



# Scripting Dokumentation der graphomate charts 2.8 für SAP Lumira Designer 2.x

Version 2.8 - Stand Juni 2019

<https://www.graphomate.com>

# 1 Scripting-Dokumentation

## Scripting-Dokumentation der graphomate charts für SAP Lumira Designer

Version 2.8 - Stand Juni 2019

### 1.1 Einführung

Die graphomate charts Extension unterstützt die Skriptsprache von Designer. Somit ist es möglich, die Extension zur Laufzeit interaktiv zu verwenden. Beispielsweise können neue Daten gesetzt, Titel bearbeitet und Skalierungen verändert werden. Weiterhin ist es möglich, Drilldown-Funktionalitäten der Extension zu verwenden und somit interaktive Dashboards zu erstellen.

Eine Auflistung der gegenwärtigen Funktionen, Ereignisse und Beispiele zur Verwendung liefert dieses Dokument. Wir sind bemüht, den Funktionsumfang ständig zu erweitern. Sollten Ihnen bestimmte Funktionen fehlen, teilen Sie uns dies gerne unter <http://www.graphomate.com/feedback> mit.

### 1.2 Ereignisse

Die nachfolgend aufgelisteten Ereignisse stehen zur Verfügung. Auf diesen können im Property Sheet unter dem Punkt Events vom Benutzer geschriebene Skripte hinterlegt werden.

Ereignis Name	Beschreibung
On Element Clicked	Wird ausgelöst, wenn der Benutzer zur Laufzeit auf ein Element der Komponente in der Zeichenfläche klickt. Um einen Drilldown zu ermöglichen, muss an dieser Stelle auf die Benutzereingabe reagiert werden. Zum Zeitpunkt dieses Ereignisses werden die Rückgabewerte folgender Funktionen aktualisiert: <code>getSelectedMemberKey()</code> , <code>getSelectedMemberText()</code> , <code>getClickedElementIndex()</code> , <code>getClickedSeriesIndex()</code>
On new data arrival	Wird ausgelöst, wenn das Chart aktualisierte Daten erhält. Dies geschieht zum Beispiel, nachdem sich die Dimensionen einer Data Source geändert haben oder wenn Filter verändert wurden. Eine Anpassung der Datenarten (Data Types) oder des Berechnungspfad für den Wasserfall (Waterfall Calculation Path) wäre an dieser Stelle zum Beispiel angebracht.

### 1.3 Funktionen

Folgende Funktionen stehen für die graphomate charts Extension zur Verfügung. Diese können innerhalb des Skript-Editors von Designer angesprochen werden.

Hinweis zu Umlauten, Sonder- und Leerzeichen

Generell gilt: Umlaute, Sonder- und Leerzeichen werden aus technischen Gründen im Advanced Property Sheet prozentkodiert (siehe <http://de.wikipedia.org/wiki/URL-Encoding>).

Beispiel:

Müslis - Entwicklung der Verkäufe in % gegenüber dem Vorjahr

wird zu:

`M%C3%BCslis%20Entwicklung%20der%20Verk%C3%A4ufe%20in%20%25%20gegen%C3%BCber%20dem%20Vorjahr`

Dies macht es erforderlich, dass auch bei der Übertragung von String-Werten aus der Skriptsprache oder dem Standard Property Sheet eine Prozentkodierung vorgenommen wird. (Im Internet sind dazu zahlreiche Tools erhältlich, bspw. <http://www.url-encoder.de>.)

Getter Name	Beispiel	Rückgabotyp	Beispiel Rückgabewert	Beschreibung
<code>getAxisLabels()</code>	<code>GRAPHOMATECHART_1.getAxisLabels();</code>	String	Budget, Actual	Gibt die Achsenlabels des Charts als kommaseparierten String wieder.
<code>getAxisThickness()</code>	<code>GRAPHOMATECHART_1.getAxisThickness();</code>	Integer	3.0	Gibt die Achsendicke als Integer-Wert zurück.
<code>getBarWidth()</code>	<code>GRAPHOMATECHART_1.getBarWidth();</code>	Float	5.0	Gibt die Breite der Balken als Float zurück.
<code>getBaseChartVisible()</code>	<code>GRAPHOMATECHART_1.getBaseChartVisible();</code>	Boolean	true	Gibt zurück, ob das Base-Chart sichtbar ist
<code>getCalculationPath()</code>	<code>GRAPHOMATECHART_1.getCalculationPath();</code>	String	<code>+,+,+,-,-,=, ,s+,+,=-,-</code>	Gibt den Waterfall Calculation Path als kommaseparierten String zurück.
<code>getCalculateDeviationsFromNull()</code>	<code>GRAPHOMATECHART_1.getCalculateDeviationsFromNull();</code>	Boolean	true	Gibt zurück, ob NULL-Werte zur Berechnung von Abweichungen herangezogen werden sollen.
<code>getCategoryAxisVisible()</code>	<code>GRAPHOMATECHART_1.getCategoryAxisVisible();</code>	Boolean	true	Gibt zurück, ob die Kategorieachse sichtbar ist.
<code>getCategoryCount()</code>	<code>GRAPHOMATECHART_1.getCategoryCount();</code>	Integer	12	Gibt zurück, ob die darzustellende Kategorieanzahl beschränkt ist. Gibt die Beschränkung als Integer-Wert zurück. Der Wert 0 stellt keine Beschränkung dar.
<code>getCategoryLabelRotation()</code>	<code>GRAPHOMATECHART_1.getCategoryLabelRotation();</code>	Integer	45	Gibt den Winkel zurück, um den die Categorylabel gedreht werden.
<code>getCategoryWidth()</code>	<code>GRAPHOMATECHART_1.getCategoryWidth();</code>	Integer	20	Gibt die Breite der Kategorien als Integer-Wert zurück, wenn diese von fester Breite sind.

getChartType()	GRAPHOMATECHART_1.getChartType();	String	line	Gibt den eingestellten Diagramm-Typen als String-Wert zurück..
getClickedElementIndex()	GRAPHOMATECHART_1.getClickedElementIndex();	Integer	3	Liefert den Index des als letztes geklickten Elementes/Kategorie zurück.
getClickedPositionValue()	GRAPHOMATECHART_1.getClickedPositionValue();	Float	234234.432	Liefert den relativen Wert der Klickposition im Diagramm zurück. Dies kann benutzt werden, um eine Zoom-Funktion zu realisieren, indem der zurückgegebene Wert als manuelles Skalierungsminimum an das Diagramm übergeben wird.
getClickedSeriesIndex()	GRAPHOMATECHART_1.getClickedSeriesIndex();	Integer	1	Liefert den Index der als letztes geklickten Datenserie zurück.
getComparisonGroup()	GRAPHOMATECHART_1.getComparisionGroup();	String	A	Gibt den Selektionsstring des Performance Measure 1 zurück.
getContextMenuCustomEventName(Integer index)	GRAPHOMATECHART_1.getContextMenuCustomEventName(1);	String	Custom Entry 1	Gibt den Namen eines benutzerdefinierten Events (Index-1 basiert) im Context Menu zurück.
getContextMenuCustomEventVisibility(Integer index )	GRAPHOMATECHART_1.getContextMenuCustomEventVisibility(1);	Boolean	true	Gibt die Sichtbarkeit eines benutzerdefinierten Events (Index-1 basiert) im Context Menu zurück.
getContextMenuOptionVisibility(Integer index)	GRAPHOMATECHART_1.getContextMenuOptionVisibility(1);	Boolean	true	Gibt die Sichtbarkeit einer Option im Context Menu wieder. Die verfügbaren Optionen können über folgende Indices angesprochen werden: 1 für Sorting, 2 für Filters, 3 für Context Sensitive Filters, 4 für Filter and Hide, 5 für Clear Filters, 6 für Drilldown, 7 für Drillup, 8 für Zero Display, 9 für Member Display, 10 für Position of Totals, 11 für Total Display.
getDatatypes(Integer seriesIndex)	GRAPHOMATECHART_1.getDatatypes(1);	String	PY,PY,PY,PY,AC,AC,AC,AC,PL,PL,PL,PL	Diese Funktion liefert die Datenarten (Data Types) der angegebenen Datenserie seriesIndex als kommaseparierten String zurück.
getDeviationBadColor()	GRAPHOMATECHART_1.getDeviationBadColor();	String	#ff3300	Diese Funktion gibt die Farbe zurück, welche für "schlechte" Abweichungen benutzt wird.
getDeviationGoodColor()	GRAPHOMATECHART_1.getDeviationGoodColor();	String	#2e2e2e	Diese Funktion gibt die Farbe zurück, welche für "gute" Abweichungen benutzt wird.
getDeviationVisibility()	GRAPHOMATECHART_1.getDeviationVisibility();	String	0,1,4	Gibt eine Auflistung der Indices sichtbarer Abweichungscharts als kommaseparierten String wieder.
getElementsOffset()	GRAPHOMATECHART_1.getElementsOffset();	Float	2.0	Diese Funktion gibt das Offset zwischen den Elementen einer Kategorie als Float zurück.
getEndingLineOverlap()	GRAPHOMATECHART_1.getEndingLineOverlap();	Integer	3	Gibt zurück, wie weit die Ending Lines über die Balken überstehen.
getEndingLineThickness()	GRAPHOMATECHART_1.getEndingLineThickness();	Integer	2	Gibt zurück, wie dick die Ending Lines sind.
getEndingLineVisible()	GRAPHOMATECHART_1.getEndingLineVisible();	Boolean	false	Gibt zurück, ob über den Balken Ending Lines angezeigt werden.
getExtendedNumberFormat()	GRAPHOMATECHART_1.getExtendedNumberFormat();	String	- . , 1 1 %25	Gibt das Nummernformat für den Extended-Modus zurück. Der Formatstring kann im Handbuch nachgelesen werden.
getFixedCategoryWidth()	GRAPHOMATECHART_1.getFixedCategoryWidth();	Boolean	true	Diese Funktion gibt als Boolean zurück, ob die Kategoriebreite fix ist.
getHighlightMode()	GRAPHOMATECHART_1.getHighlightMode();	String	single	Diese Funktion gibt die aktuelle Einstellung für den Highlight Mode zurück.
getLabelFormatMode()	GRAPHOMATECHART_1.getLabelFormatMode();	String	extended	Gibt zurück, welcher Modus für die Label-Formatierung verwendet wird.
getLabelPositioningMode()	GRAPHOMATECHART_1.getLabelPositioningMode();	String	auto	Gibt die aktuelle Einstellung zurück, ob, und wenn ja wie die Value Labels im Chart positioniert werden.
getLockData()	GRAPHOMATECHART_1.getLockData();	Boolean	true	Gibt zurück, ob die Komponente Daten-Updates ignoriert.
getManualScalePercentMax()	GRAPHOMATECHART_1.getManualScalePercentMax();	Float	80.5	Gibt das manuelle prozentuale Skalierungsmaximum als Float zurück.
getManualScalePercentMin()	GRAPHOMATECHART_1.getManualScalePercentMin();	Float	-80.5	Gibt das manuelle prozentuale Skalierungsminimum als Float zurück.
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}]	Gibt eine Liste der aktuellen Multi Highlights (hervorgehobene Werte überhalb des Charts) als JSON-String zurück, welcher ein Array enthält, in welchem für jedes Multihighlight ein Objekt enthalten ist. Das Objekt ist nach dem Schema {"startElement":1,"endElement":2,"startSeries":1,"endSeries":1} aufgebaut.
getNegativeDeviationIsGood()	GRAPHOMATECHART_1.getNegativeDeviationIsGood();	Boolean	true	Gibt zurück, ob negative Abweichungen als positiv dargestellt werden sollen.
getNegativeOutlierThreshold()	GRAPHOMATECHART_1.getNegativeOutlierThreshold();	Float	-80.0	Gibt die negative Ausreißergrenze als Float zurück.
getNumberFormat()	GRAPHOMATECHART_1.getNumberFormat();	String	0,0.0	Gibt das Nummernformat für den Basic-Modus zurück. Der Formatstring entspricht dem Format von numeral.js, einzusehen unter <a href="http://www.numeraljs.com">http://www.numeraljs.com</a> .
getNumberFormatPercentage()	GRAPHOMATECHART_1.getNumberFormatPercentage();	String	0%	Gibt das Nummernformat für die Prozentwerte im Basic-Modus zurück. Der Formatstring entspricht dem Format von numeral.js, einzusehen unter <a href="http://www.numeraljs.com">http://www.numeraljs.com</a> .
getPositiveOutlierThreshold()	GRAPHOMATECHART_1.getPositiveOutlierThreshold();	Float	80.0	Gibt die positive Ausreißergrenze als Float zurück.
getProperty(String)	GRAPHOMATECHART_1.getProperty("labelFormatMode");	String		Gibt den Wert der angeforderten Property als String zurück.
getRulesets()	GRAPHOMATECHART_1.getRulesets();	String	Ruleset1,Ruleset2	Gibt eine Liste von Rulesets zurück, die auf das gewählte Chart angewendet werden.
getSelectedMember(Dimension dimensionKey)	GRAPHOMATECHART_1.getSelectedMember("Keyfigures");	Member	Net Sales	Nachdem der User auf dem Chart ein Element/eine Kategorie geklickt hat, gibt diese Funktion den Member der spezifizierten Dimension dimensionKey zurück. Der Member stellt verschiedene Properties wie text, internalKey und externalKey zur Verfügung.

getSelectedMemberKey(String dimensionKey)	GRAPHOMATECHART_1.getSelectedMemberKey("0D_NWI_RCOD");	String	10273	Nachdem der User auf dem Chart ein Element/eine Kategorie geklickt hat, gibt diese Funktion den Schlüssel (Key) des Members der spezifizierten Dimension dimensionKey zurück.
getSelectedMemberText(String dimensionKey)	GRAPHOMATECHART_1.getSelectedMemberText("0D_NWI_RCOD");	String	North West	Nachdem der User auf dem Chart ein Element / eine Kategorie geklickt hat, gibt diese Funktion den Text des Members der spezifizierten Dimension dimensionKey zurück. Die Rückgabe ist analog zu der Funktion getSelectedMemberKey().
getSeparatorLength()	GRAPHOMATECHART_1.getSeparatorLength();	Integer	5	Gibt die Länge der Trennzeichen (Seperators) in Pixeln zurück.
getSeparatorMode()	GRAPHOMATECHART_1.getSeparatorMode();	String	datatypes	Gibt den Separator-Mode zurück.
getSeparatorPositions()	GRAPHOMATECHART_1.getSeparatorPositions();	String	2,4,6	Gibt eine Auflistung der (manuell eingegebenen) Trennzeichenpositionen als komma-separierten String wieder.
getSeparatorSourceRowIndex()	GRAPHOMATECHART_1.getSeparatorSourceRowIndex();	Integer	3	Gibt den Index der Zeile/Serie zurück, aus der sich die Trennzeichen (Separators) ableiten. Abhängig vom Separator-Mode (setSeperatorMode()).
getSeparatorThickness()	GRAPHOMATECHART_1.getSeparatorThickness();	Integer	2	Gibt die Dicke der Trennzeichen (Seperators) in Pixeln zurück.
getSeries(Integer seriesIndex)	GRAPHOMATECHART_1.getSeries(1);	String	{"Keyfigures":"0D_NWI_NSAL"}	Liefert die Datenbeschreibung der angegebenen Datenserie seriesIndex als JSON-String zurück.
getSeriesLabels()	GRAPHOMATECHART_1.getSeriesLabels();	String	Budget,Actual,Target	Gibt die Serienlabels des Charts als kommaseparierten String wieder.
getShowAxisLabelsLeft()	GRAPHOMATECHART_1.getShowAxisLabelsLeft();	Boolean	true	Gibt zurück, ob die Achsenbeschriftungen auf der linken Seite angezeigt werden.
getShowAxisLabelsRight()	GRAPHOMATECHART_1.getShowAxisLabelsRight();	Boolean	true	Gibt zurück, ob die Achsenbeschriftungen auf der rechten Seite angezeigt werden.
getShowCategoryLabels()	GRAPHOMATECHART_1.getShowCategoryLabels();	Boolean	false	Gibt zurück, ob die Kategoriebeschriftungen (Category Labels) angezeigt werden.
getShowDatatypesInAxis()	GRAPHOMATECHART_1.getShowDatatypesInAxis();	Boolean	true	Gibt zurück, ob die Datenarten (Data Types) in der Achse oder in den Elementen angezeigt werden.
getShowTitle()	GRAPHOMATECHART_1.getShowTitle();	Boolean	true	Gibt zurück, ob der Titel angezeigt wird.
getShowTooltips()	GRAPHOMATECHART_1.getShowTooltips();	Boolean	false	Gibt zurück, ob die Tooltips in der Runtime angezeigt werden.
getSpaceFactorBaseChart()	GRAPHOMATECHART_1.getSpaceFactorBaseChart();	Float	0.8	Gibt den Platz-Faktor zurück, welchen das Base Chart anteilig (im Verhältnis zu den anderen Platz-Faktoren) einnimmt.
getSpaceFactorDeviationChart()	GRAPHOMATECHART_1.getSpaceFactorDeviationChart();	Float	0.2	Gibt den Platz-Faktor zurück, welchen ein Abweichungschart anteilig (im Verhältnis zu den anderen Platz-Faktoren) einnimmt.
getStackedBarLabelPosition()	GRAPHOMATECHART_1.getStackedBarLabelPosition();	String	middle	Gibt die Position für die Labels im Stapel zurück.
getStackedBarNegativeValues()	GRAPHOMATECHART_1.getStackedBarNegativeValues();	Boolean	true	Diese Funktion gibt zurück, ob negative Stapel erlaubt sind.
getStackedBarSumsVisible()	GRAPHOMATECHART_1.getStackedBarSumsVisible();	Boolean	true	Diese Funktion gibt zurück, ob die Summen auf den Stapelbalken angezeigt werden.
getStructureMode()	GRAPHOMATECHART_1.setStructureMode();	Boolean	true	Diese Funktion gibt zurück, ob sich ein Diagramm aktuell im Structure Mode befindet (identisch mit showStructureMode()).
getSuppressRepeatingCategoryLabels()	GRAPHOMATECHART_1.getSuppressRepeatingCategoryLabels();	String	true,true,false,false,true	Gibt die Einstellung zurück, die für die Eigenschaft Suppress Repeating Category Labels konfiguriert wurde. Zurückgegeben wird ein kommaseparierter String.
getSuppressZeroLabels()	GRAPHOMATECHART_1.getSuppressZeroLabels();	Boolean	true	Gibt zurück, ob Null-Labels angezeigt werden.
getSynchronizeAxesPosition()	GRAPHOMATECHART_1.getSynchronizeAxesPosition();	Boolean	true	Diese Funktion gibt ein Boolean zurück, ob die Achsenpositionen des Charts mit der Comparison Group abgeglichen werden.
getSynchronizePadding()	GRAPHOMATECHART_1.getSynchronizePadding();	Boolean	false	Diese Funktion gibt ein Boolean zurück, ob die Ränder des Charts mit der Comparison Group abgeglichen werden.
getSynchronizeScaling()	GRAPHOMATECHART_1.getSynchronizeScaling();	Boolean	true	Diese Funktion gibt ein Boolean zurück, ob die Skalierung des Charts mit der Comparison Group abgeglichen wird.
getTitle()	GRAPHOMATECHART_1.getTitle();	String	Sales in Mio. EUR  2013 ACT and BUD  North West	Gibt den Titel des Charts als String wieder. Zeilenumbrüche werden durch einen senkrechten Strich ( ) repräsentiert.
getUseOutlierThreshold()	GRAPHOMATECHART_1.getUseOutlierThreshold();	Boolean	true	Gibt zurück, ob die Ausreißergrenze verwendet wird.
getUserHighlightColor()	GRAPHOMATECHART_1.getUserHighlightColor();	String	#ff0000,#00ff00,#0000ff	Gibt eine Liste der hinterlegten User Highlight Farben als kommaseparierten String zurück. Die Farben werden in der angegebenen Reihenfolge bei den User Highlights (zur Laufzeit hervorgehobene Werte) durchgeschaltet, bis die Wunschfarbe eingestellt ist.
getUserHighlightList()	GRAPHOMATECHART_1.getUserHighlightList();	String	[[1,3],[2,3][1,5]]	Gibt eine Liste der aktuellen User Highlights (zur Laufzeit hervorgehobene Werte) als JSON-String zurück, der die Positionen der Highlights jeweils mit zwei Zahlen übergibt. Dabei gibt die erste Zahl die Datenreihe an, während die zweite Zahl die Position des Wertes in der Datenreihe darstellt.
getValueAxisVisible()	GRAPHOMATECHART_1.getValueAxisVisible();	Boolean	true	Gibt zurück, ob die Value Axis dargestellt wird oder nicht.
getVisibleElementLabels()	GRAPHOMATECHART_1.getVisibleElementLabels();	String	1,3,5	Gibt die Visible Element Labels als Liste kommaseparierter Indices zurück
getVisibleSeries()	GRAPHOMATECHART_1.getVisibleSeries();	String	0,1,4	Gibt eine Auflistung der Indices sichtbarer Serien als kommaseparierten String wieder.

refresh(String)

Zeichnet das Chart neu. Dies ist ein Workaround für den Fall, dass das Chart nicht neu gezeichnet wurde, nachdem sich die Daten zur Laufzeit durch Einwirkung der Skriptsprache auf die Data Source geändert haben. Dies kann passieren, nachdem zum Beispiel Filter geändert oder Dimensionen hinzugefügt oder entfernt wurden. Sollte sich das Chart unerwartet nicht neu zeichnen, dann sollte dieser Aufruf helfen:

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

Hierbei ist es unbedingt erforderlich, als Parameter val den Wert von .refreshValue() zu übergeben!

refreshValue() ist eine Hilfsfunktion für refresh(). Sie hat keinen anderweitigen Nutzen.

scalingHelperMode(String)

Setzt den Skalierungshelfer. Erlaubte Werte sind none, line, area.

Beispiel: den Skalierungshelfer aktivieren

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

Setter Name	Beispiel	Beschreibung
addRuleset(String name)	GRAPHOMATECHART_1.addRuleset("Ruleset2");	Fügt ein Ruleset zu der Liste der auf ein Chart angewendeten Rulesets hinzu. Rulesets mit gleichem Namen werden nicht mehrfach angewendet.
removeRuleset(String name)	GRAPHOMATECHART_1.removeRuleset("Ruleset2");	Entfernt ein Ruleset aus der Liste der auf ein Chart angewendeten Rulesets. Nicht vorhandene Rulesets werden ignoriert.
setAxisLabels(String)	GRAPHOMATECHART_1.setAxisLabels("Net Sales,Returns");	Setzt die Achsenlabels des Charts. Erwartet value als kommaseparierten String. Das erste Element wird als linkes, das zweite Element als rechtes Achsenlabel interpretiert.
setAxisThickness(Integer)	GRAPHOMATECHART_1.setAxisThickness(3);	Setzt die Achsendicke.
setBarWidth(Float)	GRAPHOMATECHART_1.setBarWidth(3);	Setzt die Breite der Balken als Float. Werte kleiner oder gleich 1 werden als Prozent interpretiert.
setBaseChartVisible(Boolean)	GRAPHOMATECHART_1.setBaseChartVisible(true);	Setzt, ob das Base-Chart sichtbar ist.
setCalculationPath(String)	GRAPHOMATECHART_1.setCalculationPath("+,+,+,-,-,=, ,s+,+,=,-,-");	Setzt den Wasserfall-Berechnungspfad (Waterfall Calculation Path). Erwartet value als kommaseparierten String.
setCalculateDeviationsFromNull (Boolean)	GRAPHOMATECHART_1.setCalculateDeviationsFromNull(true);	Diese Funktion setzt, ob NULL-Werte zur Berechnung von Abweichungen herangezogen werden sollen.
setCategoryAxisVisible(Boolean)	GRAPHOMATECHART_1.setCategoryAxisVisible(true);	Setzt, ob das Base-Chart sichtbar ist.
setCalculateDeviationsFromNull (Boolean)	GRAPHOMATECHART_1.setCalculateDeviationsFromNull(true);	Diese Funktion setzt, ob NULL-Werte zur Berechnung von Abweichungen herangezogen werden sollen.
setCategoryAxisVisible(Boolean)	GRAPHOMATECHART_1.setCategoryAxisVisible(true);	Diese Funktion setzt die Sichtbarkeit der Kategorieachse.
setCategoryCount(Integer)	GRAPHOMATECHART_1.setCategoryCount(12);	Setzt eine Begrenzung der darzustellenden Kategorieanzahl. Der Wert 0 hebt die Beschränkung auf.
setCategoryLabelRotation(int rotation)	GRAPHOMATECHART_1.setCategoryLabelRotation(45);	Dreht die Category Label um den übergebenen Winkel. Die Drehung findet gegen den Uhrzeigersinn statt.
setCategoryWidth(Integer)	GRAPHOMATECHART_1.setCategoryWidth(3);	Diese Funktion setzt die Breite der Kategorien. Sie wird jedoch nur interpretiert, wenn fixedCategoryWidth gesetzt ist.
setChartType(String)	GRAPHOMATECHART_1.setChartType("line");	Setzt den Diagramm-Typen. Mögliche String-Werte sind: bar, line, offsetbar, waterfall, pin, stackedbar.
setComparisonGroup(String id)	GRAPHOMATECHART_1.setComparisonGroup("A");	Setzt die Comparison Group des Charts als String. Charts mit der gleichen Comparison Group werden gleich skaliert und ausgerichtet. Weitere Feinheiten können über die folgenden Parameter eingestellt werden: getSynchronizePadding(), setSynchronizePadding(), getSynchronizeScaling(), setSynchronizeScaling(), getSynchronizeAxesPosition(), setSynchronizeAxesPosition()
setContextMenuCustomEventName(Integer index, String name)	GRAPHOMATECHART_1.setContextMenuCustomEventName(1, "Script 1");	Setzt den Namen eines benutzerdefinierten Events (Index-1 basiert) im Context Menu.
setContextMenuCustomEventVisibility(Integer index, Boolean visibility)	GRAPHOMATECHART_1.setContextMenuCustomEventVisibility(1, false);	Setzt die Sichtbarkeit eines benutzerdefinierten Events (Index-1 basiert) im Context Menu.
setContextMenuOptionVisibility(Integer index, Boolean visibility)	GRAPHOMATECHART_1.setContextMenuOptionVisibility(1, false);	Setzt die Sichtbarkeit einer Option im Context Menu. Die verfügbaren Optionen können über folgende Indices angesprochen werden: 1 für Sorting, 2 für Filters, 3 für Context Sensitive Filters, 4 für Filter and Hide, 5 für Clear Filters, 6 für Drilldown, 7 für Drillup, 8 für Zero Display, 9 für Member Display, 10 für Position of Totals, 11 für Total Display.
setDatatypeAt(Integer series, Integer index, String datatype)	GRAPHOMATECHART_1.setDatatypeAt(1, 1, "AC");	Diese Funktion setzt die übergebene Datenart für ein einzelnes Element einer Datenserie.
setDatatypeForSeries(Integer series, Integer count, String datatype)	GRAPHOMATECHART_1.setDatatypeForSeries(1, 5, "AC");	Diese Funktion setzt die übergebenen Datenarten für die übergebene Datenserie. Die Anzahl der generierten Datenarten-Einträge entspricht dem Übergabeparameter count. Der Aufruf GRAPHOMATECHART_1.setDatatypeForSeries(1, 5, "AC"); ist äquivalent mit 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");	Mit dieser Funktion werden die Datenarten (Data Types) der angegebenen Datenserie seriesIndex gesetzt. Erwartet value als kommaseparierten String. Beispiel: Datenarten der ersten fünf Elemente der ersten Datenserie auf "PP" setzen
setDeviationBadColor(String)	GRAPHOMATECHART_1.setDeviationBadColor("#ff3300");	Diese Funktion setzt die Farbe, welche für "schlechte" Abweichungen benutzt wird. Es wird ein String im Format #ff3300 erwartet.

setDeviationGoodColor(String)	GRAPHOMATECHART_1.setDeviationGoodColor("#2e2e2e");	Diese Funktion setzt die Farbe, welche für "gute" Abweichungen benutzt wird. Es wird ein String im Format #2e2e2e erwartet.
setDeviationLabels(String labels, Boolean visibleOnly)	GRAPHOMATECHART_1.setDeviationLabels("Abs. Dev. FC,Perc. Dev. BUD", true);	Diese Funktion setzt die Beschriftungen aller Abweichungsserien. Wenn der zweite Wert auf true gesetzt wird, werden nur die sichtbaren/eingeblendeten Abweichungsserien berücksichtigt.
setDeviationVisibility (String)	GRAPHOMATECHART_1.setDeviationVisibility("0,1");	Setzt die Auflistung der Indices sichtbarer Abweichungsserien. Erwartet value als kommaseparierten String.
setElementsOffset(Float)	GRAPHOMATECHART_1.setElementsOffset(5);	Diese Funktion setzt das Offset zwischen den Elementen einer Kategorie. Werte kleiner oder gleich 1 werden als Prozent interpretiert.
setEndingLineOverlap(Integer)	GRAPHOMATECHART_1.setEndingLineOverlap(5)	Setzt, wie weit die Ending Lines über die Balken überstehen.
setEndingLineThickness(Integer)	GRAPHOMATECHART_1.setEndingLineThickness(3);	Setzt, wie dick die Ending Lines sind.
setEndingLineVisible(Boolean)	GRAPHOMATECHART_1.setEndingLineVisible(true);	Setzt, ob über den Balken Ending Lines angezeigt werden sollen.
setExtendedNumberFormat(String)	GRAPHOMATECHART_1.setExtendedNumberFormat("- . , 1 ");	Setzt das Nummernformat für den Extended -Modus. Der Formatstring kann im Handbuch nachgelesen werden.
setExtendedNumberFormatPercentage(String)	GRAPHOMATECHART_1.setExtendedNumberFormatPercentage("- . , 1 1 %25");	Setzt das Nummernformat für die Prozentwerte im Extended -Modus. Der Formatstring kann im Handbuch nachgelesen werden.
setFixedCategoryWidth(Boolean)	GRAPHOMATECHART_1.setFixedCategoryWidth(true);	Diese Funktion setzt, ob die Kategoriebreite fix sein soll. Ist dieser Wert nicht gesetzt, wird die Breite automatisch ermittelt.
setHighlightEndIndex(Integer)	GRAPHOMATECHART_1.setHighlightEndIndex(4);	Setzt den Kategorienindex value des Endwertes der Hervorhebung. Analog zu setHighlightStartIndex().
setHighlightEndSeries(Integer)	GRAPHOMATECHART_1.setHighlightEndSeries(3);	Setzt den Serienindex value der Endserie der Hervorhebung. Analog zu setHighlightStartSeries().
setHighlightMode(String)	GRAPHOMATECHART_1.setHighlightMode("single");	Mit dieser Funktion kann der Highlight Mode geändert werden. Gültige Übergabewerte sind none, single und multi.
setHighlightStartIndex(Integer)	GRAPHOMATECHART_1.setHighlightStartIndex(0);	Setzt den Kategorienindex value des Startwertes der Hervorhebung. Analog zu setHighlightEndIndex().
setHighlightStartSeries(Integer)	GRAPHOMATECHART_1.setHighlightStartSeries(1);	Setzt den Serienindex value der Startserie der Hervorhebung. Analog zu setHighlightEndSeries().
setLabelFormatMode(String)	GRAPHOMATECHART_1.setLabelFormatMode("extended");	Setzt als String, welcher Modus für die Label-Formatierung verwendet wird. Hierbei stehen basic und extended zur Verfügung.
setLabelPositioningMode(String)	GRAPHOMATECHART_1.setLabelPositioningMode("auto");	Diese Funktion setzt den Label Positioning Mode. Gültige Übergabewerte sind none (Labels werden ausgeblendet), fix (Labels werden direkt am dazugehörigen Element positioniert, können sich aber ggf. überschneiden) und auto (Labels überschneiden sich nicht, werden aber ggf. verschoben bzw. ausgeblendet).
setLockData(Boolean)	GRAPHOMATECHART_1.setLockData(true);	Setzt, ob die Komponente Daten-Updates ignorieren soll.
setManualCategoryLabels(String labels)	GRAPHOMATECHART_1.setManualCategoryLabels('[[["2001", "2001", "2001", "2001"], ["Q1", "Q2", "Q3", "Q4"]]]	Setzt die manuellen Kategorie Labels. Als Wert wird ein zweidimensionaler JSON-Array-String erwartet. Beispiele: [[["Q1", "Q2", "Q3", "Q4"], ["2017", "2017", "2017", "2017"]]] oder [[["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);	Setzt das Maximum der Skalierung des Base Charts auf den Integer-Wert max. Damit der Wert zum Tragen kommt, ist es erforderlich, dass der Skalierungsmodus auf Manual gesetzt ist (siehe auch setScaleModeManual() und unter Scaling der Eigenschaftenbeschreibung).
setManualScaleMaximumFloat(Float max)	GRAPHOMATECHART_1.setManualScaleMaximumFloat(1000000);	Setzt das Maximum der Skalierung des Base Charts auf den Float-Wert max. Damit der Wert zum Tragen kommt, ist es erforderlich, dass der Skalierungsmodus auf Manual gesetzt ist (siehe auch setScaleModeManual() und den Punkt Scaling der Eigenschaftenbeschreibung). Diese Funktion ist eine Alternative zu setManualScaleMaximum(), falls nur Float-Werte zur Verfügung stehen.
setManualScaleMinimum(Integer min)	die manuelle Skalierung bei 1.000.000 beginnen GRAPHOMATECHART_1.setManualScaleMinimum(1000000);	Setzt das Minimum der Skalierung des Base Charts auf den Integer-Wert min. Damit der Wert zum Tragen kommt, ist es erforderlich, dass der Skalierungsmodus auf Manual gesetzt ist (siehe auch setScaleModeManual() und den Punkt Scaling der Eigenschaftenbeschreibung).
setManualScaleMinimumFloat(Float min)	GRAPHOMATECHART_1.setManualScaleMinimumFloat(1000000);	Setzt das Minimum der Skalierung des Base Charts auf den Float-Wert min. Damit der Wert zum Tragen kommt, ist es erforderlich, dass der Skalierungsmodus auf Manual gesetzt ist (siehe auch setScaleModeManual() und den Punkt Scaling der Eigenschaftenbeschreibung). Diese Funktion ist eine Alternative zu setManualScaleMinimum(), falls nur Float-Werte zur Verfügung stehen.
setManualScalePercentMax(Float)	GRAPHOMATECHART_1.setManualScalePercentageMax(80);	Setzt das manuelle prozentuale Skalierungsmaximum als Float-Wert.
setManualScalePercentMin(Float)	GRAPHOMATECHART_1.setManualScalePercentageMin(-80);	Setzt das manuelle prozentuale Skalierungsminimum als Float-Wert.
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}");	Die MultiHighlights werden als JSON-String gesetzt, welcher ein Array enthält, in welchem für jedes Multihighlight ein Objekt enthalten ist. Das Objekt ist nach dem Schema {"startElement":1,"endElement":2,"startSeries":1,"endSeries":1} aufgebaut. Ein fertiger JSON-String kann dann beliebig viele Multi-Highlights enthalten.
setNegativeDeviationIsGood(Boolean)	GRAPHOMATECHART_1.setNegativeDeviationIsGood(true);	Setzt, ob die negativen Abweichungen als positiv, oder als negativ interpretiert werden sollen.
setNegativeOutlierThreshold(Float)	GRAPHOMATECHART_1.setNegativeOutlierThreshold(-80);	Setzt die negative Ausreißergrenze.
setNumberFormat(String)	GRAPHOMATECHART_1.setNumberFormat("0,0.0");	Setzt das Nummernformat für den Basic-Modus. Der Formatstring entspricht dem Format von numeral.js, einzusehen unter <a href="http://www.numeraljs.com">http://www.numeraljs.com</a> .
setNumberFormatPercentage(String)	GRAPHOMATECHART_1.setNumberFormatPercentage("0 %");	Setzt das Nummernformat für die Prozentwerte im Basic-Modus. Der Formatstring entspricht dem Format von numeral.js, einzusehen unter <a href="http://www.numeraljs.com">http://www.numeraljs.com</a> .
setPositiveOutlierThreshold(Float)	GRAPHOMATECHART_1.setPositiveOutlierThreshold(80);	Setzt die positive Ausreißergrenze.
setProperty(String, String)	GRAPHOMATECHART_1.setProperty("suppressZeroLabels", "true");	Setzt eine beliebige Property. Notwendig ist der technische Name für das Setzen der Werte der Property. Der technische wird in der linken unteren Ecke der Statusleiste im Lumira Designer angezeigt, nachdem die entsprechende Property im APS/SPS gewählt wurde. Der neue Wert für die Property muss als String formatiert sein, bspw. "Name", "2", "true", "#334455", "3.5" etc.
setRulesets(String newRulesetList);	GRAPHOMATECHART_1.setRulesets("Ruleset1,Ruleset2");	Wendet die gewünschten Rulesets auf ein Chart an. Erfordert eine gesetzte und valide Server URL.
setScaleModeAutomatic()	GRAPHOMATECHART_1.setScaleModeAutomatic();	Setzt den Skalierungsmodus des Base Charts auf Automatic.

setScaleModeManual()	GRAPHOMATECHART_1.setScaleModeManual();	Setzt den Skalierungsmodus des Base Charts auf Manual.
setScaleHelperValue(Integer)	GRAPHOMATECHART_1.setScaleHelperValue(500);	Setzt die Position des Skalierungshelfers. Erwartet die Position value als Integer-Wert.
setScaleHelperValueFloat(Float)	GRAPHOMATECHART_1.setScaleHelperValueFloat(500);	Setzt die Position des Skalierungshelfers. Erwartet die Position value als Float-Wert.
setSeparatorLength(Integer)	GRAPHOMATECHART_1.setSeparatorLength(5);	Setzt die Länge der Trennzeichen (Separators) in Pixeln.
setSeparatorMode(String)	GRAPHOMATECHART_1.setSeparatorMode("datatypes");	Setzt den Separator Mode. Gültige Werte sind: none, categorylabels, datatypes, manually.
setSeparatorPositions(String)	GRAPHOMATECHART_1.setSeparatorPositions("100,200,300");	Setzt die Trennzeichenpositionen des Charts. Erwartet value als kommaseparierten String.
setSeparatorSourceRowIndex(Integer)	GRAPHOMATECHART_1.setSeparatorSourceRowIndex(3);	Setzt den Index der Zeile/Serie, aus der sich die Trennzeichen (Separators) ableiten. Abhängig vom Separator-Mode (setSeperatorMode()).
setSeparatorThickness(Integer)	GRAPHOMATECHART_1.setSeparatorThickness(3);	Setzt die Dicke der Trennzeichen (Separators) in Pixeln.
setSeries(Integer seriesIndex, String)	die Datenbeschreibung der ersten Datenserie auf den übergebenen Wert setzen GRAPHOMATECHART_1.setSeries(1, '{"Keyfigures":"0D_NWI_NSAL"}');	Setzt die Datenbeschreibung der spezifizierten Datenserie seriesIndex. Erwartet value als JSON- String. Dieser String beschreibt entweder eine Zeile oder eine Spalte in der Initial View. Der String kann ausgelesen werden, nachdem man mit der graphomate charts Extension im Property Sheet Daten für eine Datenserie selektiert hat.
setSeriesLabels(String)	GRAPHOMATECHART_1.setSeriesLabels("Series1, Series2");	Setzt die Serienlabels des Charts. Erwartet value als kommaseparierten String.
setShowAxisLabelsLeft(Boolean)	GRAPHOMATECHART_1.setShowAxisLabelsLeft(true);	Setzt, ob die Achsenbeschriftungen auf der linken Seite angezeigt werden sollen.
setShowAxisLabelsRight(Boolean)	GRAPHOMATECHART_1.setShowAxisLabelsRight(true);	Setzt, ob die Achsenbeschriftungen auf der rechten Seite angezeigt werden sollen.
setShowCategoryLabels(Boolean)	GRAPHOMATECHART_1.setShowCategoryLabels(true);	Setzt, ob die Kategoriebeschriftungen (Category Labels) angezeigt werden sollen.
setShowDatatypesInAxis(Boolean)	GRAPHOMATECHART_1.setShowDatatypesInAxis(true);	Setzt, ob die Datenarten (Data Types) in der Achse angezeigt werden sollen.
setShowTitle(Boolean)	GRAPHOMATECHART_1.setShowTitle(true);	Setzt, ob der Titel angezeigt werden soll.
setShowTooltips(Boolean)	GRAPHOMATECHART_1.setShowTooltips(true);	Setzt, ob die Tooltips in der Runtime angezeigt werden.
setSpaceFactorBaseChart(Float)	GRAPHOMATECHART_1.setSpaceFactorBaseChart(0.8);	Setzt den Platz-Faktor, welchen das Base Chart anteilig (im Verhältnis zu den anderen Platz-Faktoren) einnimmt.
setSpaceFactorDeviationChart(Float)	GRAPHOMATECHART_1.setSpaceFactorDeviationChart(0.2);	Setzt den Platz-Faktor, welchen ein Abweichungschart anteilig (im Verhältnis zu den anderen Platz-Faktoren) einnimmt.
setStackedBarLabelPosition(String)	GRAPHOMATECHART_1.setStackedBarLabelPosition("middle");	Setzt die Position der Labels im Stapel. Gültige Werte sind middle und right.
setStackedBarNegativeValues(Boolean)	GRAPHOMATECHART_1.setStackedBarNegativeValues(true);	Setzt, ob negative Stapel erlaubt sein sollen.
setStackedBarSumsVisible(Boolean)	GRAPHOMATECHART_1.setStackedBarSumsVisible(true);	Setzt, ob die Summen auf den Stapelbalken angezeigt werden sollen.
setStructureMode(Boolean)	GRAPHOMATECHART_1.setStructureMode(true);	Mit dieser Funktion kann ein Diagramm in den Structure Mode (true) bzw. den Time Mode (false) gesetzt werden.
setSuppressRepeatingCategoryLabels(String)	GRAPHOMATECHART_1.setSuppressZeroLabels("true,false,false,true");	Diese Funktion setzt die Einstellung für die Eigenschaft Suppress Repeating Category Labels. Es wird pro benötigter Zeile eine Angabe benötigt, die besagt, ob wiederholende Kategoriebeschriftungen ausgeblendet werden sollen.
setSuppressZeroLabels(Boolean)	GRAPHOMATECHART_1.setSuppressZeroLabels(true);	Diese Funktion setzt, ob Null-Labels angezeigt werden sollen.
setSynchronizeAxesPosition(Boolean)	GRAPHOMATECHART_1.setSynchronizeAxesPosition(true);	Diese Funktion setzt, ob die Achsenpositionen des Charts mit der Comparison Group abgeglichen werden sollen.
setSynchronizePadding(Boolean)	GRAPHOMATECHART_1.setSynchronizePadding(true);	Diese Funktion setzt, ob die Ränder des Charts mit der Comparison Group abgeglichen werden sollen.
setSynchronizeScaling(Boolean)	GRAPHOMATECHART_1.setSynchronizeScaling(true);	Diese Funktion setzt, ob die Skalierung des Charts mit der Comparison Group abgeglichen werden soll.
setThresholdLinesConfig(String)	GRAPHOMATECHART_1.setThresholdLinesConfig('{"visible":true,"label":"Mean","type":"Average","base":1,"lineSize":1,"color":"#0000FF","lineStyle":"solid"}')	Diese Funktion ersetzt die bisherige Threshold Lines Configuration durch das übergebene JSON. Für ein Beispiel eines Config-JSONs, können Sie dieses aus dem Standard Property Sheet kopieren. In dem hier gezeigten Beispiel wird der Mittelwert für die Serie1 mit Label "Mean" angezeigt.
setTitle(String)	den Titel auf den übergebenen String setzen GRAPHOMATECHART_1.setTitle("Sales in Mio. EUR 2013 ACT and BUD North West");	Setzt den Titel des Charts. Erwartet value als String. Zeilenumbrüche (\n) werden auch als solche interpretiert.
setUseOutlierThreshold(Boolean)	GRAPHOMATECHART_1.setUseOutlierThreshold(true);	Diese Funktion setzt, ob die Ausreißergrenze verwendet werden soll.
setUserHighlightColor(String)	GRAPHOMATECHART_1.setUserHighlightColor("#ff0000,#00ff00,#0000ff");	Setzt eine Liste der hinterlegten User Highlight-Farben. Erwartet einen kommaseparierten String. Die gesetzten Farben werden dabei in der angegebenen Reihenfolge bei den User Highlights (zur Laufzeit hervorgehobene Werte) durchgeschaltet, bis die Wunschfarbe eingestellt ist.
setUserHighlightList(String)	GRAPHOMATECHART_1.setUserHighlightList("[[1,3],[2,3][1,5]]");	Setzt ein oder mehrere User Highlights (zur Laufzeit hervorgehobene Werte). Erwartet einen JSON-String bestehend aus einem oder mehreren Arrays mit jeweils zwei Zahlen. Dabei gibt die erste Zahl die Datenreihe an, während die zweite Zahl die Position des Wertes in der Datenreihe darstellt.
setValueAxisVisible(Boolean)	GRAPHOMATECHART_1.setValueAxisVisible(true);	Diese Funktion schaltet die Anzeige der Value Axis an bzw. aus.
setVisibleElementLabels(String)	GRAPHOMATECHART_1.setVisibleElementLabels("1,2,4");	Setzt die Visible Element Labels. Erwartet eine Liste von Indices aus kommaseparierten Werten, z.B. 1,2,3,4,5,6,7, 8,9,10
setVisibleSeries(String)	nur die Datenserien mit den Indices 1 und 2 anzeigen GRAPHOMATECHART_1.setVisibleSeries("[_href#{1,2}]");	Setzt die Auflistung der Indices sichtbarer Serien des Charts. Erwartet value als kommaseparierten String.
showStructureMode (Boolean)	GRAPHOMATECHART_1.showStructureMode(true);	Setzt die Ausrichtung des Charts. Wenn der Wert value TRUE ist, wird das Chart im Strukturmodus angezeigt, sonst im Zeitmodus.

## Beispiele

### Mit Hilfe einer Dropdown-Box eine Datasource dynamisch Filtern

Um auf einer Dimension einer Data Source dynamisch zu filtern, muss zuerst eine Dropdown-Box mit allen zulässigen Werten befüllt werden, damit der User aus dieser später dann seine Wahl treffen kann.

Dafür wird auf dem Canvas-Event On Startup von der gewünschten Data Source mit getMemberList eine Liste aller Member der Dimension OD\_NWI\_ACOD geholt. Die Parameter von getMemberList werden dabei vom Autocomplete vorgeschlagen.

Die so erlangte Liste kann dann per setItem auf die Dropdown-Box gesetzt werden.

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

Nun wird die Dropdown-Box beim Applikations-Start mit den Mitgliedern der Dimension 0D\_NWI\_ACOD befüllt. Selektiert nun ein User ein Element aus der Liste, kann mit folgendem Skript, welches auf das Ereignis On Element Clicked der Dropdown-Box gelegt wird, der Filter der Data Source angepasst werden:

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

### Die Filter einer Data Source entfernen

Sollen die Filter einer Data Source wieder entfernt werden, kann dies mit folgendem Aufruf getan werden:

```
DS_1.clearAllFilters();
```

### Per Drilldown auf einem graphomate Chart eine Data Source filtern

Soll ein Drilldown mit den graphomate charts realisiert werden, muss vorher eine Bedingung erfüllt sein: Es muss die Property categoryLabels befüllt sein! Die Kategorienbeschriftungen müssen dabei nicht sichtbar sein, können dies aber natürlich. Das ist notwendig, weil das Chart nur über diese Property Informationen über Dimensionen und Members erhalten kann. Wenn diese Vorbedingung erfüllt ist, kann auf dem Event On Element Clicked des graphomate Charts mit folgendem Skript der Schlüssel (Key) des selektierten Dimensionsmember erhalten werden:

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

Als Parameter muss der Schlüssel (Key) der gewünschten Dimension übergeben werden.

Dieser so erlangte Memberkey kann nun - analog wie bei dem Beispiel der Dropdown-Box - als Filter auf eine Data Source gesetzt werden:

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

bzw. zusammengezogen:

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

### Titel nach Drilldown setzen

Um auch den Titel des Charts an den Drilldown anzupassen – zum Beispiel um die selektierte Region anzuzeigen - kann parallel zum Memberkey auch der Membertext ausgelesen werden.

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

Dieser Text kann dann natürlich beliebig weiter verwendet werden oder wie hier in den Titel hinein geschrieben werden:

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

### Zooming-Funktion

Um eine Zoom-Funktionalität für ein Chart umzusetzen, muss der Skalierungsmodus des Charts auf Manual stehen. Dies kann entweder voreingestellt oder über die Skriptfunktion realisiert werden:

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



Danach wird das onElementClicked-Event des entsprechenden Charts mit folgendem Skript versehen:

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

Zunächst wird dabei der relative Wert der Klickposition ermittelt. In der zweiten Zeile wird dieser Wert dann als Minimum für die manuelle Skalierung eingesetzt und somit ein Zoom erreicht.

## 1.4 Scripting

Die graphomate charts Extension unterstützt die Skriptsprache von Designer umfassend. Somit ist es möglich graphomate charts zur Laufzeit interaktiv zu steuern.

Beispielsweise können neue Daten gesetzt, Titel bearbeitet und Skalierungen verändert werden. Weiterhin ist es möglich, Drilldown-Funktionalitäten der Extension zu verwenden und interaktive Dashboards zu erstellen.

Beispiel: Mit Hilfe einer Dropdown-Box eine Data Source dynamisch Filtern

Um auf einer Dimension einer Data Source dynamisch zu filtern, muss zuerst eine Dropdown-Box mit allen zulässigen Werten befüllt werden, damit der User aus dieser später dann seine Wahl treffen kann.

Dafür wird auf dem Canvas-Event On Startup von der gewünschten Data Source mit getMemberList eine Liste aller Member der Dimension 0D\_NWI\_ACOD geholt. Die Parameter von getMemberList werden dabei vom Autocomplete vorgeschlagen.

Die so erlangte List kann dann per setItem auf die Dropdown-Box gesetzt werden.

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

Nun wird die Dropdown-Box beim Applikations-Start mit den Mitgliedern der Dimension 0D\_NWI\_ACOD befüllt.

Selektiert nun ein User ein Element aus der Liste, kann mit folgendem Skript, welches auf das Ereignis On Element Clicked der Dropdown-Box gelegt wird, der Filter der Data Source angepasst werden:

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

Beispiel: User Highlights mittels Bookmarks speichern

Zur Laufzeit lassen sich sogenannte User Highlights um ausgewählte Werte setzen (siehe [Interaktivität zur Laufzeit](#)). Um diesen Zustand zu speichern, bedarf es zweier Schaltflächen:

Um die ID des Bookmarks zwischen zu speichern, bedienen wir uns der Einfachheit halber eines Textfeldes (im Beispiel TEXT\_2), in das die ID geschrieben wird. Diesen Schritt kann man mittels Variablen auch visuell ausblenden.

Der für das Speichern zuständige Button erhält folgendes Skript, um in die Variable id den Bookmark-Zustand zu speichern. Die Variable wird zur Sicherung dann in ein Textfeld geschrieben:

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

Der für das Laden zuständige Button erhält folgendes Skript und liest damit die ID aus dem Textfeld aus und stellt diesen Bookmark-Zustand wieder her:

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

Weitere Erläuterungen und Beispiele zur Skriptsprache finden Sie in unserer [Scripting-Dokumentation](#).