Interface Definition of the NETRONIC Visual Scheduling Add-in Developer Toolbox for Microsoft Dynamics 365 Business Central

Valid for the VS Add-in Developer Toolbox as of Version 9.1.8
2026-11-20

Contents

1 Change Log 4
2 Introduction 46
3 System Requirements 47
3.1 Supported Browsers and Versions 47
3.2 Supported BC Versions 47
3.3 Included 3rd Party Libraries and Versions 47
4 Data Model 48
4.1 Activities 49
4.1.1 Activity 49
4.1.2 ActivityEntry 69
4.2 Allocations 71
4.2.1 Allocation 71
4.2.2 AllocationEntry 92
4.3 Calendars 94
4.3.1 Calendar 94
4.3.2 CalendarEntry 95
4.4 ContextMenus 95
4.4.1 ContextMenu 95
4.4.2 ContextMenuItem 96
4.5 Curves 98
4.5.1 Curve 98
4.5.2 CurvePointEntry 99
4.6 DateLine 100
4.7 Entity 104
4.8 HierarchySupplementaryDefinitions 109

© 2026 Boyum IT Solutions GmbH

4.8.1 HierarchySupplementaryDefinition ... 109
4.8.2 HierarchyLevelSupplementaryDefinition ... 110
4.8.3 GroupingLevelDefinition ... 112
4.9 Links ... 114
4.9.1 Link ... 114
4.9.2 LinkEntry ... 117
4.10 PeriodHighlighters ... 117
4.10.1 PeriodHighlighter ... 117
4.10.2 PeriodHighlighterEntry ... 118
4.11 Resource ... 119
4.12 Settings ... 128
4.13 Skill ... 174
4.14 Symbol ... 178
4.15 TableRowDefinitions ... 179
4.15.1 TableRowDefinition ... 180
4.15.2 TableCellDefinition ... 181
4.16 TooltipTemplate ... 185
5 Control Add-in ... 187
5.1 Methods ... 187
5.2 Events ... 200
6 Enumerations ... 211
6.1 BarDesigns ... 211
6.2 BarDragModes ... 212
6.3 BarShape ... 212
6.4 BarSortMode ... 212
6.5 CollapseState ... 213
6.6 CollapseStateTargets ... 213
6.7 CurveInterpolationType ... 213
6.8 CurveType ... 213
6.9 DateLineAppearanceType ... 213
6.10 DateLineCaptionOrientation ... 213
6.11 DateLineCaptionPosition ... 214
6.12 DateLineGridModes ... 215
6.13 DayOfWeek ... 215
6.14 HorizontalAlignment ... 215
6.15 HorizontallyScrollableViewArea ... 215
6.16 HorizontalScrollPosition ... 215
6.17 LinkMarker ... 215
6.18 LinkRoutingType ... 215

© 2026 Boyum IT Solutions GmbH

6.19 ObjectType 216
6.20 PageOrientation 216
6.21 PanningMode 216
6.22 PatternType 216
6.23 PrintingMode 216
6.24 ProgressBarWidthCalculationMode 216
6.25 RelationType 216
6.26 RowDesigns 217
6.27 RowDragModes 217
6.28 RowInsertionMode 218
6.29 RowSortMode 218
6.30 SnapTargets 218
6.31 TableType 218
6.32 TargetPositions 218
6.33 TextWrapMode 219
6.34 TimescaleInteractionModes 219
6.35 TimescaleNavigationMode 219
6.36 TimeType 219
6.37 TreeVisualizationMode 220
6.38 UpdateModes 220
6.39 VerticalAlignment 220
6.40 VerticallyScrollableViewArea 220
6.41 VerticalScrollPosition 220
6.42 ViewArea 220
6.43 ViewType 220
6.44 ViewTypesForDateLines 221
6.45 VisualType 221
6.46 WarningCode 221
6.47 WorldViewPosition 222
APPENDIX 222
Past Release Notes 222

© 2026 Boyum IT Solutions GmbH

1 Change Log

ATTENTION: Please look at the breaking changes in versions 7.0.0 and 8.0.0, when updating the VS-CAI in your application!

ATTENTION: Since we always are searching for performance improvements, we will remove the support for deprecated options and properties with next major version 10.0.0: All settings and properties starting with PM_! So please modify your code as soon as possible!

Version	Description of changes
9.1.8	PATCH: When removing all allocation objects at once using the methods *RemoveAllocations* or *RemoveAll*, then sometimes this was not reflected on screen i.e. the former existent allocations remained visible as rows and bars. After that e.g. any attempt to drag an allocation bar threw an exception. PATCH: Setting one of the settings *IgnoreCalendarOnActivityBarInteractions* or *IgnoreCalendarOnAllocationBarInteractions* was not recognized anymore when set inside the event handler functions for *OnDragStart*. PATCH: Now all events are triggered again one after the other by using an internal queue, originally introduced with version 9.0.6, but not active since 9.1.0.
9.1.7	PATCH: When dragging more than one bar at once, the event *OnDrop* eventually threw an exception, when one of the objects were removed before the event occurred.
9.1.6	PATCH: Under some circumstances dragging a bar threw an exception when auto scrolling was active. PATCH: When dragging a bar and a period highlighter was updated by the application at the same time and that lead to a graphical update of the row where the bar belongs to, the dragging action was aborted, several exceptions were triggered afterwards and the bar became unusable. Now the bar dragging is not aborted anymore. PATCH: Under some circumstances the tooltip started to appear immediately ignoring the value in setting *TooltipDelay*.
9.1.5	PATCH: The chart in some cases was not updated when Activity, Entity, and Resource objects of these types were updated by only changing the *Visible* property of these objects.
9.1.4	PATCH: Top right bar symbols were shown on incorrect horizontal positions when the time area width was wider than 100,000 pixels. PATCH: Text in a table cell now is shown vertically centered, when the height of the table row is smaller than the value of the setting *TableCellContentTopOffset* or *EntitiesTableCellContentTopOffset*, resp., plus the font height of the text. This is what the user expects. PATCH: On resizing bars interactively, the bar phantom was incorrectly drawn when crossing a nonworking time with the pointer. PATCH: Bars using the summary bar shape are now draggable and resizable again and have a hovering feedback again.

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
	PATCH: The font attribute options for the entities table did not work when the entities table was made visible after setting one of the attributes.
9.1.2	PATCH: Undocumented support was removed for script tags with code included in tooltip templates. The old behavior now is considered as being a vulnerability issue. PATCH: Fixed exception when switching visibility of nonworking times off the first time. PATCH: Summary bars now show text again. This document now has better description for font attributes, especially see property FontFamily of Settings objects. This document now describes the following properties not mentioned so far: • On DateLine objects: SymbolHeight, SymbolID, SymbolWidth • On Activity and Allocation objects: TopLeftBarSymbolHeight, TopLeftBarSymbolID, TopLeftBarSymbolWidth, TopRightBarSymbolHeight, TopRightBarSymbolID, TopRightBarSymbolWidth.
9.1.0	MINOR: Now it is possible to show a text per entry in allocation and activity bars: • New properties *Text*, *TextColor*, and *TextWrapMode* on AllocationEntry and ActivityEntry objects. • See also the new font properties for these entry objects. MINOR: Now the application can set font attributes for texts in table title, table rows, bars, and date lines: • New settings *FontFamily*, *FontSize*, *FontStyle*, *FontWeight*. • New settings *TableTitleFontFamily*, *TableTitleFontSize*, *TableTitleFontStyle*, and *TableTitleFontWeight*. • New settings *EntitiesTableTitleFontFamily*, *EntitiesTableTitleFontSize*, *EntitiesTableTitleFontStyle*, and *EntitiesTableTitleFontWeight*. • New properties *FontFamily*, *FontSize*, *FontStyle*, *FontWeight* for TableRowDefinition and TableCellDefinition objects. • New properties *TableFontFamily*, *TableFontSize*, *TableFontStyle*, *TableFontWeight* for Activity, Allocation, Entity, GroupingLevelDefinition, Resource, and Skill objects. • New properties *BarFontFamily*, *BarFontSize*, *BarFontStyle*, *BarFontWeight* for Allocation and Activity objects. • New properties *BarBottomOutsideFontFamily*, *BarBottomOutsideFontSize*, *BarBottomOutsideFontStyle*, *BarBottomOutsideFontWeight* for Allocation and Activity objects. • New properties *BarTopOutsideFontFamily*, *BarTopOutsideFontSize*, *BarTopOutsideFontStyle*, *BarTopOutsideFontWeight* for Allocation and Activity objects. • New properties *FontFamily*, *FontSize*, *FontStyle*, *FontWeight* for AllocationEntry and ActivityEntry objects. • New properties *CaptionFontFamily*, *CaptionFontSize*, *CaptionFontStyle*, *CaptionFontWeight* for DateLine and PeriodHighlighterEntry objects. MINOR: Now it is possible to show inventory curves additionally to the capacity and load curves in a separate curve area in each resource row: • New properties *InventoryCurveID*, *InventoryCurvePaneVisible*, and

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
	InventoryCurvePaneHeight for Resource objects. • New settings *DefaultResourceInventoryCurvePaneVisible*, *DefaultResourceInventoryCurvePaneColor*, *MinimumResourceInventoryCurvePaneHeight*, *DefaultResourceInventoryCurvePaneHeight*, *MaximumResourceInventoryCurvePaneHeight*. • New settings *DefaultResourceBarsPaneVisible* and *DefaultResourceLoadCurvePaneVisible*. • New keyword *>InventoryCurve* for HTMLFormat property of TooltipTemplate objects and TextFormat property of TableCellDefinition objects concerning Resource objects. MINOR: Small graphical enhancements: • New settings *VerticalSplitterWidth* and *TimescaleWeekendTextColor* for further completing the adaptability. • The scroll buttons in the timescale now are placed in the vertical middle of the timescale. • The rectangle besides the timescale above the vertical scrollbar now is filled with the timescale background color. MINOR: Settings *TitleText* and *EntitiesTitleText* changed to *TableTitleText* and *EntitiesTableTitleText*, resp.. MINOR: New keyword *#Activity* for HTMLFormat property of TooltipTemplate objects and TextFormat property of TableCellDefinition objects concerning Resource objects in ActivitiesView with setting AllocationRowsVisibleInActivitiesView set to True. PATCH: 3rd party libraries updated to latest versions: Moment-Timescale 0.6.2 and PDFKit 0.18.0. PATCH: When a table column is too small to show the textual content in a row or in the table title, then the text is shown left aligned ignoring another horizontal alignment setting. This helps the user to read the text. PATCH: When removing all activities, all allocations, all entities, all resources or all skills, the memory for the internal row objects was not fully freed. PATCH: The default background color for rows in the entities table was always white but should show the same behavior as in the Gantt table (gray tones depending on the hierarchy level). PATCH: The caption of date lines was not positioned perfectly, so that sometimes characters were not visible or the text was displaced to the date line itself. PATCH: When dragging a date line and auto scrolling got active at the left or right end of the view, then the date line did not move accordingly. PATCH: When the vertical splitter between table and time area first was shifted by making the widget less wide and then the widget was made wider again, then the splitter could not be dragged to the full table width again. PATCH: The symbols shown for the properties *Status1Color*, *Status2Color*, *Status3Color*, and *BarTopRightSymbolID* of Activity and Allocation objects were not correctly positioned when the time area was wider than 100,000 pixels. The same was the case

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
	with horizontal tick lines in load curve panes. PATCH: When more than one table row definition is used by the table rows and the table title, then shrinking one table column interactively did not work correctly. PATCH: When adding, updating or removing data objects represented as table rows, the corresponding table view width was not always recalculated. PATCH: The methods *SetCollapseStatesForRows* and *SetCollapseStatesForEntityRows* do not trigger the events *OnCollapseStateChanged* and *OnCurveCollapseStateChanged* anymore, because BC does not allow many events to be triggered in short time. PATCH: Fixed an exception when setting the setting *TableTitleAndTimescaleHeight* to 0 while using date lines with property *CaptionPosition* set to TopLeft, TopCenter, or TopRight. PATCH: Added missing ability to set the setting *TableTitleAndTimescaleHeight* initially to 0 and later back to a value > 0. PATCH: The settings *CalendarGridColor*, *DefaultActivityStatusFrameColor,* and *DefaultAllocationStatusFrameColor* did not work when setting their value back to default. PATCH: The setting *DetailedAllocationConstraintSymbolsEnabled* was not working for allocation bars in activities view with setting *AllocationRowsVisibleInActivitiesView* set to true. PATCH: When using the trackpad for horizontal scrolling, the view under the cursor scrolled in the wrong direction. Additionally, it is now possible to scroll horizontally when using the mouse wheel while pressing the Shift key. PATCH: When using the method *SaveAsPDF* the generated document contained only bold texts and the captions of date lines were partly not correct.
9.0.6	PATCH: When moving one or more bars by calling one of the methods *UpdateActivities* or *UpdateAllocations* while showing links that changed their figure because of this, sometimes a link at the mouse cursor position got a hover effect falsely. PATCH: Fix for an active bar dragging or row dragging got aborted, when the setting *NonworkingTimeVisible* was set to False and row objects were updated, even row objects were not visible in the current view. In addition, the user then saw white empty rows. PATCH: Fixed exception when deleting an object when the cursor was hovering on its representation. PATCH: When touching an element, no hover effect should be shown. PATCH: When dropping entity objects in the time area, the event *OnDragEnd* was missing. PATCH: When a row was expanded, the period highlighters rotated with animation in rows where the curve pane is visible. PATCH: Attached date lines on allocation bars or activity bars did not work correctly when setting *TriggeringOfOnShowTooltipForEntriesInBarsEnabled* is set is True. PATCH: Now all events are triggered one after the other by using an internal queue. This avoids BC showing the message for “reduced functionality”.

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
9.0.5	PATCH: Fixed a new issue of Chrome and Edge from version 144 up to at least version 146 not reliably triggering *pointerleave* events inside a BC client anymore. This led to hanging hover effects on buttons and bars, hanging tooltips and context sub menus. Now a workaround is implemented. PATCH: Fixed rare exceptions on vertical scrolling and dropping bars. PATCH: Fixed appearance of the cursor date line after dropping an entity into the time area. PATCH: When moving one or more bars by calling one of the methods *UpdateActivities* or *UpdateAllocations* while showing links that changed their figure because of this, sometimes a link at the mouse cursor position got a hover effect falsely. PATCH: Calling the method *RemoveAll* with parameter ObjectType.Calendar did not work anymore. PATCH: Fixed exception when using *UpdateResources* in the handler for the event *OnDrop* after dragging an allocation bar assigned to a resource to update.
9.0.4	PATCH: The property ViewArea of activity and resource objects did not work anymore, so it was not possible to show them in the top view area. PATCH: When removing all activities in activities view at once or all resources in the other views and afterwards adding new ones, then the old rows now will be removed without animation. PATCH: When removing all activities in activities view at once or all resources in the other views, the memory for the internal row objects was not fully freed. PATCH: Under some circumstances there was an exception when using the context menu by touch.
9.0.3	PATCH: Fixed another exception when the user taps on a period highlighter by touch. PATCH: Fixed exception when the user tries to open the context menu by touch below the chart. PATCH: Fixed exception, when updating the *ParentID* property of a row data object with an unknown ID while grouping is visible (see *HierarchySupplementaryDefinition* object type). PATCH: When updating the property *ParentID* of a data object that was showing a bar in a parent row or when removing such a data object, while its own row is hidden because parent row is collapsed, the parent row was not reduced in height when necessary. PATCH: When the application removes an object by calling a method, while the user drags the representation of it (bar, row, or date line), then the drag-and-drop operation now will be canceled. PATCH: The context menu was not shown when opening it on a symbol within a bar or a table cell and no context menu was defined for this symbol. Now fallbacks to the context menu defined for the bar or table row, resp.. PATCH: When a context menu item is disabled, now a submenu does not open anymore. PATCH: The connection line from the insertion line to the dragged row object on a row

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
	drag-and-drop operation was not shown anymore. PATCH: Dragging a bar using touch by press and pan now works again. PATCH: Dragging a bar horizontally while the settings *NonworkingTimeVisible* set to False and *IgnoreCalendarOnActivityBarInteractions* or *IgnoreCalendarOnAllocationBarInteractions* set to True now works more intuitively. PATCH: The setting *IgnoreCalendarOnAllocationBarInteractions* was not respected when dragging allocation bars in activities view. PATCH: Fixed graphical issues on bar dragging at the absolute start of the time area. PATCH: Smaller fixes on link routing.
9.0.2	PATCH: Fixed exception when tapping on a period highlighter by touch. PATCH: Fixed exception on opening a context menu when an item existed showing a symbol (since version 9.0.1). PATCH: Fixed ignoring options when using the method *SaveAsPDF* (since version 9.0.0). PATCH: When removing all resources by using the method *RemoveResources* or *RemoveAll(ObjectType.Resource)*, the allocation bars did not get visible again, when resources with the same IDs were added afterwards.
9.0.1	PATCH: Fixed BC warning for reduced functionality when using the setting *PingTimeInMilliseconds* and the user opened a dialog. PATCH: Fixed separator lines in submenus of context menu being invisible. PATCH: Fixed eventual exception, when the user modified the table column width interactively. PATCH: Fixed eventual exception when using the method *ProcessOnDrop* in an *OnDrop* event after a bar drag&drop action. PATCH: Fixed eventual exception, when using the method *ScrollToObject* while setting *ScrollToObjectAnimationEnabled* was set to True. PATCH: When a *TableRowDefinition* object was removed then the widget threw an exception on changing the view type to another view that was visible before and used this table row definition. PATCH: When a row showed the first cell of a *TableRowDefinition* with a defined *BackgroundColor*, then the color of the angle cells in dependent rows did not change their color, when the *TableRowDefinition* object was removed. PATCH: When using a value lower than 1 in the setting *VisualZoomFactor*, using the mouse wheel scrolled the table area when the cursor was positioned at the far left of the time area. PATCH: Defining the settings *MaximumTimeResolutionUnit* and *MaximumTimeResolutionUnitFactor* initially or in separate *SetSettings* method calls did not always get visible immediately but after changing the time resolution by user interaction. PATCH: The vertical separation line at the beginning of the fixed column in resources view and skilled resources view was not visible in allocation rows, when they showed no text.

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
	PATCH: Removing an activity object did not remove visible links between allocation bars in the resources view that were created by using activity links (see setting *DefinedAllocationLinksVisibleInResourcesView*). PATCH: Links drawn between bars that are shown in the same sub row with no or low distance to each other now look better again.
9.0.0	**ATTENTION: Since we always are searching for performance improvements, we will remove the support for deprecated options and properties with next major version 10.0.0: All settings and properties starting with PM_! So please modify your code as soon as possible!** MINOR: The context menu now is vertically scrollable when it is higher than the control add-in. This is also true for submenus. MINOR: New setting *OnLogWarningsEventsEnabled* for applications that let the control add-in trigger many warnings. This is only meant for short-time suppression of this event. MINOR: Tooltip functionality extended: • Date lines now can show tooltips. This is accompanied by the new property *TooltipTemplateID* on *DateLine* objects. • Now separate tooltips can be defined for the constraint symbols, for the baseline symbols and bar, and for the outside texts of allocation and activity bars. This is done by setting the appropriate IDs into new properties ending with *TooltipTemplateID* in activity and allocation objects. MINOR: The event arguments for the event *OnDrop* now include the property *ProposedMethodCalls* to help the application in processing the action. If the application wants to execute the proposed method calls, then it can call the new method *ProcessOnDrop* afterwards. This helps to process the event in the back end of the application. MINOR: New options *Start* and *End* can be used in method *SaveAsPDF* to narrow down the time range for the created document. MINOR: Improvements to the user interface: • When auto-scrolling is active while dragging a bar, a row, or a date line, then the hot regions at the borders of the view each now are split in four zones: The more the user drags the object against the border, the faster the view is scrolled. This helps finding the searched drop position faster than before. • The arrow buttons in the timescale now modify the time resolution or time range on DOM event *pointerdown* instead of *pointerup*, so that repetition is possible by holding the pointer down (mouse, touch, or pen). MINOR: The setting *SelectionColor* now also changes the color of the phantom used in row drag & drop. PATCH: Horizontally drawn captions in positions *TopLeft*, *TopCenter*, or *TopRight* of date lines were not visible anymore at the bottom of the timescale. PATCH: When calling the method *ScrollToObject* with *highlightingEnabled* set to *False*, an existing highlighting of other objects was not abandoned.

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
	PATCH: When a load curve used a pure transparent color for the stroke, the stroke of the capacity curve got visible unintendedly. Now the stroke color internally is set to the fill color, when it is transparent. PATCH: When using *PeriodHighlighter* objects on resources, allocation rows shown in activities view were not updated, when removing those period highlighters. PATCH: Removing an activity did not remove the internal copies for resource objects and allocation objects of child activities when using the setting *ResourcesVisibleInActivitiesView*, which led to higher memory consumption. PATCH: Removing the assigned resource from an allocation object did not remove the internal copies of the allocation object when using the skilled resources view, which led to higher memory consumption. PATCH: Removing a skill object did not remove the internal copies of resource and allocation objects when using the skilled resources view, which led to higher memory consumption.
8.4.5	PATCH: When a link was deleted, while it was hovered on screen, the hover shadow remained visible. PATCH: Under some circumstances the numeric scales in rows showing curves disappeared when interactively resizing a table column. PATCH: When setting BottomRowMarginInTimeArea while curve areas are visible, the height and position of the curve areas were not correct afterwards.
8.4.4	PATCH: An exception occurred, when dragging an entity row into the time area. PATCH: Sometimes link target markers were not shown correctly in saved PDF files.
8.4.3	PATCH: Now the complete daylight-saving times for all time zones are included, as far as they are known in the IANA time zone database. In the past we only provided data from 5 years of the past to 5 years into the future from delivery of the control add-in. Because of issues using a control add-in version more than 5 years, we decided to put the whole data into the package. PATCH: When hovering on one of the constraint symbols of a bar, it now is easier to get the tooltip shown, because now the sensitive area is the bounding rectangle of the symbol. PATCH: When a row was selected, the selection frame did not change when changing a TableRowDefinition object. PATCH: When using TableRowDefinition objects and interactively changing the width of a column, this did not update the table rows anymore. PATCH: When the settings AllocationRowsVisibleInActivitiesView and ResourcesVisibleInActivitiesView were both set to True, the resource rows did not show the calendar grid in the activities view. PATCH: When the setting AllocationRowsVisibleInActivitiesView was set to True, the allocation rows did not show the calendar grid in the activities view. PATCH: When one of the properties SourceActivityID or TargetActivityID of a Link object changed, the appropriate allocation links in the resources view were not updated.

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
	PATCH: When using links with entries, switching the routing type let the links disappear. PATCH: When using links with entries, they were invisible if consisted of a single horizontal line. PATCH: When using a link with orthogonal routing type, changing the property TargetMarker of this link from None to FilledArrow, this did not get visible. PATCH: The setting DefaultLinkTargetMarker did not work, so that links got filled arrows, when not setting their property TargetMarker. PATCH: When using the properties TopOutsideBarText or BottomOutsideBarText together with the BarShape set to Symbol on activity or allocation objects, the texts were not visible. PATCH: Using the property CollapseStateInLoadsView for resources in the loads view did not work anymore. PATCH: When changing the default calendar, this was not getting visible in the resources view anymore.
8.4.2	**Documentation added for feature that was implemented for 8.4.0:** MINOR: Now you can define a context menu on HTML elements in the top or bottom status area. See settings BottomStatusAreaContent, and TopStatusAreaContent, and event OnContextMenuItemClicked. PATCH: When using activity bars or allocation bars with outside texts, then the clipping did not work when using constraint symbols. PATCH: When dragging a row, a no-drop cursor was shown when the target position would be the same as the current one for the row. Now the cursor is changed as dropping here would only change nothing but is not forbidden. PATCH: The visualization of the table header for the fixed column at the right side of the table for the curve buttons is not overpainting the column resize separator anymore and the curve expand/collapse buttons are emphasized when overpainting text within the rows. PATCH: Better animation for bars when collapsing or expanding rows and when options changing the row height are set. PATCH: The hover effect did not disappear sometimes when using allocation entries or activity entries with top offsets and/or different heights. PATCH: When dragging an activity row or a resource row, there is no automatic expansion of collapsed allocation rows anymore. PATCH: When setting one of the settings for the table view width, now this also changes the table view width on screen when the settings value is not changed at all, because after user interactions, the settings value is not necessarily the same as the actual view width. PATCH: A call to the Render method now will be ignored when no Add, Update or Remove method was called before. Background here is that the unnecessary call disrupted active animations. PATCH: When a row was newly selected, the gray corridor visualizing EarliestDragStart

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
	and LatestDragEnd on a single selected allocation or activity bar was not disappearing (see setting DragDatesShownForSingleSelectedObject). PATCH: An open context menu now can be closed using the Escape key.
8.4.1	PATCH: When using the method SaveAsPDF some links were not visible in the created PDF document. PATCH: Exception fixed in method UpdateSymbols. PATCH: When using allocation bars with outside texts, then the clipping did not work when collapsing a resource row with CollapsedRowDesign including BarsInHiddenDescendantRows.
8.4.0	MINOR: Now it is possible to show release date and due dates for the baseline of activities. They get visible by showing symbols besides the baseline bar below the main activity bar in the activities view. See the properties BaselineDueDate, BaselineDueDateColor, BaselineDueDateSymbolID, BaselineReleaseDate, BaselineReleaseDateColor, BaselineReleaseDateSymbolID of Activity objects. MINOR: Tooltips optionally now can be entered with the pointer and therefore can be used interactively without causing the tooltip to disappear. This interactivity is enabled by the new IsInteractive property on TooltipTemplate objects. MINOR: New properties BarTopOutsideText and BarBottomOutsideText are implemented now not only on Activity objects, but also on Allocation objects. The texts then are clipped automatically at the date where another allocation bar starts in the same horizontal position. The color of the texts can be set by using the properties BarTopOutsideTextColor and BarBottomOutsideTextColor. MINOR: The new settings BarsDraggable and RowsDraggable allow to separate enabling the drag&drop feature for bars and rows. The setting Editable exists as before and can disable both new settings. MINOR: New property TableColorVisibleInTimeArea on Activity, Resource, and Skill objects. This provides an easy way to show a colored background in the time area. MINOR: When more symbols are to be shown in the symbol column of a row than space is available there, it was needed to click onto the "…" icon to make them visible. Now it is enough to hover the mouse cursor above the symbol column. MINOR: The new method RemoveAllOfTypes now allows to remove all objects of more than one object type at once by providing an array with the object types. MINOR: New key word #FreeCapacity for tooltips by TooltipTemplate object shown on a curve pane. MINOR: The collapse and expand buttons in the table rows now have a slightly changed appearance for better comprehensibility. MINOR: To get more loading performance, the internal processing of calendars and period highlighters is postponed until they get visible on screen now. MINOR: The events OnClicked, OnDoubleClicked, OnDragEnd, and OnDrop now contain the new properties PointerType, AltKey, CtrlKey, MetaKey, and ShiftKey in the EventArgs object. The application can react on these properties that are taken from the original browser's DOM event object if needed. PATCH: The border width and border color of activity bars and allocation bars did not

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
	work as expected anymore. PATCH: When using the value Symbol as the bar shape type, the symbol did not get the appropriate height and width as described in this document. Additionally, now a diamond symbol is shown when no symbol ID is defined by the application. PATCH: When setting the setting ReleaseDueDateConnectionsVisible was set at initiation of the widget, the table view width shrank to zero. PATCH: When dragging the release date symbol or due date symbol of an activity bar while the setting ReleaseDueDateConnectionsVisible was set to True, the connection line was visible and not updated. PATCH: Setting the properties Status4Visible and BarTextPrefixSymbolID on Activity or Allocation objects did not work when used at the same time. PATCH: The context menu now is closed also, when the time resolution changes e.g. by stretching it using the mouse wheel on the timescale. PATCH: Predefined groups of an active HierarchyLevelSupplementaryDefinition were not shown, when no row objects (activities, resources, or entities) exist. PATCH: When resizing a table column interactively, the mouse cursor and the hover effect on the column headers flickered.
8.3.5	PATCH: Exception fixed in call to method removeAll(ObjectType.Link), when world view was visible. PATCH: Adaption of row heights was missing when adding the calendars after adding resources and allocations, while the setting NonworkingTimeVisible was set to false. PATCH: When updating a data object represented by a bar that is currently hovered, the tooltip did not disappear if necessary. PATCH: The callbacks onClicked, onDoubleClicked, onShowContextMenu, and onShowTooltip did not contain the property symbolID for bar symbols. PATCH: When using row drag&drop by dragging the last row in the chart downwards, now the dragged row object is put one level above or when the Shift key is pressed, to the topmost level, when the allowed row drag modes do not give restrictions.
8.3.4	PATCH: Fixed issue for tooltips not updating when two bars lie next to each other. PATCH: Fixed issue of missing update to row heights when changing the options start and end when bars got visible or invisible inside the new time range. PATCH: Fixed exception when calling one of the methods UpdateActivities, RemoveActivities, UpdateAllocations, RemoveAllocations before the first call of the method Render.
8.3.3	PATCH: When changing the time resolution, then links outside of the currently visible part of the chart sometimes were not re-routed at all. PATCH: Using the method SaveAsPDF did not show bar symbols correctly in size and position anymore since version 8.3.0. PATCH: When using one of the methods UpdateActivities or UpdateAllocations while dragging an activity bar or allocation bar, resp., the bar at the original position was not shown half-transparent anymore.

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
	PATCH: Links starting or ending at a bar with a symbol or diamond bar shape did not route perfectly. PATCH: When dragging an activity bar, the event OnDrag was not triggered under some circumstances. PATCH: When dragging a bar vertically, sometimes there was an exception inside the code showing the tooltip. PATCH: When using one of the methods SelectObjects or HighlightObjects before the first call to render, they showed no effect. PATCH: The setting DateLineGridMode with value Auto did not work correctly together with the setting NonworkingTimesVisible set to false. PATCH: When dragging a bar horizontally in the first row, then the shown start date was clipped and therefore not readable. Now the start date is placed below the bar in this case.
8.3.2	PATCH: Fixed exception when clicking using the secondary mouse button into the empty time area below all rows. PATCH: Fixed exception when the settings Start or End are changed so that bars get out of sight. PATCH: Fixed number of tick descriptions on numeric scales of curve panes. PATCH: Fixed color of curve expand/collapse button to the same color as the other collapse/expand buttons.
8.3.1	PATCH: The method RemoveAll did not remove the ContextMenu objects and the method RemoveAllOfType was not able to remove all ContextMenu objects because of a missing ObjectType enumeration value for their object type. PATCH: The use of properties in the setting StatusAreaVariables did not work anymore in the status area content. PATCH: When using a wrap mode in table cells, some text lines could be shown partly below the cell in the last row within the chart. PATCH: Sometimes the navigation buttons in the timescale remained visible although no further reduction of the time resolution was possible. PATCH: The ZIP file created by the setting LoggingEnabled now contains a modification date and time for the included file. PATCH: Fixed an exception when starting to drag a selected bar after deleting another selected bar and the setting MultipleBarDraggingEnabled is True. PATCH: Fixed an exception when selecting an object using the Ctrl key additionally after deleting another selected object. PATCH: The text color and the background color of the numerical scale were false when using color settings in the appropriate TableRowDefinition object. PATCH: When expanding a parent row in the Loads view for the second time, the time areas of all affected rows showed an artifact during the animation.
8.3.0	MINOR: Now you can show a special context menu on symbols:

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
	• New property ContextMenuID for Symbol objects. • New properties SymbolID and SymbolIndex for EventArgs object of the event OnContextMenuItemClicked. MINOR: New properties BarBottomOutsideText and BarTopOutsideText for Activity objects. Accompanying new properties BarBottomOutsideTextColor and BarTopOutsideTextColor. MINOR: New possibility to show date lines when hovering with the pointer over an activity or allocation bar. These date lines then are not shown when not hovering over a bar. The application can provide one single date line object that then is shown on several bars by referencing the properties of the appropriate data object: • New property AttachedDateLineIDs for Activity and Allocation objects. • New properties AppearanceType, CaptionFormat, CaptionColorSource, ColorSource, WidthSource, and DashArraySource for DateLine objects. • New settings DefaultActivityAttachedDateLineIDs and DefaultAllocationAttachedDateLineIDs. • New enumeration DateLineAppearanceType. MINOR: New possibility to select the view types where a date line is shown: • New property IsVisibleInViewTypes for DateLine objects. • New enumeration ViewTypesForDateLines. MINOR: The setting BarSortModeForOptimizedRowDesign now is refined as options ActivityBarSortModeForStackedRowDesign and AllocationBarSortModeForStackedRowDesign. The sort mode can now be defined for activity bars and allocation bars separately. The old option now is deprecated but working for compatibility reasons. MINOR: Now texts can be shown outside the bar rectangle of activity bars: • New properties BarBottomOutsideText and BarTopOutsideText for Activity objects. • Accompanying new properties BarBottomOutsideTextColor and BarTopOutsideTextColor. MINOR: New property AllocationRowsCollapsibleInActivitiesView for Resource object. This helps to define the visibility of the collapse button separately in resources view and activities view. MINOR: New property BarShownUnstackedInBackground for Allocation objects. This serves for showing bars that are eventually already finished in the background with a dimmed design to clarify the resource usage. MINOR: Now there is also a hovering effect on links. MINOR: Captions and the symbols of date lines are again remaining visible when the view is scrolled vertically. This feature was lost beginning with version 8.1.0. MINOR: Log files now can be written as ZIP files directly. See new setting LoggingFileCompressionEnabled. MINOR: The about box now shows not only the names and license texts of the 3rd party libraries, but also license type and the appropriate URL. To give a better overview the list of libraries now is shown collapsed when the dialog is opened and can be expanded by clicking on each library name.

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
	MINOR: The control add-in now uses newer versions of many 3rd party libraries (see also chapter 3.3). MINOR: Extended warning codes in enumeration WarningCode: • New value EmptyArrayOnAddUpdateRemove ('W1004') when providing an empty array when calling an add, update, or remove method. • New value MissingCallToRenderMethod ('W1103') for a missing call to the render method. • New value ParameterValueIncorrect ('W1104') when a parameter value is incorrect in a method call. • New value OptionValueIncorrect ('W1105') when a value is incorrect when setting an option. • New value RelatedIDUnknown ('W2000') when an ID to a related object is unknown when calling an add, update, remove, or render method. PATCH: Under some circumstances calendar entries were not parsed correctly into the internal flat representation of the calendar timeline, especially when the entries were not ordered with ascending start times. PATCH: When instantiating the control add-in using the setting NonworkingTimesVisible with value False and at least one of the settings ending with …SortCodeSources representing rows in the current view type, it was possible that the rows were not high enough for showing the bars in full height. PATCH: Zero-width bars were not hoverable or draggable anymore. PATCH: Fixed exception when using unknown symbol IDs within format templates and status areas were addressed by {{@symbolID}}. PATCH: The PDF document generated by calling the method SaveAsPDF did not contain bars anymore when the setting AsynchronousRendering was set to true. PATCH: The date shown at the mouse cursor of the cursor date line was not positioned correctly, when the view was shown scrolled far down vertically. PATCH: Now a bar represented by a symbol only (see enumeration value Symbol for enumeration and property BarShape on Allocation and Activity objects) is respected by bar stacking (see properties CollapsedRowDesign and ExpandedRowDesign in Activity, Resource, and Skill objects).
8.2.3	PATCH: The setting BarSortModeForOptimizedRowDesign did not work correctly when allocation objects or activity objects were shown unsorted. PATCH: When the chart width exceeded 100,000 pixels (caused by a fine time resolution) and then the time resolution was reduced so that the chart width became lower than 100,000 pixels, then sometimes content of the time area near the end of the time range was not shown. PATCH: When using the setting VisualZoomFactor with a value unequal to 1, then tooltips on curves showed false values shown by using the keys #Date, #Load, #Capacity, and #SingleLoads in TooltipTemplate formats. PATCH: When using the setting AsynchronousInteractiveTimeAreaStretching and actively stretching the time area by using the mouse wheel, the final update on the time area was missing, when clicking into the timescale in between. PATCH: When instantiating the control add-in using the setting

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
	NonworkingTimesVisible with value False and at least one of the settings ending with …SortCodeSources representing rows in the current view type, it was possible that the rows were not high enough for showing the bars in full height.
8.2.2	PATCH: Now the properties SymbolIndex and SymbolID of the EventArgs object are set consistently on the events OnClicked, OnDoubleClicked, and OnContextMenuItemClicked. PATCH: Symbols shown in the full view in the symbol column of table rows now can show a context menu as expected. PATCH: When switching the visibility of the main view area or the top view area, date lines now remain draggable again. PATCH: The context menu items had a minimum width of 150 pixels. Now deactivated. PATCH: Bars were not clipped on the left table when dragging bars horizontally. PATCH: After zooming out the timescale by using the arrow button, it was not possible anymore to pan the timescale. PATCH: Internal exception fixed that occurred when clicking/pressing the timescale for showing a context menu.
8.2.1	PATCH: The selection frame for table rows now works again when the table is scrolled horizontally. PATCH: Zooming out using the Up button in the timescale now works again perfectly also when the time is stretched very wide. PATCH: When using Auto mode in setting DateLineGridMode the grid now scrolls horizontally correct again. PATCH: The caption on a cursor date line and on date lines shown when dragging a bar now is visible again also when the chart is scrolled vertically. PATCH: A bar shown toned down (see property BarDesign of Allocation and Activity objects) was shown with full intensity when the data object was updated. PATCH: When resizing the page with the VS-CAI on it, it could happen that the table view shrunk to size 0. PATCH: When using symbols in context menu items it could happen, that a symbol from a previously displayed context menu became visible. PATCH: When using symbols in the table title together with a fixed table title text (see settings TableTitle and EntitiesTableTitle), the symbols were scrolled horizontally when scrolling the appropriate table horizontally. PATCH: Fixed usage of keyword "#RowObject" within content of properties DisabledSource and SymbolIDSource in ContextMenuItem objects. PATCH: Dragging a bar vertically outside the time area containing rows now is not allowed anymore. PATCH: Now the mouse cursor is changed to a hand when hovering the mouse pointer over a clickable symbol, over a sortable table column title, over the timescale. This is for unifying the UI because the hand was already shown when hovering over an expand/collapse button.

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
	PATCH: Changing the setting ActivityHierarchySupplementaryDefinition did not update the chart. PATCH: Additionally changing the property RowMinimumHeight of a GroupingLevelDefinition and subsequent call to updateHierarchySupplementaryDefinitions did not update the chart when the HierarchySupplementaryDefinition object was used in one of the appropriate settings and therefore active. PATCH: Repaired cursor when dragging an entity row. PATCH: No cancelation of dragging mode anymore when dragging a row and entering a target row that does not allow dragging. PATCH: No dragging of allocation bars was possible after changing the setting ResourcesVisibleInActivitiesView to True. PATCH: The drag mode DragSmartHorOrVer did not work for dragging multiple bars at once. PATCH: Sometimes the horizontal separation line below the last row in the chart was not positioned correctly after expanding a row or after modifying the data objects for the currently visible rows.
8.2.0	MINOR: Visualization of a link now also, when source bar or target bar do not exist. See setting LinksWithDanglingStartOrEndVisible. MINOR: Hover effects unified for several graphical objects: • Clickable symbols in the symbol column of a table (see also properties clickableInTable and clickableInEntitiesTable for Symbol objects) • Buttons for collapsing/expanding child rows • Table column headers when sorting can be switched interactively • Splitters • Timescale ribbon cells and navigation buttons in the timescale • Bars • Draggable date lines • Cursors on table rows for row dragging now follow the same schema as for bar dragging MINOR: New property TooltipTemplateID for Symbol objects for showing own context-specific tooltips on symbols.
8.1.6	PATCH: When changing from a very big time resolution to a far smaller one (e.g. from seconds to days), then sometimes there was an exception that lead to missing separation lines on screen. PATCH: The fixed table title was not shown after switching the view type when the table had a horizontal scroll offset greater than zero.
8.1.5	PATCH: When using a fixed table header (see settings TitleText and EntitiesTitleText), then this was shifted horizontally when the table was scrolled horizontally to the right end. PATCH: When collapsing and expanding a row with bars in child rows with a time distance under the animation duration, then the bars vanished.

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
8.1.4	PATCH: Now the setting MultipleSelectionEnabled with value 0 is supported again (issue since version 8.1.0).
8.1.3	PATCH: Now the context menu is guaranteed to be visible fully when the page is big enough to show it. This also is true for sub menus. PATCH: Fixed long-running loop when using setting DateLineCaptionOptimizedPositioningEnabled set to True in Firefox browser.
8.1.2	PATCH: Fixed exception when dragging a selected bar while another selected bar was not visible due to a collapsed row. PATCH: Fixed exception when dragging an allocation bar vertically with property SuitableResourceIDs set. PATCH: Fixed exception when a curve with no points was used in a curve stack.
8.1.1	PATCH: Fixed exception when opening a context menu. PATCH: Fixed exception in PDF export. PATCH: Under some circumstances it was possible to select activity bars and allocation bars at the same time by using selection by dragging a rectangle. PATCH: When the chart exceeded 100.000 pixels in horizontal direction, the selection rectangle for selecting bars using mouse dragging was not visible. PATCH: When switching top view area on, draggable date line were not shown fully. PATCH: When dragging an entity into the time area, the vertical scroll offset of that was not considered for the shown bar phantom.

8.1.0	MINOR: Internal redesign of DOM structure to support overlay scrollbar behavior of Firefox and Safari directly. This concerned Firefox on Windows 11 having a latency on mouse interactions because of special handling in our code, and it concerned Safari where no scrollbars were shown anymore. As a result, scrolling now shows no latency anymore on all browsers and platforms, but especially on Firefox.
	MINOR: Hovering with the mouse cursor over the collapse/expand symbols or the sorting indicators in the table now shows a shadow as known from other software.
	MINOR: Now DOM elements in the status areas can get their own tooltips defined by the ID of a TooltipTemplate.
	MINOR: Now it is possible to resize curve pane heights on resource rows interactively:
	• PATCH: Renamed setting properties to better fit the naming schema (old names are allowed to use but deprecated):
	  ○ DefaultLoadCurvePaneHeight -> DefaultResourceLoadCurvePaneHeight
	  ○ DefaultLoadCurvePaneColor -> DefaultResourceLoadCurvePaneColor
	• MINOR: New settings CurvePanesResizable, MinimumResourceLoadCurvePaneHeight, MaximumResourceLoadCurvePaneHeight.
	• MINOR: New event OnCurvePaneResized.
	PATCH: When changing the property Selectable on a selected link to false, then the link did not get unselected. Same when changing the setting DefaultLinkSelectable to False.

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
8.0.7	PATCH: Now the context menu is guaranteed to be visible fully when the page is big enough to show it. This also is true for sub menus.
8.0.6	PATCH: Under some circumstances it was possible to select activity bars and allocation bars at the same time by using selection by dragging a rectangle.
8.0.5	PATCH: The events OnTimeAreaViewParametersChanged and OnVerticalScrollOffsetChanged now are delayed when an interaction is active. PATCH: A different tooltip content now also can be shown when the mouse is moved horizontally from one entry to another one on the same period highlighter representation.
8.0.4	PATCH: A grouping code 0 (given as number) was not shown correctly as a grouping row title. PATCH: When all children rows below a grouping row were filtered out (aka got invisible), then the grouping row and its ancestors remained visible. PATCH: When the user selected a sub-context menu item the event OnContextMenuItemClicked occurred also for the parent context menu item. PATCH: In some cases, the optimized placement of a date line caption did not work correctly directly after adding the appropriate date line.
8.0.3	PATCH: Fixed exception when switching on one of the status areas. PATCH: When shrinking the time area resolution, the optimized placement of date line captions did not work correctly. PATCH: Fixed mixed selection of bars or rows of different type. PATCH: Fixed eventual exception when switching views. PATCH: When stretching the time area to a huge extent the bars were not displayed correctly anymore after horizontal scrolling. PATCH: When switching views then eventually link target marker were positioned with a surprising animation. PATCH: Fixed eventual exception when switching grouping off for the current view. PATCH: Fixed an exception when trying to drag an allocation in activities view with visible resource rows while using the property SuitableResourceIDs and not using the property SuitableActivityIDs at the same time. PATCH: Now the longest text on a context menu is not broken into two lines anymore.
8.0.2	PATCH: Fixed issue with sorted rows when a row object was updated. PATCH: When dragging an entity row into the Gantt area, a collapsed entity row laying on the mouse Y coordinate expanded unexpectedly. PATCH: When dragging an entity or bar onto a resource row with no child rows in the Gantt area, then a event OnCollapseStateChanged was triggered without making sense. PATCH: When using tree view lines within a view showing allocation rows, then the lines were broken under certain circumstances.

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
	PATCH: The new bar drag mode DragSmartHorOrVer did not work when switching interactively from horizontal dragging to vertical dragging. PATCH: Fixed issue with allowed target rows when dragging multiple allocation bars at once while using one of the allocation properties SuitableResourceIDs and SuitableActivityIDs. PATCH: Fixed hanging while auto scrolling was active while row dragging. PATCH: When a bar contained a longer text clipped at the end of the bar, then the appropriate tooltip appeared too far at the right side, when the tooltip could not be positioned above, below, or left of the bar. PATCH: When using the method SaveAsPDF, a following call to it was blocked, when the PDF file could not be saved e.g. because the user aborted it. PATCH: After tooltips for curve values were shown, the defined delay for showing all tooltips was ignored from then on. PATCH: After switching allocation rows off and on while using the allocation property Visible the chart did not show the allocation rows anymore and the collapse/expand button on the parent rows disappeared under certain circumstances.
8.0.1	PATCH: Fixed missing creation of baseline bars, due date symbols, release date symbols, or entries shown the time area of parent rows of hidden child rows when changing one of the settings ActivityBaselineBarsVisible, DefaultValuesForAllocationEntryProperties, and DefaultActivityBarDesign. PATCH: In activities view with shown resources (see setting ResourcesVisibleInActivitiesView) curves now are not shown any longer on activity rows (see setting CurvePanesVisibleInActivitiesView). PATCH: On images referenced by Symbol objects that are not quadratic, they were extended with transparent background to be a quadratic image when being used on a bar. Now this is not the case anymore, so that non-quadratic images now are shown at full size when the corresponding width property on the bar object is set. PATCH: Extended EventArgs of event OnDrop by missing properties NewSkillID or NewActivityID, resp., when the view contains multiplied objects. PATCH: Fixed issues when using SuitableResourceIDs in SkilledResourcesView. PATCH: Fixed issues when using SuitableResourceIDs and/or SuitableActivityIDs in ActivitiesView with setting ResourcesVisibleInActivitiesView set to True. PATCH: Fixed issues when dragging allocation bars with drag mode DragSmartHorOrVer and additionally using SuitableResourceIDs and/or SuitableActivityIDs. PATCH: Fixed issue with frames of selected bars remaining visible when the corresponding rows got filtered out. PATCH: Fixed exception on starting to drag a bar after changing one of the options start or end when the bar was positioned outside the visible time range before the change. PATCH: Fixed missing update of rows when switching one of the settings for TopViewAreaVisible and MainViewAreaVisible. PATCH: Fixed an issue within calendar handling when an entry was doubled that additionally was the latest one.

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
	PATCH: Fixed an issue with missing links and date lines after changing setting Start. PATCH: Fixed missing scrollbars on macOS. ATTENTION: Unfortunately, the issue remains open for Safari and will be addressed with the next minor release since it needs more effort. It seems that current Safari versions have a glitch here.
8.0.0	MAJOR (**BREAKING CHANGE**): When the property AllocationRowsCollapseState on activity or resource objects is not set or set to -1 on startup, then the allocation rows are now shown collapsed. In the past they were shown expanded. The change was done to gain startup performance. MAJOR: Library dependencies reduced. No inclusion of jQuery, jQuery UI, and parts of core.js and polyfill-library anymore. Library jquery-ui-contextmenu was replaced by d3-context-menu. So please be aware of the changes when incorporating the VS-CAI in your source code! MINOR: Now it is possible to group allocation rows in activities view by the assigned resources: • MINOR: New settings ResourcesVisibleInActivitiesView, AllocationBarDesignOfOtherActivity, DefaultResourceTableRowDefinitionIDInActivitiesView. • MINOR: New method SetResourcePropertiesForActivities. • MINOR: New property AllocationRowsCollapseStateInActivitiesView for resource objects. • MINOR: New row sort mode for allocations and activities that sorts the rows by ascending start date, see settings ActivityRowSortMode, AllocationRowSortMode, and enumeration RowSortMode. MINOR: For coloring links partition-wise, there is a new property in Link object @Link.Entries and the new object type LinkEntry. MINOR: A new bar drag mode named DragSmartHorOrVer (see enumeration BarDragModes) allows to drag bars focused horizontally or vertically and change the direction after choosing the desired row or time range within the same action. MINOR: Skill objects now can show allocation bars when the resources are collapsed: • MINOR: New property CollapsedRowDesign for skill objects. • MINOR: New setting DefaultSkillCollapsedRowDesign. MINOR: New setting FinishedAllocationBarsShownUnstackedInBackground. MINOR: New setting LinksWithDanglingStartOrEndVisible. MINOR: New settings TableColumnSeparatorColor and EntitiesTableColumnSeparatorColor for coloring the separators between adjacent table columns. MINOR: The keyword {{@symbolID}} as placeholder for a defined symbol is now also available in HTMLFormat property of the TooltipTemplate object. MINOR: New property VerticalAlignment for TableCellDefinition objects. MINOR: Simplification of API: • MINOR: The enumeration BarDragModes replaces ActivityBarDragModes and AllocationBarDragModes. • MINOR: The enumeration BarShape replaces ActivityBarShape and

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
	AllocationBarShape. • MINOR: The callbackArgs parameter ScrollOffset in event OnTimeAreaViewParametersChanged was renamed to HorizontalScrollOffset. ScrollOffset is deprecated now. PATCH: Performance enhancements of about 30% in average when loading data and when switching views. PATCH: When whole rows were made invisible by using the property IsVisible of several data object then the selection frames of selected bars remained visible.
7.1.3	PATCH: Several fixes of missing updates when changing the property Visible of several objects or when changing one of the settings ending with …SortCodeSources. PATCH: Fixed issue of impossible row insertion position as previous sibling on row drag & drop. PATCH: Fixed browser-generated error within method FitTimeAreaIntoView for the case that start is greater than end. PATCH: Fixed missing rendering of links when time area was stretched asynchronously. PATCH: Fixed exception in method ScrollToObject for a not existing skill object.
7.1.2	PATCH: Fixed missing OnCurveCollapseStateChanged events when setting OnCollapseStateChangedTriggeredByUpdateCalls is True (since version 6.4.0). PATCH: Fixed exception when trying to use context menu (since version 7.1.0). PATCH: When the time area was stretched while using the setting AsynchronousInteractiveTimeAreaStretching, the selection frames of bars were not updated horizontally. PATCH: Fixed exception when using one of the methods ScrollToObject or FitTimeAreaIntoView after adding, updating or removing data objects without a call to the Render method in between. PATCH: Fixed missing detection of row insertion mode InsertAsChild, when the target row is less high than default on row dragging.
7.1.1	PATCH: The tooltip did not vanish later if an Allocation or Activity object was updated, while the tooltip was currently shown on just that object. PATCH: Settings MaximumTimeResolutionUnit and TimeStepUnit did not accept a value of the enumeration TimeUnit. PATCH: Texts and symbols were not shown in skill rows when accessed by a TableRowDefinition object.
7.1.0	MINOR: New settings and properties for completion of skilled resources view: • new settings SkillRowSortCodeSources, DefaultSkillAllowedRowDragModes, ForcedSkillAllowedRowDragModes, MainViewAreaVisibleInSkilledResourcesView, TopViewAreaVisibleInSkilledResourcesView • new Skill object properties TableSortCode, TableColorVisibleInTimeArea, ViewArea

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
	• new property TableSortCode for objects used in method setResourcePropertiesForSkill. MINOR: New options, properties, and enumerations for completion of row dragging: • new settings DefaultAllocationAllowedRowDragModes, DefaultAllocationAllowedRowDragModesInActivitiesView, ForcedAllocationAllowedRowDragModes, and ForcedAllocationAllowedRowDragModesInActivitiesView • new Allocation object properties AllowedRowDragModes and AllowedRowDragModesInActivitiesView • new value DragInSameTableParentOnly for enumeration RowDragModes. MINOR: The event OnDrop now provides information about the new start and end dates of entries when dragging activity bars or allocation bars. MINOR: New methods SetCollapseStatesForRows and SetCollapseStatesForEntityRows. PATCH: Settings DefaultResourceMinimumRowHeight, DefaultSkillMinimumRowHeight, and DefaultSkillRowCollapsible did not work in skilled resources view. PATCH: The setting StatusAreaVariables did not work anymore. PATCH: Tooltips faded in without delay (since version 7.0.0). PATCH: Snapping did not work correctly for dragging and sizing of bars.
7.0.2	PATCH: When resizing a table column interactively, the texts within the column were not clipped accordingly under some circumstances. PATCH: Sometimes clicking or tapping on the expand button within the table was ignored when using Firefox. PATCH: The rectangle drawn by properties Status1Color and Status1Visible on allocation bars and activity bars now is fixed in height to be the same as the bottom of the circle drawn by properties Status3Color/Status3Visible. Background: Beginning with version 8.0.0 it was stretched to the height of bar, while the height was taken from setting DefaultAllocationBarHeight or DefaultActivityBarHeight, resp., in previous versions. PATCH: Settings DefaultSkilledAllocationBarTooltipTemplateID and DefaultSkilledAllocationRowTooltipTemplateID did not work. PATCH: When the setting UpdateOptimizationEnabled was set to True, then updates eventually did not work correctly under some circumstances (issue since version 7.0.0). PATCH: Property AddIn_TooltipText in several object types fixed: This deprecated property was not usable anymore. Now it is available again as a fallback to property TooltipText. This is now compliant with the documentation (issue since version 7.0.0).
7.0.1	PATCH: Symbols in table cells were wrongly positioned vertically when the row height was lower than default. PATCH: Fixed exception when scrolling vertically and some rows were filtered out. PATCH: Calendar weekend grid was falsely visible in skill rows and grouping rows. PATCH: Bars with reduced height were not drawn vertically centered (see property BarDesign of Allocation and Activity objects and flag BarDesigns.ReducedHeight).

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
7.0.0	MAJOR (**BREAKING CHANGE**): • Now by default there is no fallback of allocation property values to activity property values anymore. We decided to change the behavior since this improves the performance when updating activities. Also, many customers did not use these fallbacks at all. If the old behavior is needed for your application, you can set the setting DecouplingOfAllocationPropertiesFromActivities to false. MAJOR: New view type SkilledResourcesView: • New object type Skill, new methods Add/Update/RemoveSkills and SetResourcePropertiesForSkill, new enumeration value Skill for ObjectType, new property SkillID for Allocation objects, and new property SkillIDs for Resource objects. • New settings AllocationRowsVisibleInSkilledResourcesView, DefinedAllocationLinksVisibleInSkilledResourcesView, EntitiesTableVisibleInSkilledResourcesView, LinksVisibleInSkilledResourcesView, TableRowDefinitionIDForTitleInSkilledResourcesView, TableViewWidthInSkilledResourcesView in analogy to the resources view. • New settings DefaultValuesForSkillProperties, DefaultSkillMinimumRowHeight, DefaultSkillRowCollapsible, DefaultSkillRowSelectable, DefaultSkillRowTooltipTemplateID, DefaultSkillTableRowDefinitionID. • New event argument SkillID on events OnClicked, OnDoubleClicked, CanDrag, OnDragStart, OnDragEnd, OnDrop, OnCollapseStateChanged, OnCurveCollapseStateChanged, and OnShowContextMenu when referencing a resource row or an allocation row or bar. New properties SkilledRowContextMenuID on resources and allocations, and SkilledBarContextMenuID on allocations. • New possibility to select, highlight, or scroll to a skilled object (see methods SelectObjects, HighlightObjects, and ScrollToObject). • New properties SkilledRowTooltipTemplateID on resources and allocations, and SkilledBarTooltipTemplateID on allocations. New settings DefaultSkilledResourceRow/SkilledAllocationRow/SkilledAllocationBarTooltipTemplateID. • New setting AllocationBarDesignOfOtherSkill for showing allocation bars differently when they belong to another skill than the resource shown below a skill row. • New accessor >Skill for formats used on resources in the new view type (see property BarTextFormat of Allocation objects, property TextFormat of TableCellDefinition objects, or property HTMLFormat of TooltipTemplates) and new accessor #Skill for TooltipTemplate objects on resource rows for referencing the current skill the cursor is hovering above. MINOR: Streamlining and simplification of API: • The prefixes "PM_"/"AddIn_" have been removed from object property names. However, there is no need to change existing code immediately as the former notation will continue to be supported. • New properties BarPatternType and BarPatternColor for Allocation objects analog to Activity objects. • Property TextColor renamed to BarTextColor on Allocation and Activity objects.

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
	• Warning texts now are prefixed with a warning code in event OnLogWarnings (see new enumeration WarningCode) to help developers to understand why the warning has occurred. Also, new warnings are established. • The word “nonworking” now consequently is documented and usable with this spelling. In the code it is allowed to use “nonWorking” everywhere. • The setting OnCollapseStateChangedTriggeredByUpdateCalls was renamed to TriggeringOfOnCollapseStateChangedByUpdateCallsEnabled for better differentiation to events beginning with the prefix “On” and for unified naming. MINOR: Possibility to switch off complexity of bars for gaining performance: • New settings DefaultActivityBarDesign, DefaultAllocationBarDesign, TonedDownOverlayColor, ReducedBarTopOffsetAndHeightScaleFactor. • New property BarDesign on Activity and Allocation objects. • New enumeration BarDesigns. MINOR: More flexibility for showing text: • New setting ApplicationVariablesMap for flexible replacements in formatted text. • New settings IntlDateTimeFormatOptionsMap and IntlNumberFormatOptionsMap for flexible formatting of dates and numbers. • New property TextFormat for TableCellDefinition objects. • New property BarTextFormat for Activity and Allocation objects. New settings DefaultActivity/AllocationBarTextFormat. • (Property InnerHTML of TooltipTemplate objects renamed to HTMLFormat.) • New formatting options for property HTMLFormat of TooltipTemplate objects. • New accessor […] for property accessor strings to get array or map content with dynamic value inside the brackets (see property BarTextFormat of Activities/Allocations, property TextFormat of TableCellDefinitions, or property HTMLFormat of TooltipTemplates). MINOR: Extended graphical representation: • New property TableColorVisibleInTimeArea for the GroupingLevelDefinition object. • New setting SeparationLinesInColoredIndentation. • New properties SymbolHeight and SymbolWidth for TableCellDefinition objects. • New setting ApplicationStyleDefinition e.g. for defining CSS variables. • The color of the three dots in the table symbol column was always black. To improve the readability, the dots now are automatically colored white if the background color of the symbol column cell is a darker one. • New setting DefaultLoadCurvePaneHeight and Resource property LoadCurvePaneHeight. MINOR: Open-source libraries File-Saver and css-element-queries are not included in the code anymore because they were replaced by own code. All other open-source libraries were updated to the latest stable release. PATCH: Missing argument CoupledObjectIDs added to event OnDrop. PATCH: Bars with zero-width were not easy to click or drag. PATCH: Texts of overlapping bars were drawn overlaid.

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
	PATCH: Sometimes when dragging a date line, the end of the interaction was not recognized. PATCH: In Firefox the drag cursor of a draggable date line was not visible.
6.4.5	PATCH: The setting OnCollapseStateChangedTriggeredByUpdateCalls did not work anymore (since version 6.4.0).
6.4.4	PATCH: Several fixes for issues concerning collapsing or expanding child rows, allocation rows, or curve panes in a currently not visible view. PATCH: Bars with zero-width were not visible anymore (since version 6.4.2).
6.4.3	PATCH: Sometimes rows were vanishing when the appropriate data objects were removed and added again together with setting AsynchronousRendering set to True (since version 6.4.0). PATCH: Links eventually were not updated correctly in the chart when row objects were removed and added again within the vertical range of the appropriate link. PATCH: Fixed a hanging cursor on Firefox when clicking on a bar once or twice and then a context menu or a dialog got visible (since version 6.4.0). PATCH: The property PM_CollapseStateInLoadsView of Resource objects did not work anymore.
6.4.2	PATCH: The event OnDoubleClicked was not triggered on Firefox when the chart was bigger and vertically scrolled to the end. PATCH: Fixed display glitches when using the bar shape Regular on activities and allocations with entries that are shown within the coordinate range of the bevel. PATCH: The property DisabledSource of ContextMenuItem objects did not work.
6.4.1	PATCH: Setting options ending with …AllowedRow/BarDragModes with value null falsely set the value to 0. This partly led to an inability to drag rows or bars, resp. (since version 6.4.0). PATCH: When bars overlapped in time only when considering the constraint or the predicted end date, then they were not placed in different sub rows. PATCH: Bars were not visible, when only a constraint or the predicted end date of a bar was inside the time range between options start and end.
6.4.0	MINOR: New possibility to sort the table interactively by table columns, see new settings InteractiveSwitchingOfSortOrderEnabled, SortingIndicatorVisible, RowSortModeNoneEnabledOnInteractiveSwitchingOfSortOrder, and new event OnRowSortingChangeRequested. MINOR: New settings AsynchronousRendering and AsynchronousInteractiveTimeAreaStretching for more performance when many objects are visible on the screen. MINOR: New setting NonWorkingTimesCalendarIDs. MINOR: New setting TimescaleContextMenuID to make it possible to show a context menu on the timescale.

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
	MINOR: New setting CanDragTriggeringEnabled to avoid triggering of CanDrag events, if not needed. MINOR: Additional keywords >SourceAllocation and >TargetAllocation in tooltip templates for links. PATCH: Because of an unwanted change of behavior with version 6.1.0, the tooltip on bars does not flicker anymore when moving the mouse pointer between the entries of an allocation or activity. PATCH: In Firefox the resize cursor for the vertical splitters hung under certain circumstances (since version 6.3.7). PATCH: Eventually a bar could not be resized interactively at the end date when it started before the date in control add-in setting “Start”. PATCH: The keyword #Entry did not work in TooltipTemplate objects for activity bars or allocation bars. PATCH: The method CancelSaveAsPDF did not work correctly when used while the first page was saved to the PDF document. PATCH: When dropping a row near the horizontal separation line, it could happen that the event OnDrop mentioned the neighbored row instead of the targeted one. PATCH: Fixed exception that occurred when hovering over a period highlighter entry in activities view with allocation rows visible.
6.3.7	PATCH: The later addition or update of row objects did not update the horizontal scroll bar of the appropriate table. PATCH: Fixed problem when user presses Escape key while dragging a vertical splitter, a column separator in the table title, or a dateline. PATCH: Fixed exception in Safari when a warning was generated internally (see event OnLogWarnings). PATCH: On macOS the cursor for row drag&drop was not the correct one (only an up arrow instead of an up-and-down arrow). PATCH: Fixed problem with tooltip remaining visible when leaving table rows to an empty table area in Firefox.
6.3.6	PATCH: Interactive resizing of columns did not work correctly when using a visual zoom factor unequal to 1. PATCH: In all events where the table cell index was included in the arguments (OnClicked, OnDoubleClicked), the horizontal table scroll offset and the visual zoom factor were not considered. PATCH: Dragging date lines within Firefox was not possible anymore and fixed missing cursor image when hovering a collapse/expand button within Firefox (since version 6.2.3). PATCH: Scrolling using a trackpad was not accurate.
6.3.5	PATCH: Row dragging did not work for entities when using setting PM_DefaultEntityAllowedRowDragModes set to DragVertically only. PATCH: When using the property Visible on row objects it could happen, that bars for

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
	invisible rows were drawn. Since version 6.3.3.
6.3.4	PATCH: When a period highlighter entry did not reference a ContextMenu object, the context menu of the row object is shown automatically as a fallback now. PATCH: The event CanDrag was triggered too often. PATCH: A visible tooltip was not removed always, when the mouse cursor left a bar. PATCH: When using a bar shape for an allocation different from the default one, this was not always visible on the first allocation entry after updating an allocation object. PATCH: Fixed rare exception when positioning links between invisible bars.
6.3.3	PATCH: Performance improvements for several remove… method calls and for artificial links from activity links in resources view. PATCH: Fixed not working press gesture directly following a double-click. PATCH: Fixed missing cursor icon when hovering an application-defined release or due date symbol of an activity bar. Additionally: This document now lists polyfill-library as included library (see chapter 3.3).
6.3.2	PATCH: Fixed exception in creation of allocation entries representation in allocation bars. PATCH: Row sorting only worked after setting of appropriate setting Activity/Allocation/Entity/ResourceTableSortCodeSources.
6.3.1	PATCH: Performance improvements for several update method calls. PATCH: The event OnDoubleClicked was not triggered anymore (since version 6.3.0). PATCH: The settings Activity/Allocation/Entity/ResourceTableSortCodeSources did not work anymore (since version 6.3.0). PATCH: The curve values displayed in the tooltips did not exactly match the definition in the curve entries due to JavaScript side effects when adding and subtracting values. PATCH: When dragging a row with allowed drag mode DragOnSameLevelOnly target rows are not expanded automatically when the level is below the one of the dragged row. PATCH: Period highlighters were not visible on allocation rows in activities view. PATCH: Rows were too high, when more than one zero-width bar with same start date existed there.
6.3.0	MINOR: Activity, entity, and resource rows optionally can be dragged and dropped vertically inside the appropriate table now: • New settings PM_DefaultActivity/ResourceAllowedRowDragModes. • New settings PM_ForcedActivity/Entity/ResourceAllowedRowDragModes (suitable for providing feedback from the event handler for the event CanDrag). • New property PM_AllowedRowDragModes in Activity and Resource objects. • New enum values DragVertically and DragOnSameLevelOnly for RowDragModes.

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
	• New properties in EventArgs when event OnDrop is triggered. Furthermore, for this now the new enum RowInsertionMode exists. • The event CanDrag is triggered for row dragging. • Precalculation of sort code value when sort mode is set to be ascending and only one property is set (see settings Activity/Entity/ResourceSortCode-Sources). MINOR: New possibility to add status areas above and below the content of the control add-in, defined using HTML text: • New settings Top/BottomStatusAreaHeight and Top/BottomStatusAreaContent. • New setting StatusAreaVariables. MINOR: New settings DefaultValuesFor*Object*Properties with *Object* standing for Activity, ActivityEntry, Allocation, AllocationEntry, Entity, Link, Resource. MINOR: New events OnLogWarnings and OnLogError. MINOR: New setting LoggingVerboseLevel. MINOR: New option Visible for Entity objects. MINOR: New property EntitiesTableViewWidth for callback OnTimeAreaViewParametersChanged. MINOR: New property SymbolIDSource for ContextMenuItem object. MINOR: Properties DisabledSource and SymbolIDSource of ContextMenuItem objects now support referencing properties of sub objects by using a syntax like “PropBag.PropName” (also more than one level allowed). MINOR: New property PM_BorderDashArray for Activity and Allocation objects. MINOR: Symbols on date lines are now also placed optimized. MINOR: Automatic recognition of duplicate IDs and cycles in hierarchy when using ParentIDs on adding or updating objects, see events OnLogWarnings and OnLogError. PATCH: Fixed issues with cursor icon when hovering draggable splitters, column separators, and date lines. PATCH: Fixed display glitches of sub menus of context menus (no shade, different font). PATCH: Fixed internal exception after calling method SaveAsPDF.
6.2.8	PATCH: Fixed missing tooltips on curves when using Firefox (since 6.2.3). PATCH: Sometimes it was not possible to drag the bar onto the original position when interactively dragging bars with settings TimeStepUnit and TimeStepUnitFactor set to coarser values (since 6.2.2). PATCH: It was impossible to gain keyboard focus when using Firefox and clicking or tapping into control add-in element (since 6.2.3). PATCH: Dragging of bars was not always possible when using a hatch pattern. PATCH: The context menu did not work on period highlighter entries. PATCH: It was not possible to access neither row object properties using #RowObject nor period highlighter properties in tooltip templates assigned to period highlighter entries.

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
	PATCH: When showing allocation rows in activities view, selecting bars by drawing a rectangle optically selected activity and allocation bars. PATCH: When setting Editable was set to false, you could nevertheless drag entities into the time area. Additionally, changing the setting at run-time had no effect. PATCH: When dragging entities into the time area with settings TimeStepUnit and TimeStepUnitFactor set to coarser values, then the calendar was not continuously considered. PATCH: Dragging bars with setting TimeStepUnit set to “year” did not work at all. PATCH: Setting PM_BarSortModeForOptimizedRowDesign did not work. PATCH: When changing setting PM_BottomRowMarginInTimeArea, row heights were not updated concerning a currently visible curve pane. PATCH: When clicking/tapping onto a curve, sometimes an exception occurred, and sometimes selected elements were not deselected. PATCH: When hovering bars with the mouse cursor fast, then it could happen, that an error message occurred in BC because of too many CanDrag events. Now the CanDrag event is triggered slightly delayed and the number of events is therefore reduced. PATCH: Neither the event OnSelectionChanged was triggered nor selected entities were deselected, when the user clicked/tap on the time area background or on curves. Attention: When using setting PM_DateLineCaptionOptimizedPositioningEnabled, the optimization of caption positions does not work when symbols are used on the date lines! This will be fixed with upcoming version 6.3.0.
6.2.7	PATCH: More performance when using Add methods for Activity, Entity, Resource objects. PATCH: Fixed additional horizontal scrollbar for top view area in Firefox (since version 7.5.3). PATCH: No line break anymore on context menu items. PATCH: Symbols in context menu items are now shown with a height of 14px instead of 8px before.
6.2.6	PATCH: Fixed functionality of property PM_BorderColor of Activity objects when entries are existing. PATCH: Fixed functionality of property PM_BorderDashArray of Activity and Allocation objects. PATCH: Fixed missing values when evaluating a TooltipTemplate for a tooltip (since version 7.5.3).
6.2.5	PATCH: Fixed false closing the tooltip when moving the mouse pointer from one bar to another one (since version 6.2.1). PATCH: Fixed internal exception when adding resources, entities, or activities with IDs that were removed before under some circumstances (since version 6.2.1). PATCH: Fixed not visible links in resources view when using activity links and adding allocation objects again after removing all existing allocation objects before (since

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
	version 6.2.1). PATCH: Fixed issue of rows remaining invisible after updating the property ParentID of Activity, Entity, or Resource objects (since version 6.2.1). PATCH: Fixed false interpretation of property PermissionToPrint in options object of method SaveAsPDF.
6.2.4	PATCH: Property DisabledSource of ContextMenu object did not work for custom properties. PATCH: Fixed an update issue when using the Activity property Visible to hide activity bars on collapsed activity rows and showing the bars in ancestor rows. PATCH: Fixed a performance issue when using the Allocation property Visible with a big number of allocations. PATCH: The about dialog now shows the open-source library “core-js” as an included component (see also chapter Included 3rd Party Libraries and Versions).
6.2.3	PATCH: The control add-in now supports showing scroll bars when using Firefox beginning with version 100 on Windows 11. PATCH: When using Firefox the inner DIV elements of the control add-in were focusable by using the Tab key. Now this is prevented. PATCH: The CSS files of VSW were not compatible to IE11 anymore since version 6.2.0 (only important for users of Windows client of NAV 2018 and BC 14). PATCH: Fixed an exception when the user clicked into the background of a row within the time area. PATCH: In event OnTimeAreaViewParametersChanged the value for TableViewWidth was missing and not documented.
6.2.2	PATCH: When showing the context menu on a currently selected object all selected objects were deselected since version 6.2.1. PATCH: The shown symbols for the ReleaseDate and DueDate properties of an Activity object were overdrawn by the activity bar when the dates in properties Start and End were in same range. PATCH: Interactively resizing bars with settings TimeStepUnit and TimeStepUnitFactor set to coarser values was not showing a smooth phantom. Attention: The control add-in does not show scroll bars when using Firefox beginning with version 100 on Windows 11. This will be fixed in a later patch release. As a workaround you can change the setting “Always show scrollbars” in System Preferences > Accessibility > Visual Effects.
6.2.1	PATCH: Much more performance when changing the property ParentID of Activity, Entity, or Resource objects. PATCH: Activity links in resources view under circumstances remained invisible when added in startup phase of widget. PATCH: When using activity links in resources view, these were not selectable interactively.

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
	PATCH: When using multiple bar dragging with one of the settings or properties concerning …AllowedBarDragModes set to DragHor+DragVer, horizontal dragging of bars in more than one resource row at the same time did not work as expected. PATCH: In some cases, the mouse cursor was not cleared when leaving a bar on screen. PATCH: The context menu of the browser is suppressed now on the timescale and in the fixed symbol column of the table when using the secondary mouse button.
6.2.0	MINOR: New property Visible for Activity, Allocation, and Resource objects. MINOR: New method RemoveAllOfType. MINOR: New options TopText, BottomText, TopHTML, BottomHTML, OwnerPassword, UserPassword, and PermissionTo…, Author, Title, Subject, Keywords for method SaveAsPDF. (For using TopHTML and BottomHTML the additional open-source library "html2canvas" is needed.) MINOR: New setting PM_DateLineCaptionOptimizedPositioningEnabled. MINOR: New property PredefinedGroups for HierachyLevelSupplementaryDefinition objects. MINOR: Renaming of the following properties in GroupingLevelDefinition objects for alignment with a fallback to the older property names: • CodeToTextMap to GroupingCodeToTextMap, • CodeSource to GroupingCodeSource, • TableBackgroundColor to TableColor. MINOR: New properties ScaleMinimumValue and ScaleMaximumValue for Curve objects. MINOR: New setting PM_LinesShownInLoadCurvePanes. MINOR: New property SuitableResourceIDs for Entity objects and new property SuitableActivityIDs for Allocation and Entity objects. MINOR: New settings PM_Suitable/UnsuitableActivityOverlayColor. PATCH: Method ScrollViewVertically was not callable. PATCH: Fixed positioning when dragging a date line or the vertical splitter and the setting VisualZoomFactor was set to value unequal to 1. PATCH: When saving a PDF document, symbols shown at the top of date lines were not exported. PATCH: More performance when updating activity objects. **New chapter named "System Requirements". Please have a look!**
6.1.8	PATCH: The world view sometimes was scaled to high, so that the view rectangles were not visible fully. PATCH: When an active HierarchySupplementaryDefinition object was updated the changes did not get visible in the entities. PATCH: The mouse cursor changed too often on allocation bars with allocation entries since version 6.1.7.

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
6.1.7	PATCH: When dragging allocation bars horizontally erroneously the DragMode property in event OnDrop had flag DragVertically switched on. PATCH: A defined symbol on a date line was invisible when either width or height left undefined. PATCH: All resources were grayed on dragging multiple allocation bars when the property SuitableResourceIDs was empty on at least one of the Allocation objects. Now an empty value is interpreted consistently on single and multiple bar dragging. PATCH: The mouse cursor did not change to “resize” in certain cases on allocation bars. PATCH: When dragging an allocation bar vertically the non-working time was not updated, when it should be visible inside the bar. PATCH: Overwriting the setting PM_SymbolColumnTitleSymbolIDs with the same value as before took too much time.
6.1.6	PATCH: After calling ScrollToObject with an allocation the bar representing this allocation was not visible fully when it was out of sight before and is positioned below the first sub row. PATCH: When the setting LoggingEnabled was set to true in the initiation options of the widget, the red recording button was not shown on screen. PATCH: When using SaveAsPDF without setting ZoomFactorInPercent, HorPageCountLimit, and VerPageCountLimit, the resulting PDF document eventually contained more than page.
6.1.5	PATCH: After calling method SelectObjects the time area did not work normally, e.g. rescaling by using the timescale did not update the time area anymore. PATCH: When a row object shows a selection frame the sensible area around the collapse/expand button was smaller than without the selection frame. PATCH: In a special case the call to FitTimeAreaIntoView started an animation for horizontal scrolling and hindered a following call to ScrollToObject.
6.1.4	PATCH: Fixed false scroll position when method ScrollToObject was called with setting PM_ScrollToObjectAnimationEnabled set to true. PATCH: Now horizontal scrolling to begin of the time area is avoided when method ScrollToObject is called for an activity or an allocation object that has no defined start date. In this case now only vertical scrolling is done. PATCH: Setting a non-existent ID into the property ParentID of an activity, resource, or entity object using an update method was not working anymore. PATCH: The revised resize mechanism introduced in version 6.1.1 now was refined because of an issue in one application.
6.1.3	PATCH: After calling the method ScrollToObject with an entity object, highlighting was not working anymore. PATCH: In TooltipTemplate objects used for curve tooltips the reserved words #Load, #Capacity, #Date, #SingleLoads did not show the appropriate values.

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
6.1.2	PATCH: Fixed missing properties PeriodHighlighterID and PeriodHighlighterEntryIndex on events OnClicked and OnDoubleClicked when VisualType is PeriodHighlighter. Also repaired tooltips on period highlighter entries.
6.1.1	PATCH: Time axis could not be shrinked enough anymore when using a big time resolution unit step. PATCH: Fixed internal exception handling. PATCH: Fixed crash, when settings Start and End were set to null. PATCH: Fixed missing update of allocation rows in activities view after updating PeriodHighlighter objects. PATCH: Fixed exception when dragging an activity bar where the property PM_HasAllocationRows of the activity is set to true without being assigned to existing allocations. PATCH: Fixed hanging mouse cursor when leaving links. PATCH: Fixed issue with actual time resolution when settings MaximumTimeResolutionUnit and/or -Factor were modified again after widget instantiation. PATCH: Fixed exception after updating activity objects while allocation rows are shown in activities view. PATCH: Fixed issues with automatic resize of control add-in when one or more page elements were placed below it. The changes caused that the setting ForcedFillOfPageSpace now is obsolete and will be ignored. PATCH: When using Allocation.SuitableResourceIDs on a bigger data model, it took longer to start dragging on bars. PATCH: When adding allocation rows right on expanding the containing row these remained invisible. **Hint for AL users: We have modified the code for being compatible with BC20. The event OnPing now is triggered only when BC client is not busy. And maybe you will have to set the setting OnCollapseStateChangedTriggeredByUpdateCalls to false to avoid errors of BC20. We did not change the behavior because we do not know whether your code relies on the event or not.**
6.1.0 **(See also release notes!)**	MINOR: New property Entries for Activity objects and new object type ActivityEntry. MINOR: Far more possibilities for context menus by new properties Disabled, DisabledSource, SubItems, SymbolID for ContextMenuItem objects. MINOR: New setting MaximumTimeResolutionUnit and MaximumTimeResolutionUnitFactor. New values for settings MaximumTimeResolutionUnit, TimeStepUnit, and for unit parameter of SetTimeResolutionInView method. MINOR: New setting PM_TimescaleInteractionMode. MINOR: New property DefaultCode for GroupingLevelDefinition objects. MINOR: New setting PM_ResourceHierarchySupplementaryDefinitionIDInLoadsView. MINOR: New settings UpdateMode and ResetValueForDifferentialUpdate. New

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
	enumeration UpdateModes. MINOR: In TooltipTemplates single curve values are now accessible by using #SingleLoads.*curveID*. MINOR: New setting OnCollapseStateChangedTriggeredByUpdateCalls. PATCH: Switching the grouping on or modifying the grouping now is much faster. PATCH: Fixed an issue with missing animation on grouping modifications. PATCH: Fixed an issue when modifying property TableRowDefinitionID in HierarchyLevelSupplementaryDefinition objects. PATCH: Fixed the visibility of allocation object values in tooltips generated by tooltip templates on allocation bars in resources view. PATCH: When an allocation referenced a non-existing resource, it could not be filtered in activities view. PATCH: Fixed world view issues concerning scaling and scroll bar visibility. PATCH: Fixed an issue in PDF export when the time range is huge. PATCH: Fixed an issue when modifying the property PM_ViewArea on row objects that have visible allocation rows.
6.0.3	PATCH: Fixed an issue with PDF export newly appeared with version 6.0.2. PATCH: The parameter ObjectType of event OnContextMenuItemClicked had the wrong value 3 in case of an allocation bar since version 6.0.2. PATCH: Period highlighter grids assigned to resources are now shown also in allocation rows of these resources, in analogy to calendar grids.
6.0.2	PATCH: The colorization for the grouping and hierarchy levels in the table was not working fully caused by vertical virtualization that is internally used to get more performance. PATCH: Event OnVerticalScrollOffsetChanged now also works correctly when grouping is used.
6.0.1	PATCH: Missing parameter “date” in OnClicked event when time area background was clicked and new setting ClickEventTriggeringOnRowInTimeArea is used. PATCH: After changing the group criteria in the data of a row object with active grouping, the grouping was not updated. PATCH: When changing the data of an activity object while allocation rows are visible in the activities view, then the allocation rows for this activity object disappeared. PATCH: The new tree view feature did not work correctly when the top view area is visible and the property PM_ViewArea was changed.
6.0.0	MINOR: New additional grouping by criteria within the current parent-child hierarchy (only available in AL version!): • New objects HierarchySupplementaryDefinition, HierarchyLevelSupplementary-Definition, GroupingLevelDefinition to specify grouping. • New settings PM_Activity/Resource/EntityHierarchySupplementaryDefinitionID to

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
	specify the active hierarchy supplementary definition object for additional grouping of activity/resource/entity objects appearing as rows in the appropriate table. • New parameters for events OnClicked and OnDoubleClicked when grouping rows are affected. MINOR: New method HighlightObjects and therefore renamed settings PM_ScrollToObjectHighlightingColor to PM_ObjectHighlightingColor and PM_ScrollToObjectHighlightFlashingEnabled to PM_ObjectHighlightFlashingEnabled. MINOR: New bar shape named Symbol for allocation and activity bars: • New enumeration value named Symbol in enumerations AllocationBarShape and ActivityBarShape used in property PM_BarShape of allocation and activity objects. • New properties PM_BarShapeSymbolID and PM_BarShapeSymbolWidth for allocation and activity objects. MINOR: The bar shape named Diamond now is usable additionally for allocation bars. MINOR: To improve a more compact layout optionally there are new settings PM_Allocation/ActivityBarTopOffsetAndHeightScaleFactor, PM_EntitiesTableCellContentTopOffset, PM_TableCellContentTopOffset, PM_TableTitleAndTimescaleHeight, PM_EntitiesTableTitleHeight, PM_ProgressBarHeight. MINOR: New settings TreeVisualizationMode, PM_TreeViewLineColor/DashArray, EntitiesTableTreeVisualizationMode, PM_EntitiesTableTreeViewLineColor/DashArray and new enumeration TreeVisualizationMode. MINOR: New property PM_StrokeDashArray for Curve objects. MINOR: New setting PM_BarSortModeForOptimizedRowDesign, new enumeration BarSortMode. MINOR: New setting PM_ClickEventTriggeringOnRowInTimeArea. MINOR: New settings PM_SymbolColumnTitleBackgroundColor and PM_EntitiesTableSymbolColumnTitleBackgroundColor. MINOR: New property PM_RowSymbolColumnBackgroundColor for Activity, Allocation, Entity, and Resource objects and new property SymbolColumnBackgroundColor for TableRowDefinition objects. MINOR: Settings PM_Top/MainViewAreaVisible for resources view extended to PM_Top/MainViewAreaVisibleInActivities/Loads/ResourcesView to cover activities view and loads view additionally. New property PM_ViewArea for activity objects. MINOR: Extension of possible context menus on visual elements: • New property BarContextMenuID for allocations and activities. • New property ContextMenuIDInBarsPane for allocations, activities, resources. • New property ContextMenuIDInCurvesPane for activities and resources. PATCH: Method UpdateTableRowDefinitions in C/AL version now usable. PATCH: When one of the settings PM_IgnoreCalendarOnAllocation/ActivityBar-Interactions is set to True, then bar dragging was not perfect when visible or even invisible non-working times were touched.
5.3.4	PATCH: Internal update to current NWAF libraries (version 6.10.2):

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
	• PATCH: More performance when using settings Activity/Allocation/Entity/-ResourceSortCodeSources.
5.3.3	PATCH: Internal update to current NWAF libraries (version 6.10.1): • PATCH: PDF export fixed (issue since 6.10.0). • PATCH: The event OnClicked was not called on curves anymore. • PATCH: In case where allocation rows are visible in resources view, collapsing a resource row did not lead allocation links disappear where needed.
5.3.2	PATCH: Internal update to current NWAF libraries (version 6.10.0): • PATCH: Fixed graphical issue when canceling dragging of a date line. • PATCH: Bars without start and end dates are not shown anymore. • PATCH: Snapping while dragging a bar now also works for date line grids in mode Automatic. • PATCH: World view now cannot get higher/wider than widget extent anymore. • PATCH: Fixed issue when dragging more than one allocation bar and the property SuitableResourceIDs is used at least on some of the allocations. • PATCH: Setting one of the properties PM_CollapseState and PM_CollapseStateInLoadsView for Resource objects is now also working when resources view or loads view, resp., is not visible.
5.3.1	PATCH: Internal update to current NWAF libraries (version 6.9.4): • PATCH: Fixes snapping to start and end dates of other allocations when dragging an allocation. • PATCH: Fixes an issue for the settings PM_IgnoreCalendarOnActivity/Allocation-BarInteractions when updating objects while dragging. • PATCH: Fixes an eventual exception when setting PM_ActivityBaselineBarsVisible was set to True.
5.3.0 **(See also release notes!**	MINOR: New method ScrollToDateWithOffset. MINOR: New property PM_CollapseStateInLoadsView for Resource objects. MINOR: New settings PM_SymbolColumnBackgroundColor and PM_EntitiesTableSymbolColumnBackgroundColor. MINOR: New setting PM_IgnoreCalendarOnActivityBarInteractions. MINOR: New event OnSaveAsPDFProgress and new method CancelSaveAsPDF. MINOR: New enumeration PatternType, new properties PM_BarPatternType and PM_BarPatternColor on Activity objects, and new properties PM_PatternType and PM_PatternColor on AllocationEntry objects. MINOR: New settings PM_DateLineGridColor, PM_DateLineGridDashArray, and PM_DateLineGridWidth. MINOR: New option PM_ActivityBaselineBarsVisible. MINOR: New link property PM_TargetMarker and new enumeration LinkMarker. PATCH: Internal update to current NWAF libraries (version 6.9.3): • PATCH: Interactive activation of logging (Shift+Ctrl+Alt+L) now also works with Polish keyboards.

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
	• PATCH: Fixed exception when modifying the property ParentID of an Activity, Entity, or Resource object with an ID of a non-existing object. • PATCH: Fixed issues with colored background rectangles in the table when saving a PDF document. • PATCH: Fixed issue when canceling dragging of a date line by pressing Escape key. • PATCH: Fixed positioning issues when using bar diamond shapes. • PATCH: Fixed issue when dragging a release date symbol or due date symbol in Firefox, when the symbol is user-defined. **Hint for AL users: Some files have changed its name! The content of sub directory "src" is moved to one level above!**
5.2.3	PATCH: Internal update to current NWAF libraries (version 6.7.0): • PATCH: Issue fixed for invisible curve pane when resource has set property PM_CurveCollapseState to 0 and was added before its parent resource. • PATCH: Fixed issues when using curves of type List. • PATCH: Fixed issues concerning symbols in table title. • PATCH: Watermarks were not scaled on screen anymore since version 6.6.1. • PATCH: When using the secondary mouse button while dragging the action now is canceled. • PATCH: Fixed issue when sizing a table column interactively that has a background color. • PATCH: Sometimes the saved PDF file showed collapsed allocation rows and vice versa. • PATCH: Fixed exception when user clicked into timescale with visible world view. • PATCH: Fixed issue of not resolving object references in an applied tooltip template. • PATCH: Fixed issue of disappearing row(s) at the bottom when user expanded some curves or row objects under special conditions.
5.2.2	PATCH: Internal update to current NWAF libraries (version 6.6.1): • PATCH: Bug fixed concerning adding allocations after first refresh in resources view. • PATCH: Issue fixed in method SaveAsPDF concerning referenced SVG images not visible in PDF. • PATCH: Infinite loop fixed in method SaveAsPDF when many images are not loadable. • PATCH: Performance issue fixed concerning allocation rows in resources view. • PATCH: The event arguments for the events OnClicked, OnDoubleClicked did not fill the property CellIndex when triggered for a table row. • PATCH: Fixed issue of not recognizing the setting PM_AllocationRowsVisibleInActivities/ResourcesView before first call to Render method. Workaround was to change the view type twice. • PATCH: In some situations, the allocation bar was not vanishing on the source row in resources view after dragging vertically to another row. • PATCH: Delivered JavaScript files now are prefixed with a UTF8-BOM since in one case concerning Firefox they were misinterpreted as being encoded in ANSI. • PATCH: Fixed hanging issue when updating resource objects or period

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
	highlighters within drag & drop interaction of allocation bars. • PATCH: Bars within a row object disappeared when the grandparent row object was collapsed, and the bars should remain visible (see PM_CollapsedRowDesign) and the parent row object was not collapsed. • PATCH: Property PM_AllocationRowsCollapseState of Resource objects was not working. • PATCH: Crash fixed when calling method SaveAsPDF without any links in the chart. This document lacked documentation for the properties PM_AllocationRowsCollapsible, PM_AllocationRowsCollapseState for Resource objects introduced with version 5.2.0. This document contained some defective hyperlinks.
5.2.1	PATCH: Internal update to current NWAF libraries (version 6.3.1): • PATCH: New default for property TextSource in TableCellDefinition objects is "", when property SymbolIDSource is set, else the default is "TableText" as before. This is for convenience. • PATCH: Animation on expanding/collapsing rows in loads view is enabled again. • PATCH: The vertical splitter can now be dragged to the left until the table has a width of 0 even if fixed columns exist. • PATCH: When dragging a bar to the border of time than the user cannot drag it out of sight anymore. PATCH: Rows were not sorted on screen after changing one of the settings Activity/Allocation/Entity/ResourceSortCodeSources.
5.2.0 **(See also release notes!)**	MINOR: Now it is possible to show allocations in own rows also in the resources view: • New setting PM_AllocationRowsVisibleInResourcesView. MINOR: Now allocation links do not require activity IDs anymore. MINOR: Now links can connect activities and allocations. MINOR: Symbols shown for properties ReleaseDate and DueDate on Activity objects are now specifiable and draggable: • New properties PM_ReleaseDateAllowedDragModes, PM_ReleaseDateSymbolHeight, PM_ReleaseDateSymbolID, and PM_ReleaseDateSymbolWidth for the release date. • New properties PM_DueDateAllowedDragModes, PM_DueDateSymbolHeight, PM_DueDateSymbolID, and PM_DueDateSymbolWidth for the due date. • New properties Start/EndPropertyName in EventArgs of events CanDrag, OnDragStart, OnDragEnd, OnDrop. MINOR: New properties Background/TextColor, on TableRowDefinition objects. MINOR: New argument CellIndex at events OnClicked and OnDoubleClicked. MINOR: New methods ScrollViewAreaHorizontally and ScrollViewAreaVertically. MINOR: New locales added for Japanese, Russian, Thai, and Chinese. MINOR: Clarification of options and properties concerning title, a.o. renaming Header to Title (compatibility is given):

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
	• New setting names are PM_TableTitleBackgroundColor, PM_TableTitleTextColor, PM_TableTitleColumnSeparatorColor, PM_TableTitleHighlightingColor, PM_EntitiesTableTitleBackgroundColor, PM_EntitiesTableTitleTextColor, PM_EntitiesTableTitleColumnSeparatorColor, PM_EntitiesTableTitleHighlightingColor. MINOR: Now additional dates on allocation and activities can be defined either as link source or link target: • New properties LinkSourceDate/LinkTargetDate on activity and allocation objects. • New values for property RelationType on link objects: SourceDateStart, SourceDateEnd, EndTargetDate, StartTargetDate, SourceDateTargetDate. MINOR: New property PM_TooltipTemplateID for PeriodHighlighterEntry objects. MINOR: Symbols at the left side of the table rows now are clickable. See event arguments SymbolIndex and SymbolID of event OnClicked. MINOR: New settings to specify default tooltip templates: PM_DefaultActivityBar/ RowTooltipTemplateID, PM_DefaultAllocationBar/RowTooltipTemplateID, PM_DefaultEntityRowTooltipTemplateID, PM_DefaultLinkTooltipTemplateID, PM_DefaultPeriodHighlighterEntryTooltipTemplateID, PM_DefaultResourceRow/ CurveTooltipTemplateID. MINOR: New setting FirstDayOfWeek. MINOR: Internal update to current NWAF libraries (version 6.3.0). PATCH: Fixed issue with tooltip template markup using keywords beginning with #, and concerning date formatting. PATCH: Fixed issue concerning eventually false week numbering in timescale.
5.0.1	PATCH: Internal update to current NWAF libraries (version 6.0.5). PATCH: Fixed issue when dragging an entity without a non-zero duration. PATCH: Fixed detecting the false PeriodHighlighterEntry for callbacks. PATCH: Fixed issue where two adjacent PeriodHighlighterEntries made tooltip disappear. PATCH: Fixed issue with update of allocation bars when changing the default calendar. PATCH: Fixed default value for property PM_BarOpacity of Allocation and Activity objects in .NET wrapper classes of C/AL edition (version of assemblies set to 5.0.1.0!)
5.0.0	MINOR: New method SaveAsPDF, new event OnSaveAsPDFFinished (not usable with IE11 and with Windows client). MINOR: Now allocation rows can be made visible in activities view by using the new option pm_allocationRowsVisibleInActivitiesView. Additionally other additions were made in this environment: • New settings PM_AllocationDefaultTableRowDefinitionID, PM_DefaultAllocationMinimumRowHeight, PM_DefaultAllocationRowSelectable, PM_DefaultActivityAllocationRowsCollapsible, PM_DefaultAllocationRowDesign,

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
	PM_DefaultAllocationAllowedBarDragModesInActivitiesView, PM_ForcedAllocationAllowedBarDragModesInActivitiesView. • New properties PM_AllocationRowsCollapsible, PM_AllocationRowsCollapseState, PM_HasAllocationRows on Activity objects. • New properties PM_MinimumRowHeight, PM_RowSelectable, PM_RowDesign, PM_AllowedBarDragModesInActivitiesView, TableText on Allocation objects. • New EventArgs property IsForAllocationRows in event OnCollapseStateChanged. • New enum value BarsInHiddenAllocationRows for RowDesigns. MINOR: Now links can be defined between allocations: • New properties Source/TargetAllocationID for links. • New settings PM_DefinedAllocationLinksVisibleInActivities/ResourcesView. MINOR: New sorting capabilities for table rows by new settings Activity/ Allocation/Entity/ResourceSortCodeSources. MINOR: New settings PM_WatermarkSymbolID and PM_WatermarkOpacity (only for AL version of control add-in, not usable with IE11 and with Windows client). MINOR: New properties PM_StatusFrameColor/Visible for activities and allocations. New settings PM_DefaultActivity/AllocationStatusFrameColor. MINOR: New property PM_BarOpacity for activities and allocations. MINOR: New properties Background/TextColor, Background/TextColorSource on TableCellDefinition objects. MINOR: New settings TableViewWidthInActivitiesView/ResourcesView/LoadsView and TableViewWidthsSynchronized (not in C/AL version!). MINOR: New settings PM_SymbolColumnTitleVisible/SymbolIDs as well as PM_EntitiesTableSymbolColumnTitleVisible/SymbolIDs. MINOR: New property TooltipText on PeriodHighlighterEntry objects. MINOR: Additional values for event OnVerticalScrollOffsetChanged. MINOR: Constraint dates are now also considered in the summary and in the diamond bar shape of allocations and activities. For the diamond shape, the PredictedEnd property is also taken into account. MINOR: Settings PM_Activity/ResourceTableRowDefinitionIDForTitle renamed to PM_TableRowDefinitionIDForTitleInActivitiesView/ResourcesView. Setting PM_EntityTableRowDefinitionIDForTitle renamed to PM_TableRowDefinitionIDForTitleInEntitiesTable. Old option names are deprecated but accepted for compatibility reasons. MINOR: New setting PM_TableRowDefinitionIDForTitleInLoadsView. MINOR: New setting TooltipDelay. MAJOR: Internal update to current NWAF libraries (version 6.0.1): • MAJOR: After dragging and dropping a draggable date line, the application now needs to update the date line object within the OnDrop event handler to apply the changed date. In older versions the date line incorrectly remained on the new date. • MAJOR: Activity rows and resource rows do not show the calendar of an

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
	ancestor anymore when the represented objects do not have an own calendar. • PATCH: Several fixes concerning tooltips and captions of PeriodHighlighterEntries.
4.0.2	PATCH: Internal update to current NWAF libraries (version 5.7.9). PATCH: Fixed and improved appearance of bars while dragging and of dragged entities in time area also especially when using the settings TimeStepUnit and TimeStepUnitFactor for a more intuitive user experience. PATCH: Fixed cursor issue on entities table, not being updated correctly when moving the mouse. PATCH: Fixed issue concerning snapping when dragging bars in collapsed rows and bars of collapsed sub rows are shown. PATCH: Fixed issue concerning snapping when dragging bars and start dates or end dates of other bars contain millisecond values. Now these start dates and end dates are rounded down or up, resp., to full seconds.
4.0.1	PATCH: Internal update to current NWAF libraries (version 5.7.6). PATCH: Collapse states (children and curves) are cached when interactively changing state. PATCH: Collapse states (children and curves) are now far more often cached (when setting ParentID, when updating objects of other views). PATCH: Behavior of dragged bars now improved. PATCH: Fixed false recognition of visual type of row objects in callback triggers. PATCH: Fixed crash when using world view together with bar symbols. PATCH: Zooming out of the currently visible time range by using the “up” button in the timescale resulted in an incorrect horizontal scroll offset. PATCH: Fixed issue when using the method updateDateLines (not all properties have been updated). PATCH: Fixed issue where the curves were not shown when adding a resource with property PM_CurveCollapseState set to 0. PATCH: Fixed issue with settings TimeStepUnit and TimeStepUnitFactor.
4.0.0	MAJOR: Additional parameters for method ScrollToObject. MAJOR: To be treated as a bug fix, the property DragMode in the event OnDrop now contains the actual drag mode of the interaction that took place and not all allowed drag modes on the object! MINOR: New settings MultipleBarDraggingEnabled, PM_ForcedActivityAllowedBarDragModes, PM_ForcedAllocationAllowedBarDragModes. New properties CoupledObjectIDs and StartsAndEndsOfCoupledObjects in event OnDrop. MINOR: New event CanDrag. MINOR: World view implemented. See settings WorldViewVisible, WorldViewPosition, WorldViewExtent.

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
	MINOR: Improved loading performance. MINOR: New settings LoggingEnabled and InteractiveActivationOfLoggingEnabled. MINOR: New properties HorizontalTitleAlignment and SymbolIDSource in TableCellDefinition object. MINOR: When dropping a date line interactively, the resulting date is rounded to the best possible date that is represented by the X coordinate the line phantom is shown on. MINOR: New options PM_ScrollToObjectAnimationEnabled, PM_ScrollToObjectHighlightFlashingEnabled, and PM_ScrollToObjectHighlightingColor. MINOR: New properties PM_BarTextPrefixSymbolID/Height/Width, PM_Left/RightBarSymbolID, PM_Left/RightBarSymbolWidth, PM_Left/RightBarSymbolHeight for Allocation and Activity objects. MINOR: Added values '1046' (pt-PT) and '2070' (pt-BR) for settings LocaleId/LanguageId. MINOR: New setting PM_MainViewAreaVisible. MINOR: New properties CustomText1/2/3 for objects of types Activity, Allocation, Entity, Link, Resource, mainly for C/AL users. MINOR: New setting PingTimeInMilliseconds and new event OnPing. MINOR: Internal update to current NWAF libraries (version 5.6.4): • PATCH: Many bug fixes.
3.0.5	MINOR: Added value '1045' (pl-PL) for settings LocaleId/LanguageId. PATCH: Internal update to current NWAF libraries (version 5.2.5). PATCH: Bug fix for exception when updating Allocation object with lower count of entries by more than one. PATCH: Added missing property NewDate in object type OnDropEventArgs for C/AL version and here in documentation of event OnDrop.
3.0.4	PATCH: Fixed non-integer values in some pixel values of event argument properties. PATCH: Fixed issue of false type of property PointInTime for object type DateLine in C/AL version. PATCH: Added missing property PM_TableRowDefinitionID for object types Activity and Resource. PATCH: Internal update to current NWAF libraries (version 5.2.4). PATCH: Fixed context menus on period highlighter presentation. See property ContextMenuID of object PeriodHighlighterEntry.
3.0.3	PATCH: Internal update to current NWAF libraries (version 5.2.2): • Bug fixed concerning using time zones together with DateLineGridMode.Auto and invisible nonworking time. • Bug fixed concerning visibility of date line for option CurrentDate.

© 2026 Boyum IT Solutions GmbH

Version	Description of changes
3.0.2	PATCH: Internal update to current NWAF libraries (version 5.2.1): • More performance when loading data. • Several bug fixes.
3.0.1	PATCH: Several fixes exclusively for C/AL version of control add-in.
3.0.0	MAJOR: New object types DateLine, PeriodHighlighter, TableRowDefinition, TooltipTemplate. MAJOR: New events OnClicked, OnTableCellDefinitionWidthChanged, OnTimeAreaViewParametersChanged, OnVerticalScrollOffsetsChanged MAJOR: Definition of property ContextMenu.Items has changed, new ContextMenuItem object. MAJOR: New methods Add/Update/Remove for new object types. MINOR: Internal update to current NWAF libraries (version 5.1.1). MINOR: New setting ForcedFillOfPageSpace for resize issue. MINOR: New setting PM_IgnoreCalendarOnAllocationBarInteractions. MINOR: Many new properties on exisiting object types. MINOR: Many new settings concerning default values. MINOR: Many new settings for additional functionality.
2.2	Internal update to current NWAF libraries (version 4.11.0) with many bug fixes.
2.1	MINOR: New method About. PATCH: Setting LocaleId '2057' (en-GB) now working.
2.0	MAJOR: The setting LicenseKey has to be used mandatory now. MAJOR: The event OnRequestSettings now has a parameter. MINOR: New setting LocaleId. MINOR: New method SetTimeResolutionForView. MINOR: New allocation properties Progress, PM_ProgressColor, and PM_ProgressNonworkingColor. MINOR: New setting PM_LinksVisibleInResourcesView.
1.0	Initial release

2 Introduction

The Visual Scheduling Add-in Developer Toolbox consists mainly of the source code for the Visual Scheduling Control Add-in (VS-CAI). How to compile the client control is explained in the installation guide. In the following the API and data model of the VS-CAI is defined.

© 2026 Boyum IT Solutions GmbH

3 System Requirements

3.1 Supported Browsers and Versions

Google Chrome (current version at delivery date of the control add-in)
Mozilla Firefox (current version at delivery date of the control add-in)
Apple Safari (current version at delivery date of the control add-in)
Microsoft Edge (current version at delivery date of the control add-in)

3.2 Supported BC Versions

BC (all versions that are supported by Microsoft at delivery date of the control add-in in referring to the supported target languages AL).

3.3 Included 3rd Party Libraries and Versions

The following open-source libraries are included automatically with the control add-in and are needed to work correctly. The libraries are provided for convenience by Boyum IT as-is and are not sold by Boyum IT. If you encounter any (security) issue with a library, you can exchange it by another version.

The license conditions of these libraries must be obeyed by the user of the control add-in when the control add-in is distributed within an application!

Library Name	Version	Comment
nwaf-apptools/ nwaf-table/ nwaf-gantt/ nwaf-rab	appropriate	Own libraries that are part of the control add-in and therefore are licensed together with it.
D3	7.9.0	License: ISC. URL: https://d3js.org/
d3-context-menu	2.1.0	License: MIT. URL: github.com/patorjk/d3-context-menu
Hammer.js	2.0.8	License: MIT. URL: https://hammerjs.github.io/
TinyColor	1.6.0	License: MIT. URL: https://bgrins.github.io/TinyColor/
Moment.js	2.30.1	License: MIT. URL: https://momentjs.com/
Moment-Timezone	0.6.2-2026b	License: MIT.

© 2026 Boyum IT Solutions GmbH

		URL: https://momentjs.com/timezone/
SVG-to-PDFKit	0.1.10	License: MIT. There is a bug in all versions up to 0.1.8 of SVG-to-PDFKit that lowers the performance from about 1 second per exported page to about 1 minute per page. We were able to find and fix this bug (see pull request here: https://github.com/alafr/SVG-to-PDFKit/pull/143). The patch is included in the master version of the source code on GitHub, but there is no release yet by the author on npmjs.com. Therefore, we deliver version 0.1.10 of a fork by the author Better-Boards including the patch for your convenience. URL: https://github.com/Better-Boards/SVG-to-PDFKit
PDFKit	0.18.0	License: MIT. URL: https://pdfkit.org/
blob-stream	0.1.3	License: MIT. URL: https://github.com/devongovett/blob-stream
html2canvas	1.4.1	License: MIT. URL: https://html2canvas.hertzen.com/

4 Data Model

The data model of the Visual Scheduling Control Add-in is designed for resource planning in general but is extended to cover presentations of activities view and resources view.

A note on the order in which you should add and remove objects to achieve a high-performance application:
The decisive factor here is the object type. The following sequence of types should be followed when adding objects:

1. Symbols / DateLines / TooltipTemplates / TableRowDefinitions (order within the group is not important)
2. Calendars / PeriodHighlighters / Curves (order within the group is not important)
3. Skills
4. Resources
5. Activities
6. Allocations
7. Links
8. Entities (can be added anywhere from 3 to 7)
9. HierarchySupplementaryDefinitions

For ContextMenu objects there is no recommended order within the other object types.

When updating objects, the same sequence is recommended, when many objects are updated. But that also depends on the counts of existing objects.

© 2026 Boyum IT Solutions GmbH

When removing objects, please proceed in reverse order.

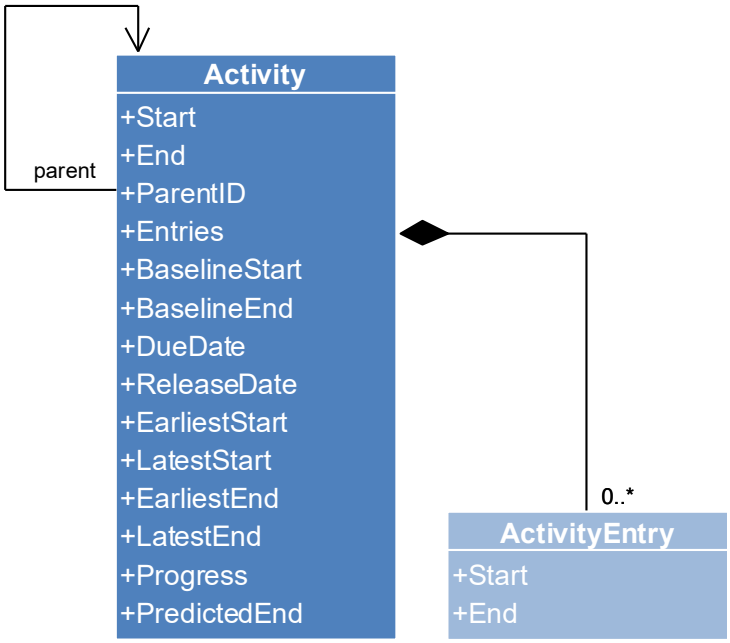
A note on the old "PM_" prefix of the object properties:
The old "PM_" prefix has been removed from the object properties for simplicity. Please change your code as soon as possible, since the support will end at some time.

A note on using CSS custom properties for coloring:
Each object property that controls the color of an element in VSW can be set a value of type string that represents a "CSS color value". This means that you have, for instance, the following options to specify a color value:

- a predefined color name that CSS supports, such as "red", "green", or "blue"
- a hexdecimal notation, such as "#FF0000", "#00FF00", or "#0000FF"
- an RGB or RGBA notation, such as "rgb(255, 0, 0)", "rgb(0, 255, 0)", or "rgba(0, 0, 255, 0.5)"
- an HSL and HSLA notation, such as "hsl(0, 100%, 50%)", " hsl(120, 100%, 50%)", or " hsla(240, 100%, 50%, 0.5)"
-

In addition, you can also use CSS Custom Properties. How to do that is described in a blog post.

4.1 Activities



4.1.1 Activity

An Activity object defines the properties of a single activity.

NOTE: On data objects that are updated missing properties normally are interpreted as being set to the default value!

© 2026 Boyum IT Solutions GmbH

Activity Property Name	Type	Description
AllocationRowsCollapseState	Integer (see enum CollapseState)	**Optional, default: -1** – Specifies whether the allocation rows of the activity should be expanded or collapsed when displayed. See also event OnCollapseStateChanged and setting AllocationRowsVisibleInActivitiesView. -1: no change 0: display activity row in an expanded way for allocation rows. 1: display activity row in a collapsed way for allocation rows.
AllocationRowsCollapsible	Boolean	**Optional, default: True** – If set to true, then the row representing this activity row will be interactively collapsible when allocation rows exist.
AllowedBarDragModes	Integer (see enum BarDragModes)	**Optional, default: -1 = value of setting DefaultActivityAllowedBarDragModes** – This option determines the allowed bar drag modes for this activity in the activities view. See also settings BarsDraggable and Editable.
AllowedRowDragModes	Integer (see enum RowDragModes)	**Optional, default: -1 = value of setting DefaultActivityAllowedRowDragModes** – This setting determines the allowed row drag modes for this activity. See also settings RowsDraggable and Editable.
AttachedDateLineIDs	JsonArray of Text	**Optional, default: value of setting DefaultActivityAttachedDateLineIDs** – This property determines the IDs of date lines to show when the user hovers the pointer of the bar. The date lines then can show data from this data object as the date, the caption, and several presentation attributes.
BarBottomOutsideFontFamily	Text (CSS font family value)	**Optional, default: '' (empty string) = value of setting FontFamily** – Font family for the bottom outside text. See property BarBottomOutsideText. The value does not influence the height of the bar! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-family.
BarBottomOutsideFontSize	Text (CSS font size value)	**Optional, default: '12px'** – Font size for the bottom outside text. See property BarBottomOutsideText. See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-size. **Hint**: Only pixel values allowed! The value influences the height of the bar!

© 2026 Boyum IT Solutions GmbH

Activity Property Name	Type	Description
		Note: If set, do not forget to append the unit 'px' to the numeric value! Otherwise, the control add-in will ignore the setting.
BarBottomOutsideFontStyle	Text (CSS font style value)	**Optional, default: '' (empty string) = value of setting FontStyle** – Font style for the bottom outside text. See property BarBottomOutsideText. The value does not influence the height of the bar! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-style.
BarBottomOutsideFontWeight	Text (CSS font weight value)	**Optional, default: '' (empty string) = value of setting FontWeight** – Font weight for the bottom outside text. See property BarBottomOutsideText. The value does not influence the height of the bar! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-weight.
BarBottomOutsideText	Text	**Optional, default: '' (empty string)** – If set, then the given text is shown below the bar. The text is not clipped at the end of the bar. Therefore, this property is not combinable with the flag BarsInHiddenDescendantRows of the property CollapsedRowDesign.
BarBottomOutsideTextColor	Text	**Optional, default: 'black'** – This sets the color for the text shown by the property BarBottomOutsideTextColor.
BarContextMenuID	Text	**Optional, default: '' (empty string) = value of property ContextMenuID** – Identifier of a ContextMenu object. The strings in it will be shown as a context menu on a bar.
BarDesign	Integer (see enum BarDesigns)	**Optional, default: -1 = value of setting DefaultActivityBarDesign** – This property determines the default design for activity bars including or excluding entries, complex shape, symbols, status, constraints, baseline, progress, and text.
BarFontFamily	Text (CSS font family value)	**Optional, default: '' (empty string) = value of setting FontFamily** – Font family for the text. See properties BarText and BarTextFormat. The value does not influence the height of the bar! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-family.
BarFontSize	Text (CSS font size value)	**Optional, default: '' (empty string) = value of setting FontSize** – Font size for the text. See properties BarText and BarTextFormat. The value does not influence the height of the bar!

© 2026 Boyum IT Solutions GmbH

Activity Property Name	Type	Description
		See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-size. **Note**: If set, do not forget to append a unit like 'px' to the numeric value! Otherwise, the browser will ignore the setting.
BarFontStyle	Text (CSS font style value)	**Optional, default: '' (empty string) = value of setting FontStyle** – Font style for the text. See properties BarText and BarTextFormat. The value does not influence the height of the bar! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-style
BarFontWeight	Text (CSS font weight value)	**Optional, default: '' (empty string) = value of setting FontWeight** – Font weight for the text. See properties BarText and BarTextFormat. The value does not influence the height of the bar! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-weight.
BarHeight	Integer (≥ 0, ≤ 1000)	**Optional, default: -1 = value of setting DefaultActivityBarHeight** – Height of the bars in pixels. This property is useful, when more than one line of text is shown inside (see property BarText). Proposal: For one line take 22, for two lines 38, for three lines 54, and so on. When no progress bar is needed, then you can subtract 4 from the value.
BarOpacity	Decimal (≥ 0.0, ≤ 1.0)	**Optional, default: 1.0** – Specifies the opacity of the entire activity bar (including the visualization of the progress bar, symbols, constraint dates, and baseline bar).
BarPatternColor	Text	**Optional, default: 'white'** – Color for the pattern when this is visible by using property BarPatternType.
BarPatternType	Integer (see enum PatternType)	**Optional, default: -1** – If set to a value >= 0, then a pattern is shown on top of the fill color and behind the text.
BarSelectable	Boolean	**Optional, default: -1 = value of setting DefaultActivityBarSelectable** – If set to True, then the bar representing this activity will be selectable.
BarShape	Integer (see enum BarShape)	**Optional, default: 0** – This option defines which shape is used for the visualization of bars.
BarShapeSymbolID	Text	**Optional, default: '' (empty string)** – If the bar shape named Symbol is used (see property BarShape), then the symbol defined here will be shown. The symbol will be resized to the height defined in property BarHeight and to the width defined in property BarShapeSymbolWidth.

© 2026 Boyum IT Solutions GmbH

Activity Property Name	Type	Description
BarShapeSymbolWidth	Integer > 0	**Optional, default: value of property BarHeight** – Width of the symbol defined in property BarShapeSymbolID when the bar shape named Symbol is used.
BarText	Text	**Optional, default: '' (empty string)** – Text to show on the bar when Regular, Summary or Rectangle bar shape is selected. See also properties BarFontFamily, BarFontSize, BarFontStyle, BarFontWeight. **Note**: Several immediately consecutive spaces are always combined into one space by the browsers. If the individual spaces are to be preserved, then each of them must be replaced by the Unicode character \u00A0.
BarTextColor	Text (CSS color value)	**Optional, default: 'white'** – Color for the texts of the bar, see property BarText.
BarTextFormat	string	**Optional, default: value of setting DefaultActivityBarTextFormat** – This property describes the format of the bar text. If not set, then the value of property BarText is displayed. This string contains the placeholders for object values surrounded by double curly braces {{ }}. For example, based on the following string a tooltip with a table containing three rows of key-value pairs is created, where the values are taken from the properties "name" and "firstName" of the referenced object: `{{name}}, {{firstName}}` As an escape, the use of three open curly braces {{{ are displayed as {{. Additionally, the property name can be extended to contain the desired property type as in {{Start:date}}. At the moment only the types 'date' and 'number' are possible besides 'string' (other property types are converted automatically with toString()). The type 'date' converts date values by default using the same format as other dates in the timescale and at the dragging date line captions. You can add another colon followed by a format name, that is defined by the settings IntlDateTimeFormatOptionsMap or IntlNumberFormatOptionsMap, resp. The referenced object is the object on which the tooltip will be shown. For period highlighter entries

© 2026 Boyum IT Solutions GmbH

Activity Property Name	Type	Description
		and allocation entries the referenced object is the main object and not the entry object. It is possible to access related objects by using the following keywords within the property accessor string: >Parent, >Calendar. It is also possible to access variables that are defined by the setting ApplicationVariablesMap by using ?*variableName*. If the value reached is an object, you can then access a property value by using a prefixed dot: .propertyName and you can use […] to access a property value, a map entry or an array entry. Within […] you can use a literal like 5 or A (with or without quotes) or even curly braces {{…}} with the same rules as above.
BarTextPrefixSymbolHeight	Integer	**Optional, default: 12** – Height of the bar symbol before the text (see property BarTextSymbolSymbolID) in pixels at a zoom factor of 100%. The height can be set bigger than the actual bar height and the symbol then will be shown above the bar shape.
BarTextPrefixSymbolID	Text	**Optional, default: undefined** – Identifier of the symbol to be shown before the text inside of the activity bar. The symbol will be shown vertically centered inside the bar.
BarTextPrefixSymbolWidth	Integer	**Optional, default: 12** – Width of the bar symbol before the text (see property BarTextPrefixSymbolID) in pixels at a zoom factor of 100%.
BarTextWrapMode	Integer (see enum TextWrapMode)	**Optional, default: TextWrapMode.None** – Specifies whether the text inside the bar is wrapped.
BarTooltipTemplateID	Text	**Optional, default: value of setting DefaultActivityBarTooltipTemplateID** – ID of a tooltip template. The template is used for the tooltip that appears on the activity bar.
BarTopOffset	number	**Optional, default: 0** – Offset of the bar in pixels relative to its upper side. A negative number will shift the bar upwards, a positive number will shift the bar downwards. It will be only considered in rows with multiple activity bars inside, e.g., in collapsed rows where the bars of the child rows are visible.

© 2026 Boyum IT Solutions GmbH

Activity Property Name	Type	Description
BarTopOutsideFontFamily	Text (CSS font family value)	**Optional, default: '' (empty string) = value of setting FontFamily** – Font family for the top outside text. See property BarTopOutsideText. The value does not influence the height of the bar! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-family.
BarTopOutsideFontSize	Text (CSS font size value)	**Optional, default: '12px'** – Font size for the top outside text. See property BarTopOutsideText. See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-size. **Hint**: Only pixel values allowed! The value influences the height of the bar! **Note**: If set, do not forget to append the unit 'px' to the numeric value! Otherwise, the control add-in will ignore the setting.
BarTopOutsideFontStyle	Text (CSS font style value)	**Optional, default: '' (empty string) = value of setting FontStyle** – Font style for the top outside text. See property BarTopOutsideText. The value does not influence the height of the bar! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-style
BarTopOutsideFontWeight	Text (CSS font weight value)	**Optional, default: '' (empty string) = value of setting FontWeight** – Font weight for the top outside text. See property BarTopOutsideText. The value does not influence the height of the bar! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-weight.
BarTopOutsideText	Text	**Optional, default: '' (empty string)** – If set, then the given text is shown above the bar. The text is not clipped at the end of the bar. Therefore, this property is not combinable with the flag BarsInHiddenDescendantRows of the property CollapsedRowDesign.
BarTopOutsideTextColor	Text	**Optional, default: 'black'** – This sets the color for the text shown by the property BarTopOutsideTextColor.
BaselineBorderColor	Text (CSS color value)	**Optional, default: '#808080'** – Color for the border of the baseline bar.
BaselineColor	Text (CSS color value)	**Optional, default: '#C8C8C8'** – Color for the working time periods of the baseline bar. The nonworking time periods of the bar will be colored with the same color as long as the property

© 2026 Boyum IT Solutions GmbH

Activity Property Name	Type	Description
		BaselineNonworkingTimeColor is undefined or set to 'calculated'.
BaselineDueDate	DateTime	**Optional, default: 0DT** – Baseline due date of the activity. As soon as a baseline due date is specified, the due date symbol appears automatically, but smaller. Another symbol can be selected via the BaselineDueDateSymbolID property. The center of the symbol is aligned with the baseline due date. See also setting ActivityBaselineBarsVisible.
BaselineDueDateColor	Text	**Optional, default: Value of property BaselineColor** – Color for the baseline due date symbol, if the symbol is the internal diamond.
BaselineDueDateSymbolID	Text	**Optional, default: Value of property DueDateSymbolID** – Identifier of the symbol to be shown at the baseline due date of the activity.
BaselineDueDateTooltipTemplateID	Text	**Optional, default: Value of property BarTooltipTemplateID** – Tooltip template for the baseline due date symbol.
BaselineEnd	DateTime	**Optional, default: 0DT** – Baseline end date of the activity. See also setting ActivityBaselineBarsVisible.
BaselineNonworkingTimeColor	Text (CSS color value or 'calculated')	**Optional, default: value of property BaselineColor** – Color for the nonworking time periods of the baseline bar. If set to 'calculated', a color will be calculated using the color defined by the BaselineColor property.
BaselineReleaseDate	DateTime	**Optional, default: 0DT** – Baseline release date of the activity. As soon as a baseline release date is specified, the release date symbol appears automatically, but smaller. Another symbol can be selected via the BaselineReleaseDateSymbolID property. The center of the symbol is aligned with the baseline release date. See also setting ActivityBaselineBarsVisible.
BaselineReleaseDateColor	Text	**Optional, default: Value of property BaselineColor** – Color for the baseline release date symbol, if the symbol is the internal diamond.
BaselineReleaseDateSymbolID	Text	**Optional, default: Value of property DueDateSymbolID** – Identifier of the symbol to be shown at the baseline release date of the activity.
BaselineReleaseDateTooltipTemplateID	Text	**Optional, default: Value of property BarTooltipTemplateID** – Tooltip template for the baseline release date symbol.
BaselineStart	DateTime	**Optional, default: 0DT** – Baseline start date of the activity. See also setting ActivityBaselineBarsVisible.

© 2026 Boyum IT Solutions GmbH

Activity Property Name	Type	Description
BaselineTooltipTemplateID	Text	**Optional, default: Value of property BarTooltipTemplateID** – Tooltip template for the baseline bar.
BorderColor	Text (CSS color value or 'calculated')	**Optional, default: 'gray'** – Color for the border of the bar. If set to 'calculated', a color will be calculated using the color defined by the Color property. This can be useful in situations where two bars are positioned next to each other, and a graphical indicator is needed to visually distinguish the two bars.
BorderDashArray	Text (SVG stroke dash array value)	**Optional, default: 'none'** – Pattern of dashes and gaps for drawing the border of the bar. For further information, please see https://www.w3.org/TR/SVG11/painting.html#StrokeDasharrayProperty or https://developer.mozilla.org/en-US/docs/Web/SVG/Attribute/stroke-dasharray. The value 'none' indicates that no dashing is used. In this case, the border is drawn solid.
CalendarGridColor	Text (CSS color value)	**Optional, default: value of setting CalendarGridColor** – Specifies a color used to color the vertical stripes representing the nonworking times for the activity object inside the diagram. If allocation rows are visible the color is used for these rows, too.
CalendarID	Text	**Optional, default: '' (empty string)** – Corresponding calendar. If '' (empty string), then the calendar specified by the option defaultCalendarID will be used.
CollapsedRowDesign	Integer (see enum RowDesigns)	**Optional, default: -1 = value in setting DefaultActivityCollapsedRowDesign** – Specifies how the time area is filled when the row is collapsed and visible. See explanation for possible values in enumerations chapter.
CollapseState	Integer	**Optional, default: -1** – Specifies whether the row of the activity should be expanded or collapsed. -1: do not change the way the activity row is displayed 0: display activity row in an expanded way 1: display activity row in a collapsed way See option CollapseState in the Enumerations chapter for details.
Color	Text (CSS color value)	**Optional, default: '#646464'** – Fallback color for the entries of the bar, see property Color of ActivityEntry objects. If no entries are defined, then color of the activity bar itself.

© 2026 Boyum IT Solutions GmbH

Activity Property Name	Type	Description
		The nonworking time periods of the bar will be colored with the same color if the property NonworkingTimeColor is '' (empty string).
ContextMenuID	Text	**Optional, default: '' (empty string)** – Identifier of a ContextMenu object. The strings in it will be shown as a context menu.
ContextMenuIDInBarsPane	Text	**Optional, default: value of property ContextMenuID** – Identifier of a ContextMenu object. The strings in it will be shown as a context menu within empty space in the bars pane of an activity row.
ContextMenuIDInCurvesPane	Text	**Optional, default: value of property ContextMenuID** – Identifier of a ContextMenu object. The strings in it will be shown as a context menu within the curves pane of an activity row.
CurveCollapseState	Integer (see enum CollapseState)	**Optional, default: -1** – Specifies whether the curves in a activity row should be expanded or collapsed when displayed (only applicable, when setting CurvePanesVisibleInActivitiesView is set). See also event OnCurveCollapseStateChanged. -1: no change 0: display curves 1: hide curves
CustomText1	Text	**Optional, default: '' (empty string)** – Freely usable property best usable in C/AL where it is not possible to use freely defined properties like with JsonObjects. The property name can then be used in TableCellDefinition.SymbolIDSource/TextSource and in TooltipTemplate objects. **Note**: Several immediately consecutive spaces are always combined into one space by the browsers. If the individual spaces are to be preserved, then each of them must be replaced by the Unicode character \u00A0.
CustomText2	Text	**Optional, default: '' (empty string)** – Freely usable property best usable in C/AL where it is not possible to use freely defined properties like with JsonObjects. The property name can then be used in TableCellDefinition.SymbolIDSource/TextSource and in TooltipTemplate objects. **Note**: Several immediately consecutive spaces are always combined into one space by the browsers. If the individual spaces are to be preserved, then each of them must be replaced by the Unicode character \u00A0.
CustomText3	Text	**Optional, default: '' (empty string)** – Freely usable property best usable in C/AL where it is not

© 2026 Boyum IT Solutions GmbH

Activity Property Name	Type	Description
		possible to use freely defined properties like with JsonObjects. The property name can then be used in TableCellDefinition.SymbolIDSource/TextSource and in TooltipTemplate objects. **Note**: Several immediately consecutive spaces are always combined into one space by the browsers. If the individual spaces are to be preserved, then each of them must be replaced by the Unicode character \u00A0.
DueDate	DateTime	**Optional, default: 0DT** – Due date of the activity. See also setting ReleaseDueDateConnectionsVisible, if you want the widget to draw a connection line between a due date and a release date.
DueDateAllowedDragModes	Integer (see enum BarDragModes)	**Optional, default: None** – Determines the allowed drag mode for the due date of this activity in the activities view (these can be overwritten using the callback canDrag). In this context only None and DragHorizontally are used.
DueDateColor	Text (CSS color value)	**Optional, default: 'black'** – Color for the due date symbol.
DueDateSymbolHeight	Integer	**Optional, default: 12** – Height of the due date symbol (see property DueDateSymbolID) in pixels at a zoom factor of 100%. Currently the default symbol cannot be sized.
DueDateSymbolID	Text	**Optional, default: internal diamond symbol** – Identifier of the symbol to be shown at the due date of the activity. See also DueDateSymbolHeight, and DueDateSymbolWidth.
DueDateSymbolWidth	Integer	**Optional, default: 12** – Width of the due date symbol (see property DueDateSymbolID) in pixels at a zoom factor of 100%. Currently the default symbol cannot be sized.
DueDateTooltipTemplateID	Text	**Optional, default: Value of property BarTooltipTemplateID** – Tooltip template for the due date symbol.
EarliestDragStart	DateTime	**Optional, default: 0DT** – If set, then the time before the given date is grayed, when beginning to drag the activity bar. If the setting DragDatesLimitingInteraction is set to True, then the bar itself cannot be dragged before the date.
EarliestEnd	DateTime	**Optional, default: 0DT** – If defined, an additional symbol will be displayed to indicate this date. Please be aware to increase the top row margin so that the symbol has enough space, see setting TopRowMarginInTimeArea.

© 2026 Boyum IT Solutions GmbH

Activity Property Name	Type	Description
EarliestEndColor	Text (CSS color value)	**Optional, default: value in setting DefaultActivityConstraintSymbolColor** – Color for the EarliestEnd constraint symbol.
EarliestEndTooltipTemplateID	Text	**Optional, default: Value of property BarTooltipTemplateID** – Tooltip template for the earliest-end constraint flag.
EarliestStart	DateTime	**Optional, default: 0DT** – If defined, an additional symbol will be displayed to indicate this date. Please be aware to increase the top row margin so that the symbol has enough space, see setting TopRowMarginInTimeArea.
EarliestStartColor	Text (CSS color value)	**Optional, default: value in setting DefaultActivityConstraintSymbolColor** – Color for the EarliestStart constraint symbol.
EarliestStartTooltipTemplateID	Text	**Optional, default: Value of property BarTooltipTemplateID** – Tooltip template for the earliest-start constraint flag.
End	DateTime	**Optional, default:** Maximum value in the End properties of contained ActivityEntry objects is calculated and used. – End date of the activity. If not defined explicitly, the latest value in the End properties of contained ActivityEntry objects is calculated and used. If no end date can be determined at all, the activity will not become visible. The end date itself is not included in the time interval between start and end. This helps to calculate a duration and helps to show bars in a line, when an end date of one bar has the same value as the start date of a next bar. The same behavior also is true for all end dates used in the widget like e.g. in calendar entries, period highlighter entries, activity entries, allocations, allocation entries.
Entries	ActivityEntry []	**Optional, default: undefined** – array of activity entries. If used, then the entries will be shown as colored rectangles within the bar representation of the activity. Additionally, the property BarShape must be set to Regular or Rectangle.
ExpandedRowDesign	Integer (see enum RowDesigns)	**Optional, default: -1 = value in setting DefaultActivityExpandedRowDesign** – Specifies how the time area is filled when the row is expanded and visible. See explanation for possible values in enumerations chapter.

© 2026 Boyum IT Solutions GmbH

Activity Property Name	Type	Description
HasAllocationRows	Boolean	**Optional, default: False** – If set to True, then the row representing this activity will be collapsible/expandable for allocation rows even when no allocations exist referencing this activity. This serves for lazy loading.
HasChildren	Boolean	**Optional, default: False** – If set to True, then the row representing this activity will be collapsible/expandable even when there are no children defined. This serves for lazy loading.
ID	Text	**Required** – Identifier of the activity.
LatestDragEnd	DateTime	**Optional, default: 0DT** – If set, then the time after the given date is grayed, when beginning to drag the activity bar. If the setting DragDatesLimitingInteraction is set to True, then the bar itself cannot be dragged after the date.
LatestEnd	DateTime	**Optional, default: 0DT** – If defined, an additional symbol will be displayed to indicate this date. Please be aware to increase the top row margin so that the symbol has enough space, see setting TopRowMarginInTimeArea.
LatestEndColor	Text (CSS color value)	**Optional, default: value in setting DefaultActivityConstraintSymbolColor** – Color for the LatestEnd constraint symbol.
LatestEndTooltipTemplateID	Text	**Optional, default: Value of property BarTooltipTemplateID** – Tooltip template for the latest-end constraint flag.
LatestStart	DateTime	**Optional, default: 0DT** – If defined, an additional symbol will be displayed to indicate this date. Please be aware to increase the top row margin so that the symbol has enough space, see setting TopRowMarginInTimeArea.
LatestStartTooltipTemplateID	Text	**Optional, default: Value of property BarTooltipTemplateID** – Tooltip template for the latest-start constraint flag.
LatestStartColor	Text (CSS color value)	**Optional, default: value in setting DefaultActivityConstraintSymbolColor** – Color for the LatestStart constraint symbol.
LeftBarSymbolHeight	number	**Optional, default: 12** – Height of the left bar symbol (see property LeftBarSymbolID) in pixels at a zoom factor of 100%. The height can be set bigger than the actual bar height and the symbol then will be shown above the bar shape.
LeftBarSymbolID	string	**Optional, default: undefined** – Identifier of the symbol to be shown at the left side of the activity bar. The symbol will be shown vertically centered inside the bar. It is drawn over any existing bar text. See also RightBarSymbolID, LeftBarSymbolHeight, and LeftBarSymbolWidth.

© 2026 Boyum IT Solutions GmbH

Activity Property Name	Type	Description
LeftBarSymbolWidth	number	**Optional, default: 12** – Width of the left bar symbol (see property LeftBarSymbolID) in pixels at a zoom factor of 100%.
LinkSourceDate	DateTime	**Optional, default: 0DT** – Additional date serving as an additional "start point" to connect a link. See also property "RelationType" of link object.
LinkTargetDate	DateTime	**Optional, default: 0DT** – Additional date serving as an additional "end point" to connect a link. See also property "RelationType" of link object.
MinimumRowHeight	Integer	**Optional, default: -1 = value in setting DefaultMinimumActivityRowHeight** – Minimum height of the activity row in pixels. This property is useful, when more than one line of text is shown inside the table cells. Proposal: For one line take 36*, for two lines 52, for three lines 68, and so on. In order to have the same height also, when no bar is placed in the row, take the maximum of the height of all bars (default bar height is 22) plus 20, so e.g. 42. For using word wrapping in table cells, it is necessary to use a table row definition by setting the property TableRowDefinitionID and setting the property WrapMode in a contained table cell definition. In general, the height of a row is determined by several facts: The height of bars and the stacking of bars determine a height, that then is overlaid by the value of this property. The height of any text inside a table column or a bar is not considered, even when using wrapping.
MustEndOn	DateTime	**Optional, default: 0DT** – If defined, an additional symbol will be displayed to indicate this date. Please be aware to increase the top row margin so that the symbol has enough space, see setting TopRowMarginInTimeArea.
MustEndOnColor	Text (CSS color value)	**Optional, default: value in setting DefaultActivityConstraintSymbolColor** – Color for the MustEndOn constraint symbol.
MustEndOnTooltipTemplateID	Text	**Optional, default: Value of property BarTooltipTemplateID** – Tooltip template for the must-end-on constraint flag.
MustStartOn	DateTime	**Optional, default: 0DT** – If defined, an additional symbol will be displayed to indicate this date. Please be aware to increase the top row margin so that the symbol has enough space, see setting TopRowMarginInTimeArea.

© 2026 Boyum IT Solutions GmbH

Activity Property Name	Type	Description
MustStartOnColor	Text (CSS color value)	**Optional, default: value in setting DefaultActivityConstraintSymbolColor** – Color for the MustStartOn constraint symbol.
MustStartOnTooltipTemplateID	Text	**Optional, default: Value of property BarTooltipTemplateID** – Tooltip template for the must-start-on constraint flag.
NonworkingTimeColor	Text (CSS color value or 'calculated')	**Optional, default: default: value of property Color** – Color for the nonworking time periods of the bar. If set to 'calculated', a color will be calculated using the color defined by the Color property.
ParentID	Text	**Optional, default: '' (empty string)** – Identifier of the parent of the activity. This serves for setting up a hierarchy of activities. If this property is '' (empty string) the current activity will be considered as a root node of the activity hierarchy.
PeriodHighlighterID	Text	**Optional, default: '' (empty string)** – Reference to a period highlighter object that contains colored time periods. This can be used to show shifts or exceptions to the calendar (see property CalendarID) that defines work and non-work times.
PredictedEnd	DateTime	**Optional, default: 0DT** – A date that indicates the predicted end of the activity. This date is used to display a bar between this date and the end of the activity.
PredictedEndColor	Text (CSS color value)	**Optional, default: '"#646464'** – Color for the predicted end bar.
Progress	Decimal (in percent; ≥ 0, ≤ 100)	**Optional, default: 0** – Used to display a completion layer.
ProgressBackgroundColor	Text (CSS color value)	**Optional, default: value in setting DefaultActivityProgressBackgroundColor** – Color for the background of the progress bar region.
ProgressColor	Text (CSS color value)	**Optional, default: '#646464'** – Color for the working time periods of the progress bar. The nonworking time periods of the bar will be colored with the same color as long as the property ProgressNonworkingTimeColor is undefined or set to 'calculated'.
ProgressNonworkingTimeColor	Text (CSS color value or 'calculated')	**Optional, default: value of property ProgressColor** – Color for the nonworking time periods of the progress bar. If set to 'calculated', a color will be calculated using the color defined by the ProgressColor property.
ReleaseDate	DateTime	**Optional, default: 0DT** – Release date of the activity.

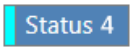
© 2026 Boyum IT Solutions GmbH

Activity Property Name	Type	Description
		See also setting releaseDueDateConnectionsVisible, if you want the widget to draw a connection line between a due date and a release date.
ReleaseDateAllowedDragModes	Integer (see enum BarDragModes)	**Optional, default: None** – Determines the allowed drag mode for the release date of this activity in the activities view (these can be overwritten using the callback canDrag). In this context only None and DragHorizontally are used.
ReleaseDateColor	Text (CSS color value)	**Optional, default: 'black'** – Color for the release date symbol.
ReleaseDateSymbolHeight	Integer	**Optional, default: 12** – Height of the due date symbol (see property ReleaseDateSymbolID) in pixels at a zoom factor of 100%. Currently the default symbol cannot be sized.
ReleaseDateSymbolID	Text	**Optional, default: internal diamond symbol** – Identifier of the symbol to be shown at the due date of the activity. See also ReleaseDateSymbolHeight, and ReleaseDateSymbolWidth.
ReleaseDateSymbolWidth	Integer	**Optional, default: 12** – Width of the release date symbol (see property ReleaseDateSymbolID) in pixels at a zoom factor of 100%. Currently the default symbol cannot be sized.
ReleaseDateTooltipTemplateID	Text	**Optional, default: Value of property BarTooltipTemplateID** – Tooltip template for the release date symbol.
RightBarSymbolHeight	Integer	**Optional, default: 12** – Height of the right bar symbol (see property RightBarSymbolID) in pixels at a zoom factor of 100%. The height can be set bigger than the actual bar height and the symbol then will be shown above the bar shape.
RightBarSymbolID	Text	**Optional, default: undefined** – Identifier of the symbol to be shown at the right side of the activity bar. The symbol will be shown vertically centered inside the bar. It is drawn over any existing bar text. See also LeftBarSymbolID, RightBarSymbolHeight, and RightBarSymbolWidth.
RightBarSymbolWidth	Integer	**Optional, default: 12** – Width of the right bar symbol (see property RightBarSymbolID) in pixels at a zoom factor of 100%.
RowCollapsible	Boolean	**Optional, default: value of setting defaultActivityRowCollapsible** – If set to True, then the row representing this activity will be interactively collapsible when children exist.
RowSelectable	Boolean	**Optional, default: value of setting defaultActivityRowSelectable** – If set to True, then the row representing this activity will be selectable.
RowSymbolColumn-BackgroundColor	Text (CSS color value)	**Optional, default: value of property SymbolColumnBackgroundColor of assigned table row definition or setting**

© 2026 Boyum IT Solutions GmbH

Activity Property Name	Type	Description
		SymbolColumnBackgroundColor – Determines the color of the symbol column within this table row.
RowSymbolIDs	JsonArray of Text	**Optional, default: empty array** – Array of identifiers of the symbols to be shown in the table symbol cell of the beginning of the table row. The symbols will be arranged one below the other. However, if the cell is not high enough to hold all symbols, then the remaining symbols are also arranged side-by-side. If this still does not fit, an additional 'show more" symbol will be displayed. An empty string ('") will cause an 'empty" symbol to be displayed. By this placeholder, you can reserve space for a symbol that may be shown at a later time. Please note: Each symbol will be resized to an image with a width and height of 16 pixels each at a zoom level of 100%.
RowTooltipTemplateID	Text	**Optional, default: value of setting DefaultActivityRowTooltipTemplateID** – ID of a tooltip template. The template is used for the tooltip that appears on the activity table row.
SnapTargetsForEnd	Integer (see enum SnapTargets)	**Optional, default: value of setting DefaultActivitySnap-TargetsForEnd** – When dragging horizontaly, then the visible end date of this allocation will optionally be snapping to date lines and calendar grids. The user can override an active snapping by pressing the ALT key while dragging.
SnapTargetsForStart	Integer (see enum SnapTargets)	**Optional, default: value of setting DefaultActivitySnap-TargetsForStart** – When dragging horizontally, then the visible start date of this activity will optionally be snapping to date lines and calendar grids. The user can override an active snapping by pressing the ALT key while dragging.
Start	DateTime	**Optional, default: Minimum value in the Start properties of contained ActivityEntry objects is calculated and used.** – Start date of the activity. If not defined explicitly, then the earliest value in the Start properties of contained ActivityEntry objects is calculated and used. If no start date can be determined at all, the activity will not become visible.
Status1Color	Text (CSS color value)	**Optional, default: '' (empty string)** – Color for the status symbol to the right of the bar. If undefined,

© 2026 Boyum IT Solutions GmbH

Activity Property Name	Type	Description
		no symbol appears. Only visible, when property Status1Visible is True.
Status1Visible	Boolean	**Optional, default: False** – If set to True and the corresponding status color is set in property Status1Color, then a predefined symbol is displayed to the right of the bar. 
Status2Color	Text (CSS color value)	**Optional, default: '' (empty string)** – Color for the status symbol to the right of the bar. If undefined, no symbol appears. Only visible, when property Status2Visible is True.
Status2Visible	Boolean	**Optional, default: False** – If set to True and the corresponding status color is set in property Status2Color, then a predefined symbol is displayed to the right of the bar. 
Status3Color	Text (CSS color value)	**Optional, default: '' (empty string)** – Color for the status symbol to the right of the bar. If undefined, no symbol appears. Only visible, when property Status3Visible is True.
Status3Visible	Boolean	**Optional, default: False** – If set to True and the corresponding status color is set in property Status3Color, then a predefined symbol is displayed to the right of the bar. 
Status4Color	Text (CSS color value)	**Optional, default: '' (empty string)** – Color for the status symbol to the left of the bar. If undefined, no symbol appears. Only visible, when property Status4Visible is True.
Status4Visible	Boolean	**Optional, default: False** – If set to True and the corresponding status color is set in property Status4Color, then a predefined symbol is displayed to the left of the bar. **Note: This property may be used with rectangle bar shapes only!** 
StatusFrameColor	Text (CSS color value)	**Optional, default: value of setting DefaultActivityStatusFrameColor** – Color for the status frame that will be shown when property StatusFrameVisible is set.
StatusFrameVisible	Boolean	**Optional, default: False** – If set to true, then a frame is shown around the bar. See also property StatusFrameColor.
TableColor	Text (CSS color value)	**Optional, default: level-dependent gray** – Color for the table row.

© 2026 Boyum IT Solutions GmbH

Activity Property Name	Type	Description
		If '' (empty string), a predefined level-dependent gray value is used.
TableColorVisibleInTimeArea	Boolean	**Optional, default: False** – If set to true, the time area row will be colored using the color defined by the TableColor property.
TableFontFamily	Text (CSS font family value)	**Optional, default: '' (empty string) = value of setting FontFamily** – Font family for all texts in the table cells. See property TableText and properties TextSource and TextFormat of TableCellDefinition object. The value does not influence the height of the row! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-family.
TableFontSize	Text (CSS font size value)	**Optional, default: '' (empty string) = value of setting FontSize** – Font size for all texts in the table cells. See property TableText and properties TextSource and TextFormat of TableCellDefinition object. The value does not influence the height of the row! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-size. **Note**: If set, do not forget to append a unit like 'px' to the numeric value! Otherwise, the browser will ignore the setting.
TableFontStyle	Text (CSS font style value)	**Optional, default: '' (empty string) = value of setting FontStyle** – Font style for all texts in the table cells. See property TableText and properties TextSource and TextFormat of TableCellDefinition object. The value does not influence the height of the row! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-style
TableFontWeight	Text (CSS font weight value)	**Optional, default: '' (empty string) = value of setting FontWeight** – Font weight for all texts in the table cells. See property TableText and properties TextSource and TextFormat of TableCellDefinition object. The value does not influence the height of the row! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-weight.
TableRowDefinitionID	Text	**Optional, default: '' (empty string) = value of setting DefaultEntityTableRowDefinitionID** – Identifier of a TableRowDefinition object that defines the composition of the table row.

© 2026 Boyum IT Solutions GmbH

Activity Property Name	Type	Description
TableSortCode	Text	**Optional, default: '' (empty string)** – If given, then the neighboring objects in the table in same level are sorted ascending using this value.
TableText	Text	**Optional, default: '' (empty string)** – Text to show in the table. **Note**: Several immediately consecutive spaces are always combined into one space by the browsers. If the individual spaces are to be preserved, then each of them must be replaced by the Unicode character \u00A0.
TableTextColor	Text (CSS color value)	**Optional, default: 'black'** – Color for the table row texts. See property TableText and properties TextSource and TextFormat of TableCellDefinition object.
TextColor		**Deprecated! Renamed to BarTextColor.**
TooltipText	Text	**Optional, default: '' (empty string)** – Text to show in the tooltip (formatted in HTML). If the text is empty, then no tooltip is shown. **Note**: Several immediately consecutive spaces are always combined into one space by the browsers. If the individual spaces are to be preserved, then each of them must be replaced by the Unicode character \u00A0.
TopLeftBarSymbolHeight	Integer	**Optional, default: 12** – Height of the top left bar symbol in pixels at a zoom factor of 100%.
TopLeftBarSymbolID	Text	**Optional, default: '' (empty string)** – Identifier of the symbol to be shown at the top left side of the activity bar. Please note: It protrudes 5 pixels vertically into the bar shape. For height and width see properties TopLeftBarSymbolHeight and TopLeftBarSymbolWidth.
TopLeftBarSymbolWidth	Integer	**Optional, default: 12** – Width of the top left bar symbol in pixels at a zoom factor of 100%.
TopLeftBarSymbolYOffset	Integer	**Optional, default: 0** – Vertical offset of the top left bar symbol in pixels relative to the default position (2/3 of the symbol height above the bar).
TopRightBarSymbolHeight	Integer	**Optional, default: 12** – Height of the top right bar symbol in pixels at a zoom factor of 100%.
TopRightBarSymbolID	Text	**Optional, default: '' (empty string)** – Identifier of the symbol to be shown at the top right side of the activity bar. Please note: It protrudes 5 pixels vertically into the bar shape. For height and width see properties TopRightBarSymbolHeight and TopRightBarSymbolWidth.
TopRightBarSymbolWidth	Integer	**Optional, default: 12** – Width of the top right bar symbol in pixels at a zoom factor of 100%.

© 2026 Boyum IT Solutions GmbH

Activity Property Name	Type	Description
TopRightBarSymbolYOffset	Integer	**Optional, default: 0** – Vertical offset of the top right bar symbol in pixels relative to the default position (2/3 of the symbol height above the bar).
ViewArea	Integer (see enum ViewArea)	**Optional, default: Default** – If set to Top, then the resource and its children are shown in a separate top view area in the activities view. Only settable on activities with no ParentID set. See settings TopViewAreaVisibleInActivitiesView and MainViewAreaVisibleInActivitiesView.
Visible	Boolean	**Optional, default: True –** If set to False, then this object is hidden in the chart.

4.1.2 ActivityEntry

Objects of this type are only used within the array of the Entries property of Activity objects.

NOTE: On data objects that are updated missing properties normally are interpreted as being set to the default value!

ActivityEntry Property Name	Type	Description
Color	Text (CSS color value)	**Optional, default: value of the corresponding property of the activity** – Color for the working time periods of the bar. The nonworking time periods of the bar will be colored with the same color as long as the property NonworkingTimeColor of the appropriate activity is undefined.
End	DateTime	**Optional, default: 0DT** – End date of the activity entry. The end date itself is not included in the time interval between start and end. This helps to calculate a duration and helps to show bars in a line, when an end date of one bar has the same value as the start date of a next bar. The same behavior also is true for all end dates used in the widget like e.g. in calendar entries, period highlighter entries, activities, allocations, allocation entries.
FontFamily	Text (CSS font family value)	**Optional, default: value of property BarFontFamily of activity** – Font family for the text of the entry. See property Text. See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-family.
FontSize	Text (CSS font size value)	**Optional, default: value of property BarFontSize of activity** – Font size for the text of the entry. See property Text. See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-size.

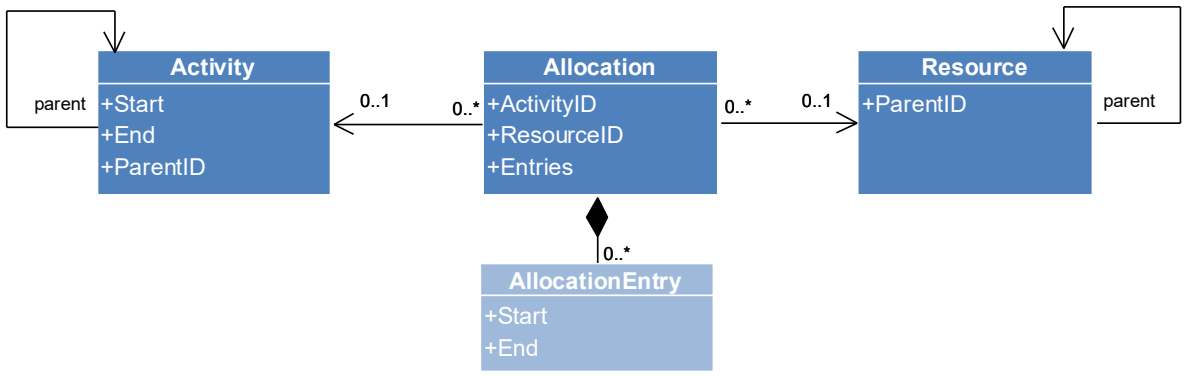
© 2026 Boyum IT Solutions GmbH

ActivityEntry Property Name	Type	Description
		Note: If set, do not forget to append a unit like 'px' to the numeric value! Otherwise, the browser will ignore the setting.
FontStyle	Text (CSS font style value)	**Optional, default: value of property BarFontStyle of activity** – Font style for the text of the entry. See property Text. See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-style
FontWeight	Text (CSS font weight value)	**Optional, default: value of property BarFontWeight of activity** – Font weight for the text of the entry. See property Text. See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-weight.
Height	Integer (≥ 0, ≤ 1000)	**Optional, default: -1 = value in setting DefaultActivityBarHeight** – Height of the entry in pixels.
NonworkingTimeColor	Text (CSS color value or 'calculated')	**Optional, default: value of property NonworkingTimeColor of activity** – Color for the nonworking time periods of the bar. If set to "calculated", a color will be calculated using the color defined by the Color property.
PatternColor	Text	**Optional, default: 'white'** – Color for the pattern when this is visible by using property PatternType.
PatternType	Integer (see enum PatternType)	**Optional, default: -1** – If set to a value >= 0, then a pattern is shown on top of the fill color and behind the text.
RelativeTopOffset	Integer	**Optional, default: 0** – Offset of the entry in pixels relative to the upper side of the corresponding activity bar. A negative number will shift the entry upwards, a positive number will shift the entry downwards.
Start	DateTime	**Optional, default: 0DT** – Start date of the activity entry.
Text	Text	**Optional, default: '' (empty string)** – Text to be displayed in the bar when Regular, Summary or Rectangle bar shape is selected. **Note**: Several immediately consecutive spaces are always combined into one space by the browsers. If the individual spaces are to be preserved, then each of them must be replaced by the Unicode character \u00A0.
TextColor	Text (CSS color value)	**Optional, default: value of the property BarTextColor of the activity** – Color for the text of the entry.

© 2026 Boyum IT Solutions GmbH

ActivityEntry Property Name	Type	Description
TextWrapMode	Integer (see enum TextWrapMode)	**Optional, default: 0 (None)** – Specifies whether the text inside the entry is wrapped. Since wrapping text is performance consuming, the application should use this feature only when really needed.

4.2 Allocations



4.2.1 Allocation

An Allocation object defines an allocation of one activity to one resource.

NOTE: On data objects that are updated missing properties normally are interpreted as being set to the default value!

Allocation Property Name	Type	Description
ActivityID	Text	**Optional, default: '' (empty string)** – Identifier of an Activity
AllowedBarDragModes	Integer (see enum BarDragModes)	**Optional, default: -1 = value of setting DefaultAllocationAllowedBarDragModes** – This option determines the allowed bar drag modes for this allocation in the resources view. See also settings BarsDraggable and Editable.
AllowedBarDragModes-InActivitiesView	Integer (see enum BarDragModes)	**Optional, default: -1 = value of setting DefaultAllocationAllowedBarDragModes InActivitiesView** – This option determines the allowed bar drag modes for this allocation in the resources view. See also settings BarsDraggable and Editable.
AllowedRowDragModes	Integer (see enum RowDragModes)	**Optional, default: value of setting DefaultAllocationAllowedRowDragModes** – This option determines the allowed row drag modes for this allocation in resources view (these can be overwritten using the setting ForcedAllocationAllowedRowDragModes)

© 2026 Boyum IT Solutions GmbH

Allocation Property Name	Type	Description
		. See also settings RowsDraggable and Editable.
AllowedRowDragModesInActivitiesView	Integer (see enum RowDragModes)	**Optional, default: value of setting DefaultAllocationAllowedRowDragModesInActivitiesView** – This option determines the allowed row drag modes for this allocation in activities view (these can be overwritten using the setting ForcedAllocationAllowedRowDragModesInActivitiesView). See also settings RowsDraggable and Editable.
AttachedDateLineIDs	JsonArray of Text	**Optional, default: value of setting DefaultActivityAttachedDateLineIDs** – This property determines the IDs of date lines to show when the user hovers the pointer of the bar. The date lines then can show data from this data object as the date, the caption, and several presentation attributes.
BarBottomOutsideFontFamily	Text (CSS font family value)	**Optional, default: '' (empty string) = value of setting FontFamily** – Font family for the bottom outside text. See property BarBottomOutsideText. The value does not influence the height of the bar! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-family.
BarBottomOutsideFontSize	Text (CSS font size value)	**Optional, default: '12px'** – Font size for the bottom outside text. See property BarBottomOutsideText. See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-size. **Hint**: Only pixel values allowed! The value influences the height of the bar! **Note**: If set, do not forget to append the unit 'px' to the numeric value! Otherwise, the control add-in will ignore the setting.
BarBottomOutsideFontStyle	Text (CSS font style value)	**Optional, default: '' (empty string) = value of setting FontStyle** – Font style for the bottom outside text. See property BarBottomOutsideText. The value does not influence the height of the bar! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-style

© 2026 Boyum IT Solutions GmbH

Allocation Property Name	Type	Description
BarBottomOutsideFontWeight	Text (CSS font weight value)	**Optional, default: '' (empty string) = value of settting FontWeight** – Font weight for the bottom outside text. See property BarBottomOutsideText. The value does not influence the height of the bar! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-weight.
BarBottomOutsideText	Text	**Optional, default: '' (empty string)** – If set, then the given text is shown below the bar. The text is not clipped at the end of the bar but is clipped when another bar starts in the same vertical position.
BarBottomOutsideTextColor	Text	**Optional, default: 'black'** – This sets the color for the text shown by the property BarBottomOutsideTextColor.
BarContextMenuID	Text	**Optional, default: '' (empty string) = value of property ContextMenuID** – Identifier of a ContextMenu object. The strings in it will be shown as a context menu on a bar.
BarDesign	Integer (see enum BarDesigns)	**Optional, default: -1 = value of setting DefaultAllocationBarDesign** – This property determines the default design for allocation bars including or excluding entries, complex shape, symbols, status, constraints, progress, and text.
BarFontFamily	Text (CSS font family value)	**Optional, default: '' (empty string) = value of setting FontFamily** – Font family for the text. See property BarText. The value does not influence the height of the bar! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-family.
BarFontSize	Text (CSS font size value)	**Optional, default: '' (empty string) = value of setting FontSize** – Font size for the text. See property BarText. The value does not influence the height of the bar! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-size. **Note**: If set, do not forget to append a unit like 'px' to the numeric value! Otherwise, the browser will ignore the setting.

© 2026 Boyum IT Solutions GmbH

Allocation Property Name	Type	Description
BarFontStyle	Text (CSS font style value)	**Optional, default: '' (empty string) = value of setting FontStyle** – Font style for the text. See property BarText. The value does not influence the height of the bar! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-style
BarFontWeight	Text (CSS font weight value)	**Optional, default: '' (empty string) = value of setting FontWeight** – Font weight for the text. See property BarText. The value does not influence the height of the bar! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-weight.
BarHeight	Integer (≥ 0, ≤ 1000)	**Optional, default: -1 = value in setting DefaultAllocationBarHeight** – Height of the bar in pixels. This is useful, when more than one line of text is shown inside (see property BarText). Proposal: For one line take 22, for two lines 38, for three lines 54, and so on. When no progress bar is needed, then you can subtract 4 from the value.
BarOpacity	Decimal (≥ 0.0, ≤ 1.0)	**Optional, default: 1.0** – Specifies the opacity of the entire activity bar (including the visualization of the progress bar, symbols, constraint dates, and baseline bar).
BarPatternColor	Text (CSS color value)	**Optional, default: 'white'** – Color for the pattern when this is visible by using property BarPatternType.
BarPatternType	Integer (see enum PatternType)	**Optional, default: None** – If set, then a pattern is shown on top of the fill color and behind the text.
BarSelectable	Boolean	**Optional, default: value of setting DefaultAllocationBarSelectable** – If set to True, then the bar representing this allocation will be selectable.
BarShape	Integer (see enum BarShape)	**Optional, default: -1 = value in setting DefaultAllocationBarShape** – This option defines which shape should be used by default for the visualization allocation bars.
BarShapeSymbolID	Text	**Optional, default: '' (empty string)** – If the bar shape named Symbol is used (see property BarShape), then the symbol defined here will be shown. The symbol will be resized to the height defined in

© 2026 Boyum IT Solutions GmbH

Allocation Property Name	Type	Description
		property BarHeight and to the width defined in property BarShapeSymbolWidth.
BarShapeSymbolWidth	Integer > 0	**Optional, default: -1 = value of property BarHeight** – Width of the symbol defined in property BarShapeSymbolID when the bar shape named Symbol is used.
BarShowUnstackedInBackground	Boolean	**Optional, default: False** – When this property is set to true, then the allocation is shown in the first sub row behind other allocation bars. This serves to clarify the resource allocation when this allocation e.g. is already finished. The application must set other properties for coloring or bar design to put the allocation bar visually into the background.
BarText	Text	**Optional, default: '' (empty string)** – Text to show on the bar when Regular, Summary or Rectangle bar shape is selected. **Note**: Several immediately consecutive spaces are always combined into one space by the browsers. If the individual spaces are to be preserved, then each of them must be replaced by the Unicode character \u00A0.
BarTextColor	Text (CSS color value)	**Optional, default: 'white'** – Color for the texts of the bar, see property BarText.
BarTextFormat	Text	**Optional, default: '' (empty string) = value of setting DefaultAllocationBarTextFormat** – This property describes the format of the bar text. If not set, then the value of property BarText is displayed. This string contains the placeholders for object values surrounded by double curly braces {{ }}. For example, based on the following string a tooltip with a table containing three rows of key-value pairs is created, where the values are taken from the properties "name" and "firstName" of the referenced object: `{{name}}, {{firstName}}` As an escape, the use of three open curly braces {{{ are displayed as {{.

© 2026 Boyum IT Solutions GmbH

Allocation Property Name	Type	Description
		Additionally, the property name can be extended to contain the desired property type as in {{Start:date}}. At the moment only the types 'date' and ‘number’ are possible besides 'string' (other property types are converted automatically with toString()). The type 'date' converts date values by default using the same format as other dates in the timescale and at the dragging date line captions. You can add another colon followed by a format name, that is defined by the options intlDateTimeFormatOptionsMap or intlNumberFormatOptionsMap, resp. The referenced object is the object on which the tooltip will be shown. For period highlighter entries and allocation entries the referenced object is the main object and not the entry object. It is possible to access related objects by using the following keywords within the property accessor string: • On allocations: >Activity, >Resource • On activities: >Parent, >Calendar • On resources: >Parent, >Calendar, >LoadCurve, >CapacityCurve • Additionally on resources in SkilledResources view: >Skill It is also possible to access variables that are defined by the option applicationVariablesMap by using ?*variableName*. If the value reached is an object, you can then access a property value by using a prefixed dot: .propertyName and you can use […] to access a property value, a map entry or an array entry. Within […] you can use a literal like 5 or A (with or without quotes) or even curly braces {{…}} with the same rules as above.
BarTextPrefixSymbolHeight	Integer	**Optional, default: 12** – Height of the bar symbol before the text (see property BarTextSymbolSymbolID) in pixels at a zoom factor of 100%. The height can be set bigger than the actual bar height and the symbol then will be shown above the bar shape.

© 2026 Boyum IT Solutions GmbH

Allocation Property Name	Type	Description
BarTextPrefixSymbolID	Text	**Optional, default: '' (empty string)** – Identifier of the symbol to be shown before the text inside of the allocation bar. The symbol will be shown vertically centered inside the bar.
BarTextPrefixSymbolWidth	Integer	**Optional, default: 12** – Width of the bar symbol before the text (see property BarTextPrefixSymbolID) in pixels at a zoom factor of 100%.
BarTextWrapMode	Integer (see enum TextWrapMode)	**Optional, default: 0 (None)** – Specifies whether the text inside the bar is wrapped.
BarTooltipTemplateID	Text	**Optional, default: '' (empty string) = value of setting DefaultAllocationBarTooltipTemplateID** – ID of a tooltip template. The template is used for the tooltip that appears on the allocation bar.
BarTopOffset	Integer	**Optional, default: 0** – Offset of the bar in pixels relative to its upper side. A negative number will shift the bar upwards, a positive number will shift the bar downwards.
BarTopOutsideFontFamily	Text (CSS font family value)	**Optional, default: '' (empty string) = value of property BarFontFamily** – Font family for the top outside text. See property BarTopOutsideText. The value does not influence the height of the bar! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-family.
BarTopOutsideFontSize	Text (CSS font size value)	**Optional, default: '14px'** – Font size for the top outside text. See property BarTopOutsideText. See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-size. **Hint**: Only pixel values allowed! The value influences the height of the bar! **Note**: If set, do not forget to append the unit 'px' to the numeric value! Otherwise, the control add-in will ignore the setting.
BarTopOutsideFontStyle	Text (CSS font style value)	**Optional, default: '' (empty string) = value of property BarFontStyle** – Font style for the top outside text. See

© 2026 Boyum IT Solutions GmbH

Allocation Property Name	Type	Description
		property BarTopOutsideText. The value does not influence the height of the bar! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-style
BarTopOutsideFontWeight	Text (CSS font weight value)	**Optional, default: '' (empty string) = value of property BarFontWeight** – Font weight for the top outside text. See property BarTopOutsideText. The value does not influence the height of the bar! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-weight.
BarTopOutsideText	Text	**Optional, default: '' (empty string)** – If set, then the given text is shown above the bar. The text is not clipped at the end of the bar but is clipped when another bar starts in the same vertical position.
BarTopOutsideTextColor	Text	**Optional, default: 'black'** – This sets the color for the text shown by the property BarTopOutsideTextColor.
BorderColor	Text (CSS color value)	**Optional, default: 'gray' *** – Color for the border of the bar. If set to 'calculated', a color will be calculated using the color value of the property Color. This can be useful in situations where two bars are positioned next to each other, and a graphical indicator is needed to visually distinguish the two bars. * If the setting DecouplingOfAllocationPropertiesFromActivities is set to False, then the default value is the current value of property BorderColor of corresponding activity.
BorderDashArray	Text (SVG stroke dash array value)	**Optional, default: 'none'** – Pattern of dashes and gaps for drawing the border of the bar. For further information, please see https://www.w3.org/TR/SVG11/painting.html#StrokeDasharrayProperty or https://developer.mozilla.org/en-US/docs/Web/SVG/Attribute/stroke-dasharray.

© 2026 Boyum IT Solutions GmbH

Allocation Property Name	Type	Description
		The value 'none' indicates that no dashing is used. In this case, the border is drawn solid.
Color	Text (CSS color value)	**Optional, default: '#646464' *** – Fallback color for the entries of the bar, see property Color of AllocationEntry objects. If no entries are defined, then color of the allocation bar itself. See also property NonworkingTimeColor for the coloring of the nonworking times. * If the setting DecouplingOfAllocationPropertiesFromActivities is set to False, then the default value is the current value of property Color of the referenced activity.
ContextMenuID	Text	**Optional, default: '' (empty string)** – Identifier of a ContextMenu object. The strings in it will be shown as a context menu.
ContextMenuIDInBarsPane	Text	**Optional, default: '' (empty string) = value of property ContextMenuID** – Identifier of a ContextMenu object. The strings in it will be shown as a context menu within empty space in the bars pane of an allocation row.
CustomText1	Text	**Optional, default: '' (empty string)** – Freely usable property best usable in C/AL where it is not possible to use freely defined properties like with JsonObjects. The property name can then be used in TableCellDefinition.SymbolIDSource/TextSource and in TooltipTemplate objects. **Note**: Several immediately consecutive spaces are always combined into one space by the browsers. If the individual spaces are to be preserved, then each of them must be replaced by the Unicode character \u00A0.
CustomText2	Text	**Optional, default: '' (empty string)** – Freely usable property best usable in C/AL where it is not possible to use freely defined properties like with JsonObjects. The property name can then be used in TableCellDefinition.SymbolIDSource/TextSource and in TooltipTemplate objects. **Note**: Several immediately consecutive spaces are always combined into one

© 2026 Boyum IT Solutions GmbH

Allocation Property Name	Type	Description
		space by the browsers. If the individual spaces are to be preserved, then each of them must be replaced by the Unicode character \u00A0.
CustomText3	Text	**Optional, default: '' (empty string)** – Freely usable property best usable in C/AL where it is not possible to use freely defined properties like with JsonObjects. The property name can then be used in TableCellDefinition.SymbolIDSource/TextSource and in TooltipTemplate objects. **Note**: Several immediately consecutive spaces are always combined into one space by the browsers. If the individual spaces are to be preserved, then each of them must be replaced by the Unicode character \u00A0.
EarliestDragStart	DateTime	**Optional, default: 0DT** – If set, then the time before the given date is grayed, when beginning to drag the allocation bar. If the setting DragDatesLimitingInteraction is set to True, then the bar itself cannot be dragged before the date.
EarliestEnd	DateTime	**Optional, default: 0DT** – If defined, an additional symbol will be displayed to indicate this date. Please be aware to increase the top row margin so that the symbol has enough space, see setting TopRowMarginInTimeArea.
EarliestEndColor	Text (CSS color value)	**Optional, default: '' (empty string) = value in setting DefaultAllocationConstraintSymbolColor** – Color for the EarliestEnd constraint symbol.
EarliestEndTooltipTemplateID	Text	**Optional, default: '' (empty string) = value of property BarTooltipTemplateID** – Tooltip template for the earliest-end constraint flag.
EarliestStart	DateTime	**Optional, default: 0DT** – If defined, an additional symbol will be displayed to indicate this date. Please be aware to increase the top row margin so that the symbol has enough space, see setting TopRowMarginInTimeArea.

© 2026 Boyum IT Solutions GmbH

Allocation Property Name	Type	Description
EarliestStartColor	Text (CSS color value)	**Optional, default: '' (empty string) = value in setting DefaultAllocationConstraintSymbolColor** – Color for the EarliestStart constraint symbol.
EarliestStartTooltipTemplateID	Text	**Optional, default: '' (empty string) = value of property BarTooltipTemplateID** – Tooltip template for the earliest-start constraint flag.
End	DateTime	**Optional, default: The latest end date of contained entries. If not existing, then value of Activity.End** – End date of the allocation. If not defined explicitly, the latest end date of all entries is calculated and used. If this is not successful and the setting *DecouplingOfAllocationPropertiesFromActivities* is false, then Activity.End is used. If no end date can be determined at all, the allocation will not get visible. The end date itself is not included in the time interval between start and end. This helps to calculate a duration and helps to show bars in a line, when an end date of one bar has the same value as the start date of a next bar. The same behavior also is true for all end dates used in the widget like e.g. in calendar entries, period highlighter entries, activities, activity entries, allocation entries.
EndIsSnapTarget	Boolean	**Optional, default: True** – If set to True, then the visible end date of this allocation in the resources view is used as a snap target for a dragged bar (see properties SnapTargetsForStart and SnapTargetsForEnd).
Entries	JsonArray of AllocationEntry	**Optional, default: empty array** – array of allocation entries.
ID	Text	**Required** – Identifier of the allocation.
LatestDragEnd	DateTime	**Optional, default: 0DT** – If set, then the time after the given date is grayed, when beginning to drag the allocation bar. If the setting DragDatesLimitingInteraction is set to True, then the bar itself cannot be dragged after the date.
LatestEnd	DateTime	**Optional, default: 0DT** – If defined, an additional symbol will be displayed to indicate this date.

© 2026 Boyum IT Solutions GmbH

Allocation Property Name	Type	Description
		Please be aware to increase the top row margin so that the symbol has enough space, see setting TopRowMarginInTimeArea.
LatestEndColor	Text (CSS color value)	**Optional, default: '' (empty string) = value in setting DefaultAllocationConstraintSymbolColor** – Color for the LatestEnd constraint symbol.
LatestEndTooltipTemplateID	Text	**Optional, default: '' (empty string) = value of property BarTooltipTemplateID** – Tooltip template for the latest-end constraint flag.
LatestStart	DateTime	**Optional, default: 0DT** – If defined, an additional symbol will be displayed to indicate this date. Please be aware to increase the top row margin so that the symbol has enough space, see setting TopRowMarginInTimeArea.
LatestStartColor	Text (CSS color value)	**Optional, default: '' (empty string) = value in setting DefaultAllocationConstraintSymbolColor** – Color for the LatestStart constraint symbol.
LatestStartTooltipTemplateID	Text	**Optional, default: '' (empty string) = value of property BarTooltipTemplateID** – Tooltip template for the latest-start constraint flag.
LeftBarSymbolHeight	Integer	**Optional, default: 12** – Height of the left bar symbol (see property LeftBarSymbolID) in pixels at a zoom factor of 100%. The height can be set bigger than the actual bar height and the symbol then will be shown above the bar shape.
LeftBarSymbolID	Text	**Optional, default: undefined** – Identifier of the symbol to be shown at the left side of the allocation bar. The symbol will be shown vertically centered inside the bar. It is drawn over any existing bar text. See also RightBarSymbolID, LeftBarSymbolHeight, and LeftBarSymbolWidth.
LeftBarSymbolWidth	Integer	**Optional, default: 12** – Width of the left bar symbol (see property LeftBarSymbolID) in pixels at a zoom factor of 100%.
LinkSourceDate	DateTime	**Optional, default: 0DT** – Additional date serving as an additional “start point” to

© 2026 Boyum IT Solutions GmbH

Allocation Property Name	Type	Description
		connect a link. See also property “RelationType” of link object.
LinkTargetDate	DateTime	**Optional, default: 0DT** – Additional date serving as an additional “end point” to connect a link. See also property “RelationType” of link object.
MinimumRowHeight	Integer	**Optional, default: -1 = value in setting DefaultAllocationMinimumRowHeight** – Minimum height of the allocation row in pixels. This property is useful, when more than one line of text is shown inside the table cells. Proposal: For one line take 36*, for two lines 52, for three lines 68, and so on. To have the same height also, when no bar is placed in the row, take the maximum of the height of all bars (default bar height is 22) plus 20, so e.g. 42. For using word wrapping in table cells, it is necessary to use a table row definition by setting the property TableRowDefinitionID and setting the property WrapMode in a contained table cell definition. In general, the height of a row is determined by several facts: The height of bars and the stacking of bars determine a height, that then is overlaid by the value of this property. The height of any text inside a table column or a bar is not considered, even when using wrapping.
MustEndOn	DateTime	**Optional, default: 0DT** – If defined, an additional symbol will be displayed to indicate this date. Please be aware to increase the top row margin so that the symbol has enough space, see setting TopRowMarginInTimeArea.
MustEndOnColor	Text (CSS color value)	**Optional, default: '' (empty string) = value in setting DefaultAllocationConstraintSymbolColor** – Color for the MustEndOn constraint symbol.
MustEndOnTooltipTemplateID	Text	**Optional, default: Value of property BarTooltipTemplateID** – Tooltip template for the must-end-on constraint flag.

© 2026 Boyum IT Solutions GmbH

Allocation Property Name	Type	Description
MustStartOn	DateTime	**Optional, default: 0DT** – If defined, an additional symbol will be displayed to indicate this date. Please be aware to increase the top row margin so that the symbol has enough space, see setting TopRowMarginInTimeArea.
MustStartOnColor	Text (CSS color value)	**Optional, default: '' (empty string) = value in setting DefaultAllocationConstraintSymbolColor** – Color for the MustStartOn constraint symbol.
MustStartOnTooltipTemplateID	Text	**Optional, default: '' (empty string) = value of property BarTooltipTemplateID** – Tooltip template for the must-start-on constraint flag.
NonworkingTimeColor	Text (CSS color value or 'calculated')	**Optional, default: '' (empty string)** – Color for the nonworking time periods of the bar. If set to 'calculated', a color will be calculated using the color defined by the Color property. If undefined, nonworking times are colored like working times (taking the value of property Color).
PredictedEnd	DateTime	**Optional, default: 0DT** – A date that indicates the predicted end of the allocation. This date is used to display a bar between this date and the end of the allocation.
PredictedEndColor	Text (CSS color value)	**Optional, default: '#646464'** – Color for the predicted end bar.
Progress	Decimal (in percent; ≥ 0, ≤ 100)	**Optional, default: 0.0** – Used to display a completion layer. * If the setting DecouplingOfAllocationPropertiesFromActivities is set to False, then the default value is the current value the property of same name in the reference activity.
ProgressBackgroundColor	Text (CSS color value)	**Optional, default: '' (empty string) = value in setting DefaultAllocationProgressBackgroundColor** – Color for the background of the progress bar region.
ProgressColor	Text (CSS color value)	**Optional, '#646464' *** – Color for the working time periods of the progress bar.

© 2026 Boyum IT Solutions GmbH

Allocation Property Name	Type	Description
		* If the setting DecouplingOfAllocationPropertiesFromActivities is set to False, then the default value is the current value the property of same name in the reference activity.
ProgressNonworkingTimeColor	Text (CSS color value or 'calculated')	**Optional, default: '' (empty string)** – Color for the nonworking time periods of the progress bar. If set to 'calculated', a color will be calculated using the color defined by the ProgressColor property. If undefined, then the nonworking times are colored like working times (taking the value of property ProgressColor).
ResourceID	Text	**Optional, default: '' (empty string)** – Identifier of a Resource
RightBarSymbolHeight	Integer	**Optional, default: 12** – Height of the right bar symbol (see property RightBarSymbolID) in pixels at a zoom factor of 100%. The height can be set bigger than the actual bar height and the symbol then will be shown above the bar shape.
RightBarSymbolID	Text	**Optional, default: '' (empty string)** – Identifier of the symbol to be shown at the right side of the allocation bar. The symbol will be shown vertically centered inside the bar. It is drawn over any existing bar text. See also LeftBarSymbolID, RightBarSymbolHeight, and RightBarSymbolWidth.
RightBarSymbolWidth	Integer	**Optional, default: 12** – Width of the right bar symbol (see property RightBarSymbolID) in pixels at a zoom factor of 100%.
RowDesign	Integer (see enum RowDesigns)	**Optional, default: -1 = value in setting DefaultAllocationRowDesign** – Specifies how the time area is filled when the row is visible. See explanation for possible values in enumerations chapter.
RowSelectable	Boolean	**Optional, default: value of setting DefaultAllocationRowSelectable** – If set to true, then the row representing this allocation will be selectable.
RowSymbolColumn-BackgroundColor	Text (CSS color value)	**Optional, default: value of property SymbolColumnBackgroundColor of**

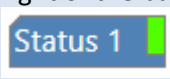
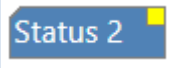
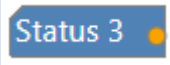
© 2026 Boyum IT Solutions GmbH

Allocation Property Name	Type	Description
		assigned table row definition or setting SymbolColumnBackgroundColor – Determines the color of the symbol column within this table row.
RowSymbolIDs	JsonArray of Text	**Optional, default: empty list** – Array of identifiers of the symbols to be shown in the table symbol cell of the beginning of the table row. The symbols will be arranged one below the other. However, if the cell is not high enough to hold all symbols, then the remaining symbols are also arranged side-by-side. If this still does not fit, an additional “show more” symbol will be displayed. An empty string ('') will cause an “empty” symbol to be displayed. By this placeholder, you can reserve space for a symbol that may be shown at a later time. Please note: Each symbol will be resized to an image with a width and height of 16 pixels each at a zoom level of 100%.
RowTooltipTemplateID	Text	**Optional, default: value of setting DefaultAllocationRowTooltipTemplateID** – ID of a tooltip template. The template is used for tooltips that appear on the entity table rows.
SkilledBarContextMenuID	Text	**Optional, default: value of option BarContextMenuID** – Identifier of a ContextMenu object. The strings in it will be shown as a context menu on a bar in skilled resources view.
SkilledBarTooltipTemplateID	Text	**Optional, default: value of setting DefaultSkilledAllocationBar-TooltipTemplateID** – ID of a tooltip template. The template is used for tooltips that appear on the allocation bars in skilled resources view. It fallbacks to evaluation of the property BarTooltipTemplateID if not set.
SkilledRowContextMenuID	Text	**Optional, default: value of option ContextMenuID** – Identifier of a ContextMenu object. The strings in it will

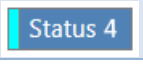
© 2026 Boyum IT Solutions GmbH

Allocation Property Name	Type	Description
		be shown as a context menu on a row in skilled resources view.
SkilledRowTooltipTemplateID	Text	**Optional, default: value of setting DefaultSkilledAllocationRowTooltip-TemplateID** – ID of a tooltip template. The template is used for tooltips that appear on the entity table rows. It fallbacks to evaluation of the property RowTooltipTemplateID if not set.
SkillID	Text	**Optional, default: '' (empty string)** – When set to a valid ID of a Skill object, this influences the appearance of the bar within the resource rows in the skilled resources view.
SnapTargetsForEnd	Integer (see enum SnapTargets)	**Optional, default: -1 = value of setting DefaultAllocationSnap-TargetsForEnd** – When dragging horizontaly, then the visible end date of this allocation will optionally be snapping to date lines, calendar grids, and start or end dates of other allocations in same row, when dragging lets these dates get near the end date (see property EndIsSnapTarget). The user can override an active snapping by pressing the ALT key while dragging.
SnapTargetsForStart	Integer (see enum SnapTargets)	**Optional, default: -1 = value of setting DefaultAllocationSnap-TargetsForStart** – When dragging horizontally, then the visible start date of this allocation will optionally be snapping to date lines, calendar grids, and start or end dates of other allocations in same row, when dragging lets these dates get near the start date (see property StartIsSnapTarget). The user can override an active snapping by pressing the ALT key while dragging.
Start	DateTime	**Optional, default: The earliest start date of contained entries. If not existing, then value in Activity.Start** – Start date of the allocation. If not defined explicitly, the latest start date of all entries is calculated and used. If this is not successful and the option *decouplingOfAllocationPropertiesFromActivities* is false, then Activity.Start is used. If no start date can be determined at all, the allocation will not get visible.

© 2026 Boyum IT Solutions GmbH

Allocation Property Name	Type	Description
StartIsSnapTarget	Boolean	**Optional, default: True** – If set to True, then the visible start date of this allocation in the resources view is used as a snap target for a dragged bar (see properties SnapTargetsForStart and SnapTargetsForEnd)
Status1Color	Text (CSS color value)	**Optional, default: '' (empty string)** – Color for the status symbol to the right of the bar. If undefined, no symbol appears. Only visible, when property Status1Visible is True.
Status1Visible	Boolean	**Optional, default: False** – If set to True and the corresponding status color is set in property Status1Color, then a predefined symbol is displayed to the right of the bar. 
Status2Color	Text (CSS color value)	**Optional, default: '' (empty string)** – Color for the status symbol to the right of the bar. If undefined, no symbol appears. Only visible, when property Status2Visible is True.
Status2Visible	Boolean	**Optional, default: False** – If set to True and the corresponding status color is set in property Status2Color, then a predefined symbol is displayed to the right of the bar. 
Status3Color	Text (CSS color value)	**Optional, default: '' (empty string)** – Color for the status symbol to the right of the bar. If undefined, no symbol appears. Only visible, when property Status3Visible is True.
Status3Visible	Boolean	**Optional, default: False** – If set to True and the corresponding status color is set in property Status3Color, then a predefined symbol is displayed to the right of the bar. 
Status4Color	Text (CSS color value)	**Optional, default: '' (empty string)** – Color for the status symbol to the left of the bar. If undefined, no symbol appears. Only visible, when property Status4Visible is True.
Status4Visible	Boolean	**Optional, default: False** – If set to True and the corresponding status color is set in property Status4Color, then a

© 2026 Boyum IT Solutions GmbH

Allocation Property Name	Type	Description
		predefined symbol is displayed to the left of the bar. **Note: This property may be used with rectangle bar shapes only!** 
StatusFrameColor	Text (CSS color value)	**Optional, default: '' (empty string) = value of setting DefaultActivityStatusFrameColor** – Color for the status frame that will be shown when property StatusFrameVisible is set.
StatusFrameVisible	Boolean	**Optional, default: False** – If set to true, then a frame is shown around the bar. See also property StatusFrameColor.
SuitableActivityIDs	JsonArray of Text	**Optional, default: empty** – An array of IDs of those activities to which the allocation could be assigned. If the array is defined, then all rows of activities that are not listed in that array will be covered by a half-transparent curtain. If the array is empty, all activity rows will be covered. If the array is not defined, then all rows are displayed in the normal way without any covering. Also see settings SuitableActivityOverlayColor and UnsuitableActivityOverlayColor.
SuitableResourceIDs	JsonArray of Text	**Optional, default: undefined** – An array of IDs of those resources to which the allocation could be assigned. If the array is defined, then all rows of resources that are not listed in that array will be covered by a half-transparent curtain. If the array is empty, all resource rows will be covered. If the array is not defined, then all rows are displayed in the normal way without any covering. Also see settings SuitableResourcesOverlayColor and UnsuitableResourcesOverlayColor.
TableColor	Text (CSS color value)	**Optional, default: level-dependent gray** – Color for the table row.

© 2026 Boyum IT Solutions GmbH

Allocation Property Name	Type	Description
		If '' (empty string), a predefined level-dependent gray value is used.
TableColorVisibleInTimeArea	Boolean	**Optional, default: False** – If set to true, the time area row will be colored using the color defined by the TableColor property.
TableFontFamily	Text (CSS font family value)	**Optional, default: '' (empty string) = value of setting FontFamily** – Font family for all texts in the table cells. See property TableText and properties TextSource and TextFormat of TableCellDefinition object. The value does not influence the height of the row! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-family.
TableFontSize	Text (CSS font size value)	**Optional, default: '' (empty string) = value of setting FontSize** – Font size for all texts in the table cells. See property TableText and properties TextSource and TextFormat of TableCellDefinition object. The value does not influence the height of the row! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-size. **Note**: If set, do not forget to append a unit like 'px' to the numeric value! Otherwise, the browser will ignore the setting.
TableFontStyle	Text (CSS font style value)	**Optional, default: '' (empty string) = value of setting FontStyle** – Font style for all texts in the table cells. See property TableText and properties TextSource and TextFormat of TableCellDefinition object. The value does not influence the height of the row! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-style
TableFontWeight	Text (CSS font weight value)	**Optional, default: '' (empty string) = value of setting FontWeight** – Font weight for all texts in the table cells. See property TableText and properties TextSource and TextFormat of TableCellDefinition object. The value does not influence the height of the row!

© 2026 Boyum IT Solutions GmbH

Allocation Property Name	Type	Description
		See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-weight.
TableRowDefinitionID	Text	**Optional, default: '' (empty string) = value of setting DefaultAllocationTableRowDefinitionID** – Identifier of a TableRowDefinition object, that defines the composition of the table row.
TableSortCode	Text	**Optional, default: '' (empty string)** – If given, then the neighboring objects in the table in same level are sorted ascending using this value.
TableText	Text	**Optional, default: '' (empty string)** – Text to display in the table row (see also property TableRowDefinitionID). **Note**: Several immediately consecutive spaces are always combined into one space by the browsers. If the individual spaces are to be preserved, then each of them must be replaced by the Unicode character \u00A0.
TextColor		**Deprecated! Renamed to BarTextColor.**
TooltipText	Text	**Optional, default: '' (empty string)** – Text to show in the tooltip (formatted in HTML).
TopLeftBarSymbolHeight	Integer	**Optional, default: 12** – Height of the top left bar symbol in pixels at a zoom factor of 100%.
TopLeftBarSymbolID	Text	**Optional, default: '' (empty string)** – Identifier of the symbol to be shown at the top left side of the allocation bar. Please note: It protrudes 5 pixels vertically into the bar shape. For height and width see properties TopLeftBarSymbolHeight and TopLeftBarSymbolWidth.
TopLeftBarSymbolWidth	Integer	**Optional, default: 12** – Width of the top left bar symbol in pixels at a zoom factor of 100%.
TopLeftBarSymbolYOffset	Integer	**Optional, default: 0** – Vertical offset of the top left bar symbol in pixels relative to the default position (2/3 of the symbol height above the bar).
TopRightBarSymbolHeight	Integer	**Optional, default: 12** – Height of the top right bar symbol in pixels at a zoom factor of 100%.

© 2026 Boyum IT Solutions GmbH

Allocation Property Name	Type	Description
TopRightBarSymbolID	Text	**Optional, default: '' (empty string)** – Identifier of the symbol to be shown at the top right side of the allocation bar. Please note: It protrudes 5 pixels vertically into the bar shape. For height and width see properties TopRightBarSymbolHeight and TopRightBarSymbolWidth.
TopRightBarSymbolWidth	Integer	**Optional, default: 12** – Width of the top right bar symbol in pixels at a zoom factor of 100%.
TopRightBarSymbolYOffset	Integer	**Optional, default: 0** – Vertical offset of the top right bar symbol in pixels relative to the default position (2/3 of the symbol height above the bar).
Visible	Boolean	**Optional, default: True –** If set to False, then this object is hidden in the chart.

4.2.2 AllocationEntry

NOTE: On data objects that are updated missing properties normally are interpreted as being set to the default value!

AllocationEntry Property Name	Type	Description
Color	Text (CSS color value)	**Optional, default: value of property Color of allocation** – Color for the working time periods of the bar.
End	DateTime	**Optional, default: 0DT** – End date of the allocation entry. The end date itself is not included in the time interval between start and end. This helps to calculate a duration and helps to show bars in a line, when an end date of one bar has the same value as the start date of a next bar. The same behavior also is true for all end dates used in the widget like e.g. in calendar entries, period highlighter entries, activities, activity entries, allocations.
FontFamily	Text (CSS font family value)	**Optional, default: value of property BarFontFamily of allocation** – Font family for the text of the entry. See property Text. See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-family.
FontSize	Text (CSS font size value)	**Optional, default: value of property BarFontSize of allocation** – Font size for the text of the entry. See property Text.

© 2026 Boyum IT Solutions GmbH

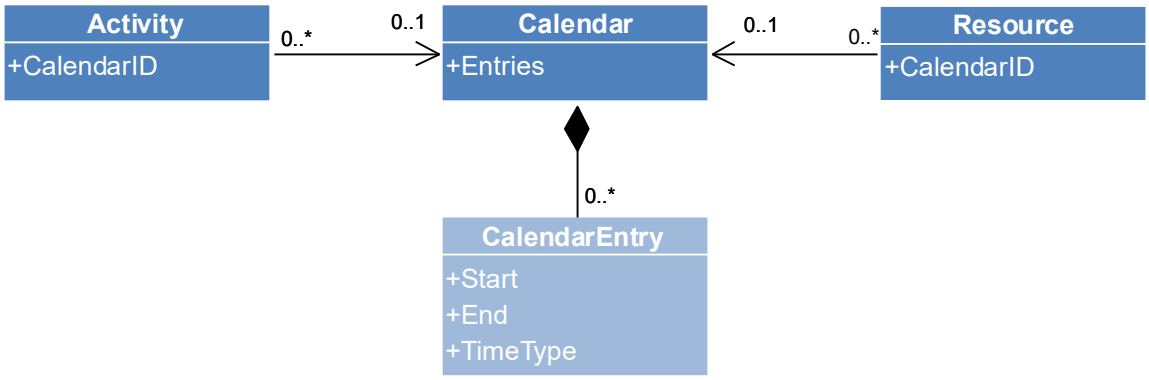
AllocationEntry Property Name	Type	Description
		See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-size. **Note**: If set, do not forget to append a unit like 'px' to the numeric value! Otherwise, the browser will ignore the setting.
FontStyle	Text (CSS font style value)	**Optional, default: value of property BarFontStyle of allocation** – Font style for the text of the entry. See property Text. See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-style
FontWeight	Text (CSS font weight value)	**Optional, default: value of property BarFontWeight of allocation** – Font weight for the text of the entry. See property Text. See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-weight.
Height	Integer (≥ 0, ≤ 1000)	**Optional, default: -1 = value in setting DefaultAllocationBarHeight** – Height of the entry in pixels.
NonworkingTimeColor	Text (CSS color value or 'calculated')	**Optional, default: value of property NonworkingTimeColor of allocation or if undefined then value of property Color** – Color for the nonworking time periods of the bar. If set to empty string, the value of the corresponding allocation, if available, will be used. If that one is also undefined, then the nonworking time periods of the bar will be colored with the same color as the working times (see Color property). If set to 'calculated', a color will be calculated using the color defined by the Color property.
PatternColor	Text	**Optional, default: "white"** – Color for the pattern when this is visible by using property PatternType.
PatternType	Integer (see enum PatternType)	**Optional, default: -1** – If set to a value >= 0, then a pattern is shown on top of the fill color and behind the text.
RelativeTopOffset	Integer	**Optional, default: 0** – Offset of the entry in pixels relative to the upper side of the corresponding allocation. A negative number will shift the entry upwards, a positive number will shift the entry downwards.
Start	DateTime	**Optional, default: 0DT** – Start date of the allocation entry.
Text	Text	**Optional, default: '' (empty string)** – Text to be displayed in the bar when Regular, Summary or Rectangle bar shape is selected.

© 2026 Boyum IT Solutions GmbH

AllocationEntry Property Name	Type	Description
		Note: Several immediately consecutive spaces are always combined into one space by the browsers. If the individual spaces are to be preserved, then each of them must be replaced by the Unicode character \u00A0.
TextColor	Text (CSS color value)	**Optional, default: value of the property BarTextColor of the allocation** – Color for the text of the entry.
TextWrapMode	Integer (see enum TextWrapMode)	**Optional, default: 0 (None)** – Specifies whether the text inside the entry is wrapped. Since wrapping text is performance consuming, the application should use this feature only when really needed.

4.3 Calendars

4.3.1 Calendar



A Calendar object defines working and non-working times to be used with resources and activities (see property CalendarID there).

NOTE: On data objects that are updated missing properties normally are interpreted as being set to the default value!

Calendar Property Name	Type	Description
Entries	JsonArray of CalendarEntry	**Optional, default: empty array** – Array of calendar entry objects. The order of the entries inside the array is important!
ID	Text	**Required** – Identifier of the calendar

© 2026 Boyum IT Solutions GmbH

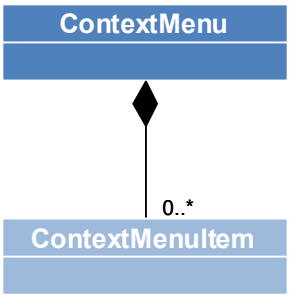
4.3.2 CalendarEntry

A CalendarEntry object defines a single time period. It has to be referenced in the Entries array of a Calendar object. If several calendar entries describe the same time period, then the last entry wins.

NOTE: On data objects that are updated missing properties normally are interpreted as being set to the default value!

CalendarEntry Property Name	Type	Description
End	DateTime	**Optional, default: 0DT** – End of the working time period. The end date itself is not included in the time interval between start and end. This helps to calculate a duration and helps to show bars in a line, when an end date of one bar has the same value as the start date of a next bar. The same behavior also is true for all end dates used in the widget like e.g. in period highlighter entries, activities, activity entries, allocations, allocation entries.
Start	DateTime	**Optional, default: 0DT** – Start of the working time period.
TimeType	Integer	**Optional, default: 1** 0: NonworkingTime, 1: WorkingTime

4.4 ContextMenus



4.4.1 ContextMenu

A ContextMenu object is used to show one when the user clicks the secondary mouse button or holds the finger pressed on an object's representation on screen.

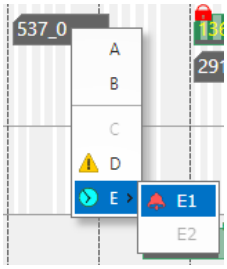
NOTE: On data objects that are updated missing properties normally are interpreted as being set to the default value!

Attention: The interface has changed from 2.2 to 3.0: Items now hold an array of ContextMenuItem objects instead of texts! The older definition is still valid.

© 2026 Boyum IT Solutions GmbH

Example of a context menu containing:

- a separator,
- a disabled menu item,
- a sub menu
- and some symbols:



ContextMenu Property Name	Type	Description
ID	Text	**Required** – Identifier of this context menu
Items	JsonArray of ContextMenuItem	**Optional, default: Empty array** – Items that will be shown in the context menu

4.4.2 ContextMenuItem

NOTE: On data objects that are updated missing properties normally are interpreted as being set to the default value!

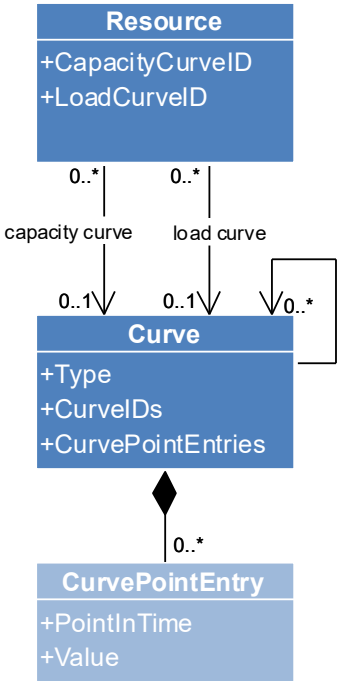
ContextMenuItem Property Name	Type	Description
Code	Text	**Optional, default: index number in items list** – Code that can be used to identify the selected menu item in the OnContextMenuItemClicked event and is mandantory, when using sub menu items in a context menu. This helps identifying the item when the sort code is used.
Disabled	Boolean	**Optional, default: False** – If set to True, then this context menu item will be shown disabled. This property can be overlaid by the evaluated value of the property DisabledSource.
DisabledSource	Text	**Optional, default: '' (empty string)** – If set to an existing property name of the object on which the context menu will be shown, then the value of the referenced property will be taken to show the context menu item disabled. It is possible to reference properties of sub objects by using a syntax like “PropBag.PropName” (also more than one level allowed). On a context menu item for a period highlighter, it is possible to reference the row object by using “#RowObject.” at the beginning of the string. See also property Disabled.
SortCode	Text	**Optional, default: value of property Code** – The value will be used to sort the menu items within the shown context menu by using ascending string sort algorithm. This allows to dynamically add or remove items in the items list.

© 2026 Boyum IT Solutions GmbH

SubItems	JsonArray of ContextMenuItem	**Optional, default: undefined** – If set then, the contained ContextMenuItem object are shown in a sub menu. Please be aware, that a value in the property Code is mandantory, when using sub menu items in a context menu. Sub menus can be used nested.
SymbolID	Text	**Optional, default: '' (empty string)** – ID of a symbol that is shown before the text of the menu item.
SymbolIDSource	Text	**Optional, default: '' (empty string)** – If set to an existing property name of the object on which the context menu will be shown, then the value of the referenced property will be taken to get a symbol that is shown before the text of the menu item. It is possible to reference properties of sub objects by using a syntax like “PropBag.PropName” (also more than one level allowed). On a context menu item for a period highlighter, it is possible to reference the row object by using “#RowObject.” at the beginning of the string. See also property SymbolID.
Text	Text	**Optional, default: '' (empty string)** – Text that will be shown in the context menu item. This text can be HTML, so please make sure, that the text is escaped if necessary. Three dashes ('---') define a separator. **Note**: Several immediately consecutive spaces are always combined into one space by the browsers. If the individual spaces are to be preserved, then each of them must be replaced by the Unicode character \u00A0.

© 2026 Boyum IT Solutions GmbH

4.5 Curves



Curve objects serve to define values over time that can be shown as capacity or load inside resource and activity rows (see properties LoadCurveID and CapacityCurveID in Resource object). Additionally, it is possible to stack curves when using curve object of stack type. Currently, there are no curve types that calculate their values automatically.

Curves are displayed in curves panes. Each pane can hold several curves (e.g., a capacity and a load curve). On the right side of a pane a numerical scale is displayed. By default, the minimum and maximum values for the tick markers are calculated automatically. However, this can be affected by setting the ScaleMinimumValue and the ScaleMaximumValue properties of the curves.

4.5.1 Curve

NOTE: On data objects that are updated missing properties normally are interpreted as being set to the default value!

Curve Property Name	Type	Description
CurveIDs	JsonArray of Text	**Optional, default: empty array** – Array of curve IDs (in case of CurveStack only)
CurvePointEntries	JsonArray of CurvePointEntry	**Optional, default: empty array** – Array of point entries (in case of PointCurve only)
FillColor	Text (CSS color value)	**Optional, default: '' (empty string)** – Color of the area below the curve
ID	Text	**Required** – Identifier of the stacked curve
InterpolationType	Integer (see enum CurveInterpolationType)	**Optional, default: 0** – Type of interpolation. Currently, there are restrictions concerning putting curves of linear interpolation type into curve stacks. It is

© 2026 Boyum IT Solutions GmbH

Curve Property Name	Type	Description
		recommended to use this interpolation type only inside curve lists.
OverloadColor	Text (CSS color value)	**Optional, default: '' (empty string)** – Used, when the curve is used as the load curve that referenced directly by the property LoadCurveID at the Resource object. Then the area above the capacity curve will be colored by this color when the load is higher than the capacity.
ScaleMaximumValue	Decimal	**Optional, default: negative infinity** – If this value here is greater than all values of this curve, then it defines the explicit maximum. Otherwise, the maximum of all curve values defines the overall maximum of this curve. Hereby together with the property ScaleMinimumValue you can define the shown value range for the curve pane.
ScaleMinimumValue	Decimal	**Optional, default: infinity** – If this value here is lower than all values of this curve, then it defines the explicit minimum. Otherwise, the minimum of all curve values defines the overall minimum of this curve. Hereby together with the property ScaleMaximumValue you can define the shown value range for the curve pane.
StrokeColor	Text (CSS color value)	**Optional, default: '' (empty string)** – Color of the curve line itself
StrokeDashArray	Text (SVG stroke dash array value)	**Optional, default: 'none'** – Pattern of dashes and gaps for drawing the curve line. For further information, please see https://www.w3.org/TR/SVG11/painting.html#StrokeDasharrayProperty or https://developer.mozilla.org/en-US/docs/Web/SVG/Attribute/stroke-dasharray. The value 'none' indicates that no dashing is used. In this case, the line is drawn solid.
Type	Integer	**Optional, default: 0** – Type of the curve: 0: PointCurve (displayed curve is determined by interpreting the given curve points) 3: CurveStack (displayed curve is determined by summing up the curve points of the contained curves)

4.5.2 CurvePointEntry

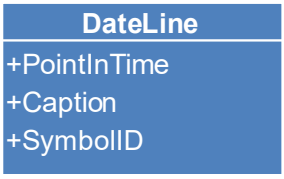
NOTE: On data objects that are updated missing properties normally are interpreted as being set to the default value!

CurvePointEntry Property Name	Type	Description
PointInTime	DateTime	**Required** – This property serves as an identifier of the point entry.

© 2026 Boyum IT Solutions GmbH

CurvePointEntry Property Name	Type	Description
Value	Decimal	**Required** – Value of the curve at the given point in time.

4.6 DateLine



A DateLine object is a pure presentation object and defines the properties of a single date line.

NOTE: On data objects that are updated missing properties normally are interpreted as being set to the default value!

DateLine Property Name	Type	Description
AppearanceType	Integer (see enum DateLineAppearanceType)	**Optional, default: Permanent** – As default the date line is visible permanently. When set to PointerOverAttachedBar, then this date line is only shown when the ID is included in the string array in the property AttachedDateLineIDs of an Activity or Allocation object and the bar representation is currently hovered by the user.
Caption	Text	**Optional, default: '' (empty string)** – Text for the caption of the date line. **Note**: Several immediately consecutive spaces are always combined into one space by the browsers. If the individual spaces are to be preserved, then each of them must be replaced by the Unicode character \u00A0.
CaptionColor	Text (CSS color value)	**Optional, default: 'black'** – Color of the caption.
CaptionColorSource	Text	**Optional, default: undefined** – Only available when the property AppearanceType is set to PointerOverAttachedBar: If set to an existing property name of the attached object, then the value of the referenced property will be taken. Otherwise, the value in the property CaptionColor is taken. It is possible to reference properties of sub objects by using a syntax like “PropBag.PropName” (also more than one level allowed). It defines the color of the caption.

© 2026 Boyum IT Solutions GmbH

DateLine Property Name	Type	Description
CaptionFontFamily	Text (CSS font family value)	**Optional, default: value of setting FontFamily** – Font family for the caption. See properties Caption and CaptionFormat. See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-family.
CaptionFontSize	Text (CSS font size value)	**Optional, default: value of setting FontSize** – Font size for the caption. See properties Caption and CaptionFormat. See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-size. **Note**: If set, do not forget to append a unit like 'px' to the numeric value! Otherwise, the browser will ignore the setting.
CaptionFontStyle	Text (CSS font style value)	**Optional, default: value of setting FontStyle** – Font style for the caption. See properties Caption and CaptionFormat. See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-style
CaptionFontWeight	Text (CSS font weight value)	**Optional, default: value of setting FontWeight** – Font weight for the caption. See properties Caption and CaptionFormat. See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-weight.
CaptionFormat	Text	**Optional, default: undefined** – Only available when the property AppearanceType is set to PointerOverAttachedBar: Text format for the caption of the date line. If set then this property is preferred over using the property Caption. Using this property allows to show a formatted caption including values of properties in an attached Activity or Allocation object (see property AppearanceType). The string contains the placeholders for object values surrounded by double curly braces {{ }}. As an escape, the use of three open curly braces {{{ are displayed as {{. Additionally, the property name can be extended to contain the desired property type as in {{Start:date}}. At the moment only the types 'date' and 'number' are possible besides 'string' (other property types are converted automatically to a string). The type 'date' converts date values by default using the same format as other dates in the timescale and at the dragging date line captions. You can add another colon followed by a format name, that is defined by the

© 2026 Boyum IT Solutions GmbH

DateLine Property Name	Type	Description
		settings IntlDateTimeFormatOptionsMap or IntlNumberFormatOptionsMap, resp. It is also possible to access variables that are defined by the setting ApplicationVariablesMap by using ?*variableName.* If the value reached is an object, you can then access a property value by using a prefixed dot: .propertyName and you can use […] to access a property value, a map entry or an array entry. Within […] you can use a literal like 5 or A (with or without quotes) or even curly braces {{…}} with the same rules as above.
CaptionOrientation	Integer (see enum DateLineCaptionOrientation)	**Optional, default: 2** – Specifies whether the caption should be oriented vertically or horizontally.
CaptionPosition	Integer (see enum DateLineCaptionPosition)	**Optional, default: 1** – Specifies where the caption should be positioned relative to the date line. See also property CaptionOrientation and setting DateLineCaptionOptimizedPositioningEnabled.
Color	Text (CSS color value)	**Optional, default: 'black'** – Color of the date line.
ColorSource	Text	**Optional, default: undefined** – Only available when the property AppearanceType is set to PointerOverAttachedBar: If set to an existing property name of the attached object, then the value of the referenced property will be taken. Otherwise, the value in the property Color is taken. It is possible to reference properties of sub objects by using a syntax like “PropBag.PropName” (also more than one level allowed). It defines the color of the date line.
DashArray	Text (SVG stroke dash array value)	**Optional, default: 'none'** – Pattern of dashes and gaps for drawing the date line. For further information, please see https://www.w3.org/TR/SVG11/painting.html#StrokeDasharrayProperty or https://developer.mozilla.org/en-US/docs/Web/SVG/Attribute/stroke-dasharray. The value 'none' indicates that no dashing is used. In this case, the line is drawn solid.
DashArraySource	Text	**Optional, default: undefined** – Only available when the property AppearanceType is set to PointerOverAttachedBar: If set to an existing property name of the attached object, then the value of the referenced property will be taken. Otherwise, the

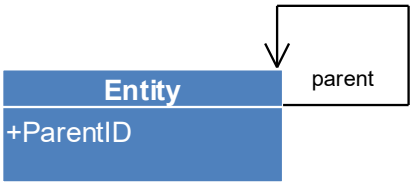
© 2026 Boyum IT Solutions GmbH

DateLine Property Name	Type	Description
		value in the property DashArray is taken. It is possible to reference properties of sub objects by using a syntax like “PropBag.PropName” (also more than one level allowed). It defines the pattern of dashes and gaps for drawing the date line.
Draggable	Boolean	**Optional, default: False** – Only available, when the property AppearanceType is set to Permanent: If set to True, then the date line is draggable and the event OnDrop is triggered, when dropping it at a new date.
ID	Text	**Required** – Identifier of this date line.
InFrontOfBars	Boolean	**Optional, default: True** – Determines how the date line is displayed. If set to False, the date line will be overlapped by the bars. Otherwise, the line will be displayed in front of the bars.
IsVisibleInViewTypes	Integer (see enum ViewTypesForDateLines)	**Optional, default: AllViews** – Determines in which view type this date line is visible.
PointInTime	DateTime	**Optional, default: 0DT** – Date, where the date line should become visible. The date line only gets visible, when the date is set and the date lies between the values of the settings Start and End.
PointInTimeSource	Text	**Optional, default: undefined** – Only available when the property AppearanceType is set to PointerOverAttachedBar: If set to an existing property name of the attached object, then the value of the referenced property will be taken. Otherwise, the value in the property PointInTime is taken. It is possible to reference properties of sub objects by using a syntax like “PropBag.PropName” (also more than one level allowed). It defines the date, where the date line should become visible.
SymbolHeight	Integer > 0	**Optional, default: 12** – Height of the symbol (see property SymbolID) in pixels at a zoom factor of 100%.
SymbolID	Text	**Optional, default: undefined** – When set, then the referenced symbol will be visible at the top of the date line.
SymbolWidth	Integer > 0	**Optional, default: 12** – Width of the symbol (see property SymbolID) in pixels at a zoom factor of 100%.
TooltipTemplateID	Text	**Optional, default: undefined** – ID of a tooltip template. The template is used for tooltips that appear on the date line.
Width	Integer (≥ 0)	**Optional, default: 1** – Line width of the date line.
WidthSource		**Optional, default: undefined** – Only available when the property AppearanceType is set to PointerOverAttachedBar: If set to an existing property

© 2026 Boyum IT Solutions GmbH

DateLine Property Name	Type	Description
		name of the attached object, then the value of the referenced property will be taken. Otherwise, the value in the property Width is taken. It is possible to reference properties of sub objects by using a syntax like “PropBag.PropName” (also more than one level allowed). It defines the line width of the date line.

4.7 Entity



An Entity object defines the properties of a single entity. Entities are shown in a separate table on the right side.

NOTE: On data objects that are updated missing properties normally are interpreted as being set to the default value!

Entity Property Name	Type	Description
AllowedRowDragModes	Integer (see enum RowDragModes)	**Optional, default: -1 = value of setting DefaultEntityAllowedRowDragModes** – This property determines the allowed row drag modes for this entity when the entities table is visible. See also settings RowsDraggable and Editable.
CollapseState	Integer (see option CollapseState)	**Optional, default: -1** – Specifies whether the row of the entity should be expanded or collapsed when displayed the very first time.
ContextMenuID	Text	**Optional, default: '' (empty string)** – Identifier of a ContextMenu object. The strings in it will be shown as a context menu.
CustomText1	Text	**Optional, default: '' (empty string)** – Freely usable property best usable in C/AL where it is not possible to use freely defined properties like with JsonObjects. The property name can then be used in TableCellDefinition.SymbolIDSource/TextSource and in TooltipTemplate objects. **Note**: Several immediately consecutive spaces are always combined into one space by the browsers. If the individual spaces are to be preserved, then each of them must be replaced by the Unicode character \u00A0.

© 2026 Boyum IT Solutions GmbH

Entity Property Name	Type	Description
CustomText2	Text	**Optional, default: '' (empty string)** – Freely usable property best usable in C/AL where it is not possible to use freely defined properties like with JsonObjects. The property name can then be used in TableCellDefinition.SymbolIDSource/TextSource and in TooltipTemplate objects. **Note**: Several immediately consecutive spaces are always combined into one space by the browsers. If the individual spaces are to be preserved, then each of them must be replaced by the Unicode character \u00A0.
CustomText3	Text	**Optional, default: '' (empty string)** – Freely usable property best usable in C/AL where it is not possible to use freely defined properties like with JsonObjects. The property name can then be used in TableCellDefinition.SymbolIDSource/TextSource and in TooltipTemplate objects. **Note**: Several immediately consecutive spaces are always combined into one space by the browsers. If the individual spaces are to be preserved, then each of them must be replaced by the Unicode character \u00A0.
Duration	Integer (in milliseconds)	**Optional, default: 0** – If Duration of the pure working time of the entity. This property is used, for example, when moving the entity from the entities table to the Gantt diagram to display a bar of correct length during interaction.
HasChildren	Boolean	**Optional, default: False** – If set to True, then the row representing this entity will be collapsible/expandable even when there are no children defined. This serves for lazy loading.
ID	Text	**Required** – Identifier of this entity
MinimumRowHeight	Integer	**Optional, default: -1 = value in setting DefaultMinimumEntityRowHeight** – Minimum height of the entity row in pixels. This property is useful, when more than one line of text is shown inside the table cells. Proposal: For one line take 36*, for two lines 52, for three lines 68, and so on. For using word wrapping in table cells, it is necessary to use a table row definition by setting the property TableRowDefinitionID and setting the property WrapMode in a contained table cell definition.
ParentID	Text	**Optional, default: '' (empty string)** – Description of the entity (freely usable)
RowCollapsible	Boolean	**Optional, default: value of setting DefaultEntityRowCollapsible** – If set to True, then

© 2026 Boyum IT Solutions GmbH

Entity Property Name	Type	Description
		the row representing this entity will be interactively collapsible when children exist.
RowSelectable	Boolean	**Optional, default: value of setting DefaultEntityRowSelectable** – If set to True, then the row representing this entity will be selectable.
RowSymbolColumn-BackgroundColor	Text (CSS color value)	**Optional, default: value of property SymbolColumnBackgroundColor of assigned table row definition or setting EntitiesTableSymbolColumnBackgroundColor** – Determines the color of the symbol column within this table row.
RowSymbolIDs	JsonArray of Text	**Optional, default: empty array** – Array of identifiers of the symbols to be shown in the table symbol cell of the beginning of the table row. The symbols will be arranged one below the other. However, if the cell is not high enough to hold all symbols, then the remaining symbols are also arranged side-by-side. If this still does not fit, an additional 'show more" symbol will be displayed. An empty string ('") will cause an 'empty" symbol to be displayed. By this placeholder, you can reserve space for a symbol that may be shown at a later time. Please note: Each symbol will be resized to an image with a width and height of 16 pixels each at a zoom level of 100%.
RowTooltipTemplateID	Text	**Optional, default: value of setting DefaultEntityRowTooltipTemplateID** – ID of a tooltip template. The template is used for the tooltip that appears on the entity table row.
SuitableActivityIDs	JsonArray of Text	**Optional, default: undefined** – An array of IDs of those activities to which the entity could be dropped. If the array is defined, then all rows of activities that are not listed in that array will be covered by a half-transparent curtain. If the array is empty, all activity rows will be covered. If the array is not defined, then all rows are displayed in the normal way without any covering. Also see settings SuitableActivityOverlayColor and _UnsuitableActivityOverlayColor and property SuitableActivityIDs of Allocation objects.

© 2026 Boyum IT Solutions GmbH

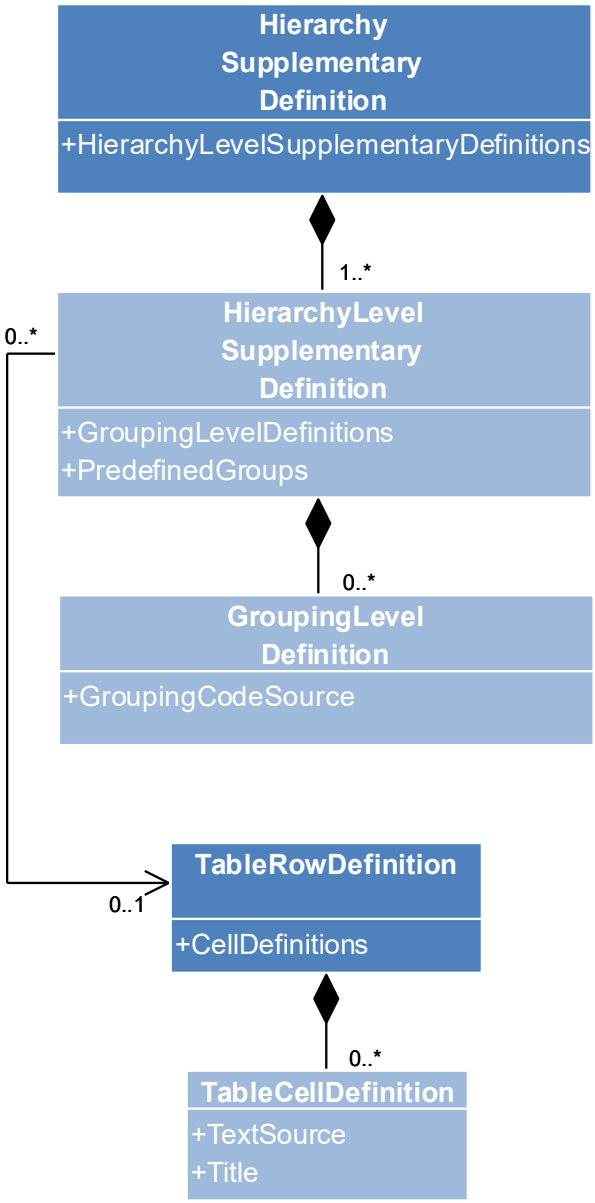
Entity Property Name	Type	Description
SuitableResourceIDs	JsonArray of Text	**Optional, default: undefined** – An array of IDs of those resources to which the entity could be dropped. If the array is defined, then all rows of resources that are not listed in that array will be covered by a half-transparent curtain. If the array is empty, all resource rows will be covered. If the array is not defined, then all rows are displayed in the normal way without any covering. Also see settings SuitableResourcesOverlayColor and UnsuitableResourcesOverlayColor and property SuitableResourceIDs of Allocation objects.
TableColor	Text (CSS color value)	**Optional, default: '' (empty string)** – Color for the table row. If '' (empty string), a predefined level-dependent gray value is used.
TableFontFamily	Text (CSS font family value)	**Optional, default: '' (empty string) = value of setting FontFamily** – Font family for all texts in the table cells. See property TableText and properties TextSource and TextFormat of TableCellDefinition object. The value does not influence the height of the row! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-family.
TableFontSize	Text (CSS font size value)	**Optional, default: '' (empty string) = value of setting FontSize** – Font size for all texts in the table cells. See property TableText and properties TextSource and TextFormat of TableCellDefinition object. The value does not influence the height of the row! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-size. **Note**: If set, do not forget to append a unit like 'px' to the numeric value! Otherwise, the browser will ignore the setting.
TableFontStyle	Text (CSS font style value)	**Optional, default: '' (empty string) = value of setting FontStyle** – Font style for all texts in the table cells. See property TableText and properties TextSource and TextFormat of TableCellDefinition object. The value does not influence the height of the row! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-style

© 2026 Boyum IT Solutions GmbH

Entity Property Name	Type	Description
TableFontWeight	Text (CSS font weight value)	**Optional, default: '' (empty string) = value of setting FontWeight** – Font weight for all texts in the table cells. See property TableText and properties TextSource and TextFormat of TableCellDefinition object. The value does not influence the height of the row! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-weight.
TableRowDefinitionID	Text	**Optional, default: '' (empty string) = value of setting DefaultEntityTableRowDefinitionID** – Identifier of a TableRowDefinition object that defines the composition of the table row.
TableSortCode	Text	**Optional, default: '' (empty string)** – If given, then the neighboring objects in the table in same level are sorted ascending using this value.
TableText	Text	**Optional, default: '' (empty string)** – Text to show in the table. **Note**: Several immediately consecutive spaces are always combined into one space by the browsers. If the individual spaces are to be preserved, then each of them must be replaced by the Unicode character \u00A0.
TableTextColor	Text (CSS color value)	**Optional, default: '' (empty string)** – Color for the table row texts. If '' (empty string), a default value of the control add-in will be used.
TooltipText	Text	**Optional, default: '' (empty string)** – Text to show in the tooltip (formatted in HTML). **Note**: Several immediately consecutive spaces are always combined into one space by the browsers. If the individual spaces are to be preserved, then each of them must be replaced by the Unicode character \u00A0.
Visible	Boolean	**Optional, default: True –** If set to False, then this object is hidden in the chart.

© 2026 Boyum IT Solutions GmbH

4.8 HierarchySupplementaryDefinitions



4.8.1 HierarchySupplementaryDefinition

A HierarchySupplementaryDefinition object defines the additional grouping of a complete hierarchy of row objects by using criteria that are taken from property values within the row objects. Each HierarchyLevelSupplementaryDefinition object defines all groupings for a hierarchy level of row objects. The hierarchy levels are already built by using the property ParentID of the row objects of type Activity, Entity, Resource. The ID of a HierarchySupplementaryDefinition object is set into one of the

© 2026 Boyum IT Solutions GmbH

options activity/entity/resourceHierarchySupplementaryDefinitionID. Currently the skilled resources view cannot be grouped.

A first simple definition for the first hierarchy level with only one grouping level consists of only a few set properties:

```
{
  "ID": "HSD1",
  "HierarchyLevelSupplementaryDefinitions": [
    // hierarchy level 0
    {
      "GroupingLevelDefinitions": [
        // first grouping level
        {
          "GroupingCodeSource": "_Grp1", // e.g., referencing an application-defined property
          "TableBackgroundColor": "gold"
        }
      ]
    }
  ]
}
```

NOTE: On data objects that are updated missing properties normally are interpreted as being set to the default value!

HierarchySupplementaryDefinition Property Name	Type	Description
HierarchyLevel-SupplementaryDefinitions	JsonArray of HierarchyLevel-SupplementaryDefinition	**Optional, default: empty list** – Array of hierarchy level supplementary definitions. The first object contains definitions for hierarchy level 0, the second defines level 1 and so on. If one hierarchy level should not be grouped additionally, then you can leave the array entry empty either by using an empty object or alternatively undefined/null.
ID	Text	**Required** – Identifier of this hierarchy definition.

4.8.2 HierarchyLevelSupplementaryDefinition

Each HierarchyLevelSupplementaryDefinition object defines additional grouping for a level of row objects. Used in HierarchySupplementaryDefinition objects.

NOTE: On data objects that are updated missing properties normally are interpreted as being set to the default value!

© 2026 Boyum IT Solutions GmbH

HierarchyLevel-SupplementaryDefinition Property Name	Type	Description
GroupingLevelDefinitions	JsonArray of GroupingLevel-Definition	**Optional, default: empty list** – Array of grouping level definitions. By using more than one GroupingLevelDefinition object in the array you can define multiple grouping criteria for one hierarchy level at once.
InitiallyCollapsed	Boolean	**Optionally, default: False** – If this property is set to True, then the table rows in this hierarchy level will initially show collapsed. This property is only effective when the property CollapseState is set to 0 or 1 on the affected table row object (Activity/Entity/Resource).

PredefinedGroups	JsonArray of JsonObject	**Optionally, default: undefined** – If this property is set to an array, then each object in the array defines one or more predefined groups.
		
		In each object you can define which predefined group(s) to create by including some or all grouping codes up to the current hierarchy level. Additionally, you can set the following graphical properties for the predefined group: MinimumRowHeight, TableColor, TableTextColor.
		
		Example for second hierarchy level:
		
		{
		"*GroupingCodeSourceOfLevel0*": "a0",
		"*GroupingCodeSourceOfLevel1*": "b1",
		"TableColor": "lime"
		}
		
		The property names in italics have to be replaced by the property names in your hierarchy level definitions on levels 0 and 1! This example defines a predefined group with code "b1" within another group with the code "a0". The group "a0" may be predefined or not. If it does not exist, it will be created additionally before creating "b1" using default coloring.
		
		In general, predefined groups are created before any other groups that are determined by grouping codes within the grouped row objects.

© 2026 Boyum IT Solutions GmbH

HierarchyLevel-SupplementaryDefinition Property Name	Type	Description
TableRowDefinitionID	Text	**Optional, default: '' (empty string)** – If set then the value overwrites the setting on an affected table row object (Activity/Entity/Resource).

4.8.3 GroupingLevelDefinition

The GroupingLevelDefinition object defines the grouping criteria for all grouping levels of one hierarchy level as seen in a table on screen. The grouping level definition also defines the display of the resulting group lines. Used in a HierarchyLevelSupplementaryDefinition object.

NOTE: On data objects that are updated missing properties normally are interpreted as being set to the default value!

GroupingLevelDefinition Property Name	Type	Description
CodeToTextMap	JsonObject	**Optional, default: undefined** – If set then the object is used to map the group codes (key) to a long text (value) that will be shown on the table row representing the generated group.
DefaultCode	Text	**Optional, default: '' (empty string)** – If this property is set, the value serves as a default grouping criterion, if not otherwise defined.
GroupingCodeSource	Text	**Optional, default: '' (empty string)** – Names a property of the objects under consideration, the content of which is used as a grouping criterion.
InitiallyCollapsed	Boolean	**Optionally, default: False** – If this property is set to True, then the generated group rows initially are shown collapsed.
MinimumRowHeight	Integer > 0	**Optional, default: -1 = value of appropriate default setting for the table** – Minimum height of the group row in pixels. This property is useful, when more than one line of text is shown inside the table cells. Proposal: For one line take 36, for two lines 52, for three lines 68, and so on. To have the same height also, when no bar is placed in the row, take the maximum bar height adding 20 (e.g. 42) as minimum.
TableColor	Text (CSS color value)	**Optional, default: value of the higher group row** – Color for the table row.
TableColorVisibleInTimeArea	Boolean	**Optional, default: False** – If set to True, the time area row will be colored using

© 2026 Boyum IT Solutions GmbH

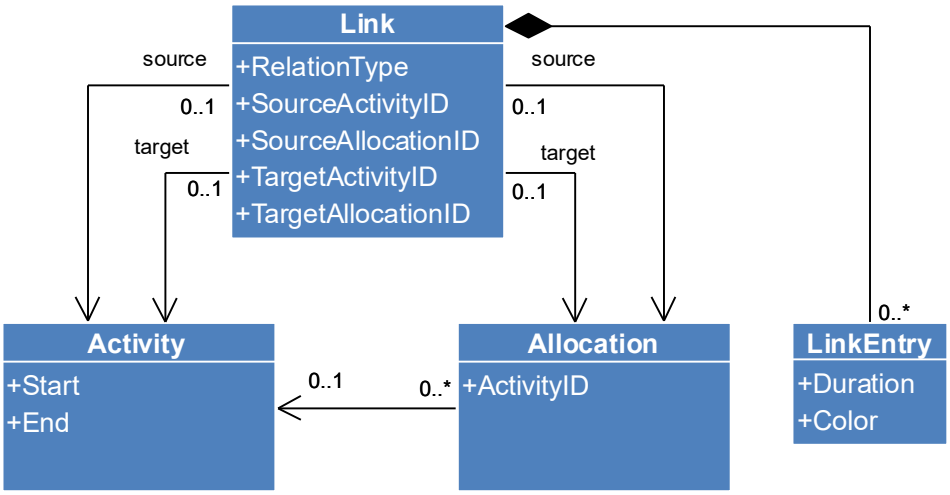
GroupingLevelDefinition Property Name	Type	Description
		the color defined by the TableColor property.
TableFontFamily	Text (CSS font family value)	**Optional, default: '' (empty string) = value of setting FontFamily** – Font family for all texts in the table cells. See property TableText and properties TextSource and TextFormat of TableCellDefinition object. The value does not influence the height of the row! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-family.
TableFontSize	Text (CSS font size value)	**Optional, default: '' (empty string) = value of setting FontSize** – Font size for all texts in the table cells. See property TableText and properties TextSource and TextFormat of TableCellDefinition object. The value does not influence the height of the row! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-size. **Note**: If set, do not forget to append a unit like 'px' to the numeric value! Otherwise, the browser will ignore the setting.
TableFontStyle	Text (CSS font style value)	**Optional, default: '' (empty string) = value of setting FontStyle** – Font style for all texts in the table cells. See property TableText and properties TextSource and TextFormat of TableCellDefinition object. The value does not influence the height of the row! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-style
TableFontWeight	Text (CSS font weight value)	**Optional, default: '' (empty string) = value of setting FontWeight** – Font weight for all texts in the table cells. See property TableText and properties TextSource and TextFormat of TableCellDefinition object. The value does not influence the height of the row!

© 2026 Boyum IT Solutions GmbH

GroupingLevelDefinition Property Name	Type	Description
		See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-weight.
TableTextColor	Text (CSS color value)	**Optional, default: value of the higher group row** – Color for the table row texts.
TableTextFormat	Text	**Optionally, default: undefined** – If this property is set, then text in table on generated group rows is formatted using the value. The string can contain the keyword {{>MapText}} to include the value of the map that is defined by the property GroupingCodeToTextMap else will show the group code itself.

4.9 Links

4.9.1 Link



A Link object defines the properties of a single link between activities or allocations.

Links between activities are shown in activities view when the option linksVisibleInActivitiesView is true. Additionally, links between allocations are shown in activities view when the options definedAllocationLinksVisibleInActivitiesView and allocationRowsVisibleInActivitiesView are true additionally.

Links between allocations in resources view are shown when the option linksVisibleInResourcesView is true. By default, the links between activities are shown as allocation links, but when the option definedAllocationLinksVisibleInResourcesView is true, then the defined allocation links are shown in resources view.

Links cannot connect bar objects that are in different view areas (see property ViewArea of activity and resource objects). Links that are defined that way are not rendered.

© 2026 Boyum IT Solutions GmbH

A link can have several entries to color it section by section. To do this, each section is defined by a duration and a color.

NOTE: On data objects that are updated missing properties normally are interpreted as being set to the default value!

Link Property Name	Type	Description
Color	Text (CSS color value)	**Optional, default: '' (empty string)** – Color for the link. If '' (empty string), a default value will be used.
ContextMenuID	Text	**Optional, default: '' (empty string)** – Identifier of a ContextMenu object. The strings in it will be shown as a context menu.
CustomText1	Text	**Optional, default: '' (empty string)** – Freely usable property best usable in C/AL where it is not possible to use freely defined properties like with JsonObjects. The property name can then be used in TableCellDefinition.SymbolIDSource/TextSource and in TooltipTemplate objects. **Note**: Several immediately consecutive spaces are always combined into one space by the browsers. If the individual spaces are to be preserved, then each of them must be replaced by the Unicode character \u00A0.
CustomText2	Text	**Optional, default: '' (empty string)** – Freely usable property best usable in C/AL where it is not possible to use freely defined properties like with JsonObjects. The property name can then be used in TableCellDefinition.SymbolIDSource/TextSource and in TooltipTemplate objects. **Note**: Several immediately consecutive spaces are always combined into one space by the browsers. If the individual spaces are to be preserved, then each of them must be replaced by the Unicode character \u00A0.
CustomText3	Text	**Optional, default: '' (empty string)** – Freely usable property best usable in C/AL where it is not possible to use freely defined properties like with JsonObjects. The property name can then be used in TableCellDefinition.SymbolIDSource/TextSource and in TooltipTemplate objects. **Note**: Several immediately consecutive spaces are always combined into one space by the browsers. If the individual spaces are to be preserved, then each of them must be replaced by the Unicode character \u00A0.
DashArray	Text	**Optional, default: 'none'** – Pattern of dashes and gaps for drawing the link. For further information, please see

© 2026 Boyum IT Solutions GmbH

Link Property Name	Type	Description
	(SVG stroke dash array value)	https://www.w3.org/TR/SVG11/painting.html#StrokeDasharrayProperty or https://developer.mozilla.org/en-US/docs/Web/SVG/Attribute/stroke-dasharray. The value 'none' indicates that no dashing is used. In this case, the link is drawn solid.
Entries	LinkEntry[]	**Optional, default: undefined** – Array of link entry objects. This serves to color link section-wise. **Note:** The order of the entries inside the array is important.
ID	Text	**Required** – Identifier of this link
RelationType	Integer	**Optional, default: 0** – The relation type is used for drawing: 0: Finish-Start, 1: Finish-Finish, 2: Start-Start, 3: Start-Finish, 4: SourceDate-Start, 5: SourceDate-Finish, 8: Finish-TargetDate, 10: Start-TargetDate, 12: SourceDate-TargetDate
RoutingType	Integer (see enum LinkRoutingType)	**Optional, default: -1 = value of setting DefaultLinkRoutingType** – type of the link routing. 1: Curved, 2: Orthogonal
Selectable	Boolean	**Optional, default: value of setting DefaultLinkSelectable** – If set to True, then the link will be selectable.
SourceActivityID	Text	**Optional, default: '' (empty string)** – Identifier of the source activity. This property or SourceAllocationID has to be set.
SourceAllocationID	Text	**Optional, default: '' (empty string)** – Identifier of the source allocation. This property or SourceActivityID has to be set.
TargetActivityID	Text	**Optional, default: '' (empty string)** – Identifier of the source activity. This property or TargetAllocationID has to be set.
TargetAllocationID	Text	**Optional, default: '' (empty string)** – Identifier of the target allocation. This property or TargetActivityID has to be set.
TargetMarker	Integer (see enum LinkMarker)	**Optional, default: 1 (FilledArrow)** – allows to change the marker at the end (target) of a link.
TooltipTemplateID	Text	**Optional, default: value of setting DefaultLinkTooltip-TemplateID** – ID of a tooltip template. The template is used for the tooltip that appears on the link.
TooltipText	Text	**Optional, default: '' (empty string)** – Text to show in the tooltip (formatted in HTML). **Note**: Several immediately consecutive spaces are always combined into one space by the browsers. If the individual spaces are to be preserved, then each of them must be replaced by the Unicode character \u00A0.

© 2026 Boyum IT Solutions GmbH

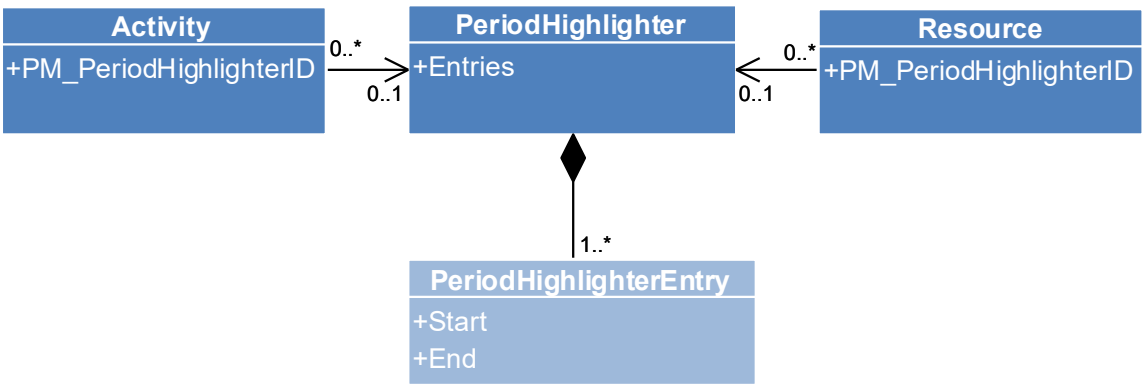
Link Property Name	Type	Description
Width	Integer (≥ 0)	**Optional, default: 1** – Line width of the link. The link arrow is also affected by this property.

4.9.2 LinkEntry

NOTE: On data objects that are updated missing properties normally are interpreted as being set to the default value!

LinkEntry Property Name	Type	Description
Color	Text	**Optional, default: value of property Color of link** – Color for the defined time period.
Duration	Integer (≥ 0)	**Optional, default: 0** – Duration in milliseconds directly following the entry before this one in the array. If it the first entry, then the time period starts at the date and time where the link starts. The duration given is taken as an absolute value (no working-time calendar is taken into account). This serves for marking times such as a transport time or cool down time after an activity or allocation is finished.

4.10 PeriodHighlighters



A PeriodHighlighter object is a pure presentation object and defines the properties of a series of time periods that can be shown on each resource row and activity row (see property PeriodHighlighterID there). Each time period can be colored independently and can have a caption. Period highlighters also support the event OnDoubleClicked. In contrast to the grids created by Calendar objects, the time periods do not define work or non-working times, but only highlight time periods visually.

4.10.1 PeriodHighlighter

NOTE: On data objects that are updated missing properties normally are interpreted as being set to the default value!

© 2026 Boyum IT Solutions GmbH

PeriodHighlighter Property Name	Type	Description
Entries	JsonArray of PeriodHighlighterEntry	**Required** – Array of entries that contain single time periods.
ID	Text	**Required** – Identifier of this period highlighter.

4.10.2 PeriodHighlighterEntry

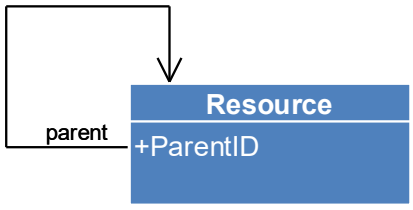
NOTE: On data objects that are updated missing properties normally are interpreted as being set to the default value!

PeriodHighlighterEntry Property Name	Type	Description
Caption	Text	**Optional, default: '' (empty string)** – Text to show on the time period. **Note**: Several immediately consecutive spaces are always combined into one space by the browsers. If the individual spaces are to be preserved, then each of them must be replaced by the Unicode character \u00A0.
CaptionColor	Text (CSS color value)	**Optional, default: 'white'** – Color of the caption.
CaptionFontFamily	Text (CSS font family value)	**Optional, default: value of setting FontFamily** – Font family for the caption. See property Caption. See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-family.
CaptionFontSize	Text (CSS font size value)	**Optional, default: value of setting FontSize** – Font size for the caption. See property Caption. See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-size. **Note**: If set, do not forget to append a unit like 'px' to the numeric value! Otherwise, the browser will ignore the setting.
CaptionFontStyle	Text (CSS font style value)	**Optional, default: value of setting FontStyle** – Font style for the caption. See property Caption. See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-style

© 2026 Boyum IT Solutions GmbH

PeriodHighlighterEntry Property Name	Type	Description
CaptionFontWeight	Text (CSS font weight value)	**Optional, default: value of setting FontWeight** – Font weight for the caption. See property Caption. See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-weight.
Color	Text (CSS color value)	**Optional, default: 'rgba(0,0,0,0.1)'** – Color of this time period.
ContextMenuID	Text	**Optional, default: '' (empty string)** – Identifier of a ContextMenu object. The strings in it will be shown as a context menu.
End	DateTime	**Required** – End of the time period. The end date itself is not included in the time interval between start and end. This helps to calculate a duration and helps to show bars in a line, when an end date of one bar has the same value as the start date of a next bar. The same behavior also is true for all end dates used in the widget like e.g. in calendar entries, activities, activity entries, allocations, allocation entries.
Start	DateTime	**Required** – Start of the time period.
TooltipTemplateID	Text	**Optional, default: value of setting DefaultPeriodHighlighterEntryTooltip-TemplateID** – ID of a tooltip template. The template is used for the tooltip that appears on the entry.
TooltipText	Text	**Optional, default: '' (empty string)** – Text to show in a tooltip. **Note**: Several immediately consecutive spaces are always combined into one space by the browsers. If the individual spaces are to be preserved, then each of them must be replaced by the Unicode character \u00A0.

4.11 Resource



© 2026 Boyum IT Solutions GmbH

A Resource object defines the properties of a single resource.

NOTE: On data objects that are updated missing properties normally are interpreted as being set to the default value!

Resource Property Name	Type	Description
AllocationRowsCollapseState	Integer (see enum CollapseState)	**Optional, default: -1** – Specifies whether the allocation rows of the resource should be expanded or collapsed when displayed. See also event OnCollapseStateChanged and setting AllocationRowsVisibleInResourcesView. -1: no change (when used on startup, the allocation rows will be shown collapsed) 0: display resource row in an expanded way for allocation rows. 1: display resource row in a collapsed way for allocation rows.
AllocationRowsCollapseStateInActivitiesView	Integer (see enum CollapseState)	**Optional, default: -1** – Specifies whether the allocation rows of the resource rows in the activities view should be expanded or collapsed when displayed. -1: no change (when used on startup, the allocation rows will be shown collapsed) 0: display resource row in an expanded way for allocation rows. 1: display resource row in a collapsed way for allocation rows.
AllocationRowsCollapsible	Boolean	**Optional, default: value of setting DefaultResourceAllocationRowsCollapsible** – If set to true, then the row representing this resource row will be interactively collapsible when allocation rows exist.
AllowedRowDragModes	Integer (see enum RowDragModes)	**Optional, default: value of setting DefaultResourceAllowedRowDragModes** – This property determines the allowed row drag modes for this resource in resources view and loads view. See also settings RowsDraggable and Editable.
CalendarGridColor	Text (CSS color value)	**Optional, default: value of setting CalendarGridColor** – Specifies a color used to color the vertical stripes representing the nonworking times for the activity object inside the diagram. If allocation rows are visible the color is used for these rows, too.
CalendarID	Text	**Optional, default: '' (empty string)** – Corresponding calendar. If undefined,

© 2026 Boyum IT Solutions GmbH

Resource Property Name	Type	Description
		then the calendar specified by the option defaultCalendarID will be used.
CapacityCurveID	Text	**Optional, default: '' (empty string)** – Identifier of any curve representing the capacity of this resource. If the identifier references a curve stack, then the summed curve is shown with the color settings of the curve stack.
CollapsedRowDesign	Integer (see enum RowDesigns)	**Optional, default: -1 = value in setting DefaultResourceCollapsedRowDesign** – Specifies how the time area is filled when the row is collapsed and visible. See explanation for possible values in enumerations chapter.
CollapseState	Integer Integer (see enum CollapseState)	**Optional, default: -1** – Specifies whether the row of the resource should be expanded or collapsed when displayed in resources view and eventually in loads view (see property CollapseStateInLoadsView). See also event OnCollapseStateChanged. -1: do not change the way the resource row is displayed 0: display resource row in an expanded way 1: display resource row in a collapsed way See option CollapseState in the Enumerations chapter for details.
CollapseStateInLoadsView	number (see enum CollapseState)	**Optional, default: -2** – Specifies whether the row of the resource should be expanded or collapsed when displayed in the loads view. If -2, then the property CollapseState is used for compatibility reasons. See also event OnCollapseStateChanged. -2: fallback to property CollapseState -1: no change 0: display resource row in an expanded way 1: display resource row in a collapsed way
ContextMenuID	Text	**Optional, default: '' (empty string)** – Identifier of a ContextMenu object. The strings in it will be shown as a context menu.
ContextMenuIDInBarsPane	Text	**Optional, default: value of property ContextMenuID** – Identifier of a

© 2026 Boyum IT Solutions GmbH

Resource Property Name	Type	Description
		ContextMenu object. The strings in it will be shown as a context menu within empty space in the bars pane of a resource row.
ContextMenuIDInCurvesPane	Text	**Optional, default: value of property ContextMenuID** – Identifier of a ContextMenu object. The strings in it will be shown as a context menu within the curves pane of a resource row.
CurveCollapseState	Integer (see enum CollapseState)	**Optional, default: -1** – Specifies whether the curves in a resource row should be expanded or collapsed when displayed. See also event OnCurveCollapseStateChanged. -1: no change 0: display curves 1: hide curves
CurveTooltipTemplateID	Text	**Optional, default: value of setting DefaultResourceCurveTooltipTemplateID** – ID of a tooltip template. The template is used for the tooltip that appears on the curve area of a resource.
CustomText1	Text	**Optional, default: '' (empty string)** – Freely usable property best usable in C/AL where it is not possible to use freely defined properties like with JsonObjects. The property name can then be used in TableCellDefinition.SymbolIDSource/TextSource and in TooltipTemplate objects. **Note**: Several immediately consecutive spaces are always combined into one space by the browsers. If the individual spaces are to be preserved, then each of them must be replaced by the Unicode character \u00A0.
CustomText2	Text	**Optional, default: '' (empty string)** – Freely usable property best usable in C/AL where it is not possible to use freely defined properties like with JsonObjects. The property name can then be used in TableCellDefinition.SymbolIDSource/TextSource and in TooltipTemplate objects. **Note**: Several immediately consecutive spaces are always combined into one space by the browsers. If the individual spaces are to be preserved, then each of them must be replaced by the Unicode character \u00A0.

© 2026 Boyum IT Solutions GmbH

Resource Property Name	Type	Description
CustomText3	Text	**Optional, default: '' (empty string)** – Freely usable property best usable in C/AL where it is not possible to use freely defined properties like with JsonObjects. The property name can then be used in TableCellDefinition.SymbolIDSource/TextSource and in TooltipTemplate objects. **Note**: Several immediately consecutive spaces are always combined into one space by the browsers. If the individual spaces are to be preserved, then each of them must be replaced by the Unicode character \u00A0.
ExpandedRowDesign	Integer (see enum RowDesigns)	**Optional, default: -1 = value in setting DefaultResourceExpandedRowDesign** – Specifies how the time area is filled when the row is expanded and visible. See explanation for possible values in enumerations chapter.
HasChildren	Boolean	**Optional, default: False** – If set to True, then the row representing this resource will be collapsible/expandable even when there are no children defined. This serves for lazy loading.
HasCurves	Boolean	**Optional, default: False** – If set to True, then the row representing this resource will be collapsible/expandable for curves even where there ae no curves defined. This serves for lazy loading.
ID	Text	**Required** – Identifier of the resource
LoadCurveID	Text	**Optional, default: '' (empty string)** – Identifier of any curve representing the load of this resource. If the identifier references a curve stack, then all curves within the curve stack are shown with their individual color settings as a stack.
LoadCurvePaneHeight	Integer (> 0)	**Optional, default: value of setting DefaultResourceLoadCurvePaneHeight** – Height in pixels of the load curve pane.
MinimumRowHeight	Integer	**Optional, default: -1 = value in setting DefaultResourceMinimumRowHeight** – Minimum height of the resource row in pixels. This option is useful, when more than one line of text is shown inside the table cells. Proposal: For one line take 36*, for two lines 52, for three lines 68, and so on. To have the same height also, when no bar is placed in the row, take the

© 2026 Boyum IT Solutions GmbH

Resource Property Name	Type	Description
		maximum of the height of all bars (default bar height is 22) plus 20, so e.g. 42. For using word wrapping in table cells, it is necessary to use a table row definition by setting the property TableRowDefinitionID and setting the property WrapMode in a contained table cell definition. In general, the height of a row is determined by several facts: The height of bars and the stacking of bars determine a height, that then is overlaid by the value of this property. The height of any text inside a table column or a bar is not considered, even when using wrapping.
ParentID	Text	**Optional, default: '' (empty string)** – Identifier of a parent resource this resource is assigned to. If this property is defined, the parent resource will become a resource group (if not yet a resource group) and it will keep its role as a resource with a capacity of its own. If this property is undefined the current resource will be considered as a root node of the resource hierarchy.
PeriodHighlighterID	Text	**Optional, default: '' (empty string)** – Reference to a period highlighter object that contains colored time periods. This can be used to show shifts or exceptions to the calendar (see property CalendarID) that defines work and non-work times.
RowCollapsible	Boolean	**Optional, default: value of setting DefaultResourceRowCollapsible** – If set to True, then the row representing this resource will be interactively collapsible when children exist.
RowSelectable	Boolean	**Optional, default: value of setting DefaultResourceRowSelectable** – If set to True, then the row representing this resource will be selectable.
RowSymbolColumnBackgroundColor	Text (CSS color value)	**Optional, default: value of property SymbolColumnBackgroundColor of assigned table row definition or setting SymbolColumnBackgroundColor** – Determines the color of the symbol column within this table row.
RowSymbolIDs	JsonArray of Text	**Optional, default: empty array** – Array of identifiers of the symbols to be shown in the table symbol cell of the beginning of the table row.

© 2026 Boyum IT Solutions GmbH

Resource Property Name	Type	Description
		The symbols will be arranged one below the other. However, if the cell is not high enough to hold all symbols, then the remaining symbols are also arranged side-by-side. If this still does not fit, an additional 'show more” symbol will be displayed. An empty string ('”) will cause an empty-symbol to be displayed. By this placeholder, you can reserve space for a symbol that may be shown at a later time. Please note: Each symbol will be resized to an image with a width and height of 16 pixels each at a zoom level of 100%.
RowTooltipTemplateID	Text	**Optional, default: value of setting DefaultResourceRowTooltipTemplateID** – ID of a tooltip template. The template is used for the tooltip that will appear on the table row when hovering the cursor above it.
SkilledRowContextMenuID	Text	**Optional, default: value of option ContextMenuID** – Identifier of a ContextMenu object. The strings in it will be shown as a context menu on a row in skilled resources view.
SkilledRowTooltipTemplateID	Text	**Optional, default: value of setting DefaultSkilledResourceRowTooltip-TemplateID** – ID of a tooltip template. The template is used for tooltips that appear on the entity table rows. It fallbacks to evaluation of the property RowTooltipTemplateID if not set.
SkillIDs	JsonArray of Text	**Optional, default: undefined** – If set, then the resource object is shown as a row below the rows of the referenced Skill objects in the skilled resources view. When setting this property, it is not allowed to set the property ParentID!
TableColor	Text (CSS color value)	**Optional, default: '' (empty string)** – Color for the table row. If '' (empty string), a predefined level-dependent gray value is used.
TableColorVisibleInTimeArea	Boolean	**Optional, default: False** – If set to true, the time area row will be colored using the color defined by the TableColor property.

© 2026 Boyum IT Solutions GmbH

Resource Property Name	Type	Description
TableFontFamily	Text (CSS font family value)	**Optional, default: '' (empty string) = value of setting FontFamily** – Font family for all texts in the table cells. See property TableText and properties TextSource and TextFormat of TableCellDefinition object. The value does not influence the height of the row! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-family.
TableFontSize	Text (CSS font size value)	**Optional, default: '' (empty string) = value of setting FontSize** – Font size for all texts in the table cells. See property TableText and properties TextSource and TextFormat of TableCellDefinition object. The value does not influence the height of the row! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-size. **Note**: If set, do not forget to append a unit like 'px' to the numeric value! Otherwise, the browser will ignore the setting.
TableFontStyle	Text (CSS font style value)	**Optional, default: '' (empty string) = value of setting FontStyle** – Font style for all texts in the table cells. See property TableText and properties TextSource and TextFormat of TableCellDefinition object. The value does not influence the height of the row! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-style
TableFontWeight	Text (CSS font weight value)	**Optional, default: '' (empty string) = value of setting FontWeight** – Font weight for all texts in the table cells. See property TableText and properties TextSource and TextFormat of TableCellDefinition object. The value does not influence the height of the row! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-weight.
TableRowDefinitionID	Text	**Optional, default: '' (empty string) = value of setting DefaultEntityTableRowDefinitionID** –

© 2026 Boyum IT Solutions GmbH

Resource Property Name	Type	Description
		Identifier of a TableRowDefinition object that defines the composition of the table row.
TableSortCode	Text	**Optional, default: '' (empty string)** – If given, then the neighboring objects in the table in same level are sorted ascending using this value. See setting ResourceSortCodeSources.
TableText	Text	**Optional, default: '' (empty string)** – Text to show in the table. **Note**: Several immediately consecutive spaces are always combined into one space by the browsers. If the individual spaces are to be preserved, then each of them must be replaced by the Unicode character \u00A0.
TableTextColor	Text (CSS color value)	**Optional, default: '' (empty string)** – Color for the table row texts. If undefined, a default value of the control add-in will be used.
TooltipText	Text	**Optional, default: '' (empty string)** – Text to show in the tooltip (formatted in HTML). If the text is empty, then no tooltip is shown. **Note**: Several immediately consecutive spaces are always combined into one space by the browsers. If the individual spaces are to be preserved, then each of them must be replaced by the Unicode character \u00A0.
ViewArea	Integer (see enum ViewArea)	**Optional, default: Default** – If set to Top, then the resource and its children are shown in a separate top view area in the resources view. Only settable on resource with no ParentID set. See settings TopViewAreaVisibleInLoadsView/ InResourcesView and MainViewAreaVisibleInLoadsView/InResourcesView.
Visible	Boolean	**Optional, default: True –** If set to False, then this object is hidden in the chart.

© 2026 Boyum IT Solutions GmbH

4.12 Settings

The Settings object is given to the control add-in by using the method SetSettings.

NOTE: The big difference from other data objects is that only the properties filled out in the JsonObject are sent to the control add-in and modified then, while for data objects that are updated missing properties normally are interpreted as being set to the default value.

Settings Property Name	Type	Description
ActivityBarSortModeForStackedRowDesign	Integer (see enum BarSortMode)	**Optional, default: StartAndEnd** – Determines how the activity bars are sorted in a row where the bars are shown vertically stacked, means that they do not overlap each other. When the bars are not shown vertically stacked, then they will be shown sorted by start and end. This means that a bar that starts earlier than another bar is placed behind the latter bar. For bars that start at the same time, the longer bar is placed behind the shorter one.
ActivityBarTopOffsetAndHeightScaleFactor	Decimal > 0.0 and <= 10.0	**Optional, default: 1.0** – This setting modifies both the top offset and the height of the activity bars by the specified factor. Values lower than 1 can help to implement a compact layout.
ActivityBaselineBarsVisible	Boolean	**Optional, default: True** – If set to False, no baseline bars are displayed for the activities.
ActivityCalendarsEnabled	Boolean	**Option, default: False** – If set to True, calendars assigned to activities by setting the activity property CalendarID are displayed in the Activities View.
ActivityHierarchySupplementaryDefinitionID	Text	**Optional (available only in AL version!), default: undefined** – ID of a HierarchySupplementaryDefinition object that will be used to specify grouping parameters for hierarchy of activity objects.
ActivitySortCodeSources	JsonArray of Text	**Optional, default: ["TableSortCode", "ID"]** – Activity rows are sorted by comparing values of the named properties of the activity objects as following: If the compared activities have the same value of the first property, the values of the second one will be compared and so on. If a property name is prefixed by a minus sign, the activity rows are sorted in descending order concerning the value of this property.

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
		Attention: If this setting used, then automatic calculation of sort codes after vertical dragging of an activity row is only possible, when only one property name is provided here!
ActivityTableRowDefinitionIDForTitle		**Deprecated! See renamed setting TableRowDefinitionIDForTitleInActivitiesView.**
AllocationBarDesignOfOtherActivity	Integer (see enum BarDesigns)	**Optional, default: DefaultReduced | Text** – The value is used for allocation bars that are shown in the activities in an activity-specific resource row with a different activity.
AllocationBarDesignOfOtherSkill	Integer (see enum BarDesigns)	**Optional, default: DefaultReduced | Text** – The value is used for allocation bars that are shown in the view type SkilledResourcesView in a skill-specific resource row with a different skill. See also setting DefaultAllocationBarDesign.
AllocationBarSortModeForStackedRowDesign	Integer (see enum BarSortMode)	**Optional, default: StartAndEnd** – Determines how the allocation bars are sorted in a row where the bars are shown vertically stacked, means that they do not overlap each other. When the bars are not shown vertically stacked, then they will be shown sorted by start and end. This means that a bar that starts earlier than another bar is placed behind the latter bar. For bars that start at the same time, the longer bar is placed behind the shorter one.
AllocationBarTopOffsetAndHeightScaleFactor	Decimal > 0 and <= 10	**Optional, default: 1** – This setting modifies both the top offset and the height of the allocation bars in activities view, resources view, and skilled resources view. Values lower than 1 can help to implement a compact layout.
AllocationRowsVisibleInActivitiesView	Boolean	**Optional, default: False** – If set to True, then allocations are shown as own rows below the row of the referenced activity in activities view.
AllocationRowsVisibleInResourcesView	Boolean	**Optional, default: False** – If set to True, then allocations are shown as own rows below the row of the referenced resource in resources view.
AllocationRowsVisibleInSkilledResourcesView	Boolean	**Optional, default: False** – If set to True, then allocations are shown as own rows below the row of the referenced resource in skilled resources view.

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
AllocationSortCodeSources	JsonArray of Text	**Optional, default: ["TableSortCode", "ID"]** – Allocation rows are sorted by comparing values of the named properties of the allocation objects as following: If the compared allocations have the same value of the first property, the values of the second one will be compared and so on. If a property name is prefixed by a minus sign, the allocation rows are sorted in descending order concerning the value of this property.
AllocationsSelectableOnlyOnOneResourceAtATime	Boolean	**Optional, default: False** – If set to True, then only allocations of one resource at a time can be selected.
ApplicationStyleDefinition	Text	**Option, default: '' (empty string)** – If set, then the contained text builds the content of a HTML style placed within the HEAD object inside the DOM. You can use {{?...}} to address application variables, see setting ApplicationVariablesMap. This setting allows it to define CSS variables, see https://developer.mozilla.org/en-US/docs/Web/CSS/Using_CSS_custom_properties. CSS variables can be used instead of a color name within color properties and options.
ApplicationVariablesMap	JsonObject	**Optional, default: empty object** – If set, then the keys serve as variable names in text formatting (see property HTMLFormat of TooltipTemplate objects, property BarTextFormat of Activity and Allocation objects, and property TextFormat of TableCellDefinition objects). The values can be of type string, number, Date, boolean, Object. Every key name must start with a letter and must not contain a dot.
AsynchronousInteractiveTime-AreaStretching	Boolean	**Optional, default: False** – If set to True, rendering the time area will be delayed when the user stretches it by using the zoom out button of the timescale, by using the mouse wheel, or by using the appropriate touch gesture. This can be used to fasten the update behavior in case of diagrams with complex or many bars. That way, the diagram will become more reactive.
AsynchronousRendering	Boolean	**Optional, default: False** – If set to True, then the rows are filled with bars

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
		asynchronously when scrolling vertically or resizing a view.
BarsDraggable	Boolean	**Optional, default: True** – Only when set to True and the setting Editable is set to True, then bars will be draggable according to the value in the property AllowedBarDragModes in Activity and Allocation objects. When this setting or the setting Editable is set to False, then bars are generally not draggable. This setting makes it possible to separate the general ability for bar dragging and row dragging, while the setting Editable controls both at once.
BarSortModeForOptimizedRowDesign	Integer (see enum BarSortMode)	**DEPRECATED! Now there are two settings named ActivityBarSortModeForStackedRowDesign and AllocationBarSortModeForStackedRowDesign. For compatibility reasons, both new settings are set by the control add-in when this deprecated setting is set.**
BottomRowMarginInTimeArea	Integer (> 0)	**Optional, default: 5** – Height of the margin between the bottom row border and bars above in pixels. The value is also used for the vertical margins of curve panes. See also settings TopRowMarginInTimeArea and SubRowDistanceInTimeArea.
BottomStatusAreaContent	Text	**Optional, default: '' (empty string)** – If set and the height is set to a value greater than 0 (see setting BottomStatusAreaHeight), the value is interpreted as HTML and shown below the chart. The HTML content is placed inside a DIV element. It is possible to use variables defined by {{variableName}}. These are replaced by the values provided by the setting StatusAreaVariables. The keyword {{*@symbolID*}} is a placeholder for a defined symbol and can be used in <image src="..."> to show a symbol whenever needed. Alternatively, also standard URLs to external images are possible. If an application wants to get OnClicked and/or OnDoubleClicked events on some

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
		HTML elements, then it should set the ID attribute und add the class name "nevd-tap" or "nevd-doubletap", resp.. If an application wants to show a tooltip on some HTML elements, then it possible add an attribute named “data-vscai-tttid” to it and setting it to the ID of an TooltipTemplate object. Analogously it is possible to use a context menu on some HTML elements: Just add an attribute named “data-vscai-ctxmid” setting it to the ID of a ContextMenu object. When the user opens it and selects an item, then the application gets an OnContextMenuItemClicked event with the property ObjectType set to 'BottomStatusArea' and the property ObjectID set to the ID of the HTML element. This is the same behavior as for the OnClicked event on such HTML elements.
BottomStatusAreaHeight	Integer >= 0	**Optional, default: 0** – If set to a value greater than 0, this defines the height of the bottom status area (see setting BottomStatusAreaContent).
CalendarGridColor	Text (CSS color value)	**Optional, default: '#f0f0f0'** – Specifies a color used to color the vertical stripes representing the nonworking times inside the diagram. If a string is given, then the widget uses the color for all view types.
CalendarGridWeekendColor	Text (CSS color value)	**Optional, default: 'transparent'** – Specifies a color used to color vertical stripes representing the weekend (Saturdays and Sundays) inside the diagram. Because these stripes are drawn directly above the calendar grid (see setting CalendarGridColor), you can use an alpha value for making the color translucent.
CanDragTriggeringEnabled	Boolean	**Optional, default: True** – If set to False, no CanDrag events are triggered. This can help to reduce the number of events sent from the web client to the BC server.
ClickEventTriggeringOnRowInTimeArea	boolean	**Optional, default: False** – When set to True, then the events OnClicked and OnDoubleClicked are triggered on time

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
		area background of a row and in the curve pane area of a row.
CurrentDate	DateTime	**Optional, default: 0DT** – When set to a valid date, then a darkened area is positioned from the timescale start up to this date. The darkened area can be attributed by using the settings PastBackgroundFillColor/ LineColor/ LineWidth/ LineDashArray.
CursorDateLineVisible	Boolean	**Optional, default: False** – If this setting is set to True, an additional labeled date line will follow the mouse cursor.
CurvePanesCollapsibleInResourcesView	Boolean	**Optional, default: True** – Specifies whether the curve panes can be interactively collapsed or expanded.
CurvePanesResizable	Boolean	**Optional, default: False** – If this setting is set to True, curve panes are resizable by dragging and dropping the lower end of the curve pane. See event OnCurvePaneResized and property LoadCurvePaneHeight of Resource objects.
CurvePanesVisibleInActivitiesView	Boolean	**Optional, default: False** – If this setting is set to True, a curve pane is displayed in the ActivitiesView for each activity row. In each pane the curves of the resource first found in an allocation related to the corresponding activity are displayed. Please note: This setting has to be set when instantiating the widget. If it is set later, it has no effect.
DateLineCaptionOptimizedPositioningEnabled	Boolean	**Optional, default: False** – Specifies whether the captions of date lines should be arranged optimized to avoid overlapping.
DateLineGridColor	Text (CSS color value)	**Optional, default: '#505050' (on weekly or daily grid) and '646464' (on automatic grid)** – Color of the date line grid (see also setting DateLineGridMode).
DateLineGridDashArray	Text (SVG stroke dash array value)	**Optional, default: '4,1' (on weekly or automatic grid) and '2,1' (on daily grid)** – Pattern of dashes and gaps for drawing the date line grid. For further information, please see https://www.w3.org/TR/SVG11/painting.html#StrokeDasharrayProperty or https://developer.mozilla.org/en-US/docs/Web/SVG/Attribute/stroke-

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
		dasharray. The value "none" indicates that no dashing is used. In this case, the grid lines are drawn solid (see also setting DateLineGridMode).
DateLineGridMode	Integer (see enum DateLineGridModes)	**Optional, default: Weekly** – This setting determines the distance of the date lines shown. See also settings DateLineGridColor, DateLineGridDashArray, DateLineGridWidth.
DateLineGridWidth	Integer	**Optional, default: 1** – Width of the date line grid in pixels.
DecouplingOfAllocationPropertiesFromActivities	Boolean	**Optional, default: True** – If set to True, then there is no internal fallback of allocation property values of properties Color, NonworkingColor, BorderColor, Progress, ProgressColor, ProgressNonworkingColor from the values of the same properties of the assigned activity. This results in a performance gain when updating activities.
DefaultActivityAllocationRowsCollapsible	Boolean	**Optional, default: True** – This setting holds the default for the property AllocationRowsCollapsible of Activity objects.
DefaultActivityAllowedBarDragModes	Integer (see enum BarDragModes)	**Optional, default: ActivityBarDragModes.DragHorizontally** – This setting holds the default for the property AllowedBarDragModes of Activity objects.
DefaultActivityAllowedRowDragModes	Integer (see enum RowDragModes)	**Optional, default: RowDragModes.None** – This setting holds the default for the property AllowedRowDragModes of Activity objects.
DefaultActivityAttachedDateLineIDs	JsonArray of Text	**Optional, default: empty array** – This option holds the default for the property AttachedDateLineIDs of Activity objects.
DefaultActivityBarDesign	Integer (see enum BarDesigns)	**Optional, default: Default** – This setting determines the default design for activity bars including or excluding entries, complex shape, symbols, status, constraints, baseline, progress, and text (see property BarDesign of Activity objects).
DefaultActivityBarHeight	Integer (≥ 0, ≤ 1000)	**Optional, default: 22** – Default height of the activity bars in pixels. See also Activity.BarHeight.

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
DefaultActivityBarSelectable	Boolean	**Optional, default: True** – This setting holds the default for the property BarSelectable of Activity objects.
DefaultActivityBarShape	Integer (see enum BarShape)	**Optional, default: Regular** – This setting defines which shape should be used by default for the visualization of activity bars.
DefaultActivityBarTooltip-TemplateID	Text	**Optional, default: '' (empty string)** – ID of a TooltipTemplate object that will be used when an activity object has set the property BarTooltipTemplateID to '' (empty string).
DefaultActivityCollapsedRow-Design	Integer (see enum RowDesigns)	**Optional, default: 11** – This setting holds the default for the property CollapsedRowDesign of Activity objects. See explanation for possible values in enumerations chapter.
DefaultActivityConstraint-SymbolColor	Text (CSS color value)	**Optional, default: '#646464'** – Specifies the color used by default for the symbols visualizing the constraint dates (EarliestStart/End, LatestStart/End, MustStart/EndOn).
DefaultActivityExpandedRow-Design	Integer (see enum RowDesigns)	**Optional, default: 11** – This setting holds the default for the property ExpandedRowDesign of Activity objects. See explanation for possible values in enumerations chapter.
DefaultActivityProgress-BackgroundColor	Text (CSS color value)	**Optional, default: 'transparent' –** Color for the background of the progress bar region for activities.
DefaultActivityRowCollapsible	Boolean	**Optional, default: True** – This setting holds the default for the property RowCollapsible of Activity objects.
DefaultActivityRowSelectable	Boolean	**Optional, default: True** – This setting holds the default for the property RowSelectable of Activity objects.
DefaultActivityRowTooltipTemplateID	Text	**Optional, default: '' (empty string)** – ID of a TooltipTemplate object that will be used when an activity object has set the property RowTooltipTemplateID to '' (empty string).
DefaultActivitySnapTargetsForEnd	Integer (see enum SnapTargets)	**Optional, default: 8** – This setting holds the default for the property SnapTargetsForEnd of Activity objects.
DefaultActivitySnapTargetsForStart	Integer (see enum SnapTargets)	**Optional, default: 8** – This setting holds the default for the property SnapTargetsForStart of Activity objects.

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
DefaultActivityStatusFrameColor	Text (CSS color value)	**Optional, default: "red"** – This option holds the default color for the property StatusFrameColor of Activity objects.
DefaultActivityTableRowDefinitionID	Text	**Optional, default: '' (empty string)** – ID of a TableRowDefinition object that will be used when an activity object has set the property TableRowDefinitionID to '' (empty string).
DefaultAllocationAllowedBarDragModes	Integer (see enum BarDragModes)	**Optional, default: DragAutoHorOrVer** – This setting holds the default for the property AllowedBarDragModes of Allocation objects. See also setting ForcedAllocationAllowedBarDragModes.
DefaultAllocationAllowedBarDragModesInActivitiesView	Integer (see enum BarDragModes)	**Optional, default: DragHorizontally** – This option holds the default for the property AllowedBarDragModesIn-AllocationView of Allocation objects. See also setting ForcedAllocationAllowed-BarDragModesInActivitiesView.
DefaultAllocationAllowedRow-DragModes	Integer (see enum RowDragModes)	**Optional, default: None** – This setting holds the default for the property AllowedRowDragModes of Allocation objects.
DefaultAllocationAllowedRow-DragModesInActivitiesView	Integer (see enum RowDragModes)	**Optional, default: None** – This setting holds the default for the property AllowedRowDragModesInActivitiesView of Allocation objects.
DefaultAllocationAttachedDateLineIDs	JsonArray of Text	**Optional, default: empty array** – This option holds the default for the property AttachedDateLineIDs of Allocation objects.
DefaultAllocationBarDesign	Integer (see enum BarDesigns)	**Optional, default: Default** – This setting determines the default design for allocation bars including or excluding entries, complex shape, symbols, status, constraints, progress, and text (see property BarDesign of Allocation objects).
DefaultAllocationBarHeight	Integer (≥ 0, ≤ 1000)	**Optional, default: 22** – Default height of the allocation bars in pixels. See also Allocation.BarHeight.
DefaultAllocationBarSelectable	Boolean	**Optional, default: True** – This setting holds the default for the property BarSelectable of Allocation objects.
DefaultAllocationBarShape	Integer (see enum BarShape)	**Optional, default: Regular** – This setting defines which shape should be used by default for the visualization of allocation bars.
DefaultAllocationBarTextFormat	Text	**Optional, default: '' (empty string)** – This setting holds the default value for the property BarTextFormat of Allocation

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
		objects. See there for an explanation of the format.
DefaultAllocationBarTooltipTemplateID	Text	Optional, default: **'' (empty string)** – ID of a TooltipTemplate object that will be used when an allocation object has set the property BarTooltipTemplateID to '' (empty string).
DefaultAllocationConstraintSymbolColor	Text (CSS color value)	**Optional, default: '#646464'** – Specifies the color used by default for the symbols visualizing the constraint dates (EarliestStart/End, LatestStart/End, MustStart/EndOn).
DefaultAllocationMinimumRowHeight	Integer	**Optional, default: 42** – Default minimum height of the allocation rows in pixels. See also Allocation.MinimumRowHeight.
DefaultAllocationProgressBackgroundColor	Text (CSS color value)	**Optional, default: 'transparent'** – Color for the background of the progress bar region for allocations.
DefaultAllocationRowDesign	Integer (see enum RowDesigns)	**Optional, default: 9** – This setting holds the default for the property RowDesign of Allocation objects. See explanation for possible values in enumerations chapter.
DefaultAllocationRowSelectable	Boolean	**Optional, default: True** – This setting holds the default for the property RowSelectable of Allocation objects.
DefaultAllocationRowTooltipTemplateID	Text	**Optional, default: '' (empty string)** – ID of a TooltipTemplate object that will be used when an allocation object has set the property RowTooltipTemplateID to '' (empty string).
DefaultAllocationSnapTargetsForEnd	Integer (see enum SnapTargets)	**Optional, default: 11** – This setting holds the default for the property SnapTargetsForEnd of Allocation objects.
DefaultAllocationSnapTargetsForStart	Integer (see enum SnapTargets)	**Optional, default: 11** – This setting holds the default for the property SnapTargetsForStart of Allocation objects.
DefaultAllocationStatusFrameColor	Text (CSS color value)	**Optional, default: "red"** – This option holds the default color for the property StatusFrameColor of Allocation objects.
DefaultAllocationTableRowDefinitionID	Text	**Optional, default: one table cell with value of property TableText of the referenced allocation object in it** – ID of a TableRowDefinition object that will be used in allocation rows when an allocation object has set the property TableRowDefinitionID to "". See setting AllocationRowsVisibleInActivitiesView.

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
DefaultCalendarID	Text	**Optional, default: '' (empty string)** – Specifies a default calendar to be used in the control add-in. If calendars are defined on activities or resource they will override this calendar. If there is no calendar defined on an activity or a resource and if this default calendar ID is undefined, then the calendar is assumed to be one with constantly non-working time only.
DefaultEntityAllowedRowDragModes	Integer (see enum RowDragModes)	**Optional, default: RowDragModes.DragOutside** – This setting holds the default for the property AllowedRowDragModes of Entity objects.
DefaultEntityRowCollapsible	Boolean	**Optional, default: True** – This setting holds the default for the property RowCollapsible of Entity objects.
DefaultEntityRowSelectable	Boolean	**Optional, default: True** – This option holds the default for the property RowSelectable of Entity objects.
DefaultEntityRowTooltipTemplateID	Text	**Optional, default: '' (empty string)** – ID of a TooltipTemplate object that will be used when an entity object has set the property RowTooltipTemplateID to '' (empty string).
DefaultEntityTableRowDefinitionID	Text	**Optional, default: '' (empty string)** – ID of a TableRowDefinition object that will be used when an entity object has set the property TableRowDefinitionID to '' (empty string).
DefaultLinkRoutingType	Integer (see enum LinkRoutingType)	**Option, default LinkRoutingType.Curved** – This setting holds the default for the property RoutingType of Links objects.
DefaultLinkSelectable	Boolean	**Optional, default: False** – This setting holds the default for the property Selectable of Link objects.
DefaultLinkTooltipTemplateID	Text	**Optional, default: '' (empty string)** – ID of a TooltipTemplate object that will be used when a link object has set the property TooltipTemplateID to '' (empty string).
DefaultMinimumActivityRow-Height	Integer (> 0)	**Optional, default: 42** – Default minimum height of the activity rows in pixels. See also Activity.MinimumRowHeight.
DefaultMinimumEntityRow-Height	Integer (> 0)	**Optional, default: 42** – Default minimum height of the entity rows in pixels. See also Entity.MinimumRowHeight.
DefaultResourceAllowedRow-DragModes	Integer (see enum RowDragModes)	**Optional, default: RowDragModes.None** – This setting holds the default for the

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
		property AllowedRowDragModes of Resource objects.
DefaultResourceBarsPaneVisible	Boolean	**Optional, default: True** – If set to false, then bars are not visible by default by hiding the pane in the time area where the bars are drawn.
DefaultResourceCollapsed-RowDesign	Integer (see enum RowDesigns)	**Optional, default: 11** – This setting holds the default for the property CollapsedRowDesign of Resource objects. See explanation for possible values in enumerations chapter.
DefaultResourceCurveTooltipTemplateID	Text	**Optional, default: '' (empty string)** – ID of a TooltipTemplate object that will be used when a resource object has set the property CurveTooltipTemplateID to '' (empty string).
DefaultResourceExpanded-RowDesign	Integer (see enum RowDesigns)	**Optional, default: 11** – This setting holds the default for the property ExpandedRowDesign of Resource objects. See explanation for possible values in enumerations chapter.
DefaultResourceInventoryCurvePaneColor	Text (CSS color value)	**Optional, default: 'rgba(213,163,42,0.13)'** – Color for the background of the inventory curve pane.
DefaultResourceInventoryCurvePaneHeight	Integer (> 0)	**Optional, default: 50** – Default value for property InventoryCurvePaneHeight of Resource object. See also the settings minimumResourceInventoryCurvePaneHeight and maximumResourceInventoryCurvePaneHeight.
DefaultResourceInventoryCurvePaneVisible	Boolean	**Optional, default: False** – If set to true, then the inventory curves are visible by default.
DefaultResourceLoadCurvePaneColor	Text (CSS color value)	**Optional, default: 'rgba(43,86,158,0.2)'** – Color for the background of the load curve pane.
DefaultResourceLoadCurvePaneHeight	Integer (> 0)	**Optional, default: 50** – Default value for property LoadCurvePaneHeight of Resource objects. See also settings MinimumResourceLoadPaneCurveHeight and MaximumResourceLoadPaneCurveHeight.

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
DefaultResourceLoadCurvePaneVisible	Boolean	**Optional, default: True** – If set to false, then the load and capacity curves are not visible by default.
DefaultResourceMinimum-RowHeight	Integer (> 0)	**Optional, default: 42** – Default minimum height of the resource rows in pixels. See also Resource.MinimumRowHeight.
DefaultResourceRowCollapsible	Boolean	**Optional, default: True** – This option holds the default for the property RowCollapsible of Resource objects.
DefaultResourceRowSelectable	Boolean	**Optional, default: True** – This setting holds the default for the property RowSelectable of Resource objects.
DefaultResourceRowTooltipTemplateID	Text	**Optional, default: '' (empty string)** – ID of a TooltipTemplate object that will be used when a resource object has set the property RowTooltipTemplateID to '' (empty string).
DefaultResourceTableRowDefinitionID	Text	**Optional, default: '' (empty string)** – ID of a TableRowDefinition object that will be used when a resource object has set the property TableRowDefinitionID to empty string.
DefaultResourceTableRowDefinitionIDInActivitiesView	Text	**Optional, default: value of setting DefaultResourceTableRowDefinitionID** – ID of a TableRowDefinition object that will be used when a resource object has set the property TableRowDefinitionID to empty string in the activities view.
DefaultSkillAllowedRowDragModes	Integer (see enum RowDragModes)	**Optional, default: RowDragModes.None** – This setting holds the default for the property AllowedRowDragModes of Skill objects.
DefaultSkillCollapsedRowDesign	Integer (see enum RowDesigns)	**Optional, default: RowDesigns.Empty** – This option holds the default for the property CollapsedRowDesign of Skill objects. Only the flags BarsOfHiddenDescendantRows and BarsStacked are processed.
DefaultSkilledAllocationBarTooltipTemplateID	Text	**Optional, default: '' (empty string)** – ID of a TooltipTemplate object that will be used when an allocation object has set the property SkilledBarTooltipTemplateID to "".
DefaultSkilledAllocationRowTooltipTemplateID	Text	**Optional, default: '' (empty string)** – ID of a TooltipTemplate object that will be used when an allocation object has set the property SkilledRowTooltipTemplateID to "".
DefaultSkilledResourceRowTooltipTemplateID	Text	**Optional, default: '' (empty string)** – ID of a TooltipTemplate object that will be used when an resource object has set

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
		the property SkilledRowTooltipTemplateID to "".
DefaultSkillMinimumRowHeight	Integer	**Optional, default: 42** – Default minimum height of the skill rows in pixels. See also Resource.MinimumRowHeight.
DefaultSkillRowCollapsible	Boolean	**Optional, default: True** – This setting holds the default for the property RowCollapsible of Skill objects.
DefaultSkillRowSelectable	Boolean	**Optional, default: True** – This setting holds the default for the property RowSelectable of Skill objects.
DefaultSkillRowTooltipTemplateID	Text	**Optional, default: '' (empty string)** – ID of a TooltipTemplate object that will be used when a Skill object has set the property RowTooltipTemplateID to "".
DefaultSkillTableRowDefinitionID	Text	**Optional, default: one table cell with value of property TableText of the referenced Skill object in it** – ID of a TableRowDefinition object that will be used when a Skill object has set the property TableRowDefinitionID to ''.
DefaultValuesForActivityEntryProperties	JsonObject	**Optional, default: null** – Specifies an object with default values used for all properties of ActivityEntry objects.
DefaultValuesForActivityProperties	JsonObject	**Optional, default: null** – Specifies an object with default values used for all properties of Activity objects except ID.
DefaultValuesForAllocationEntryProperties	JsonObject	**Optional, default: null** – Specifies an object with default values used for all properties of AllocationEntry objects.
DefaultValuesForAllocationProperties	JsonObject	**Optional, default: null** – Specifies an object with default values used for all properties of Allocation objects except ID.
DefaultValuesForEntityProperties	JsonObject	**Optional, default: null** – Specifies an object with default values used for all properties of Entity objects except ID.
DefaultValuesForLinkProperties	JsonObject	**Optional, default: null** – Specifies an object with default values used for all properties of Link objects except ID.
DefaultValuesForResourceProperties	JsonObject	**Optional, default: null** – Specifies an object with default values used for all properties of Resource objects except ID.
DefaultValuesForSkillProperties	JsonObject	**Optional, default: null** – Specifies an object with default values used for all properties of Skill objects except ID.
DefinedAllocationLinksVisibleInAcrtivitiesView	Boolean	**Optional, default: False** – If set to True and the settings LinksVisibleInActivities-View and AllocationRowsVisibleInActivitiesView are also True, then links that are defined

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
		between allocations are shown additionally.
DefinedAllocationLinksVisibleInResourcesView	Boolean	**Optional, default: False** – If set to True and the setting LinksVisibleInResources-View is also True, then links that are defined between allocations are shown instead of calculated allocation links defined by activity links.
DefinedAllocationLinksVisibleInSkilledResourcesView	Boolean	**Optional, default: False** – If set to True and the setting LinksVisibleInSkilledResourcesView is also True, then links that are defined between allocations are shown instead of calculated allocation links defined by activity links.
DetailedActivityConstraintSymbolsEnabled	Boolean	**Optional, default: True** – If set to True, there will be shown different symbols for the constraint dates depending on their constraint types: • EarliestStart:  • LatestStart:  • MustStartOn:  • EarliestEnd:  • LatestEnd:  • MustEndOn:  Otherwise, a simple down arrow will be shown: . Please consider to set the setting TopRowMarginInTimeArea when using detailed symbols.
DetailedAllocationConstraintSymbolsEnabled	Boolean	**Optional, default: True** – If set to True, there will be shown different symbols for the constraint dates depending on their constraint types: • EarliestStart:  • LatestStart:  • MustStartOn:  • EarliestEnd:  • LatestEnd:  • MustEndOn:  Otherwise, a simple down arrow will be shown: . Please consider to set the setting TopRowMarginInTimeArea when using detailed symbols.

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
DragDatesLimitingInteraction	Boolean	**Option, default: False** – If set to True, then bars cannot be dragged before the value in the property EarliestDragStart and later than LatestDragEnd, respectively.
Editable	Boolean	**Optional, default: True** – If set to False, nothing can be edited. Currently this only controls the ability to drag bars and rows. See also settings BarsDraggable and RowsDraggable.
End	DateTime	**Optional, default: actual value of option start plus 4 weeks** – Date where the time area ends absolutely, means when the horizontal scroll bar is placed at the end of the time area view. When using the method *FitTimeAreaIntoView*, it is not possible there to make dates visible beyond the end date set here. The end date itself is not included in the visible time area. The same behavior also is true for all end dates used in the widget like e.g. in calendar entries, period highlighter entries, activities, activity entries, allocations, allocation entries. If this setting was not set before the first rendering, then a warning is triggered (see event OnLogWarnings). It is strongly recommended to set start and end together in a single settings object. This way VS-CAI reacts faster.
EntitiesTableColumnSeparatorColor	Text (CSS color value)	**Optional, default: '#a0a0a0'** – Specifies a color used to color the column separators in the entities table. If a string is given, then the widget uses the color for all view types.
EntitiesTableHeaderBackgroundColor		**Deprecated! Renamed to EntitiesTableTitleBackgroundColor.**
EntitiesTableHeaderColumnSeparatorColor		**Deprecated! Renamed to EntitiesTableTitleColumnSeparatorColor.**
EntitiesTableHeaderHighlightingColor		**Deprecated! Renamed to EntitiesTableTitleHighlightingColor.**
EntitiesTableHeaderTextColor		**Deprecated! Renamed to EntitiesTableTextColor.**
EntitiesTableSymbolColumnBackgroundColor	Text (CSS color value)	**Optional, default: 'white'** – If set then the symbol column of the entities table will show this color in the background.

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
EntitiesTableSymbolColumnTitleBackgroundColor	Text (CSS color value)	**Optional, default: value of setting EntitiesTableSymbolColumnBackgroundColor** – If set then the symbol column title of the entities table will show this color in the background when the setting EntitiesTableSymbolColumnTitleVisible is set to true.
EntitiesTableSymbolColumnTitleSymbolIDs	JsonArray of Text	**Optional, default: empty array** – Array of identifiers of the symbols to be shown in the entities table in the title cell of the symbol column. They will only appear when the setting EntitiesTable-SymbolColumnTitleVisible is set to True. The symbols will be arranged one below the other. However, if the cell is not high enough to hold all symbols, then the remaining symbols are also arranged side-by-side. If this still does not fit, an additional “show more” symbol will be displayed. An empty string ('') will cause an “empty” symbol to be displayed. By this placeholder, you can reserve space for a symbol that may be shown later. Please note: Each symbol will be resized to an image with a width and height of 16 pixels each at a zoom level of 100%.
EntitiesTableSymbolColumnTitleVisible	Boolean	**Optional, default: False** – If set to True, the symbols specified in the setting EntitiesTableSymbolColumnTitleSymbolIDs will be displayed in the title cell of the symbol column. Otherwise, the title cell will have the same color as defined in the setting Entities-TableHeaderBackgroundColor.
EntitiesTableSymbolColumnVisible	Boolean	**Optional, default: False** – If set to True, a special column at the left of the entities table will be displayed to show the row symbols of the entities.
EntitiesTableSymbolColumn-Width	Integer	**Optional, default: 22** – Width of the symbol column in the entities table in pixels. If set to a value less than the default, it will be set to the default automatically.
EntitiesTableTitleBackgroundColor	Text (CSS color value)	**Optional, default: '#646464'** – Specifies a color used to color the background of the entities table header. If a string is

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
		given, then the widget uses the color for all view types.
EntitiesTableTitleColumn-SeparatorColor	Text (CSS color value)	**Optional, default: 'white'** – Specifies a color used to color the column separators in the entities table header. If a string is given, then the widget uses the color for all view types.
EntitiesTableTitleHighlighting-Color	Text (CSS color value)	**Optional, default: '#f7c365'** – Specifies the color to be used during the interaction, e.g. to highlight the separation line between two adjacent columns when altering the column widths.
EntitiesTableTitleText	Text	**Optional, default: '' (empty string)** – This text will be shown in the table header of the entities table. It will appear only when not using TableRowDefinition objects or the setting TableRowDefinitionIDForTitleInEnetititiesTable is not set. **Note**: Several immediately consecutive spaces are always combined into one space by the browsers. If the individual spaces are to be preserved, then each of them must be replaced by the Unicode character \u00A0.
EntitiesTableTitleTextColor	Text (CSS color value)	**Optional, default: 'white'** – Specifies a color used to color the text in the entities table header. If a string is given, then the widget uses the color for all view types.
EntitiesTableTreeViewLineColor	Text (CSS color value)	**Optional, default: 'black'** – Determines the color of tree view lines in the entities table. See setting EntitiesTableTreeVisualizationMode.
EntitiesTableTreeViewLine-DashArray	Text (SVG stroke dash array value)	**Optional, default: '' (empty string)** – Pattern of dashes and gaps for drawing the tree view lines in the entities table. For further information, please see https://www.w3.org/TR/SVG11/painting.html#StrokeDasharrayProperty or https://developer.mozilla.org/en-US/docs/Web/SVG/Attribute/stroke-dasharray. The value 'none' indicates that no dashing is used. In this case, the line is drawn solid.

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
		See setting EntitiesTableTreeVisualizationMode.
EntitiesTableTreeVisualizationMode	Integer (see enum TreeVisualizationMode)	**Optional, default: ColoredIndentation** – Determines how the tree of objects is visualized in the entities table. See also settings EntitiesTableTreeViewLineColor and EntitiesTableTreeViewLineDashArray.
EntitiesTableViewWidth	Integer	**Optional, default: -1** – This setting defines the width of the entities table view in pixels. When set to -1, then the control add-in determines the view width automatically.
EntitiesTableVisibleInActivitiesView	Boolean	**Optional, default: False –** This setting lets appear/disappear the entities table on the right side in the Activities View.
EntitiesTableVisibleInResourcesView	Boolean	**Optional, default: False –** This setting lets appear/disappear the entities table on the right side in the resources view.
EntitiesTableVisibleInSkilledResourcesView	Boolean	**Optional, default: False –** This setting lets appear/disappear the entities table on the right side in the skilled resources view.
EntitiesTableTitleFontFamily	Text (CSS font family value)	**Optional, default: value of setting FontFamily** – Font family for all texts in the entities table title. See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-family.
EntitiesTableTitleFontSize	Text (CSS font size value)	**Optional, default: value of setting FontSize** – Font size for all texts in the entities table title. See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-size. **Note**: If set, do not forget to append a unit like 'px' to the numeric value! Otherwise, the browser will ignore the setting.
EntitiesTableTitleFontStyle	Text (CSS font style value)	**Optional, default: value of setting FontStyle** – Font style for all texts in the entities table title. See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-style

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
EntitiesTableTitleFontWeight	Text (CSS font weight value)	**Optional, default: value of setting FontWeight** – Font weight for all texts in the entities table title. See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-weight.
EntitiesTableWidth	Integer	**Optional, default: -1** – Not recommended when using TableRowDefinition objects! This setting defines the width of the entities table. -1 leaves the setting unchanged.
EntitiesTitleText	Text	**Deprecated! See renamed setting EntitiesTableTitleText.**
EntityHierarchy-SupplementaryDefinitionID	Text	**Optional (available only for AL version), default: undefined** – ID of a HierarchySupplementaryDefinition object that will be used to specify grouping parameters for hierarchy of entity objects.
EntitySortCodeSources	JsonArray of Text	**Optional, default: ["TableSortCode", "ID"]** – Entity rows are sorted by comparing values of the named properties of the entity objects as following: If the compared entities have the same value of the first property, the values of the second one will be compared and so on. If a property name is prefixed by a minus sign, the entity rows are sorted in descending order concerning the value of this property. Attention: If this setting used, then automatic calculation of sort codes after vertical dragging of an entity row is only possible, when only one property name is provided here!
EntityTableRowDefinitionIDForTitle		**Deprecated! See renamed setting TableRowDefinitionIDForTitleInEntitiesTable.**
FinishedAllocationBarsShownUnstackedInBackground	Boolean	**Optional, default: False** – If set to True, then allocation bars with progress set to the value 100 are shown without vertical stacking (see BarsStacked value in @Enum.RowDesigns) and optically behind other bars.
FirstDayOfWeek	Integer (see enum DayOfWeek)	**Optional, default: -1** – Specifies the first day of a week.

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
		If set unequal -1, this value overwrites the implicit values of the settings WeekNumbering and LocaleId, respectively.
FontFamily	Text (CSS font family value)	**Optional, default: inherited value from BC web client** – Font family for all texts in bars, table cells, date lines, and timescale. See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-family and the setting ApplicationStyleDefinition. Since the value is set without any conversion to the appropriate DOM representation, the application can use all the possibilities that the browser supports: single font names, font names with fallbacks, and CSS custom properties. It is recommended to use a value with fallback names comma-separated in respect to the different operating systems, e.g. '`"Segoe UI", Verdana, Geneva, sans-serif`'. If you want to use the method saveAsPDF: If you set any font family properties within CSS or inside options or data objects, please ensure to include a fallback to one of the standard fonts "serif", "sans serif", or "monospace" that all browsers guarantee to work. The PDF will be generated using these fallback names, else "sans serif" is used. The fonts used internally do not know many characters besides the common letters, numbers, and standard characters!
FontSize	Text (CSS font size value)	**Optional, default: inherited value from BC web client** – Font size for all texts in bars (not outside texts), table cells, and date lines. See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-size. **Note**: If set, do not forget to append a unit like 'px' to the numeric value! Otherwise, the browser will ignore the setting.

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
FontStyle	Text (CSS font style value)	**Optional, default: inherited value from BC web client** – Font style for all texts in bars, table cells, date lines and timescale. See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-style
FontWeight	Text (CSS font weight value)	**Optional, default: inherited value from BC web client** – Font weight for all texts in bars, table cells, date lines and timescale. See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-weight.
ForcedActivityAllowedBar-DragModes	Integer (see enum BarDragModes)	**Optional, default: -1** – If set to a number >= 0, then this value overrides any setting in DefaultActivityAllowedBarDragModes and Activity property AllowedBarDragModes. See also event CanDrag.
ForcedActivityAllowedRow-DragModes	Integer (see enum RowDragModes)	**Optional, default: -1** – If set to a number >= 0, then this value overrides any value in setting DefaultActivityAllowedRow-DragModes and Activity property AllowedRowDragModes. See also event CanDrag.
ForcedAllocationAllowedBar-DragModes	Integer (see enum BarDragModes)	**Optional, default: -1** – If set to a number >= 0, then this value overrides any setting in DefaultAllocationAllowedBar-DragModes and Allocation property AllowedBarDragModes. See also event CanDrag.
ForcedAllocationAllowedBar-DragModesInActivitiesView	Integer (see enum BarDragModes)	**Optional, default: -1** – If set to a number >= 0, then this value overrides any setting DefaultAllocationAllowedBar-DragModesInActivitiesView, Allocation property AllowedBarDragModesInActivitiesView. See also event CanDrag.
ForcedAllocationAllowedRow-DragModes	Integer (see enum RowDragModes)	**Optional, default: -1** – If set to a number >= 0, then this value overrides any value in setting DefaultAllocationAllowedRow-DragModes and Allocation property AllowedRowDragModes. See also event CanDrag.
ForcedAllocationAllowedRow-DragModesInActivitiesView	Integer (see enum RowDragModes)	**Optional, default: -1** – If set to a number >= 0, then this value overrides any value in setting DefaultAllocationAllowedRow-DragModesInActivitiesView and

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
		Allocation property AllowedRowDragModesInActivitiesView. See also event CanDrag.
ForcedEntityAllowedRow-DragModes	Integer (see enum RowDragModes)	**Optional, default: -1** – If set to a number >= 0, then this value overrides any value in setting DefaultEntityAllowedRow-DragModes and Entity property AllowedRowDragModes. See also event CanDrag.
ForcedFillOfPageSpace		**Deprecated! Value is ignored.**
ForcedResourceAllowedRow-DragModes	Integer (see enum RowDragModes)	**Optional, default: -1** – If set to a number >= 0, then this value overrides any value in setting DefaultResourceAllowedRow-DragModes and Resource property AllowedRowDragModes. See also event CanDrag.
IgnoreCalendarOnActivityBar-Interactions	Boolean	**Optional, default: False** – If set to True, then the activity calendar is not taken into account when dragging an activity bar.
IgnoreCalendarOnAllocation-BarInteractions	Boolean	**Optional, default: False** – If set to True, then the resource calendar is not taken into account when dragging an allocation bar.
InteractiveActivationOfLoggingEnabled	Boolean	**Optional, default: False** – If set to True, the user can activate the logging by using the keyboard shortcut Shift+Ctrl+Alt+L. (Before using this shortcut, the user has to ensure that the widget has the focus, e.g. by clicking with the mouse pointer in an empty area of the chart.) The record symbol will appear, the current state of the widget is saved and from then on, all calls to the API are recorded. Pressing Shift+Ctrl+Alt+L once again will stop the recording and download a file with the recorded data. See also setting **LoggingEnabled**.
InteractiveSwitchingOfSortOrderEnabled	Boolean	**Optional, default: False –** If set to True, the user can change the sort order of the rows by clicking or tapping into a column of the table title. When unsorted or sorted by another column, then the sort order is expected to change to ascending. If sorted ascending already, it is expected to change to descending. When sorted descending already, it is expected to change to be unsorted again. The latter mode change can be switched to ascending by using the

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
		setting RowSortModeNoneEnabledOnInteractiveSwitchingOfSortOrder. The requested mode change triggers the event OnRowSortingChangeRequested. The application will have to change the appropriate sorting settings itself to set a realize a changed sort mode (see settings Activity/Allocation/Entity/ Resource/SkillRowSortCodeSources). The current sorting can be made visible by setting the setting SortingIndicatorVisible to True additionally.
IntlDateTimeFormatOptionsMap	JsonObject	**Optional, default: empty object** – If set, then the content is used to define options for objects of type Intl.DateTimeFormat (see https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Intl/DateTimeFormat). The properties/keys define names for a format, the options of which are defined as value. The value therefore is another object (description see second parameter of constructor function of Intl.DateTimeFormat). The format names can be used in the properties TableCellDefinition.TextFormat, Activity/Allocation.BarTextFormat, TooltipTemplate.HTMLFormat, and the settings DefaultActivity/AllocationBarTextFormat. For the case that no format name is defined or it cannot be defined (this is the case for dates shown while dragging or numbers in the scale of curves), you can define a format with the name "default". This is then used in these cases. Every format name must start with a letter and must not contain a dot. The options should not contain a property named timeZone, since this is filled in by the VSW itself using the value of the setting TimeZone. Also, the first parameter for the constructor of Intl.DateTimeFormat objects is filled with

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
		the converted value of the setting LocaleId.
IntlNumberFormatOptionsMap	JsonObject	**Optional, default: empty object** – If set, then the content is used to define options for objects of type Intl.NumberFormat (see https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Intl/NumberFormat). The properties/keys define names for a format, the options of which are defined as value. The value therefore is another object (description see second parameter of constructor function of Intl.NumberFormat). The format names can be used in the properties TableCellDefinition.TextFormat, Activity/Allocation.BarTextFormat, TooltipTemplate.HTMLFormat, and the options defaultActivity/AllocationBarTextFormat. For the case that no format name is defined or it cannot be defined (this is the case for dates shown while dragging or numbers in the scale of curves), you can define a format with the name "default". This is then used in these cases. Every format name must start with a letter and must not contain a dot. The first parameter for the constructor of Intl.NumberFormat objects is filled with the converted value of the setting LocaleId.
LanguageId	Text (possible values: see LocaleId)	**Optional, default: value of setting LocaleId –** This setting will be used for showing regional texts in the timescale. For an example see setting LocaleId.
LicenseKey	Text	**Optional, default: '' (empty string)** – The license key has to be set on instantiation of the control add-in. The key itself is provided by Boyum IT. If the key is not set, not valid, or expired, this will be shown in a message on the screen.
LinesShownInLoadCurvePanes	Boolean	**Optional, default: False** – If set to True, in all load curve panes horizontal auxiliary lines are displayed for each tick mark of the numerical scales in the table. These lines help the user to read the curve values.

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
LinksVisibleInActivitiesView	Boolean	**Optional, default: True** – If set to False, the activities view does not show links. When True, it shows at least activity links. See also setting Defined-AllocationLinksVisibleInActivitiesView.
LinksVisibleInResourcesView	Boolean	**Optional, default: False** – If set to True, the resources view shows links. See also setting DefinedAllocationLinks-VisibleInResourcesView.
LinksVisibleInSkilledResourcesView	Boolean	**Optional, default: False** – If set to True, the skilled resources view shows links. See also setting DefinedAllocationLinks-VisibleInSkilledResourcesView.
LinksWithDanglingStartOrEndVisible	Boolean	**Optional, default: False** – If set to True, links are visible even if their source or target bar is invisible due to filtering or collapsed rows. In this case, the dangling start or end of the link retains its position in time. False:  True:  Additionally, links then are even visible when the object of their source or target bar is not added to the data model. In this case, the dangling start or end of the link is drawn vertically. True: 
LocaleId	Text (possible values: '1030' (da-DK), '1031' (de-DE), '2057' (en-GB), '1033' (en-US), '1034' (es-ES), '3082' (es-ES),	**Optional, default: '1033'** – This setting will be used for formatting date and time values in the timescale and numbers in the numeric scales of curves. Sample for AL (including LanguageId and TimeZone settings): `var _settings: JsonObject;`

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description

	'1035' (fi-FI), '1036' (fr-FR), '1040' (it-IT), '1041' (ja-JP), '1043' (nl-NL), '1044' (nb-NO), '1045' (pl-PL), '1046' (pt-PT), '2070' (pt-BR), '1049' (ru-RU), '1053' (sv-SE), '1054' (th-TH), '2052' (zh-CN)	`_sessionSettings: SessionSettings;`
		
		`begin`
		
		`_sessionSettings.Init();`
		…
		`_settings.Add('LanguageId', _sessionSettings.LanguageId());`
		`_settings.Add('LocaleId', _sessionSettings.LocaleId());`
		`_settings.Add('TimeZone', _sessionSettings.TimeZone());`
		…
		
		`CurrPage.conVSControlAddIn.SetSettings(_settings);`
		`end;`

LoggingEnabled	Boolean	**Optional, default: False** – If set to True, the record symbol will appear, the current state of the widget is saved and from then on all calls to the API are recorded. Resetting this setting to False will stop the recording and download a file with the recorded data. See also settings **InteractiveActivationOfLoggingEnabled** and **LoggingVerboseLevel**.
LoggingFileCompressionEnabled	boolean	**Optional, default: False** – If set to True, then the log file will automatically be written (aka downloaded) as a ZIP file containing the raw log file as a single file. This serves as a comfortable option e.g. to send this ZIP file directly by mail.
LoggingVerboseLevel	number	**Optional, default: 0** – The default verbose level for logging omits loggings of callback triggerings that occur often. If set to 10, then the callbacks canDrag and onShowTooltip are logged additionally. If set to 30, also the callbacks visibilityFilter and compareObjects are logged.
MainViewAreaVisible		**Deprecated! Renamed to MainViewAreaVisibleInResourcesView.**
MainViewAreaVisibleInActivitiesView	Boolean	**Optional, default: True** – When set to False, then in activities view the main view area is invisible. The main view area contains the rows for activities with ViewArea set to Default. If TopViewAreaVisibleInActivitiesView is also False, then the main view area will be visible nevertheless.
MainViewAreaVisibleInLoadsView	Boolean	**Optional, default: True** – When set to False, then in loads view the main view area is invisible. The main view area

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
		contains the rows for resources with ViewArea set to Default. If TopViewAreaVisibleInLoadsView is also False, then the main view area will be visible nevertheless.
MainViewAreaVisibleInResourcesView	Boolean	**Optional, default: True** – When set to False, then in resources view the main view area is invisible. The main view area contains the rows for resources with ViewArea set to Default. If TopViewAreaVisibleInResourcesView is also False, then the main view area will be visible nevertheless.
MainViewAreaVisibleInSkilledResourcesView	Boolean	**Optional, default: True** – When set to False, then in skilled resources view the main view area is invisible. The main view area contains the rows for skills with ViewArea set to Default. If TopViewAreaVisibleInSkilledResourcesView is also False, then the main view area will be visible nevertheless.
MaximumResourceInventoryCurvePaneHeight	Integer (> 0)	**Optional, default: 200** – Value for maximum of property InventoryCurvePaneHeight of Resource objects.
MaximumResourceLoadCurvePaneHeight	Integer (> 0)	**Optional, default: 200** – Default maximum height of curve panes when changed interactively, see setting CurvePanesResizable.
MaximumSnapDistance	Integer (>= 0)	**Optional, default: 8** – Maximum distance in pixels of a currently dragged bar to a snap target, within which a dragged bar will get snapped to the snap target.
MaximumTimeResolutionUnit	Text (one of 'second', 'minute', 'hour', 'day', 'week', 'month', 'quarter', 'year')	**Optional, default: value of setting TimeStepUnit or 'second'** – Unit for maximum time resolution in the time area. Used together with setting MaximumTimeResolutionUnitFactor. Neither interactively nor by using the method SetTimeResolutionForView the time area can be show a finer time resolution than defined here. When you set this option and do not set the settings TimeStepUnit/-Factor, the value here also changes the default value of TimeStepUnit! This is done for compatibility reasons.
MaximumTimeResolutionUnitFactor	Integer (>= 1)	**Optional, default: value of setting TimeStepUnitFactor or 1** – Number of

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
		units for maximum time resolution in the time area. Integer values are recommended. Used together with setting MaximumTimeResolutionUnit. When you set this option and do not set the settings TimeStepUnit/-Factor, the value here also changes the default value of TimeStepUnitFactor! This is done for compatibility reasons.
MaximumTopViewArea-HeightRatio	Decimal (-0.8 ≤ n ≤ 0.8)	**Optional, default: 0.5** – If positive, this value determines the maximum height of the top view area expressed as a fraction of the full view height. If negative, the absolute value instead determines the maximum height of the common view area (so the common view area can be used for unassigned resource allocations alternatively. Vertical scroll bars are shown in both view areas if necessary. See also settings TopViewAreaVisibleInActivities/Loads/ResourcesView.
MinimumResourceInventoryCurvePaneHeight	Integer (> 0)	**Optional, default: 30** – Value for interactive minimum of property InventoryCurvePaneHeight of resource objects.
MinimumResourceLoadCurvePaneHeight	Integer (> 0)	**Optional, default: 30** – Default minimum height of curve panes when changed interactively, see setting CurvePanesResizable.
MultipleBarDraggingEnabled	Boolean	**Optional, default: False** – If set to True, all selected bars are dragged at once. Also see events CanDrag, OnDragStart, OnDrop. Currently, the allocation/activity properties EarliestDragStart and LatestDragEnd are not supported when dragging multiple bars. The allocation property SuitableResourceIDs is supported. When dragging starts, the allowed drag modes are taken from the allocation/activity that is dragged directly as default. This is modifiable by using the event CanDrag or one of the settings ForcedActivity/AllocationAllowedBarDragModes.
MultipleSelectionEnabled	Integer (allowed values: 0, 1, 2)	**Optional, default: 1** – If set to 1, then multiple bars or rows can be selected either by clicking on the appropriate object representations while pressing

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
		the CTRL key or by tapping (the CTRL key then is used reversely) or by dragging a rectangle with the mouse. If set to 0, it is only possible to select one bar or row at once and it is possible to pan the table area or the time area by using the mouse. Panning by touch is possible always, see also setting TimeAreaPanningMode. Additionally, if set to 2 instead of 1, then the behavior for dragging a bar selection rectangle is differentiated for a selection left-to-right from right-to-left. If left-to-right, only the bars that are completely inside the rectangle are selected. If right-to-left, all bars that are completely or partially inside the rectangle are selected.
NonworkingTimesCalendarIDs	JsonArray of Text	**Optional, default: empty array** – This setting defines the IDs of calendars considered in the calculation of the common non-working time, means that a non-working time will get invisible only, when all calendars contain it. See setting NonworkingTimeVisible.
NonworkingTimeVisible	Boolean	**Optional, default: True** – This option defines whether the common non-working time is visible. The common time is calculated by all calendar information that are relevant to the visualization. Therefore, the calendars of visible activities and resources are used or alternatively, the IDs of calendars considered are specified via setting NonworkingTimesCalendarIDs.
ObjectHighlightFlashingEnabled	Boolean	**Optional, default: True** – Specifies whether or not the frame displayed around an object that has been scrolled to by using the method scrollToObject should flash or around objects after using the method HighlightObjects.
ObjectHighlightingColor	Text (CSS color value)	**Optional, default: '#7f0000' –** Color of the frame displayed around an object that has been scrolled to by using the method scrollToObject or around objects after using the method HighlightObjects.
OnCollapseStateChanged-TriggeredByUpdateCalls		**Deprecated! Renamed to TriggeringOfOnCollapseStateChangedBy UpdateCallsEnabled.**
OnLogWarningsEventsEnabled	Boolean	**Optional, default: True** – When set to False, then the event OnLogWarnings is not triggered. This is meant for helping

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
		applications, that let the control add-in trigger many of these events in short time. The application developer should create a log file (see setting LoggingEnabled) and remove all warnings by changing the application!
PastBackgroundFillColor	Text	**Optional, default: 'rgba(0,0,0,0.2)'** – This setting defines the color of the darkened area between timescale start and value of the setting CurrentDate.
PastBackgroundLineColor	Text	**Optional, default: 'darkgrey'** – This setting defines the color of the date line at the value of the setting CurrentDate.
PastBackgroundLineDashArray	Text (SVG stroke dash array value)	**Optional, default: '1,1'** – Pattern of dashes and gaps for drawing the date line at the value of the setting CurrentDate. For further information, please see https://www.w3.org/TR/SVG11/painting.html#StrokeDasharrayProperty or https://developer.mozilla.org/en-US/docs/Web/SVG/Attribute/stroke-dasharray. The value 'none' indicates that no dashing is used. In this case, the line is drawn solid.
PastBackgroundLineWidth	Integer	**Optional, default: 1** – This setting defines the width of the date line at the value of the setting CurrentDate.
PingTimeInMilliseconds	Integer (≥0)	**Optional, default: 0** – If set to a value greater than 0, then the event OnPing is triggered in an interval of the given value in milliseconds. If set to 0, then the event OnPing is not triggered. The timer events serve to hold the session active, so it cannot run into timeout.
ProgressBarHeight	Integer >= 0	**Optional, default: 4** – Determines the height of all progress bars within allocation bars and activity bars.
ProgressBarWidthCalculation-Mode	Integer (see ProgressBarWidthCalculationMode)	**Option, default: ProgressBarWidthCalculationMode.ConsiderWorkingTimesOnly** – This setting determines how the widths of the progress bars are calculated. Possible values: • ConsiderWorkingTimesOnly – If this value is used, it is assumed that there is no progress during non-working times.

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
		• ConsiderWorkingAndNonworkingTimes – If this value is used, it is assumed that there is progress during both working and non-working times.
ReducedBarTopOffsetAndHeightScaleFactor	Decimal (> 0, <= 1)	**Optional, default: 1** – This setting is used as a scale factor for bars where the flag ReducedHeight in property BarDesign of an Activity or Allocation object is set to true or Allocation object is set to true or when visualizing allocations of other skill in SkilledResources view (see setting AllocationBarDesignOfOtherSkill).
ReleaseDueDate-ConnectionsVisible	Boolean	**Optional, default: False** – If set to True and an activity has set both a ReleaseDate and a DueDate, a line will be displayed to visually connect both dates: 
ResetValueForDifferentialUpdate	JsonValue* (!) * but not a Date, a DateTime, or a Time	**Optional, default: null** – Determines a value that will be replaced by “undefined” (default value) for differential updates when using an update method with setting UpdateMode set to DifferentialValues.
ResourceHierarchySupplementaryDefinitionID	Text	**Optional, default: '' (empty string)** – ID of a HierarchySupplementaryDefinition object that will be used to specify grouping parameters for hierarchy of resource objects in resources view.
ResourceHierarchySupplementaryDefinitionIDInLoadsView	Text	**Optional, default: '' (empty string)** – ID of a HierarchySupplementaryDefinition object that will be used to specify grouping parameters for hierarchy of resource objects in loads view.
ResourceSortCodeSources	JsonArray of Text	**Optional, default: ["TableSortCode", "ID"]** – Resource rows are sorted by comparing values of the named properties of the resource objects as following: If the compared resources have the same value of the first property, the values of the second one will be compared and so on. If a property name is prefixed by a minus sign, the resource rows are sorted in descending order concerning the value of this property.

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
		Attention: If this setting used, then automatic calculation of sort codes after vertical dragging of a resource row is only possible, when exactly one property name is provided here!
ResourcesVisibleInActivitiesView	Boolean	**Optional, default: False** – If set to true additional to the setting AllocationRowsVisibleInActivitiesView, then the allocation rows in activities view will be shown grouped by the respective assigned resource. The resource rows show all assigned allocations and not only the ones that are assigned to the referenced activity. The application can define the appearance of allocation bars that are assigned to other activities by using the setting AllocationBarDesignOfOtherActivity.
ResourceTableRow-DefinitionIDForTitle		**Deprecated! See renamed setting TableRowDefinitionIDForTitleInResourcesView.**
RowsDraggable	Boolean	**Optional, default: True** – Only when set to True and the setting Editable is set to True, then rows will be draggable according to the value in the property AllowedRowDragModes in Activity, Allocation, Resource, Skill, and Entity objects. When this setting or the setting Editable is set to False, then rows are generally not draggable. This setting makes it possible to separate the general ability for bar dragging and row dragging, while the setting Editable controls both at once.
RowSortModeNoneEnabledOn-InteractiveSwitchingOfSortOrder	Boolean	**Optional, default: True** – If set to True, then a sort mode can interactively be switched back to unsorted.
ScrollToObjectAnimation-Enabled	Boolean	**Optional, default: False** – If set to True, then scrolling to the target position is animated when using the method ScrollToObject.
ScrollToObjectHighlight-FlashingEnabled		**Deprecated! See renamed setting ObjectHighlightFlashingEnabled.**
ScrollToObjectHighlightingColor		**Deprecated! See renamed setting ObjectHighlightingColor.**
SelectionColor	Text (CSS color value)	**Optional, default: '#ffa000'** – Specifies a color used to highlight selected bars, links or table rows. If a string is given, then the widget uses the color for all view types. This color is also used for

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
		coloring lines and frames while dragging a row interactively.
SeparationLinesInColoredIndentation	Boolean	**Optional, default: False** – If set to True, then vertical separation lines are shown between the colored indentation rectangles at the beginning of the scrollable part of the tables.
SkillSortCodeSources	JsonArray of Text	**Optional, default: ["TableSortCode", "ID"]** – Skill rows are sorted by comparing values of the named properties of the skill objects as following: If the compared skills have the same value of the first property, the values of the second one will be compared and so on. If a property name is prefixed by a minus sign, the skill rows are sorted in descending order concerning the value of this property. Attention: If this setting used, then automatic calculation of sort codes after vertical dragging of a skill row is only possible, when exactly one property name is provided here!
SortingIndicatorVisible	Boolean	**Optional, default: False** – If set to True, then the current sort mode is visible in the table title by showing a small arrow in the appropriate column: Up arrow for ascending order and down arrow for descending order. See also settings InteractiveSwitchingOfSortOrderEnabled, Activity/Allocation/Entity/ ResourceSortCodeSources.
SplitterHighlightingColor	Text (CSS color value)	**Optional, default: '#ffa000'** – Specifies a color used to highlight the splitters when a splitter is dragged. This refers to the splitters between the table or entities table and the Gantt area. If a string is given, then the widget uses the color for all view types.
Start	DateTime	**Optional, default: beginning of today when control add-in is instantiated** – Date where the time area starts absolutely, means when the horizontal scroll bar is placed at the start of the time area view. When using the method fitTimeAreaIntoView, it is not possible there to make dates visible before the start date set here.

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
		If this option was not set before the first rendering, then a warning is triggered (see event OnLogWarnings). It is strongly recommended to set start and end together in a single settings object. This way VS-CAI reacts faster.
StatusAreaVariables	JsonObject	**Optional, default: empty array** – If set to a JsonObject, the property names define names of variables that can be used in HTML strings for the top and the bottom status area (see settings Top/BottomStatusAreaContent). The values then replace the appropriate placeholders. Values that are not of type Text (Integer, Decimal, DateTime) are converted to strings. When this setting is updated, the status areas are updated accordingly.
SubRowDistanceInTimeArea	Integer (> 0)	**Optional, default: 5** – Vertical distance between two bars in pixels. See also TopRowMarginInTimeArea and BottomRowMarginInTimeArea. Please have in mind that symbols are drawn inside this distance.
SuitableActivitiesOverlayColor	Text (CSS color value)	**Optional, default: 'transparent'** – This setting determines the color that is added to activity rows that are mentioned in the allocation property SuitableActivityIDs when dragging. See setting UnsuitableActivitiesOverlayColor.
SuitableResourcesOverlayColor	Text (CSS color value)	**Optional, default: 'transparent'** – This setting determines the color that is added to resource rows that are mentioned in the allocation property SuitableResourceIDs when dragging. See setting UnsuitableResourcesOverlay-Color.
SymbolColumnBackgroundColor	Text (CSS color value)	**Optional, default: 'white'** – If set then the symbol column of the activities/resources table will show this color in the background.
SymbolColumnTitleBackgroundColor	Text (CSS color value)	**Optional, default: value of setting SymbolColumnBackgroundColor** – If set then the symbol column title of the activities/resources table will show this color in the background when the setting SymbolColumnTitleVisible is set to true.
SymbolColumnTitleSymbolIDs	JsonArray of Text	**Optional, default: empty array** – Array of identifiers of the symbols to be shown in the table in the title cell of the symbol

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
		column. They will only appear when the setting SymbolColumnTitleVisible is set to True and setting TableTitleText is not set, so that the table title shows columns. The symbols will be arranged one below the other. However, if the cell is not high enough to hold all symbols, then the remaining symbols are also arranged side-by-side. If this still does not fit, an additional “show more” symbol will be displayed. An empty string ('') will cause an “empty” symbol to be displayed. By this placeholder, you can reserve space for a symbol that may be shown later. Please note: Each symbol will be resized to an image with a width and height of 16 pixels each at a zoom level of 100%.
SymbolColumnTitleVisible	Boolean	**Optional, default: False** – If set to True, the symbols specified in the setting SymbolColumnTitleSymbolIDs will be displayed in the title cell of the symbol column. Otherwise, the title cell will have the same color as defined in the setting TableHeaderBackgroundColor.
SymbolColumnVisible	Boolean	**Optional, default: False** – If set to True, a special column at the left of the table will be displayed to show the row symbols of the activities in the Activities view and of the resources in the Resources or Loads view.
SymbolColumnWidth	Integer	**Optional, default: 22** – Width of the symbol column on the left side of the table in pixels. If set to a value less than the default, it will be set to the default automatically.
TableCellContentTopOffset	Integer > 0	**Optional, default: 21** – Top offset for cell content in table cells of left table. This number is valid for the base line of the first line of text inside the table cell and is only taken into account when it is lower than half of default row height and half of an optional row maximum height.
TableColumnSeparatorColor	Text (CSS color value)	**Optional, default: '#a0a0a0'** – Specifies a color used to color the column separators in the table of the Gantt

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
		diagram. If a string is given, then the widget uses the color for all view types.
TableHeaderBackgroundColor		**Deprecated! Renamed to TableTitleBackgroundColor.**
TableHeaderColumnSeparatorColor		**Deprecated! Renamed to TableTitleColumnSeparatorColor.**
TableHeaderHighlightingColor		**Deprecated! Renamed to TableTitleHighglightingColor.**
TableHeaderTextColor		**Deprecated! Renamed to TableTitleTextColor.**
TableRowDefinitionIDForTitleInActivitiesView	Text	**Optional, default: value of setting DefaultActivityTableRowDefinitionID** – ID of a TableRowDefinition object that will be used to show the table title in the activities view. In parallel, it is currently only possible to interactively change the column widths for the TableRowDefinition object that is referenced here.
TableRowDefinitionIDForTitleInEntitiesTable	Text	**Optional, default: value of setting DefaultEntityTableRowDefinitionID** – ID of a TableRowDefinition object that will be used to show the table title in the entities table. In parallel, it is currently only possible to interactively change the column widths for the TableRowDefinition object that is referenced here.
TableRowDefinitionIDForTitleInLoadsView	Text	**Optional, default: value of setting TableRowDefinitionIDForTitleInResourcesView or, when undefined, value of setting DefaultResourceTableRowDefinitionID** – ID of a TableRowDefinition object that will be used to show the table title in the resources view. In parallel, it is currently only possible to interactively change the column widths for the TableRowDefinition object that is referenced here.
TableRowDefinitionIDForTitleInResourcesView	Text	**Optional, default: value of setting DefaultResourceTableRowDefinitionID** – ID of a TableRowDefinition object that will be used to show the table title in the resources view. In parallel, it is currently only possible to interactively change the column widths for the TableRowDefinition object that is referenced here.
TableRowDefinitionIDForTitleInSkilledResourcesView	Text	**Optional, default: value of setting DefaultSkilledResourceTableRowDefiniti**

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
		onID – ID of a TableRowDefinition object that will be used to show the table title in the resources view. In parallel, it is currently only possible to interactively change the column widths for the TableRowDefinition object that is referenced here.
TableTitleAndTimescaleHeight	Integer >= 0	**Optional, default: 60** – Specifies the height of the left table and the timescale.
TableTitleBackgroundColor	Text (CSS color value)	**Optional, default: '#646464'** – Specifies a color used to color the background of the table header of the Gantt diagram. If a string is given, then the widget uses the color for all view types.
TableTitleColumnSeparatorColor	Text (CSS color value)	**Optional, default: 'white'** – Specifies a color used to color the column separators in the table header of the Gantt diagram. If a string is given, then the widget uses the color for all view types.
TableTitleFontFamily	Text (CSS font family value)	**Optional, default: value of setting FontFamily** – Font family for all texts in the table title. See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-family.
TableTitleFontSize	Text (CSS font size value)	**Optional, default: value of setting FontSize** – Font size for all texts in the table title. See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-size. **Note**: If set, do not forget to append a unit like 'px' to the numeric value! Otherwise, the browser will ignore the setting.
TableTitleFontStyle	Text (CSS font style value)	**Optional, default: value of setting FontStyle** – Font style for all texts in the table title. See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-style
TableTitleFontWeight	Text (CSS font weight value)	**Optional, default: value of setting FontWeight** – Font weight for all texts in the table title.

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
		See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-weight.
TableTitleHighlightingColor	Text (CSS color value)	**Optional, default: '#f7c365'** – Specifies the color to be used during the interaction, e.g. to highlight the separation line between two adjacent columns when altering the column widths.
TableTitleText	Text	**Optional, default: '' (empty string)** – This text will be shown in the table header. It will appear only when not using TableRowDefinition objects or the property TableRowDefinitionIDForTitleInActivities/Resources/LoadsView appropriate to the corresponding view type is **not** set. **Note**: Several immediately consecutive spaces are always combined into one space by the browsers. If the individual spaces are to be preserved, then each of them must be replaced by the Unicode character \u00A0.
TableTitleTextColor	Text (CSS color value)	**Optional, default: 'white'** – Specifies a color used to color the text in the table header of the Gantt diagram. If a string is given, then the widget uses the color for all view types.
TableViewWidth	Integer	**Optional, default: -1** – This setting defines the width of the table view. When set to -1, then the control add-in determines the view width automatically. A change to the vertical splitter is not changing this setting.
TableViewWidthInActivitiesView	Integer	**Optional, default: -1 = null (means table width)** – This setting defines the width of the table view in activities view. A change to the vertical splitter is not changing this setting.
TableViewWidthInLoadsView	Integer	**Optional, default: -1 = null (means table width)** – This setting defines the width of the table view in loads view. A change to the vertical splitter is not changing this setting.
TableViewWidthInResourcesView	Integer	**Optional, default: -1 = null (means table width)** – This setting defines the width of the table view in resources view. A change to the vertical splitter is not changing this setting.

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
TableViewWidthInSkillResourcesView	Integer	**Optional, default: -1 = null (means table width)** – This setting defines the width of the table view in skilled resources view. A change to the vertical splitter is not changing this setting.
TableViewWidthsSynchronized	Boolean	**Optional, default: True** – This setting defines whether an interactive change of the table view width sets the view width of all views or not.
TableWidth		**Deprecated! Calculated automatically when using TableRowDefinition objects.**
TimeAreaBackgroundColor	Text (CSS color value)	**Optional, default: 'white'** – Specifies a color used to color the background of the time area. If a string is given, then the widget uses the color for all view types.
TimeAreaPanningMode	Integer (see enum PanningMode)	**Optional, default: 3** – Specifies, how the control add-in reacts to user interactions inside the empty space of the time area. **Note:** When panning with the mouse, this option is only considered if the setting MultipleSelectionEnabled is set to False.
TimescaleBackgroundColor	Text (CSS color value)	**Optional, default: '#646464'** – Specifies a color used to color the background of the timescale. If a string is given, then the widget uses the color for all view types.
TimescaleContextMenuID	Text	**Optional, default: '' (empty string)** – If set to an ID for an existing ContextMenu object then the appropriate context menu is shown when the users clicks the secondary button or presses the finger on the timescale.
TimescaleHighlightingColor	Text (CSS color value)	**Optional, default: '#f7c365'** – Specifies the color to be used during the interaction on the timescale, e.g. to highlight the time period under the mouse cursor.
TimescaleInteractionMode	number (see enum TimescaleInteractionModes)	**Optional, default: TimescaleInteractionModes.Default** – Specifies which interactions are allowed on the timescale.
TimescaleInteractionsEnabled		**Deprecated! Use setting TimescaleInteractionMode instead.**
TimescaleNavigationMode	Integer (see TimescaleNavigationMode)	**Optional, default: 0** – Mode of navigation in the timescale.
TimescaleTextColor	Text (CSS color value)	**Optional, default: 'white'** – Specifies a color used to color the text in the

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
		timescale. If a string is given, then the widget uses the color for all view types.
TimescaleTickColor	Text (CSS color value)	**Optional, default: 'white'** – Specifies a color used to color the ticks in the timescale. If a string is given, then the widget uses the color for all view types.
TimescaleWeekendBackgroundColor	Text (CSS color value)	**Optional, default: '#888888'** – Specifies a color used to color the background of the weekend cells of the timescale. If a string is given, then the widget uses the color for all view types.
TimescaleWeekendTextColor	Text (CSS color value)	**Optional, default: 'white'** – Specifies a color used to color the text in the timescale weekends.
TimeStepUnit	Text (one of 'second', 'minute', 'hour', 'day', 'week', 'month', 'quarter', 'year')	**Optional, default: value of setting MaximumTimeResolutionUnit or 'second'** – Unit for time steps on horizontal drag interactions of bars. Used together with setting TimeStepUnitFactor. When using a time unit 'day' or above, the stepping is done day-by-day without concerning nonworking times within the period. When you set this value and do not set the settings MaximumTimeResolution-Unit/-Factor, the value here also changes the default value of MaximumTimeResolutionUnit! This is done for compatibility reasons. **Attention!** Currently, the dates of the bars as well as the dates in the calendar must not be defined more finely than this unit together with the TimeStepUnitFactor indicate. Otherwise, unexpected jumps will occur when moving bars.
TimeStepUnitFactor	Integer (>= 1)	**Optional, default: value of setting MaximumTimeResolutionUnitFactor or 1** – Number of units for a single time step on horizontal drag interactions of bars. Used together with setting TimeStepUnit. Integer values are recommended. When you set this value and do not set the settings MaximumTimeResolution-Unit/-Factor, the value here also changes

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
		the default value of MaximumTimeResolutionUnitFactor! This is done for compatibility reasons. **Attention!** Currently, the dates of the bars as well as the dates in the calendar must not be defined finer than this factor together with the setting TimeStepUnit indicate. Otherwise, unexpected jumps may occur when moving bars.
TimeZone	Text	**Optional, default: '' (empty string) –** Time zone setting like in SessionSettings object of Dynamics 365 BC and Dynamics NAV. When set to an empty string, then the time zone of the machine is used (browser default). For an example see setting LocaleId.
TitleText	Text	**Deprecated! See renamed option TableTitleText.**
TonedDownOverlayColor	Text (CSS color value)	**Optional, default: 'rgba(64,64,64,0.5)'** – This setting is used as the overlay color for bars where the flag TonedDownColoring in property BarDesign of an Activity or Allocation object is set to true or when visualizing allocations of other skill in SkilledResources view (see setting AllocationBarDesignOfOtherSkill).
TooltipDelay	Integer > 0	**Optional, default: 500** – This setting determines a delay in milliseconds until a tooltip gets visible.
TopBarSymbolsVisible	Boolean	**Optional, default: True** – If set to False, then no symbols are shown at the top left and top right of allocation bars and activity bars.
TopRowMarginInTimeArea	Integer (> 0)	**Optional, default: 10** – Height of the margin between the top row border and bars in pixels. See also settings BottomRowMarginInTimeArea and SubRowDistanceInTimeArea. Please have in mind that symbols are drawn inside this margin. When one of the options detailedActivity/AllocationConstraintSymbolsEnabled is set to true, then the value here should be set to a value of 15 or above to avoid a vertical overlap of the row contents. For maximizing the performance there is no clipping of

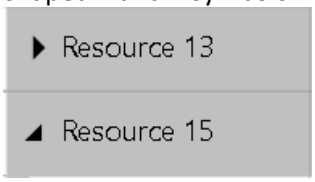
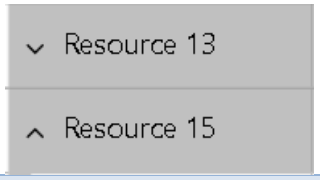
© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
		elements that are drawn above the top row border.
TopStatusAreaContent	Text	**Optional, default: '' (empty string)** – If set and the height is set to a value greater than 0 (see setting TopStatus-AreaHeight), the value is interpreted as HTML and shown above the chart. The HTML content is placed inside a DIV element. It is possible to use variables defined by {{variableName}}. These are replaced by the values provided by the setting StatusAreaVariables. The keyword {{*@symbolID*}} is a placeholder for a defined symbol and can be used in <image src="…"> to show a symbol whenever needed. Alternatively, also standard URLs to external images are possible. If an application wants to get OnClicked and/or OnDoubleClicked events on some HTML elements, then it should set the ID attribute und add the class name "nevd-tap" or "nevd-doubletap", resp.. If an application wants to show a tooltip on some HTML elements, then it possible add an attribute named “data-vscai-tttid” to it and setting it to the ID of an TooltipTemplate object. Analogously it is possible to use a context menu on some HTML elements: Just add an attribute named “data-vscai-ctxmid” setting it to the ID of a ContextMenu object. When the user opens it and selects an item, then the application gets an OnContextMenuItemClicked event with the property ObjectType set to 'TopStatusArea' and the property ObjectID set to the ID of the HTML element. This is the same behavior as for the OnClicked event on such HTML elements.
TopStatusAreaHeight	Integer >= 0	**Optional, default: 0** – If set to a value greater than 0, this defines the height of

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
		the top status area (see setting TopStatusAreaContent).
TopViewAreaVisible		**Deprecated! See renamed option TopViewAreaVisibleInResourcesView.**
TopViewAreaVisibleInActivitiesView	Boolean	**Optional, default: False** – If set to True, then activities in the activities view are shown in a separate top view area, that have the property ViewArea set to Top. See also settings MainViewAreaVisibleInActivitiesView and MaximumTopViewAreaHeightRatio.
TopViewAreaVisibleInLoadsView	Boolean	**Optional, default: False** – If set to True, then resources in the loads view are shown in a separate top view area, that have the property ViewArea set to Top. See also settings MainViewAreaVisibleInLoadsView and MaximumTopViewAreaHeightRatio.
TopViewAreaVisibleInResourcesView	Boolean	**Optional, default: False** – If set to True, then resources in the resources view are shown in a separate top view area, that have the property ViewArea set to Top. See also settings MainViewAreaVisibleInResourcesView and MaximumTopViewAreaHeightRatio.
TopViewAreaVisibleInSkilledResourcesView	Boolean	**Optional, default: False** – If set to True, then skills in the skilled resources view are shown in a separate top view area, that have the property ViewArea set to Top. See also settings MainViewAreaVisibleInSkilledResourcesView and MaximumTopViewAreaHeightRatio.
TreeViewLineColor	Text (CSS color value)	**Optional, default: 'black'** – Determines the color of tree view lines in the left table. See setting TreeVisualizationMode.
TreeViewLineDashArray	Text (SVG stroke dash array value)	**Optional, default: '' (empty string)** – Pattern of dashes and gaps for drawing the tree view lines in the left table. For further information, please see https://www.w3.org/TR/SVG11/painting.html#StrokeDasharrayProperty or https://developer.mozilla.org/en-US/docs/Web/SVG/Attribute/stroke-dasharray. The value 'none' indicates that no dashing is used. In this case, the line is drawn solid. See setting TreeVisualizationMode.

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
UniformExpandCollapseButtonSymbol	Boolean	**Optional, default: False** – If set to true, the symbols for the buttons to expand and collapse the allocation rows are filled triangles instead of the wire-shaped < and > symbols.  instead of 
UnsuitableActivitiesOverlay-Color	Text (CSS color value)	**Optional, default: 'rgba(0,0,0,0.2)'** – This option determines the color that is added to activity rows that are NOT mentioned in the allocation property SuitableActivityIDs when dragging. See setting SuitableActivitiesOverlayColor.
UnsuitableResourcesOverlay-Color	Text (CSS color value)	**Optional, default: 'rgba(0,0,0,0.2)'** – This option determines the color that is added to resource rows that are NOT mentioned in the allocation property SuitableResourceIDs when dragging. See setting SuitableResourcesOverlayColor.
UpdateMode	Integer (see enum UpdateModes)	**Optional, default: Default** – Specifies the behavior of all update methods.
VerticalSplitterWidth	Integer (> 0)	**Optional, default: 8** – This setting determines the width of the vertical splitters between left table and time area and between time area and right table.
ViewType	Integer (see option ViewType)	**Optional, default: ViewType.Resources** – Determines the type of view. In the resources view and in the loads view the table shows resources, in the activities view the table shows activities. All views show the objects in hierarchical order.
VisualZoomFactor	Decimal	**Optional, default: 1.0** – Factor used to zoom in (>1) and out (<1) the whole widget. Values <= 0 will be ignored.
WatermarkOpacity	Decimal (>= 0.0, <= 1.0)	**Optional, default: 0.2** – Opacity of the watermark. See also setting WatermarkSymbolID.
WatermarkSymbolID	Text	**Optional, default: '' (empty string)** – Identifier of the symbol to be shown in the time area of the Gantt chart. The

© 2026 Boyum IT Solutions GmbH

Settings Property Name	Type	Description
		symbol is stretched while maintaining the ratio between width and height of the symbol so that it is as large as possible. See also setting WatermarkOpacity.
WeekNumbering	Text (currently possible values: 'ISO8601', 'USA', '')	**Optional, default: '' (empty string)** – This option determines the first day of the week (ISO8601: Monday, USA: Sunday) and the week numbering scheme. If set to undefined, then the implicit setting of the setting LocaleID is used.
WorkDate	DateTime	**Optional, default: 0DT** – Date on which the work date line will be displayed. If outside of the time range between start and end of the time area, then no line will be visible. Please note: The work date line is a simple line only. There are no further properties like color, line width, or line pattern to be set. If such properties are needed, then a DateLine object should be used.
WorkDateLineCaption	Text	**Optional, default: '' (empty string)** – Text to be displayed at the work date line. **Note**: Several immediately consecutive spaces are always combined into one space by the browsers. If the individual spaces are to be preserved, then each of them must be replaced by the Unicode character \u00A0.
WorldViewExtent	Integer	**Optional, default: 150** – Defines the extent of the world view in pixels.
WorldViewPosition	Integer (see enum WorldViewPosition)	**Optional, default: Bottom** – Defines the position of the world view within the control add-in.
WorldViewVisible	Boolean	**Optional, default: False –** If set to true, then a world view is visible at the bottom of the Gantt chart. Only the table row background colors and bar colors are shown. Also date lines and separation lines between left table, timescale, top view area are shown. Additionally, selections are shown and frames for the visible parts shown in the widget (separately for table and time area). These frames can also be dragged to modify the visible parts.

© 2026 Boyum IT Solutions GmbH

4.13 Skill



A Skill object is used for defining a skill for allocations and resources. It is visible when switching to the skilled resources view.

In this view, skills are shown as rows in the highest hierarchy level with the associated resources in rows of the next hierarchy level. Since the property SkillIDs of resources can hold references to more than one skill, resources show up more than once in this view. Only resources without parent are visible within the skilled resources view.

Each resource row shows the allocations assigned to it, but the allocations that reference the skill of the hierarchy parent row of the resource row are shown in full bar design while the others are shown with less details (see setting AllocationBarDesignOfOtherSkill). So, allocations also show up more than once in this view.

NOTE: On data objects that are updated missing properties normally are interpreted as being set to the default value!

Skill Property Name	Type	Description
AllowedRowDragModes	Integer (see enum RowDragModes)	**Optional, default: value of setting DefaultSkillAllowedRowDragModes** – This property determines the allowed row drag modes for this skill. See also settings RowsDraggable and Editable.
CollapsedRowDesign	Integer (see enum RowDesigns)	**Optional, default: value of setting DefaultSkillCollapsedRowDesign** – Specifies how the time area is filled when the skill row is collapsed and visible. Only the flags BarsOfHiddenDescendantRows and BarsStacked are supported here.
CollapseState	Integer (see enum CollapseState)	**Optional, default: -1** – Specifies whether the row of the skill should be expanded or collapsed. See also event OnCollapseStateChanged. -1: no change 0: display resource row in an expanded way 1: display resource row in a collapsed way
ID	Text	**Required** – Identifier of this skill. Any content is allowed besides an empty string, a contained dot, or contained characters with code points 0 to 31 and 127. The first character should be a letter. Each object

© 2026 Boyum IT Solutions GmbH

Skill Property Name	Type	Description
		type has its own identifier name space, so it is allowed to use the same ID for each object type separately.
MinimumRowHeight	Integer	**Optional, default: value in setting DefaultSkillMinimumRowHeight** – Minimum height of the skill row in pixels. This option is useful, when more than one line of text is shown inside the table cells. Proposal: For one line take 36, for two lines 52, for three lines 68, and so on. To have the same height also, when no bar is placed in the row, take the maximum of the height of all bars (default bar height is 22) plus 20, so e.g. 42. For using word wrapping in table cells, it is necessary to use a table row definition by setting the property TableRowDefinitionID and setting the property WrapMode in a contained table cell definition. In general, the height of a row is determined by several facts: The height of bars and the stacking of bars determine a height, that then is overlaid by the value of this property. The height of any text inside a table column or a bar is not considered, even when using wrapping.
RowCollapsible	Boolean	**Optional, default: value of setting DefaultSkillRowCollapsible** – If set to True, then the row representing this skill will be interactively collapsible when children exist.
RowSelectable	Boolean	**Optional, default: value of setting DefaultSkillRowSelectable** – If set to True, then the row representing this skill will be selectable.
RowSymbolColumnBackgroundColor	Text (CSS color value)	**Optional, default: value of property SymbolColumnBackgroundColor of assigned table row definition or option symbolColumnBackgroundColor** – Determines the color of the symbol column within the table row.
RowSymbolIDs	JsonArray of Text	**Optional, default: undefined** – Array of identifiers of the symbols to be shown in the table symbol cell of the beginning of the table row. The symbols will be arranged one below the other. However, if the cell is not high enough to hold all symbols, then the remaining symbols are also arranged side-

© 2026 Boyum IT Solutions GmbH

Skill Property Name	Type	Description
		by-side. If this still does not fit, an additional “show more” symbol will be displayed. An empty string ('') will cause an “empty” symbol to be displayed. By this placeholder, you can reserve space for a symbol that may be shown later. Please note: Each symbol will be resized to an image with a width and height of 16 pixels each at a zoom level of 100%.
RowTooltipTemplateID	Text	**Optional, default: '' (empty string) = value of setting DefaultSkillRowTooltipTemplateID** – ID of a tooltip template. The template is used for the tooltip that will appear on the table row when hovering the cursor above it.
TableColor	Text (CSS color value)	**Optional, default: level-dependent gray** – Color for the table row. If '' (empty string), a predefined level-dependent gray value is used.
TableColorVisibleInTimeArea	Boolean	**Optional, default: False** – If set to true, the time area row will be colored using the color defined by the TableColor property.
TableFontFamily	Text (CSS font family value)	**Optional, default: '' (empty string) = value of setting FontFamily** – Font family for all texts in the table cells. See property TableText and properties TextSource and TextFormat of TableCellDefinition object. The value does not influence the height of the row! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-family.
TableFontSize	Text (CSS font size value)	**Optional, default: '' (empty string) = value of setting FontSize** – Font size for all texts in the table cells. See property TableText and properties TextSource and TextFormat of TableCellDefinition object. The value does not influence the height of the row! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-size.

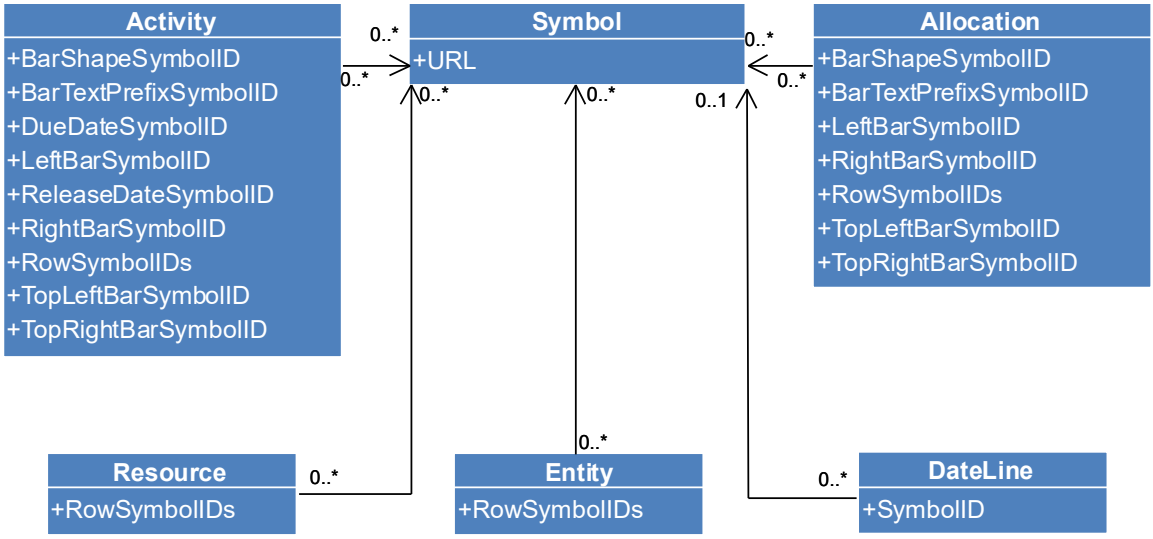
© 2026 Boyum IT Solutions GmbH

Skill Property Name	Type	Description
		Note: If set, do not forget to append a unit like 'px' to the numeric value! Otherwise, the browser will ignore the setting.
TableFontStyle	Text (CSS font style value)	**Optional, default: '' (empty string) = value of setting FontStyle** – Font style for all texts in the table cells. See property TableText and properties TextSource and TextFormat of TableCellDefinition object. The value does not influence the height of the row! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-style
TableFontWeight	Text (CSS font weight value)	**Optional, default: '' (empty string) = value of setting FontWeight** – Font weight for all texts in the table cells. See property TableText and properties TextSource and TextFormat of TableCellDefinition object. The value does not influence the height of the row! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-weight.
TableRowDefinitionID	Text	**Optional, default: '' (empty string) = value of setting DefaultSkillTableRowDefinitionID** – Identifier of a TableRowDefinition object, that defines the composition of the table row.
TableSortCode	Text	**Optional, default: '' (empty string)** – If set, then the value will be used when sorting skill rows. The value type can be anyone that can be compared using JavaScript. See also settings SkillRowSortCodeSources.
TableText	Text	**Optional, default: '' (empty string)** – Text to display in the table row (see also property TableRowDefinitionID). **Note**: Several immediately consecutive spaces are always combined into one space by the browsers. If the individual spaces are to be preserved, then each of them must be replaced by the Unicode character \u00A0.
TableTextColor	Text (CSS color value)	**Optional, default: 'black'** – Color for the table row texts.

© 2026 Boyum IT Solutions GmbH

Skill Property Name	Type	Description
ViewArea	Integer (see enum ViewArea)	**Optional, default: 0 (Main)** – If set to Top, then the skill and its subtree are shown in a separate top view area in the skilled resources view. See also settings MainViewAreaVisibleInSkilledResourcesView and TopViewAreaVisibleInSkilledResourcesView.

4.14 Symbol



A Symbol object defines the properties of a single symbol. Symbols are used by resources, activities, and allocations. They can be displayed at different locations inside the table and the diagram area.

An image contained in an PNG file or SVG file can use transparent backgrounds and can contain its own font includes. For the image you should avoid big empty borders around the graphics itself, because this results in smaller graphics on screen than maybe expected.

The symbols will be resized to an image with an appropriate width and height depending on their application. Therefore, when designing the symbols, you should ensure that they are clearly recognizable and visually distinguishable. For more details regarding the size, please see the descriptions of the properties related to symbols.

It is not possible to use paths in the property URL, but instead you have the possibility to use 'data URIs', that can be created using an online service (e.g. https://websemantics.uk/tools/image-to-data-uri-converter/) to convert your SVG file to a string containing the SVG.

One hint: If the symbol is not shown correctly scaled in the NAV/BC Windows client, maybe the SVG tag inside the symbol does not contain the property viewBox that has to be added in that case.

Another hint: If you want to use our method saveAsPDF, then you will have to ensure that your SVG image files do not contain <style> tags, since the contained selectors may change the appearance of

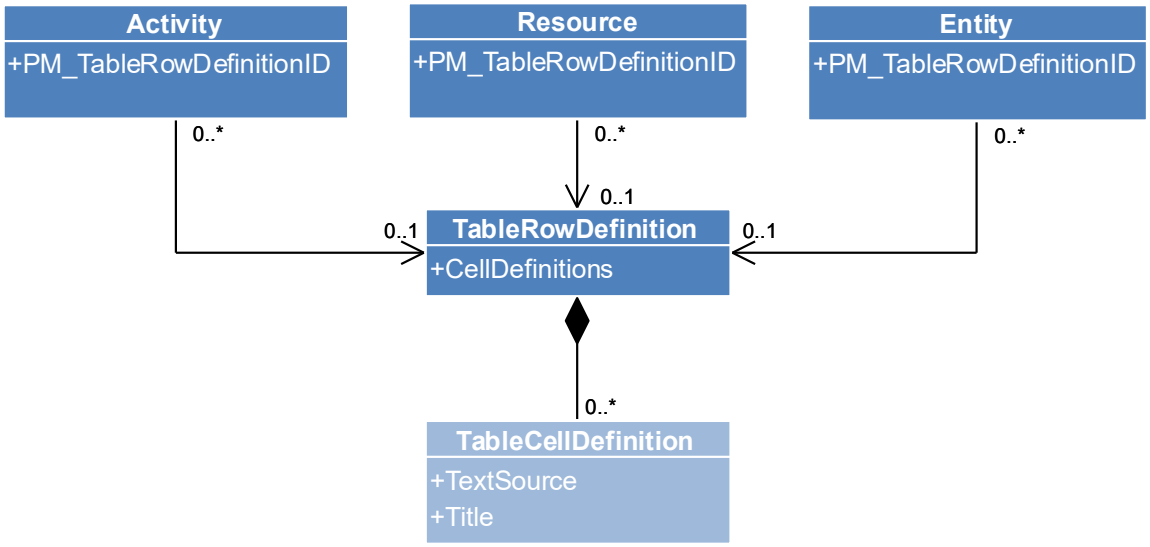
© 2026 Boyum IT Solutions GmbH

the exported SVG content. In a case of existing <style> try to replace them by using style attributes on other tags. We can help if there are problems arising.

NOTE: On data objects that are updated missing properties normally are interpreted as being set to the default value!

Symbol Property Name	Type	Description
ClickableInEntitiesTable	Boolean	**Optional, default: False** – If set to True, then the symbol is clickable in the entities table.
ClickableInTable	Boolean	**Optional, default: False** – If set to True, then the symbol is clickable in the table.
ContextMenuID	Text	**Optional, default: '' (empty string)** – ID of a ContextMenu object. The context menu is shown when the user clicks the secondary mouse button or presses the symbol with the finger.
ID	Text	**Required** – Identifier of this symbol.
TooltipTemplateID	Text	**Optional, default: '' (empty string)** – ID of a TooltipTemplate object. The template is used for the tooltip that appears when the mouse pointer hovers over it.
URL	Text	**Required** – URL of a SVG image containing the symbol. Three types of URLs are allowed: • absolute URL (e.g. 'https://www.aaazzz.com/symbol.svg") • relative URL (e.g. 'images/symbol.svg") – In this case, the anchor path for the symbol directory is the application directory. • Data URI (e.g. '...'). See https://en.wikipedia.org/wiki/Data_URI_scheme)

4.15 TableRowDefinitions



© 2026 Boyum IT Solutions GmbH

A TableRowDefinition object defines the composition of a table row containing one or more cells. You can reference these objects with the property TableRowDefinitionID of Activity, Allocation, Entity, and Resource objects. There are settings for each property defining a default value for the corresponding property (see DefaultActivity/Allocation/Entity/ResourceTableRowDefinitionID).

Additionally, it is possible to declare one table row definition to provide the table title for the views and the entities table by using the settings TableRowDefinitionIDForTitleInActivities/Resources/LoadsView or TableRowDefinitionIDForTitleInEntitiesTable.

4.15.1 TableRowDefinition

NOTE: On data objects that are updated missing properties normally are interpreted as being set to the default value!

TableRowDefinition Property Name	Type	Description
BackgroundColor	Text (CSS color value)	**Optional, default: '' (empty string)** – Background color of the table row. Would be overlayed by the background color of the table row defined in the property TableColor of the row object this object is applied to.
CellDefinitions	JsonArray of JsonObject (TableCellDefinition objects)	**Optional, default: [{ Title: '', TextSource: 'TableText', Width: 200, HorizontalAlignment: HorizontalAlignment.Left }] –** Array of TableCellDefinition objects.
FontFamily	Text (CSS font family value)	**Optional, default: value of property TableFontFamily of the row object this object is applied to** – Font family for the text of the table cells. See also property of same name in TableCellDefinition object. The value does not influence the height of the row! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-family.
FontSize	Text (CSS font size value)	**Optional, default: value of property TableFontSize of the row object this object is applied to** – Font size for the text of the table cells. See also property of same name in TableCellDefinition object. The value does not influence the height of the row! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-size.

© 2026 Boyum IT Solutions GmbH

TableRowDefinition Property Name	Type	Description
		Note: If set, do not forget to append a unit like 'px' to the numeric value! Otherwise, the browser will ignore the setting.
FontStyle	Text (CSS font style value)	**Optional, default: value of property TableFontStyle of the row object this object is applied to** – Font style for the text of the table cells. See also property of same name in TableCellDefinition object. The value does not influence the height of the row! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-style
FontWeight	Text (CSS font weight value)	**Optional, default: value of property TableFontWeight of the row object this object is applied to** – Font weight for the text of the entry. See property Text. See also property of same name in TableCellDefinition object. The value does not influence the height of the row! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-weight.
ID	Text	**Required** – Identifier of this table row definition.
SymbolColumnBackgroundColor	Text (CSS color value)	**Optional, default: value of setting SymbolColumnBackgroundColor or EntitiesTableSymbolColumnBackgroundColor, resp.** – Determines the color of the symbol column within this table row when the property RowSymbolColumnBackgroundColor is not set on the object where this table row definition is applied to.
TextColor	Text (CSS color value)	**Optional, default: '' (empty string)** – TextColor of the table row definition. Would be overlayed by the text color of the table row defined in the property TableTextColor of the row object this object is applied to.

4.15.2 TableCellDefinition

NOTE: On data objects that are updated missing properties normally are interpreted as being set to the default value!

TableCellDefinition Property Name	Type	Description
BackgroundColor	Text	**Optional, default: '' (empty string)** – If set and property BackgroundColorSource is empty or the

© 2026 Boyum IT Solutions GmbH

TableCellDefinition Property Name	Type	Description
		referenced property on a row object is empty, then this color overlays the background color of the table row defined in the property TableColor of the row object.
BackgroundColorSource	Text	**Optional, default: '' (empty string)** – If set to an object's property name and the value of the referenced property on a row object is not empty, then the value there overlays the background color defined by property BackgroundColor and property TableColor of the row object.
FontFamily	Text (CSS font family value)	**Optional, default: value of property FontFamily of the table row definition** – Font family for the text. The value does not influence the height of the row! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-family.
FontSize	Text (CSS font size value)	**Optional, default: value of property FontSize of the table row definition** – Font size for the text. The value does not influence the height of the row! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-size. **Note**: If set, do not forget to append a unit like 'px' to the numeric value! Otherwise, the browser will ignore the setting.
FontStyle	Text (CSS font style value)	**Optional, default: value of property FontStyle of the table row definition** – Font style for the text. The value does not influence the height of the row! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-style
FontWeight	Text (CSS font weight value)	**Optional, default: value of property FontWeight of the table row definition** – Font weight for the text. The value does not influence the height of the row! See https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/Properties/font-weight.
HorizontalAlignment	Integer (see enum HorizontalAlignment)	**Optional, default: Left** – Horizontal alignment of the shown text. The first column is always shown with left alignment because of the tree symbols on the left side.
HorizontalTitleAlignment	Integer (see enum HorizontalAlignment)	**Optional, default: Center** – Horizontal alignment of the shown title text. In the entities table the last column is always shown with center alignment.
MaximumWidth	Integer	**Optional, default: infinity** – Maximum width of the table cell, when cell width is interactively modified.

© 2026 Boyum IT Solutions GmbH

TableCellDefinition Property Name	Type	Description
MinimumWidth	Integer	**Optional, default: 3** – Minimum width of the table cell, when cell width is interactively modified.
SymbolHeight	Integer	**Optional, default: -1 = not set** – If set, then the symbol height is constant. If not set, then the symbol height is determined on each table row to fill the full height of that.
SymbolIDSource	Text	**Optional, default: '' (empty string)** – Property name to take the symbol ID out of the referencing activity, resource, or entity object. The symbol will be displayed in the cell inside a square that has the size of the minimum row height. The symbol will obey the HorizontalAlignment property. It is also possible to use the TextSource property along with this property, but there are the following restrictions: If using left alignment, the text will be indented so that it is to the right of the symbol. If using center or right alignment, the symbol will be overlapped by the text.
SymbolWidth	Integer	**Optional, default: value of property SymbolHeight** – If set, then the symbol width is constant. If not set, then the symbol will be stretched to be square.
TextColor	Text	**Optional, default: '' (empty string)** – If set and property TextColorSource is empty or the referenced property on a row object is empty, then this color overlays the text color of the table row defined in the property TableTextColor of the row object.
TextColorSource	Text	**Optional, default: '' (empty string)** – If set to an object's property name and the value of the referenced property on a row object is not empty, then the value there overlays the text color defined by property TextColor and property TableTextColor of the row object.
TextFormat	Text	**Optional, default: undefined** – String that describes the format of the content of the cell. This property overlays the property TextSource. This string contains the placeholders for object values surrounded by double curly braces {{ }}. For example, based on the following string the content for the table cell is created, in which the "name" and "firstName" properties of the referenced object are concatenated separated by a comma: `{{name}}, {{firstName}}` As an escape, the use of three open curly braces {{{ are displayed as {{.

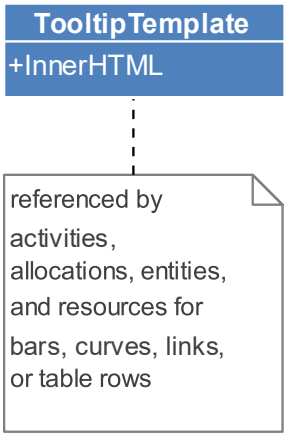
© 2026 Boyum IT Solutions GmbH

TableCellDefinition Property Name	Type	Description
		Additionally, the property name can be extended to contain the desired property type as in {{Start:date}}. At the moment only the types 'date' and 'number' are possible besides 'string' (other property types are converted automatically with toString()). The type 'date' converts date values by default using the same format as other dates in the timescale and at the dragging date line captions. You can add another colon followed by a format name, that is defined by the settings IntlDateTimeFormatOptionsMap or IntlNumberFormatOptionsMap, resp. The referenced object is the object on which the tooltip will be shown. For period highlighter entries and allocation entries the referenced object is the main object and not the entry object. It is possible to access related objects by using the following keywords within the property accessor string: • On activities: >Parent, >Calendar • On resources: >Parent, >Calendar, >LoadCurve, >CapacityCurve • Additionally on resources in SkilledResources view: >Skill • On entities: >Parent • On allocations: >Activity, >Resource • On links: >SourceActivity, >TargetActivity, >SourceAllocation, >TargetAllocation It is also possible to access variables that are defined by the setting ApplicationVariablesMap by using ?*variableName*. If the value reached is an object, you can then access a property value by using a prefixed dot: .propertyName and you can use […] to access a property value, a map entry or an array entry. Within […] you can use a literal like 5 or A (with or without quotes) or even curly braces {{…}} with the same rules as above.
TextSource	Text	**Optional, default: 'TableText'** – Property to take the text out of the referencing activity, resource, or entity object.
Title	Text	**Optional, default: '' (empty string)** – When the table row definition, that contains this table cell definition, is referenced by one of the settings TableRowDefinitionIDForTitleActivities/

© 2026 Boyum IT Solutions GmbH

TableCellDefinition Property Name	Type	Description
		Resources/LoadsView, then the title defined here will be shown on the table title. **Note**: Several immediately consecutive spaces are always combined into one space by the browsers. If the individual spaces are to be preserved, then each of them must be replaced by the Unicode character \u00A0.
VerticalAlignment	Integer (see enum VerticalAlignment)	**Optional, default: FirstLineOnBaseLine** – Vertical alignment of the shown text. This is only working when WrapMode is not set None.
Width	Integer	**Optional, default: 200** – Width of the table cell.
WrapMode	Integer (see enum TextWrapMode)	**Optional, default: None** – If set, then it possible to show more than one line of text using newline characters ('\n').

4.16 TooltipTemplate



A TooltipTemplate object describes the appearance of a tooltip in the form of an HTML string. This string describes a DOM subtree and contains placeholders with references to the object properties to be displayed. At runtime, the placeholders are replaced by the values of the referenced object properties. This helps building tooltip texts as a replacement for the properties named TooltipText at several objects.

1. You can use the properties TooltipTemplateID, BarTooltipTemplateID, RowTooltipTemplateID, and CurveTooltipTemplateID of the activities, resources, allocations, links, entities, skills, and period highlighter entries. All these properties have fallback settings named:
    - DefaultActivityBar/AllocationBar/SkilledAllocationBarTooltipTemplateID
    - DefaultActivityRow/AllocationRow/ResourceRow/EntityRow/SkillRow/SkilledAllocationRowTooltipTemplateID
    - DefaultResourceCurveTooltipTemplateID
    - DefaultPeriodHighlighterEntryTooltipTemplateID

© 2026 Boyum IT Solutions GmbH

NOTE: On data objects that are updated missing properties normally are interpreted as being set to the default value!

TooltipTemplate Property Name	Type	Description

HTMLFormat	Text	**Optional, default: '' (empty string)** – HTML string that describes the structure of a tooltip.
		
		This string contains the placeholders for object values surrounded by double curly braces {{ }}. For example, based on the following string a tooltip with a table containing three rows of key-value pairs is created, where the values are taken from the object properties "name", "firstName", and "age":
		
		`<table>`
		`  <tr><td>Name: </td><td>{{name}}</td></tr>`
		`  <tr><td>First name: </td><td>{{firstName}}</td></tr>`
		`  <tr><td>Age: </td><td>{{age}}</td></tr>`
		`</table>`
		
		As an escape, the use of three open curly braces {{{ are displayed as {{.
		
		Additionally, the property name can be extended to contain the desired property type as in {{Start:date}}. At the moment only the type 'date' is possible besides 'string' (other property types are converted automatically with toString()). The type 'date' converts date values using the same format as other dates in the timescale and at the dragging date line captions.
		
		It is possible to get associated objects by using the following keywords:
		On activities: >Parent, >Calendar
		On resources: >Parent, >Calendar, >LoadCurve, >CapacityCurve
		On entities: >Parent
		On allocations: >Activity, >Resource
		On links: >SourceActivity, >TargetActivity, >SourceAllocation, >TargetAllocation
		
		It is also possible to access other objects that are otherwise reachable by the callback arguments of the callback onShowTooltip by using the following keywords at the beginning of the property accessor string:
		• On allocation bars: #Entry
		• On period highlighters: #Entry, #RowObject.
		• On curves: #Date, #Capacity, #Load, #FreeCapacity, #SingleLoads. For #SingleLoads you have to add *.curveID* to get the curve value for a single curve. #FreeCapacity represents the capacity reduced by the load.
		• On resources and allocation rows in the skilled resources view: #Skill.
		
		It is also possible to access variables that are defined by the option applicationVariablesMap by using ?*variableName*.

© 2026 Boyum IT Solutions GmbH

TooltipTemplate Property Name	Type	Description
		If the value reached is an object, you can then access a property value by using a prefixed dot: .propertyName and you can use […] to access a property value, a map entry or an array entry. Within […] you can use a literal like 5 or A (with or without quotes) or even curly braces {{…}} with the same rules as above. Also, the keyword {{*@symbolID*}} is a placeholder for a defined symbol and can be used in <image src="…"> to show a symbol if needed (other URLs to external images are also possible).
ID	Text	**Required** – Identifier of this tooltip template.
InnerHTML		**Deprecated. Renamed to HTMLFormat.**
IsInteractive	Boolean	**Optional, default: False** – When set to false, which is the default, the tooltip is treated as being not interactive. This means that the user cannot interact with it e.g. by clicking into it. Therefore, the tooltip in this case does not accept pointer events and disappears when the pointer cursor leaves the rectangle of the object that made the tooltip appear before. When set to true, the tooltip is treated as being interactive. This means that the user can interact with it e.g. by clicking into it. The HTML therefore can include scrollable areas or hyperlinks or other input elements. The tooltip will disappear when the pointer cursor leaves the rectangle of the object that made the tooltip appear before only when the cursor moves away from the tooltip or when leaving the tooltip rectangle after entering it.

5 Control Add-in

This is the central object that an application talks to. Here are methods to add, update and remove the data objects meant above and there also are many settings and events to refine the appearance of the control add-in.

5.1 Methods

Method Name	Parameters	Description
About	-	Shows a dialog with version and copyright information. The dialog can be made visible also directly by the user by pressing Shift+Ctrl+Alt+F12.
AddActivities	activities : JsonArray of Activity	Adds activities. [1]

© 2026 Boyum IT Solutions GmbH

Method Name	Parameters	Description
AddAllocations	allocations : JsonArray of Allocation	Adds allocations. [1]
AddCalendars	calendars : JsonArray of Calendar	Adds calendars. [1]
AddContextMenus	contextMenus : JsonArray of ContextMenu	Adds context menus.
AddCurves	curves : JsonArray of Curve	Adds curves. [1]
AddDateLines	dateLines : JsonArray of DateLine	Adds date lines. [1]
AddEntities	entities : JsonArray of Entity	Adds entities. [1]
AddLinks	links : JsonArray of Link	Adds links. [1]
AddPeriodHighlighters	links : JsonArray of PeriodHighlighter	Adds period highlighters. [1]
AddResources	resources : JsonArray of Resource	Adds resources. [1]
AddSkills	symbols: JsonArray of Skill	Adds skills. [1]
AddSymbols	symbols: JsonArray of Symbol	Adds symbols. [1]
AddTableRowDefinitions	tableRowDefinitions: JsonArray of TableRowDefinition	Adds table row definitions. [1]
AddTooltipTemplates	tooltipTemplates: JsonArray of TooltipTemplate	Adds tooltip templates. [1]
CancelSaveAsPDF	-	Cancels the execution of the SaveAsPDF method.
FitTimeAreaIntoView	start : DateTime, end : DateTime	Fits the given time range into the visible area. As a result, the time resolution is set in dependence from the width of the time area view. The method will be processed using animation when not called before the first call to the method *Render*. Attention: The dates provided as arguments here cannot exceed the time

[1] After changing the data model, the changes will not become visible until the method "Render" is called. These calls should be made after all changes are made once. If forgotten, there is a timeout which calls the method "Render" automatically, but this eventually leads to flickering within the visualization of the Control Add-in.

© 2026 Boyum IT Solutions GmbH

Method Name	Parameters	Description
		range previously defined in the settings *Start* and *End*. The end date itself is not included in the visible time area. The same behavior also is true for all end dates used in the widget like e.g. in calendar entries, period highlighter entries, activities, activity entries, allocations, allocation entries. **Note:** The time resolution cannot be set finer than the maximum time resolution defined by the settings *MaximumTimeResolutionUnit* and *MaximumTimeResolutionUnitFactor*!
FitFullTimeAreaIntoView	-	Fits the whole defined time range into the visible area. As a result, the time resolution is set in dependence from the width of the time area view. The method will be processed using animation when not called before the first call to the method *Render*. **Note:** The time resolution cannot be set finer than the maximum time resolution defined by the settings *MaximumTimeResolutionUnit* and *MaximumTimeResolutionUnitFactor*!
HighlightObjects	objectType : ObjectType, objectIDs : JsonArray of Text, visualType : VisualType	Highlights the given objects or the objects addressed by the given IDs. In the activities view only activities and allocations can be highlighted. In the resources view only resources and allocations can be highlighted. In the loads view only resources can be highlighted. When objects are provided, these can be the original objects that are registered by one of the add or update methods, or they can be new (literal) objects, since only the properties ID and SkillID (in skilled resources view for resources and allocations, see below) are read on them. The parameter visualType is only required for objects of type Activity or Allocation. In this case you can define whether the rows (VisualType.Row) or the bars (VisualType.Bar) should be highlighted.

© 2026 Boyum IT Solutions GmbH

Method Name	Parameters	Description
		Note: In the resources view, VisualType.Row can be applied only to allocations that reside in separate rows (allocation rows), but not to allocations inside resource rows. The highlighting is shown by a flashing frame around the objects. The highlighting ends automatically when changing the data or with any user interaction or by using the method ScrollToObject. When skilled resources view is visible, it is allowed to provide a SkillID property to highlight a specific graphical object in the view. **Note**: An empty array will remove active highlightings. In this case the item type is not important and can be set to 0. See also methods SelectObjects and ScrollToObject. See also settings ObjectHighlightingColor, ObjectHighlight-FlashingEnabled.
RemoveActivities	activityIDs : JsonArray of Text	Removes activities referenced by their IDs. [1]
RemoveAll	-	Removes all objects. [1]
RemoveAllOfType	objectType : ObjectType	Removes all objects of the given object type. The application can also use the special value -2 to remove all data objects at once (activities, allocations, calendars, curves, entities, links, resources, skills). [1]
RemoveAllOfTypes	objectTypes : JsonArray of ObjectType	Removes all objects of the given object types. If objects of more than one object type are removed, then VS-CAI decides automatically for the order with the most performance. [1]
RemoveAllocations	allocationIDs : JsonArray of Text	Removes allocations referenced by their IDs. [1]
RemoveCalendars	calendarIDs : JsonArray of Text	Removes calendars referenced by their IDs. [1]
RemoveContextMenus	contextMenuIDs : JsonArray of Text	Removes context menus referenced by their IDs.
RemoveCurves	curveIDs : JsonArray of Text	Removes curves referenced by their IDs. [1]
RemoveDateLines	dateLineIDs : JsonArray of Text	Removes date lines referenced by their IDs. [1]
RemoveEntities	entityIDs : JsonArray of Text	Removes entities referenced by their IDs. [1]

© 2026 Boyum IT Solutions GmbH

Method Name	Parameters	Description
RemoveLinks	linkIDs : JsonArray of Text	Removes links referenced by their IDs. [1]
RemovePeriodHighlighters	periodHighlighterIDs : JsonArray of Text	Removes period highlighters referenced by their IDs. [1]
RemoveResources	resourceIDs : JsonArray of Text	Removes resources referenced by their IDs. [1]
RemoveSkills	skillIDs : JsonArray of Text	Removes skills referenced by their IDs. [1]
RemoveSymbols	symbolIDs : JsonArray of Text	Removes symbols referenced by their IDs. [1]
RemoveTableRowDefinitions	tableRowDefinitionIDs : JsonArray of Text	Removes table row definitions referenced by their IDs. [1]
RemoveTooltipTemplates	tooltipTemplateIDs : JsonArray of Text	Removes tooltip templates referenced by their IDs. [1]
Render	-	Refreshes the view after changes to data objects. When the application forgets to call this method, then it is called automatically when the application goes idle (but this can lead to flickering in the control add-in's visualization). The application should not call this method when only other methods than add/update/remove are called, since this could interfere with current animations.
SaveAsPDF	fileName : Text, options : JsonObject	Saves the entire chart into a PDF document that is downloaded after creation. Possibly the browser asks whether to wait for completion or not. The optional file name has to be pure (without any path information), and the file will be saved to the downloads folder of the browser by default. If no file name is specified, a new one is generated automatically. The optional options object can be used to specify additional properties for the export. The following properties are allowed: • `'Author' : Text` (default: undefined) • `'BottomHTML' : Text` (default: '' = empty string) • `'BottomPageMargin' : Decimal` (>= 0.0; default 10.0; in millimeters) • `'BottomText' : Text` (default: '' = empty string) • `'BottomTimescaleVisible' : Boolean` (default: False)

© 2026 Boyum IT Solutions GmbH

Method Name	Parameters	Description
		• 'CutMarksVisible' : Boolean (default: False) • 'End' : DateTime (default: value of setting End) • 'HorPageCountLimit' : Integer (default: 0=not active, if "zoomFactorInPercent" is 0, then 1) * • 'Keywords' : Text (default: undefined) • 'LeftPageMargin' : Decimal (>= 0.0; default 10.0; in millimeters) • 'OwnerPassword' : Text (default: undefined; if defined, you can edit the document in an appropriate application by entering this password) • 'PageFormat' : Text (default: 'A4'; possible values 'A0'/'A1'/'A2'/'A3'/'A4'/'A5'/'A6'/'Legal' /'Letter' or 'w*h' with width and height in millimeters) • 'PageOrientation' : Integer (default: Portrait; see enum PageOrientation) • 'PermissionToAnnotate' : Boolean (default:BTrue; if False, then it will not be possible to annotate text in the document) • 'PermissionToAssembleDocument' : Boolean (default:True; if False, then it will not be possible to combine the document with others) • 'PermissionToCopy' : Boolean (default: True; if False, then it will not be possible to copy text using the clipboard) • 'PermissionToCopyForContentAccessibility' : Boolean (default: True; if False, then it will not be possible to copy content for accessibility) • 'PermissionToModify' : Boolean (default: True; if False, then the PDF document can only be changed by the owner) • 'PermissionToPrint' : Text (default: 'highResolution'; possible values are 'lowResolution', 'highResolution', 'none'; if not set to 'lowResolution' or 'highResolution', then it will not be possible to print the document) • 'PrintingMode' : Integer (default: Cutting, see enum PrintingMode) • 'RightPageMargin' : Decimal (>= 0.0; default 10.0; in millimeters)

© 2026 Boyum IT Solutions GmbH

Method Name	Parameters	Description
		• `'Start' : DateTime` (default: value of setting Start) • `'Subject' : Text` (default: undefined) • `'Title' : Text` (default: undefined) • `'TopHTML' : Text` (default: '' = empty string) • `'TopPageMargin' : Decimal` (>= 0.0; default 10.0; in millimeters) • `'TopText' : Text` (default: '' = empty string) • `'UserPassword' : Text` (default: undefined; if given, then it is possible to read the PDF document only by entering the password in an appropriate viewer application) • `"VerPageCountLimit" : Integer` (default: 0=not active, if "zoomFactorInPercent" is 0, then 1) * • `"WatermarkSymbolID" : Text` (default: '' = empty string) • `"ZoomFactorInPercent" : Decimal` (default: 0=not active, else > 0) * In printing mode Single the control content is placed in one single page (ZoomFactorIn-Percent and Hor/VerPageCountLimit not respected). In printing mode Paging table and timescale are repeated on each page. In printing mode Cutting the pages are filled that way you can cut the pages and glue them. *If the ZoomFactorInPercent is 0 and at least one of Hor/VerPageCountLimit are 0, then this limit value(s) will be set to 1. This way only a minimum of parameters have to be set to get the expected output. Without setting any parameter you will get a single page as output. The properties Bottom/Left/Right/TopPageMargin define the margins that are left empty on each page of the PDF document. This serves for a proper layout for printing it later. The properties BottomText/TopText allow to specify additional texts for top/bottom frame area. The keywords {{#PageNo}}, {{#PageCount}}, {{#Date}} maybe used as placeholders.

© 2026 Boyum IT Solutions GmbH

Method Name	Parameters	Description
		These texts are only usable if the corresponding properties TopHTML/BottomHTML are not specified. The text is shown with the font size 10px. The properties BottomHTML/TopHTML allow to specify additional HTML content for top/bottom frame area. The keywords {{#PageNo}}, {{#PageCount}}, {{#Date}} maybe used as placeholders. Also the keyword {{@*symbolID*}} is a placeholder for a defined symbol and can be used in <image src="…"> to show a symbol if needed (other URLs to external images are also possible). If TopHTML and/or BottomHTML is used, then TopText and BottomText are not usable, respectively. Text is shown with the font size 14px if the style is not modified within the HTML code. Please ensure that the given HTML code is valid. The property BottomTimescaleVisible determines showing an additional timescale at the bottom of the chart in the PDF document. The property CutMarksVisible determines showing marks at the four corners of each page in order to make it possible to cut the empty margins of printed pages and put the pages together. This only makes sense in printing mode Cutting. The properties HorPageCountLimit and VerPageCountLimit determine a zoom factor for the chart indirectly by setting the limits of page count. It is possible to set one of the limits only or to leave them both zero. The properties PageFormat and PageOrientation determine the size and orientation of each page in the PDF document. The property WatermarkSymbolID allows to put a watermark on each page.

© 2026 Boyum IT Solutions GmbH

Method Name	Parameters	Description
		The property ZoomFactorInPercent determines the zoom factor for the chart, when not left zero. The properties start and end allow to narrow down the time range put into the PDF document. If not defined, the values of the appropriate options start and end, resp., are used. The properties Author, Keywords, Subject, Title, PermissionTo… are put into the PDF document properties. When the command is finished asynchronously the event OnSaveAsPDFFinished is triggered. See also method CancelSaveAsPDF and event OnSaveAsPDFProgress. ATTENTION: Method is not usable with the Windows client!
ScrollToDate	date : DateTime	Scrolls to the given date and time.
ScrollToDateWithOffset	date : DateTime, offset : Text	Scrolls to the given date and time. The view additionally will be scrolled back by the given offset to get a distance between the left margin of the time area view and the given date. The offset can be a string with • a number that specifies a number of pixels (e.g. "50px") • a percent string that specifies the size of the offset as a percentage of the time area view width (e.g. "10%").
ScrollToObject	objectType : ObjectType, objectID: Text, targetPositionInView : Integer (see TargetPositions), highlightingEnabled : Boolean	Scrolls to the object (activity/allocation/entity/resource). If the object is not visible, the corresponding rows are expanded automatically. When an object is provided, this can be the original object that is registered by one of the add or update methods, or it can be a new (literal) object, since only the properties ID and SkillID (in skilled resources view for resources and allocations) is read on it. When skilled resources view is visible, it is allowed to provide a SkillID property to highlight a scroll to a specific graphical object in the view. When the SkillID is not given, the

© 2026 Boyum IT Solutions GmbH

Method Name	Parameters	Description
		allocation bar in the assigned skill is addressed or the resource row of the first mentioned skill in property SkillID, resp. The parameter determines the position of the object in the view after scrolling to it; if set to None, only it will be scrolled if necessary to get the object visible. If highlightingEnabled is set to True, then a (eventually blinking) frame is shown until another method is called or a user interaction takes place. If set to False, then the current highlighting of objects will be removed. When migrating from 3.x, the new additional two parameters default to 0, True. See also settings ObjectHighlightingColor, -ObjectHighlightFlashingEnabled, ScrollTo-ObjectAnimationEnabled.
SelectObjects	objectType : ObjectType, objectIDs : JsonArray of Text, visualType: VisualType	Selects the given objects or the objects addressed by the given IDs. When objects are provided, these can be the original objects that are registered by one of the add or update methods, or they can be new (literal) objects, since only the properties ID and SkillID (in skilled resources view for resources and allocations, see below) are read on them. In the activity mode only activities and links can be selected. In the resource mode only resources and allocations can be selected. The parameter visualType is only required in the activity mode if objects of type Activity are to be selected. In this case you can define whether the activity rows (VisualType.Row) or the activity bars (VisualType.Bar) should be selected. It is possible to select objects that are hidden in the collapsed parent object. The event OnSelectionChanged is not triggered by the control add-in. When skilled resources view is visible, it is allowed to provide a SkillID property to select a specific graphical object in the view.

© 2026 Boyum IT Solutions GmbH

Method Name	Parameters	Description
SetCollapseStatesForEntityRows	newCollapseState : CollapseState, fromLevel : Integer, toLevel : Integer	Sets the collapse state of the rows in the entities table. Only entity rows within the given level range are considered. See also method SetCollapseStatesForRows for the rows in the primary table. This method does not trigger OnCollapseStateChanged events, since BC does not allow many events to be triggered in short time.
SetCollapseStatesForRows	viewType : ViewType, newCollapseState : CollapseState, fromLevel : Integer, toLevel : Integer, collapseStateTargets : CollapseStateTargets	Sets the collapse state of targets in the rows of the given view type. Only rows within the given level range are considered. By the parameter collapseStateTargets can be specified, whether the rows themselves (default), the allocation rows or the curve panes are affected. See also method SetCollapseStatesForEntityRows for the rows in the entities table. This method does not trigger OnCollapseStateChanged and OnCurveCollapseStateChanged events, since BC does not allow many events to be triggered in short time.

SetResourcePropertiesForActivities	properties : Array of JsonObject	Sets some additional graphical attributes and states for resources referenced by their ID shown below a referenced activity in the activities view.
		
		The objects in the array have the following profile:
		`{`
		`  ID : Text,`
		`  ActivityID: Text,`
		`  AllocationRowsCollapseState : CollapseState|undefined,`
		`  CurveCollapseState : CollapseState|undefined,`
		`  RowSymbolIDs : Array of Text|undefined`
		`  SortCode : Integer|Decimal|Text|DateTime|undefined,`
		`  TableColor : Text|undefined,`
		`  TableRowDefinitionID : Text|undefined`
		`}`
		
		If a property is set, then this setting will be used only for the resource row below the referenced activity. If a property is not set, then the property value with the same

© 2026 Boyum IT Solutions GmbH

Method Name	Parameters	Description
		name within the resource referenced by the ID will be used. This method can be used define some graphical attributes or states that are different between the resource rows below different activity rows.

SetResourcePropertiesForSkills	properties : Array of JsonObject	Sets some additional graphical attributes and states for resources referenced by their ID shown below a referenced skill.
		
		The objects in the array have the following profile:
		`{`
		`ID : Text,`
		`SkillID: Text,`
		`AllocationRowsCollapseState : CollapseState|undefined,`
		`CurveCollapseState : CollapseState|undefined,`
		`RowSymbolIDs : Array of Text|undefined`
		`SortCode : Integer|Decimal|Text|DateTime|undefined,`
		`TableColor : Text|undefined,`
		`TableRowDefinitionID : Text|undefined`
		`}`
		
		If a property is set, then this setting will be used only for the resource row below the referenced skill. If a property is not set, then the property value with the same name within the resource referenced by the ID will be used.
		
		This method can be used define some graphical attributes or states that are different between the resource rows below different skill rows.

SetSettings	settings : Settings	Sets settings.
SetTimeResolutionForView	unit : Text ('seconds', 'minutes', 'hours', 'days', 'weeks', 'months', 'quarters', 'years'), unitCount : Integer, start : DateTime	Sets the time resolution in the time area view. The time resolution is set so that the given unit, including unitCount, is recognizable for the user, e.g. TimeUnit.Hours sets the time resolution so that the user can differentiate every hour in the view. A unit count greater than 1 reduces the time resolution, e.g. TimeUnit.Hours with count 4 sets the time resolution so that the user can differentiate every 4 hours in the view. The timescale reflects the change accordingly. The time resolution is set independently from the width of the time area view.

© 2026 Boyum IT Solutions GmbH

Method Name	Parameters	Description
		If unitCount is undefined or lower or equal 0, then 1 is used. If start is set to 0DT, then the visible start remains unchanged. The method will be processed using animation when not called before the first call to the method *Render*. **Note:** The time resolution cannot be set finer than the maximum time resolution defined by the settings *MaximumTimeResolutionUnit* and *MaximumTimeResolutionUnitFactor*!
UpdateActivities	activities : JsonArray of Activity	Update activities. Allowed changes are modification of all properties besides ID. [1] Please see setting UpdateMode and enumeration UpdateModes for behavior.
UpdateAllocations	allocations : JsonArray of Allocation	Updates allocations. Allowed changes are modification of all properties besides ID. [1] Please see setting UpdateMode and enumeration UpdateModes for behavior.
UpdateCalendars	calendars : JsonArray of Calendar	Updates calendars visually. Allowed changes are modification of all properties besides ID. [1] Please see setting UpdateMode and enumeration UpdateModes for behavior.
UpdateContextMenus	contextMenus : JsonArray of ContextMenu	Updates context menus. It is allowed to modify all properties except ID.
UpdateCurves	curves : JsonArray of Curve	Updates curves. Allowed changes are modification of all properties but ID and Type. [1] Please see setting UpdateMode and enumeration UpdateModes for behavior.
UpdateDateLines	dateLines : JsonArray of DateLine	Updates date lines. Allowed changes are modification of all properties besides ID. [1] Please see setting UpdateMode and enumeration UpdateModes for behavior.
UpdateEntities	entities : JsonArray of Entity	Update entities. Allowed changes are modification of all properties besides ID. [1] Please see setting UpdateMode and enumeration UpdateModes for behavior.
UpdateLinks	links : JsonArray of Link	Updates links. Allowed changes are modification of all properties besides ID. [1]

© 2026 Boyum IT Solutions GmbH

Method Name	Parameters	Description
		Please see setting UpdateMode and enumeration UpdateModes for behavior.
UpdatePeriodHighlighters	links : JsonArray of PeriodHighlighter	Updates period highlighters. [1] Please see setting UpdateMode and enumeration UpdateModes for behavior.
UpdateResources	resources : JsonArray of Resource	Updates resources. Allowed changes are modification of all properties besides ID. [1] Please see setting UpdateMode and enumeration UpdateModes for behavior.
UpdateSkills	skills : JsonArray of Skill	Updates skills. Allowed changes are modification of all properties besides ID. [1] Please see setting UpdateMode and enumeration UpdateModes for behavior.
UpdateSymbols	symbols : JsonArray of Symbol	Updates symbols. Allowed changes are modification of all properties besides ID. [1] Please see setting UpdateMode and enumeration UpdateModes for behavior.
UpdateTableRowDefinitions	tableRowDefinitions : JsonArray of TableRowDefinition	Updates table row definitions. Allowed changes are modification of all properties besides ID. [1] Please see setting UpdateMode and enumeration UpdateModes for behavior.
UpdateTooltipTemplates	tooltipTemplates : JsonArray of TooltipTemplate	Updates tooltip templates. Allowed changes are modification of all properties besides ID.[1] Please see setting UpdateMode and enumeration UpdateModes for behavior.

5.2 Events

Events are triggered on the start of the control add-in and on user interactions.

Attention: Since there is an issue with the AL environment at least at the moment, dates have to be handled with extra effort. You can see examples in the demonstration app.

At page start-up, the first event sent is OnRequestSettings, followed by OnControlAddInReady after using the method SetSettings.

Event Name	Description
CanDrag(EventArgs)	This event is triggered when the user is moving the mouse cursor over an activity/allocation or touches an activity/allocation with a finger.

© 2026 Boyum IT Solutions GmbH

Event Name	Description

	In order to reduce the number of events sent from the web client to the BC server, the application can switch this event off by using the setting CanDragEventTriggeringEnabled.
	
	EventArgs object is a JsonObject with the following keys/values:
	
	`'ObjectType' : ObjectType,`
	`'ObjectID' : Text,`
	`'VisualType' : VisualType,`
	`'AllowedDragModes' : BarDragModes|RowDragModes,`
	`'SelectedObjectIDs' : JsonArray of Text`
	`'StartPropertyName' : Text, *1`
	`'EndPropertyName' : Text, *1`
	`'SkillID' : Text, *2`
	`'ActivityID' : Text, *3`
	
	If the application sets the setting ForcedActivity/AllocationAllowedBarDragModes or ForcedActivity/Entity/Resource-AllowedRowDragModes, resp., to None, then no dragging will be possible. AllowedDragModes contains the value of the property Allowed(Row/Bar)DragModes of the object to drag.
	
	If the setting MultipleBarDraggingEnabled is set to True and more than one bar is selected, the property SelectedObjects will contains all selected objects, so that the application can determine the value for AllowedDragModes.
	
	If the mouse touches a date symbol or bar of an activity, then the properties Start/EndPropertyName contain the name of the property to be modified when dragging or dropping the symbol or bar, resp.
	
	This callback is called only once every time when the mouse enters the visual representation of the object (bar).
	
	*1: Available only for activity bars.
	*2: Available only for allocations and resources in skilled resources view.
	*3: Available only for allocations in activities view, when resource rows are visible.

OnClicked(EventArgs)	This event is triggered when an object is clicked by the user.
	
	EventArgs object is a JsonObject with the following keys/values:
	
	`'ObjectType' : ObjectType|Text`
	`'ObjectID' : Text`
	`'VisualType' : VisualType`
	`'EntryIndex' : Integer, *1`

© 2026 Boyum IT Solutions GmbH

Event Name	Description
	'Date' : DateTime, *2 'PeriodHighlighterID' : Text, *3 'PeriodHighlighterEntryIndex' : Integer, *3 'CellIndex': Integer, *4 'SymbolIndex' : Integer, *5 'SymbolID' : Text, *5 'HierarchySupplementaryDefinitionID' : Text, *6 'HierarchyLevelSupplementaryDefinitionIndex' : Integer, *6 'GroupingLevelDefinitionIndex' : Integer, *6 'GroupingCodes' : JsonArray of Text, *6 'SkillID' : Text, *7 'ActivityID' : Text, *8 'PointerType' : Text, *9 ('mouse'|'touch'|'pen'|undefined) 'AltKey' : Boolean|undefined, *9 'CtrlKey' : Boolean|undefined, *9 'MetaKey' : Boolean|undefined, *9 'ShiftKey' : Boolean|undefined *9 On grouping rows, the ObjectID is '' (empty string). In skilled resources view the argument SkillID is set when clicking on a resource row, allocation row, or allocation bar. On clicks within the status areas the application receives this event with VisualType set to -1, ObjectID is the ID of the clicked DOM element and ObjectType is set to 'TopStatusArea' or 'BottomStatusArea'. *1: Available only for activity and allocation bars. *2: Available only if interaction in time area. *3: Available only if VisualType === VisualType.PeriodHighlighter. *4: Available only if VisualType === VisualType.Row. *5: Available only if VisualType === VisualType.Row and symbol in symbol column is clicked. *6: Available only set when this event is referencing a grouping row (AL version only!). The ObjectID property then always is empty. *7: Available only for allocations and resources in skilled resources view. *8: Available only for allocations in activities view, when resource rows are visible. *9: Taken from DOM event object.
OnCollapseStateChanged(EventArgs)	This event is triggered when a group was expanded or collapsed either in the table of the Gantt diagram or of the entities table. This can happen: • by the user clicking on the appropriate symbol in the resource, activity, skill, or entity row • by automatic row expansion when dragging objects • by using the method ScrollToObject • by setting the property CollapseState on a resource, an activity, a skill, or an entity object

© 2026 Boyum IT Solutions GmbH

Event Name	Description

	and setting TriggeringOfOnCollapseStateChangedByUpdateCallsEnabled is not set to false.
	
	EventArgs object is a JsonObject with the following keys/values:
	
	`'ObjectType' : ObjectType`
	`'ObjectID' : Text`
	`'NewCollapseState' : CollapseState`
	`'IsForAllocationRows' : Boolean`
	`'Interactively' : Boolean`
	`'SkillID' : Text, *1`
	`'ActivityID' : Text, *2`
	
	In skilled resources view the argument SkillID is set when clicking on a resource row, allocation row, or allocation bar.
	
	*1: Available only for allocations and resources in skilled resources view.
	*2: Available only for allocations in activities view, when resource rows are visible.

OnContextMenuItemClicked(EventArgs)	This event is triggered when a context menu item was clicked by the user.
	
	EventArgs object is a JsonObject with the following keys/values:
	
	`'ContextMenuID' : Text`
	`'ContextMenuItemIndex' : Integer`
	`'ContextMenuItemCode' : Text|undefined`
	`'ObjectType' : ObjectType`
	`'ObjectID' : Text|undefined`
	`'VisualType' : VisualType`
	`'Date' : DateTime`
	`'TimePeriodStart' : DateTime|undefined,`
	`'TimePeriodEnd' : DateTime|undefined,`
	`'SkillID' : Text, *1`
	`'ActivityID' : Text, *2`
	`'SymbolID' : Text, *3`
	`'SymbolIndex' : Text *4`
	
	In skilled resources view the argument SkillID is set when clicking on a resource row, allocation row, or allocation bar.
	
	In the case of the timescale, the start and end dates of the associated time period are supplied in addition to the date actually hit (see TimePeriodStart and TimePeriodEnd).
	
	Within the status areas the application receives this event with VisualType set to -1, ObjectID is the ID of the addressed DOM element and ObjectType is set to 'TopStatusArea' or 'BottomStatusArea'.

© 2026 Boyum IT Solutions GmbH

Event Name	Description
	*1: Available only for allocations and resources in skilled resources view. *2: Available only for allocations in activities view, when resource rows are visible. *3: Available only on symbols. *4: Available only on symbols in the symbol column of a row.
OnControlAddInReady()	This event is triggered after the control add-in is instantiated and has received settings (at least start and end dates for the time area to show). The control add-in is ready to get data.

OnCurveCollapseStateChanged(EventArgs)	This event is triggered when a curves pane was expanded or collapsed table of the Gantt diagram. This callback is triggered by the user clicking on the appropriate symbol in the resource row.
	
	EventArgs object is a JsonObject with the following keys/values:
	
	`'ObjectType' : ObjectType,`
	`'ObjectID' : Text,`
	`'NewCollapseState' : CollapseState,`
	`'SkillID' : Text, *1`
	`'ActivityID' : Text *2`
	
	In skilled resources view the argument SkillID is set when clicking on a resource row, allocation row, or allocation bar.
	
	*1: Available only for allocations in activities view, when resource rows are visible.
	*2: Available only for allocations in skilled resources view.

OnCurvePaneResized(EventArgs)	This event is triggered after the height of a curve pane has been changed interactively.
	
	If the application does not update the property LoadCurvePaneHeight of the resource within the event handler code, then the resizing is made undone.
	
	EventArgs object is a JsonObject with the following keys/values:
	
	`'ObjectType' : ObjectType,`
	`'ObjectID' : Text,`
	`'NewHeight' : Integer`

OnDoubleClicked(EventArgs)	This event is triggered when an object is double-clicked by the user.
	
	EventArgs object is a JsonObject with the following keys/values:
	
	`'ObjectType' : ObjectType`
	`'ObjectID' : Text`

© 2026 Boyum IT Solutions GmbH

Event Name	Description
	'VisualType' : VisualType 'EntryIndex' : Integer, *1 'Date' : DateTime, *2 'PeriodHighlighterID' : Text, *3 'PeriodHighlighterEntryIndex' : Integer, *3 'CellIndex': Integer, *4 'SymbolIndex' : Integer, *5 'SymbolID' : Text, *5 'HierarchySupplementaryDefinitionID' : Text, *6 'HierarchyLevelSupplementaryDefinitionIndex' : Integer, *6 'GroupingLevelDefinitionIndex' : Integer, *6 'GroupingCodes' : JsonArray of Text, *6 'SkillID' : Text, *7 'ActivityID' : Text, *8 'PointerType' : Text, *9 ('mouse'|'touch'|'pen'|undefined) 'AltKey' : Boolean|undefined, *9 'CtrlKey' : Boolean|undefined, *9 'MetaKey' : Boolean|undefined, *9 'ShiftKey' : Boolean|undefined *9 On grouping rows, the ObjectID is '' (empty string). In skilled resources view the argument SkillID is set when clicking on a resource row, allocation row, or allocation bar. On double-clicks within the status areas the application receives this event with VisualType set to -1, ObjectID is the ID of the clicked DOM element and ObjectType is set to 'TopStatusArea' or 'BottomStatusArea'. *1: Available only for activity and allocation bars. *2: Available only if interaction in time area. *3: Available only if VisualType === VisualType.PeriodHighlighter. *4: Available only if VisualType === VisualType.Row. *5: Available only if VisualType === VisualType.Row and symbol in symbol column is clicked. *6: Available only set when this event is referencing a grouping row (AL version only!). The ObjectID property then always is empty. *7: Available only for allocations and resources in skilled resources view. *8: Available only for allocations in activities view, when resource rows are visible. *9: Taken from DOM event object.
OnDragEnd(EventArgs)	This event I triggered when the user finishes a dragging action either by dropping the dragged object or by canceling the action using the Escape key. EventArgs object is a JsonObject with the following keys/values: 'ObjectType' : ObjectType 'ObjectID' : Text

© 2026 Boyum IT Solutions GmbH

Event Name	Description
	'VisualType' : VisualType 'DragMode' : BarDragModes 'StartPropertyName' : Text, *1 'EndPropertyName' : Text, *1 'SkillID' : Text, *2 'ActivityID' : Text, *3 'PointerType' : Text, *4 ('mouse'|'touch'|'pen'|undefined) 'AltKey' : Boolean|undefined, *4 'CtrlKey' : Boolean|undefined, *4 'MetaKey' : Boolean|undefined, *4 'ShiftKey' : Boolean|undefined *4 If a date symbol or bar of an activity was dragged, then the properties Start/EndPropertyName contain the name of the property to be modified when a dragging or dropping the symbol or bar, resp. *1: Available only for activity bars. *2: Available only for allocations and resources in skilled resources view. *3: Available only for allocations in activities view, when resource rows are visible. *4: Taken from DOM event object.
OnDragStart(EventArgs)	This event I triggered when the user starts a dragging action. EventArgs object is a JsonObject with the following keys/values: 'ObjectType' : ObjectType 'ObjectID' : Text 'VisualType' : VisualType 'DragMode' : BarDragModes 'StartPropertyName' : Text, *1 'EndPropertyName' : Text, *1 'SkillID' : Text, *2 'ActivityID' : Text *3 If a date symbol or bar of an activity will be dragged, then the properties Start/EndPropertyName contain the name of the property to be modified when dragging or dropping the symbol or bar, resp. *1: Available only for activity bars. *2: Available only for allocations and resources in skilled resources view. *3: Available only for allocations in activities view, when resource rows are visible.
OnDrop(EventArgs)	This event is triggered when an activity/allocation is dropped by the user after dragging it. EventArgs object is a JsonObject with the following keys/values: 'ObjectType' : ObjectType 'ObjectID' : Text 'VisualType' : VisualType 'ProposedMethodCalls' : JsonArray of JsonObject

© 2026 Boyum IT Solutions GmbH

Event Name	Description
	'DragMode' : BarDragModes|RowDragModes 'NewRowObjectType' : ObjectType 'NewRowObjectID' : Text 'RowInsertionMode': RowInsertionMode 'NewSortCode': Integer 'NewStart' : DateTime 'NewEnd' : DateTime 'NewDate' : DateTime 'StartPropertyName' : Text, *1 'EndPropertyName' : Text, *1 'WorkingTimeDistance' : Integer, /*in milliseconds*/ 'CoupledObjectIDs' : JsonArray of Text 'StartsAndEndsOfCoupledObjects' : JsonArray of JsonObject 'StartsAndEndsOfEntries' : JsonArray of JsonObject 'OtherNewSortCodesAndRowObjects' : JsonArray of JsonObject 'SkillID' : Text, *2 'NewSkillID' : Text, *2 'ActivityID' : Text, *3 'NewActivityID' : Text, *3 'PointerType' : Text, *4 ('mouse'|'touch'|'pen'|undefined) 'AltKey' : Boolean|undefined, *4 'CtrlKey' : Boolean|undefined, *4 'MetaKey' : Boolean|undefined, *4 'ShiftKey' : Boolean|undefined *4 NewStart/NewEnd are filled for dragged bars, while NewDate is filled for dragged date lines. The property ProposedMethodCalls contains an array of objects of the form: { MethodName : string, Args : any[] }. It can be empty e.g. when a row object is dropped onto itself. In other cases it contains the proposal to the methods to call for processing the drop action. The first object currently is always an Update… or SetResourcePropertiesFor… method. The second object then is a call to the Render method and the third one is a call to ScrollToObject. The application then can inform the backend and it can loop through the array and call the methods. These actions are supported: • Dragged table rows of all appropriate object types (modifying properties ParentID, SortCode; on sibling objects also). For resources in skilled resources view the method SetResourcePropertiesForSkills is used). For resources in activities view the method SetResourcePropertiesForActivities is used). • Dragged allocation bars vertically and horizontally (modifying properties ActivityID or ResourceID, and/or modifying AllocationEntry properties Start/End). • Dragged activity bars vertically and horizontally (modifying properties ParentID, Start/End,

© 2026 Boyum IT Solutions GmbH

Event Name	Description

	LinkSourceDate, LinkTargetDate, and eventually modifying ActivityEntry properties Start/End).
	• Dragged date lines (modifying property PointInTime).
	
	When the setting MultipleBarDraggingEnabled is set to True and more than one object was dragged, then the properties CoupledObjectIDs and StartsAndEndsOfCoupledObjects are set. The latter one contains JsonObjects with the following properties:
	`'ObjectID' : Text,`
	`'NewStart' : DateTime,`
	`'NewEnd' : DateTime`
	
	Remark: If one of the properties NewStart or NewEnd hat a value of 0DT, then the user dragged this object outside of the visible time area and there is no working time in the calendar to calculate the appropriate date.
	
	If a date symbol or bar of an activity is dropped, then the properties Start/EndPropertyName contain the name of the property to be modified when dragging or dropping the symbol or bar, resp.
	
	The property startsAndEndsOfEntries contains an array of objects of the following form:
	`{`
	`  EntryIndex : Integer,`
	`  NewStart : DateTime,`
	`  NewEnd : DateTime`
	`}`
	
	When multiple bars are dragged and dropped, this also applies to coupled objects: In the objects of the property startsAndEndsOfCoupledObjects there also exists a property named startsAndEndsOfEntries.
	
	If the VisualType is Row, then the property RowInsertionMode is set. If the sort mode for this type of row object is set to Ascending (see settings Activity/Entity/ResourceSortCodeSources), then also the properties NewSortCode and OtherNewSortCodesForSiblingObjects are set. The latter is a JsonArray of JsonObjects with the following properties:
	`'ObjectID' : Text`
	`'NewSortCode' : Integer`
	
	This array contains all sibling objects for which is new sort code is needed.
	
	*1: Available only for activity bars.

© 2026 Boyum IT Solutions GmbH

Event Name	Description
	*2: Available only for allocations and resources in skilled resources view. *3: Available only for allocations in activities view, when resource rows are visible. *4: Taken from DOM event object.

OnLogError(EventArgs)	This event is triggered when an exception occurs on any method that is called on the control add-in. The exception than has occurred within the client running on a JavaScript engine.
	
	EventArgs object is a JsonObject with the following keys/values:
	
	`'CommandName' : Text,`
	`'ErrorName' : Text,`
	`'ErrorMessage' : Text,`
	`'ErrorStack' : Text`
	
	You can use this in your application to make the exceptions persistent. Normally no error exception should be triggered at all. See also event OnLogWarnings.

OnLogWarnings(EventArgs)	This event is triggered when data is inconsistent among other incidents. The warnings are generated within the client running on a JavaScript engine. To reduce the number of events, only one event is called per method call, that shows all warning texts at once.
	
	EventArgs object is a JsonObject with the following keys/values:
	
	`'CommandName' : Text,`
	`'WarningTexts' : JsonArray of Text`
	
	The CommandName property contains the pure current method name. The description contains an English text like “Option "xyz" is unknown”, “Object ID empty”, or “Object with ID "xyz" not unique”. The texts are prefixed with a warning code, see enumeration WarningCode.
	
	You can use this in your application to debug your application or to make the warnings persistent. Normally no warning should be triggered at all. See also event OnLogError.

OnPing()	This event is triggered in an interval of number of milliseconds, where the number is given by the setting PingTimeInMilliseconds.

© 2026 Boyum IT Solutions GmbH

Event Name	Description
OnRequestSettings(EventArgs)	This event is triggered after instantiation of the control add-in and only when the method SetSettings was not called before. EventArgs object is a JsonObject with the following keys/values: `'Version' : Text`

OnRowSortingChangeRequested(EventArgs)	This event is triggered when the user clicks or taps on a column in the table title and the option interactiveSwitchingOfSortOrderEnabled is set to true.
	
	EventArgs object is a JsonObject with the following keys/values:
	
	`'ObjectType' : ObjectType,`
	`'SortMode' : RowSortMode,`
	`'SortCodeSource' : Text`

OnSaveAsPDFFinished()	This event is triggered when the method SaveAsPDF has finished.

OnSaveAsPDFProgress(EventArgs)	This event is triggered constantly during the execution of the SaveAsPDF method. Especially when saving a diagram to a PDF with many pages, this callback is helpful for the application to be continuously informed about the progress of the processing.
	
	EventArgs object is a JsonObject with the following keys/values:
	
	`'PageCount' : Integer,`
	`'CurrentPage' : Integer`

OnSelectionChanged(EventArgs)	This event is triggered when the user selects/deselects an object solely or in addition.
	
	EventArgs object is a JsonObject with the following keys/values:
	
	`'ObjectType' : ObjectType|0,`
	`'SelectedObjectIDs' : JsonArray of Text`
	`'VisualType' : VisualType`
	`'ObjectID' : Text`

OnTableCellDefinitionWidthChanged(EventArgs)	This event is triggered when a table column width is changed interactively.
	
	EventArgs object is a JsonObject with the following keys/values:
	
	`'TableType' : TableType`
	`'TableRowDefinitionID' : Text`
	`'CellIndex' : Integer`
	`'OldWidth' : Integer`
	`'NewWidth' : Integer`

© 2026 Boyum IT Solutions GmbH

Event Name	Description

OnTimeAreaViewParametersChanged(EventArgs)	This event is triggered when the time area view is changed, e.g. by scrolling, changes in the time resolution, view size.
	
	The end date itself is not included in the visible time area. The same behavior also is true for all end dates used in the widget like e.g. in calendar entries, period highlighter entries, activities, activity entries, allocations, allocation entries.
	
	EventArgs object is a JsonObject with the following keys/values:
	
	`'HorizontalScrollOffset' : Integer`
	`'ScrollOffset' : Integer /* deprecated */`
	`'Width' : Integer`
	`'Start' : DateTime`
	`'End' : DateTime`
	`'TimeResolutionUnit' : Text`
	`'TimeResolutionUnitCount' : Decimal`
	`'TableViewWidth' : Integer /* current width, not to be confused with the setting TableViewWidth */`
	`'EntitiesTableViewWidth' : Integer /* current width, not to be confused with the setting EntitiesTableViewWidth */`

OnVerticalScrollOffsetChanged(EventArgs)	This event is triggered when the Gantt area or the entities table is scrolled vertically.
	
	EventArgs object is a JsonObject with the following keys/values:
	
	`'TableType' : TableType`
	`'ScrollOffset' : Integer`
	`'RowObjectTypeAtTop' : ObjectType`
	`'RowObjectIDAtTop' : Text`

6 Enumerations

Note: For the time being, we have not defined options or enumerations for the types. So, you will have to use the number literals in your application.

6.1 BarDesigns

```
Simple: 0
Default: 65535,                 // Bits 0 to 15;
                                //   this value leaves some bits reserved for future extensions
DefaultReduced: 16711680,       // Bits 16 to 23;
                                //   this value leaves some bits reserved for future extensions
```

Note: Values are flags, i.e. they can be combined by using bitwise OR operators.

```
Entries: 1,                     // Bit 0
ComplexShape: 2,                // Bit 1; currently only Regular is changed to Rectangle when unset
Symbols: 4,                     // Bit 2
Status: 8,                      // Bit 3
Constraints: 16,                // Bit 4
```


© 2026 Boyum IT Solutions GmbH

```
ReleaseAndDueDateSymbols: 32,     // Bit 5; ignored on allocations
Baseline: 64,                     // Bit 6; ignored on allocations
ProgressAndPredictedEnd: 128,     // Bit 7
Text: 256,                        // Bit 8
TonedDownColoring: 65536          // Bit 16; see setting TonedDownOverlayColor
ReducedHeight: 131072             // Bit 17; see setting ReducedBarTopOffsetAndHeightScaleFactor
```

6.2 BarDragModes

Note: Values are flags, i.e. they can be combined by using bitwise OR operators.

```
None: 0
DragStart: 1
DragEnd: 2
DragHorizontally: 4
DragVertically: 8
DragAutoHorOrVer: 16
DragSmartHorOrVer: 256
```

DragAutoHorOrVer: When starting to drag, the user can decide to drag horizontally or vertically. After that the drag direction is fixed and can be overridden by pressing the SHIFT key.

DragSmartHorOrVer: Like DragAutoHorOrVer, but additionally the user can change the drag direction afterwards by moving the cursor in the other direction. So, the user can e.g. first choose a row and the choose a time range for a bar, but in the same drag action.

6.3 BarShape

```
Regular: 0
Summary: 1
Diamond: 2
Rectangle: 3
Symbol: 4
```



When the shapes Regular, Summary, or Rectangle are used, then the bar will span the time range defined by the Start and End properties of the allocation or activity and by the time spans defined by the contained entries. When the shapes Diamond or Symbol are used, the shape is set to the start date of the same time span.

When the shapes Diamond or Symbol are used, then the bar text is shown only when the bar is visible within an expanded row, otherwise the bar text is shown unconditionally.

6.4 BarSortMode

`StartAndEnd: 0` (Bars are shown sorted by start and end. This means that a bar that starts earlier than another bar is placed behind the latter bar. For bars that start at the same time, the longer bar is placed behind the shorter one)
`ByRowSorting: 1` (Bars are sorted by considering the appropriate setting in one of the settings Activity/AllocationSortCodeSources)
`ByRowSortingOnSameStart: 2` (Bars are sorted as in StartAndEnd, but for two bars with the same start, they are sorted by considering the appropriate setting in one of the settings Activity/AllocationSortCodeSources)

© 2026 Boyum IT Solutions GmbH

6.5 CollapseState

`Unchanged: -1 /*` (when used on startup, allocation rows will be shown collapsed, other row types will be shown expanded, and curves will be shown collapsed) */
`Expanded: 0`
`Collapsed: 1`

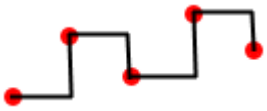
6.6 CollapseStateTargets

Note: Values are flags, i.e. they can be combined by using bitwise OR operators.

```
Default: 1
Rows: 1
AllocationRows: 2
CurvePanes: 4
```

6.7 CurveInterpolationType

`StepAfter: 0`



`Linear: 1`



6.8 CurveType

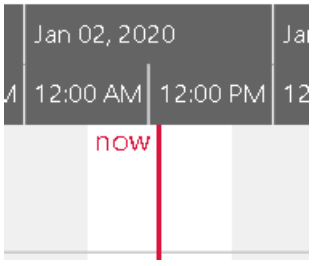
```
PointCurve: 0
CurveStack: 3
CurveList: 4
```

6.9 DateLineAppearanceType

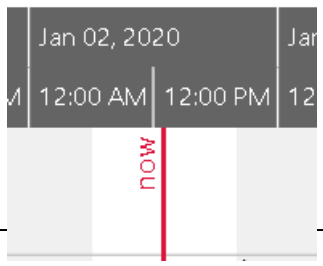
```
Permanent: 0
PointerOverAttachedBar: 1
```

6.10 DateLineCaptionOrientation

`Horizontal: 1`



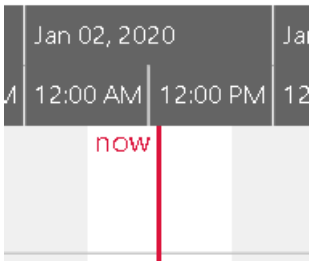
`Vertical: 2`



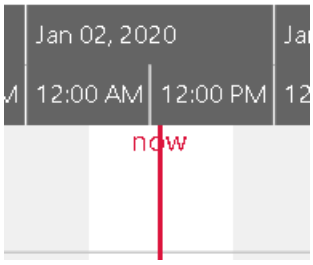
© 2026 Boyum IT Solutions GmbH

6.11 DateLineCaptionPosition

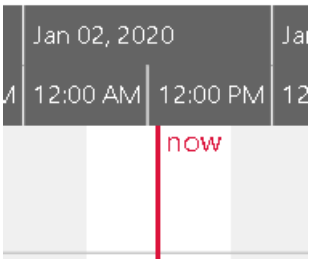
Left: 1



Center: 2

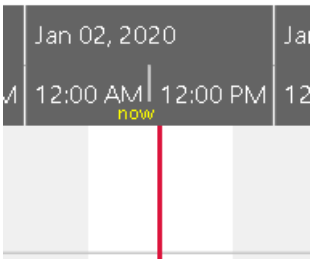


Right: 4



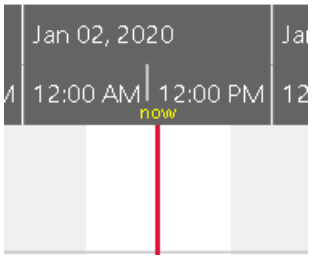
TopLeft: 9

```
// inside
// timescale
// area
```



TopCenter: 10

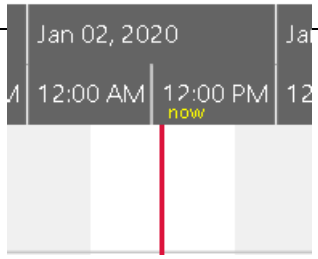
```
// inside
// timescale
// area
```



© 2026 Boyum IT Solutions GmbH

```
TopRight: 12

// inside
// timescale
// area
```



6.12 DateLineGridModes

```
None: 0
Auto: 1
Weekly: 2
Daily: 4
```

6.13 DayOfWeek

```
Sunday: 0
Monday: 1
Tuesday: 2
Wednesday: 3
Thursday: 4
Friday: 5
Saturday: 6
```

6.14 HorizontalAlignment

```
Left: 0
Center: 1
Right: 2
```

6.15 HorizontallyScrollableViewArea

```
Table: 0
TimeArea: 1
EntitiesTable: 2
```

6.16 HorizontalScrollPosition

```
Left: 1
Right: 2
```

6.17 LinkMarker

```
None: 0
FilledArrow: 1
```

6.18 LinkRoutingType

```
Curved: 1
Orthogonal: 2
```


© 2026 Boyum IT Solutions GmbH

6.19 ObjectType

```
TimeArea: -2
Timescale: -1
Activity: 1
Allocation: 2
Resource: 5
Link: 6
Curve: 7
Entity: 13
PeriodHighlighter: 14
Symbol: 15
DateLine: 16
TooltipTemplate: 17
TableRowDefinition: 18
Calendar: 20
HierarchySupplementaryDefinition: 21
Skill: 22
ContextMenu: 100
```

6.20 PageOrientation

```
Portrait: 0
Landscape: 1
```

6.21 PanningMode

```
None: 0
HorizontallyOnly: 1
VerticallyOnly: 2
HorAndVer: 3
AutoHorOrVer: 4
```

6.22 PatternType

```
VerticalHatch: 0
ForwardHatch: 1
BackwardHatch: 2
```

6.23 PrintingMode

Note: See also here for an overview of the different modes.

```
Single: 0
Cutting: 1
Paging: 2
```

6.24 ProgressBarWidthCalculationMode

```
ConsiderWorkingTimesOnly: 0
ConsiderWorkingAndNonworkingTimes: 1
```

6.25 RelationType

```
FinishToStart: 0
FinishToFinish: 1
```


© 2026 Boyum IT Solutions GmbH

```
StartToStart: 2
StartToFinish: 3
SourceDateToStart: 4
SourceDateToFinish: 5
FinishToTargetDate: 8
StartToTargetDate: 10
SourceDateToTargetDate: 12
```

6.26 RowDesigns

Note: Values are flags, i.e. they can be combined by using bitwise OR operators.

```
Empty: 0
Bars: 1                              // Shows bars assigned to row object directly
BarsStacked: 2                       // Shows all bars without vertical overlapping
BarsInHiddenDescendantRows: 4        // Shows bars of other hidden descendant rows
CalendarGrid: 8                      // Shows calendar grid of row object
```

Further explanation:

- Empty: No content is displayed in the row section of the time range.
- Bars: In the activities view, the bar in an activity row represents the same object as the row itself. In the resources view, the bars represent the allocations that are assigned to the resource represented by the row. In the loads view there are generally no bars shown. For allocation rows in the activities view or resources view the bars represent the same object as the row itself. If this flag is set then the bars are shown, else they remain invisible. When the flag BarsStacked is not set, then the bars in one row are sorted with BarSortMode.StartAndEnd.
- BarsInHiddenDescendantRows: This flag is only effective when the row is shown in collapsed state but does no harm when the row is shown in expanded state. Bars of hidden rows below the collapsed row are projected into the row.
- BarsStacked: When bars are shown in this row then they will be shown stacked vertically so that they do not overlap graphically when they allocate common time ranges. When bars are stacked the row is getting higher. When the flag is not set then the bars will graphically overlap and the row height is kept stable. Stacking only exists when more than one bar is shown, which is the case e.g. for resource rows and collapsed activity rows. When the flag is not set, then the bars in a row are sorted with BarSortMode.StartAndEnd, else see settings. ActivityBarSortModeForStackedRowDesign and AllocationBarSortModeForStackedRowDesign.
- CalendarGrid: If this flag is set then the calendar assigned to the object represented by the row is made visible through a so-called calendar grid.

6.27 RowDragModes

Note: Values are flags, i.e. they can be combined by using bitwise OR operators.

```
None: 0
DragVertically: 8
DragOutside: 32
DragOnSameLevelOnly: 64
DragInSameTableParentOnly: 128
```

Further explanation:

- “DragVertically”: Dragging is possible within the same table.
  When no restriction flag is set, then it is possible to drop a row object above or below the

© 2026 Boyum IT Solutions GmbH

targeted row object or as the first child of the targeted row object. The user can switch the insertion mode by using the Shift key.

When dragging the last row in the chart downwards, the dragged row object is put one level above or when the Shift key is pressed, to the topmost level, when no restriction flag is set.

- "DragOutside": Dragging is possible leaving the table. Currently this is only possible for entities in the entities table.
- "DragOnSameLevelOnly": Restriction to DragVertically: When set additionally to DragVertically, then a row can only be dropped on the same level as it was before. Not combinable with DragInSameTableParentOnly.
- "DragInSameTableParentOnly": Restriction to DragVertically: When set additionally to DragVertically, then a row can only be dropped below the same direct ancestor within the visual table (e.gh. for activities it is the parent, while for allocation rows it can be either the resource or the activity depending on the view). Not combinable with DragOnSameLevelOnly.

6.28 RowInsertionMode

```
None: 0
InsertAsChild: 1
InsertAsNextSibling: 2
InsertAsPreviousSibling: 3
```

6.29 RowSortMode

```
None: 0
Ascending: 1
Descending: 2
AscendingStartAndEnd: 3 // only available for activities and allocations.
```

6.30 SnapTargets

Note: Values are flags, i.e. they can be combined by using bitwise OR operators.

`None: 0`
`Start: 1 //` Snapping at the start of other bars in the same row and in child rows. Only valid for bars representing allocations.
`End: 2 //` Snapping at the end of other bars in the same row and in child rows. Only valid for bars representing allocations.
`DateLines: 4 //` Snapping at objects of type DateLine.
`CalendarGrids: 8 //` Snapping at starts and ends of non-working time.
`DateLineGrids: 16 //` Snapping at date lines shown by the setting DateLineGridMode.

6.31 TableType

```
Gantt: 0
Entities: 1
```

6.32 TargetPositions

Note: Values are flags, i.e. they can be combined by using bitwise OR operators.

© 2026 Boyum IT Solutions GmbH

```
None: 0
Left: 1
HCenter: 2
Right: 4
Top: 8
VCenter: 16
Bottom: 32
NoHScroll: 64
```

6.33 TextWrapMode

```
None: 0 // no wrapping at all
Line: 1 // text is wrapped at LF characters (ASCII code 10)
```

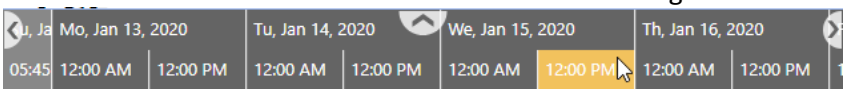
6.34 TimescaleInteractionModes

Note: Values are flags, i.e. they can be combined by using bitwise OR operators.

```
None: 0               // no interactions at all
ScrollingByButtons: 1 // horizontal scrolling by using scrolling buttons
Rescaling: 2          // rescaling by time period selection, up-button or
                      // mouse wheel
Default: 3            // all interactions
```

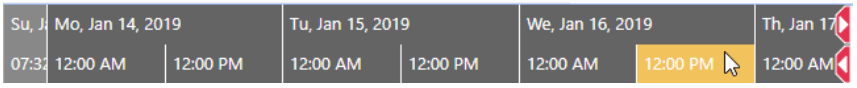
6.35 TimescaleNavigationMode

`Latest: 0` `// use the latest version of the timescale navigation`



- A click onto the left and right button scrolls the chart sideward by the width of the view.
- A click onto the up button reduces the timescale resolution.
- A click onto a highlighted period (see orange area) fits this period completely into the view.
- Use the mouse wheel for increasing and reducing the timescale resolution.

`LegacyVersion1: 1`



- A click onto the left and right button scrolls the chart sideward by the widths of one unit in the upper timescale ribbon
- A click onto a highlighted period (see orange area) fits this period completely into the view.
- Use the mouse wheel for increasing and reducing the timescale resolution.

6.36 TimeType

```
WorkingTime: 1
NonworkingTime: 2
```


© 2026 Boyum IT Solutions GmbH

6.37 TreeVisualizationMode

```
ColoredIndentation: 0
TreeViewLines: 1
```

6.38 UpdateModes

Note: Values are flags, i.e. they can be combined by using bitwise OR operators.

```
UpdateOnly: 0          // deprecated, use Default
Default: 0
ImplicitAddObjects: 1  // If an object to be updated does not exist,
                       // it will be added automatically.
DifferentialValues: 2  // If set, then the object data given in the update
                       // method can contain only changed property values. If a
                       // property is omitted, it will be supplemented by the
                       // value of the property in the current object. If a
                       // property value shall be set to undefined explicitly,
                       // please use a null value instead or another value that
                       // can be set by the setting
                       // ResetValueForDifferentialUpdate.*
```

* When using this flag, the application will have to use simple objects in the update methods, because the VSW will modify them and supplement missing property values. Also, these objects have to be different to the ones given in former calls to the appropriate add or update method.

Please be aware that sub objects in arrays like entries must be fully defined despite of the active update mode.

6.39 VerticalAlignment

```
FirstLineOnBaseline: 0
AllLinesCenteredAroundBaseline: 1
```

6.40 VerticallyScrollableViewArea

```
Top: -1,
Main: 0,
EntitiesTable: 2
```

6.41 VerticalScrollPosition

```
Top: 1
Bottom: 2
```

6.42 ViewArea

```
Top: -1
Default: 0
```

6.43 ViewType

```
Activities: 0
```


© 2026 Boyum IT Solutions GmbH

```
Resources: 1
Loads: 2
SkilledResources: 3
```

6.44 ViewTypesForDateLines

Note: Values are flags, i.e. they can be combined by using bitwise OR operators.

```
None: 0
ActivitiesView: 1
ResourcesView: 2
LoadsView: 4
SkilledResourcesView: 8
AllViews: 15
```

6.45 VisualType

```
Background: -1
Bar: 0
Row: 1
Curve: 2
Link: 3
PeriodHighlighter: 4
DateLine: 5
Timescale: 6
```

6.46 WarningCode

Key	Value	Explanation
Miscellaneous	'W0'	Other uncategorizable warnings like messages from PDF export libraries.
EmptyIDAtIndex	'W1000'	When adding, updating, or removing objects, an empty ID was detected (the index in array of object references is mentioned in the description text). The object reference will be ignored.
UnknownIDAtIndex	'W1001'	When updating objects an ID of an unknown object was detected. (the index in array of object references is mentioned in the description text). The given object data will be ignored.
ExistingIDAtIndex	'W1002'	When adding objects, an ID was detected that already exists (the key is mentioned in the description text). The object reference will be ignored.
DuplicateIDAtIndex	'W1003'	When adding or updating objects, an ID was detected that is duplicate within the given array (the key is mentioned in the description text). When adding objects, the second object reference will be ignored. When updating objects, the first object reference will be ignored.
EmptyArrayOnAddUpdateRemove	'W1004'	The application provided an empty array on adding, updating, or removing objects.
CyclicByParentID	'W1010'	When adding or updating activity, resource, or entity objects, one object has a given ParentID that leads to a cycle (the key is mentioned in the description text). Additionally, an exception will then be thrown after processing the whole given array of

© 2026 Boyum IT Solutions GmbH

Key	Value	Explanation
		objects, because the VSW cannot work with such a cycle in object relations.
RelatedIDUnknownAtIndex	'W1011'	(Unused/deprecated, see RelatedIDUnknown aka 'W2000'.)
UnknownID	'W1021'	When using the method ScrollToObject, then the given object ID is unknown.
ObjectNotVisibleInView	'W1022'	When using the method ScrollToObject, then the object referenced by the given object ID has no presentation in the current view.
UnknownOptionName	'W1100'	The option name given when setting or getting options is unknown to the VSW.
RequiredOptionUnset	'W1101'	A required option (namely "start" or "end") is unset by the application. The internally calculated value is mentioned in the description text.
MissingCallToRenderMethod	'W1103'	After changes to data by using a sequence of calls to the appropriate Add, Remove and Update methods, the render method should be called to avoid flickering. If the render method is not called in such a situation, this warning is triggered.
ParameterValueIncorrect	'W1104'	A parameter value in a method call is incorrect.
OptionValueIncorrect	'W1105'	A value is incorrect for setting an option.
RelatedIDUnknown	'W2000'	When changing data or composing the chart content for rendering, the widget has detected that an ID for a related object is unknown. E.g. this can be a skill ID for a resource or an allocation object.
SymbolURLInvalid	'W2001'	When using a Symbol object the browser threw a DOM exception because the URL is invalid.
HTMLCanvasFailedOnSaveAsPDF	'W3000'	When saving a PDF and at least one of the options TopHTML or BottomHTML is used, then it was not possible to create a canvas from the HTML content, because of an image URL that references a file on the disk, which is not allowed by the browser security settings. Please replace the URL by another one. The PDF will be created without a top or bottom text.

6.47 WorldViewPosition

```
Left: 1
Right: 2
Top: 3
Bottom: 4
```

APPENDIX

Past Release Notes

Version 9.0

© 2026 Boyum IT Solutions GmbH

Version 8.4
Version 8.3
Version 8.2
Version 8.1
Version 8.0
Version 7.1
Version 7.0
Version 6.4
Version 6.3
Version 6.2
Version 6.1
Version 6.0
Version 5.3
Version 5.2
Version 5.0

© 2026 Boyum IT Solutions GmbH