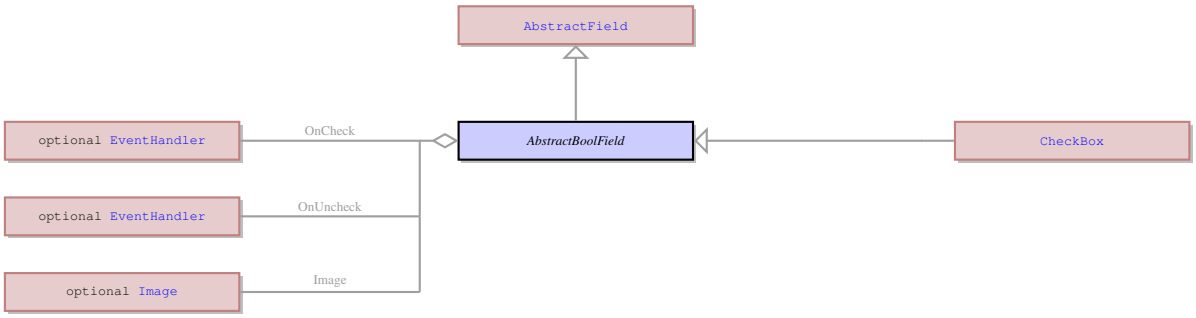
Model's reference

July 3, 2022

Contents

1 AbstractBoolField Not-referenced

1.1 Diagram



1.2 Description

Name: AbstractBoolField

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 .

Parent: AbstractField - 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.

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 .

1.3 Children

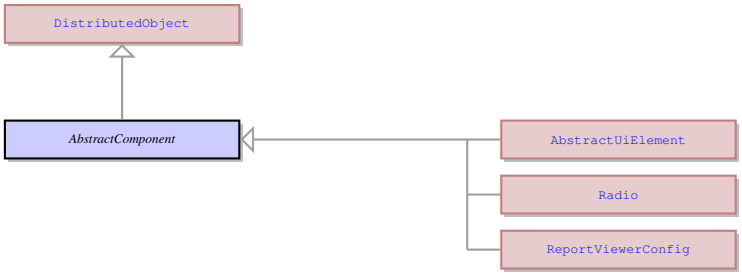
- CheckBox - It is a concrete UI element that consists of a single check box and a label attached to it. It can be in only one of 2 states at a time - either checked or unchecked. Changing of the state can either change the value that will be written to the underlying variable, or trigger an event handler.

1.4 Fields

Name	Type	Description
IsChecked	optional Bool	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.
OnCheck	optional EventHandler	The OnCheck field defines the event which will be triggered if the IsChecked field of the UI element is changed to TRUE.
OnUncheck	optional EventHandler	The OnUncheck field defines the event which will be triggered if the IsChecked field of the UI element is changed to FALSE.
Title	optional String	This is the inscription attached to the UI element. Usually this is the text of all sorts of labels.
Image	optional Image	It is an image that can be applied to other UI elements, e.g. to a button.
AllowNewlines	Bool	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.

2 AbstractComponent Not-referenced

2.1 Diagram



2.2 Description

Name: AbstractComponent
This is the common parent of all UI elements.
Parent: DistributedObject - This is the root of the UI element hierarchy.
This is the common parent of all UI elements.

2.3 Children

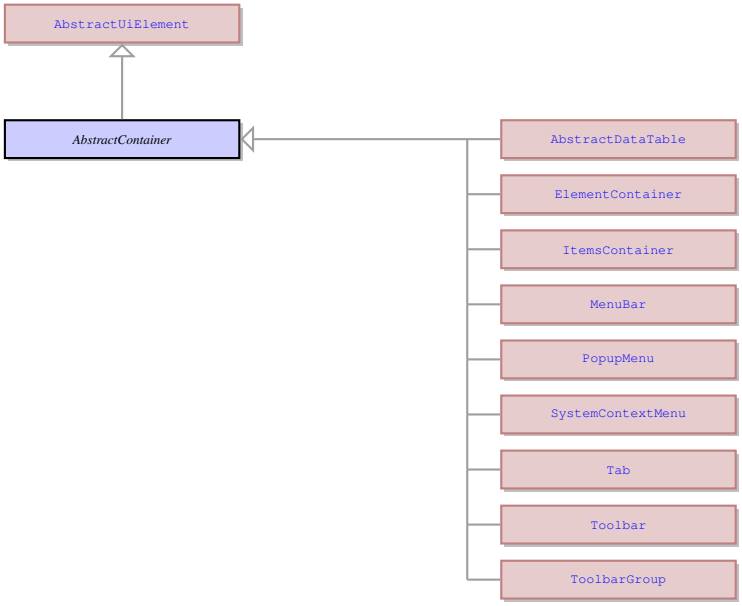
- AbstractUiElement - AbstractUiElement is the base class for UI widgets. It is a generic UI element that can accept user actions. Most of concrete UI elements must inherit the properties and action types from the AbstractUIElement.
- Radio - A Radio is a UI element that can only occur inside a ui.RadioGroup . 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.
- ReportViewerConfig - No information

2.4 Fields

Name	Type	Description
Identifier	String	It is a unique name of a UI element by which it can be referenced.

3 AbstractContainer Not-referenced

3.1 Diagram



3.2 Description

Name: AbstractContainer
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.
Parent: AbstractUiElement - 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.
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.

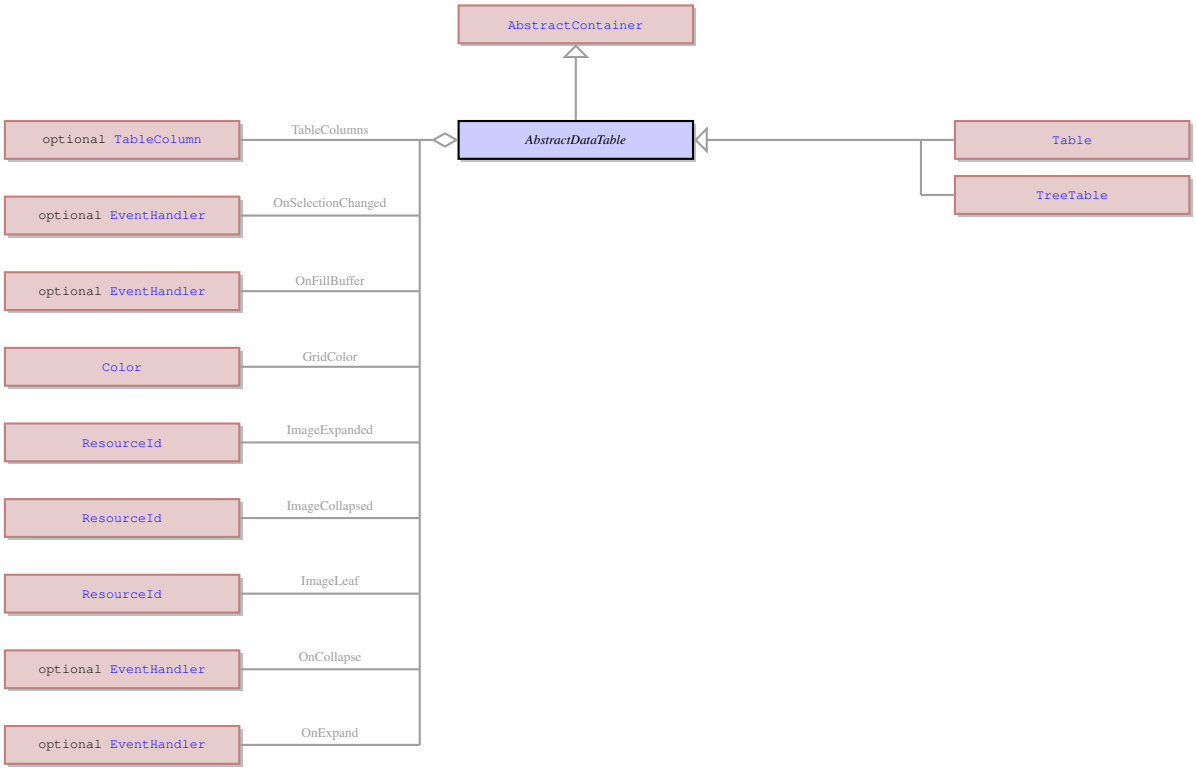
3.3 Children

- AbstractDataTable - 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.
- ElementContainer - 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.

- ItemsContainer - 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.
- MenuBar - This is the area for the top menu (is not applied to ring menus). It includes menu options and menu option groups.
- PopupMenu - 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.
- SystemContextMenu - This is the context menu which is invoked by right-clicking the title bar of the 4GL window.
- Tab - 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.
- Toolbar - This is the container that incorporates toolbar buttons.
- ToolbarGroup - 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.

4 AbstractDataTable Not-referenced

4.1 Diagram



4.2 Description

Name: AbstractDataTable

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.

Parent: AbstractContainer - 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.

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.

4.3 Children

- Table - 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.
- TreeTable - 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.

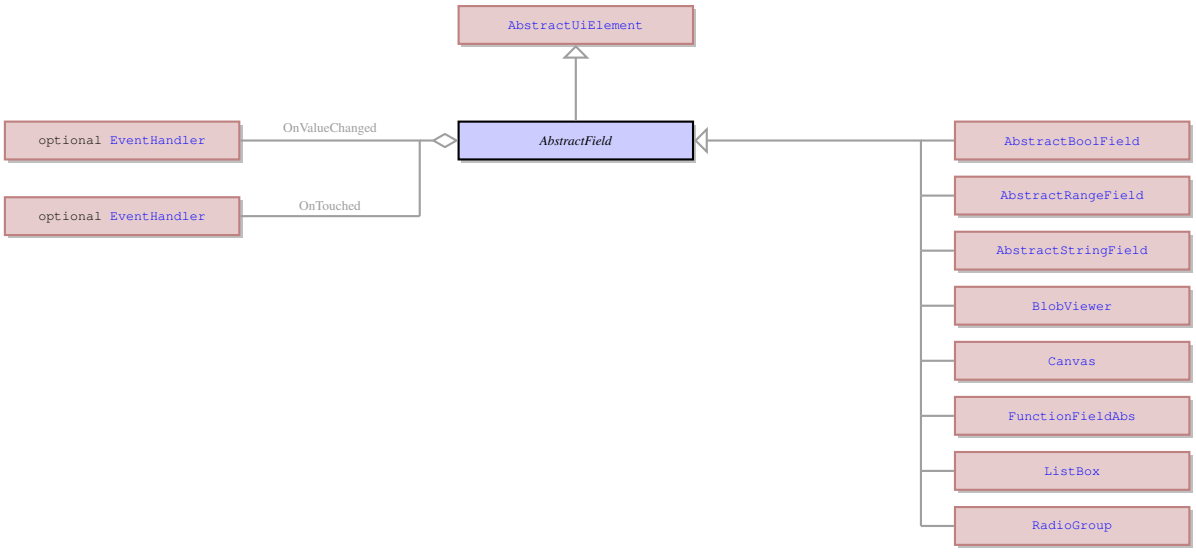
4.4 Fields

Name	Type	Description
TableColumns	list of TableColumn	A set of columns that belong to the same table.
RowHeight	optional String	It defines the default height of a table row in pixels.
IsMultiSelect	Bool	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.
SelectedMany	Bool	No information
OnSelectionChanged	optional 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.
ScrollBarMaxValue	optional 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.
OnFillBuffer	optional EventHandler	If the dialog is using the paged mode, this event is triggered every time a new page is loaded.
BufferLength	optional 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.
GridColor	Color	The color of the grid lines that separate one table cell from the other cells.
MultipleSelect	Bool	No information
FirstRowNum	Int	No information
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.
Indent	optional 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.
ImageExpanded	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.
ImageCollapsed	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.
ImageLeaf	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.
OnCollapse	optional 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).
OnExpand	optional 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).
ColumnParentId	optional 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.
ColumnId	optional 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.
ColumnExpanded	optional 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.

ColumnIsNode	optional 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.
ColumnImage	optional 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.
ColumnEdit	optional 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.

5 AbstractField Not-referenced

5.1 Diagram



5.2 Description

Name: AbstractField

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.

Parent: AbstractUiElement - 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.

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.

5.3 Children

- AbstractBoolField - 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 .
- AbstractRangeField - 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 .
- AbstractStringField - 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.
- BlobViewer - This UI element is used to display and edit BYTE or TEXT values e.g a text or a picture.

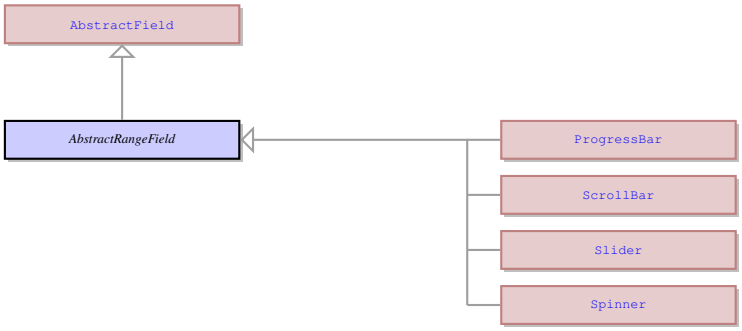
- Canvas - It is a concrete UI element that serves as a container for SVG images and allows interactions with such images.
- FunctionFieldAbs - This UI entity is a function field that is a combination of a text field and a button attached to it. It serves mainly for grouping the button element and the text field element in one object. The properties of the field and button are independent.
- ListBox - 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.
- RadioGroup - 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.

5.4 Fields

Name	Type	Description
ReadOnly	Bool	If enabled, it prevents the user from entering values into the field at runtime even if the field is included into the input routine.
OnValueChanged	optional 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.
OnTouched	optional EventHandler	No information
InvokeAction	optional String	No information
AutoCompleteList	list of String	No information

6 AbstractRangeField Not-referenced

6.1 Diagram



6.2 Description

Name: AbstractRangeField

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 .

Parent: AbstractField - 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.

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 .

6.3 Children

- ProgressBar - 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).
- ScrollBar - 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.

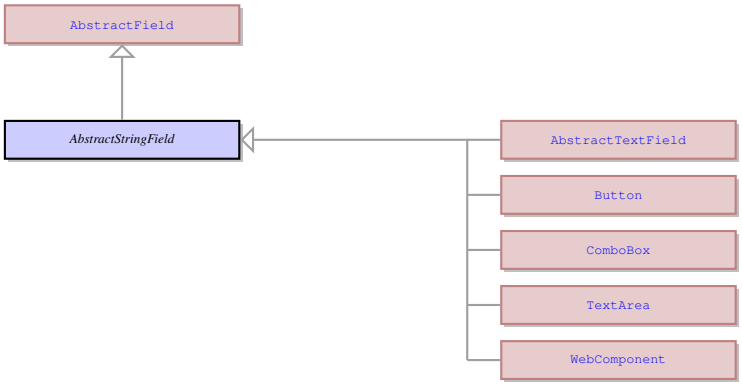
- Slider - This is a concrete UI element that consists of a scale and a slider that can move across this scale. The slider widget has the minimum and maximum value which present the start and the end of the scale. It can be moved directly by the user during the input, or it can be moved if a value within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

6.4 Fields

Name	Type	Description
MinValue	Int	The minimum value in the range of values accepted by a UI element.
MaxValue	Int	The maximum value in the range of values accepted by a UI element.
CurrentValue	optional Int	The value that the UI element has at the moment, it must be within the range of accepted values.

7 AbstractStringField Not-referenced

7.1 Diagram



7.2 Description

Name: AbstractStringField

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.

Parent: AbstractField - 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.

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.

7.3 Children

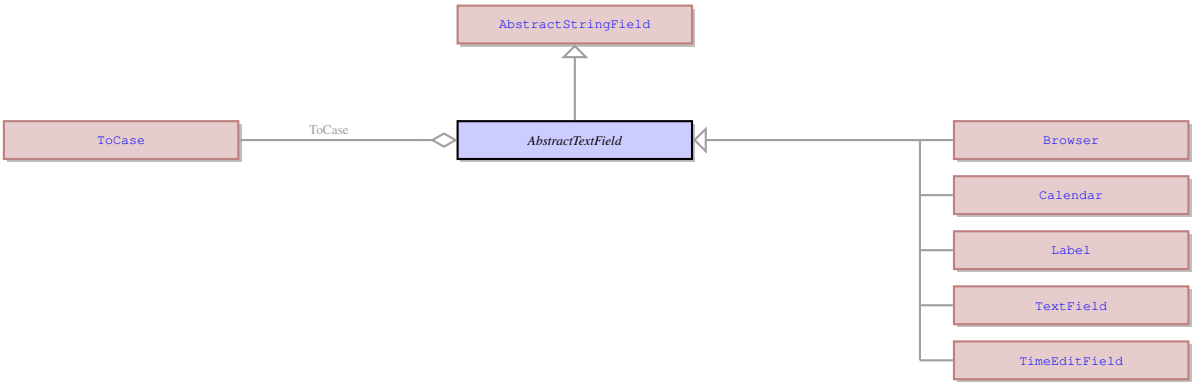
- AbstractTextField - It is an abstract UI element, which unites a subset of ui.AbstractStringField elements with the exception of ui.TextArea , ui.ComboBox , and ui.Button . Typically it includes the UI elements which allow entering values, like normal text fields, and usually are only one line wide.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

7.4 Fields

Name	Type	Description
Text	optional 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.

8 AbstractTextField Not-referenced

8.1 Diagram



8.2 Description

Name: AbstractTextField

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.

Parent: AbstractStringField - 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.

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.

8.3 Children

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

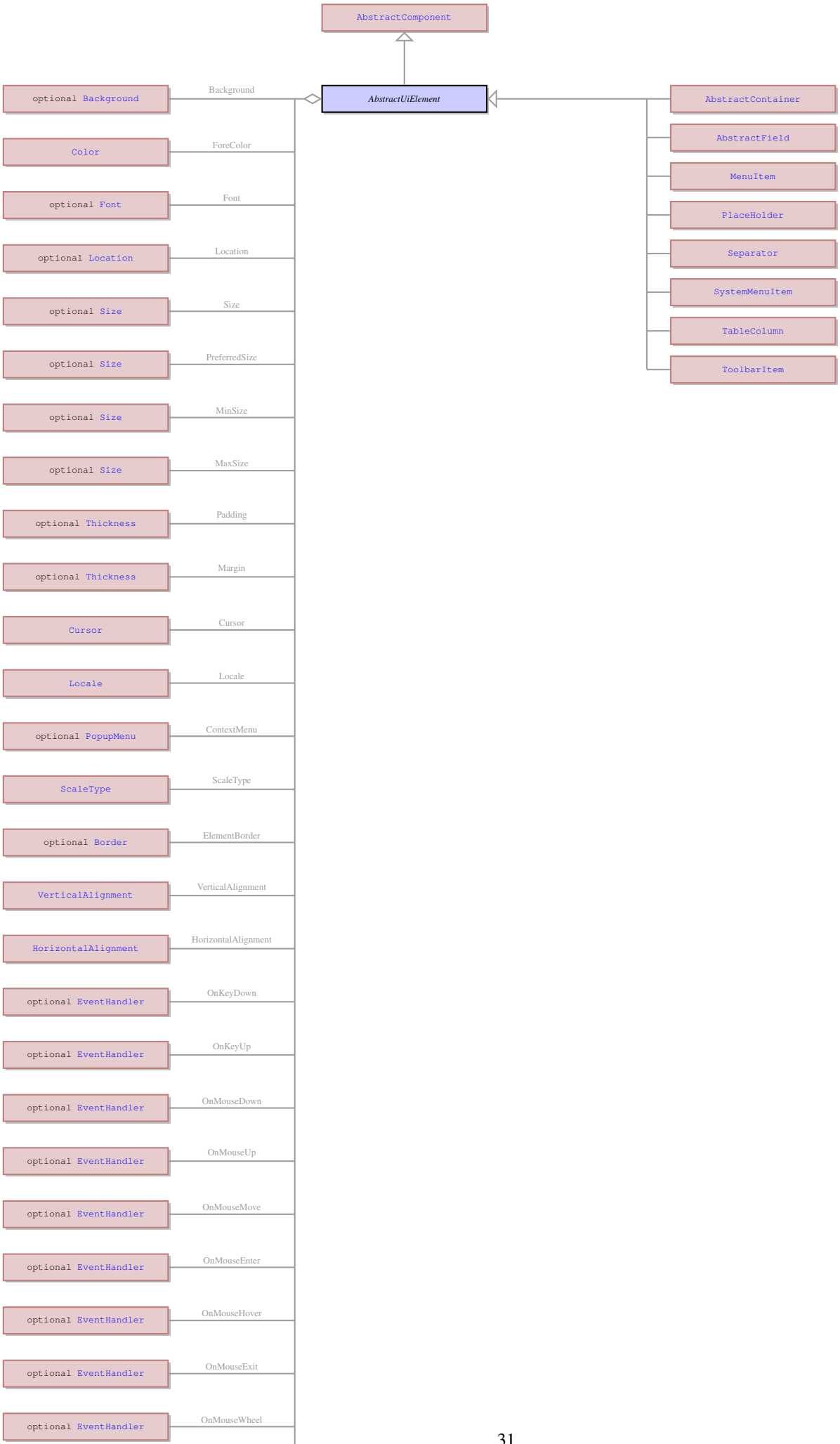
8.4 Fields

Name	Type	Description
IsPasswordMask	Bool	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.
MaxLength	optional 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.
Format	optional 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.

ToCase	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.
TextPicture	optional 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.
Autonext	Bool	If enabled, moves the cursor to the next field during input automatically, when the MaxLength of the current field is met.
Editor	optional String	Specifies the program to be used for opening and editing the BYTE or TEXT value.
Required	Bool	No information

9 AbstractUiElement Not-referenced

9.1 Diagram



9.2 Description

Name: AbstractUiElement

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.

Parent: AbstractComponent - This is the common parent of all UI elements.

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.

9.3 Children

- AbstractContainer - 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.
- AbstractField - 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.
- MenuItem - This UI element serves as the base class for all menu items: menu commands, menu groups, and menu separators.
- PlaceHolder - No information
- Separator - Any kind of separator, e.g. the status bar separator.
- SystemMenuItem - It is a single menu option that belongs the the ui.SystemContextMenu .
- TableColumn - 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.
- ToolbarItem - This is an abstract element that unites the toolbar buttons and toolbar separators.

9.4 Fields

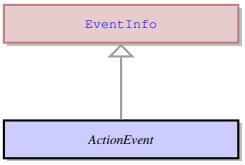
Name	Type	Description
ClassNames	list of ClassName	The name of a class that is applied to the UI element. There can be a customly created class or one of the default classes. The default classes depend on the 4GL attributes applied to the element by means of the 4GL code or form file and usually specify the colour or intensity attribute.
Background	optional Background	Background - defines the background type, color and other parameters.
ForeColor	optional Color	ForeColor - foreground color of the control(used to draw text and/or control border)
Font	optional Font	The font to be used for the UI element.
Location	optional Location	The location of the UI element specified in pixels.
Size	Size	The size of the UI element in pixels that.
PreferredSize	optional Size	The size of the UI element in pixels that specified by the user that will override the size dynamically calculated at runtime.
MinSize	optional Size	The minimum size of the UI element smaller than which an element cannot shrink when resized.
MaxSize	optional Size	The maximum size of the UI element bigger than which an element cannot become when resized.
NotNull	Bool	If enabled, it forbids to save NULL values to the variable linked to the field.
Padding	optional Thickness	The space between the contents of the UI element (e.g. text in a text field) and the border of this element.
Margin	optional Thickness	The space between the border of the UI element and other UI elements surrounding it.
Cursor	Cursor	The type of the cursor that should be applied when the mouse cursor is hovering above the UI Element.
Locale	Locale	The custom locate of the UI element that may be different from the default locale of the application.

Visible	Bool	If enabled, the UI element is visible at runtime. If disabled, it is hidden. The default value is TRUE.
Collapsed	Bool	No information
Enable	Bool	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.
ContextMenu	optional PopupMenu	It contains the information about the context to be displayed when the user right-clicks the UI element at runtime.
ToolTip	optional 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.
TabIndex	optional Int	It specifies the order of the UI elements located on a single form. This order can be used during input for cursor navigation.
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.
EnableBorder	Bool	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.
ScaleType	ScaleType	It defines whether the element contents will be scaled, if the element is resized.
ElementBorder	optional Border	Sets the custom border for a UI element.
VerticalAlignment	VerticalAlignment	Specifies the vertical alignment of the UI element inside its container.
HorizontalAlignment	HorizontalAlignment	Specifies the horizontal alignment of the UI element inside its container.
OnKeyDown	optional 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.
OnKeyUp	optional 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.
OnMouseDown	optional EventHandler	The event specified will be triggered when left mouse button is clicked on the UI element.
OnMouseUp	optional EventHandler	The event specified will be triggered when the left mouse button is released after it was clicked on the UI element.
OnMouseMove	optional EventHandler	The event specified will be triggered when the mouse cursor is moved inside the UI element area.
OnMouseEnter	optional EventHandler	The event specified will be triggered when the mouse cursor enters the UI element area.
OnMouseHover	optional 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.
OnMouseExit	optional EventHandler	The event specified will be triggered when the mouse cursor exits the UI element.
OnMouseWheel	optional EventHandler	The event specified will be triggered when the mouse wheel is rotated while the cursor hovers over the UI element.
OnMouseDoubleClick	optional EventHandler	The event specified will be triggered when the the user double-clicks on the UI element.
OnMouseClick	optional EventHandler	The event specified will be triggered when the the user left-clicks on the UI element.
OnMenuDetect	optional EventHandler	This event is triggered when the user right-clicks the UI element to invoke context menu.
OnDragStart	optional EventHandler	The event is triggered when the user clicks on an element, holds the mouse key and starts moving it away from its location.
OnDragEnter	optional 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.
OnDragOver	optional EventHandler	The event is triggered when the mouse cursor with the item is dragged over a drop target. Typically invoked after OnDragEnter event.
OnDragFinished	optional EventHandler	Triggered after OnDragStart was invoked and then OnDrop executed successfully or the drag and drop action was terminated.

OnDrop	optional 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.
OnResize	optional EventHandler	The event is triggered when the size of a UI element is changed.
OnSelection	optional EventHandler	The event is triggered when a UI element is selected by mouse cursor.
OnFocusIn	optional 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.
OnFocusOut	optional 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.
TextAlignment	optional 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.
Wrapper	optional 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.
ElementRole	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.
IsProtected	Bool	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.
Focusable	Bool	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.
HasFocus	Bool	It indicates that the UI element is selected in the moment and the 4GL cursor is located in it.
TranslateTo	TranslateTo	EMPTY.
BorderPanelItemLocation	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.
GridItemLocation	optional 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.
AllowDrag	Bool	If set to TRUE indicates that the dragging items from the UI element is allowed to perform Drag and Drop activities.
AllowDrop	Bool	If set to TRUE indicates that the dropping items into the UI element is allowed to perform Drag and Drop activities.
TrackSizes	Bool	If set to true, the client tracks any resizing the element might undergo and sends the information about the changes to the server.
TrackLocation	Bool	If set to true, the client tracks any changes of the element location and sends the information about the changes to the server.
StyleClassName	optional 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.
Target	optional String	No information
Comment	optional String	A character string with some sort of description.
Xpath	optional String	No information
FieldTable	String	No information

10 ActionEvent Not-referenced

10.1 Diagram



10.2 Description

Name: ActionEvent

This the event that sends the the action name to the server when it is triggered.

Parent: EventInfo - 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.

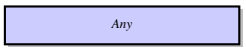
This the event that sends the the action name to the server when it is triggered.

10.3 Fields

Name	Type	Description
ActionName	optional String	It is a string that contains the name of the 4GL action. It can consist of any printable symbols.

11 Any Not-referenced

11.1 Diagram



11.2 Description

Name: Any

EMPTY.

No parents.

EMPTY.

12 AnyElement Not-referenced

12.1 Diagram

12.2 Description

Name: AnyElement

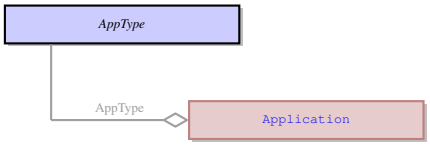
EMPTY.

No parents.

EMPTY.

13 AppType

13.1 Diagram



13.2 Description

Name: AppType

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.

No parents.

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.

13.3 Options

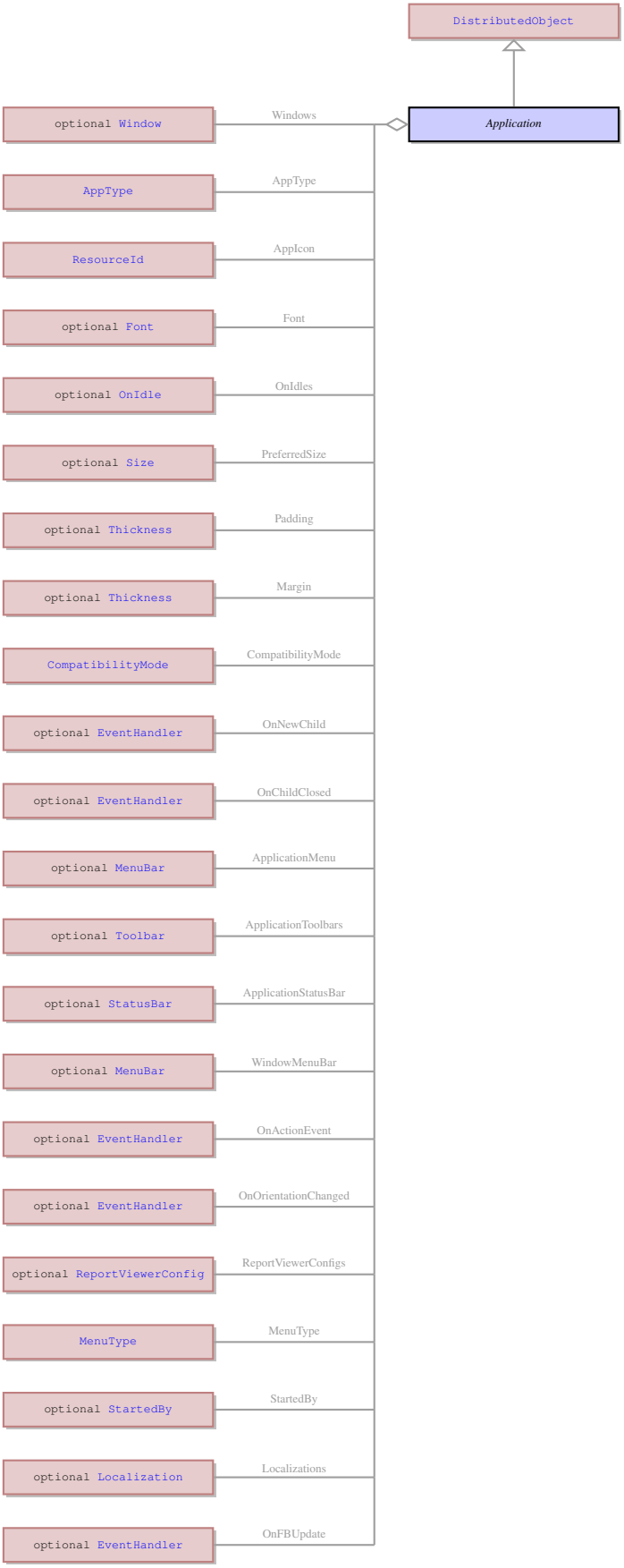
Name	Description
Child	An application launched inside an MDI container as one of its child applications.
Normal	A normal application is an independednt application that is launched outside the MDI interface.
Container	An application launched as an MDI container which can house other applications.

13.4 Referenced in

- AppType field in optional Application - 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.

14 Application Not-referenced

14.1 Diagram



14.2 Description

Name: Application

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

Parent: DistributedObject - This is the root of the UI element hierarchy.

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

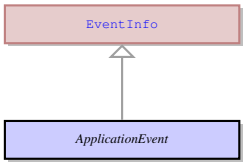
14.3 Fields

Name	Type	Description
Windows	list of Window	This is the list of 4GL window objects.
AppTitle	String	This is the application title to be displayed on the titlebar of 4GL screen (the default application window).
AppType	AppType	This is the application type with regard to its role in the MDI interface.
AppContainer	optional String	This is the name of the MDI container to which the application belongs.
AppIcon	optional ResourceId	This is the icon to be displayed in the left side of the application windows titlebar.
CodeSet	optional String	This is the definition of the character encoding used for processing text in the application.
DisableConsole	Bool	It disables the console into which goes the output of the DISPLAY statements used without coordinates.
Host	optional String	The name or address of the host where the application server is running.
MdiMode	optional Bool	Enables or disables MDI interface of the application.
MdiTaskBarItemHide	Bool	Hides the taskbar within the MDI container where the child MDI appliations are located and to which they can be minimized.
Port	optional String	The port on the application server on which the application runs.
ShowSplash	Bool	Defines whether splash screen should be displayed when the application is launched.
SplashImage	optional String	Indicates the image file that should be used as the application splash screen.
Timeout	optional Int	The time after which the idling application will terminate.
Font	optional Font	The font to be used for the UI element.
SystemTheme	list of SsmStyleSheet	This is the system theme that defines the default application look and feel.
Identifier	String	It is a unique name of a UI element by which it can be referenced.
OnIdles	list of OnIdle	No information
PreferredSize	optional Size	The size of the UI element in pixels that specified by the user that will override the size dynamically calculated at runtime.
NoScalePixelCoord	Bool	EMPTY.
Padding	optional Thickness	The space between the contents of the UI element (e.g. text in a text field) and the border of this element.
Margin	optional Thickness	The space between the border of the UI element and other UI elements surrounding it.
CompatibilityMode	CompatibilityMode	No information
OnNewChild	optional EventHandler	No information
OnChildClosed	optional EventHandler	No information
ApplicationMenu	optional MenuBar	No information
ApplicationToolbars	list of Toolbar	No information
ApplicationStatusBar	optional StatusBar	No information
WindowMenuBar	optional MenuBar	This is the menu bar of the window used for the top menus (not for the ring menus).
OnActionEvent	optional EventHandler	No information
OnOrientationChanged	optional EventHandler	No information
DeviceOrientation	optional Int	No information
ReportViewerConfigs	list of ReportViewerConfig	No information
ClassNames	list of ClassName	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.
MenuType	MenuType	No information

StartMenuShortcut	optional String	No information
StartedBy	optional StartedBy	No information
Localizations	list of Localization	No information
OnFBUpdate	optional EventHandler	No information

15 ApplicationEvent Not-referenced

15.1 Diagram



15.2 Description

Name: ApplicationEvent

No information

Parent: EventInfo - 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.

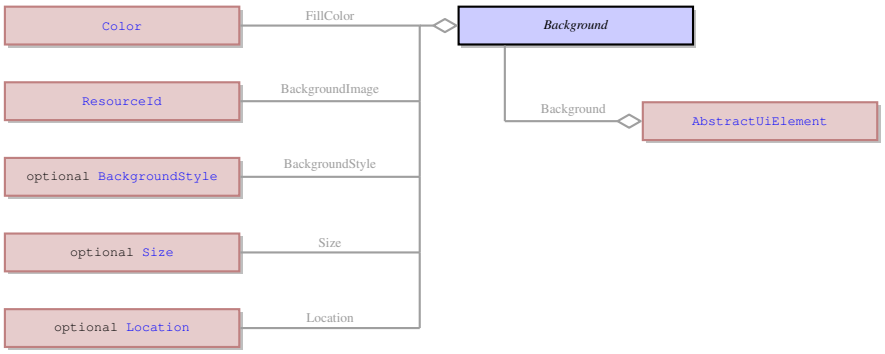
No information

15.3 Fields

Name	Type	Description
Identifier	String	It is a unique name of a UI element by which it can be referenced.
AppTitle	String	This is the application title to be displayed on the titlebar of 4GL screen (the default application window).

16 Background

16.1 Diagram



16.2 Description

Name: Background

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

No parents.

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

16.3 Fields

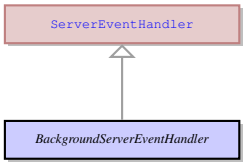
Name	Type	Description
FillColor	optional Color	The color of the background of an element.
BackgroundImage	optional ResourceId	A background image for the UI element.
BackgroundStyle	optional BackgroundStyle	The position of the background image of the UI element.
Size	optional Size	The size of the UI element in pixels that.
Location	optional Location	The location of the UI element specified in pixels.

16.4 Referenced in

- Background field in optional AbstractUiElement - This element determines the colour of the background of an element, the background image, if any, and its properties.

17 BackgroundServerEventHandler Not-referenced

17.1 Diagram



17.2 Description

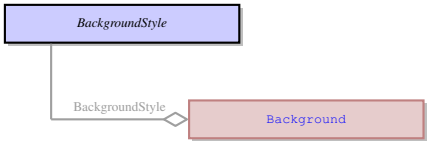
Name: BackgroundServerEventHandler
EMPTY.
Parent: ServerEventHandler - EMPTY.
EMPTY.

17.3 Fields

Name	Type	Description
Selector	optional String	No information
Delegate	Bool	No information

18 BackgroundStyle

18.1 Diagram



18.2 Description

Name: BackgroundStyle
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.
No parents.
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.

18.3 Options

Name	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.

18.4 Referenced in

- BackgroundStyle field in optional Background - 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.

19 BatchEventHandler Not-referenced

19.1 Diagram

19.2 Description

Name: BatchEventHandler

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

Parent: EventHandler - This is common class for all the specific event handler types.

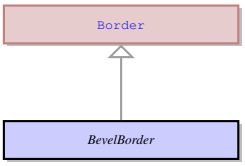
This is an event handler which allows a UI element to have more than one event handler assigned to one event.

19.3 Fields

Name	Type	Description
Handlers	list of EventHandler	A set of event handlers assigned to a single event.

20 BevelBorder Not-referenced

20.1 Diagram



20.2 Description

Name: BevelBorder

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.

Parent: Border - 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 .

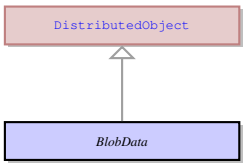
This UI element is used to apply a custom bevel border to any concrete UI element. The border can be lowered or raised, its thickness or colour can be changed.

20.3 Fields

Name	Type	Description
IsRaised	Bool	This property specifies whether custom the bevel or etched border should be raised or lowered.

21 BlobData Not-referenced

21.1 Diagram



21.2 Description

Name: BlobData

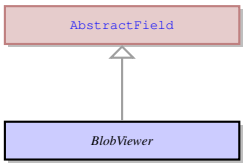
This is a large binary object (a text, an image, etc.) that can be viewed and edited in a ui.BlobViewer .

Parent: DistributedObject - This is the root of the UI element hierarchy.

This is a large binary object (a text, an image, etc.) that can be viewed and edited in a ui.BlobViewer .

22 BlobViewer Not-referenced

22.1 Diagram



22.2 Description

Name: BlobViewer

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

Parent: AbstractField - 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.

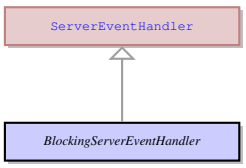
This UI element is used to display and edit BYTE or TEXT values e.g a text or a picture.

22.3 Fields

Name	Type	Description
Editor	optional String	Specifies the program to be used for opening and editing the BYTE or TEXT value.
IsTouched	Bool	It indicates whether the BLOB data in the blob viewed was modified by the user at runtime.

23 BlockingServerEventHandler Not-referenced

23.1 Diagram



23.2 Description

Name: BlockingServerEventHandler

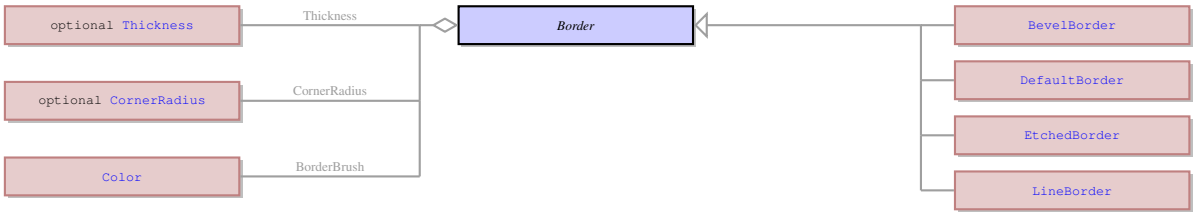
EMPTY.

Parent: ServerEventHandler - EMPTY.

EMPTY.

24 Border Not-referenced

24.1 Diagram



24.2 Description

Name: Border

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 .

No parents.

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 .

24.3 Children

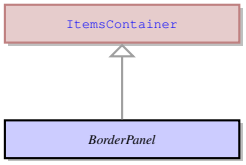
- BevelBorder - This UI element is used to apply a custom bevel border to any concrete UI element. The border can be lowered or raised, its thickness or colour can be changed.
- DefaultBorder - No information
- EtchedBorder - It sets a custom etched border around the UI element. The border can be raised and lowered, its colour can be changed.
- LineBorder - 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.

24.4 Fields

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

25 BorderPanel Not-referenced

25.1 Diagram



25.2 Description

Name: BorderPanel

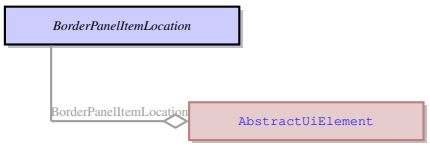
It is a concrete UI element - a container for arranging the layout of other UI elements. Other UI elements can be located either alongside the panel borders or in its center, thus this panel can incorporate up to 5 elements - 1 for each side and 1 in the center. The elements are stretched by default, 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.

Parent: ItemsContainer - 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.

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.

26 BorderPanelItemLocation

26.1 Diagram



26.2 Description

Name: BorderPanelItemLocation

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.

No parents.

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.

26.3 Options

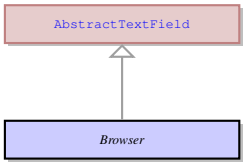
Name	Description
Center	The element is located without adjoining to any of the 4 borders of the container, in the space in the middle of the panel.
Left	The element is located adjoined to the left side of the border panel.
Right	The element is located adjoined to the right side of the border panel.
Top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

26.4 Referenced in

- BorderPanelItemLocation field in optional AbstractUiElement - 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.

27 Browser Not-referenced

27.1 Diagram



27.2 Description

Name: Browser

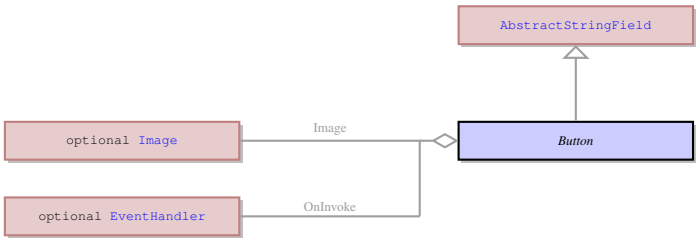
It is a concrete UI element that encompasses a built-in web browser with a somewhat limited functionality. It is used to display web pages, but can also work as a file explorer, display contents of files (e.g. text or image files), etc.

Parent: AbstractTextField - 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.

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.

28 Button Not-referenced

28.1 Diagram



28.2 Description

Name: Button

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.

Parent: AbstractStringField - 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.

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.

28.3 Fields

Name	Type	Description
IsPressed	Bool	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.
IsToggleButton	Bool	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.
Image	optional 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.
OnInvoke	optional 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.
AllowNewlines	Bool	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.

29 Calendar Not-referenced

29.1 Diagram

29.2 Description

Name: Calendar

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

Parent: AbstractTextField - 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.

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

29.3 Fields

Name	Type	Description
DateValue	optional String	A date value in format ”yyyy-mm-dd”.
SystemDate	Bool	It’s set if ’DateValue is in format ”yyyy-mm-dd”.
OnSelectDate	optional 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.)
LabelText	optional String	No information
HelperText	optional String	No information
PlaceholderText	optional String	No information

30 Canvas Not-referenced

30.1 Diagram

30.2 Description

Name: Canvas

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

Parent: AbstractField - 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.

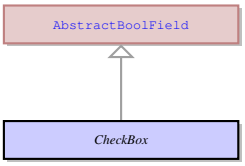
It is a concrete UI element that serves as a container for SVG images and allows interactions with such images.

30.3 Fields

Name	Type	Description
Image	optional Image	It specifies the SVG image that should be displayed to the canvas area.

31 CheckBox Not-referenced

31.1 Diagram



31.2 Description

Name: CheckBox

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.

Parent: AbstractBoolField - 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 .

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.

31.3 Fields

Name	Type	Description
Required	Bool	No information

32 ClearBlob Not-referenced

32.1 Diagram

32.2 Description

Name: ClearBlob

Clears the content of the BlobViewer element specified in the ui.Viewer property.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

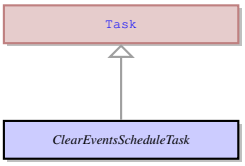
Clears the content of the BlobViewer element specified in the ui.Viewer property.

32.3 Fields

Name	Type	Description
Viewer	optional BlobViewer	The target blob viewer field.

33 ClearEventsScheduleTask Not-referenced

33.1 Diagram



33.2 Description

Name: ClearEventsScheduleTask

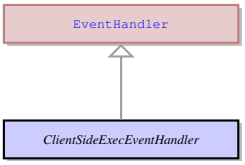
Clears the schedule of events to be handled. Doesn't send any result to server.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

Clears the schedule of events to be handled. Doesn't send any result to server.

34 ClientSideExecEventHandler Not-referenced

34.1 Diagram



34.2 Description

Name: ClientSideExecEventHandler

No information

Parent: EventHandler - This is common class for all the specific event handler types.

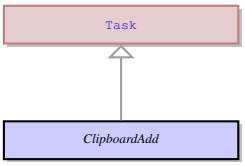
No information

34.3 Fields

Name	Type	Description
ExecCommand	optional String	No information
ExecParam	optional String	No information

35 ClipboardAdd Not-referenced

35.1 Diagram



35.2 Description

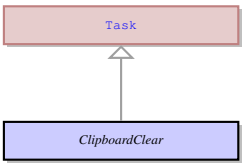
Name: ClipboardAdd
Adds to the content of the clipboard. Sends the execution result in the ui.ClipboardResult object.
Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.
Adds to the content of the clipboard. Sends the execution result in the ui.ClipboardResult object.

35.3 Fields

Name	Type	Description
ClipboardText	optional String	The string to be operated.

36 ClipboardClear Not-referenced

36.1 Diagram

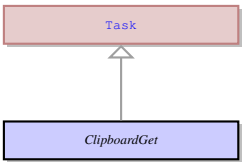


36.2 Description

Name: ClipboardClear
Clears the content of the clipboard. Sends the execution result in the ui.ClipboardResult object.
Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.
Clears the content of the clipboard. Sends the execution result in the ui.ClipboardResult object.

37 ClipboardGet Not-referenced

37.1 Diagram



37.2 Description

Name: ClipboardGet
Gets the content of the clipboard. Sends the text in the clipboard in the ui.ClipboardResult object.
Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.
Gets the content of the clipboard. Sends the text in the clipboard in the ui.ClipboardResult object.

38 ClipboardPaste Not-referenced

38.1 Diagram

38.2 Description

Name: ClipboardPaste

Pastes the content of the clipboard to the current field. Sends the execution result in the ui.ClipboardResult object.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

Pastes the content of the clipboard to the current field. Sends the execution result in the ui.ClipboardResult object.

38.3 Fields

Name	Type	Description
FocusElement	optional AbstractUiElement	The target focused field.

39 ClipboardResult Not-referenced

39.1 Diagram

39.2 Description

Name: ClipboardResult

The result of the clipboard tasks.

No parents.

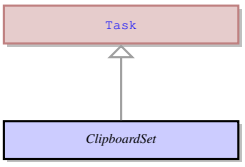
The result of the clipboard tasks.

39.3 Fields

Name	Type	Description
ClipboardText	optional String	The text of the current clipboard.
ExecutionResult	Bool	Indicates whether the operation succeded or failed.

40 ClipboardSet Not-referenced

40.1 Diagram



40.2 Description

Name: ClipboardSet

Sets the content of the clipboard. Sends the execution result in the ui.ClipboardResult object.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

Sets the content of the clipboard. Sends the execution result in the ui.ClipboardResult object.

40.3 Fields

Name	Type	Description
ClipboardText	optional String	The string to be operated.

41 CloseWindow Not-referenced

41.1 Diagram

41.2 Description

Name: CloseWindow

Closes specified window. Doesn't send any result to the server.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

Closes specified window. Doesn't send any result to the server.

41.3 Fields

Name	Type	Description
WindowRef	optional Window	The target window.

42 Color Not-referenced

42.1 Diagram

42.2 Description

Name: Color

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

No parents.

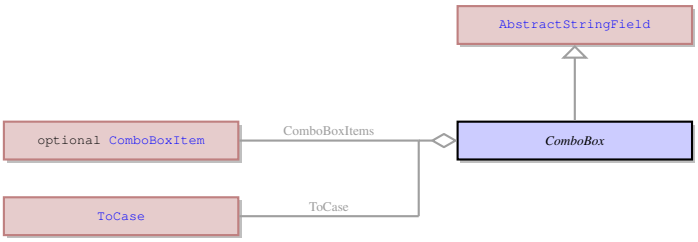
It is the root element to all color properties that can be applied to any UI element.

42.3 Children

- CustomizedColor - This enum defines a custom color in the RGB encoding plus the transparency.
- DefaultColor - No information
- SystemColor - 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.

43 ComboBox Not-referenced

43.1 Diagram



43.2 Description

Name: ComboBox

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.

Parent: AbstractStringField - 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.

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.

43.3 Fields

Name	Type	Description
ComboBoxItems	list of ComboBoxItem	The set values that should be present in the drop-down list of a combo box.
Editable	optional Bool	It indicates that the combo box accepts values that are not in its drop-down list.
ToCase	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.
MaxLength	optional 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.
Autonext	Bool	If enabled, moves the cursor to the next field during input automatically, when the MaxLength of the current field is met.
Required	Bool	No information
LabelText	optional String	No information
HelperText	optional String	No information
SelectedItem	optional Int	No information

44 ComboBoxItem Not-referenced

44.1 Diagram

44.2 Description

Name: ComboBoxItem

It is single item in a combo box drop-down list. If it is selected during input, its value is recorded into the variable linked to the combo box.

No parents.

It is single item in a combo box drop-down list. If it is selected during input, its value is recorded into the variable linked to the combo box.

44.3 Fields

Name	Type	Description
Text	optional String	This is the value of the combobox item, which is recorded to the variable linked to it after the input.
ComboBoxItemValue	optional String	No information

45 Command Not-referenced

45.1 Diagram

45.2 Description

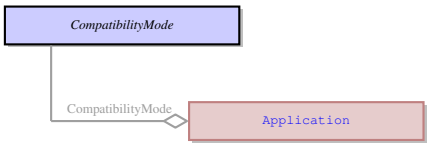
Name: Command
EMPTY.
No parents.
EMPTY.

45.3 Children

- GetResourceChunk - Requests the server to get the next chunk of the resource that is being downloaded.
- MethodCall - EMPTY.

46 CompatibilityMode

46.1 Diagram



46.2 Description

Name: CompatibilityMode
No information
No parents.
No information

46.3 Options

Name	Description
Lycia	Not described yet
Informix4GL	Not described yet
GBDS	Not described yet

46.4 Referenced in

- CompatibilityMode field in optional Application - No information

47 ComponentProperty Not-referenced

47.1 Diagram

47.2 Description

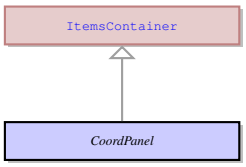
Name: ComponentProperty
This is the property of a ui.WebComponent UI element. Each property is defined by the HTML file that describes the web component.
No parents.
This is the property of a ui.WebComponent UI element. Each property is defined by the HTML file that describes the web component.

47.3 Fields

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

48 CoordPanel Not-referenced

48.1 Diagram



48.2 Description

Name: CoordPanel

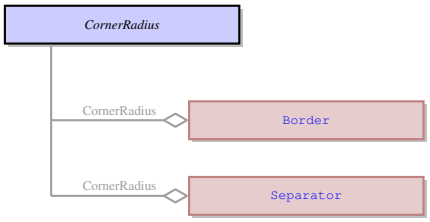
This is a container the location of the elements inside which is determined by the coordinates of the component. The coordinates are stored in pixels and specify the ui.Location on the coord panel where the top left corner of the child element is placed.

Parent: ItemsContainer - 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.

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.

49 CornerRadius

49.1 Diagram



49.2 Description

Name: CornerRadius

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.

No parents.

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.

49.3 Fields

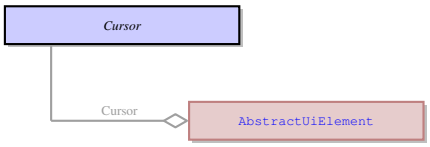
Name	Type	Description
BottomLeft	Float	The bottom left corner of the border frame.
BottomRight	Float	The bottom right corner of the border frame.
TopLeft	Float	The top left corner of the border frame.
TopRight	Float	The top right corner of the border frame.

49.4 Referenced in

- CornerRadius field in optional Border - 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.
- CornerRadius field in optional Separator - 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.

50 Cursor

50.1 Diagram



50.2 Description

Name: Cursor

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.

No parents.

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.

50.3 Options

Name	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.

50.4 Referenced in

- Cursor field in optional AbstractUiElement - 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.

51 CursorPosition Not-referenced

51.1 Diagram

51.2 Description

Name: CursorPosition

The value retrieved by getting the currsor position.

No parents.

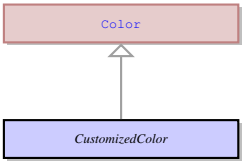
The value retrieved by getting the currsor position.

51.3 Fields

Name	Type	Description
Position	optional Int	The current poaition of the cursor.

52 CustomizedColor Not-referenced

52.1 Diagram



52.2 Description

Name: CustomizedColor

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

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

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

52.3 Fields

Name	Type	Description
RedColor	Int	The value of the red colour in the RGB color model (0-255).
GreenColor	Int	The value of the green colour in the RGB color model (0-255).
BlueColor	Int	The value of the blue colour in the RGB color model (0-255).
Alpha	Int	The value of the transparency applied to the color. 0 - completely transparent. 255 - completely solid color.

53 DDData Not-referenced

53.1 Diagram

53.2 Description

Name: DDData

Drag and Drop data object.

No parents.

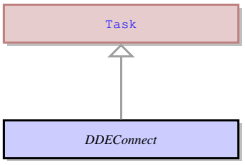
Drag and Drop data object.

53.3 Fields

Name	Type	Description
MimeType	optional String	The mime type of draged data.
Buffer	optional String	The value of draged data.

54 DDEConnect Not-referenced

54.1 Diagram



54.2 Description

Name: DDEConnect

Opens a connection to an application which supports DDE. Passes the result of the operation to ui.IsDDEError of the ui.DDEResult object and sends it to the server.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

Opens a connection to an application which supports DDE. Passes the result of the operation to ui.IsDDEError of the ui.DDEResult object and sends it to the server.

54.3 Fields

Name	Type	Description
ProgName	optional String	The name of the program with which the data exchange is performed.
DocName	optional String	The document name, specifies the name of the file where the data is held.

55 DDEError Not-referenced

55.1 Diagram

55.2 Description

Name: DDEError

A DDE operaation error message.

No parents.

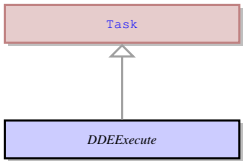
A DDE operaation error message.

55.3 Fields

Name	Type	Description
DDEMessage	optional String	The error message of a DDE operation.

56 DDEExecute Not-referenced

56.1 Diagram



56.2 Description

Name: DDEExecute

Executes a command in the specified document, using the program opened by ui.DDEConnect . Passes the result of the operation to ui.IsDDEError of the ui.DDEResult object and sends it to the server.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

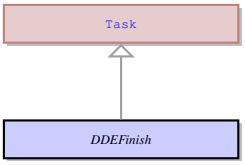
Executes a command in the specified document, using the program opened by ui.DDEConnect . Passes the result of the operation to ui.IsDDEError of the ui.DDEResult object and sends it to the server.

56.3 Fields

Name	Type	Description
ProgName	optional String	The name of the program with which the data exchange is performed.
DocName	optional String	The document name, specifies the name of the file where the data is held.
DdeCommand	optional String	The command to be carried out within the file that has been opened.

57 DDEFinish Not-referenced

57.1 Diagram



57.2 Description

Name: DDEFinish

Closes the connection channel to the program and document. Passes the result of the operation to ui.IsDDEError of the ui.DDEResult object and sends it to the server.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

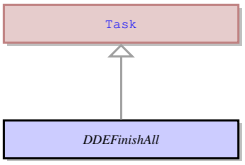
Closes the connection channel to the program and document. Passes the result of the operation to ui.IsDDEError of the ui.DDEResult object and sends it to the server.

57.3 Fields

Name	Type	Description
ProgName	optional String	The name of the program with which the data exchange is performed.
DocName	optional String	The document name, specifies the name of the file where the data is held.

58 DDEFinishAll Not-referenced

58.1 Diagram



58.2 Description

Name: DDEFinishAll

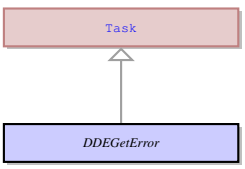
Is used to close all DDE connections, and the program that is being communicated with via DDE. Passes the result of the operation to ui.IsDDEError of the ui.DDEResult object and sends it to the server.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

Is used to close all DDE connections, and the program that is being communicated with via DDE. Passes the result of the operation to ui.IsDDEError of the ui.DDEResult object and sends it to the server.

59 DDEGetError Not-referenced

59.1 Diagram



59.2 Description

Name: DDEGetError

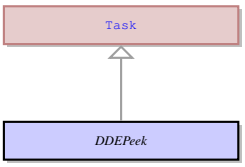
Retrieves the last error recorded for the DDE channel. Sends the result to the ui.DDEError object.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

Retrieves the last error recorded for the DDE channel. Sends the result to the ui.DDEError object.

60 DDEPeek Not-referenced

60.1 Diagram



60.2 Description

Name: DDEPeek

Gets values from a specified place within a specific file. Passes the result of the operation to ui.DDEMessage of the ui.DDEResult object and sends it to the server.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

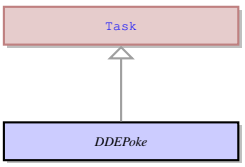
Gets values from a specified place within a specific file. Passes the result of the operation to ui.DDEMessage of the ui.DDEResult object and sends it to the server.

60.3 Fields

Name	Type	Description
ProgName	optional String	The name of the program with which the data exchange is performed.
DocName	optional String	The document name, specifies the name of the file where the data is held.
Cells	optional String	The description of the place from where the data is taken, such as cell names in a spreadsheet.

61 DDEPoke Not-referenced

61.1 Diagram



61.2 Description

Name: DDEPoke

Sends data to the open document, and places it in the specified part of the document. Passes the result of the operation to ui.IsDDEError of ui.DDEResult object and sends it to the server.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

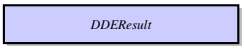
Sends data to the open document, and places it in the specified part of the document. Passes the result of the operation to ui.IsDDEError of ui.DDEResult object and sends it to the server.

61.3 Fields

Name	Type	Description
ProgName	optional String	The name of the program with which the data exchange is performed.
DocName	optional String	The document name, specifies the name of the file where the data is held.
Cells	optional String	The description of the place from where the data is taken, such as cell names in a spreadsheet.
Values	optional String	The data to be inserted to the place defined by ui.Cells .

62 DDEResult Not-referenced

62.1 Diagram



62.2 Description

Name: DDEResult
The result of a DDE operation.
No parents.
The result of a DDE operation.

62.3 Fields

Name	Type	Description
IsDDEError	Bool	Indicates which operation failed.
DDEMessage	optional String	The result message of a DDE operation.

63 DDFeedback Not-referenced

63.1 Diagram

63.2 Description

Name: DDFeedback
Drag and Drop action feedback.
No parents.
Drag and Drop action feedback.

63.3 Options

Name	Description
All	Not described yet
Insert	Not described yet
Select	Not described yet

64 DDOperation Not-referenced

64.1 Diagram

64.2 Description

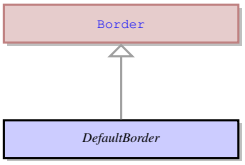
Name: DDOperation
Drag and Drop operation.
No parents.
Drag and Drop operation.

64.3 Options

Name	Description
None	The property is not applied and the default behaviour is used.
Copy	Not described yet
Move	Not described yet
Cancel	Not described yet

65 DefaultBorder Not-referenced

65.1 Diagram

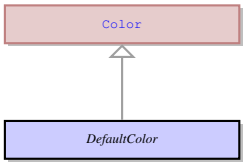


65.2 Description

Name: DefaultBorder
No information
Parent: Border - 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 .
No information

66 DefaultColor Not-referenced

66.1 Diagram

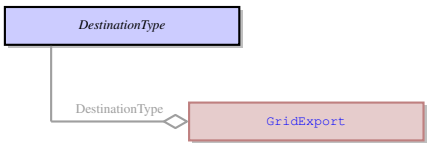


66.2 Description

Name: DefaultColor
No information
Parent: Color - It is the root element to all color properties that can be applied to any UI element.
No information

67 DestinationType

67.1 Diagram



67.2 Description

Name: DestinationType
Specifies where the contents of a grid should be saved to.
No parents.
Specifies where the contents of a grid should be saved to.

67.3 Options

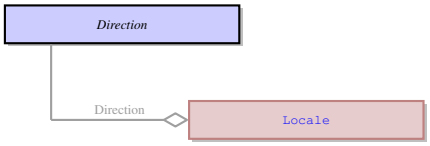
Name	Description
File	Indicates that the grid contents should be saved into a file.
Clipboard	Indicates that the grid contents should be saved to the clipboard.

67.4 Referenced in

- DestinationType field in optional GridExport - Specifies where the contents of a grid should be saved to.

68 Direction

68.1 Diagram



68.2 Description

Name: Direction

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

No parents.

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

68.3 Options

Name	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.

68.4 Referenced in

- Direction field in Locale - This enum defines the direction of the text: left to right or right to left.

69 DisplayFileDialog Not-referenced

69.1 Diagram

69.2 Description

Name: DisplayFileDialog

Calls a message box dialog allowing a user to save or open a particular file. Sends the result to the server in the ui.MessageBoxResult object.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

Calls a message box dialog allowing a user to save or open a particular file. Sends the result to the server in the ui.MessageBoxResult object.

69.3 Fields

Name	Type	Description
FileDialog	optional FileDialog	A file dialog which offers default tools for managing files (save, open, etc.).

70 DistributedModelItem Not-referenced

70.1 Diagram

70.2 Description

Name: DistributedModelItem

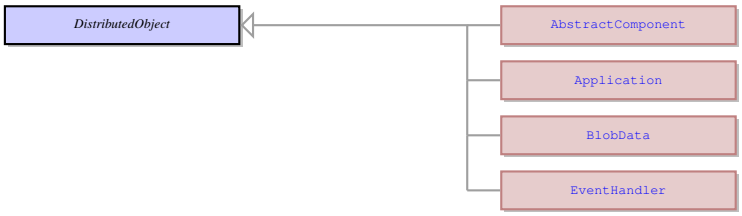
EMPTY.

No parents.

EMPTY.

71 DistributedObject Not-referenced

71.1 Diagram



71.2 Description

Name: DistributedObject

This is the root of the UI element hierarchy.

No parents.

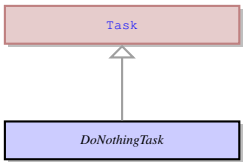
This is the root of the UI element hierarchy.

71.3 Children

- AbstractComponent - This is the common parent of all UI elements.
- Application - This UI entity serves as a parent for the windows, other application elements and general application properties.
- BlobData - This is a large binary object (a text, an image, etc.) that can be viewed and edited in a ui.BlobViewer .
- EventHandler - This is common class for all the specific event handler types.

72 DoNothingTask Not-referenced

72.1 Diagram



72.2 Description

Name: DoNothingTask

This task is necessary solely to synchronize the client state with the server state, i.e., in case when all the server updates should be automatically passed to the client. Doesn't send any result to server.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

This task is necessary solely to synchronize the client state with the server state, i.e., in case when all the server updates should be automatically passed to the client. Doesn't send any result to server.

73 DownloadBlob Not-referenced

73.1 Diagram

73.2 Description

Name: DownloadBlob

Downloads a file in the binary format from the server and displays it to the Blob Viewer element.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

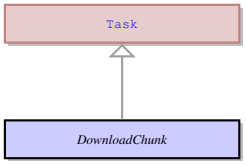
Downloads a file in the binary format from the server and displays it to the Blob Viewer element.

73.3 Fields

Name	Type	Description
Viewer	optional BlobViewer	The target blob viewer field.

74 DownloadChunk Not-referenced

74.1 Diagram



74.2 Description

Name: DownloadChunk

Downloads chunk of file with specified size. Doesn't send any result to server.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

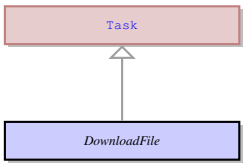
Downloads chunk of file with specified size. Doesn't send any result to server.

74.3 Fields

Name	Type	Description
ChunkSize	Int	The size of the chunk which is available for downloading.
IsEof	Bool	Indicates whether the current chunk is the last one.

75 DownloadFile Not-referenced

75.1 Diagram



75.2 Description

Name: DownloadFile

Downloads a file in the binary format from the server and saves it to the path specified in the ui.ClientPath property .

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

Downloads a file in the binary format from the server and saves it to the path specified in the ui.ClientPath property .

75.3 Fields

Name	Type	Description
ClientPath	String	The target file path.

76 DownloadResources Not-referenced

76.1 Diagram

76.2 Description

Name: DownloadResources

Downloads resources from the specified list. Doesn't send any result to server.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

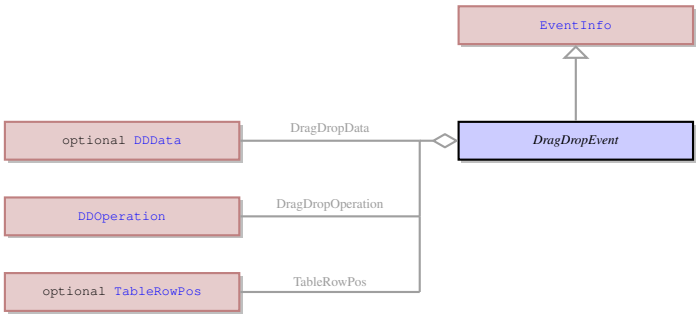
Downloads resources from the specified list. Doesn't send any result to server.

76.3 Fields

Name	Type	Description
Resources	list of ResourceId	The list of the resources that should be loaded.
ShowProgress	Bool	Indicates whether the proggres bar will show the downloading progres.

77 DragDropEvent Not-referenced

77.1 Diagram



77.2 Description

Name: DragDropEvent

Event Info of Drag and Drop events.

Parent: EventInfo - 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.

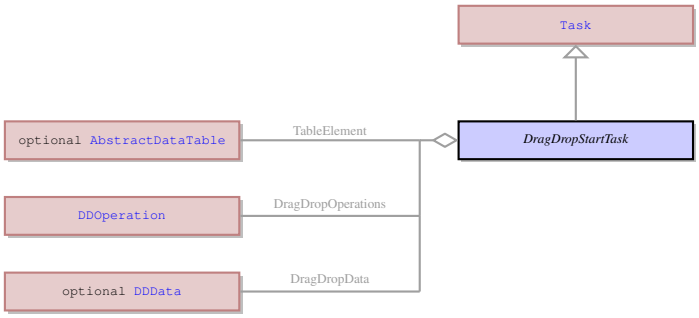
Event Info of Drag and Drop events.

77.3 Fields

Name	Type	Description
DragDropData	optional DDData	The droped data.
MimeTypes	optional String	Mime types which droped data object contains.
DragDropOperation	optional DDOperation	Finel droped operation.
TableRowPos	optional TableRowPos	No information

78 DragDropStartTask Not-referenced

78.1 Diagram



78.2 Description

Name: DragDropStartTask

This task comes to client as answer for event ui.OnDragStart and informs the client that Drag and Drop operation is allowed and can be performed.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

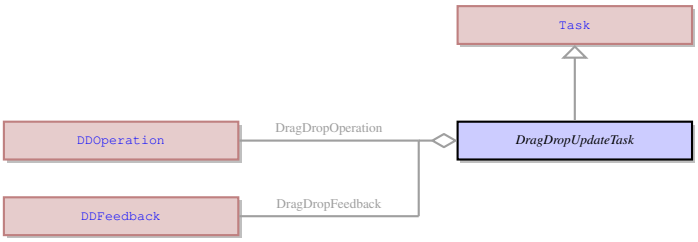
This task comes to client as answer for event ui.OnDragStart and informs the client that Drag and Drop operation is allowed and can be performed.

78.3 Fields

Name	Type	Description
TableElement	AbstractDataTable	Target Table/TreeTable element.
DragDropOperations	list of DDOperation	List of Drag and Drop operations that are allowed.
DragDropData	optional DDData	Drag & Drop buffer data.

79 DragDropUpdateTask Not-referenced

79.1 Diagram



79.2 Description

Name: DragDropUpdateTask

This task comes to client as answer for events ui.OnDragEnter and ui.OnDragOver if it needs to update Drag And Drop action's preview (feetback).

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

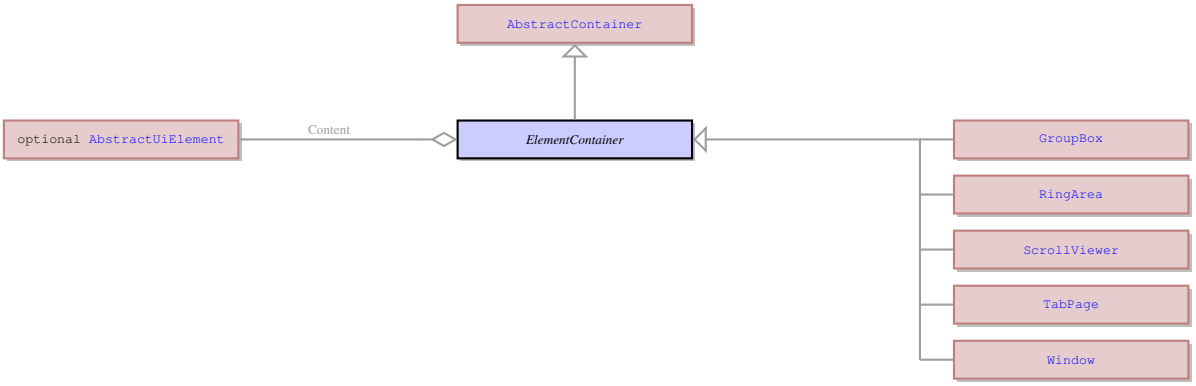
This task comes to client as answer for events ui.OnDragEnter and ui.OnDragOver if it needs to update Drag And Drop action's preview (feetback).

79.3 Fields

Name	Type	Description
DragDropOperation	optional DDOperation	Current Drag and Drop operation.
DragDropFeedback	optional DDFeedback	Current Drag and Drop feedback.
MimeType	optional String	The mime type which should be got from droped data.

80 ElementContainer Not-referenced

80.1 Diagram



80.2 Description

Name: ElementContainer

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.

Parent: AbstractContainer - 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.

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.

80.3 Children

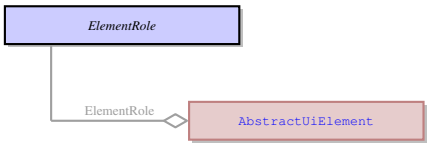
- GroupBox - It is a container that groups the UI elements inside a visible border with an optional title at the top. It can contain only one other UI element. It can be another container or a form widget. Thus though it can encompass UI elements of the ui.AbstractField group, having only one element of this group in a container makes little sense. So it should include one of the other containers first.
- RingArea - This is the area that incorporates ring menu and its options. It must not be mistaken with the MenuBar used for top menu.
- ScrollViewer - 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.
- TabPage - 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.
- Window - It is a 4GL window that contains other UI elements at runtime.

80.4 Fields

Name	Type	Description
Content	AbstractUiElement	It specifies the UI element that is located inside the ElementContainer.

81 ElementRole

81.1 Diagram



81.2 Description

Name: 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.

No parents.

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.

81.3 Options

Name	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.

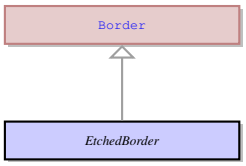
Input	The UI element takes part in a INPUT statement.
Message	The UI element is a result of the MESSAGE statement.
Error	The UI element is a result of the ERROR statement.
Comment	The UI element is a result of the COMMENT property of a widget is displayed.
Prompt	The UI element takes part in a PROMPT statement.
Form	The UI element is a form.
RingMenu	Not described yet

81.4 Referenced in

- ElementRole field in optional AbstractUiElement - 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.

82 EtchedBorder Not-referenced

82.1 Diagram



82.2 Description

Name: EtchedBorder

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

Parent: Border - 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 .

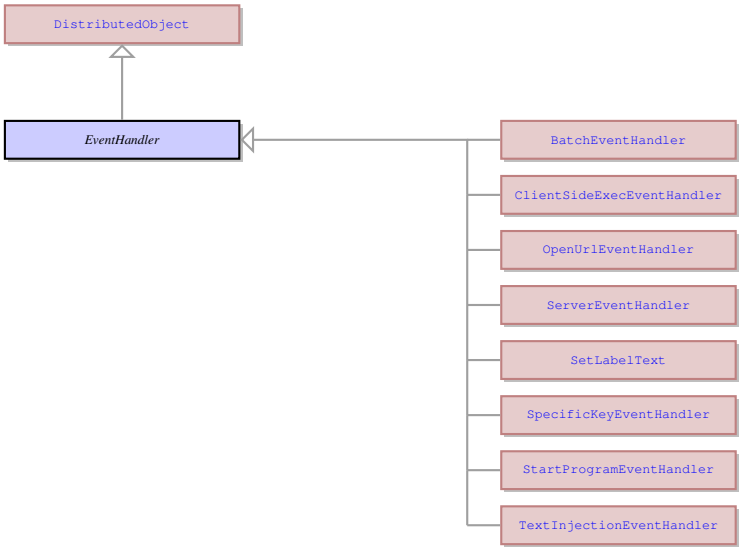
It sets a custom etched border around the UI element. The border can be raised and lowered, its colour can be changed.

82.3 Fields

Name	Type	Description
IsRaised	Bool	This property specifies whether custom the bevel or etched border should be raised or lowered.

83 EventHandler Not-referenced

83.1 Diagram



83.2 Description

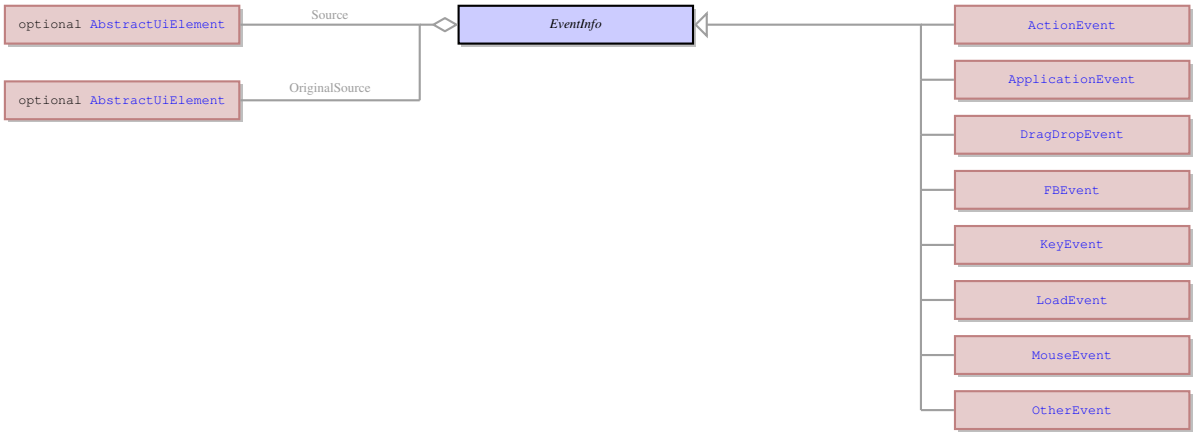
Name: EventHandler
This is common class for all the specific event handler types.
Parent: DistributedObject - This is the root of the UI element hierarchy.
This is common class for all the specific event handler types.

83.3 Children

- BatchEventHandler - This is an event handler which allows a UI element to have more than one event handler assigned to one event.
- ClientSideExecEventHandler - No information
- OpenUrlEventHandler - This is an event handler that can be assigned to any event. This handler opens the URL specified in the default system web browser.
- ServerEventHandler - EMPTY.
- SetLabelText - 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.
- SpecificKeyEventHandler - This event handler specifies what event handler should be triggered when a specific key is pressed. It links the keypress with a 4GL event.
- StartProgramEventHandler - 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.
- TextInjectionEventHandler - This event handler injects the text specified as its parameter into the current input widget. It can be assigned to any event.

84 EventInfo Not-referenced

84.1 Diagram



84.2 Description

Name: EventInfo
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.
No parents.
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.

84.3 Children

- ActionEvent - This the event that sends the the action name to the server when it is triggered.
- ApplicationEvent - No information
- DragDropEvent - Event Info of Drag and Drop events.
- FBEvent - No information

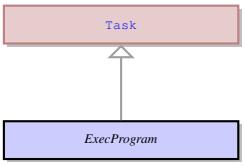
- KeyEvent - It is an event that is triggered when the specified key on the keyboard is pressed. This event is sent to the Application server on the keypress.
- LoadEvent - This is th event info that is sent to server when a virtual table triggers OnFillBuffer event.
- MouseEvent - This is event information that describes an event triggered by mouse. It is sent to the server when events like OnMouseClick or other mouse events are invoked.
- OtherEvent - This event information described the source of the event - i.e. the widget which triggered the event (Radio - CheckedChanged, CheckBox - CheckedChanged, ComboBox - DropDown, etc.)

84.4 Fields

Name	Type	Description
Source	optional 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.
OriginalSource	optional AbstractUiElement	No information

85 ExecProgram Not-referenced

85.1 Diagram



85.2 Description

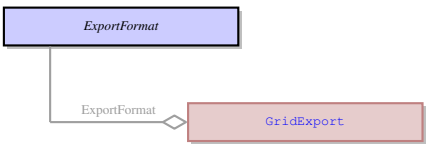
Name: ExecProgram
Launches a specified 4gl program using the same client. Doesn't send any result to server.
Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.
Launches a specified 4gl program using the same client. Doesn't send any result to server.

85.3 Fields

Name	Type	Description
ProgramName	optional String	The 4gl application name.
ProgramServer	optional String	The name of the host - the application server on which the program is deployed and should run.
ProgramPort	optional String	The port on the application server.
UserId	optional String	The name of the user who runs the application.
DoWait	Bool	Forces the client to wait till the procces is finished.

86 ExportFormat

86.1 Diagram



86.2 Description

Name: ExportFormat
Indicates the fomat in which the data from a table should be saved to an external location. Two export formats are available: HTML and CSV.
No parents.
Indicates the fomat in which the data from a table should be saved to an external location. Two export formats are available: HTML and CSV.

86.3 Options

Name	Description
Html	Indicates that the data format should be HTML - values are represented with html tags.
Csv	Indicates that the data format should be CSV - the delimiter is comma (,).

86.4 Referenced in

- ExportFormat field in optional GridExport - Indicates the fomat in which the data from a table should be saved to an external location. Two export formats are available: HTML and CSV.

87 FBEvent Not-referenced

87.1 Diagram

87.2 Description

Name: FBEvent

No information

Parent: EventInfo - 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.

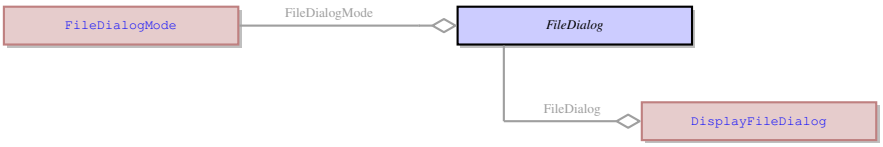
No information

87.3 Fields

Name	Type	Description
FbTarget	optional AbstractUiElement	No information
FbEntity	optional String	No information
FbElPos	optional Int	No information

88 FileDialog

88.1 Diagram



88.2 Description

Name: FileDialog

A file dialog which offers default tools for managing files (save, open, etc.).

No parents.

A file dialog which offers default tools for managing files (save, open, etc.).

88.3 Fields

Name	Type	Description
FileDialogMode	FileDialogMode	Indicates the purpose of the file dialog - defines what actions can be performed with files in this dialog.
IsMultiSelect	Bool	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.

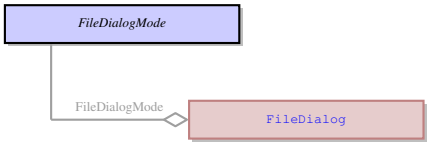
Title	optional String	This is the inscription attached to the UI element. Usually this is the text of all sorts of labels.
Path	optional String	This the absolute or relative path.
FileName	String	Destination file name/path.
FileFilter	optional String	This is the list of allowed file extensions separated by comma.

88.4 Referenced in

- FileDialog field in optional DisplayFileDialog - A file dialog which offers default tools for managing files (save, open, etc.).

89 FileDialogMode

89.1 Diagram



89.2 Description

Name: FileDialogMode

Indicates the purpose of the file dialog - defines what actions can be performed with files in this dialog.
No parents.
Indicates the purpose of the file dialog - defines what actions can be performed with files in this dialog.

89.3 Options

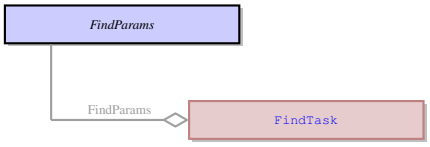
Name	Description
Open	Open file mode - allows to open existing files.
Save	Save file mode - allows to save edited or new files.
OpenDir	The file path at which the file dialog will be opened.

89.4 Referenced in

- FileDialogMode field in optional FileDialog - Indicates the purpose of the file dialog - defines what actions can be performed with files in this dialog.

90 FindParams

90.1 Diagram



90.2 Description

Name: FindParams

No information
No parents.
No information

90.3 Fields

Name	Type	Description
FindValue	optional String	No information
FindColumn	optional String	No information
FindIgnoreCase	Bool	No information
FindWrapAround	Bool	No information

90.4 Referenced in

- FindParams field in optional FindTask - No information

91 FindTask Not-referenced

91.1 Diagram

91.2 Description

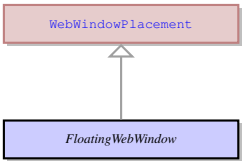
Name: FindTask
No information
Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.
No information

91.3 Fields

Name	Type	Description
FindParams	FindParams	No information
ColumnNames	list of String	No information

92 FloatingWebWindow Not-referenced

92.1 Diagram

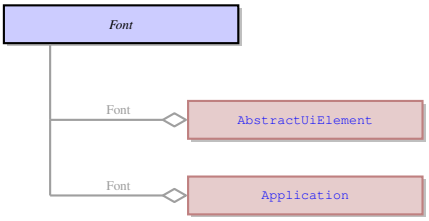


92.2 Description

Name: FloatingWebWindow
This is a type of a 4GL window in a web client when the window can be moved around inside its container (i.e. inside a page of a web browser).
Parent: WebWindowPlacement - No information
This is a type of a 4GL window in a web client when the window can be moved around inside its container (i.e. inside a page of a web browser).

93 Font

93.1 Diagram



93.2 Description

Name: Font

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

No parents.

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

93.3 Fields

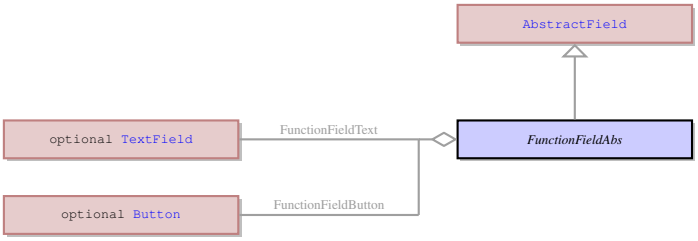
Name	Type	Description
Family	non-empty list of Name	This is the name of the font. E.g. Arial or Tahoma.
Bold	optional Bool	It indicates whether the text should be bold.
Italic	optional Bool	It indicates whether the text should be in italics.
Underline	optional Bool	It indicates whether the text should be underlined.
FontSize	optional Int	It specifies the font size.

93.4 Referenced in

- Font field in optional AbstractUiElement - The font to be used for any text that is a part of the UI element - either label ot inputted text.
- Font field in optional Application - The font to be used for any text that is a part of the UI element - either label ot inputted text.

94 FunctionFieldAbs Not-referenced

94.1 Diagram



94.2 Description

Name: FunctionFieldAbs

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.

Parent: AbstractField - 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.

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.

94.3 Fields

Name	Type	Description
FunctionFieldText	optional TextField	It is the text field widget that is included into a function field.
FunctionFieldButton	optional Button	It is the button widget that is included into a function field.

95 GetChildCountResult Not-referenced

95.1 Diagram

95.2 Description

Name: GetChildCountResult

The result of the ui.GetChildCountTask task.

No parents.

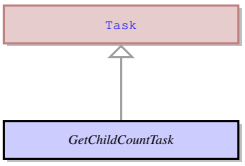
The result of the ui.GetChildCountTask task.

95.3 Fields

Name	Type	Description
ChildCount	Int	The number of children applications.

96 GetChildCountTask Not-referenced

96.1 Diagram



96.2 Description

Name: GetChildCountTask

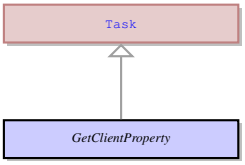
Calculates the number of children in specefied parent container. Sends result to server in the ui.GetChildCountResult object.
Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.
Calculates the number of children in specefied parent container. Sends result to server in the ui.GetChildCountResult object.

96.3 Fields

Name	Type	Description
ChildName	optional String	The child application name.

97 GetClientProperty Not-referenced

97.1 Diagram



97.2 Description

Name: GetClientProperty

Returns the value of the required property an sends the result in the ui.ResultValue object.
Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.
Returns the value of the required property an sends the result in the ui.ResultValue object.

97.3 Fields

Name	Type	Description
ClientPropertyType	String	The type of the property.
ClientPropertyName	String	The actual name of the property.
ClientPropertyValue	String	The value, can be a specific location/file.

98 GetContainerResult Not-referenced

98.1 Diagram

98.2 Description

Name: GetContainerResult

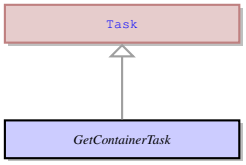
The result of the ui.GetContainerTask task.
No parents.
The result of the ui.GetContainerTask task.

98.3 Fields

Name	Type	Description
ContainerIdentifier	String	The parent container identifier.

99 GetContainerTask Not-referenced

99.1 Diagram



99.2 Description

Name: GetContainerTask

Gets the parent container of the application. Sends the result to server in the ui.GetContainerResult object.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

Gets the parent container of the application. Sends the result to server in the ui.GetContainerResult object.

100 GetCursor Not-referenced

100.1 Diagram

100.2 Description

Name: GetCursor

Gets the position of the cursor in the specified field. Sends the result to server in ui.CursorPosition object.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

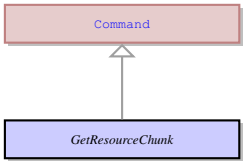
Gets the position of the cursor in the specified field. Sends the result to server in ui.CursorPosition object.

100.3 Fields

Name	Type	Description
Source	optional AbstractUiElement	Source UI field.

101 GetResourceChunk Not-referenced

101.1 Diagram



101.2 Description

Name: GetResourceChunk

Requests the server to get the next chunk of the resource that is being downloaded.

Parent: Command - EMPTY.

Requests the server to get the next chunk of the resource that is being downloaded.

101.3 Fields

Name	Type	Description
FileName	String	Destination file name/path.
ChunkSize	Int	The size of the chunk which is available for downloading.
ChunkShift	optional Int	The index of the first byte of chunk in the file.

102 GetSelectionEnd Not-referenced

102.1 Diagram

102.2 Description

Name: GetSelectionEnd

Gets the position of the last selected character in the specified field. Sends the result in ui.IntResult object.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

Gets the position of the last selected character in the specified field. Sends the result in ui.IntResult object.

102.3 Fields

Name	Type	Description
UiElement	optional AbstractUiElement	Source UI field.

103 GetTopRowNum Not-referenced

103.1 Diagram

103.2 Description

Name: GetTopRowNum

Gets the number of the top visible row in the specified table. Sends the result in the ui.IntResult object.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

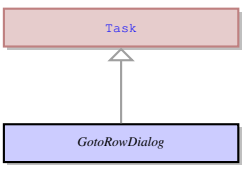
Gets the number of the top visible row in the specified table. Sends the result in the ui.IntResult object.

103.3 Fields

Name	Type	Description
TableElement	AbstractDataTable	Target Table/TreeTable element.

104 GotoRowDialog Not-referenced

104.1 Diagram



104.2 Description

Name: GotoRowDialog
No information
Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.
No information

104.3 Fields

Name	Type	Description
MaxValue	Int	The maximum value in the range of values accepted by a UI element.
FocusRow	optional Int	No information

105 GridColumnDefinition Not-referenced

105.1 Diagram

105.2 Description

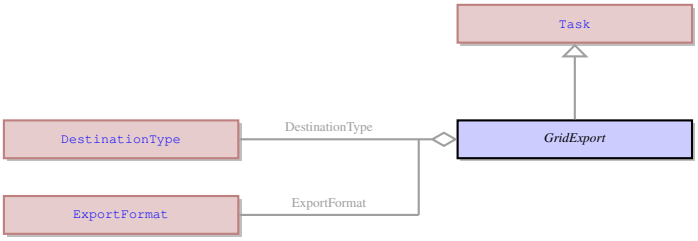
Name: GridColumnDefinition
This UI element defines the properties of a columns in a ui.GridPanel container and their properties.
No parents.
This UI element defines the properties of a columns in a ui.GridPanel container and their properties.

105.3 Fields

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

106 GridExport Not-referenced

106.1 Diagram



106.2 Description

Name: GridExport
Exports a grid contents to a clipboard or file, in either a text or html format. Doesn't send any result to the server.
Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.
Exports a grid contents to a clipboard or file, in either a text or html format. Doesn't send any result to the server.

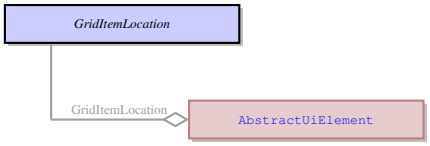
106.3 Fields

Name	Type	Description
DestinationType	DestinationType	Export type. Two export types are available: file and clipboard.

ExportFormat	ExportFormat	Indicates the fomat in which the data from a table should be saved to an external location. Two export formats are available: HTML and CSV.
FileName	String	Destination file name. Optional. If ui.DestinationType is 'file' and a ui.FileName is not specified, a Save File dialog box should be sisplayd allowing users to enter a filename and file location.

107 GridItemLocation

107.1 Diagram



107.2 Description

Name: GridItemLocation

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.

No parents.

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.

107.3 Fields

Name	Type	Description
GridX	optional 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	optional 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	optional Int	It specifies the number of horizontal cells that the element occupies. It cannot be less than 1.
GridHeight	optional Int	It specifies the number of vertical cells that the element occupies. It cannot be less than 1.

107.4 Referenced in

- GridItemLocation field in optional AbstractUiElement - 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.

108 GridLength Not-referenced

108.1 Diagram

108.2 Description

Name: GridLength

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.

No parents.

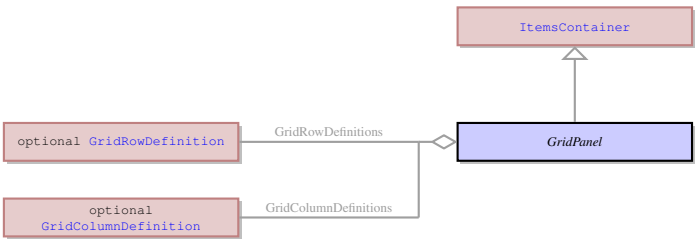
This UI element defines the length of the grid columns and width of the rows. Thus it can define the size of the ui.GridPanel cells. The size can be absolute or relative. It can also define the length of the table columns.

108.3 Fields

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

109 GridPanel Not-referenced

109.1 Diagram



109.2 Description

Name: GridPanel

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.

Parent: ItemsContainer - 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.

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.

109.3 Fields

Name	Type	Description
GridRowDefinitions	list of GridRowDefinition	This UI element defines the number of rows in a grid panel container and their properties.
GridColumnDefinitions	list of GridColumnDefinition	This UI element defines the number of rows in a grid panel container and their properties.

110 GridRowDefinition Not-referenced

110.1 Diagram

110.2 Description

Name: GridRowDefinition

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

No parents.

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

110.3 Fields

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

111 GridSetCurrentLine Not-referenced

111.1 Diagram

111.2 Description

Name: GridSetCurrentLine

Displays a specific line of the program array into the specified row of the screen array.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

Displays a specific line of the program array into the specified row of the screen array.

111.3 Fields

Name	Type	Description
TableElement	AbstractDataTable	Target Table/TreeTable element.
OffsetRow	optional Int	This is the number of rows by which the table or screen record must be scrolled. The value is absolute and defines the number of the program array row to be displayed at the top of the screen array (as the first row of the screen array).

112 GroupBox Not-referenced

112.1 Diagram

112.2 Description

Name: GroupBox

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.

Parent: ElementContainer - 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.

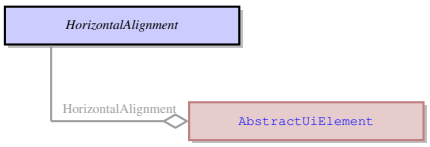
It is a container that groups the UI elements inside a visible border with an optional title at the top. It can contain only one other UI element. It can be another container or a form widget. Thus though it can encompass UI elements of the ui.AbstractField group, having only one element of this group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

112.3 Fields

Name	Type	Description
Title	optional String	This is the inscription attached to the UI element. Usually this is the text of all sorts of labels.
TitleJustification	TitleJustification	It specifies the horizontal alignment of the text of the title.

113 HorizontalAlignment

113.1 Diagram



113.2 Description

Name: HorizontalAlignment

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.

No parents.

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.

113.3 Options

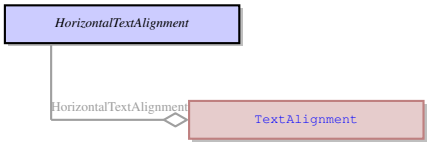
Name	Description
Default	The window size is the size with which it was opened or which was set after opening by 4GL or graphical theme means.
Stretch	The UI element will be stretched to fit the container (or container cell) without preserving the aspect ratio.
Left	The UI element will be aligned to the 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).

113.4 Referenced in

- HorizontalAlignment field in optional AbstractUiElement - 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.

114 HorizontalTextAlignment

114.1 Diagram



114.2 Description

Name: HorizontalTextAlignment

No parents.

114.3 Options

Name	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.

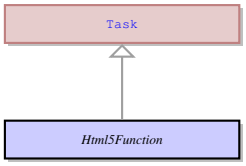
Left	The UI element will be aligned to the left side of the container (or container cell).
Center	The UI element will be equidistant from both sides.
Right	The UI element will be aligned to the right side of the container (or container cell).

114.4 Referenced in

- HorizontalTextAlignment field in optional TextAlignment -

115 Html5Function Not-referenced

115.1 Diagram



115.2 Description

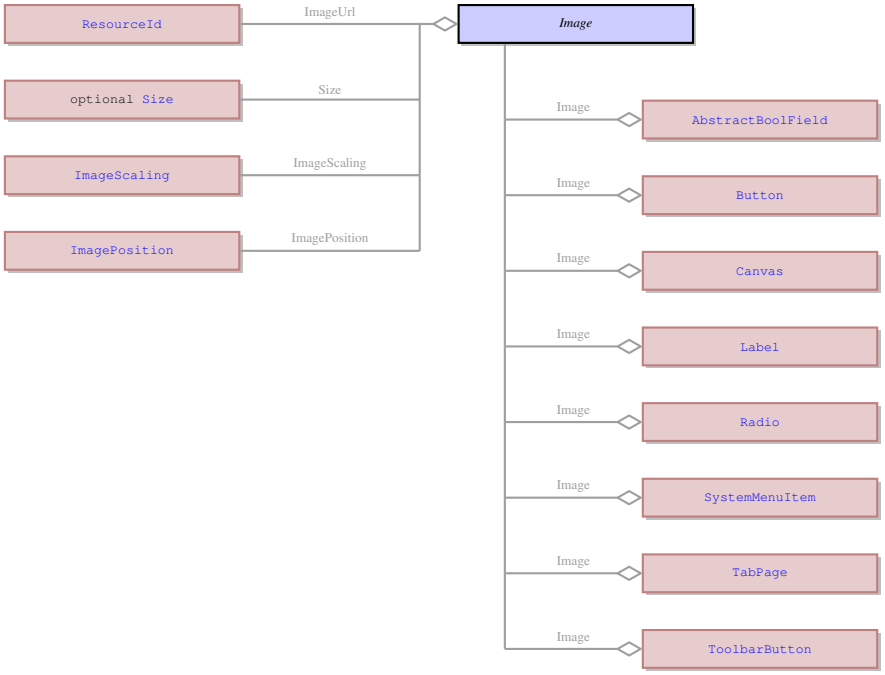
Name: Html5Function
No information
Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.
No information

115.3 Fields

Name	Type	Description
ModuleName	optional String	No information
FuncName	optional String	No information
FuncParam	optional String	No information

116 Image

116.1 Diagram



116.2 Description

Name: Image
It is an image that can be applied to other UI elements, e.g. to a button.
No parents.
It is an image that can be applied to other UI elements, e.g. to a button.

116.3 Fields

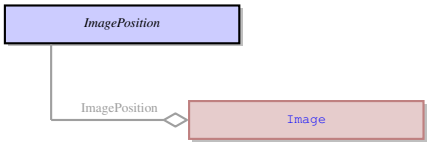
Name	Type	Description
ImageUrl	ResourceId	It specifies the URI of an image file. The image should be located on the application server and inside the folder into which the application is deployed. The URL should begin with: qx://application/... .
Size	optional Size	The size of the UI element in pixels that.
ImageScaling	ImageScaling	It specifies whether the image should be scaled to fit the UI element it is applied to.
ImagePosition	ImagePosition	No information

116.4 Referenced in

- Image field in optional AbstractBoolField - It is an image that can be applied to other UI elements, e.g. to a button.
- Image field in optional Button - It is an image that can be applied to other UI elements, e.g. to a button.
- Image field in optional Canvas - It is an image that can be applied to other UI elements, e.g. to a button.
- Image field in optional Label - It is an image that can be applied to other UI elements, e.g. to a button.
- Image field in optional Radio - It is an image that can be applied to other UI elements, e.g. to a button.
- Image field in optional SystemMenuItem - It is an image that can be applied to other UI elements, e.g. to a button.
- Image field in optional TabPage - It is an image that can be applied to other UI elements, e.g. to a button.
- Image field in optional ToolbarButton - It is an image that can be applied to other UI elements, e.g. to a button.

117 ImagePosition

117.1 Diagram



117.2 Description

Name: ImagePosition
No information
No parents.
No information

117.3 Options

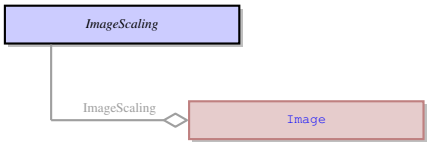
Name	Description
Left	The UI element will be aligned to the left side of the container (or container cell).
Right	The UI element will be aligned to the right side of the container (or container cell).
Top	The UI element will be aligned to the top of the container (or container cell).
Bottom	The UI element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

117.4 Referenced in

- ImagePosition field in optional Image - No information

118 ImageScaling

118.1 Diagram



118.2 Description

Name: ImageScaling

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.

No parents.

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.

118.3 Options

Name	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.

118.4 Referenced in

- ImageScaling field in optional Image - 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.

119 IntResult Not-referenced

119.1 Diagram

119.2 Description

Name: IntResult

The integer result of an operation.

No parents.

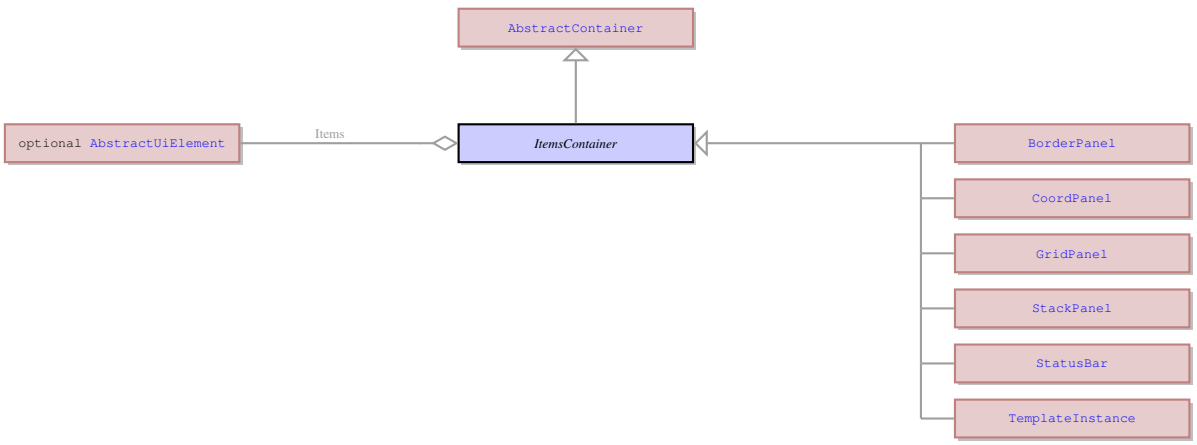
The integer result of an operation.

119.3 Fields

Name	Type	Description
IntValue	optional Int	An integer result of the operation.

120 ItemsContainer Not-referenced

120.1 Diagram



120.2 Description

Name: ItemsContainer

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.

Parent: AbstractContainer - 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.

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.

120.3 Children

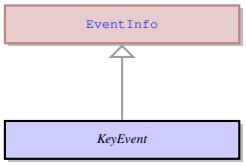
- BorderPanel - It is a concrete UI element - a container for arranging the layout of other UI elements. Other UI elements can be located either alongside the panel borders or in its center, thus this panel can incorporate up to 5 elements - 1 for each side and 1 in the center. The elements are stretched 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.
- CoordPanel - 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.
- GridPanel - 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.
- StackPanel - 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.
- StatusBar - 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.
- TemplateInstance - No information

120.4 Fields

Name	Type	Description
Items	list of AbstractUiElement	A set of UI elements that are placed inside the container.

121 KeyEvent Not-referenced

121.1 Diagram



121.2 Description

Name: KeyEvent

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.

Parent: EventInfo - 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.

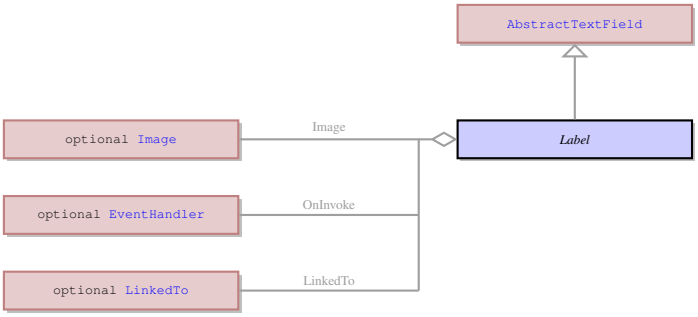
It is an event that is triggered when the specified key on the keyboard is pressed. This event is sent to the Application server on the keypress.

121.3 Fields

Name	Type	Description
KeyValue	optional String	The name of the key pressed. The key name is the name written on the key, e.g. F12 or A.
VirtualKeyValue	optional 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.
ControlModifier	Bool	It indicates whether the Ctrl key should be held down when the key is pressed.
AltModifier	Bool	It indicates whether the Alt key should be held down when the key is pressed.
ShiftModifier	Bool	It indicates whether the Shift key should be held down when the key is pressed.

122 Label Not-referenced

122.1 Diagram



122.2 Description

Name: Label

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.

Parent: AbstractTextField - 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.

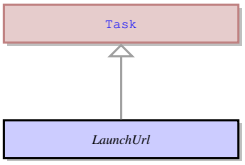
It is a concrete UI element that has the form of a label with some text, image or both. The label is not an interactive widget and cannot be used for input, but the information displayed by it can be changed dynamically.

122.3 Fields

Name	Type	Description
Image	optional Image	The image that is displayed to a label.
IsDynamic	Bool	It specifies whether the information displayed by the label can be changed dynamically by means of the DISPLAY TO statement.
OnInvoke	optional 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.
AllowNewlines	Bool	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.
LinkedTo	optional LinkedTo	No information

123 LaunchUrl Not-referenced

123.1 Diagram



123.2 Description

Name: LaunchUrl
No information
Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.
No information

123.3 Fields

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

124 LifeScope Not-referenced

124.1 Diagram

124.2 Description

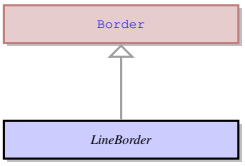
Name: LifeScope
EMPTY.
No parents.
EMPTY.

124.3 Options

Name	Description
Request	EMPTY.
Session	EMPTY.

125 LineBorder Not-referenced

125.1 Diagram

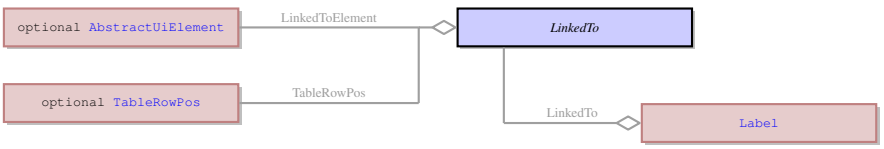


125.2 Description

Name: LineBorder
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.
Parent: Border - 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 .
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.

126 LinkedTo

126.1 Diagram



126.2 Description

Name: LinkedTo
No information
No parents.
No information

126.3 Fields

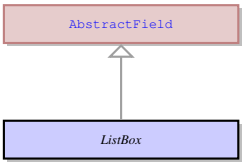
Name	Type	Description
LinkedToElement	optional AbstractUiElement	No information
TableRowPos	optional TableRowPos	No information

126.4 Referenced in

- LinkedTo field in optional Label - No information

127 ListBox Not-referenced

127.1 Diagram



127.2 Description

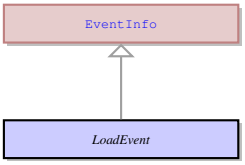
Name: ListBox
It is a concrete UI element that has the form of a form field with a list of values inside available for selection. It does not accept values entered from the keyboard, but can participate in the input and records into the underlying variable the value that was selected from the list.
Parent: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

127.3 Fields

Name	Type	Description
EnableMultiSelection	Bool	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.
ListBoxValues	list of String	No information
SelectedItems	String	No information
HelperText	optional String	No information

128 LoadEvent Not-referenced

128.1 Diagram



128.2 Description

Name: LoadEvent

This is th event info that is sent to server when a virtual table triggers OnFillBuffer event.

Parent: EventInfo - 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.

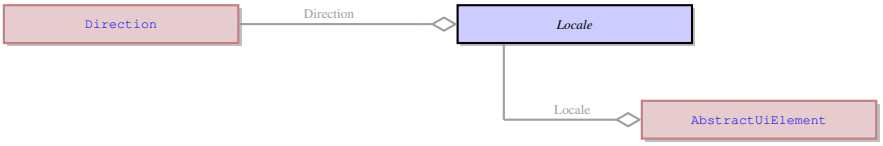
This is th event info that is sent to server when a virtual table triggers OnFillBuffer event.

128.3 Fields

Name	Type	Description
StartPosition	Int	It containes the row number - StartPosition - from which the rows should be losede to the buffer.

129 Locale

129.1 Diagram



129.2 Description

Name: Locale

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.

No parents.

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.

129.3 Fields

Name	Type	Description
Language	optional String	The language of the locale, e.g. FR for French.
Country	optional String	The territory where the specified locale language is used. E.g. CA - for French language in Canada.
Variant	optional String	The code set of the selected locale. E.g. ISO-8859-1 or UTF-8.
Direction	Direction	The direction of the text: from left to right or from right to left.

129.4 Referenced in

- Locale field in optional AbstractUiElement - 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.

130 Localization Not-referenced

130.1 Diagram

130.2 Description

Name: Localization

No information

No parents.

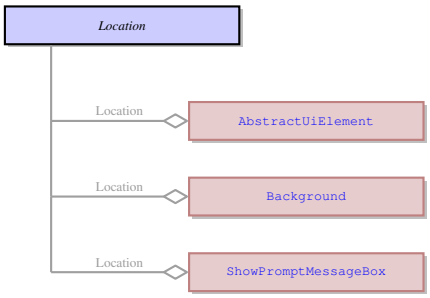
No information

130.3 Fields

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

131 Location

131.1 Diagram



131.2 Description

Name: Location

This is the coordinates of the position of a UI element inside a coordinate panel in pixels.
No parents.
This is the coordinates of the position of a UI element inside a coordinate panel in pixels.

131.3 Fields

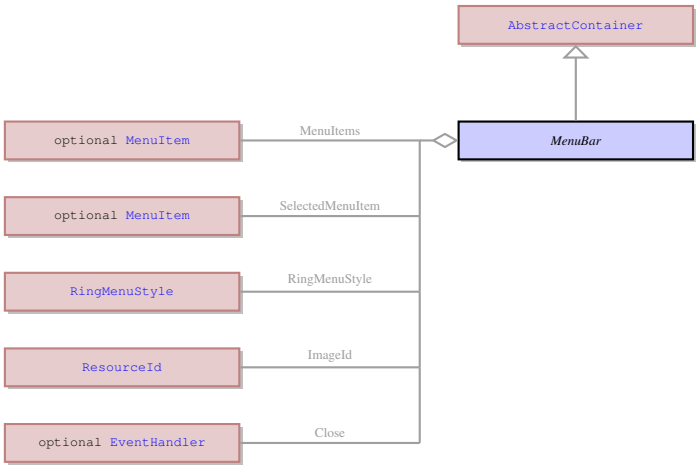
Name	Type	Description
XCoord	String	The coordinate of the top left corner of the element on X axis of the coord panel.
YCoord	String	The coordinate of the top left corner of the element on Y axis of the coord panel.

131.4 Referenced in

- Location field in optional AbstractUiElement - This is the coordinates of the position of a UI element inside a coordinate panel in pixels.
- Location field in optional Background - This is the coordinates of the position of a UI element inside a coordinate panel in pixels.
- Location field in optional ShowPromptMessageBox - This is the coordinates of the position of a UI element inside a coordinate panel in pixels.

132 MenuBar Not-referenced

132.1 Diagram



132.2 Description

Name: MenuBar

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

Parent: AbstractContainer - 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.

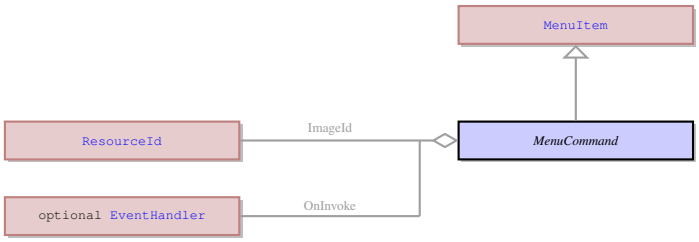
This is the area for the top menu (is not applied to ring menus). It includes menu options and menu option groups.

132.3 Fields

Name	Type	Description
MenuItems	list of MenuItem	A set of menu options belonging to the same menu.
SelectedMenuItem	optional MenuItem	It identifies one of the menu options that currently has the focus.
RingMenuStyle	RingMenuStyle	No information
ImageId	ResourceId	A reference to an image file.
Close	optional EventHandler	This event is triggered when the close button on the title bar of a window is pressed.

133 MenuCommand Not-referenced

133.1 Diagram



133.2 Description

Name: MenuCommand

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

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

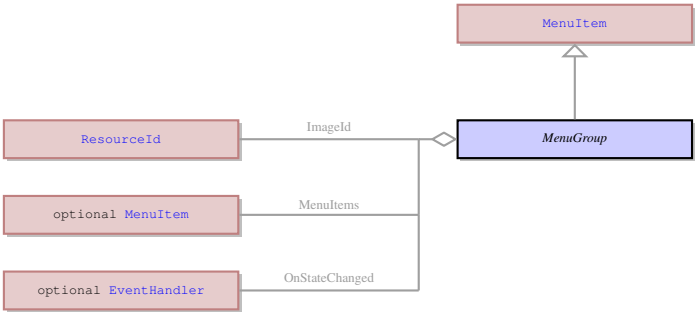
This is the menu option that can be invoked by the user. It has a label and/or icon and an even attached.

133.3 Fields

Name	Type	Description
Text	optional String	This is the label of the menu option.
ImageId	ResourceId	The image that will be used as the icon on the menu option button.
IsChecked	optional Bool	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.
ShortCut	optional 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.
OnInvoke	optional 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.

134 MenuGroup Not-referenced

134.1 Diagram



134.2 Description

Name: MenuGroup

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.

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

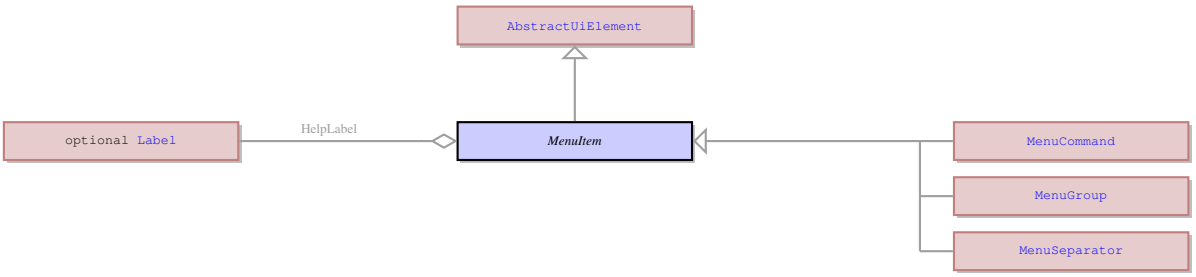
It is a group that unites several menu options and possibly menu separators. It offers a drop-down menu containing these options and separators, when the mouse cursor hovers over its label.

134.3 Fields

Name	Type	Description
Text	optional String	This is the of the menu group.
ImageId	ResourceId	A reference to an image file.
MenuItems	list of MenuItem	A set of menu options belonging to the same menu.
IsExpanded	Bool	No information
OnStateChanged	optional EventHandler	No information

135 MenuItem Not-referenced

135.1 Diagram



135.2 Description

Name: MenuItem

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

Parent: AbstractUiElement - 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.

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

135.3 Children

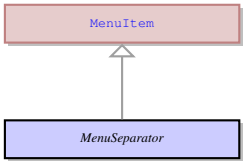
- MenuCommand - This is the menu option that can be invoked by the user. It has a label and/or icon and an even attached.
- MenuGroup - It is a group that unites several menu options and possibly menu separators. It offers a drop-down menu containing these options and separators, when the mouse cursor hovers over its label.
- MenuSeparator - It is a horizontal line that visually separates menu options in the drop-down list of the menu group.

135.4 Fields

Name	Type	Description
HelpLabel	optional Label	The description of the menu option that is shown on the second menu line for the ring menu.

136 MenuSeparator Not-referenced

136.1 Diagram



136.2 Description

Name: MenuSeparator

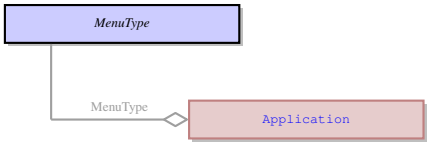
It is a horizontal line that visually separates menu options in the drop-down list of the menu group.

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

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

137 MenuType

137.1 Diagram



137.2 Description

Name: MenuType

No information

No parents.

No information

137.3 Options

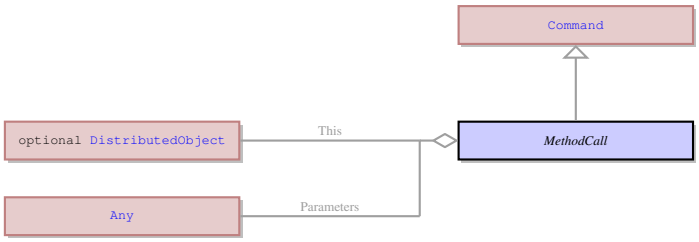
Name	Description
None	The property is not applied and the default behaviour is used.
Menu	Not described yet
Tree	Not described yet
PopTree	Not described yet

137.4 Referenced in

- MenuType field in optional Application - No information

138 MethodCall Not-referenced

138.1 Diagram



138.2 Description

Name: MethodCall
EMPTY.
Parent: Command - EMPTY.
EMPTY.

138.3 Fields

Name	Type	Description
Ident	Ident	No information
This	optional DistributedObject	No information
Parameters	list of Parameters	No information

139 ModelItem Not-referenced

139.1 Diagram

139.2 Description

Name: ModelItem
EMPTY.
No parents.
EMPTY.

140 MouseEvent Not-referenced

140.1 Diagram

140.2 Description

Name: MouseEvent
This is event information that describes an event triggered by mouse. It is sent to the server when events like OnMouseClick or other mouse events are invoked.
Parent: EventInfo - 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.
This is event information that describes an event triggered by mouse. It is sent to the server when events like OnMouseClick or other mouse events are invoked.

140.3 Fields

Name	Type	Description
LeftButton	Bool	This is the indicator that tracks the state of the left mouse button.
RightButton	Bool	This is the indicator that tracks the state of the right mouse button.
MiddleButton	Bool	This is the indicator that tracks the state of the left middle button and mouse scroll wheel.
XButton1	Bool	This is the indicator that tracks the state of the first additional mouse button.
XButton2	Bool	This is the indicator that tracks the state of the second additional mouse button.
X	Float	This is the X coordinate of the mouse cursor position at the time when the event was triggered.
Y	Float	This is the X coordinate of the mouse cursor position at the time when the event was triggere.

ControlModifier	Bool	It indicates whether the Ctrl key should be held down when the key is pressed.
AltModifier	Bool	It indicates whether the Alt key should be held down when the key is pressed.
ShiftModifier	Bool	It indicates whether the Shift key should be held down when the key is pressed.
TableRowPos	optional TableRowPos	No information

141 OnIdle Not-referenced

141.1 Diagram

141.2 Description

Name: OnIdle

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

No parents.

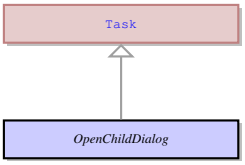
This event is triggered after the application has been idle for some time.

141.3 Fields

Name	Type	Description
IdleSeconds	optional 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.
Handler	optional EventHandler	It specifies the event handler that should be invoked on the keypress.

142 OpenChildDialog Not-referenced

142.1 Diagram



142.2 Description

Name: OpenChildDialog

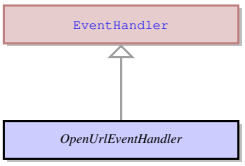
No information

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

No information

143 OpenUrlEventHandler Not-referenced

143.1 Diagram



143.2 Description

Name: OpenUrlEventHandler

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

Parent: EventHandler - This is common class for all the specific event handler types.

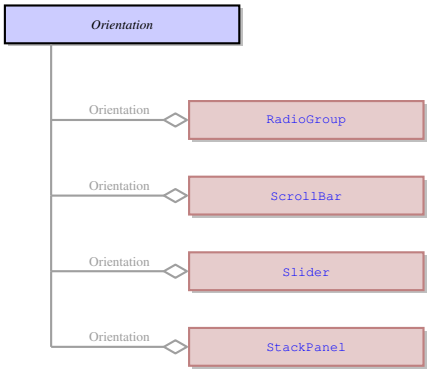
This is an event handler that can be assigned to any event. This handler opens the URL specified in the default system web browser.

143.3 Fields

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

144 Orientation

144.1 Diagram



144.2 Description

Name: Orientation

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.

No parents.

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.

144.3 Options

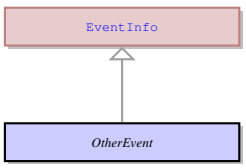
Name	Description
Horizontal	The UI element will be placed horizontally and directed from left to right.
Vertical	The UI element will be placed vertically and directed from top to bottom.

144.4 Referenced in

- Orientation field in optional RadioGroup - 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.
- Orientation field in optional ScrollBar - 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.
- Orientation field in optional Slider - 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.
- Orientation field in optional StackPanel - 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.

145 OtherEvent Not-referenced

145.1 Diagram



145.2 Description

Name: OtherEvent

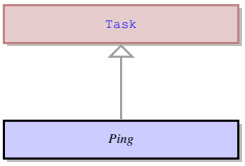
This event information described the source of the event - i.e. the widget which triggered the event (Radio - CheckedChanged, CheckBox - CheckedChanged, ComboBox - DropDown, etc.)

Parent: EventInfo - 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.

This event information described the source of the event - i.e. the widget which triggered the event (Radio - CheckedChanged, CheckBox - CheckedChanged, ComboBox - DropDown, etc.)

146 Ping Not-referenced

146.1 Diagram



146.2 Description

Name: Ping

this task is necessary solely to synchronize the server state with the client state, i.e, when all the client updates should be automatically passed to the server. Sends to server ui.PingResult object.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

this task is necessary solely to synchronize the server state with the client state, i.e, when all the client updates should be automatically passed to the server. Sends to server ui.PingResult object.

147 PingResult Not-referenced

147.1 Diagram

147.2 Description

Name: PingResult

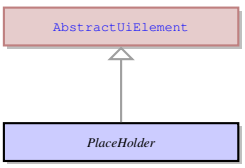
An empty object sent to server as result of the ui.Ping task.

No parents.

An empty object sent to server as result of the ui.Ping task.

148 PlaceHolder Not-referenced

148.1 Diagram



148.2 Description

Name: PlaceHolder

No information

Parent: AbstractUiElement - 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.

No information

149 PopupMenu Not-referenced

149.1 Diagram

149.2 Description

Name: PopupMenu

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.

Parent: AbstractContainer - 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.

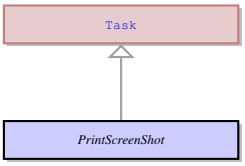
This is the context menu that is invoked by right-clicking the application area at runtime. Typically the menu items of the pop-up menu correspond to the toolbar buttons currently active/visible.

149.3 Fields

Name	Type	Description
MenuItems	list of MenuItem	A set of menu options belonging to the same menu.

150 PrintScreenShot Not-referenced

150.1 Diagram



150.2 Description

Name: PrintScreenShot

Prints a screenshot of the current window. Sends the execution result in the ui.PrintScreenShotResult object.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

Prints a screenshot of the current window. Sends the execution result in the ui.PrintScreenShotResult object.

150.3 Fields

Name	Type	Description
AdaptToPageSize	Bool	If true, the screenshot needs to be scaled to page.

151 PrintScreenShotResult Not-referenced

151.1 Diagram

151.2 Description

Name: PrintScreenShotResult

The result of the ui.PrintScreenShot task.

No parents.

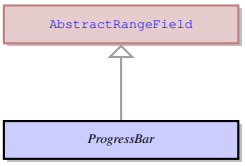
The result of the ui.PrintScreenShot task.

151.3 Fields

Name	Type	Description
ExecutionResult	Bool	Indicates whether the operation succeded or failed.

152 ProgressBar Not-referenced

152.1 Diagram



152.2 Description

Name: ProgressBar

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).

Parent: AbstractRangeField - 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 .

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).

152.3 Fields

Name	Type	Description
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.

153 PseudoClassName Not-referenced

153.1 Diagram

153.2 Description

Name: PseudoClassName

enum which lists all supported pseudo-classes

No parents.

enum which lists all supported pseudo-classes

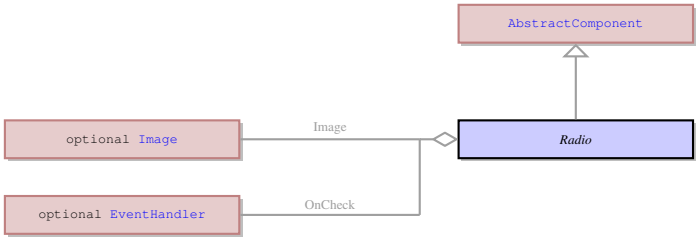
153.3 Options

Name	Description
Focus	Applies if the UI element is the current element in focus, e.g. when the cursor enters the field, the field becomes in focus.
NoFocus	Applies if the UI element is not currently in focus, e.g. when the cursor leaves the field, the field looses focus.
Inactive	Applies to elements that do not participate in any user interaction statement at the moment or were disabled explicitly.
Active	Applies to elements that participate in any user interaction statement at the moment.
Query	Applies to form widgets (fields) that take part in the execution of the CONSTRUCT statement at the moment.

Display	Applies to form widgets (fields) that were referenced by the DISPLAY or DISPLAY ARRAY statement.
Input	Applies to form widgets (fields) that take part in the execution of the INPUT or INPUT ARRAY statement at the moment.
Prompt	Applies to form widgets (fields) that take part in the execution of the PROMPT statement at the moment.
Message	Applies to form elements that were created using the MESSAGE statement.
Error	Applies to form elements that were created using the ERROR statement.
Comment	Applies to form elements that were created using the COMMENT property of a widget is displayed.
DynamicLabel	Applies to Label UI entity, if its isDynamic property is set to TRUE.
StaticLabel	Applies to any character string displayed to a form that does not belong to any form widget. E.g. a string displayed by means of DISPLAY AT, MESSAGE, ERROR statements, menu comments. It also applies to Label UI element, if its isDynamic property is set to FALSE.
IsProtected	Applied to all widgets which have isProtected property set to TRUE.
Border	Applied to bordered windows.
NoBorder	Applied to flat widows.
Form	Applied to the Form element (root element of the form xml file).
Selected	Applied to the currently active row of the table (i.e. current row during DISPLAY ARRAY statement) and to curently active TabPage in Tab container.
PrintableWidgets	Applies to all UI elements that can accept input from the keyboard.
GreyableWidgets	Applies to all UI elements that can be enabled or disabled (greyed).
DataWidgets	Applies to the following UI elements: TextField, TextAream, Spinner, Calendar, TimeEdit-Field, ComboBox, ListBox, Label, Radio, Button
SelectedMany	Not described yet
Lycia	Not described yet
Informix4GL	Not described yet
GBDS	Not described yet

154 Radio Not-referenced

154.1 Diagram



154.2 Description

Name: Radio

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.

Parent: AbstractComponent - This is the common parent of all UI elements.

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.

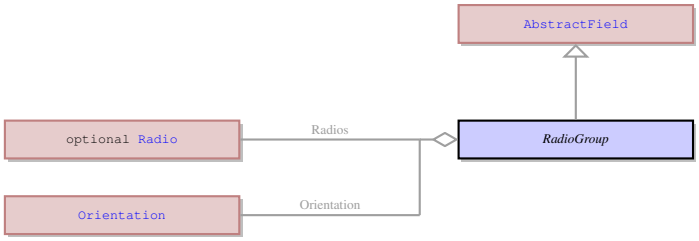
154.3 Fields

Name	Type	Description
Title	optional String	This is the inscription attached to the UI element. Usually this is the text of all sorts of labels.
Image	optional Image	It is an image that can be applied to other UI elements, e.g. to a button.
IsChecked	optional Bool	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.

OnCheck	optional EventHandler	The OnCheck field defines the event which will be triggered if the IsChecked field of the UI element is changed to TRUE.
AllowNewlines	Bool	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.

155 RadioGroup Not-referenced

155.1 Diagram



155.2 Description

Name: RadioGroup

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.

Parent: AbstractField - 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.

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.

155.3 Fields

Name	Type	Description
Radios	list of Radio	This is the list of Radios that belong to the specified RadioGroup element.
Orientation	Orientation	This enum specifies whether the UI element should have vertical or horizontal layout.
Required	Bool	No information

156 ReportViewerConfig Not-referenced

156.1 Diagram

156.2 Description

Name: ReportViewerConfig

No information

Parent: AbstractComponent - This is the common parent of all UI elements.

No information

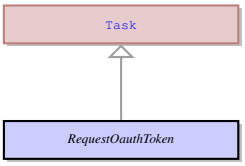
156.3 Fields

Name	Type	Description

ViewerType	ViewerType	No information
Parameter	optional String	This is the type of the wrapper to be applied to the table.

157 RequestOauthToken Not-referenced

157.1 Diagram



157.2 Description

Name: RequestOauthToken
No information
Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.
No information

157.3 Fields

Name	Type	Description
ProviderName	optional String	No information

158 ResourceId Not-referenced

158.1 Diagram

158.2 Description

Name: ResourceId
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.
No parents.
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.

158.3 Fields

Name	Type	Description
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/... .

159 ResponseOauthToken Not-referenced

159.1 Diagram

159.2 Description

Name: ResponseOauthToken
No information
No parents.
No information

159.3 Fields

Name	Type	Description
ErrorMessage	optional String	No information

160 Result Not-referenced

160.1 Diagram

160.2 Description

Name: Result
No information
No parents.
No information

160.3 Fields

Name	Type	Description
ButtonName	String	The name of the button that was pressed.
MessageBoxInput	optional String	The value that was entered to the prompt message box.

161 ResultValue Not-referenced

161.1 Diagram

161.2 Description

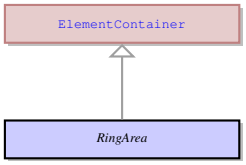
Name: ResultValue
The value retrieved from a target property.
No parents.
The value retrieved from a target property.

161.3 Fields

Name	Type	Description
ClientPropertyValue	String	The value of the requested property.

162 RingArea Not-referenced

162.1 Diagram

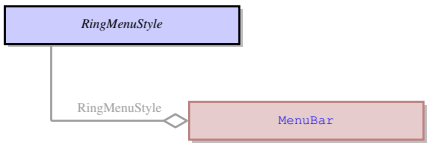


162.2 Description

Name: RingArea
This is the area that incorporates ring menu and its options. It must not be mistaken with the MenuBar used for top menu.
Parent: ElementContainer - 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.
This is the area that incorporates ring menu and its options. It must not be mistaken with the MenuBar used for top menu.

163 RingMenuStyle

163.1 Diagram



163.2 Description

Name: RingMenuStyle
No information
No parents.
No information

163.3 Options

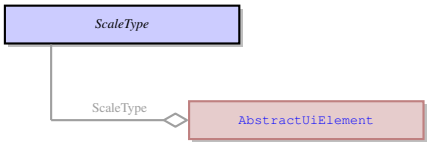
Name	Description
None	The property is not applied and the default behaviour is used.
Dialog	Not described yet
Popup	Not described yet

163.4 Referenced in

- RingMenuStyle field in optional MenuBar - No information

164 ScaleType

164.1 Diagram



164.2 Description

Name: ScaleType
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.
No parents.
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.

164.3 Options

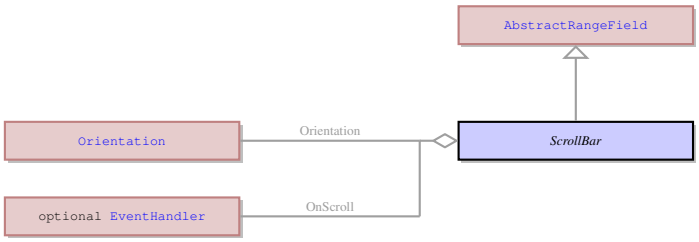
Name	Description
NoScale	The scaling is not applied when the element is resized. It will be resized only according to its layout position; e.g. the button will be enlarged, but the text on it will remain unchanged.
Both	When an element is resized, its contents is also resized: if a button gets bigger,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

164.4 Referenced in

- ScaleType field in optional AbstractUiElement - 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.

165 ScrollBar Not-referenced

165.1 Diagram



165.2 Description

Name: ScrollBar

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.

Parent: AbstractRangeField - 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 .

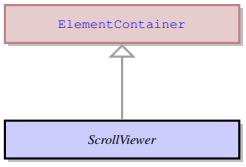
It is a concrete UI element that is represented by a scrollbar. It as the maximum and minimum values and the slider can be moved by the user at runtime or by displaying values to the element.

165.3 Fields

Name	Type	Description
Orientation	Orientation	This enum specifies whether the UI element should have vertical or horizontal layout.
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.
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.
OnScroll	optional EventHandler	This is the event invoked when the slider of the UI element moves.
ViewportSize	optional Int	No information

166 ScrollViewer Not-referenced

166.1 Diagram



166.2 Description

Name: ScrollViewer

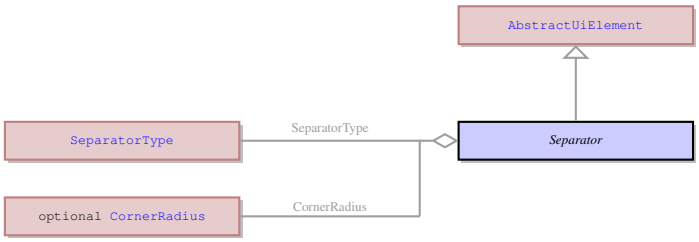
It is a container the content of which can be bigger than the container. The scrollbars are used to view the content that does not fit. It can contain exactly one element. E.g. it can contain a stack panel container, the number of elements inside which can be bigger than fit the size of the Scroll Viewer.

Parent: ElementContainer - 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.

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.

167 Separator Not-referenced

167.1 Diagram



167.2 Description

Name: Separator

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

Parent: AbstractUiElement - 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.

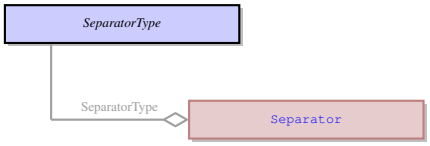
Any kind of separator, e.g. the status bar separator.

167.3 Fields

Name	Type	Description
SeparatorType	SeparatorType	This is the type of the separator to be displayed
CornerRadius	optional CornerRadius	The radius of a corner of a custom border around the UI element. It is used to make the border corners rounded.

168 SeparatorType

168.1 Diagram



168.2 Description

Name: SeparatorType

This is the type of the separator to be displayed

No parents.

This is the type of the separator to be displayed

168.3 Options

Name	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.

168.4 Referenced in

- SeparatorType field in optional Separator - This is the type of the separator to be displayed

169 ServerEventHandler Not-referenced

169.1 Diagram

169.2 Description

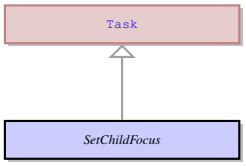
Name: ServerEventHandler
EMPTY.
Parent: EventHandler - This is common class for all the specific event handler types.
EMPTY.

169.3 Children

- BackgroundServerEventHandler - EMPTY.
- BlockingServerEventHandler - EMPTY.

170 SetChildFocus Not-referenced

170.1 Diagram



170.2 Description

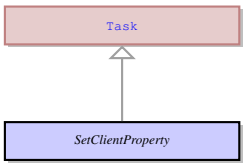
Name: SetChildFocus
No information
Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.
No information

170.3 Fields

Name	Type	Description
Identifier	String	It is a unique name of a UI element by which it can be referenced.

171 SetClientProperty Not-referenced

171.1 Diagram



171.2 Description

Name: SetClientProperty

Sets the properties on the client side. Doesn't send any result to server.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

Sets the properties on the client side. Doesn't send any result to server.

171.3 Fields

Name	Type	Description
ClientPropertyType	String	The type of the property.
ClientPropertyName	String	The actual name of the property.
ClientPropertyValue	String	The value, the new desired setting.

172 SetCursor Not-referenced

172.1 Diagram

172.2 Description

Name: SetCursor

Moves the cursor in the specified field to a specific position. Doesn't send any result to server.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

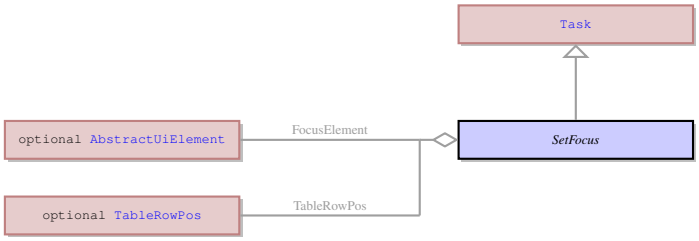
Moves the cursor in the specified field to a specific position. Doesn't send any result to server.

172.3 Fields

Name	Type	Description
Source	optional AbstractUiElement	Source UI field.
Position	optional Int	Target position of the cursor.

173 SetFocus Not-referenced

173.1 Diagram



173.2 Description

Name: SetFocus

Moves the focus to the specified field. Doesn't send any result to the server.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

Moves the focus to the specified field. Doesn't send any result to the server.

173.3 Fields

Name	Type	Description
FocusElement	optional AbstractUiElement	The target focused field.
TableRowPos	optional TableRowPos	No information

174 SetFocusToRow Not-referenced

174.1 Diagram

174.2 Description

Name: SetFocusToRow

No information

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

No information

174.3 Fields

Name	Type	Description
TableElement	AbstractDataTable	Target Table/TreeTable element.
FocusRow	optional Int	No information

175 SetLabelText Not-referenced

175.1 Diagram

175.2 Description

Name: SetLabelText

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.

Parent: EventHandler - This is common class for all the specific event handler types.

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.

175.3 Fields

Name	Type	Description
DstLabel	optional Label	This is the label which text should be changed by this event.
HelpString	optional String	This is the text that will be displayed to the corresponding predefined line.

176 SetSelection Not-referenced

176.1 Diagram

176.2 Description

Name: SetSelection

Selects the text in the specified field. Doesn't send any result to server.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

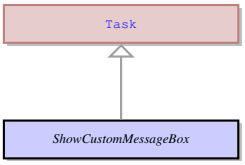
Selects the text in the specified field. Doesn't send any result to server.

176.3 Fields

Name	Type	Description
UiElement	optional AbstractUiElement	Source UI field.
SelectionBegin	optional Int	The start position of the selection.
SelectionEnd	optional Int	The end position of the selection.

177 ShowCustomMessageBox Not-referenced

177.1 Diagram



177.2 Description

Name: ShowCustomMessageBox

Displays a custom simple message box, with a specifiable range of button options. Sends the result to the server in the ui.MessageBoxResult object.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

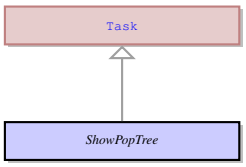
Displays a custom simple message box, with a specifiable range of button options. Sends the result to the server in the ui.MessageBoxResult object.

177.3 Fields

Name	Type	Description
MessageBoxIcon	optional String	The icon to be shown in the message box.
MessageBoxButtons	optional String	The list of the buttons to be shown in the message box. They are given in one string and separated by the '—' symbol.
Title	optional String	The message box title.
Text	optional String	The message.
DefaultMessageBoxButton	optional String	The default result button of the message box.

178 ShowPopTree Not-referenced

178.1 Diagram



178.2 Description

Name: ShowPopTree
No information
Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.
No information

179 ShowPromptMessageBox Not-referenced

179.1 Diagram

179.2 Description

Name: ShowPromptMessageBox
Displays a dialog box with a field that accepts a value. Sends the result to server in the ui.MessageBoxResult object.
Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.
Displays a dialog box with a field that accepts a value. Sends the result to server in the ui.MessageBoxResult object.

179.3 Fields

Name	Type	Description
Title	optional String	The message box title.
Text	optional String	The message.
DefaultString	optional String	The default value of a prompt message box.
Location	optional Location	The prompt message box disposition coordinates.
MaxLength	optional Int	The maximum number of characters that can be inputted to the prompt text field.

180 ShowSvgImage Not-referenced

180.1 Diagram

180.2 Description

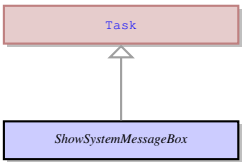
Name: ShowSvgImage
Shows the svg image to the specified canvas widget. Doesn't send any result to server.
Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.
Shows the svg image to the specified canvas widget. Doesn't send any result to server.

180.3 Fields

Name	Type	Description
ImageContainer	optional Canvas	The field which displays the svg image.
SvgValue	optional String	The svg image.

181 ShowSystemMessageBox Not-referenced

181.1 Diagram



181.2 Description

Name: ShowSystemMessageBox

Displays a simple system message box with a specifiable range of button options. Sends the result to the server in ui.MessageBoxResult object.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

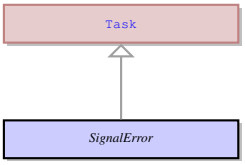
Displays a simple system message box with a specifiable range of button options. Sends the result to the server in ui.MessageBoxResult object.

181.3 Fields

Name	Type	Description
MessageBoxIcon	optional String	The icon to be shown in the message box.
MessageBoxButtons	optional String	The list of the buttons to be shown in the message box. They are given in one string and separated by the '—' symbol.
Title	optional String	The message box title.
Text	optional String	The message.
DefaultMessageBoxButton	optional String	The default result button of the message box.

182 SignalError Not-referenced

182.1 Diagram



182.2 Description

Name: SignalError

This task informs the client that there was an error on the server. It contains the error message. It is sent to the client if there was an error during loading of the recourses on the server.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

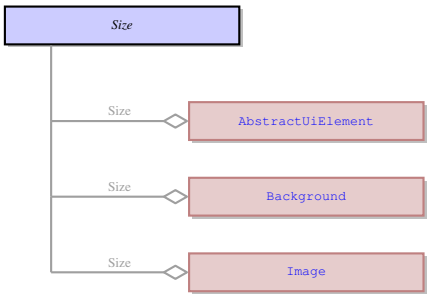
This task informs the client that there was an error on the server. It contains the error message. It is sent to the client if there was an error during loading of the recourses on the server.

182.3 Fields

Name	Type	Description
Message	String	The output message.

183 Size

183.1 Diagram



183.2 Description

Name: Size

The size of a UI element in pixels.

No parents.

The size of a UI element in pixels.

183.3 Fields

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

183.4 Referenced in

- Size field in optional AbstractUiElement - The size of a UI element in pixels.
- Size field in optional Background - The size of a UI element in pixels.
- Size field in optional Image - The size of a UI element in pixels.

184 Slider Not-referenced

184.1 Diagram

184.2 Description

Name: Slider

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.

Parent: AbstractRangeField - 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 .

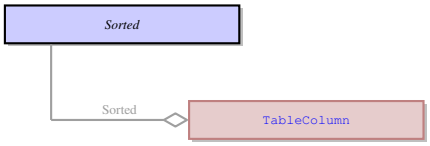
This is a concrete UI element that consists of a scale and a slider that can move across this scale. The slider widget has the minimum and maximum value which present the start and the end of the scale. It can be moved directly by the user during the input, or it can be moved if a value within its values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

184.3 Fields

Name	Type	Description
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.
Orientation	Orientation	This enum specifies whether the UI element should have vertical or horizontal layout.

185 Sorted

185.1 Diagram



185.2 Description

Name: Sorted

No information

No parents.

No information

185.3 Options

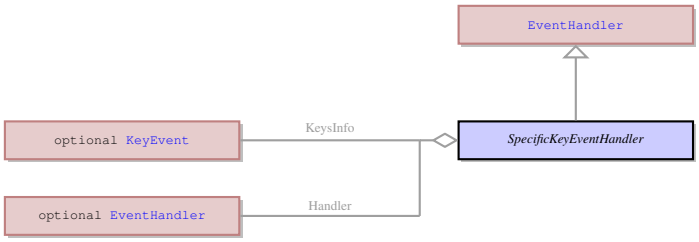
Name	Description
None	The property is not applied and the default behaviour is used.
Asc	Not described yet
Desc	Not described yet

185.4 Referenced in

- Sorted field in optional TableColumn - No information

186 SpecificKeyEventHandler Not-referenced

186.1 Diagram



186.2 Description

Name: SpecificKeyEventHandler

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

Parent: EventHandler - This is common class for all the specific event handler types.

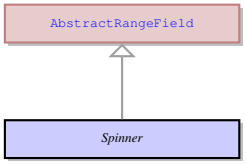
This event handler specifies what event handler should be triggered when a specific key is pressed. It links the keypress with a 4GL event.

186.3 Fields

Name	Type	Description
KeysInfo	list of KeyEvent	It specifies the concrete keys that must be pressed to trigger the event.
Handler	optional EventHandler	It specifies the event handler that should be invoked on the keypress.

187 Spinner Not-referenced

187.1 Diagram



187.2 Description

Name: Spinner

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.

Parent: AbstractRangeField - 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 .

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.

187.3 Fields

Name	Type	Description
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.

188 StackPanel Not-referenced

188.1 Diagram

188.2 Description

Name: StackPanel

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.

Parent: ItemsContainer - 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.

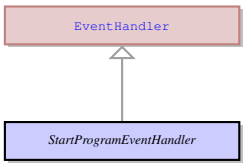
This is a container which arranges the elements in horizontal or vertical stacks. Any number of elements can be placed inside this container one next to the other. At runtime the contents of the stack panel can be resized only in the direction opposite to the orientation of the container.

188.3 Fields

Name	Type	Description
Orientation	Orientation	This enum specifies whether the UI element should have vertical or horizontal layout.
Reverse	Bool	No information

189 StartProgramEventHandler Not-referenced

189.1 Diagram



189.2 Description

Name: StartProgramEventHandler

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.

Parent: EventHandler - This is common class for all the specific event handler types.

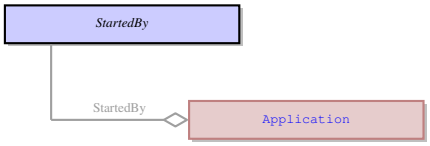
This event handler specifies the child 4GL program that should be launched and the parameters of this program. It is normally used for the MDI mode, but can be used in other cases.

189.3 Fields

Name	Type	Description
ProgramName	optional String	The name of the child program.
ProgramParameters	optional String	The parameters of the child program.
ProgramServer	optional String	The name of the host - the application server on which the program is deployed and should run.
ProgramPort	optional String	The port on the application server.
UserId	optional String	The name of the user who runs the application.
Waiting	Bool	It indicates whether the parent program should be suspended until the child program is closed.

190 StartedBy

190.1 Diagram



190.2 Description

Name: StartedBy

No information

No parents.

No information

190.3 Fields

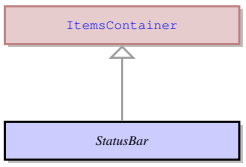
Name	Type	Description
EventId	optional Int	No information
ClientPID	optional Int	No information
ParentWait	Bool	No information

190.4 Referenced in

- StartedBy field in optional Application - No information

191 StatusBar Not-referenced

191.1 Diagram



191.2 Description

Name: StatusBar

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.

Parent: ItemsContainer - 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.

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.

192 StringResult Not-referenced

192.1 Diagram

192.2 Description

Name: StringResult

No information

No parents.

No information

192.3 Fields

Name	Type	Description
StringValue	optional String	This is one or more printable characters or white space characters enclosed in quotation marks.

193 SyncTableClassTask Not-referenced

193.1 Diagram

193.2 Description

Name: SyncTableClassTask

No information

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

No information

193.3 Fields

Name	Type	Description
TableElement	AbstractDataTable	Target Table/TreeTable element.
Position	optional Int	The start position of the range.
SyncData	optional String	No information

194 SyncTableInputTask Not-referenced

194.1 Diagram

194.2 Description

Name: SyncTableInputTask

No information

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

No information

194.3 Fields

Name	Type	Description
TableElement	AbstractDataTable	Target Table/TreeTable element.
SyncData	optional String	No information

195 SyncTask Not-referenced

195.1 Diagram

195.2 Description

Name: SyncTask

Synchronizes the the states (the number and the stacks of rows) of the virtual table on the client and server sides.

ui.DeleteRange describes which rows should be deleted from the table child widget and moved to the end of the free rows stack.

ui.InsertRange describes which rows should be deleted from the beginning of the free rows stack and moved to the table children widget. Deliting should be done before inserting. It doesn't do any changes in Table datamodel structure. Doesn't send any result to server.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

Synchronizes the the states (the number and the stacks of rows) of the virtual table on the client and server sides.

ui.DeleteRange describes which rows should be deleted from the table child widget and moved to the end of the free rows stack.

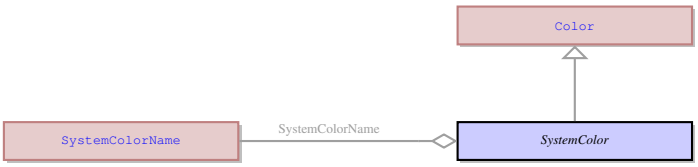
ui.InsertRange describes which rows should be deleted from the beginning of the free rows stack and moved to the table children widget. Deliting should be done before inserting. It doesn't do any changes in Table datamodel structure. Doesn't send any result to server.

195.3 Fields

Name	Type	Description
TableElement	AbstractDataTable	Target Table/TreeTable element.
Position	optional Int	The start position of the range.
SyncData	optional String	No information

196 SystemColor Not-referenced

196.1 Diagram



196.2 Description

Name: SystemColor

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.

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

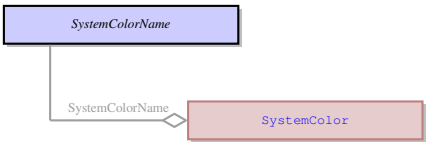
The system color defines a list of preset colours that can be applied to widgets, as opposed to the custom colour where the user needs to specify RGB of the color.

196.3 Fields

Name	Type	Description
SystemColorName	SystemColorName	It is the name of one of the predefined system colors.

197 SystemColorName

197.1 Diagram



197.2 Description

Name: SystemColorName

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

No parents.

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

197.3 Options

Name	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.

197.4 Referenced in

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

198 SystemContextMenu Not-referenced

198.1 Diagram

198.2 Description

Name: SystemContextMenu

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

Parent: AbstractContainer - 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.

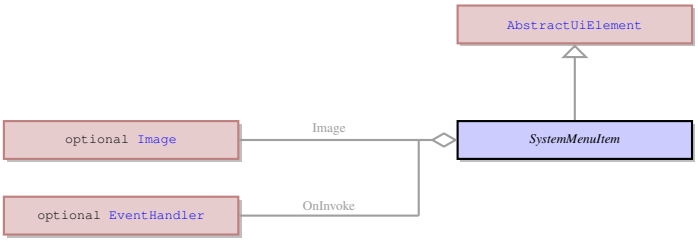
This is the context menu which is invoked by right-clicking the title bar of the 4GL window.

198.3 Fields

Name	Type	Description
SystemMenuItems	list of SystemMenuItem	It is the list of items belonging to the system context menu.

199 SystemMenuItem Not-referenced

199.1 Diagram



199.2 Description

Name: SystemMenuItem

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

Parent: AbstractUiElement - 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.

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

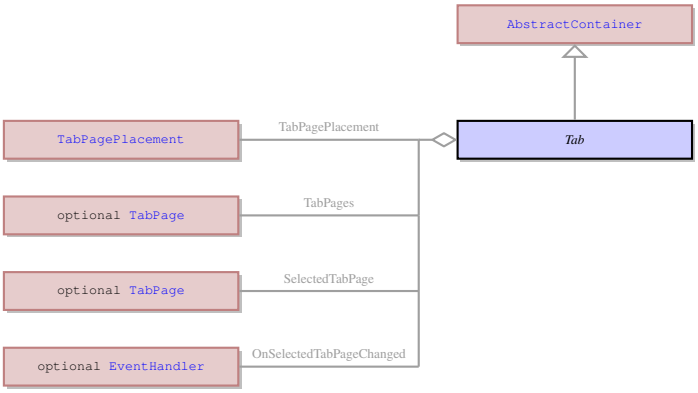
199.3 Fields

Name	Type	Description
Text	optional String	A character string.
Image	optional Image	It specifies the icon next to the system menu option. The icon must be 12x12 pixels, monochrome.

OnInvoke	optional 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.

200 Tab Not-referenced

200.1 Diagram



200.2 Description

Name: Tab

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.

Parent: AbstractContainer - 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.

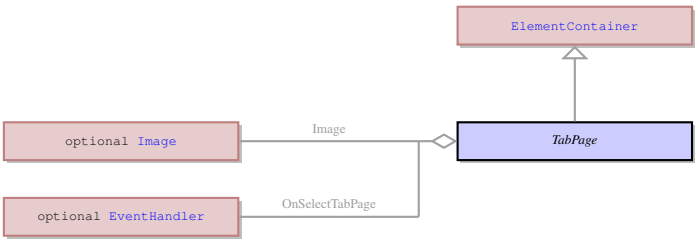
This is a special type of container which can contain any number of elements, but these elements can only be of ui.TabPage . The Tab serves as the container for a stack of tab pages with only one page visible at a time. Other pages can be brought forward by clicking on their tabs.

200.3 Fields

Name	Type	Description
TabPagePlacement	TabPagePlacement	It defines where the tabs should be located - to which side of the tab panel should they adjoin.
TabPages	list of TabPage	This is the set of tab pages that belong to the same tab container.
SelectedTabPage	optional TabPage	It defines which tab page is the current one - the contents of which tab page is now visible.
OnSelectedTabPageChanged	optional EventHandler	This is an event that is triggered every time the current tab page is changed.

201 TabPage Not-referenced

201.1 Diagram



201.2 Description

Name: TabPage

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.

Parent: ElementContainer - 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.

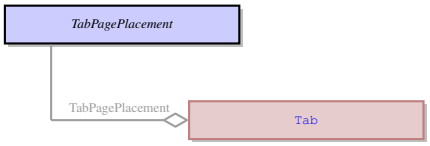
This is a container that can only be placed inside the ui.Tab container. A tab page can contain a single element of any type. Each tab page has a tab with the page title which is used to bring the page forward from the stack of other tab pages at runtime or during form modification.

201.3 Fields

Name	Type	Description
Title	optional String	This is the inscription attached to the UI element. Usually this is the text of all sorts of labels.
Image	optional Image	This is an icon that can be displayed to the tab of the page with or instead of the page title.
OnSelectTabPage	optional 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.

202 TabPagePlacement

202.1 Diagram



202.2 Description

Name: TabPagePlacement

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.

No parents.

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.

202.3 Options

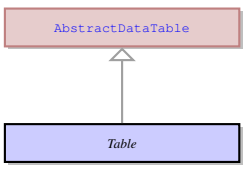
Name	Description
Top	The UI element will be aligned to the top of the container (or container cell).
Left	The UI element will be aligned to the left side of the container (or container cell).
Right	The UI element will be aligned to the right side of the container (or container cell).
Bottom	The UI element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

202.4 Referenced in

- TabPagePlacement field in optional Tab - 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.

203 Table Not-referenced

203.1 Diagram



203.2 Description

Name: Table

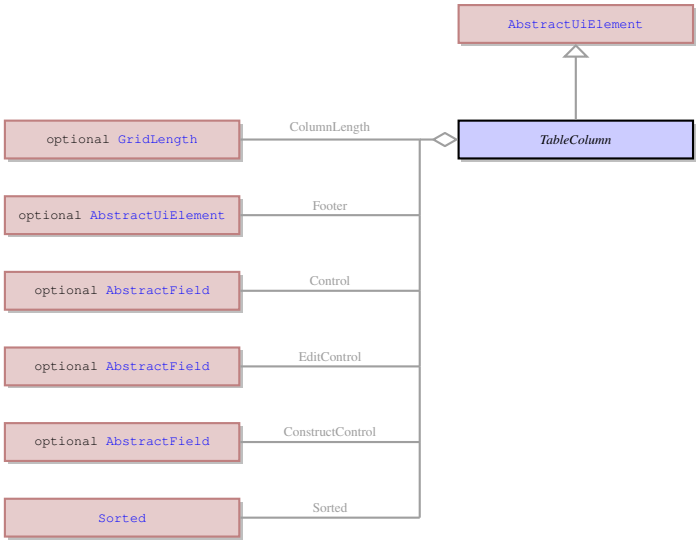
This is a container that can only contain a specific type of element - ui.TableColumn . It serves as the root container of a table with rows and columns of widgets used to display and input data.

Parent: AbstractDataTable - 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.

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.

204 TableColumn Not-referenced

204.1 Diagram



204.2 Description

Name: TableColumn

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.

Parent: AbstractUiElement - 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.

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.

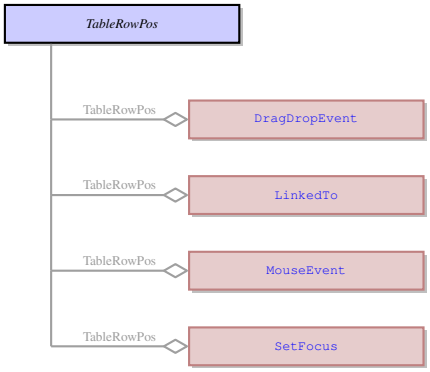
204.3 Fields

Name	Type	Description
Text	optional String	This is the text used as the header of the column.
ColumnLength	optional GridLength	It specifies the length of a column. The column length determines how many rows of widgets the table will have.
Resizable	Bool	It indicates whether the user is allowed to resize the column at run-time using the mouse cursor.
ReadOnly	Bool	If enabled, it prevents the user from entering values into the field at runtime even if the field is included into the input routine.
Footer	optional AbstractUiElement	No information

AllowNewlines	Bool	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.
ColumnNum	optional Int	No information
Control	optional AbstractField	No information
EditControl	optional AbstractField	No information
ConstructControl	optional AbstractField	No information
Unsortable	Bool	No information
Sorted	Sorted	No information

205 TableRowPos

205.1 Diagram



205.2 Description

Name: TableRowPos
No information
No parents.
No information

205.3 Fields

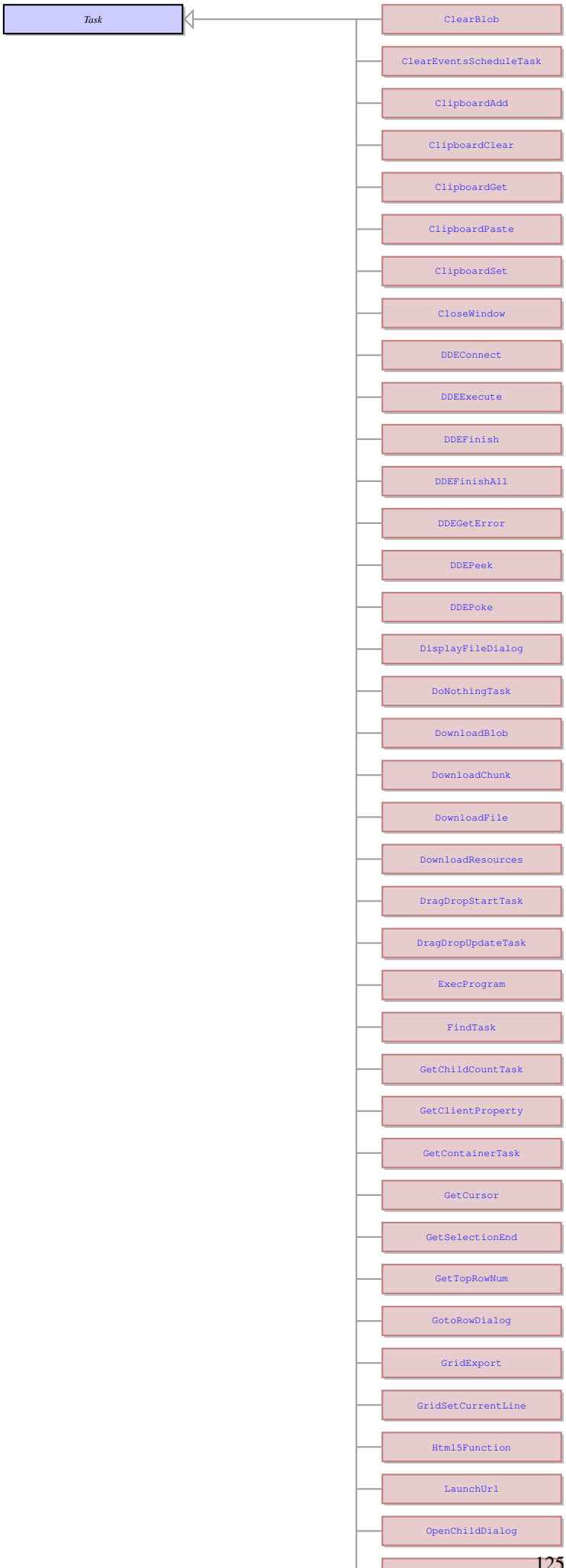
Name	Type	Description
RowIndex	Int	No information
ColumnIndex	Int	No information

205.4 Referenced in

- TableRowPos field in optional DragDropEvent - No information
- TableRowPos field in optional LinkedTo - No information
- TableRowPos field in optional MouseEvent - No information
- TableRowPos field in optional SetFocus - No information

206 Task Not-referenced

206.1 Diagram



206.2 Description

Name: Task

This an abstract entity that serves as a parent for the most of the tasks performed by the client.
No parents.
This an abstract entity that serves as a parent for the most of the tasks performed by the client.

206.3 Children

- ClearBlob - Clears the content of the BlobViewer element specified in the ui.Viewer property.
- ClearEventsScheduleTask - Clears the schedule of events to be handled. Doesn't send any result to server.
- ClipboardAdd - Adds to the content of the clipboard. Sends the execution result in the ui.ClipboardResult object.
- ClipboardClear - Clears the content of the clipboard. Sends the execution result in the ui.ClipboardResult object.
- ClipboardGet - Gets the content of the clipboard. Sends the text in the clipboard in the ui.ClipboardResult object.
- ClipboardPaste - Pastes the content of the clipboard to the current field. Sends the execution result in the ui.ClipboardResult object.
- ClipboardSet - Sets the content of the clipboard. Sends the execution result in the ui.ClipboardResult object.
- CloseWindow - Closes specified window. Doesn't send any result to the server.
- DDEConnect - Opens a connection to an application which supports DDE. Passes the result of the operation to ui.IsDDEError of the ui.DDEResult object and sends it to the server.
- DDEExecute - Executes a command in the specified document, using the program opened by ui.DDEConnect . Passes the result of the operation to ui.IsDDEError of the ui.DDEResult object and sends it to the server.
- DDEFinish - Closes the connection channel to the program and document. Passes the result of the operation to ui.IsDDEError of the ui.DDEResult object and sends it to the server.
- DDEFinishAll - Is used to close all DDE connections, and the program that is being communicated with via DDE. Passes the result of the operation to ui.IsDDEError of the ui.DDEResult object and sends it to the server.
- DDEGetError - Retrieves the last error recorded for the DDE channel. Sends the result to the ui.DDEError object.
- DDEPeek - Gets values from a specified place within a specific file. Passes the result of the operation to ui.DDEMessage of the ui.DDEResult object and sends it to the server.
- DDEPoke - Sends data to the open document, and places it in the specified part of the document. Passes the result of the operation to ui.IsDDEError of ui.DDEResult object and sends it to the server.
- DisplayFileDialog - Calls a message box dialog allowing a user to save or open a particular file. Sends the result to the server in the ui.MessageBoxResult object.
- DoNothingTask - This task is necessary solely to synchronize the client state with the server state, i.e., in case when all the server updates should be automatically passed to the client. Doesn't send any result to server.
- DownloadBlob - Downloads a file in the binary format from the server and displays it to the Blob Viewer element.
- DownloadChunk - Downloads chunk of file with specified size. Doesn't send any result to server.
- DownloadFile - Downloads a file in the binary format from the server and saves it to the path specified in the ui.ClientPath property .
- DownloadResources - Downloads resources from the specified list. Doesn't send any result to server.
- DragDropStartTask - This task comes to client as answer for event ui.OnDragStart and informs the client that Drag and Drop operation is allowed and can be performed.
- DragDropUpdateTask - This task comes to client as answer for events ui.OnDragEnter and ui.OnDragOver if it needs to update Drag And Drop action's preview (feetback).
- ExecProgram - Launches a specified 4gl program using the same client. Doesn't send any result to server.
- FindTask - No information
- GetChildCountTask - Calculates the number of children in specefied parent container. Sends result to server in the ui.GetChildCountResul object.
- GetClientProperty - Returns the value of the required property an sends the result in the ui.ResultValue object.

- GetContainerTask - Gets the parent container of the application. Sends the result to server in the ui.GetContainerResult object.
- GetCursor - Gets the position of the cursor in the specified field. Sends the result to server in ui.CursorPosition object.
- GetSelectionEnd - Gets the position of the last selected character in the specified field. Sends the result in ui.IntResult object.
- GetTopRowNum - Gets the number of the top visible row in the specified table. Sends the result in the ui.IntResult object.
- GotoRowDialog - No information
- GridExport - Exports a grid contents to a clipboard or file, in either a text or html format. Doesn't send any result to the server.
- GridSetCurrentLine - Displays a specific line of the program array into the specified row of the screen array.
- Html5Function - No information
- LaunchUrl - No information
- OpenChildDialog - No information
- Ping - this task is necessary solely to synchronize the server state with the client state, i.e, when all the client updates should be automatically passed to the server. Sends to server ui.PingResult object.
- PrintScreenShot - Prints a screenshot of the current window. Sends the execution result in the ui.PrintScreenShotResult object.
- RequestOauthToken - No information
- SetChildFocus - No information
- SetClientProperty - Sets the properties on the client side. Doesn't send any result to server.
- SetCursor - Moves the cursor in the specified field to a specific position. Doesn't send any result to server.
- SetFocus - Moves the focus to the specified field. Doesn't send any result to the server.
- SetFocusToRow - No information
- SetSelection - Selects the text in the specified field. Doesn't send any result to server.
- ShowCustomMessageBox - Displays a custom simple message box, with a specifiable range of button options. Sends the result to the server in the ui.MessageBoxResult object.
- ShowPopTree - No information
- ShowPromptMessageBox - Displays a dialog box with a field that accepts a value. Sends the result to server in the ui.MessageBoxResult object.
- ShowSvgImage - Shows the svg image to the specified canvas widget. Doesn't send any result to server.
- ShowSystemMessageBox - Displays a simple system message box with a specifiable range of button options. Sends the result to the server in ui.MessageBoxResult object.
- SignalError - This task informs the client that there was an error on the server. It contains the error message. It is sent to the client if there was an error during loading of the recourses on the server.
- SyncTableClassTask - No information
- SyncTableInputTask - No information
- SyncTask - Synchronizes the the states (the number and the stacks of rows) of the virtual table on the client and server sides.

  ui.DeleteRange describes which rows should be deleted from the table child widget and moved to the end of the free rows stack.

  ui.InsertRange describes which rows should be deleted from the beginning of the free rows stack and moved to the table children widget. Deliting should be done before inserting. It doesn't do any changes in Table datamodel structure. Doesn't send any result to server.
- TaskLoadStyleSheet - Applies specified style sheet. Doesn't send any result to server.
- TaskRingBell - This task plays alert sound on client side. Doesn't send any result to server.
- UploadBlob - Uploads a binary file displayed to the BlobViewer element specified in the ui.Viewer prperty.
- UploadFile - Uploads a file in the binary format to the path specified in the ui.ClientPath property .
- WaitChildTask - No information
- WinExec - Executes a specified command. Sends the result to the server in th ui.MessageBoxResult object.
- WriteTextConsole - Shows a message to the console. Doesn't send any result to the server.
- WriteTextViewer - Displays a message to the text viewer. Doesn't send any result to the server.
- WriteToPipe - No information

207 TaskList Not-referenced

207.1 Diagram

207.2 Description

Name: TaskList

The list of the tasks that should be done one by one according to the order in which they are specified.

No parents.

The list of the tasks that should be done one by one according to the order in which they are specified.

207.3 Fields

Name	Type	Description
Seq	list of Task	The list of the tasks.
IsLastTask	Bool	Indecates whether this list is the last one and whether the client should get the interaction or wait for the next task list before the interaction.

208 TaskLoadStyleSheet Not-referenced

208.1 Diagram

208.2 Description

Name: TaskLoadStyleSheet

Applies specified style sheet. Doesn't send any result to server.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

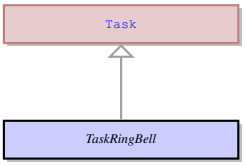
Applies specified style sheet. Doesn't send any result to server.

208.3 Fields

Name	Type	Description
StyleSheet	Any	The style sheet.
IsSystemTheme	Bool	No information
Url	optional String	An URL, generally it requires the explicit specification of the protocol: http, ftp, etc..

209 TaskRingBell Not-referenced

209.1 Diagram



209.2 Description

Name: TaskRingBell

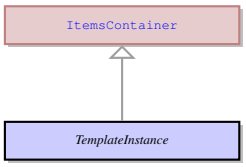
This task plays alert sound on client side. Doesn't send any result to server.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

This task plays alert sound on client side. Doesn't send any result to server.

210 TemplateInstance Not-referenced

210.1 Diagram



210.2 Description

Name: TemplateInstance

No information

Parent: ItemsContainer - 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.

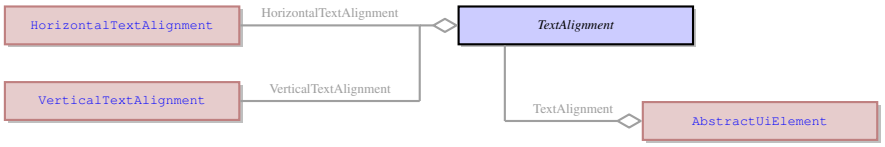
No information

210.3 Fields

Name	Type	Description
TemplateName	optional Name	No information

211 TextAlignment

211.1 Diagram



211.2 Description

Name: TextAlignment

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.

No parents.

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.

211.3 Fields

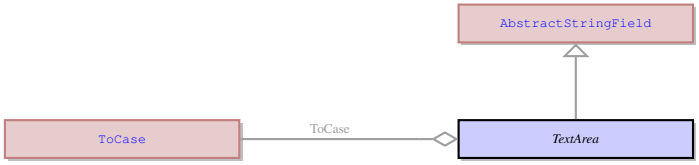
Name	Type	Description
HorizontalTextAlignment	HorizontalTextAlignment	
VerticalTextAlignment	VerticalTextAlignment	

211.4 Referenced in

- TextAlignment field in optional AbstractUiElement - 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.

212 TextArea Not-referenced

212.1 Diagram



212.2 Description

Name: TextArea

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.

Parent: AbstractStringField - 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.

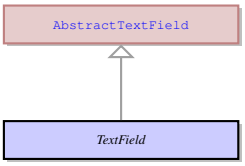
This is a concrete UI element that has the form of a text field and shares many features with ui.TextField , but is designed for working with multiline text instead of single lines of text. It does not have some features of the text field that deal with the navigation between fields, but instead it had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

212.3 Fields

Name	Type	Description
ToCase	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.
TextChanged	Bool	It indicates whether the text displayed in the text area was changed by the user or by the program.
MaxLength	optional 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.
AllowTabulation	Bool	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.
Editor	optional String	Specifies the program to be used for opening and editing the BYTE or TEXT value.
Autonext	Bool	If enabled, moves the cursor to the next field during input automatically, when the MaxLength of the current field is met.
Required	Bool	No information
PlaceholderText	optional String	No information
LabelText	optional String	No information
HelperText	optional String	No information

213 TextField Not-referenced

213.1 Diagram



213.2 Description

Name: TextField

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.

Parent: AbstractTextField - 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.

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.

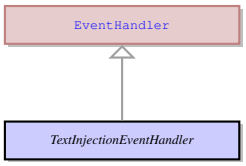
213.3 Fields

Name	Type	Description
AllowNewlines	Bool	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.

InvisibleValue	optional Bool	If enabled, the value displayed to the field will be invisible. During input the value will be masked with *.
PlaceholderText	optional String	No information
LabelText	optional String	No information
HelperText	optional String	No information

214 TextInjectionEventHandler Not-referenced

214.1 Diagram



214.2 Description

Name: TextInjectionEventHandler

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

Parent: EventHandler - This is common class for all the specific event handler types.

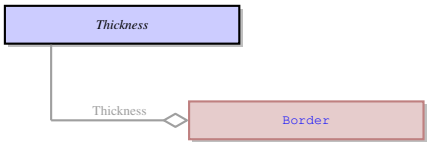
This event handler injects the text specified as its parameter into the current input widget. It can be assigned to any event.

214.3 Fields

Name	Type	Description
Text	optional String	A character string.

215 Thickness

215.1 Diagram



215.2 Description

Name: Thickness

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.

No parents.

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.

215.3 Fields

Name	Type	Description
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.

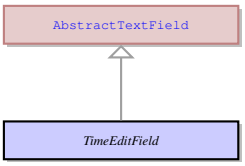
215.4 Referenced in

- Thickness field in optional Border - 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.

216 TimeEditField Not-referenced

216.1 Diagram



216.2 Description

Name: TimeEditField

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.

Parent: AbstractTextField - 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.

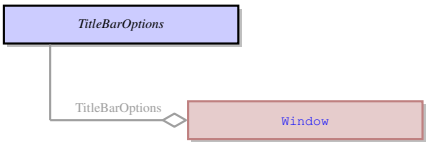
This is a concrete UI element that accepts a limited range of time values. The value inside the field is formatted into hh:mm:ss format. It also has up and down arrows that can scroll the data in the field - whether hours, minutes or seconds are scrolled depends on there inside the field the cursor is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

216.3 Fields

Name	Type	Description
LabelText	optional String	No information
HelperText	optional String	No information
PlaceholderText	optional String	No information

217 TitleBarOptions

217.1 Diagram



217.2 Description

Name: TitleBarOptions

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

No parents.

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

217.3 Fields

Name	Type	Description
DisableTitleBarCloseButton	Bool	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	Bool	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	Bool	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.

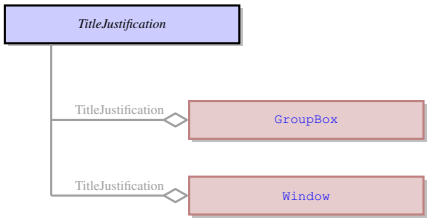
HideTitleBar	Bool	It hides the window title bar together with all its buttons. In this case though the buttons may not have been disabled, thew are still not usable.

217.4 Referenced in

- TitleBarOptions field in optional Window - This UI element unites the options that can influence the default buttons on the 4GL window title bar.

218 TitleJustification

218.1 Diagram



218.2 Description

Name: TitleJustification

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..

No parents.

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..

218.3 Options

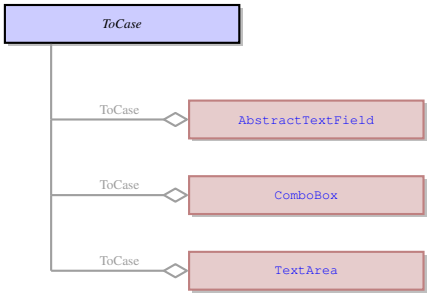
Name	Description
Left	The UI element will be aligned to the left side of the container (or container cell).
Center	The UI element will be equidistant from both sides.
Right	The UI element will be aligned to the right side of the container (or container cell).

218.4 Referenced in

- TitleJustification field in optional GroupBox - 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..
- TitleJustification field in optional Window - 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..

219 ToCase

219.1 Diagram



219.2 Description

Name: ToCase

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

No parents.

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

219.3 Options

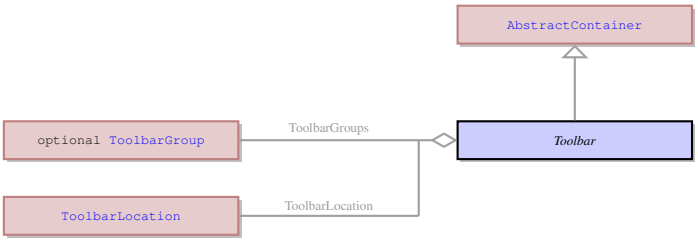
Name	Description
None	The property is not applied and the default behaviour is used.
Up	All the letters entered into the UI element will be uppercase letters regardless of their original case.
Down	All the letters entered into the UI element will be lowercase letters regardless of their original case.

219.4 Referenced in

- ToCase field in optional AbstractTextField - This is the case (lower case or upper case) to be applied to the text in the UI element.
- ToCase field in optional ComboBox - This is the case (lower case or upper case) to be applied to the text in the UI element.
- ToCase field in optional TextArea - This is the case (lower case or upper case) to be applied to the text in the UI element.

220 Toolbar Not-referenced

220.1 Diagram



220.2 Description

Name: Toolbar

This is the container that incorporates toolbar buttons.

Parent: AbstractContainer - 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.

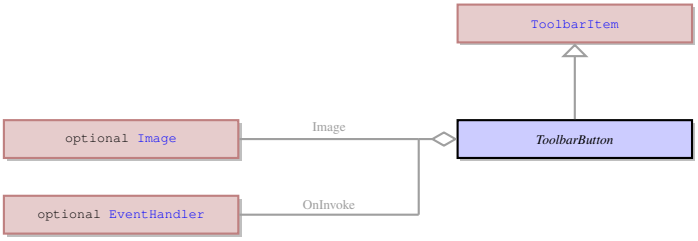
This is the container that incorporates toolbar buttons.

220.3 Fields

Name	Type	Description
ToolbarGroups	list of ToolbarGroup	A set of all toolbar groups that belong to the toolbar.
HideLabels	Bool	It specifies whether the text on the toolbar buttons should be visible or not. If set to true - only the icons will be visible.
ToolbarLocation	ToolbarLocation	No information

221 ToolbarButton Not-referenced

221.1 Diagram



221.2 Description

Name: ToolbarButton

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

Parent: ToolbarItem - This is an abstract element that unites the toolbar buttons and toolbar separators.

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

221.3 Fields

Name	Type	Description
Text	optional String	This is the label of the toolbar button.
AllowNewlines	Bool	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.
Image	optional Image	It specifies the icon that should be displayed to the toolbar button. The button is resized to the size of the icon applied.
OnInvoke	optional 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.

222 ToolbarGroup Not-referenced

222.1 Diagram

222.2 Description

Name: ToolbarGroup

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.

Parent: AbstractContainer - 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.

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.

222.3 Fields

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

223 ToolbarItem Not-referenced

223.1 Diagram

223.2 Description

Name: ToolbarItem

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

Parent: AbstractUiElement - 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.

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

223.3 Children

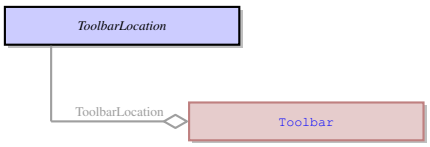
- ToolbarButton - This is an individual toolbar button that belongs to the toolbar.
- ToolbarSeparator - This is a visual separator that can visually divide the toolbar into logical sets of buttons.

223.4 Fields

Name	Type	Description
Place	optional String	No information

224 ToolbarLocation

224.1 Diagram



224.2 Description

Name: ToolbarLocation
No information
No parents.
No information

224.3 Options

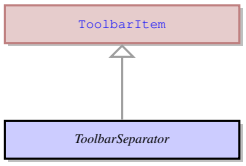
Name	Description
Top	The UI element will be aligned to the top of the container (or container cell).
Right	The UI element will be aligned to the right side of the container (or container cell).

224.4 Referenced in

- ToolbarLocation field in optional Toolbar - No information

225 ToolbarSeparator Not-referenced

225.1 Diagram

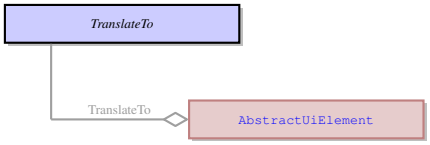


225.2 Description

Name: ToolbarSeparator
This is a visual separator that can visually divide the toolbar into logical sets of buttons.
Parent: ToolbarItem - This is an abstract element that unites the toolbar buttons and toolbar separators.
This is a visual separator that can visually divide the toolbar into logical sets of buttons.

226 TranslateTo

226.1 Diagram



226.2 Description

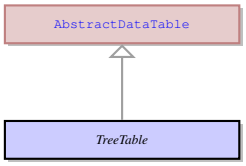
Name: TranslateTo
EMPTY.
No parents.
EMPTY.

226.3 Referenced in

- TranslateTo field in optional AbstractUiElement - EMPTY.

227 TreeTable Not-referenced

227.1 Diagram

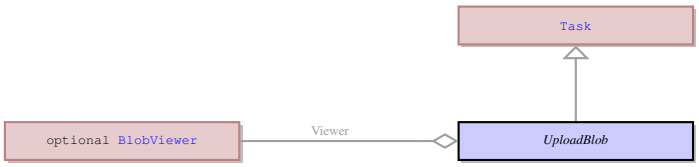


227.2 Description

Name: TreeTable
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.
Parent: AbstractDataTable - 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.
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.

228 UploadBlob Not-referenced

228.1 Diagram



228.2 Description

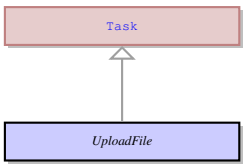
Name: UploadBlob
Uploads a binary file displayed to the BlobViewer element specified in the ui.Viewer prperty.
Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.
Uploads a binary file displayed to the BlobViewer element specified in the ui.Viewer prperty.

228.3 Fields

Name	Type	Description
Viewer	optional BlobViewer	The target blob viewer field.

229 UploadFile Not-referenced

229.1 Diagram



229.2 Description

Name: UploadFile

Uploads a file in the binary format to the path specified in the ui.ClientPath property .

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

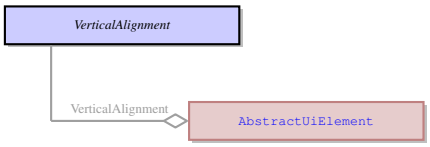
Uploads a file in the binary format to the path specified in the ui.ClientPath property .

229.3 Fields

Name	Type	Description
ClientPath	String	The source file path.

230 VerticalAlignment

230.1 Diagram



230.2 Description

Name: VerticalAlignment

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.

No parents.

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.

230.3 Options

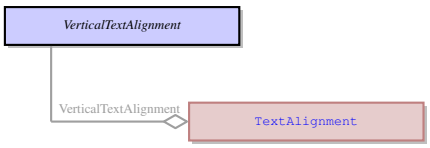
Name	Description
Default	The window size is the size with which it was opened or which was set after opening by 4GL or graphical theme means.
Stretch	The UI element will be stretched to fit the container (or container cell) without preserving the aspect ratio.
Top	The UI element will be aligned to the 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).

230.4 Referenced in

- VerticalAlignment field in optional AbstractUiElement - 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.

231 VerticalTextAlignment

231.1 Diagram



231.2 Description

Name: VerticalTextAlignment

No parents.

231.3 Options

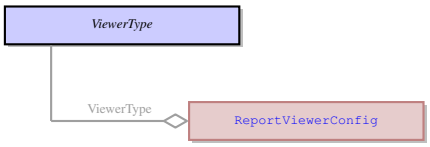
Name	Description
Default	The window size is the size with which it was opened or which was set after opening by 4GL or graphical theme means.
Top	The UI element will be aligned to the top of the container (or container cell).
Center	The UI element will be equidistant from both sides.
Bottom	The UI element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

231.4 Referenced in

- VerticalTextAlignment field in optional TextAlignment -

232 ViewerType

232.1 Diagram



232.2 Description

Name: ViewerType
No information
No parents.
No information

232.3 Options

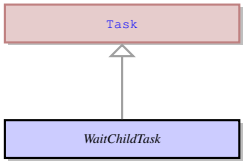
Name	Description
Default	The window size is the size with which it was opened or which was set after opening by 4GL or graphical theme means.
ShellOpen	Not described yet
NewWindow	Not described yet
TextViewer	Not described yet
Download	Not described yet
Print	Not described yet
Inject	Not described yet

232.4 Referenced in

- ViewerType field in optional ReportViewerConfig - No information

233 WaitChildTask Not-referenced

233.1 Diagram

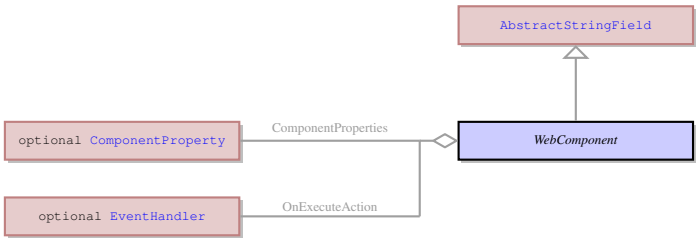


233.2 Description

Name: WaitChildTask
No information
Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.
No information

234 WebComponent Not-referenced

234.1 Diagram



234.2 Description

Name: WebComponent

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.

Parent: AbstractStringField - 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.

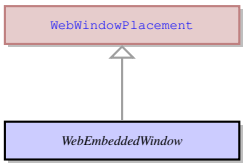
It is a concrete UI element that serves as a container for third party web components. It is basically just the space which is filled by the web component at runtime.

234.3 Fields

Name	Type	Description
ComponentType	optional 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'.
ComponentProperties	list of ComponentProperty	These are specific properties. Their types and number are defines by the HTML file describing the web component.
ComponentPath	optional String	EMPTY.
OnExecuteAction	optional EventHandler	This event is triggered every time any action is executed on the web component - which means any of the gICAPI methods is invoked.

235 WebEmbeddedWindow Not-referenced

235.1 Diagram



235.2 Description

Name: WebEmbeddedWindow

It specifies the behavior of a 4GL window for the web client. The window should be embedded into the browser web page - should not be possible to move around.

Parent: WebWindowPlacement - No information

It specifies the behavior of a 4GL window for the web client. The window should be embedded into the browser web page - should not be possible to move around.

235.3 Fields

Name	Type	Description
WebElementSelector	optional String	Controlling which elements get evaluated by the explorer for actions and state-checking.

236 WebWindowPlacement Not-referenced

236.1 Diagram

236.2 Description

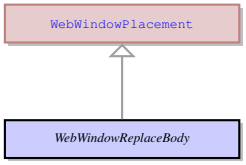
Name: WebWindowPlacement
No information
No parents.
No information

236.3 Children

- FloatingWebWindow - This is a type of a 4GL window in a web client when the window can be moved around inside its container (i.e. inside a page of a web browser).
- WebEmbeddedWindow - It specifies the behavior of a 4GL window for the web client. The window should be embedded into the browser web page - should not be possible to move around.
- WebWindowReplaceBody - It specifies the behavior of a 4GL window for the web client. The window should replace the body of the browser web page.

237 WebWindowReplaceBody Not-referenced

237.1 Diagram



237.2 Description

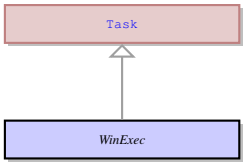
Name: WebWindowReplaceBody
It specifies the behavior of a 4GL window for the web client. The window should replace the body of the browser web page.
Parent: WebWindowPlacement - No information
It specifies the behavior of a 4GL window for the web client. The window should replace the body of the browser web page.

237.3 Fields

Name	Type	Description
ResizeBrowserWindow	Bool	Defines whether the browser window should be automatically resized to fit the 4GL windows embedded into it.

238 WinExec Not-referenced

238.1 Diagram



238.2 Description

Name: WinExec

Executes a specified command. Sends the result to the server in th ui.MessageBoxResult object.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

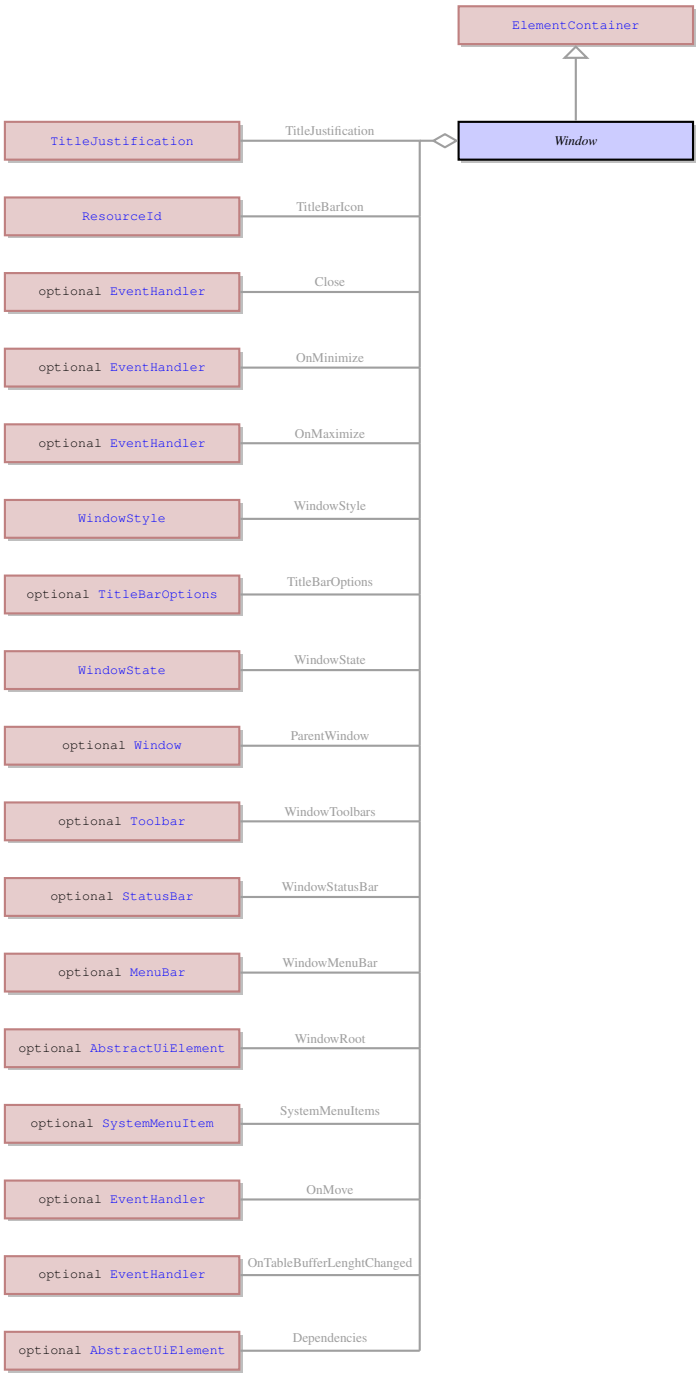
Executes a specified command. Sends the result to the server in th ui.MessageBoxResult object.

238.3 Fields

Name	Type	Description
WinCommand	optional String	The command to be executed.
Verb	optional String	The verb to use when opening the application.
WindowStyleName	optional String	The window style to use when the process is started. It is normal by default.
DoWait	Bool	Forces the client to wait till the procces is finished.
DoByShell	Bool	Indicates whether the process should be started with the operating system shell.
UserId	optional String	The name of the user who runs the application.

239 Window Not-referenced

239.1 Diagram



239.2 Description

Name: Window

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

Parent: ElementContainer - 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.

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

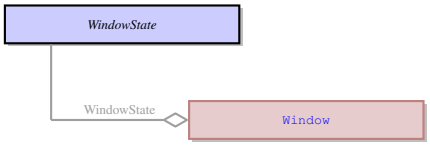
239.3 Fields

Name	Type	Description
Title	optional String	This is the inscription attached to the UI element. Usually this is the text of all sorts of labels.
TitleJustification	TitleJustification	It specifies the horizontal alignment of the text of the title.

TitleBarIcon	optional ResourceId	This is the icon to be displayed in the top left corner of a window - at the left end of the title bar.
Close	optional EventHandler	This event is triggered when the close button on the title bar of a window is pressed.
OnMinimize	optional EventHandler	This event is triggered when the minimize button on the title bar of a window is pressed.
OnMaximize	optional EventHandler	This event is triggered when the maximize button on the title bar of a window is pressed.
WindowStyle	WindowStyle	It specifies whether the window has a border and title bar, or it is a flat window.
MessageLine	optional String	It specifies the position of the line in a window where the output of the MESSAGE statement is displayed.
CommentLine	optional String	It specifies the position of the line in a window where the text of the Comment property of a widget is displayed.
MenuHelpTextLine	optional String	It specifies the position of the line in a window where the descriptions of the ring menu options are displayed.
ErrorLine	optional String	It specifies the position of the line in a window where the output of the ERROR statement is displayed.
RemoveGridHeadings	Bool	It indicates whether the grid headings from the tables inside the window would be removed.
StatusInWindow	Bool	EMPTY - not used.
TitleBarOptions	optional TitleBarOptions	These are options aimed at manipulating the window title bar and its buttons.
WindowState	WindowState	It defines whether the window is maximized, minimized, etc..
DisableReverse	Bool	It negates the effect of the REVERSE 4GL attribute.
RelativeToParent	Bool	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.
HorizontalPadding	optional Int	EMPTY - not used.
HorizontalScale	optional Percents	EMPTY - not used.
VerticalPadding	optional Int	EMPTY - not used.
VerticalScale	optional Percents	EMPTY - not used.
ParentWindow	optional 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.
WindowToolbars	list of Toolbar	This are the toolbars displayed in the window.
WindowStatusBar	optional StatusBar	This is the status bar of the window.
WindowMenuBar	optional MenuBar	This is the menu bar of the window used for the top menus (not for the ring menus).
WindowRoot	optional AbstractUiElement	EMPTY.
SystemMenuItems	list of SystemMenuItem	It is the list of items belonging to the system context menu.
NoResize	Bool	It specifies whether the user will be allowed to resize the window.
OnMove	optional EventHandler	The event is triggered when the position of a UI element is changed.
OnTableBufferLenghtChanged	optional EventHandler	No information
FullScreen	Bool	No information
Dependencies	list of AbstractUiElement	No information

240 WindowState

240.1 Diagram



240.2 Description

Name: WindowState

This enum defines the current state of the window.

No parents.

This enum defines the current state of the window.

240.3 Options

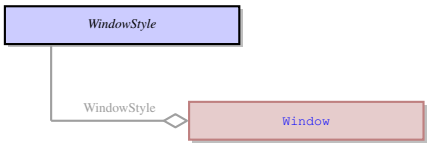
Name	Description
Default	The window size is the size with which it was opened or which was set after opening by 4GL or graphical theme means.
Minimized	The window is minimized to the task bar because the minimize button was pressed.
Maximized	The window is maximized to take up the whole desktop because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

240.4 Referenced in

- WindowState field in optional Window - This enum defines the current state of the window.

241 WindowStyle

241.1 Diagram



241.2 Description

Name: WindowStyle

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.

No parents.

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.

241.3 Options

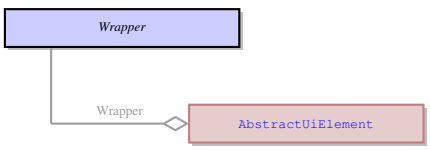
Name	Description
Bordered	The window has border and other attributes associated with it.
Flat	The window has no border.

241.4 Referenced in

- WindowStyle field in optional Window - 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.

242 Wrapper

242.1 Diagram



242.2 Description

Name: Wrapper

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.

No parents.

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.

242.3 Fields

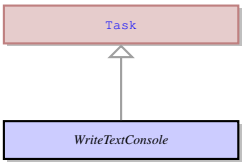
Name	Type	Description
Name	Name	The name of a wrapper
Parameter	optional String	This is the type of the wrapper to be applied to the table.

242.4 Referenced in

- Wrapper field in optional AbstractUiElement - 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.

243 WriteTextConsole Not-referenced

243.1 Diagram



243.2 Description

Name: WriteTextConsole

Shows a message to the console. Doesn't send any result to the server.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

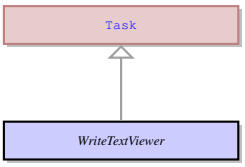
Shows a message to the console. Doesn't send any result to the server.

243.3 Fields

Name	Type	Description
Message	String	The output message.

244 WriteTextViewer Not-referenced

244.1 Diagram



244.2 Description

Name: WriteTextViewer

Displays a message to the text viewer. Doesn't send any result to the server.

Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.

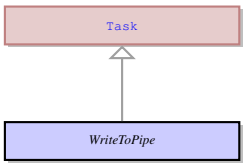
Displays a message to the text viewer. Doesn't send any result to the server.

244.3 Fields

Name	Type	Description
ToolName	optional String	The tool name or command for viewing the ui.Message .
Message	String	The output message.
PageLength	optional Int	The page length.

245 WriteToPipe Not-referenced

245.1 Diagram



245.2 Description

Name: WriteToPipe
No information
Parent: Task - This an abstract entity that serves as a parent for the most of the tasks performed by the client.
No information

245.3 Fields

Name	Type	Description
ToolName	optional String	The tool name or command for viewing the ui.Message .
Message	String	The output message.
PageLength	optional Int	The page length.