



# Scripting Documentation of the graphomate charts 2.7 for SAP Lumira Designer 2.x

Version 2.7 - as of October 2018

<https://www.graphomate.com>

# 1 Scripting Documentation

## Scripting Documentation for graphomate charts for SAP Lumira Designer

Rev. 2.7 as of September 2018

### 1.1 Introduction

The graphomate charts extension supports the scripting language of Designer. Therefore, it is possible to determine properties of the extension at run-time. New data can be set, titles can be adapted and scales be changed. Furthermore, it is possible to use drilldown functionalities of the extension and thus create interactive dashboards.

This document provides a list of available functions, events and examples. We are steadily working on expanding the range of functions. If you miss certain functions, please give us a note. Therefore you can use our online form under <http://www.graphomate.com/en/feedback>.

### 1.2 Events

The following events are available. User-made scripts can be deposited on the Standard Properties Sheet under the point Events.

Event name	description
On Element Clicked	Is triggered if the user clicks on an element on the chart at run-time. In order to allow a drill-down, it is necessary to react on the user input at this point. The return values of the following functions are updated at the moment of the event: <code>getSelectedMemberKey()</code> , <code>getSelectedMemberText()</code> , <code>getClickedElementIndex()</code> , <code>getClickedSeriesIndex()</code>
On new data arrival	Is triggered when the chart receives updated data. This is happening, for example, when the dimension of a Data Source or if a filter has been changed. This would be the appropriate point to change Data Types or the Waterfall Calculation Path.

### 1.3 Functions

The following functions are available in the graphomate charts extension. They can be addressed in the scripting editor of Designer.

#### Guideline to umlauts, special characters and blank spaces

In general umlauts, special characters and blank spaces Umlauts are percent encoded for technical reasons in the Additional Properties Sheet (see: <http://en.wikipedia.org/wiki/Percent-encoding>).

Example:

Nestlé - Development of sales in % compared to the previous year

becomes:

Nestl%C3%A9%20 %20Development%20of%20sales%20in%20%25%20compared%20to%20the%20previous%20year

This makes it necessary to percent encode string values even while transferring them from the scripting language or Standard Properties Sheet. (On the Internet there are many free tools for en-/decoding, for example: <http://www.url-encode-decode.com/>.)

Getter Name	example	return type	example return value	description
<code>getAxisLabels()</code>	<code>GRAPHOMATECHART_1.getAxisLabels();</code>	String	Budget, Actual	Returns axis labels as a comma-separated string value.
<code>getAxisThickness()</code>	<code>GRAPHOMATECHART_1.getAxisThickness();</code>	Integer	3.0	Returns the axis thickness as an integer value.
<code>getBarWidth()</code>	<code>GRAPHOMATECHART_1.getBarWidth();</code>	Float	5.0	Returns the width of the bars as a float value.
<code>getBaseChartVisible()</code>	<code>GRAPHOMATECHART_1.getBaseChartVisible();</code>	Boolean	true	Returns whether the base chart is visible.
<code>getCalculationPath()</code>	<code>GRAPHOMATECHART_1.getCalculationPath();</code>	String	<code>+,+,+,-,-,=, ,S+,+,=,-,-</code>	Returns the waterfall calculation path as a comma-separated string value.
<code>getCalculateDeviationsFromNull()</code>	<code>GRAPHOMATECHART_1.getCalculateDeviationsFromNull();</code>	Boolean	true	Returns whether NULL values are used to calculate deviations.
<code>getCategoryAxisVisible()</code>	<code>GRAPHOMATECHART_1.getCategoryAxisVisible();</code>	Boolean	true	Returns whether the category axis is visible.
<code>getCategoryCount()</code>	<code>GRAPHOMATECHART_1.getCategoryCount();</code>	Integer	12	Returns whether the number of to be displayed categories is limited. Returns an integer value. A value of 0 means that there is no limitation.
<code>getCategoryLabelRotation()</code>	<code>GRAPHOMATECHART_1.getCategoryLabelRotation();</code>	Integer	45	Returns the rotation angle of the category labels.
<code>getCategoryWidth()</code>	<code>GRAPHOMATECHART_1.getCategoryWidth();</code>	Integer	20	Returns the width of the categories as an integer value if the Category Width is fixed.
<code>getChartType()</code>	<code>GRAPHOMATECHART_1.getChartType();</code>	String	line	Returns the chart type as a string-value.
<code>getClickedElementIndex()</code>	<code>GRAPHOMATECHART_1.getClickedElementIndex();</code>	Integer	3	Returns the index of the last clicked element/category.

getClickedPositionValue()	GRAPHOMATECHART_1.getClickedPositionValue();	Float	234234.432	Returns the relative value of the clicked position in the chart. This can be utilized to create a zoom function, when the clicked value is passed back to the chart as the manual scale minimum.
getClickedSeriesIndex()	GRAPHOMATECHART_1.getClickedSeriesIndex();	Integer	1	Returns the index of the last clicked Data Series.
getComparisonGroup()	GRAPHOMATECHART_1.getComparisionGroup();	String	A	Returns the Comparison Group as a string value.
getContextMenuCustomEventName(Integer index)	GRAPHOMATECHART_1.getContextMenuCustomEventName(1);	String	Custom Entry 1	Returns the name of a specified custom event (Index-1 based) in the Context Menu.
getContextMenuCustomEventVisibility(Integer index )	GRAPHOMATECHART_1.getContextMenuCustomEventVisibility(1);	Boolean	true	Returns the visibility of a specified custom event (Index-1 based) in the Context Menu.
getContextMenuOptionVisibility(Integer index)	GRAPHOMATECHART_1.getContextMenuOptionVisibility(1);	Boolean	true	Returns the visibility of a specified option in the Context Menu. Use the following indices to access the options: 1 for Sorting, 2 for Filters, 3 for Context Sensitive Filters, 4 for Filter and Hide, 5 for Clear Filters, 6 for Drilldown, 7 for Drillup, 8 for Zero Display, 9 for Member Display, 10 for Position of Totals, 11 for Total Display.
getDatatypes(Integer seriesIndex)	GRAPHOMATECHART_1.getDatatypes(1);	String	PY,PY,PY,PY,AC,AC,AC,AC,PL,PL,PL,PL	This function returns the Data Types of the selected Data Series seriesIndex as a comma-separated string.
getDeviationBadColor()	GRAPHOMATECHART_1.getDeviationBadColor();	String	#ff3300	This function returns the color used for "bad" deviations.
getDeviationGoodColor()	GRAPHOMATECHART_1.getDeviationGoodColor();	String	#2e2e2e	This function returns the color used for "good" deviations.
getDeviationVisibility()	GRAPHOMATECHART_1.getDeviationVisibility();	String	0,1,4	Returns a list of indexes of visible deviation charts as a comma-separated string.
getElementsOffset()	GRAPHOMATECHART_1.getElementsOffset();	Float	2.0	This function returns the offset between the elements of a category as a float value.
getEndingLineOverlap()	GRAPHOMATECHART_1.getEndingLineOverlap();	Integer	3	Returns how far the Ending Lines protrude over the bars.
getEndingLineThickness()	GRAPHOMATECHART_1.getEndingLineThickness();	Integer	2	Returns the thickness of the Ending Lines.
getEndingLineVisible()	GRAPHOMATECHART_1.getEndingLineVisible();	Boolean	false	Returns whether to display Ending Lines above the bars.
getExtendedNumberFormat()	GRAPHOMATECHART_1.getExtendedNumberFormat();	String	- . , 1 1 %25	Returns the number format for the Extended Mode. The format string can be found in the manual.
getExtendedNumberFormatPercentage()	GRAPHOMATECHART_1.getExtendedNumberFormatPercentage();	String	- . , 1 1 %25	Returns the number format for the percentage values in the Extended Mode. The format string can be found in the manual.
getFixedCategoryWidth()	GRAPHOMATECHART_1.getFixedCategoryWidth();	Boolean	true	This function returns a boolean value whether the Category Width is fixed.
getHighlightMode()	GRAPHOMATECHART_1.getHighlightMode();	String	single	This function returns the current setting for the highlight mode.
getLabelFormatMode()	GRAPHOMATECHART_1.getLabelFormatMode();	String	extended	Returns which mode is used for the label format.
getLabelPositioningMode()	GRAPHOMATECHART_1.getLabelPositioningMode();	String	auto	Returns the current label positioning mode.
getLockData()	GRAPHOMATECHART_1.getLockData();	Boolean	true	Returns whether this component ignores data updates.
getManualScalePercentMax()	GRAPHOMATECHART_1.getManualScalePercentMax();	Float	80.5	Returns the manual percentage scaling maximum as a float value.
getManualScalePercentMin()	GRAPHOMATECHART_1.getManualScalePercentMin();	Float	-80.5	Returns the manual percentage scaling maximum as a float value.
getMultiHighlights()	GRAPHOMATECHART_1.getMultiHighlights();	String	[{"startElement":1,"endElement":2,"startSeries":1,"endSeries":1}, {"startElement":3,"endElement":4,"startSeries":1,"endSeries":1}, {"startElement":5,"endElement":6,"startSeries":1,"endSeries":1}]	Returns a list of the current multi highlights (highlighted values above the chart). Returns a JSON-string, which contains an array consisting of an object for each multi highlight. The following elements are part of an object (the order of these elements must be maintained): {"startElement":1,"endElement":2,"startSeries":1,"endSeries":1}.
getNegativeDeviationIsGood()	GRAPHOMATECHART_1.getNegativeDeviationIsGood();	Boolean	true	Returns whether the negative deviations are interpreted as positive or negative.
getNegativeOutlierThreshold()	GRAPHOMATECHART_1.getNegativeOutlierThreshold();	Float	-80.0	Returns the negative outlier threshold as a float value.
getNumberFormat()	GRAPHOMATECHART_1.getNumberFormat();	String	0,0.0	Returns the number format for the Basic Mode. The format string corresponds to the format numeral.js (see <a href="http://www.numeraljs.com">http://www.numeraljs.com</a> ).
getNumberFormatPercentage()	GRAPHOMATECHART_1.getNumberFormatPercentage();	String	0%	Returns the number format for the percentage values in the Basic Mode. The format string corresponds to the format numeral.js (see <a href="http://www.numeraljs.com">http://www.numeraljs.com</a> ).
getPositiveOutlierThreshold()	GRAPHOMATECHART_1.getPositiveOutlierThreshold();	Float	80.0	Returns the positive outlier threshold as a float value.
getProperty(String)	GRAPHOMATECHART_1.getProperty("labelFormatMode");	String	any value	Returns the value of the given property as a string representation.
getRulesets()	GRAPHOMATECHART_1.getRulesets();	String	Ruleset1,Ruleset2	Returns the list of rulesets used by the chart.
getSelectedMember(Dimension dimensionKey)	GRAPHOMATECHART_1.getSelectedMember();	Member	0D_NWI_RCOD	After the user clicked on an element/a category in the chart, this function returns the member of the specified dimension dimensionKey. The member provides different properties such as text, internalKey and externalKey.
getSelectedMemberKey(String dimensionKey)	GRAPHOMATECHART_1.getSelectedMemberKey("0D_NWI_RCOD");	String	10273	After the user clicked on an element/a category in the chart, this function returns the Key of the member of the selected dimension dimensionKey.
getSelectedMemberText(String dimensionKey)	GRAPHOMATECHART_1.getSelectedMemberText("0D_NWI_RCOD");	String	North West	After the user clicked on an element/a category in the chart, this function returns the text of the member of the selected dimension dimensionKey. The return value is similar to the function getSelectedMemberKey().
getSeparatorLength()	GRAPHOMATECHART_1.getSeparatorLength();	Integer	5	Returns the length of the Seperators in pixels.
getSeparatorMode()	GRAPHOMATECHART_1.getSeparatorMode();	String	datatypes	Returns the Separator Mode.

getSeparatorPositions()	GRAPHOMATECHART_1.getSeparatorPositions();	String	2,4,6	Returns a list of (manual) set positions of Separators as comma-separated string value.
getSeparatorSourceRowIndex()	GRAPHOMATECHART_1.getSeparatorSourceRowIndex();	Integer	3	Returns the index of the row/series, from which the Separators can be derived. This function is dependent on the Separator Mode (setSeparatorMode ()).
getSeparatorThickness()	GRAPHOMATECHART_1.getSeparatorThickness();	Integer	2	Returns the thickness of the Serators in pixels.
getSeries(Integer seriesIndex)	GRAPHOMATECHART_1.getSeries(1);	String	{"Keyfigures":"0D_NWI_NSAL"}	Returns the data description of the selected Data Series seriesIndex as JSON string.
getSeriesLabels()	GRAPHOMATECHART_1.getSeriesLabels();	String	Budget,Actual,Target	Returns the series label of the chart as comma-separated string.
getShowAxisLabelsLeft()	GRAPHOMATECHART_1.getShowAxisLabelsLeft();	Boolean	true	Returns whether the Axis Labels are displayed on the left.
getShowAxisLabelsRight()	GRAPHOMATECHART_1.getShowAxisLabelsRight();	Boolean	true	Returns whether the Axis Labels are displayed on the right.
getShowCategoryLabels()	GRAPHOMATECHART_1.getShowCategoryLabels();	Boolean	false	Returns whether the Category Labels are displayed.
getShowDatatypesInAxis()	GRAPHOMATECHART_1.getShowDatatypesInAxis();	Boolean	true	Returns whether the Data Types are displayed in the axis or in the elements.
getShowTitle()	GRAPHOMATECHART_1.getShowTitle();	Boolean	true	Returns whether the title is displayed.
getShowTooltips()	GRAPHOMATECHART_1.getShowTooltips();	Boolean	false	Returns whether the tooltips are displayed in runtime mode.
getSpaceFactorBaseChart()	GRAPHOMATECHART_1.getSpaceFactorBaseChart();	Float	0.8	Returns the proportional Space Factor of the Base Chart (relative to the other Space Factors).
getSpaceFactorDeviationChart()	GRAPHOMATECHART_1.getSpaceFactorDeviationChart();	Float	0.2	Returns the proportional Space Factor of the deviation chart (relative to the other Space Factors).
getStackedBarLabelPosition()	GRAPHOMATECHART_1.getStackedBarLabelPosition();	String	middle	Returns the position of the labels in the stacked bar chart.
getStackedBarNegativeValues()	GRAPHOMATECHART_1.getStackedBarNegativeValues();	Boolean	true	This function returns whether negative values are allowed in stacked bar charts.
getStackedBarSumsVisible()	GRAPHOMATECHART_1.getStackedBarSumsVisible();	Boolean	true	This function returns whether sums are displayed in the stacked bars.
getStructureMode()	GRAPHOMATECHART_1.setStructureMode();	Boolean	true	This function returns whether a chart is currently in structure mode (identical to showStructureMode()).
getSuppressRepeatingCategoryLabels()	GRAPHOMATECHART_1.getSuppressRepeatingCategoryLabels();	String	true,true,false,false,true	Returns the configuration of the property Suppress Repeating Category Labels. Returns a comma separated string value.
getSuppressZeroLabels()	GRAPHOMATECHART_1.getSuppressZeroLabels();	Boolean	true	Returns whether Zero Labels are displayed.
getSynchronizeAxesPosition()	GRAPHOMATECHART_1.getSynchronizeAxesPosition();	Boolean	true	This function returns a boolean value whether the axes positions are synchronized with the Comparison Group.
getSynchronizePadding()	GRAPHOMATECHART_1.getSynchronizePadding();	Boolean	false	This function returns a boolean value whether the padding of the chart is synchronized with the Comparison Group.
getSynchronizeScaling()	GRAPHOMATECHART_1.getSynchronizeScaling();	Boolean	true	The function returns a boolean value whether the scaling of the chart is synchronized with the Comparison Group.
getTitle()	GRAPHOMATECHART_1.getTitle();	String	Sales in Mio. EUR  2013 ACT and BUD  North West	Returns the title of the chart as a string value. A line break is represented with a pipe ( ).
getUseOutlierThreshold()	GRAPHOMATECHART_1.getUseOutlierThreshold();	Boolean	true	Returns whether the Outlier Threshold is used.
getUserHighlightColor()	GRAPHOMATECHART_1.getUserHighlightColor();	String	#ff0000,#00ff00,#0000ff	Returns a list of deposited colors for the User Highlights as comma separated string-value. The colors are switched at each click (during runtime) until the desired color is set.
getUserHighlightList()	GRAPHOMATECHART_1.getUserHighlightList();	String	[[1,3],[2,3][1,5]]	Returns a list of the current user highlights (highlighted values at runtime). Returns a JSON-string, which passes the positions of each highlight with two numbers; the first number indicates the data series, while the second number represents the position of the value in the data series.
getValueAxisVisible()	GRAPHOMATECHART_1.getValueAxisVisible();	Boolean	true	Returns whether the value axis is visible or not.
getVisibleElementLabels()	GRAPHOMATECHART_1.getVisibleElementLabels();	String	1,3,5	Retrieves the visible element labels as a comma separated list of indices.
getVisibleSeries()	GRAPHOMATECHART_1.getVisibleSeries();	String	0,1,4	Returns a list of indexes of visible series as comma-separated string.

refresh(string value)  
Redraws the chart. This is a workaround in case the chart has not been refreshed after data has changed at run-time by the impact of the scripting language on the Data Source. This could happen, for example, when filters have been changed, dimensions have been added or removed.

If the chart should unexpectedly not refresh, this command should help:

```
GRAPHOMATECHART_1.refresh(GRAPHOMATECHART_1.refreshValue());
```

It is absolutely necessary to pass the value of refreshValue() as parameter val!

refreshValue()  
Help function for refresh(). It has no further use.

scalingHelperMode(string value)

Sets the Scaling Helper. Valid values are none, line, area.

Example: activate the Scaling Helper

```
GRAPHOMATECHART_1.scalingHelperMode("line");
```

Setter Name	Beispiel	Beschreibung
addRuleset(String name)	GRAPHOMATECHART_1.addRuleset("Ruleset2");	Adds a ruleset to the list of rulesets. Does not create duplicates.
removeRuleset(String name)	GRAPHOMATECHART_1.removeRuleset("Ruleset2");	Removes a ruleset from the list of rulesets used by the chart. Does nothing, if the ruleset is not present.
scalingHelperMode(string value)	activate the Scaling Helper	Sets the Scaling Helper. Valid values are none, line, area.
setAxisLabels(String)	GRAPHOMATECHART_1.setAxisLabels("Net Sales,Returns");	Sets the axes labels of the chart. Expects value as comma-separated string. The first value is interpreted as left axis label, the second as right axis label.
setAxisThickness(Integer)	GRAPHOMATECHART_1.setAxisThickness(3);	Sets the axes thickness.
setBarWidth(Float)	GRAPHOMATECHART_1.setBarWidth(3);	Sets the width of the bars as a float value. Values less than or equal to 1 are interpreted as percentages.
setBaseChartVisible(Boolean)	GRAPHOMATECHART_1.setBaseChartVisible(true);	Sets whether the base chart is visible.
setCalculationPath(String)	GRAPHOMATECHART_1.setCalculationPath("+,+,+,-,-, ,s+,+,-,-");	Sets the Waterfall Calculation Path. Expects value as comma-separated string.
setCalculateDeviationsFromNull (Boolean)	GRAPHOMATECHART_1.setCalculateDeviationsFromNull(true);	This function sets whether NULL values are used to calculate deviations.
setCategoryAxisVisible(Boolean)	GRAPHOMATECHART_1.setCategoryAxisVisible(true);	Sets whether the base chart is visible.
setCategoryCount(Integer)	GRAPHOMATECHART_1.setCategoryCount(12);	Sets a limit on the number of displayed category. A value of 0 removes the restriction.
setCategoryLabelRotation(int rotation)	GRAPHOMATECHART_1.setCategoryLabelRotation(45);	Rotates the category labels according to the passed angle. Rotation takes place counter-clockwise.
setCategoryWidth(Integer)	GRAPHOMATECHART_1.setCategoryWidth(3);	This function sets the width of the categories. But it is only interpreted when fixedCategoryWidth is set.
setChartType(String)	GRAPHOMATECHART_1.setChartType("line");	Sets the chart type. The following string values are allowed: bar, line, offsetbar, waterfall, pin, stackedbar.
setComparisonGroup(String id)	GRAPHOMATECHART_1.setComparisonGroup("A");	Sets the Comparison Group of the chart as a string. Charts with the same Comparison Group are equal scaled and aligned. More details can be set using the following parameters: <code>getSynchronizePadding()</code> , <code>setSynchronizePadding()</code> , <code>getSynchronizeScaling()</code> , <code>setSynchronizeScaling()</code> , <code>getSynchronizeAxesPosition()</code> , <code>setSynchronizeAxesPosition()</code>
setContextMenuCustomEventName(Integer index, String name)	GRAPHOMATECHART_1.setContextMenuCustomEventName(1, "Script 1");	Sets the name of a specified custom event (Index-1 based) in the Context Menu.
setContextMenuCustomEventVisibility(Integer index, Boolean visibility)	GRAPHOMATECHART_1.setContextMenuCustomEventVisibility(1, false);	Sets the visibility of a specified custom event (index-1 based) in the Context Menu.
setContextMenuOptionVisibility(Integer index, Boolean visibility)	GRAPHOMATECHART_1.setContextMenuOptionVisibility(1, false);	Sets the visibility of a specified option in the Context Menu. Use the following indices to access the options: 1 for Sorting, 2 for Filters, 3 for Context Sensitive Filters, 4 for Filter and Hide, 5 for Clear Filters, 6 for Drilldown, 7 for Drillup, 8 for Zero Display, 9 for Member Display, 10 for Position of Totals, 11 for Total Display.
setDatatypeAt(Integer series, Integer index, String datatype)	GRAPHOMATECHART_1.setDatatypeAt(1, 1, "AC");	This function sets the passed datatype for a single element of the passed data series.
setDatatypeForSeries(Integer series, Integer count, String datatype)	GRAPHOMATECHART_1.setDatatypeForSeries(1, 5, "AC");	This function sets the passed datatype for the data series with the passed index. The number of datatype entries generated is governed by the parameter count. The function call GRAPHOMATECHART_1.setDatatypeForSeries(1, 5, "AC"); is equivalent to GRAPHOMATECHART_1.setDatatypeForSeries(1, "AC,AC,AC,AC,AC");
setDatatypes (Integer seriesIndex, String)	Beispiel: Datenarten der ersten fünf Elemente der ersten Datenserie auf "PP" setzen GRAPHOMATECHART_1.setDatatypes(1, "PP,PP,PP,PP,PP");	This function sets the Data Types of the selected Data Series seriesIndex. Expects value as comma-separated string.
setDeviationBadColor(String)	GRAPHOMATECHART_1.setDeviationBadColor("#ff3300");	This function sets the color used for "bad" deviations. Expects a string value in the format #ff3300.
setDeviationGoodColor(String)	GRAPHOMATECHART_1.setDeviationGoodColor("#2e2e2e");	This function sets the color used for "good" deviations. Expects a string value in the format #2e2e2e
setDeviationLabels(String labels, Boolean visibleOnly)	GRAPHOMATECHART_1.setDeviationLabels("Abs. Dev. FC,Perc. Dev. BUD", true);	Sets the labels for every deviation series. If the second value is true only visible deviations are considered.
setDeviationVisibility (String)	GRAPHOMATECHART_1.setDeviationVisibility("0,1");	Sets a list of indexes of visible deviation charts. Expects value as comma-separated string.
setElementsOffset(Float)	GRAPHOMATECHART_1.setElementsOffset(5);	This function sets the offset between the elements of a category. Values less than or equal to 1 are interpreted as percentages.
setEndingLineOverlap(Integer)	GRAPHOMATECHART_1.setEndingLineOverlap(5)	Sets how far the Ending Lines protrude over the bars.
setEndingLineThickness(Integer)	GRAPHOMATECHART_1.setEndingLineThickness(3);	Sets the thickness of the Ending Lines.
setEndingLineVisible(Boolean)	GRAPHOMATECHART_1.setEndingLineVisible(true);	Sets whether to display Ending Lines above the bars.
setExtendedNumberFormat(String)	GRAPHOMATECHART_1.setExtendedNumberFormat("-  , 1 1 ");	Sets the number format for the Extended Mode. The format string can be found in the manual.
setExtendedNumberFormatPercentage(String)	GRAPHOMATECHART_1.setExtendedNumberFormatPercentage("-  , 1 1 25%");	Sets the number format for the percentages in the Extended Mode. The format string can be found in the manual.
setFixedCategoryWidth(Boolean)	GRAPHOMATECHART_1.setFixedCategoryWidth(true);	This functions sets whether the Category Width should be fixed. If it is not fixed, the width is determined automatically.
setHighlightEndIndex(Integer)	GRAPHOMATECHART_1.setHighlightEndIndex(4);	Sets the category index value of the end value of the Highlight. Analogous to <code>setHighlightStartIndex()</code> .
setHighlightEndSeries(Integer)	GRAPHOMATECHART_1.setHighlightEndSeries(3);	Sets the series index value of the end series of the Highlight. Analogous to <code>setHighlightStartSeries()</code> .
setHighlightMode(String)	GRAPHOMATECHART_1.setHighlightMode("single");	This function can be used to change the highlight mode. Valid values are none, single or auto.
setHighlightStartIndex(Integer)	GRAPHOMATECHART_1.setHighlightStartIndex(0);	Sets the category index value of the start value of the Highlight Analogous to <code>setHighlightEndIndex()</code> .

setHighlightStartSeries(Integer)	GRAPHOMATECHART_1.setHighlightStartSeries(1);	Sets the series index value of the start series of the Highlight. Analogous to setHighlightEndSeries().
setLabelFormatMode(String)	GRAPHOMATECHART_1.setLabelFormatMode("extended");	Sets a string value for the used label format mode. Available values are basic and extended.
setLabelPositioningMode(String)	GRAPHOMATECHART_1.setLabelPositioningMode("auto");	This function sets the label positioning mode. Valid arguments are none (no labels are displayed), fix (labels are displayed at the exact position; overlapping might occur) and auto (no overlapping, but labels may be moved or hidden).
setLockData(Boolean)	GRAPHOMATECHART_1.setLockData(true);	Sets whether the component ignores data updates.
setManualCategoryLabels(String labels)	GRAPHOMATECHART_1.setManualCategoryLabels('[[["2001","2001","2001","2001"],["Q1","Q2","Q3","Q4"]]]')	Sets the manual category labels. Expects value as two dimensional JSON array string. Examples: [[["Q1","Q2","Q3","Q4"],["2017","2017","2017","2017"]] or [[["Q1","Q2","Q3","Q4","Q1","Q2","Q3","Q4"],["North","North","North","North","South","South","South","South"],["2017","2017","2017","2017","2017","2017","2017","2017"]]]
setManualScaleMaximum(Integer max)	die manuelle Skalierung auf 1.000.000 begrenzen GRAPHOMATECHART_1.setManualScaleMaximum(1000000);	Sets the maximum of the scaling of the Base Chart to the integer value max. It is necessary to set the scale mode to Manual for the integer to become effective. (see also setScaleModeManual() and Scaling in the features description)
setManualScaleMaximumFloat(Float max)	GRAPHOMATECHART_1.setManualScaleMaximumFloat(1000000);	Sets the maximum of the scaling in the Base Chart to the float value max. It is necessary to set the scale mode to Manual. (see also setScaleModeManual() and Scaling in the features description). This function is an alternative function to setManualScaleMaximum(), in case only float values are available.
setManualScaleMinimum(Integer min)	die manuelle Skalierung bei 1.000.000 beginnen GRAPHOMATECHART_1.setManualScaleMinimum(1000000);	Sets the minimum of the scaling of the Base Chart to the integer value min. It is necessary to set the scale mode to Manual for the integer to become effective. (see also setScaleModeManual() and Scaling in the features description)
setManualScaleMinimumFloat(Float min)	GRAPHOMATECHART_1.setManualScaleMinimumFloat(1000000);	Sets the minimum of the scaling in the Base Chart to float value min. It is necessary to set the scale mode to Manual. (see also setScaleModeManual and Scaling in the features description) This function is an alternative function to setManualScaleMinimum(), in case only float values are available.
setManualScalePercentMax(Float)	GRAPHOMATECHART_1.setManualScalePercentageMax(80);	Sets the manual percentage scaling maximum as a float value.
setManualScalePercentMin(Float)	GRAPHOMATECHART_1.setManualScalePercentageMin(-80);	Sets the manual percentage scaling minimum as a float value.
setMultiHighlights(String)	GRAPHOMATECHART_1.setMultiHighlights("{{\"startElement\":1,\"endElement\":2,\"startSeries\":1,\"endSeries\":1}, {\"startElement\":3,\"endElement\":4,\"startSeries\":1,\"endSeries\":1}, {\"startElement\":5,\"endElement\":6,\"startSeries\":1,\"endSeries\":1}}");	Sets the list of multi highlights (highlighted values above the chart). Expects a JSON-string which contains an array consisting of an object for each multi highlight. The following elements are part of an object (the order of these elements must be maintained): {"startElement":1,"endElement":2,"startSeries":1,"endSeries":1}.
setNegativeDeviationIsGood(Boolean)	GRAPHOMATECHART_1.setNegativeDeviationIsGood(true);	Sets whether negative deviations are interpreted as positive or negative.
setNegativeOutlierThreshold(Float)	GRAPHOMATECHART_1.setNegativeOutlierThreshold(-80);	Sets the negative outlier threshold.
setNumberFormat(String)	GRAPHOMATECHART_1.setNumberFormat("0,0.0");	Sets the number format for the Basic Mode. The format string corresponds to the format numeral.js (see <a href="http://www.numerajs.com">http://www.numerajs.com</a> ).
setNumberFormatPercentage(String)	GRAPHOMATECHART_1.setNumberFormatPercentage("0 %");	Sets the number format for the percentages in the Basic Mode. The format string corresponds to the format numeral.js (see <a href="http://www.numerajs.com">http://www.numerajs.com</a> ).
setPositiveOutlierThreshold(Float)	GRAPHOMATECHART_1.setPositiveOutlierThreshold(80);	Sets the positive outlier threshold.
setProperty(String, String)	GRAPHOMATECHART_1.setProperty("suppressZeroLabels", "true");	Sets an arbitrary property.  The technical name of the property is required, in order to set the property. You can get the technical name in the lower left corner in the status bar of Lumira Designer / Design Studio, after selecting the according property in the property sheet. The new value of the property must be a string representation of the value, e.g. "Name", "2", "true", "#334455", "3.5" etc.
setRulesets(String newRulesetList);	GRAPHOMATECHART_1.setRulesets("Ruleset1, Ruleset2");	Sets the rulesets of the chart. Requires the server url to be set and valid.
setScaleModeAutomatic()	GRAPHOMATECHART_1.setScaleModeAutomatic();	Sets the scale mode of the Base Chart to Automatic.
setScaleModeManual()	GRAPHOMATECHART_1.setScaleModeManual();	Sets the scale mode of the Base Chart to Manual.
setScalingHelperValue(Integer)	GRAPHOMATECHART_1.setScalingHelperValue(500);	Sets the position of the Scaling Helper. Expects the position value as integer value.
setScalingHelperValueFloat(Float)	GRAPHOMATECHART_1.setScalingHelperValueFloat(500);	Sets the position of the Scaling Helper. Expects the position value as float value.
setSeparatorLength(Integer)	GRAPHOMATECHART_1.setSeparatorLength(5);	Sets the length of the Separators in pixels.
setSeparatorMode(String)	GRAPHOMATECHART_1.setSeparatorMode("datatypes");	Sets the Separator Mode. Valid values are: none, categorylabels, datatypes, manually.
setSeparatorPositions(String)	GRAPHOMATECHART_1.setSeparatorPositions("100,200,300");	Sets a list of separator positions of the chart. Expects value as comma-separated string.
setSeparatorSourceRowIndex(Integer)	GRAPHOMATECHART_1.setSeparatorSourceRowIndex(3);	Sets the index of the row/series, from which the Separators can be derived. This function is dependent on the Separator Mode (setSeperatorMode ()).
setSeparatorThickness(Integer)	GRAPHOMATECHART_1.setSeparatorThickness(3);	Sets the thickness of the Separators in pixels.
setSeries(Integer seriesIndex, String)	die Datenbeschreibung der ersten Datenserie auf den übergebenen Wert setzen GRAPHOMATECHART_1.setSeries(1, '{"Keyfigures":"0D_NWI_NSAL"}');	Set the data description of the selected Data Series seriesIndex. Expects value as JSON string. This string either describes a row or a column in the Initial View. The string can be read out after data for a Data Series has been selected on the Property Sheet with the graphomate charts extension.
setSeriesLabels(String)	GRAPHOMATECHART_1.setSeriesLabels("Series1, Series2");	Sets the series labels of the charts. Expects value as comma-separated string.
setShowAxisLabelsLeft(Boolean)	GRAPHOMATECHART_1.setShowAxisLabelsLeft(true);	Sets whether the Axis Labels are displayed on the left..
setShowAxisLabelsRight(Boolean)	GRAPHOMATECHART_1.setShowAxisLabelsRight(true);	Sets whether the Axis Labels are displayed on the right.
setShowCategoryLabels(Boolean)	GRAPHOMATECHART_1.setShowCategoryLabels(true);	Sets whether the Category Labels are displayed.
setShowDatatypesInAxis(Boolean)	GRAPHOMATECHART_1.setShowDatatypesInAxis(true);	Sets whether the Data Types are displayed in the Axis.
setShowTitle(Boolean)	GRAPHOMATECHART_1.setShowTitle(true);	Sets whether the title is displayed.
setShowTooltips(Boolean)	GRAPHOMATECHART_1.setShowTooltips(true);	Sets whether the tooltips in runtime mode are displayed.
setSpaceFactorBaseChart(Float)	GRAPHOMATECHART_1.setSpaceFactorBaseChart(0.8);	Sets the proportional Space Factor of the Base Chart (relative to the other Space Factors).

setSpaceFactorDeviationChart(Float)	GRAPHOMATECHART_1.setSpaceFactorDeviationChart(0.2);	Sets the proportional Space Factor of the Base Chart (relative to the other Space Factors).
setStackedBarLabelPosition(String)	GRAPHOMATECHART_1.setStackedBarLabelPosition("middle");	Sets the position of the labels in the stacked bar chart. Valid values are middle and right.
setStackedBarNegativeValues(Boolean)	GRAPHOMATECHART_1.setStackedBarNegativeValues(true);	Sets the proportional Space Factor of the Base Chart (relative to the other Space Factors).
setStackedBarSumsVisible(Boolean)	GRAPHOMATECHART_1.setStackedBarSumsVisible(true);	Sets whether the sums in the stacked bars are displayed.
setStructureMode(Boolean)	GRAPHOMATECHART_1.setStructureMode(true);	This function can be used to toggle the structure mode on or off.
setSuppressRepeatingCategoryLabels(String)	GRAPHOMATECHART_1.setSuppressZeroLabels("true,false,false,true");	Sets the configuration of the property Suppress Repeating Category Labels. Expects a comma separated string with one statement for each row of the Category Labels whether repeating category labels should be suppressed.
setSuppressZeroLabels(Boolean)	GRAPHOMATECHART_1.setSuppressZeroLabels(true);	This function sets whether Zero Labels are displayed.
setSynchronizeAxesPosition(Boolean)	GRAPHOMATECHART_1.setSynchronizeAxesPosition(true);	This function sets whether the axes position of the chart are synchronized with the Comparison Group.
setSynchronizePadding(Boolean)	GRAPHOMATECHART_1.setSynchronizePadding(true);	This function sets whether the padding of the chart is synchronized with the Comparison Group.
setSynchronizeScaling(Boolean)	GRAPHOMATECHART_1.setSynchronizeScaling(true);	This function sets whether the scaling of the chart is synchronized with the Comparison Group.
setThresholdLinesConfig(String)	GRAPHOMATECHART_1.setThresholdLinesConfig(['{"visible":true,"label":"Mean","type":"Average","base":1,"lineSize":1,"color":"#0000FF","lineStyle":"solid"}'])	This function replaces the current threshold lines config with the given JSON. For an example of the configuration JSON, please copy an example from the standard property sheet. In the example shown here, the mean value for the series 1 with the label "Mean" is displayed.
setTitle(String)	den Titel auf den übergebenen String setzen GRAPHOMATECHART_1.setTitle("Sales in Mio. EUR 2013 ACT and BUD North West");	Sets the title of the chart. Expects value as a string. Line breaks (\n) are interpreted as such.
setUseOutlierThreshold(Boolean)	GRAPHOMATECHART_1.setUseOutlierThreshold(true);	This function sets whether the outlier threshold is used.
setUserHighlightColor(String)	GRAPHOMATECHART_1.setUserHighlightColor("#ff0000,#00ff00,#0000ff");	Sets a list of disposed colors for the User Highlights. Expects a comma separated string-value. The colors of the list are switched at each click (during runtime) until the desired color is set.
setUserHighlightList(String)	GRAPHOMATECHART_1.setUserHighlightList("[[1,3],[2,3][1,5]]");	Sets a list of disposed colors for the User Highlights. Expects a comma separated string-value. The colors of the list are switched at each click (during runtime) until the desired color is set.
setValueAxisVisible(Boolean)	GRAPHOMATECHART_1.setValueAxisVisible(true);	This function can be used to turn the visibility of the value axis on or off.
setVisibleElementLabels(String)	GRAPHOMATECHART_1.setVisibleElementLabels("1,2,4");	Sets the visible element labels. Expected is a comma separated list of series indices, e.g. 1,2,3,4,5,6,7,8,9
setVisibleSeries(String)	nur die Datenserien mit den Indices 1 und 2 anzeigen GRAPHOMATECHART_1.setVisibleSeries("1,2");	Sets the listing of indices of visible series in the chart. Expects value as comma-separated string.
showStructureMode (Boolean)	GRAPHOMATECHART_1.showStructureMode(true);	Sets the orientation of the chart. If the value is TRUE, the chart will be shown in structure mode, otherwise in time mode.

## Examples

Dynamically filter a Data Source using a dropdown box

In order to allow a user to dynamically filter on a dimension of a Data Source, we first have to fill a dropdown box with all valid value. We first retrieve on the canvas event On Startup of the selected Data Source a list of all members of the dimension 0D\_NWI\_ACOD with the command `getMemberList` autocomplete proposes all parameters of `getMemberList`.

Using `setItem` puts the list in the dropdown box.

```
DROPDOWN_1.setItem(DS_1.getMemberList("0D_NWI_ACOD",
MemberPresentation.EXTERNAL_KEY, MemberDisplay.TEXT, 20);
```

On application start the dropdown box will be filled with the members of the dimension 0D\_NWI\_ACOD. If a user selects an element of the list, we can adapt the filter of the Data Source with the following script, which is set on the event On Element Clicked of the dropdown box:

```
DS_1.setFilterExt("0D_NWI_ACOD", DROPDOWN_1.getSelectedValue());
```

## Deleting filter of data source

The following command allows you to delete a filter of a Data Source:

```
DS_1.setFilterExt("0D_NWI_ACOD", DROPDOWN_1.getSelectedValue());DS_1.clearAllFilters();Filter a Data Source via drilldown with graphomate charts
```

Setting a filter on a data source by drilldown

Before it is possible to make a drilldown with the graphomate charts, the following condition must be fulfilled: The property `categoryLabels` must be filled with data! The Category Labels therefore do not have to be visible, but can be visible. This is necessary, because the chart receives its information about

dimension and members only from this property.

If this condition is fulfilled, it is possible to receive the Key of the selected dimension member with the following script, which is set on the event On Element Clicked of graphomate charts:

```
var key = GRAPHOMATECHART_1.getSelectedMemberKey("0D_NWI_RCOD");
```

The Key of the selected dimension must be passed as a parameter.

This retrieved Memberkey can be set as a filter on a Data Source – similar to the example with the dropdown box:

```
DS_1.setFilter("0D_NWI_RCOD", key);
```

or – in a combined command:

```
DS_1.setFilter("0D_NWI_RCOD", GRAPHOMATECHART_1.getSelectedMemberKey("0D_NWI_RCOD"));
```

### Set title after drilldown

In order to adapt the title to the drilldown – for example, to show the selected area, it is possible to read out the Membertext in the same way as Memberkey.

```
var region = GRAPHOMATECHART_1.getSelectedMemberText("0D_NWI_RCOD");
```

This text can be arbitrarily used – here it is written in the title of the chart.

```
GRAPHOMATECHART_1.setTitle("Selected region: " + region);
```

### Realize a zoom function

In order to implement a zoom functionality on a chart, the scaling mode must be set to Manual. This can be either preset or realized via scripting:

```
GRAPHOMATECHART_1.setScaleModeManual();
```

After that the following script must be attached to the onElementClicked event:

```
var value = GRAPHOMATECHART_1.getClickedPositionValue();
GRAPHOMATECHART_1.setManualScaleMinimumFloat(value);
```

At first the relative value of the clicked position is determined. In the second line, this value is then used as the manual scale minimum and by doing that a zooming function is established.

## 1.4 Scripting

The graphomate charts extension fully supports the scripting language of Designer. Therefore it is possible to control graphomate charts interactively at runtime. For example, it is possible to set new data, change the title or scales. Furthermore, it is possible to use drilldown functions of the extension and create interactive dashboards.

Example: Dynamic filtering of a Data Source with a dropdown box

In order to use a dynamic filter on a dimension of a Data Source, we first need to fill a dropdown box with all valid values to enable the user to choose from the data later on.

We use `getMemberList` on the canvas-event On Startup to retrieve a list of all members of the dimension 0D\_NWI\_ACOD of the selected Data Source. Autocomplete shows all parameters of `getMemberList`.



We can then fill the dropdown box with the retrieved list using setItem.

```
var memberList = DS_1.getMemberList("0D_NWI_ACOD", MemberPresentation.INTERNAL_KEY, MemberDisplay.TEXT, 10);  
DROPDOWN_1.setItems(memberList);
```

The dropdown box will be filled with the members of Dimension 0D\_NWI\_ACOD on application start.

If a user selects an element from the list, we can adapt the filter of the Data Source by using the following script that is put on the events On Element Clicked:

```
var selectedMemberKey = DROPDOWN_1.getSelectedValue();  
DS_1.setFilter("0D_NWI_ACOD", selectedMemberKey);
```

Example: Save User Highlights by bookmarks

At runtime, you can set so-called User Highlights to selected values (see [Interactivity at runtime](#)). To save this state of the dashboard, we need two buttons.

To save the id of the bookmark, for simplicity we use a text field (in the example TEXT\_2), in which the id is written. This step can also hide visually using a variable for the id.

The following skript is assigned to the save button in order to save the actual state of the dashboard to variable named id. The variable is then being backed-up into a text box:

```
var id = Bookmark.saveBookmark("Bookmark");  
TEXT_2.setText(id);
```

The next script is assigned to the load button in order to read the id from the text box and restore this bookmark state:

```
var id = TEXT_2.getText();  
Bookmark.loadBookmark(id);
```

You find more explanations and examples for the scripting language in our [Scripting Documentation](#).