



Scripting Documentation of the graphomate trees 2.2 for SAP Lumira Designer 2.x

Version 2.2 - as of June 2019

<https://www.graphomate.com>

1 Scripting Documentation

Scripting Documentation for graphomate trees for SAP Lumira Designer

Version 2.2 as of June 2019

1.1 Introduction

graphomate trees support the script language of Designer. Thus, it is possible to use the extension interactively during runtime. For example, new data can be set, titles edited and scaling can be changed. A list of the current functions and examples of use is provided by this document. We are constantly trying to expand the range of functions. If certain functions are missing, please let us know by using our contact form.

1.2 Events

The events listed below are available. In the property sheet, user-defined scripts can be deposited.

Event Name	Description
On Element Clicked	Dispatched when the user clicks an element of the component in the drawing area at run time. In order to enable a drilldown, the user input must be responded to at this point. At the time of this event, the return values of the following functions are updated: getSelectedMemberKey()getSelectedMemberText()getClickedElementIndex()getClickedSeriesIndex()
On New Data Arrival	Dispatched when the tree receives updated data. This happens, for example, after the dimensions of a DataSource have changed or if filters have been changed. Adaptation of the DataTypes would be possible at this point, for example.

1.3 Functions

The following functions are available. These can be accessed within Designer's script editor.

Getter Name	Beispiel	Rückgabotyp	Beispiel Rückgabewert	Beschreibung
getColorScheme()	GRAPHOMATETREE_1.getColorScheme();	String	Spectral,9	Returns the ColorBrewer color scheme currently selected for the tree. The return value contains the schema and - separated by a comma - the number of classes.
getColorCalculationMethod()	GRAPHOMATETREE_1.getColorCalculationMethod();	String	hierarchy	Returns which method is used to color the rectangles of the tree.
getDepth()	GRAPHOMATETREE_1.getDepth();	int	3	Returns how many levels the hierarchy depth constraint is set to.
getExtendedNumberFormat()	GRAPHOMATETREE_1.getExtendedNumberFormat();	String	- . , 1 1	Returns the number format for Extended mode. The format string can be read in the manual.
getExtendedNumberFormatPercentage()	GRAPHOMATETREE_1.getExtendedNumberFormatPercentage();	String	- . , 1 1 %25	Returns the number format for percentage values in Extended mode. The format string can be read in the manual.
getLabelFormatMode()	GRAPHOMATETREE_1.getLabelFormatMode();	String	extended	Returns the mode of label formatting.
getLimitDepth()	GRAPHOMATETREE_1.getLimitDepth();	boolean	true	Returns whether the limitation of the hierarchy depth represented in the tree is possible.
getLocale()	GRAPHOMATETABLE_1.getLocale();	String	fr	Returns the setting currently selected for locale.
getNegativeDeviationIsGood()	GRAPHOMATETREE_1.getNegativeDeviationIsGood();	boolean	true	Returns whether the negative deviations are interpreted as positive or negative.
getNumberFormat()	GRAPHOMATETREE_1.getNumberFormat();	String	0,0.0	Returns the number format for Basic mode. The format string corresponds to the format of numeral.js, see manual.
getNumberFormatPercentage()	GRAPHOMATETREE_1.getNumberFormatPercentage();	String	0 %	Returns the number format for the percentage values in Basic mode. The format string corresponds to the format of numeral.js, see manual.
getRestPadding()	GRAPHOMATETREE_1.getRestPadding();	int	3	Returns the width of the left, right, and bottom of the frame drawn in embedded mode.
getSelectedMember(Dimension dimensionKey)	GRAPHOMATETREE_1.getSelectedMember("0D_NWI_RCOD");	Member		After the user has clicked an element / category of the tree, this function returns the member of the specified Dimension dimensionKey. The member provides various properties such as text, internalKey and externalKey.
getSelectedMemberKey(String dimensionKey)	GRAPHOMATETREE_1.getSelectedMemberKey("0D_NWI_RCOD");	String	10274	After the user has clicked an element / category of the tree, this function returns the key (key) of the member of the specified Dimension dimensionKey. The keys of the dimensions are shown in the Initial View.
getSelectedMemberText(String dimensionKey)	GRAPHOMATETREE_1.getSelectedMemberText("0D_NWI_RCOD");	String	North West	After the user has clicked an element / category of the tree, this function returns the text of the member of the specified dimension dimensionKey. The return is similar to the getSelectedMemberKey () function.
getTooltipVisible()	GRAPHOMATETREE_1.getTooltipVisible();	boolean	true	Returns whether tooltips are displayed for the tree during runtime.

getUseFormattedData()	GRAPHOMATETREE_1.getUsedFormattedData();	boolean	true	Returns whether the pre-formatted data from the query is used to display in the tree.
getTreeType()	GRAPHOMATETREE_1.getTreeType();	String	vertical	Returns the selected type of visualization. The type is one of the following values: circle, horizontal, vertical
getFontSize()	GRAPHOMATETREE_1.getFontSize();	float	12	Returns the font size.
getTextColor()	GRAPHOMATETREE_1.getTextColor();	String	#000000	Returns the font color.
getNodeFillColor()	GRAPHOMATETREE_1.getNodeFillColor();	String	#c0ffee	Returns the fill color of the nodes of the tree.
getNodeStrokeColor()	GRAPHOMATETREE_1.getNodeStrokeColor();	String	#c0ffee	Returns the color in which the border of the nodes of the tree is drawn.
getNodeStrokeThickness()	GRAPHOMATETREE_1.getNodeStrokeThickness();	float	0.7	Returns the border thickness.
getLinkStrokeColor()	GRAPHOMATETREE_1.getLinkStrokeColor();	String	#c0ffee	Returns the color of the links between the nodes.
getLinkStrokeThickness()	GRAPHOMATETREE_1.getLinkStrokeThickness();	float	0.9	Returns the thickness of the links.
getShowValue()	GRAPHOMATETREE_1.getShowValue();	boolean	true	Returns whether the value of a value is displayed next to the caption.
getFontFamily()	GRAPHOMATETREE_1.getFontFamily();	String	Comic Sans	Returns the selected font.
getLinkLayout()	GRAPHOMATETREE_1.getLinkLayout();	String	cross	Returns the layout of the links. One of the following values is possible: swung, straight, elbow, cross
getRootNodeDescription()	GRAPHOMATETREE_1.getRootNodeDescription();	String	Summe	Returns the label of the root node if a caption has been created manually.
getInfinityColor()	GRAPHOMATETREE_1.getInfinityColor();	String	#c0ffee	Returns the color assigned to nodes whose value is NaN or Infinity (for example, by division by 0).
getNullColor()	GRAPHOMATETREE_1.getNullColor();	String	#0ff1ce	Returns the color assigned to nodes whose value is Null (for example, because there is no value in the DataSource).
getAscendingSortOrder()	GRAPHOMATETREE_1.getAscendingSortOrder();	boolean	false	Returns the selected type of node sorting.
getAreaScale()	GRAPHOMATETREE_1.getAreaScale();	String	log	Returns the selected method for scaling the areas. Possible values are: linear, log
getMinNodeRadius()	GRAPHOMATETREE_1.getMinNodeRadius();	float	23	Returns the minimum radius of a node.
getMaxNodeRadius()	GRAPHOMATETREE_1.getMaxNodeRadius();	float	42	Returns the maximum radius of a node.
getDescriptionDelimiter()	GRAPHOMATETREE_1.getDescriptionDelimiter();	String		Returns the separator displayed between label and value on a node.

Setter Name	Beispiel	Beschreibung
setColCalculationMethod(String val)	GRAPHOMATETREE_1.setColorCalculationMethod("hierarchy");	Sets the method after which the rectangles of the tree are colored. Allowed parameter values are: hierarchy, arealdensity, colorIdentity, absoluteDeviation, percentDeviation.
setColorSchema(String val)	GRAPHOMATETREE_1.setColorScheme("Set1, 4");	Allows you to set a new ColorBrewer color scheme and the number of classes. Both entries must be separated by a comma and passed as a string.
setDepth(int level)	GRAPHOMATETREE_1.setDepth(2);	Restricts the hierarchy depth shown in the tree to the value level.
setExtendedNumberFormat(String val)	GRAPHOMATETREE_1.setExtendedNumberFormat("- . , 1 1 ");	Sets the number format for the extended mode. The format string can be read in the manual.
setExtendedNumberFormatPercentage(String val)	GRAPHOMATETREE_1.setExtendedNumberFormatPercentage("- . , 1 1 1 25%");	Sets the number format for percentage values in Extended mode. The format string can be read in the manual.
setLabelFormatMode(String val)	GRAPHOMATETREE_1.setLabelFormatMode("extended");	Sets the label formatting mode. Possible parameter values are "basic" and "extended".
setLimitDepth(boolean val)	GRAPHOMATETREE_1.setLimitDepth(false);	Sets whether the limitation of the hierarchy depth is activated.
setLocale(String val)	GRAPHOMATETREE_1.setLocale("de");	Sets a new locale value. Allowed values for the locale are: de, en, fr and auto. If auto is selected, the locale is taken from the query for the formatting of the texts.
setNegativeDeviationIsGood(boolean val)	GRAPHOMATETREE_1.setNegativeDeviationIsGood(true);	Sets whether negative deviations are interpreted as positive or negative.
setNumberFormat(String val)	GRAPHOMATETREE_1.setNumberFormat("0,0.0");	Sets the number format for Basic mode. The format string corresponds to the format of numeral.js, see manual.
setNumberFormatPercentage(String val)	GRAPHOMATETREE_1.setNumberFormatPercentage("0 %");	Sets the number format for percentage values in Basic mode. The format string corresponds to the format of numeral.js, see manual.
setTooltipVisible(boolean val)	GRAPHOMATETREE_1.setTooltipVisible(true);	Sets whether tooltips are displayed in the tree.
setUseFormattedData(boolean val)	GRAPHOMATETREE_1.setUseFormattedData(true);	Sets whether the pre-formatted data from Designer is used for the table.
setTreeType(String val)	GRAPHOMATETREE_1.setTreeType("circle");	Sets the selected type of visualization. The type must be one of the following values: circle, horizontal, vertical
setFontSize(float val)	GRAPHOMATETREE_1.setFontSize(14);	Sets the font size.
setTextColor(String val)	GRAPHOMATETREE_1.setTextColor("#c0ffee");	Sets the font color:
setNodeFillColor(String val)	GRAPHOMATETREE_1.setNodeFillColor("#c0ffee");	Sets the fill color of the nodes of the tree.
setNodeStrokeColor(String val)	GRAPHOMATETREE_1.setNodeStrokeColor("#c0ffee");	Sets the color in which the outline of the nodes of the tree is drawn.
setNodeStrokeThickness(float val)	GRAPHOMATETREE_1.setNodeStrokeThickness(3);	Sets the border thickness.
setLinkStrokeColor(String val)	GRAPHOMATETREE_1.setLinkStrokeColor("#0ff1ce");	Sets the color of the links between the nodes.
setLinkStrokeThickness(float val)	GRAPHOMATETREE_1.setLinkStrokeThickness(1.2);	Sets the thickness of the links.
setShowValue(boolean val)	GRAPHOMATETREE_1.setShowValue(false);	Sets whether the value of a node is displayed next to the label.
setFontFamily(String val)	GRAPHOMATETREE_1.setFontFamily("Wingdings");	Sets the font.
setLinkLayout(String val)	GRAPHOMATETREE_1.setLinkLayout("elbow");	Sets the layout of the links. One of the following values is possible: swung, straight, elbow, cross
setRootNodeDescription(String val)	GRAPHOMATETREE_1.setRootNodeDescription("Welt");	Sets the label of the root node.

setInfinityColor(String val)	GRAPHOMATETREE_1.setInfinityColor("#0ff1ce");	Sets the color given to nodes whose value is NaN or Infinity (for example, by division by 0).
setNullColor(String val)	GRAPHOMATETREE_1.setNullColor("#0ff1ce");	Sets the color assigned to nodes whose value is null (for example, because there is no value in the DataSource).
setAscendingSortOrder(boolean val)	GRAPHOMATETREE_1.setAscendingSortOrder(true);	Sets whether nodes are sorted or not.
setAreaScale(String val)	GRAPHOMATETREE_1.setAreaScale("linear");	Sets which method is used to scale the areas. Possible values include: linear, log
setMinNodeRadius(float val)	GRAPHOMATETREE_1.setMinNodeRadius("3");	Sets the minimum radius of a node.
setMaxNodeRadius(float val)	GRAPHOMATETREE_1.setMaxNodeRadius("30");	Sets the maximum radius of a node.
setDescriptionDelimiter(String val)	GRAPHOMATETREE_1.setDescriptionDelimiter("~");	Sets which separator is used to represent label and value on a node.
