concrete UI element that has the form of a form field with a list of values inside available for selection. It does not accept values entered from the keyboard, but can participate in the input and records into the underlying variable the value that was selected from the list.

1.62.2 Inheritance Diagram

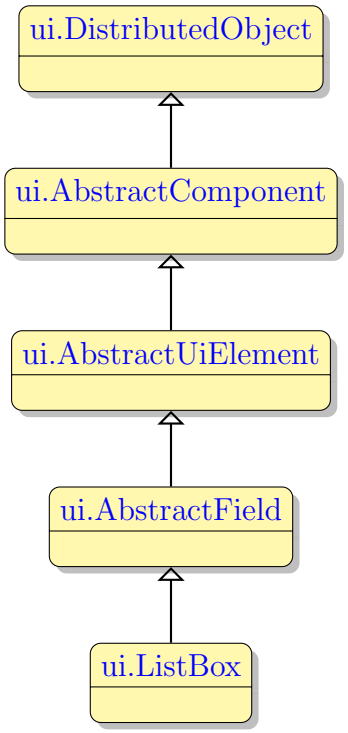

Figure 1.37: Inheritance Diagram of ui.ListBox

1.62.3 Fields Description

Table 1.90: Fields description

Field Name	Type	Description	Field Accessors

Continued on next page

Table 1.90 – *Continued from previous page*

Field Name	Type	Description	Field Accessors
EnableMultiSelection	BOOLEAN	It specifies how many items can be simultaneously selected inside a list box widget. If set to FALSE, only one item cal be selected at a time.	SetEnableMultiSelection, GetEnableMulti-Selection
HelperText	String	No information	SetHelperText, GetHelperText
ListBoxValues	list of String	No information	SetListBoxValues, GetListBoxVal-ues
SelectedItems	String	No information	SetSelectedItems, GetSelectedItems

1.62.4 Extra Methods Description

Table 1.91: Extra methods description

Name	Parameters	Description
AddItem	Type name, String text	Adds ListBoxItem with "text" for the Text property and "name" for the Value property
Clear		Removes all the items of the ListBox specified
GetColumnName		Returns (String) an Identifier of the specified form field
GetIndexOf	Type name	Returns (Int) a position of the first ComboboxItem in the list matching the value of the Value property and an argument passed to the method. If Value is not set, then the Text property value and an argument matching takes place
GetItemCount		Returns (Int) the total number of the ListBox items
GetItemName	Int index	Returns (Type) Value of the ListBoxItem specified by its index position as an argument
GetItemText	Int index	Returns (String) Text of ListBoxItem specified by its index position as an argument
GetTableName		Returns (String) the table prefix of the specified form field
GetTag		Returns (String) the value specified in the TAG at-tribute in a form file for the referenced ListBox widget (ListBox identifier)

Continued on next page

Table 1.91 – *Continued from previous page*

Name	Parameters	Description
GetTextOf	Type name	Returns (String) Text of the first ListBoxItem from the list matching the Value property and an argument passed to the method If Value is not set, then the Text property value and an argument matching takes place
RemoveItem	Type name	Removes the first ListBoxItem matching the Value property and an argument passed to the method. If Value is not set, then the Text property value and an argument matching takes place
Sort	Int direction	Sorts ListBoxItems specified by direction as an argument (0 = DES, 1 = ASC)

1.62.5 Static Methods Description

Table 1.92: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.63 ui.Locale

1.63.1 Brief description

It specifies a custom locate of a UI element that can be different from the default application locale. It can mainly be used for to make a form fir the requirements of several locales at once.

1.63.2 Fields Description

Table 1.93: Fields description

Field Name	Type	Description	Field Accessors
Country	String	The territory where the specified locale language is used. E.g. CA - for French language in Canada.	SetCountry, GetCountry
Direction	String catalog for Direction	The direction of the text: from left to right or from right to left.	SetDirection, GetDirection
Language	String	The language of the locale, e.g. FR for French.	SetLanguage, GetLanguage
Variant	String	The code set of the selected locale. E.g. ISO-8859-1 or UTF-8.	SetVariant, GetVariant

1.64 ui.Localization

1.64.1 Brief description

No information

—This type is represented as 4GL record—

1.64.2 Fields Description

Table 1.94: Fields description

Field Name	Type	Description	Field Accessors
Text	String	A character string.	
Translate	String	No information	

1.65 ui.Location

1.65.1 Brief description

This is the coordinates of the position of a UI element inside a coordinate panel in pixels.

—This type is represented as 4GL record—

1.65.2 Fields Description

Table 1.95: Fields description

Field Name	Type	Description	Field Accessors
XCoord	String	The coordinate of the top left corner of the element on X axis of the coord panel.	
YCoord	String	The coordinate of the top left corner of the element on Y axis of the coord panel.	

1.66 ui.MenuBar

1.66.1 Brief description

This is the area for the top menu (is not applied to ring menus). It includes menu options and menu option groups.

1.66.2 Inheritance Diagram

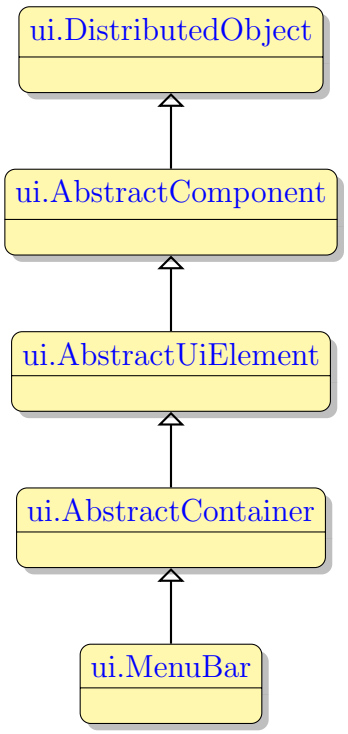

Figure 1.38: Inheritance Diagram of ui.MenuBar

1.66.3 Fields Description

Table 1.96: Fields description

Field Name	Type	Description	Field Accessors
Close	ui.EventHandler	This event is triggered when the close button on the title bar of a window is pressed.	SetClose, GetClose

Continued on next page

Table 1.96 – *Continued from previous page*

Field Name	Type	Description	Field Accessors
ImageId	ui.ResourceId	A reference to an image file.	SetImageId, GetImageId
MenuItems	list of ui.MenuItem	A set of menu options belonging to the same menu.	SetMenuItems, GetMenuItems
RingMenuStyle	String catalog for RingMenuStyle	No information	SetRingMenuStyle, GetRing-MenuStyle
SelectedMenuItem	ui.MenuItem	It identifies one of the menu options that currently has the focus.	SetSelectedMenuItem, GetSelectedMe-nuItem

1.66.4 Static Methods Description

Table 1.97: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.67 ui.MenuCommand

1.67.1 Brief description

This is the menu option that can be invoked by the user. It has a label and/or icon and an even attached.

1.67.2 Inheritance Diagram

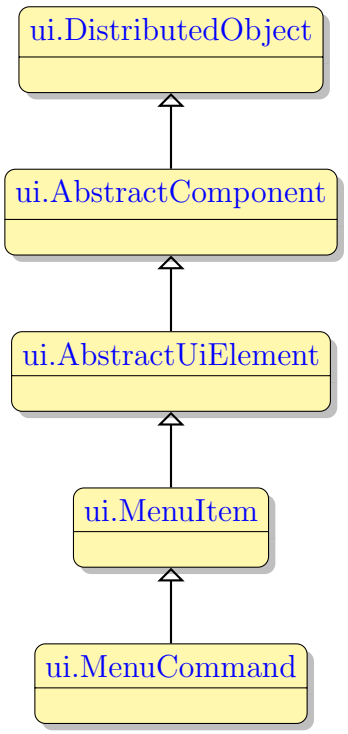

Figure 1.39: Inheritance Diagram of ui.MenuCommand

1.67.3 Fields Description

Table 1.98: Fields description

Field Name	Type	Description	Field Accessors
ImageId	ui.ResourceId	The image that will be used as the icon on the menu option button.	SetImageId, GetImageId

Continued on next page

Table 1.98 – *Continued from previous page*

Field Name	Type	Description	Field Accessors
IsChecked	BOOLEAN	The UI element that has such field can be either in checked state (TRUE) or unchecked state (FALSE). UI elements like check boxes or radio buttons typically contain such field. Every time the element is clicked, the state is flipped.	SetIsChecked, GetIsChecked
OnInvoke	ui.EventHandler	The event which is triggered when the UI element is invoked. It can be invoked by mouse click, by pressing Enter, or in some cases Space, when the cursor is in the element.	SetOnInvoke, GetOnInvoke
ShortCut	String	The name of a key that can be used as a shortcut to invoke the menu option. It just adds a label with the key name to the right end of the menu option label. To actually enable the key as a shortcut key one should add it to Accelerators or assign the KeyEvent to the OnInvoke event.	SetShortCut, GetShortCut
Text	String	This is the label of the menu option.	SetText, GetText

1.67.4 Static Methods Description

Table 1.99: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.68 ui.MenuGroup

1.68.1 Brief description

It is a group that unites several menu options and possibly menu separators. It offers a drop-down menu containing these options and separators, when the mouse cursor hovers over its label.

1.68.2 Inheritance Diagram

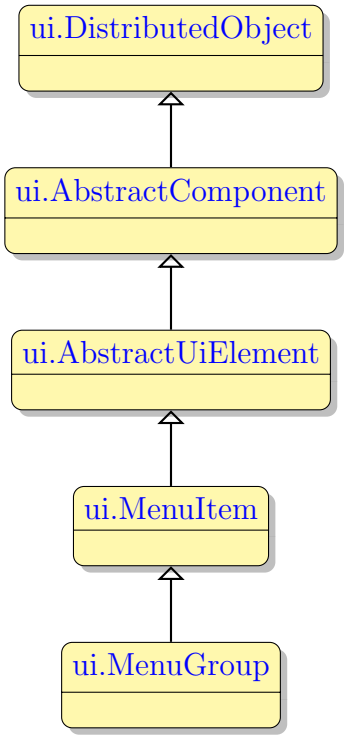

Figure 1.40: Inheritance Diagram of ui.MenuGroup

1.68.3 Fields Description

Table 1.100: Fields description

Field Name	Type	Description	Field Accessors
ImageId	ui.ResourceId	A reference to an image file.	SetImageId, GetImageId

Continued on next page

Table 1.100 – *Continued from previous page*

Field Name	Type	Description	Field Accessors
IsExpanded	BOOLEAN	No information	SetIsExpanded, GetIsExpanded
MenuItems	list of ui.MenuItem	A set of menu options belonging to the same menu.	SetMenuItems, GetMenuItems
OnStateChanged	ui.EventHandler	No information	SetOnStateChanged, GetOnState-Changed
Text	String	This is the of the menu group.	SetText, GetText

1.68.4 Static Methods Description

Table 1.101: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.69 ui.MenuItem

1.69.1 Brief description

This UI element serves as the base class for all menu items: menu commands, menu groups, and menu separators.

1.69.2 Inheritance Diagram

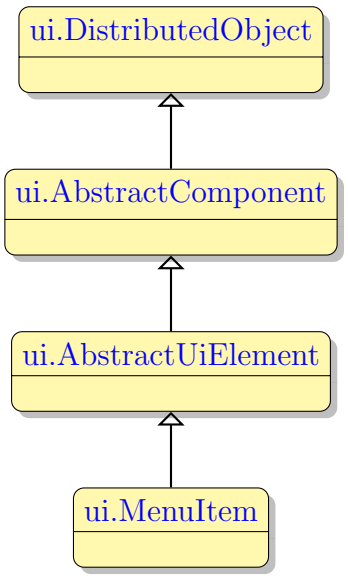

Figure 1.41: Inheritance Diagram of ui.MenuItem

1.69.3 Fields Description

Table 1.102: Fields description

Field Name	Type	Description	Field Accessors
HelpLabel	ui.Label	The description of the menu option that is shown on the second menu line for the ring menu.	SetHelpLabel, GetHelpLabel

1.69.4 Static Methods Description

Table 1.103: Static methods description

Name	**Parameters**	**Description**
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.70 ui.MenuSeparator

1.70.1 Brief description

It is a horizontal line that visually separates menu options in the drop-down list of the menu group.

1.70.2 Inheritance Diagram

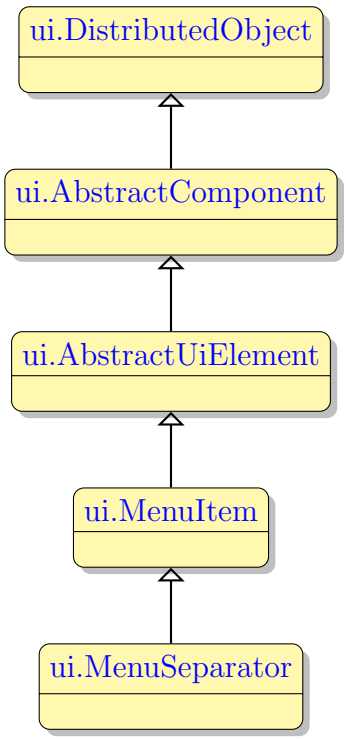

Figure 1.42: Inheritance Diagram of ui.MenuSeparator

1.70.3 Static Methods Description

Table 1.104: Static methods description

Name	Parameters	Description

Continued on next page

Table 1.104 – *Continued from previous page*

Name	**Parameters**	**Description**
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.71 MenuType

1.71.1 Brief description

No information

1.71.2 Possible values

Table 1.105: Possible values

Value	Description
None	The property is not applied and the default behaviour is used.
Menu	Not described yet
Tree	Not described yet
PopTree	Not described yet

1.72 ui.OnIdle

1.72.1 Brief description

This event is triggered after the application has been idle for some time.

1.72.2 Fields Description

Table 1.106: Fields description

Field Name	Type	Description	Field Accessors
Handler	ui.EventHandler	It specifies the event handler that should be invoked on the keypress.	SetHandler, GetHandler
IdleSeconds	Int	It specifies the time the system should be idle in order for the OnIdle event to be triggered. The time is specified in seconds.	SetIdleSeconds, GetIdleSeconds

1.73 ui.OpenUrlEventHandler

1.73.1 Brief description

This is an event handler that can be assigned to any event. This handler opens the URL specified in the default system web browser.

1.73.2 Inheritance Diagram

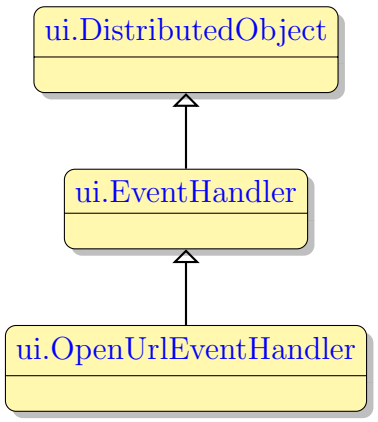

Figure 1.43: Inheritance Diagram of ui.OpenUrlEventHandler

1.73.3 Fields Description

Table 1.107: Fields description

Field Name	Type	Description	Field Accessors
Url	String	An URL, generally it requires the explicit specification of the protocol: http, ftp, etc..	SetUrl, GetUrl

1.73.4 Static Methods Description

Table 1.108: Static methods description

Name	Parameters	Description

Continued on next page

Table 1.108 – *Continued from previous page*

Name	**Parameters**	**Description**
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.

1.74 Orientation

1.74.1 Brief description

This enum specifies whether the UI element should have vertical or horizontal layout. The horizontal layout is the default one. It is applied to some containers which defines the layout of the elements inside the container. It is also applied to ui.Slider , ui.ProgressBar and ui.ScrollBar UI elements.

1.74.2 Possible values

Table 1.109: Possible values

Value	Description
Horizontal	The UI element will be placed horizontally and directed from left to right.
Vertical	The UI element will be placed vertically and directed from top to bottom.

1.75 ui.PlaceHolder

1.75.1 Brief description

No information

1.75.2 Inheritance Diagram

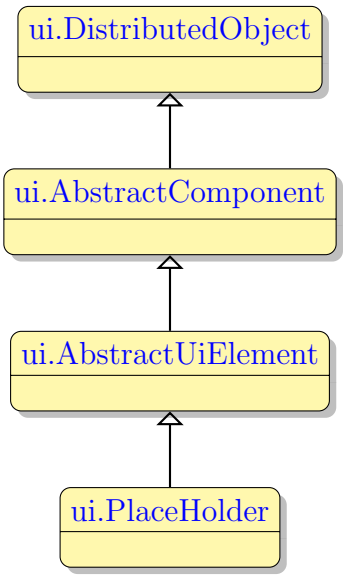

Figure 1.44: Inheritance Diagram of ui.PlaceHolder

1.75.3 Static Methods Description

Table 1.110: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.76 ui.PopupMenu

1.76.1 Brief description

This is the context menu that is invoked by right-clicking the application area at runtime. Typically the menu items of the pop-up menu correspond to the toolbar buttons currently active/visible.

1.76.2 Inheritance Diagram

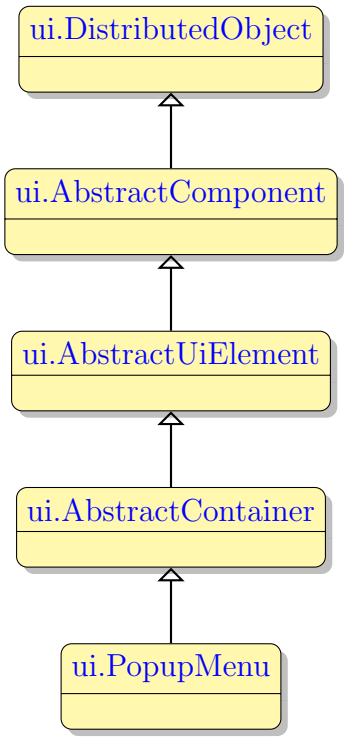

Figure 1.45: Inheritance Diagram of ui.PopupMenu

1.76.3 Fields Description

Table 1.111: Fields description

Field Name	Type	Description	Field Accessors
MenuItems	list of ui.MenuItem	A set of menu options belonging to the same menu.	SetMenuItems, GetMenuItems

1.76.4 Static Methods Description

Table 1.112: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.77 ui.ProgressBar

1.77.1 Brief description

This is a concrete UI element that has a form of a rectangular bar that can show the progress of the application execution by means of being filled with colour background gradually. For it to reflect the progress, the DISPLAY TO statement should be used to indicate the degree to which it must be filled after each stage. The progress bar should have the maximum value (when it is displayed to the progress bar it becomes 100 percent filled) and minimum value (when displayed makes the progress bar 0 percent filled).

1.77.2 Inheritance Diagram

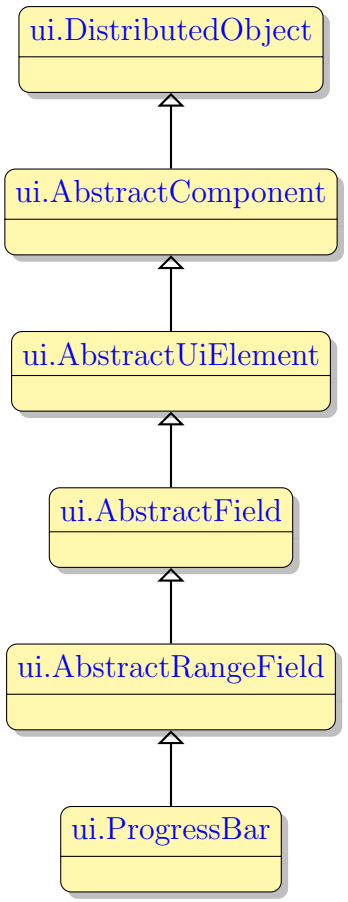

Figure 1.46: Inheritance Diagram of ui.ProgressBar

1.77.3 Fields Description

Table 1.113: Fields description

Field Name	Type	Description	Field Accessors
Step	Int	This is a number by which the value of the UI element can be increased or decreased at a time. It must be within the maximum and minimum value range. It prevents floating value changing.	SetStep, GetStep

1.77.4 Static Methods Description

Table 1.114: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.78 ui.Radio

1.78.1 Brief description

A Radio is a UI element that can only occur inside a ui.RadioGroup . It can be in either of the two states at a time - checked or unchecked. The state of one Radio in a list influences and depends on the state of other items in the same list.

1.78.2 Inheritance Diagram

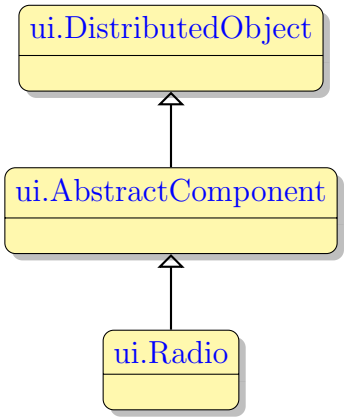

Figure 1.47: Inheritance Diagram of ui.Radio

1.78.3 Fields Description

Table 1.115: Fields description

Field Name	Type	Description	Field Accessors
AllowNewlines	BOOLEAN	This property specifies whether the Enter key will be used to move to another form element at runtime (if the value is FALSE), or it will create a newline symbol inside the current field (if the value is TRUE). It is typically applied for the ui.TextArea element.	SetAllowNewlines, GetAllowNewlines
Image	ui.Image	It is an image that can be applied to other UI elements, e.g. to a button.	SetImage, GetImage

Continued on next page

Table 1.115 – *Continued from previous page*

Field Name	Type	Description	Field Accessors
IsChecked	BOOLEAN	The UI element that has such field can be either in checked state (TRUE) or unchecked state (FALSE). UI elements like check boxes or radio buttons typically contain such field. Every time the element is clicked, the state is flipped.	SetIsChecked, GetIsChecked
OnCheck	ui.EventHandler	The OnCheck field defines the event which will be triggered if the IsChecked field of the UI element is changed to TRUE.	SetOnCheck, GetOnCheck
Title	String	This is the inscription attached to the UI element. Usually this is the text of all sorts of labels.	SetTitle, GetTitle

1.78.4 Extra Methods Description

Table 1.116: Extra methods description

Name	Parameters	Description
GetValue		Returns the value of a Radio
SetValue	Type value	Assigns the value to a Radio

1.78.5 Static Methods Description

Table 1.117: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.

Continued on next page

Table 1.117 – *Continued from previous page*

Name	**Parameters**	**Description**
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.79 ui.RadioGroup

1.79.1 Brief description

The Radio is a UI element - a form widget - that contains a set of ui.Radio which are either in selected or deselected state. The user can select only one Radio belonging to the same RadioGroup at a time, selecting a new item from the set deselects the previously selected element.

1.79.2 Inheritance Diagram

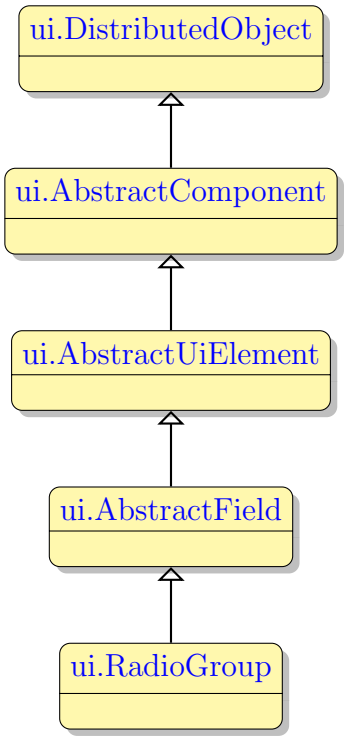

Figure 1.48: Inheritance Diagram of ui.RadioGroup

1.79.3 Fields Description

Table 1.118: Fields description

Field Name	Type	Description	Field Accessors

Continued on next page

Table 1.118 – *Continued from previous page*

Field Name	Type	Description	Field Accessors
Orientation	String catalog for Orientation	This enum specifies whether the UI element should have vertical or horizontal layout.	SetOrientation, GetOrientation
Radios	list of ui.Radio	This is the list of Radios that belong to the specified RadioGroup element.	SetRadios, GetRadios
Required	BOOLEAN	No information	SetRequired, GetRequired

1.79.4 Static Methods Description

Table 1.119: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.80 ui.ReportViewerConfig

1.80.1 Brief description

No information

1.80.2 Inheritance Diagram

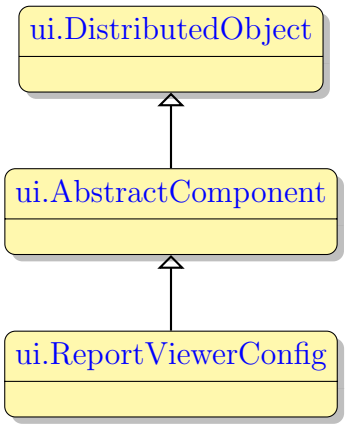

Figure 1.49: Inheritance Diagram of ui.ReportViewerConfig

1.80.3 Fields Description

Table 1.120: Fields description

Field Name	Type	Description	Field Accessors
Parameter	String	This is the type of the wrapper to be applied to the table.	SetParameter, GetParameter
ViewerType	String catalog for ViewerType	No information	SetViewerType, GetViewerType

1.80.4 Static Methods Description

Table 1.121: Static methods description

Name	Parameters	Description

Continued on next page

Table 1.121 – *Continued from previous page*

Name	**Parameters**	**Description**
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.81 ui.ResourceId

1.81.1 Brief description

This is the specification of a media resource that is to be applied to the UI element, normally of an image or an icon. It specifies the media file, the path to it and other information about this media file.

—This type is represented as 4GL record—

1.81.2 Fields Description

Table 1.122: Fields description

Field Name	Type	Description	Field Accessors
Uri	String	It is the URI of a media resource. The resource should be located on the application server and the URI should begin with qx://application/... .	

1.82 ui.RingArea

1.82.1 Brief description

This is the area that incorporates ring menu and its options. It must not be mistaken with the MenuBar used for top menu.

1.82.2 Inheritance Diagram

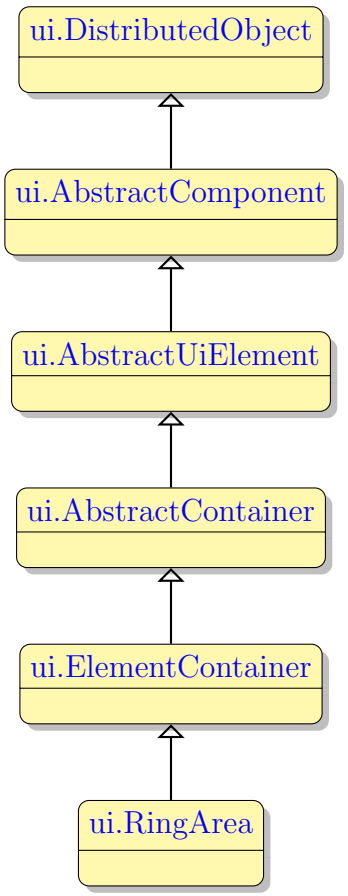

Figure 1.50: Inheritance Diagram of ui.RingArea

1.82.3 Static Methods Description

Table 1.123: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.83 RingMenuStyle

1.83.1 Brief description

No information

1.83.2 Possible values

Table 1.124: Possible values

Value	Description
None	The property is not applied and the default behaviour is used.
Dialog	Not described yet
Popup	Not described yet

1.84 ScaleType

1.84.1 Brief description

It indicates whether the UI element contents will be scaled, when the element is resized. The element resizing depends on the layout of the form and is predefined by the container. The scaling does not influence whether or not the physical size of the element will be changed by the attempt to resize it, it only influences the element contents. during the resizing.

1.84.2 Possible values

Table 1.125: Possible values

Value	Description
NoScale	The scaling is not applied when the element is resized. It will be resized only according to its layout position; e.g. the button will be enlarged, but the text on it will remain unchanged.
Both	When an element is resized, its contents is also resized: if a button gets bigger, the text in it also gets the bigger font.

1.85 ui.ScrollBar

1.85.1 Brief description

It is a concrete UI element that is represented by a scrollbar. It as the maximum and minimum values and the slider can be moved by the user at runtime or by displaying values to the element.

1.85.2 Inheritance Diagram

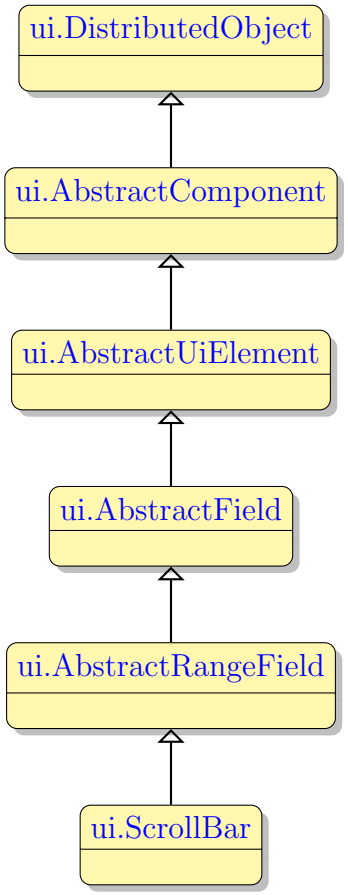

Figure 1.51: Inheritance Diagram of ui.ScrollBar

1.85.3 Fields Description

Table 1.126: Fields description

Field Name	Type	Description	Field Accessors
LargeStep	Int	It indicates the value by which the slider will be moved at a time, if the user moves it by holding down the arrow key.	SetLargeStep, GetLargeStep
OnScroll	ui.EventHandler	This is the event invoked when the slider of the UI element moves.	SetOnScroll, GetOnScroll
Orientation	String catalog for Orientation	This enum specifies whether the UI element should have vertical or horizontal layout.	SetOrientation, GetOrientation
SmallStep	Int	It indicates the smallest value by which the slider can be moved at a time. The slider cannot move smoothly and stop at values that won't make a complete step. E.g.: if the step is 2, the slider cannot stop at values 1, 3, 5, etc., it can stop at values 0,2,4,6 and so on. The small step is used when the user moves the slider by a single press of the arrow key on the keyboard.	SetSmallStep, GetSmallStep
ViewportSize	Int	No information	SetViewportSize, GetViewportSize

1.85.4 Static Methods Description

Table 1.127: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.86 ui.ScrollViewer

1.86.1 Brief description

It is a container the content of which can be bigger than the container. The scrollbars are used to view the content that does not fit. It can contain exactly one element. E.g. it can contain a stack panel container, the number of elements inside which can be bigger than fit the size of the Scroll Viewer.

1.86.2 Inheritance Diagram

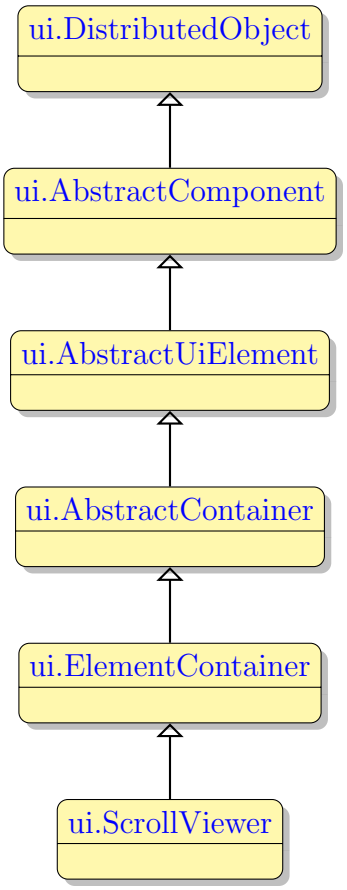

Figure 1.52: Inheritance Diagram of ui.ScrollViewer

1.86.3 Static Methods Description

Table 1.128: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.87 ui.Separator

1.87.1 Brief description

Any kind of separator, e.g. the status bar separator.

1.87.2 Inheritance Diagram

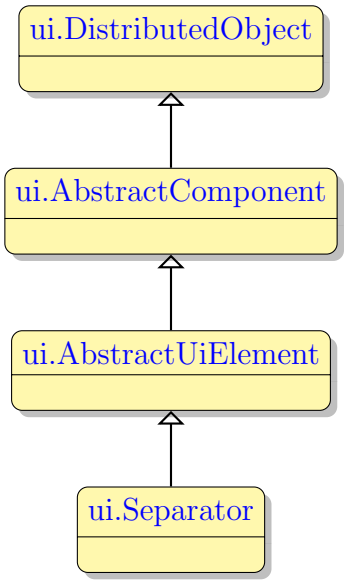

Figure 1.53: Inheritance Diagram of ui.Separator

1.87.3 Fields Description

Table 1.129: Fields description

Field Name	Type	Description	Field Accessors
CornerRadius	ui.CornerRadius	The radius of a corner of a custom border around the UI element. It is used to make the border corners rounded.	SetCornerRadius, GetCornerRadius
SeparatorType	String catalog for SeparatorType	This is the type of the separator to be displayed	SetSeparatorType, GetSeparatorType

1.87.4 Static Methods Description

Table 1.130: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.88 SeparatorType

1.88.1 Brief description

This is the type of the separator to be displayed

1.88.2 Possible values

Table 1.131: Possible values

Value	Description
Horizontal	Separator in the form of a single horizontal line
Vertical	Separator in the form of a single vertical line.
LeftTop	Separator in the form of two short lines adjoining orthogonally and forming a left top corner of a rectangle.
RightTop	Separator in the form of two short lines adjoining orthogonally and forming a right top corner of a rectangle.
LeftBottom	Separator in the form of two short lines adjoining orthogonally and forming a left bottom corner of a rectangle.
RightBottom	Separator in the form of two short lines adjoining orthogonally and forming a right bottom corner of a rectangle.
Cross	Separator in the form of two short lines intersecting orthogonally and forming an equilateral cross. Serves for connecting vertical and horizontal separators that overlap separators.
LeftJunction	Separator in the form of one longer vertical and one shorter horizontal line with the shorter line adjoining the longer one orthogonally at the middle from its left side. Serves for connecting a horizontal separator to the middle of vertical one.
RightJunction	LeftJunction - Separator in the form of one longer vertical and one shorter horizontal line with the shorter line adjoining the longer one orthogonally at the middle from its right side. Serves for connecting a horizontal separator to the middle of vertical one.
TopJunction	Separator in the form of one longer horizontal and one shorter vertical line with the shorter line adjoining the longer one orthogonally at the middle from the top. Serves for connecting a vertical separator to the middle of horizontal one.
BottomJunction	Separator in the form of one longer horizontal and one shorter vertical line with the shorter line adjoining the longer one orthogonally at the middle from the bottom. Serves for connecting a vertical separator to the middle of horizontal one.

1.89 ui.ServerEventHandler

1.89.1 Brief description

EMPTY.

1.89.2 Inheritance Diagram

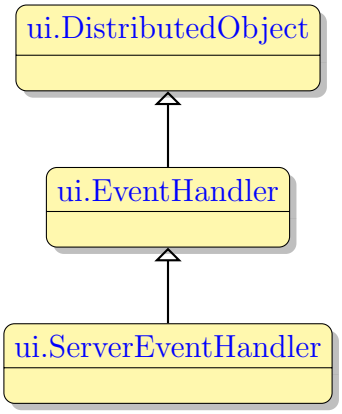

Figure 1.54: Inheritance Diagram of ui.ServerEventHandler

1.89.3 Extra Methods Description

Table 1.132: Extra methods description

Name	Parameters	Description
SetCallBackAction	String action name, String subdialog name	Set callback which triggers ACTION
SetCallBackFunction	String function name, String subdialog name	Set callback which triggers FUNCTION
SetCallBackKey	String key name, String subdialog name	Set callback which triggers ON KEY

1.90 ui.SetLabelText

1.90.1 Brief description

This event is triggered when a ring menu option is activated. It displays the description pf the selected menu option to the menu help line. In this case help string is the line below the menu line and the text displayed is the menu option description. This event is also used to clear the error line. It displays empty string to the error line when any event occurs.

1.90.2 Inheritance Diagram

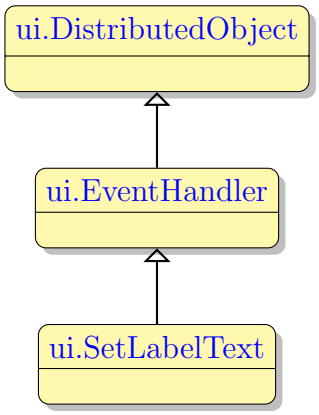

Figure 1.55: Inheritance Diagram of ui.SetLabelText

1.90.3 Fields Description

Table 1.133: Fields description

Field Name	Type	Description	Field Accessors
DstLabel	ui.Label	This is the label which text should be changed by this event.	SetDstLabel, GetDstLabel
HelpString	String	This is the text that will be displayed to the corresponding predefined line.	SetHelpString, GetHelpString

1.90.4 Static Methods Description

Table 1.134: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.

1.91 ui.Size

1.91.1 Brief description

The size of a UI element in pixels.

—This type is represented as 4GL record—

1.91.2 Fields Description

Table 1.135: Fields description

Field Name	Type	Description	Field Accessors
Width	String	The width of the UI element in pixels.	
Height	String	The height of the UI element in pixels.	

1.92 ui.Slider

1.92.1 Brief description

This is a concrete UI element that consists of a scale and a slider that can move across this scale. The slider widget has the minimum and maximum value which present the start and the end of the scale. It can be moved directly by the user during the input, or it can be moved if a value within its values range is displayed to it by the 4GL means.

1.92.2 Inheritance Diagram

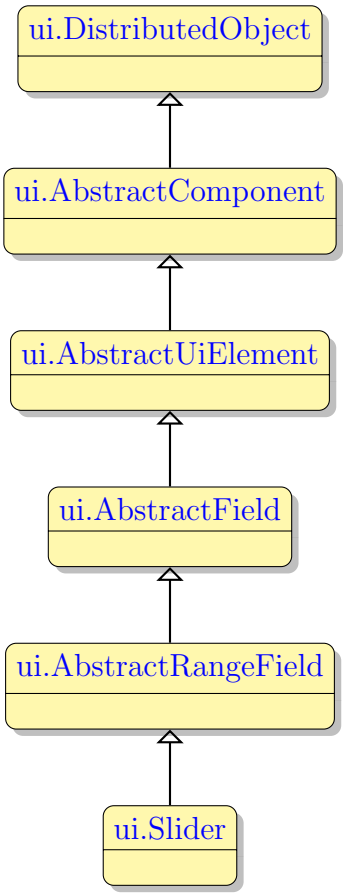

Figure 1.56: Inheritance Diagram of ui.Slider

1.92.3 Fields Description

Table 1.136: Fields description

Field Name	Type	Description	Field Accessors
Orientation	String catalog for Orientation	This enum specifies whether the UI element should have vertical or horizontal layout.	SetOrientation, GetOrientation
Step	Int	This is a number by which the value of the UI element can be increased or decreased at a time. It must be within the maximum and minimum value range. It prevents floating value changing.	SetStep, GetStep

1.92.4 Static Methods Description

Table 1.137: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.93 Sorted

1.93.1 Brief description

No information

1.93.2 Possible values

Table 1.138: Possible values

Value	Description
None	The property is not applied and the default behaviour is used.
Asc	Not described yet
Desc	Not described yet

1.94 ui.SpecificKeyEventHandler

1.94.1 Brief description

This event handler specifies what event handler should be triggered when a specific key is pressed. It links the keypress with a 4GL event.

1.94.2 Inheritance Diagram

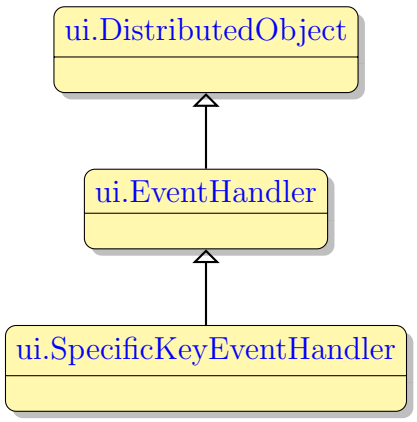

Figure 1.57: Inheritance Diagram of ui.SpecificKeyEventHandler

1.94.3 Fields Description

Table 1.139: Fields description

Field Name	Type	Description	Field Accessors
Handler	ui.EventHandler	It specifies the event handler that should be invoked on the keypress.	SetHandler, GetHandler
KeysInfo	list of ui.KeyEvent	It specifies the concrete keys that must be pressed to trigger the event.	SetKeysInfo, GetKeysInfo

1.94.4 Static Methods Description

Table 1.140: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.

1.95 ui.Spinner

1.95.1 Brief description

This is a concrete UI element that has a form of a field available for inputting and displaying data that accepts only values inside the allowed range of values. It has the up and down arrows on the right that allow the user to scroll through the acceptable values and prevents the user from entering values from keyboard.

1.95.2 Inheritance Diagram

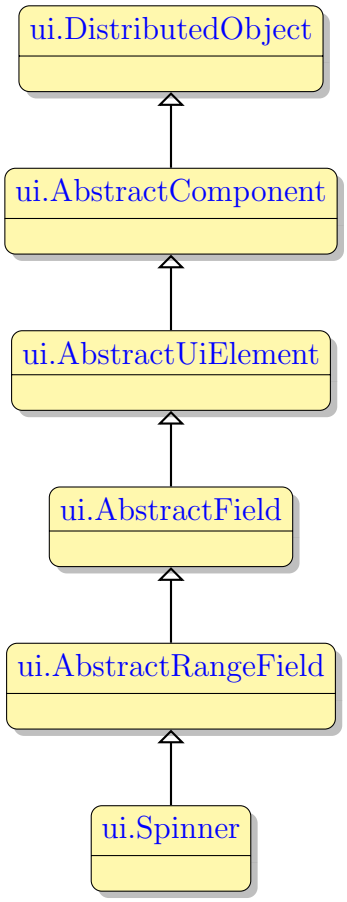

Figure 1.58: Inheritance Diagram of ui.Spinner

1.95.3 Fields Description

Table 1.141: Fields description

Field Name	Type	Description	Field Accessors
Step	Int	This is a number by which the value of the UI element can be increased or decreased at a time. It must be within the maximum and minimum value range. It prevents floating value changing.	SetStep, GetStep

1.95.4 Static Methods Description

Table 1.142: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.96 ui.StackPanel

1.96.1 Brief description

This is a container which arranges the elements in horizontal or vertical stacks. Any number of elements can be placed inside this container one next to the other. At runtime the contents of the stack panel can be resized only in the direction opposite to the orientation of the container.

1.96.2 Inheritance Diagram

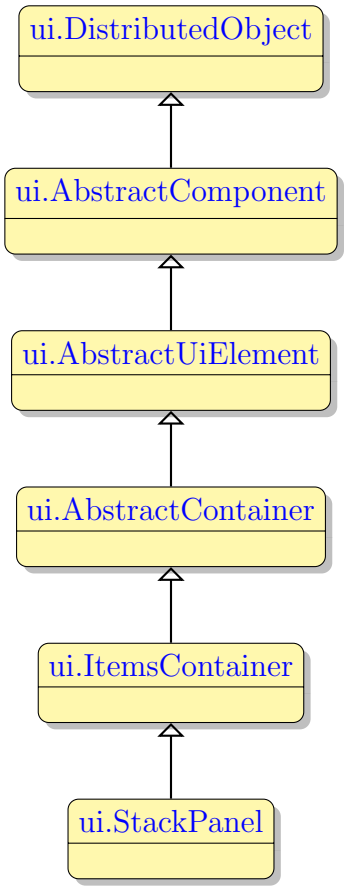

Figure 1.59: Inheritance Diagram of ui.StackPanel

1.96.3 Fields Description

Table 1.143: Fields description

Field Name	Type	Description	Field Accessors
Orientation	String catalog for Orientation	This enum specifies whether the UI element should have vertical or horizontal layout.	SetOrientation, GetOrientation
Reverse	BOOLEAN	No information	SetReverse, GetReverse

1.96.4 Static Methods Description

Table 1.144: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.97 ui.StartProgramEventHandler

1.97.1 Brief description

This event handler specifies the child 4GL program that should be launched and the parameters of this program. It is normally used for the MDI mode, but can be used in other cases.

1.97.2 Inheritance Diagram

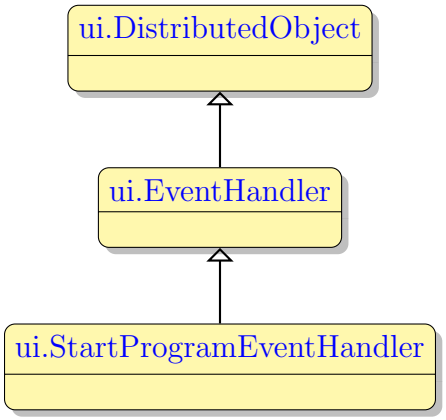

Figure 1.60: Inheritance Diagram of ui.StartProgramEventHandler

1.97.3 Fields Description

Table 1.145: Fields description

Field Name	Type	Description	Field Accessors
ProgramName	String	The name of the child program.	SetProgramName, GetProgramName
ProgramParameters	String	The parameters of the child program.	SetProgramParameters, GetProgramParameters
ProgramPort	String	The port on the application server.	SetProgramPort, GetProgramPort
ProgramServer	String	The name of the host - the application server on which the program is deployed and should run.	SetProgramServer, GetProgramServer

Continued on next page

Table 1.145 – *Continued from previous page*

Field Name	Type	Description	Field Accessors
UserId	String	The name of the user who runs the application.	SetUserId, GetUserId
Waiting	BOOLEAN	It indicates whether the parent program should be suspended until the child program is closed.	SetWaiting, GetWaiting

1.97.4 Static Methods Description

Table 1.146: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.

1.98 ui.StartedBy

1.98.1 Brief description

No information

—This type is represented as 4GL record—

1.98.2 Fields Description

Table 1.147: Fields description

Field Name	Type	Description	Field Accessors
EventId	Int	No information	
ClientPID	Int	No information	
ParentWait	BOOLEAN	No information	

1.99 ui.StatusBar

1.99.1 Brief description

It is the last line of any 4Gl window which is not included into the window size from the 4Gl perspective. It is used to display the errors, messages and comments. By default it is divided in two parts. The first half displays the field comments, the second part displays errors and messages.

1.99.2 Inheritance Diagram

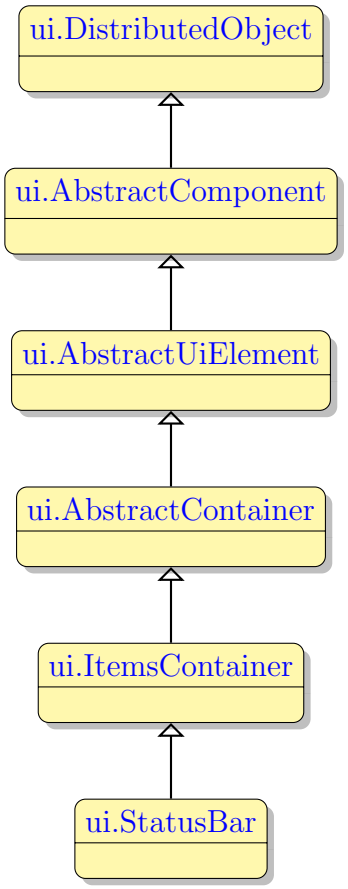

Figure 1.61: Inheritance Diagram of ui.StatusBar

1.99.3 Static Methods Description

Table 1.148: Static methods description

Name	**Parameters**	**Description**
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.100 ui.SystemColor

1.100.1 Brief description

The system color defines a list of preset colours that can be applied to widgets, as opposed to the custom colour where the user needs to specify RGB of the color.

1.100.2 Inheritance Diagram

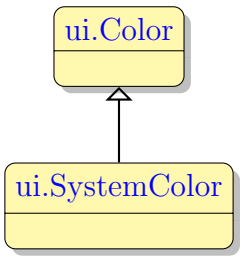

Figure 1.62: Inheritance Diagram of ui.SystemColor

1.100.3 Fields Description

Table 1.149: Fields description

Field Name	Type	Description	Field Accessors
SystemColorName	String catalog for SystemColorName	It is the name of one of the predefined system colors.	SetSystemColorName, GetSystemColorName

1.100.4 Static Methods Description

Table 1.150: Static methods description

Name	Parameters	Description

Continued on next page

Table 1.150 – *Continued from previous page*

Name	**Parameters**	**Description**
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.

1.101 SystemColorName

1.101.1 Brief description

It is a name of a preset system color the color code for which is hard-coded and associated with this name.

1.101.2 Possible values

Table 1.151: Possible values

Value	Description
None	The property is not applied and the default behaviour is used.
Black	RGB 0 0 0.
Gray	RGB 230 230 230.
DarkGray	RGB 75 75 75.
LightGray	RGB 217 217 217.
White	RGB 255 255 255.
Red	RGB 156 0 6.
LightRed	RGB 255 183 186.
Magenta	RGB 197 28 90.
LightMagenta	RGB 250 207 221.
Green	RGB 0 97 0.
LightGreen	RGB 190 240 200.
Blue	RGB 31 73 125.
LightBlue	RGB 190 210 240.
Cyan	RGB 49 134 155.
LightCyan	RGB 205 235 235.
Yellow	RGB 156 101 0.
LightYellow	RGB 255 235 156.
Purple	RGB 172 5 76.
LightPurple	RGB 228 186 232.
Orange	RGB 226 107 10.
LightOrange	RGB 253 233 217.

1.102 ui.SystemContextMenu

1.102.1 Brief description

This is the context menu which is invoked by right-clicking the title bar of the 4GL window.

1.102.2 Inheritance Diagram

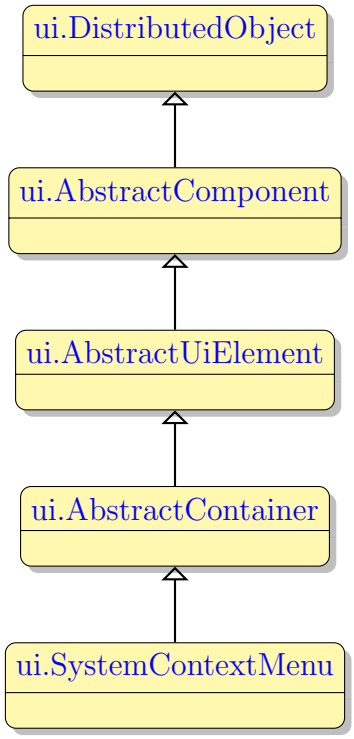

Figure 1.63: Inheritance Diagram of ui.SystemContextMenu

1.102.3 Fields Description

Table 1.152: Fields description

Field Name	Type	Description	Field Accessors
SystemMenuItems	list of ui.SystemMenuItem	It is the list of items belonging to the system context menu.	SetSystemMenuItems, GetSystemMenu-Items

1.102.4 Static Methods Description

Table 1.153: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.103 ui.SystemMenuItem

1.103.1 Brief description

It is a single menu option that belongs the the ui.SystemContextMenu .

1.103.2 Inheritance Diagram

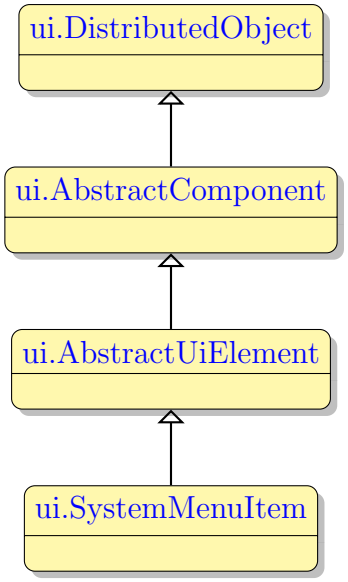

Figure 1.64: Inheritance Diagram of ui.SystemMenuItem

1.103.3 Fields Description

Table 1.154: Fields description

Field Name	Type	Description	Field Accessors
Image	ui.Image	It specifies the icon next to the system menu option. The icon must be 12x12 pixels, monochrome.	SetImage, GetImage
OnInvoke	ui.EventHandler	The event which is triggered when the UI element is invoked. It can be invoked by mouse click, by pressing Enter, or in some cases Space, when the cursor is in the element.	SetOnInvoke, GetOnInvoke
Text	String	A character string.	SetText, GetText

1.103.4 Static Methods Description

Table 1.155: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.104 ui.Tab

1.104.1 Brief description

This is a special type of container which can contain any number of elements, but these elements can only be of ui.TabPage . The Tab serves as the container for a stack of tab pages with only one page visible at a time. Other pages can be brought forward by clicking on their tabs.

1.104.2 Inheritance Diagram

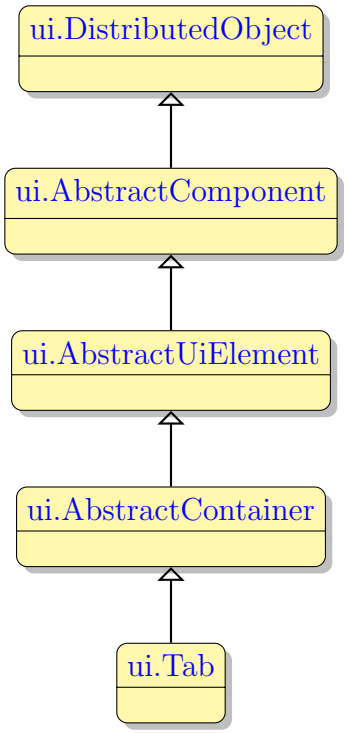

Figure 1.65: Inheritance Diagram of ui.Tab

1.104.3 Fields Description

Table 1.156: Fields description

Field Name	Type	Description	Field Accessors

Continued on next page

Table 1.156 – *Continued from previous page*

Field Name	Type	Description	Field Accessors
OnSelectedTabPageChanged	ui.EventHandler	This is an event that is triggered every time the current tab page is changed.	SetOnSelectedTabPageCh GetOnSelectedTabPageChanged
SelectedTabPage	ui.TabPage	It defines which tab page is the current one - the contents of which tab page is now visible.	SetSelectedTabPage, GetSelectedTabPage
TabPagePlacement	String catalog for TabPagePlacement	It defines where the tabs should be located - to which side of the tab panel should they adjoin.	SetTabPagePlacement, GetTabPagePlacement
TabPages	list of ui.TabPage	This is the set of tab pages that belong to the same tab container.	SetTabPages, GetTabPages

1.104.4 Static Methods Description

Table 1.157: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.105 ui.TabPage

1.105.1 Brief description

This is a container that can only be placed inside the ui.Tab container. A tab page can contain a single element of any type. Each tab page has a tab with the page title which is used to bring the page forward from the stack of other tab pages at runtime or during form modification.

1.105.2 Inheritance Diagram

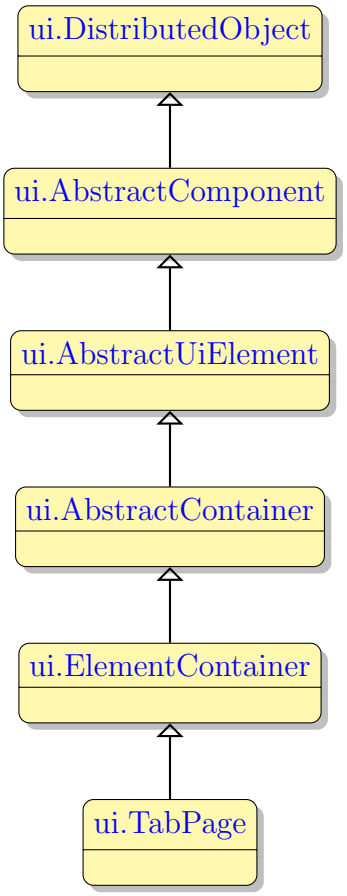

Figure 1.66: Inheritance Diagram of ui.TabPage

1.105.3 Fields Description

Table 1.158: Fields description

Field Name	Type	Description	Field Accessors
Image	ui.Image	This is an icon that can be displayed to the tab of the page with or instead of the page title.	SetImage, GetImage
OnSelectTabPage	ui.EventHandler	This is an event that is triggered every time the tab page becomes the current tab page of the tab container and its contents is brought forward.	SetOnSelectTabPage, GetOnSelectTabPage
Title	String	This is the inscription attached to the UI element. Usually this is the text of all sorts of labels.	SetTitle, GetTitle

1.105.4 Static Methods Description

Table 1.159: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.106 TabPagePlacement

1.106.1 Brief description

This enum defined where th list of tabs should be located. By default it is located horizontally below the top border of the tab container. They can also be located horizontally at the bottom of the container or vertically at its either side.

1.106.2 Possible values

Table 1.160: Possible values

Value	Description
Top	The UI element will be aligned to the top of the container (or container cell).
Left	The UI element will be aligned to the left side of the container (or container cell).
Right	The UI element will be aligned to the right side of the container (or container cell).
Bottom	The UI element will be aligned to the bottom of the container (or container cell).

1.107 ui.Table

1.107.1 Brief description

This is a container that can only contain a specific type of element - ui.TableColumn . It serves as the root container of a table with rows and columns of widgets used to display and input data.

1.107.2 Inheritance Diagram

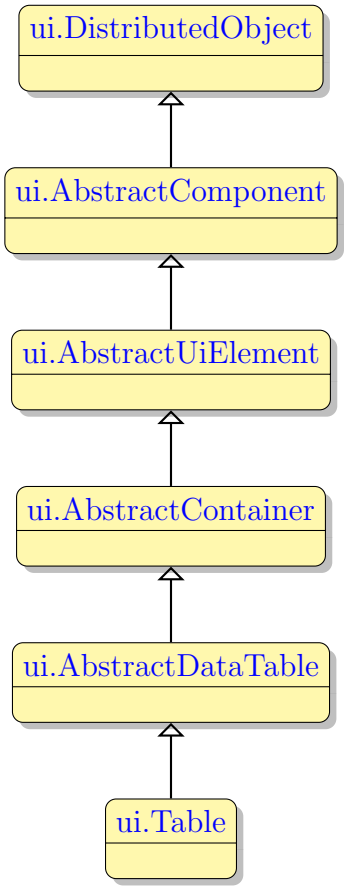

Figure 1.67: Inheritance Diagram of ui.Table

1.107.3 Static Methods Description

Table 1.161: Static methods description

Name	**Parameters**	**Description**
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.108 ui.TableColumn

1.108.1 Brief description

This is a container that can only be placed inside the ui.Table container or ui.TreeTable container. It can contain only one element belonging to the ui.AbstractField class. Though only one element can be placed into a column, this element will be repeated till the bottom of the column, creating table row together with the elements in other columns, if any. All the duplicates of the element will have the same identifier and will be treated as a single element by th e form designer. The 4GL can differentiate between the instances of the element belonging to different rows by means of using the element identifier together with the number of the table row. The table row numbers start at number 1 at the top of the table.

1.108.2 Inheritance Diagram

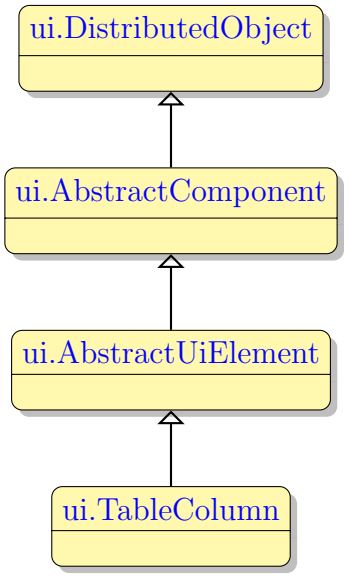

Figure 1.68: Inheritance Diagram of ui.TableColumn

1.108.3 Fields Description

Table 1.162: Fields description

Field Name	Type	Description	Field Accessors

Continued on next page

Table 1.162 – *Continued from previous page*

Field Name	Type	Description	Field Accessors
AllowNewlines	BOOLEAN	This property specifies whether the Enter key will be used to move to another form element at runtime (if the value is FALSE), or it will create a newline symbol inside the current field (if the value is TRUE). It is typically applied for the ui.TextArea element.	SetAllowNewlines, GetAllowNewlines
ColumnLength	ui.GridLength	It specifies the length of a column. The column length determines how many rows of widgets the table will have.	SetColumnLength, GetColumnLength
ColumnNum	Int	No information	SetColumnNum, GetColumnNum
ConstructControl	ui.AbstractField	No information	SetConstructControl, GetConstructControl
Control	ui.AbstractField	No information	SetControl, GetControl
EditControl	ui.AbstractField	No information	SetEditControl, GetEditControl
Footer	ui.AbstractUiElemen	No information	SetFooter, GetFooter
ReadOnly	BOOLEAN	If enabled, it prevents the user from entering values into the field at runtime even if the field is included into the input routine.	SetReadOnly, GetReadOnly
Resizable	BOOLEAN	It indicates whether the user is allowed to resize the column at runtime using the mouse cursor.	SetResizable, GetResizable
Sorted	String catalog for Sorted	No information	SetSorted, GetSorted
Text	String	This is the text used as the header of the column.	SetText, GetText
Unsortable	BOOLEAN	No information	SetUnsortable, GetUnsortable

1.108.4 Extra Methods Description

Table 1.163: Extra methods description

Name	Parameters	Description
Complete		Complete dynamically created control.

1.108.5 Static Methods Description

Table 1.164: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.109 ui.TableRowPos

1.109.1 Brief description

No information

—This type is represented as 4GL record—

1.109.2 Fields Description

Table 1.165: Fields description

Field Name	Type	Description	Field Accessors
RowIndex	Int	No information	
ColumnIndex	Int	No information	

1.110 ui.TemplateInstance

1.110.1 Brief description

No information

1.110.2 Inheritance Diagram

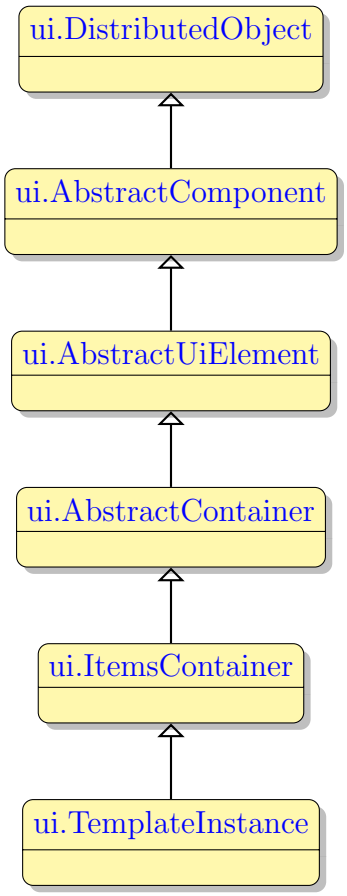

Figure 1.69: Inheritance Diagram of ui.TemplateInstance

1.110.3 Fields Description

Table 1.166: Fields description

Field Name	Type	Description	Field Accessors

Continued on next page

Table 1.166 – *Continued from previous page*

Field Name	Type	Description	Field Accessors
TemplateName	String	No information	SetTemplateName, GetTemplate-Name

1.110.4 Static Methods Description

Table 1.167: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.111 ui.TextAlignment

1.111.1 Brief description

It defines the alignment of the text inside the UI element to which it belongs. For example, it can define the alignment of the text inside a table cell or inside a text area.

1.111.2 Fields Description

Table 1.168: Fields description

Field Name	Type	Description	Field Accessors
HorizontalTextAlignment	String catalog for HorizontalTextAlignment		SetHorizontalTextAlignm GetHorizontal-TextAlignment
VerticalTextAlignment	String catalog for VerticalTextAlignment		SetVerticalTextAlignment GetVerticalTextAlignment

1.112 ui.TextArea

1.112.1 Brief description

This is a concrete UI element that has the form of a text field and shares many features with ui.TextField , but is designed for working with multiline text instead of single lines of text. It does not have some features of the text field that deal with the navigation between fields, but instead it had improved facilities for navigating inside the field.

1.112.2 Inheritance Diagram

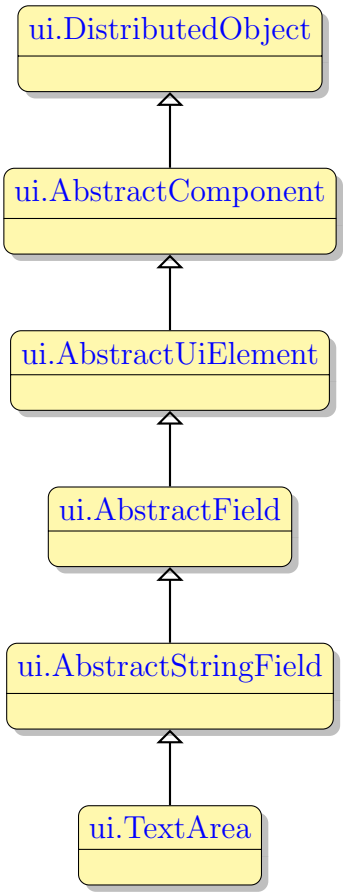

Figure 1.70: Inheritance Diagram of ui.TextArea

1.112.3 Fields Description

Table 1.169: Fields description

Field Name	Type	Description	Field Accessors
AllowTabulation	BOOLEAN	It indicates whether the Tab key will move the cursor to the next field (FALSE - default value) or create a TAB symbol inside the field.	SetAllowTabulation, GetAllowTabulation
Autonext	BOOLEAN	If enabled, moves the cursor to the next field during input automatically, when the MaxLength of the current field is met.	SetAutonext, GetAutonext
Editor	String	Specifies the program to be used for opening and editing the BYTE or TEXT value.	SetEditor, GetEditor
HelperText	String	No information	SetHelperText, GetHelperText
LabelText	String	No information	SetLabelText, GetLabelText
MaxLength	Int	It specifies the maximum length in bytes allowed for entering into the filed. Its value is normally taken from the data type and size of the variable linked to the field.	SetMaxLength, GetMaxLength
PlaceholderText	String	No information	SetPlaceholderText, GetPlaceholderText
Required	BOOLEAN	No information	SetRequired, GetRequired
TextChanged	BOOLEAN	It indicates whether the text displayed in the text area was changed by the user or by the program.	SetTextChanged, GetTextChanged
ToCase	String catalog for ToCase	This property specifies the case of a UI element. It can be applied to any UI element that allows entering text from keyboard. By default its value is None, meaning that the case of the letters does not change and remains as they were inputted.	SetToCase, GetToCase

1.112.4 Static Methods Description

Table 1.170: Static methods description

Name	**Parameters**	**Description**
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.113 ui.TextField

1.113.1 Brief description

This is a concrete UI element that is commonly used for input and displaying information. Normally it is used to process a single line of data.

1.113.2 Inheritance Diagram

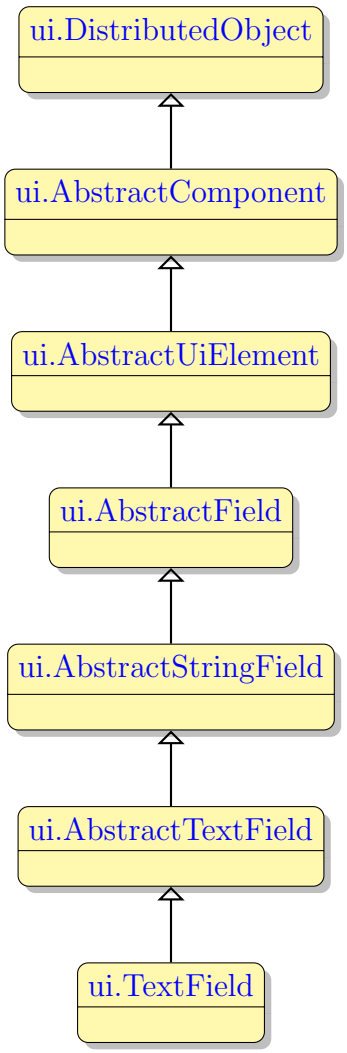

Figure 1.71: Inheritance Diagram of ui.TextField

1.113.3 Fields Description

Table 1.171: Fields description

Field Name	Type	Description	Field Accessors
AllowNewlines	BOOLEAN	This property specifies whether the Enter key will be used to move to another form element at runtime (if the value is FALSE), or it will create a newline symbol inside the current field (if the value is TRUE). It is typically applied for the ui.TextArea element.	SetAllowNewlines, GetAllowNewlines
HelperText	String	No information	SetHelperText, GetHelperText
InvisibleValue	BOOLEAN	If enabled, the value displayed to the field will be invisible. During input the value will be masked with *.	SetInvisibleValue, GetInvisibleValue
LabelText	String	No information	SetLabelText, GetLabelText
PlaceholderText	String	No information	SetPlaceholderText, GetPlaceholderText

1.113.4 Static Methods Description

Table 1.172: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.114 ui.TextInjectionEventHandler

1.114.1 Brief description

This event handler injects the text specified as its parameter into the current input widget. It can be assigned to any event.

1.114.2 Inheritance Diagram

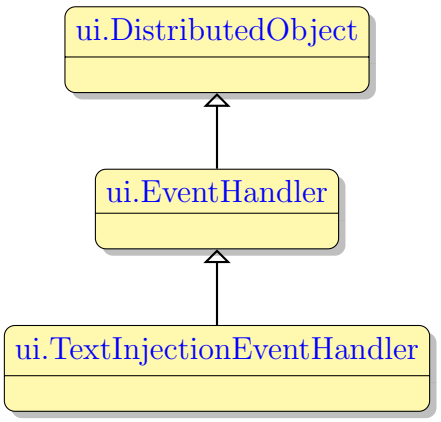

Figure 1.72: Inheritance Diagram of ui.TextInjectionEventHandler

1.114.3 Fields Description

Table 1.173: Fields description

Field Name	Type	Description	Field Accessors
Text	String	A character string.	SetText, GetText

1.114.4 Static Methods Description

Table 1.174: Static methods description

Name	Parameters	Description

Continued on next page

Table 1.174 – *Continued from previous page*

Name	**Parameters**	**Description**
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.

1.115 ui.Thickness

1.115.1 Brief description

This is a property which defines the thickness of elements or their parts. It is use to define the thickness of the border, the width or padding and margin offsets. The parts of the same object (e.g. border) can have different thickness in its different parts - for example a border can be 1 pixel wide at the top and 2 pixels wide at the bottom. If the thickness of any side is set to 0 - this side of the element absent.

—This type is represented as 4GL record—

1.115.2 Fields Description

Table 1.175: Fields description

Field Name	Type	Description	Field Accessors
Left	Int	The size of the left standoff in pixels.	
Top	Int	The size of the top standoff in pixels.	
Right	Int	The size of the right standoff in pixels.	
Bottom	Int	The size of the bottom standoff in pixels.	

1.116 ui.TimeEditField

1.116.1 Brief description

This is a concrete UI element that accepts a limited range of time values. The value inside the field is formatted into hh:mm:ss format. It also has up and down arrows that can scroll the data in the field - whether hours, minutes or seconds are scrolled depends on there inside the field the cursor is located.

1.116.2 Inheritance Diagram

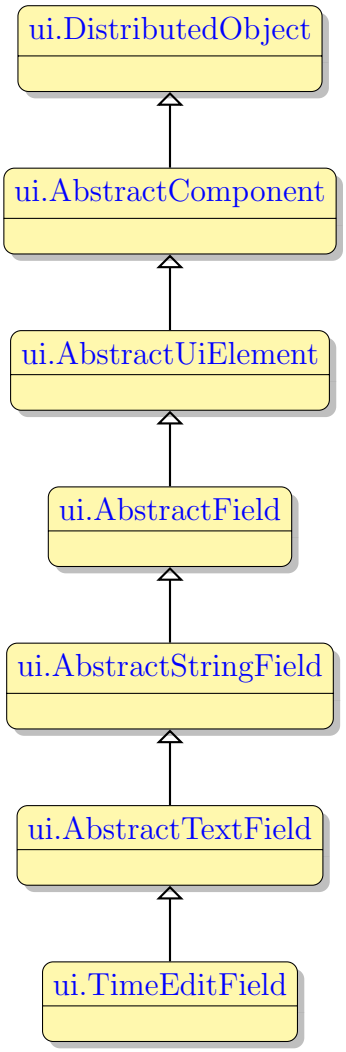

Figure 1.73: Inheritance Diagram of ui.TimeEditField

1.116.3 Fields Description

Table 1.176: Fields description

Field Name	Type	Description	Field Accessors
HelperText	String	No information	SetHelperText, GetHelperText
LabelText	String	No information	SetLabelText, GetLabelText
PlaceholderText	String	No information	SetPlaceholderText, GetPlaceholder-Text

1.116.4 Static Methods Description

Table 1.177: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.117 ui.TitleBarOptions

1.117.1 Brief description

This UI element unites the options that can influence the default buttons on the 4GL window title bar.

—This type is represented as 4GL record—

1.117.2 Fields Description

Table 1.178: Fields description

Field Name	Type	Description	Field Accessors
DisableTitleBarCloseButton	BOOLEAN	It disables the (x) close button on the right side of the window title bar. It gets gray and unclickable, but still remains visible.	
DisableTitleBarMaximizeButton	BOOLEAN	It disables the maximize button on the right side of the window title bar. It gets gray and unclickable, but still remains visible, if the minimize button is enabled. It is hidden, if the minimize button is also disabled.	
DisableTitleBarMinimizeButton	BOOLEAN	It disables the minimize button on the right side of the window title bar. It gets gray and unclickable, but still remains visible, if the maximize button is enabled. It is hidden, if the maximize button is also disabled.	
HideTitleBar	BOOLEAN	It hides the window title bar together with all its buttons. In this case though the buttons may not have been disabled, thew are still not usable.	

1.118 TitleJustification

1.118.1 Brief description

This enum defines the horizontal justification of the title text. It is typically is applied to window titles, column header titles, tab page titles, etc..

1.118.2 Possible values

Table 1.179: Possible values

Value	Description
Left	The UI element will be aligned to the left side of the container (or container cell).
Center	The UI element will be equidistant from both sides.
Right	The UI element will be aligned to the right side of the container (or container cell).

1.119 ToCase

1.119.1 Brief description

This is the case (lower case or upper case) to be applied to the text in the UI element.

1.119.2 Possible values

Table 1.180: Possible values

Value	Description
None	The property is not applied and the default behaviour is used.
Up	All the letters entered into the UI element will be uppercase letters regardless of their original case.
Down	All the letters entered into the UI element will be lowercase letters regardless of their original case.

1.120 ui.Toolbar

1.120.1 Brief description

This is the container that incorporates toolbar buttons.

1.120.2 Inheritance Diagram

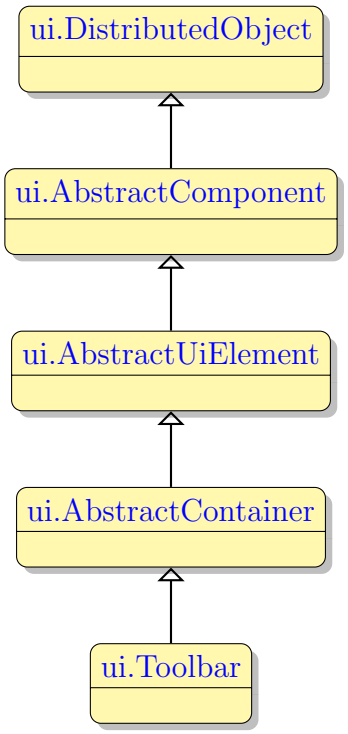

Figure 1.74: Inheritance Diagram of ui.Toolbar

1.120.3 Fields Description

Table 1.181: Fields description

Field Name	Type	Description	Field Accessors
HideLabels	BOOLEAN	It specifies whether the text on the toolbar buttons should be visible or not. If set to true - only the icons will be visible.	SetHideLabels, GetHideLabels

Continued on next page

Table 1.181 – *Continued from previous page*

Field Name	Type	Description	Field Accessors
ToolbarGroups	list of ui.ToolbarGroup	A set of all toolbar groups that belong to the toolbar.	SetToolbarGroups, GetToolbarGroups
ToolbarLocation	String catalog for ToolbarLocation	No information	SetToolbarLocation, GetToolbarLocation

1.120.4 Static Methods Description

Table 1.182: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.121 ui.ToolbarButton

1.121.1 Brief description

This is an individual toolbar button that belongs to the toolbar.

1.121.2 Inheritance Diagram

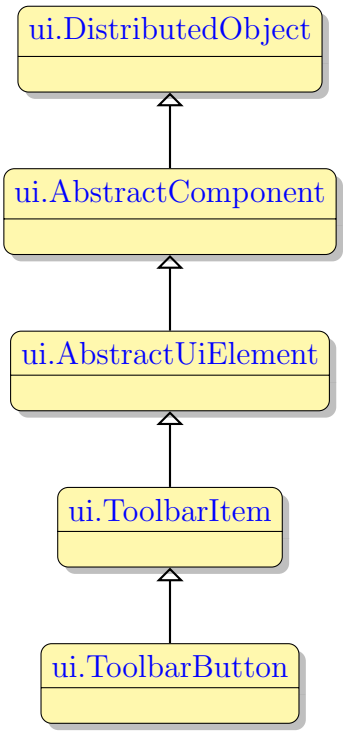

Figure 1.75: Inheritance Diagram of ui.ToolbarButton

1.121.3 Fields Description

Table 1.183: Fields description

Field Name	Type	Description	Field Accessors

Continued on next page

Table 1.183 – *Continued from previous page*

Field Name	Type	Description	Field Accessors
AllowNewlines	BOOLEAN	This property specifies whether the Enter key will be used to move to another form element at runtime (if the value is FALSE), or it will create a newline symbol inside the current field (if the value is TRUE). It is typically applied for the ui.TextArea element.	SetAllowNewlines, GetAllowNewlines
Image	ui.Image	It specifies the icon that should be displayed to the toolbar button. The button is resized to the size of the icon applied.	SetImage, GetImage
OnInvoke	ui.EventHandler	The event which is triggered when the UI element is invoked. It can be invoked by mouse click, by pressing Enter, or in some cases Space, when the cursor is in the element.	SetOnInvoke, GetOnInvoke
Text	String	This is the label of the toolbar button.	SetText, GetText

1.121.4 Static Methods Description

Table 1.184: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.122 ui.ToolbarGroup

1.122.1 Brief description

This is a set of toolbar buttons that are united into a single group. The group unites the toolbar buttons that have the same conditions for being displayed. It was designed to make the toolbar more dynamic - to display or hide the toolbar groups depending on what widgets are active and to combine different groups freely.

1.122.2 Inheritance Diagram

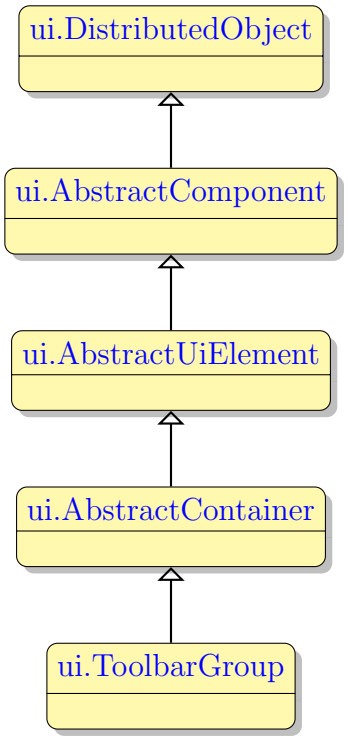

Figure 1.76: Inheritance Diagram of ui.ToolbarGroup

1.122.3 Fields Description

Table 1.185: Fields description

Field Name	Type	Description	Field Accessors

Continued on next page

Table 1.185 – *Continued from previous page*

Field Name	Type	Description	Field Accessors
ToolbarItems	list of ui.ToolbarItem	This is the list of Toolbar elements - toolbar buttons, toolbar separators - present in the toolbar UI element.	SetToolbarItems, GetToolbarItems

1.122.4 Static Methods Description

Table 1.186: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.123 ui.ToolbarItem

1.123.1 Brief description

This is an abstract element that unites the toolbar buttons and toolbar separators.

1.123.2 Inheritance Diagram

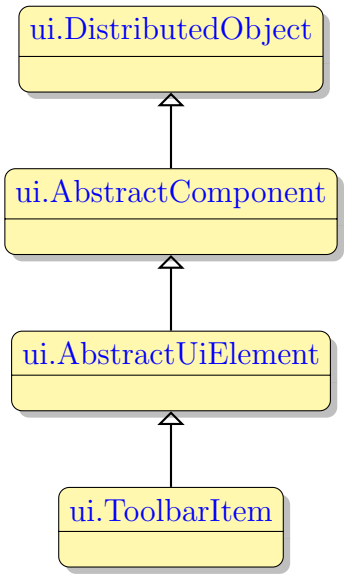

Figure 1.77: Inheritance Diagram of ui.ToolbarItem

1.123.3 Fields Description

Table 1.187: Fields description

Field Name	Type	Description	Field Accessors
Place	String	No information	SetPlace, GetPlace

1.123.4 Static Methods Description

Table 1.188: Static methods description

Name	**Parameters**	**Description**
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.124 ToolbarLocation

1.124.1 Brief description

No information

1.124.2 Possible values

Table 1.189: Possible values

Value	Description
Top	The UI element will be aligned to the top of the container (or container cell).
Right	The UI element will be aligned to the right side of the container (or container cell).

1.125 ui.ToolbarSeparator

1.125.1 Brief description

This is a visual separator that can visually divide the toolbar into logical sets of buttons.

1.125.2 Inheritance Diagram

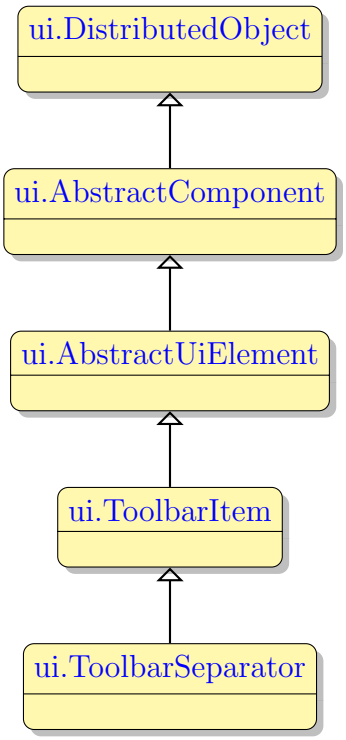

Figure 1.78: Inheritance Diagram of ui.ToolbarSeparator

1.125.3 Static Methods Description

Table 1.190: Static methods description

Name	Parameters	Description

Continued on next page

Table 1.190 – *Continued from previous page*

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.126 ui.TreeTable

1.126.1 Brief description

This is a special container that can contain only ui.TableColumn elements. It is similar to a table, but arranges the items in a hierarchical order and allows to fold and unfold rows.

1.126.2 Inheritance Diagram

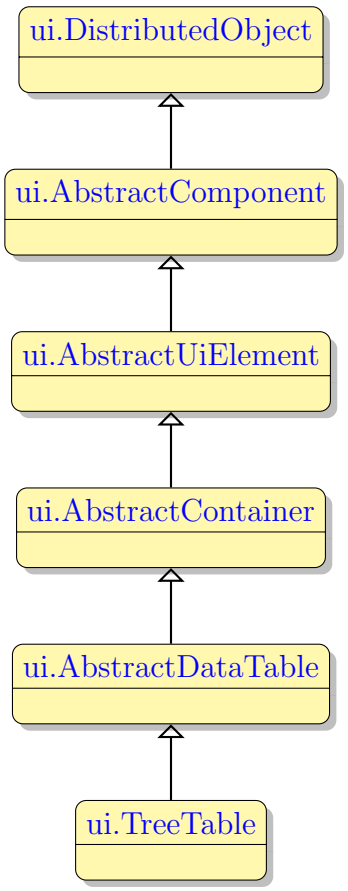

Figure 1.79: Inheritance Diagram of ui.TreeTable

1.126.3 Static Methods Description

Table 1.191: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.127 VerticalAlignment

1.127.1 Brief description

This enum specifies the vertical alignment of a UI element inside a container. It is applicable to UI elements inside any container except coord panel. It defines to which border of the container (or container cell) - top or bottom - the element must adjoin.

1.127.2 Possible values

Table 1.192: Possible values

Value	Description
Default	The window size is the size with which it was opened or which was set after opening by 4GL or graphical theme means.
Stretch	The UI element will be stretched to fit the container (or container cell) without preserving the aspect ratio.
Top	The UI element will be aligned to the top of the container (or container cell).
Center	The UI element will be equidistant from both sides.
Bottom	The UI element will be aligned to the bottom of the container (or container cell).

1.128 VerticalTextAlignment

1.128.1 Brief description

1.128.2 Possible values

Table 1.193: Possible values

Value	Description
Default	The window size is the size with which it was opened or which was set after opening by 4GL or graphical theme means.
Top	The UI element will be aligned to the top of the container (or container cell).
Center	The UI element will be equidistant from both sides.
Bottom	The UI element will be aligned to the bottom of the container (or container cell).

1.129 ViewerType

1.129.1 Brief description

No information

1.129.2 Possible values

Table 1.194: Possible values

Value	Description
Default	The window size is the size with which it was opened or which was set after opening by 4GL or graphical theme means.
ShellOpen	Not described yet
NewWindow	Not described yet
TextViewer	Not described yet
Download	Not described yet
Print	Not described yet
Inject	Not described yet

1.130 ui.WebComponent

1.130.1 Brief description

It is a concrete UI element that serves as a container for third party web components. It is basically just the space which is filled by the web component at runtime.

1.130.2 Inheritance Diagram

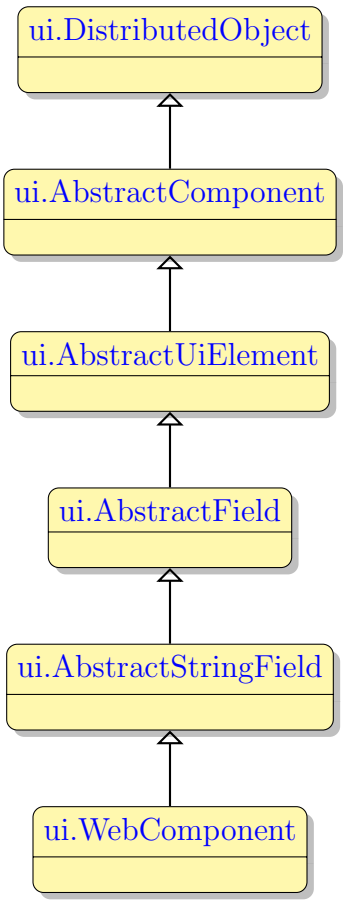

Figure 1.80: Inheritance Diagram of ui.WebComponent

1.130.3 Fields Description

Table 1.195: Fields description

Field Name	Type	Description	Field Accessors
ComponentPath	String	EMPTY.	SetComponentPath, GetComponent-Path
ComponentProperties	list of ui.ComponentProperty	These are specific properties. Their types and number are defines by the HTML file describing the web component.	SetComponentProperties, GetComponent-Properties
ComponentType	String	This is the name of a web component. The web component folder should be located in the components directory on the application server. The HTML file describing the component should be located in the same folder as the component sources and have the same name as the component folder. For example: C:/ProgramDat/Querix/Lycia 6/components/Charts/charts.html - in this case the component type will be 'charts'.	SetComponentType, GetComponent-Type
OnExecuteAction	ui.EventHandler	This event is triggered every time any action is executed on the web component - which means any of the gICAPI methods is invoked.	SetOnExecuteAction, GetOnExecute-Action

1.130.4 Static Methods Description

Table 1.196: Static methods description

Name	Parameters	Description
Create	String identifier, String parent identifier	Creates a UI element with the ID passed as an argument and insert created element to element with ID passed as second argument. Second argument is optional. If parent identifier is empty string, then created element will be inserted to current window root container.

Continued on next page

Table 1.196 – *Continued from previous page*

Name	**Parameters**	**Description**
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.131 ui.Window

1.131.1 Brief description

It is a 4GL window that contains other UI elements at runtime.

1.131.2 Inheritance Diagram

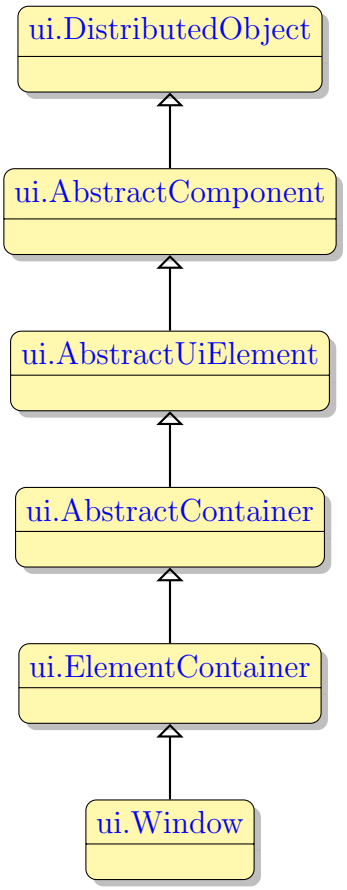

Figure 1.81: Inheritance Diagram of ui.Window

1.131.3 Fields Description

Table 1.197: Fields description

Field Name	Type	Description	Field Accessors

Continued on next page

Table 1.197 – *Continued from previous page*

Field Name	Type	Description	Field Accessors
Close	ui.EventHandler	This event is triggered when the close button on the title bar of a window is pressed.	SetClose, GetClose
CommentLine	String	It specifies the position of the line in a window where the text of the Comment property of a widget is displayed.	SetCommentLine, GetCommentLine
Dependencies	list of ui.AbstractUiElement	No information	SetDependencies, GetDependencies
DisableReverse	BOOLEAN	It negates the effect of the REVERSE 4GL attribute.	SetDisableReverse, GetDisableReverse
ErrorLine	String	It specifies the position of the line in a window where the output of the ERROR statement is displayed.	SetErrorLine, GetErrorLine
FullScreen	BOOLEAN	No information	SetFullScreen, GetFullScreen
HorizontalPadding	Int	EMPTY - not used.	SetHorizontalPadding, GetHorizontalPadding
HorizontalScale	Int	EMPTY - not used.	SetHorizontalScale, GetHorizontalScale
MenuHelpTextLine	String	It specifies the position of the line in a window where the descriptions of the ring menu options are displayed.	SetMenuHelpTextLine, GetMenuHelpTextLine
MessageLine	String	It specifies the position of the line in a window where the output of the MESSAGE statement is displayed.	SetMessageLine, GetMessageLine
NoResize	BOOLEAN	It specifies whether the user will be allowed to resize the window.	SetNoResize, GetNoResize
OnMaximize	ui.EventHandler	This event is triggered when the maximize button on the title bar of a window is pressed.	SetOnMaximize, GetOnMaximize
OnMinimize	ui.EventHandler	This event is triggered when the minimize button on the title bar of a window is pressed.	SetOnMinimize, GetOnMinimize
OnMove	ui.EventHandler	The event is triggered when the position of a UI element is changed.	SetOnMove, GetOnMove

Continued on next page

Table 1.197 – *Continued from previous page*

Field Name	Type	Description	Field Accessors
OnTableBufferLenghtChanged	ui.EventHandler	No information	SetOnTableBufferLenght GetOnTable-Buffer-LenghtChanged
ParentWindow	ui.Window	This is the Window element that was opened (or made current) before the current Window element was opened. It serves as the parent if the window is opened relative to parent.	SetParentWindow, GetParentWindow
RelativeToParent	BOOLEAN	It specifies whether the window will be opened on with its coordinates relative to the window that was opened before it ot relative to the screen.	SetRelativeToParent, GetRelativeTo-Parent
RemoveGridHeadings	BOOLEAN	It indicates whether the grid headings from the tables inside the window would be removed.	SetRemoveGridHeadings, GetRemoveGrid-Headings
StatusInWindow	BOOLEAN	EMPTY - not used.	SetStatusInWindow, GetStatusInWindow
SystemMenuItems	list of ui.SystemMenuItem	It is the list of items belonging to the system context menu.	SetSystemMenuItems, GetSystemMenu-Items
Title	String	This is the inscription attached to the UI element. Usually this is the text of all sorts of labels.	SetTitle, GetTitle
TitleBarIcon	ui.ResourceId	This is the icon to be displayed in the top left corner of a window - at the left end of the title bar.	SetTitleBarIcon, GetTitleBarIcon
TitleBarOptions	ui.TitleBarOptions	These are options aimed at manipulating the window title bar and its buttons.	SetTitleBarOptions, GetTitleBarOptions
TitleJustification	String catalog for TitleJustification	It specifies the horizontal alignment of the text of the title.	SetTitleJustification, GetTitleJustification
VerticalPadding	Int	EMPTY - not used.	SetVerticalPadding, GetVertical-Padding

Continued on next page

Table 1.197 – *Continued from previous page*

Field Name	Type	Description	Field Accessors
VerticalScale	Int	EMPTY - not used.	SetVerticalScale, GetVerticalScale
WindowMenuBar	ui.MenuBar	This is the menu bar of the window used for the top menus (not for the ring menus).	SetWindowMenuBar, GetWindow-MenuBar
WindowRoot	ui.AbstractUiElement	EMPTY.	SetWindowRoot, GetWindowRoot
WindowState	String catalog for WindowState	It defines whether the window is maximized, minimized, etc..	SetWindowState, GetWindowState
WindowStatusBar	ui.StatusBar	This is the status bar of the window.	SetWindowStatusBar, GetWindowSta-tusBar
WindowStyle	String catalog for WindowStyle	It specifies whether the window has a border and title bar, or it is a flat window.	SetWindowStyle, GetWindowStyle
WindowToolbars	list of ui.Toolbar	This are the toolbars displayed in the window.	SetWindowToolbars, GetWindowTool-bars

1.131.4 Static Methods Description

Table 1.198: Static methods description

Name	Parameters	Description
ForName	String identifier	Binds the widget with the ID passed as an argument to a variable of a corresponding data type

1.132 WindowState

1.132.1 Brief description

This enum defines the current state of the window.

1.132.2 Possible values

Table 1.199: Possible values

Value	Description
Default	The window size is the size with which it was opened or which was set after opening by 4GL or graphical theme means.
Minimized	The window is minimized to the task bar because the minimize button was pressed.
Maximized	The window is maximized to take up the whole desktop because the maximize button was pressed.
Hidden	EMPTY - not used.

1.133 WindowStyle

1.133.1 Brief description

This enum defines whether the window is flat or bordered. A window is normally bordered if it has the BORDER 4GL attribute. In this case it has a border, titlebar, statusbar and toolbar (either default or custom). If this attribute is absent, the window is opened inside its parent window (the window that was opened before it) and does not have all the features listed above.

1.133.2 Possible values

Table 1.200: Possible values

Value	Description
Bordered	The window has border and other attributes associated with it.
Flat	The window has no border.

1.134 ui.Wrapper

1.134.1 Brief description

A wrapper is applied to a ui.Table UI element and converts its contents into a chart, a barcode, a picture viewer or to other elements at runtime depending on the contents of the table.

—This type is represented as 4GL record—

1.134.2 Fields Description

Table 1.201: Fields description

Field Name	Type	Description	Field Accessors
Parameter	String	This is the type of the wrapper to be applied to the table.	

1.135 ui.Application

1.135.1 Brief description

This UI entity serves as a parent for the windows, other application elements and general application properties.

1.135.2 Inheritance Diagram

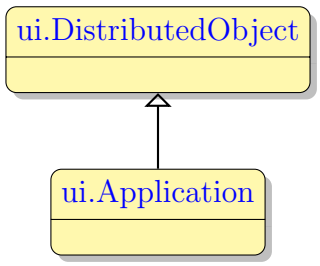

Figure 1.82: Inheritance Diagram of ui.Application

1.135.3 Fields Description

Table 1.202: Fields description

Field Name	Type	Description	Field Accessors
AppContainer	String	This is the name of the MDI container to which the application belongs.	SetAppContainer, GetAppContainer
AppIcon	ui.ResourceId	This is the icon to be displayed in the left side of the application windows titlebar.	SetAppIcon, GetAppIcon
AppTitle	String	This is the application title to be displayed on the titlebar of 4GL screen (the default application window).	SetAppTitle, GetAppTitle
AppType	String catalog for AppType	This is the application type with regard to its role in the MDI interface.	SetAppType, GetAppType
ApplicationMenu	ui.MenuBar	No information	SetApplicationMenu, GetApplication-Menu

Continued on next page

Table 1.202 – *Continued from previous page*

Field Name	Type	Description	Field Accessors
ApplicationStatusBar	ui.StatusBar	No information	SetApplicationStatusBar, GetApplication-StatusBar
ApplicationToolbars	list of ui.Toolbar	No information	SetApplicationToolbars, GetApplication-Toolbars
ClassNames	list of String	The name of a class that is applied to the UI element. There can be a customly created class or one of the default classes. The default classes depend on the 4GL attributes applied to the element by means of the 4GL code or form file and usually specify the colour or intensity attribute.	SetClassNames, GetClassNames
CodeSet	String	This is the definition of the character encoding used for processing text in the application.	SetCodeSet, GetCodeSet
CompatibilityMode	String catalog for CompatibilityMode	No information	SetCompatibilityMode, GetCompatibilityMode
DeviceOrientation	Int	No information	SetDeviceOrientation, GetDeviceOrientation
DisableConsole	BOOLEAN	It disables the console into which goes the output of the DISPLAY statements used without coordinates.	SetDisableConsole, GetDisableConsole
Font	ui.Font	The font to be used for the UI element.	SetFont, GetFont
Host	String	The name or address of the host where the application server is running.	SetHost, GetHost
Identifier	String	It is a unique name of a UI element by which it can be referenced.	SetIdentifier, GetIdentifier
Localizations	list of ui.Localization	No information	SetLocalizations, GetLocalizations

Continued on next page

Table 1.202 – *Continued from previous page*

Field Name	Type	Description	Field Accessors
Margin	ui.Thickness	The space between the border of the UI element and other UI elements surrounding it.	SetMargin, GetMargin
MdiMode	BOOLEAN	Enables or disables MDI interface of the application.	SetMdiMode, GetMdiMode
MdiTaskBarItemHide	BOOLEAN	Hides the taskbar within the MDI container where the child MDI appliations are located and to which they can be minimized.	SetMdiTaskBarItemHide, GetMdiTaskBarItemHide
MenuType	String catalog for MenuType	No information	SetMenuType, GetMenuType
NoScalePixelCoord	BOOLEAN	EMPTY.	SetNoScalePixelCoord, GetNoScalePixelCoord
OnActionEvent	ui.EventHandler	No information	SetOnActionEvent, GetOnActionEvent
OnChildClosed	ui.EventHandler	No information	SetOnChildClosed, GetOnChildClosed
OnIdles	list of ui.OnIdle	No information	SetOnIdles, GetOnIdles
OnNewChild	ui.EventHandler	No information	SetOnNewChild, GetOnNewChild
OnOrientationChanged	ui.EventHandler	No information	SetOnOrientationChange GetOnOrientationChanged
Padding	ui.Thickness	The space between the contents of the UI element (e.g. text in a text field) and the border of this element.	SetPadding, GetPadding
Port	String	The port on the application server on which the application runs.	SetPort, GetPort
PreferredSize	ui.Size	The size of the UI element in pixels that specified by the user that will override the size dynamically calculated at runtime.	SetPreferredSize, GetPreferredSize
ReportViewerConfigs	list of ui.ReportViewerConfig	No information	SetReportViewerConfigs, GetReportViewerConfigs

Continued on next page

Table 1.202 – *Continued from previous page*

Field Name	Type	Description	Field Accessors
ShowSplash	BOOLEAN	Defines whether splash screen should be displayed when the application is launched.	SetShowSplash, GetShowSplash
SplashImage	String	Indicates the image file that should be used as the application splash screen.	SetSplashImage, GetSplashImage
StartMenuShortcut	String	No information	SetStartMenuShortcut, GetStart-MenuShortcut
StartedBy	ui.StartedBy	No information	SetStartedBy, GetStartedBy
SystemTheme	list of Unit	This is the system theme that defines the default application look and feel.	SetSystemTheme, GetSystemTheme
Timeout	Int	The time after which the idling application will terminate.	SetTimeout, Get-Timeout
WindowMenuBar	ui.MenuBar	This is the menu bar of the window used for the top menus (not for the ring menus).	SetWindowMenuBar, GetWindow-MenuBar
Windows	list of ui.Window	This is the list of 4GL window objects.	SetWindows, GetWindows

1.135.4 Static Methods Description

Table 1.203: Static methods description

Name	Parameters	Description
GetCurrent		Binds the Application object with the currently active application