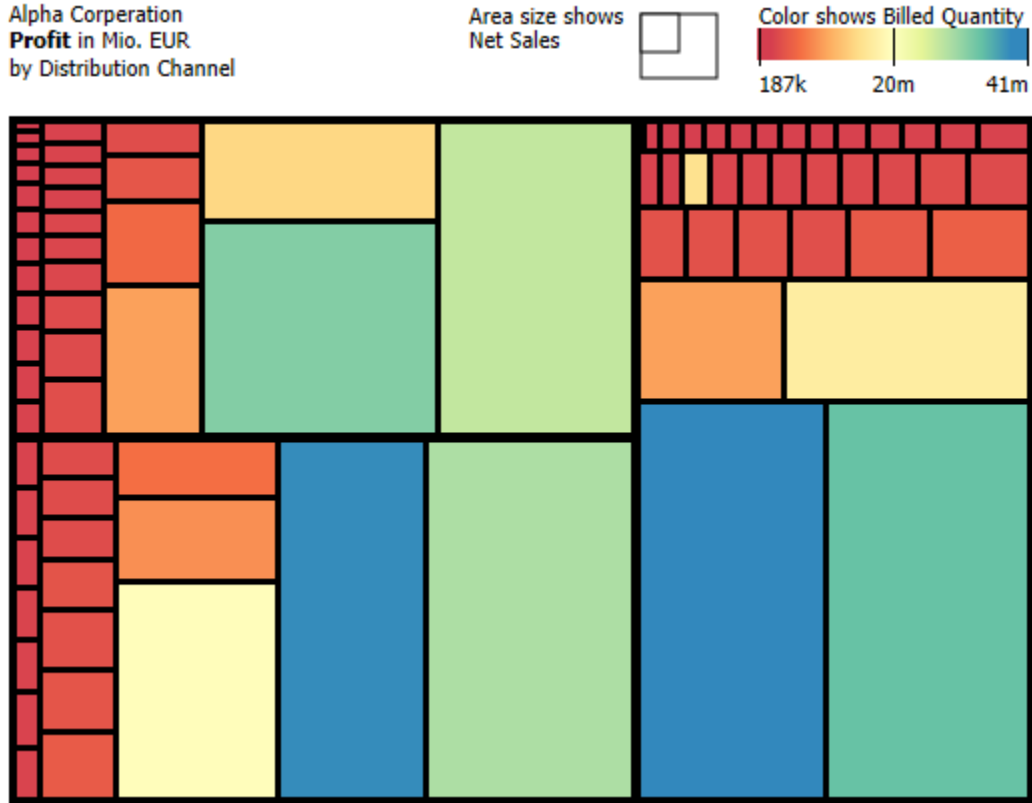


User manual for the graphomate treemaps for SAP Lumira Designer



Version 2020.4 – as of October 2020

<https://www.graphomate.com>



Table of contents

- Introduction
- Installation
- Quick Start
- Properties
- Standard Properties
- Known Issues
- Number Formatting

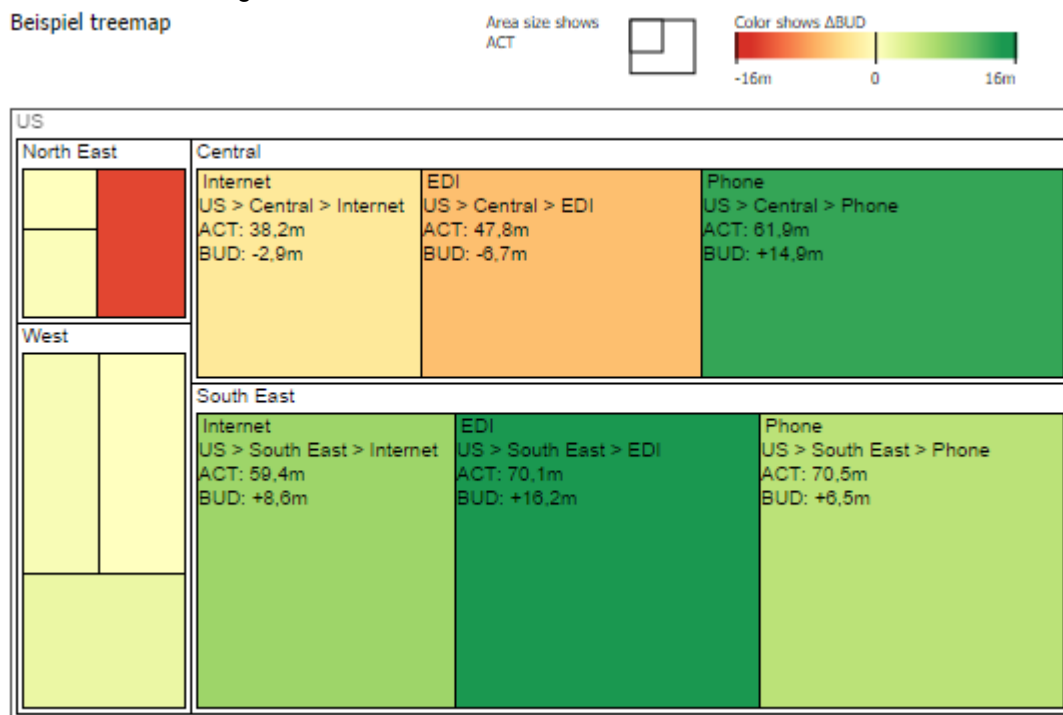
Introduction

Treemaps – hierarchies with areabased visualizations

The treemap developed from a specific problem. Ben Shneiderman therefore searched a way to display a large number of hierarchical data in the smallest possible space, without wasting space.

The result was the elimination of the common tree structure with nodes and edges. Instead the treemap visualizes hierarchical structures as a set of nested rectangles. Thus ratios in size can be visualized clearly, because the area of the rectangles is proportional to the size of the data unit to be displayed.

The belonging to a certain hierarchy level or a second dimension (eg. the deviation to a previous period) can be shown via the color of the rectangles.



That way treemaps can give a high level view of the data while also showing you the line item details. Your eye visually aggregates rectangles in the same group, allowing you to see patterns quickly. Therefore our *graphomate treemaps* are very suitable as an entry for an interactive dashboard (quick overview). But in order to show the exact values and precise comparisons, other visualization elements have to follow in the navigation path, eg. components such as our *graphomate charts* or *graphomate tables*.

In the sense of a uniform language regulation we speak in the further manual of Lumira Designer or briefly Designer.

For our graphomate treemaps extension Lumira Designer >= 2.0 SP03 and IE 11+ are required. For further information, for example about the deployment on the Business Intelligence Platform (BIP) or SAP Netweaver, please consult the Product Availability Matrix.

Please note that we will no longer support Internet Explorer 9 and 10 from 2017.

Installation

- Before opening your old dashboards/BI apps with Designer 2.0, you need to migrate them to m mode with Design Studio 1.6.
- Our extensions for Design Studio 1.6 will also run in Designer 2.0. When starting Lumira 2.0 Designer, please do not import our graphomate extension from SAP Design Studio 1.6. Instead we suggest using the versions specifically built for Designer 2.0.

Local installation of the extension to Designer

You have installed Designer 2.0 SP03+ on your computer.

1. Save the zipfile *graphomate_treemaps_2020.4.x_LumiraDesigner.zip* to a folder of your choice.
2. In Designer, choose *Tools Install Extension to Designer*, click on *Archive* and select the zipfile which has been saved before.
3. Choose *Finish* to start the installation.
4. Choose *Next* and again *Next* to confirm the installation.
5. Accept the terms of the license agreement and choose *Finish*.
6. Choose *Yes* to allow a restart of Designer.
7. After the restart, the graphomate extension appears in the Component View of Designer.

Removing the extension from Designer

In Designer choose *Help > About...*

1. Click the *Installation Details* button.
2. Select the component *graphomate treemaps 2020.4.x*
3. Choose *Uninstall...*
4. In the *UninstallWizard* choose *Finish*.
5. Choose *Yes* to allow Designer to restart.

Server installation of the extension

Users need to deploy the locally installed extension to the BI platform before launching the *graphomate treemaps* from the BI platform.



1. Choose in the BI Platform Mode *Tools > Platform Extensions*.
2. Select the extension *graphomate treemaps* that is installed on your local computer.
3. Choose *Install on Platform*.
4. Manually restart the BI Platform Adaptive Processing Servers that host the Analysis Application Service and accept the warnings in Designer.
5. The graphomate extension appears under *Extensions Installed on Platform*.
6. Choose *Close*.
7. Again restart the BI Platform Adaptive Processing Servers that host the Analysis Application manually.

Uninstalling the extension from the Server

1. Choose in the BI Platform Mode *Tools > Platform Extensions*.
2. Choose the *graphomate treemaps*.
3. Choose *Uninstall from Platform*.
4. Confirm the uninstalling by choosing *Yes*.
5. In order to complete the uninstalling process, manually restart the BI Platform Adaptive Processing Servers that host the Analysis Application Service and accept the warnings in Designer.
6. The *graphomate treemaps* is then removed from the *Extensions Installed on Platform*.
7. Choose *Close*.
8. Manually restart the BI Platform Adaptive Processing Servers that host the Analysis Application Service.

Quick Start

You have defined a *Data Source* on basis of a SAP BW Query and now want to depict data series from this query with *graphomate treemaps*.

1. Drag the component *graphomate treemaps* to the canvas and bind the *Data Source* to this component via Drag&Drop.
2. On the *Data tab* select the Data Series from the Query in the Initial view – in rows or columns – which you want to show by clicking the corresponding button . Just click on the row or column header. The selected Data Series is highlighted in blue and after clicking  it will be shown in the *treemap*.
3. The other UI-elements in the *Additional Properties Sheet* give you the opportunity to control the appearance of the *treemaps*.

Properties

Basically, there are two possibilities to maintain the features of *graphomate treemaps*: with the *Standard Properties Sheet*, which lists all parameters in a structured list or with the *Additional Properties Sheet*. The latter provides a user-friendly interface. There are some special features, which can only be maintained on the Standard Properties Sheet.

The Checkboxes next to the features in the *Additional Properties Sheet* are used to control the visibility of chart elements. The chosen *Data Source* can only be attributed to the component via Drag&Drop or via the *Standard Properties Sheet* under *Data Binding*.

Clicking the **F5** key (Reload Application) reinitializes the canvas as well as the *Additional Properties Sheet*. This is sometimes necessary when changes in the properties are not directly accepted in the canvas.

Data Tab

DATA SELECTION

For the *graphomate treemaps* two *Data Series* with different semantics can be selected:

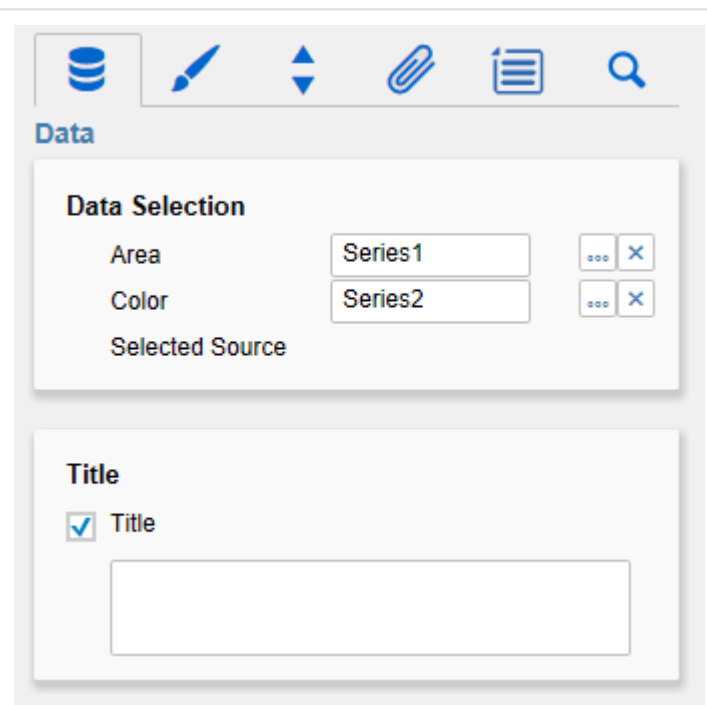
- **Area Series:**
The size of the treemap rectangles is drawn in relation and proportional to the values of the *Area Series*.
- **Color Series:**
The coloring of the treemap is done with the *Color Series*. Either the values of the data series are mapped to a color scale directly or the deviation to the *Area Series* is calculated first and mapped afterwards. The color scheme can be chosen on the *Appearance* tab and the calculation method (direct mapping or deviation) can be set on the *Behavior* tab.

In order to assign the data series, a data source must be selected first. The series labels can be customized via the input box and will be displayed in the legend, descriptions and tooltips.

Click on "x" to delete the assignment of the *Data Series*.

TITLE

Enter a title for your treemap at this point. If the Checkbox is activated, the *Title* will be shown. Line breaks are adopted.



Appearance Tab

LABELS

Sets the appearance of the font.

FONT SIZE

Sets the font size for the rectangle labels.

TEXT COLOR

Sets the text color for the rectangle labels.

VALUE FORMAT

Select the format of the treemap values:

- Basic:**
 Select the format of the absolute and percentage values according to guidelines of numeral.js
- Extended:**
 If a fixed format is preferred, put the format for absolute and percentage values here. Choosing a "(" in the box *Negative Sign* will show negative values in brackets: (1234). The parameter in the box *Scaling* is used as a divisor for scaling the data label. The resulting format string can also be set via scripting language.

Set the property *Use Formatted Data* in the *Standard Properties Sheet* on *true* to use the formatting of the Data Source.

🗄️
✍️
⬆️
📎
📄
🔍

Appearance

Labels

Font Size px

Text Color

Value Format

Basic
Extended

Percentage Format

Absolute Format

Locale ▼

Basic Format according to numeral.js

Value Format

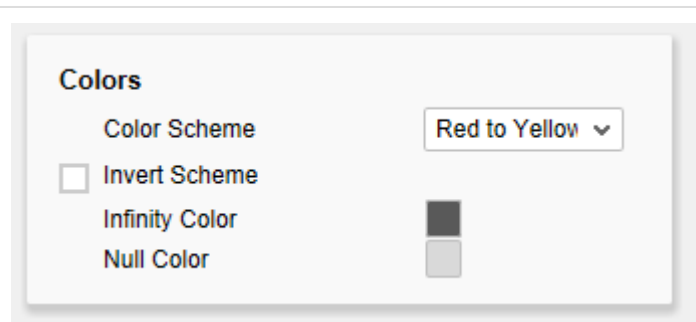
Basic
Extended

	abs	%
Decimal Separator	, ▼	, ▼
Thousands Separator	. ▼	. ▼
Negative Sign	- ▼	- ▼
Scaling	1 ▼	1 ▼
Decimal Places	1 ▼	1 ▼
Prefix	<input style="width: 100%;" type="text"/>	<input style="width: 100%;" type="text"/>
Suffix	<input style="width: 100%;" type="text"/>	%

COLORS

At this point, different settings for coloring the treemap can be made:

- **Scheme:**
Change the color scheme for the treemap here. The default setting is a gradient from red to yellow to green. Especially suitable for showing hierarchies are the color scales *Hierarchy1*, *2* and *3*.
- **Invert Scheme:**
If the Checkbox is activated, the color scheme is reversed (eg. useful if negative values are shown as positive (or vice versa))
- **Infinity:**
The color selected here will be displayed when the deviation calculation returns an infinite value (eg. division by zero).
- **Null:**
The color set here is displayed when no value can be derived from the color series (eg. if the color calculating is based on the series, but no series was assigned).



Behavior Tab

COLOR CALCULATION BASED ON

The *Color Calculation Method* determines in which way the values of the *Color Series* will be calculated for coloring the treemap. The following options are possible:

- *<name of the Color Series>*: Choose this method to depict the values of the *Color Series* linearly on the color scale (ie, the largest value is on the right end of the scale, the lowest on the left).
- *Absolute*: Choose this method to calculate the absolute deviation between *Area* und *Color Series*; the result is depicted linearly on the color scale.
- *Hierarchy*: Choose this method to color the treemap according to its hierarchical structure.

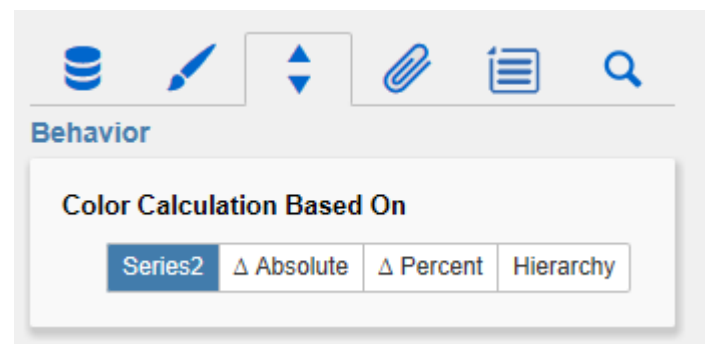
HIERARCHY

The different settings here determine how the hierarchy of the treemap is displayed:

Embedded

Embedded treemaps show the hierarchy by additional frames in order to display the hierarchy level explicitly. If the *Checkbox* is activated, the frames are displayed and there are the following additional options:

- *Root Node Description*: The *Root Node* is the root of the hierarchy, the node that is displayed as the outer frame in the *embedded* mode. Often this node is simply labeled as totals in the data source. At this point you can rename the label of the outer frame.
- *Top Padding*: Set the width of the spacing between the top side of the frames and the treemap in the *embedded* mode at this point.
- *Rest Padding*: Set the width of the spacing between the frames and the treemap (for the left, right and bottom side) in the *embedded* mode at this point.

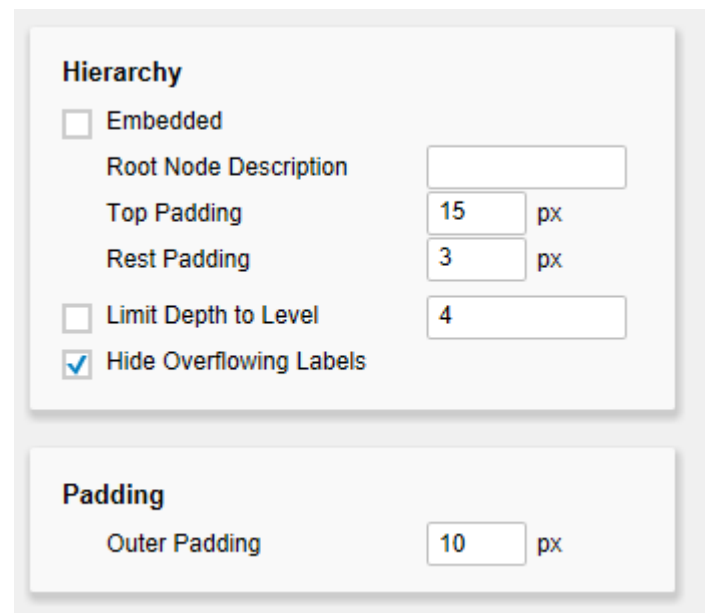


Formula for absolute deviation calculation:

$$(Area - Color)$$

Formula for percentual deviation calculation:

$$\left(\left(\frac{(Area - Color)}{abs(Color)} \right) * 100 \right)$$



Is the *Top Padding* smaller than the set font size, the description in the frame is automatically hidden.

If you want just thin frames without labels in order to make the hierarchy more legible in efficient space, you have to set both *Padding* values to *1*.

Limit Depth

If the Checkbox is activated, the hierarchy shown is limited to a certain depth (entry in field *Level*). That way the treemap shows only the hierarchy from the root up to the specified depth, all other levels of the hierarchy are cut off.

HIDE OVERFLOWING DESCRIPTIONS

If the Checkbox is activated, all rectangle labels, that would extend horizontally beyond the rectangle width, are hidden. Thus the treemap appears visually balanced and the focus is on the major elements.

OUTER PADDING

While changing the *Outer Padding* value the distance between the chart and the surrounding frame in Designer can be set.

Helper Tab

LEGEND

Here you can change the settings for the legend. In addition to the visibility (activating the Checkbox) you have the following options:

- *Area Descriptor:*
The labeling of the legend is generated from the metadata of the corresponding data series. If you want to change the labeling of *Area Series*, you can do it here.
- *Color Descriptor:*
Customize the labeling of the Color Series in the legend here.

DESCRIPTION CONFIGURATION

Here you can change the rectangle labels. The labels are showing information about the corresponding *dimension member*. In addition to the visibility (activating the Checkbox) you have the following options:

- *Path:*
The path shows the higher-order nodes of the hierarchy, leading to the current *dimension member*.
- *Name:*
This option displays the name of the dimension member.
- *Area Value:*
This option displays the name and the value for the *Area Series*.
- *Color Value:*
This option displays the name and the value for the *Color Series* (or the calculated deviation tot he *Area Series*).

TOOLTIP CONFIGURATION

The settings in the *Tooltip Configuration* are analogous to those in the *Description Configuration* , but apply to the tooltips shown at runtime in the treemap (thus the effect of the settings in the *Tooltip Configuration* is not visible at design time). Possible settings are also these: *Path*, *Name*, *Area Value* and *Color Value*.

The screenshot shows the 'Helper' configuration window. At the top is a toolbar with icons for database, edit, zoom, save, list, and search. Below the toolbar is the 'Helper' title. The main content area is divided into three sections:

- Legend:** Contains a checked checkbox for 'Legend'. Below it are two input fields: 'Area Descriptor' and 'Color Descriptor'.
- Description Configuration:** Contains a checked checkbox for 'Descriptions'. Below it are four unchecked checkboxes: 'Path', 'Name', 'Area Value', and 'Color Value'.
- Tooltip Configuration:** Contains a checked checkbox for 'Tooltips'. Below it are four unchecked checkboxes: 'Path', 'Name', 'Area Value', and 'Color Value'.

Context Menu Tab

In this tab you will be able to choose which options are available in the context menu. An activated checkbox means this options will be shown. The first checkbox enables or disables the context menu altogether. Once activated the context menu will be accessible during runtime by right clicking on the component. The active operations will be usable as follows.

SORTING

With *Sorting* data from the available dimensions can be sorted in ascending or descending order.

FILTERS

Allows you to filter the data by member for all available dimensions.

CONTEXT SENSITIVE FILTERS

Shows filters corresponding to the currently clicked dimension.

FILTER AND HIDE

In addition to the normal filtering this operation also removes the corresponding dimension from our initial view.

CLEAR FILTERS

Resets previously set filters.

DRILLDOWN

Adds another dimension to the component.

DRILLUP

Removes a dimension from the chart (this can cause an error which makes a reload necessary).

ZERO DISPLAY

Sets the way zero values are displayed.

MEMBER DISPLAY

Lets you choose the way member names are displayed (text or key).

Context Menu

Settings

- Context Menu

Features

- Sorting
- Filters
- Context Sensitive Filters
- Filter and Hide
- Clear Filters
- Drilldown
- Drillup
- Zero Display
- MemberDisplay
- Position for Totals
- Total Display

Custom Event Names

- customEvent1
- customEvent2
- [▶ more](#)

A reload resets all changes made with the context menu.

POSITION FOR TOTALS

Sets the positions for each total value per dimension to be displayed before or after other values.

TOTAL DISPLAY

Enable or disable totals for each dimension.

CUSTOM EVENTS

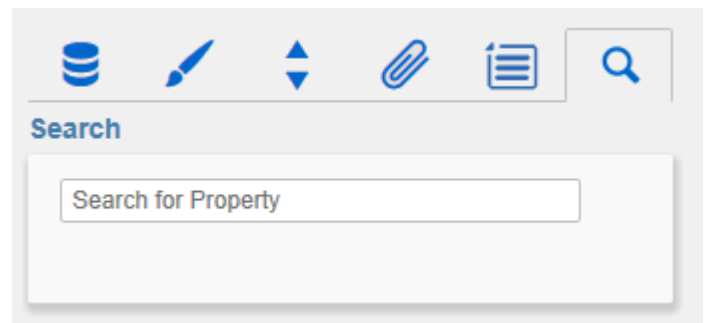
There are 10 possible events which can be linked to your own scripts. These can be defined in the property sheet and triggered using the corresponding button in the context menu.

In the *Additional Properties Sheet* the names for these buttons is customizable.

All operations are executed directly on the data source, but without changing it.

Search Tab

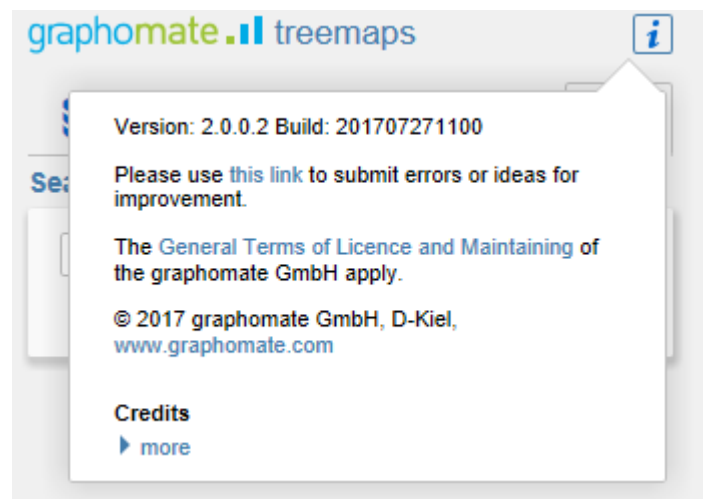
At this point you can search for the name, function or group of a property which are displayed at the *Additional Properties Sheet*. The appropriate settings are then displayed directly and can be changed here.



Info Popup

Here you can find the version number of our *graphomate treemaps* as well as a link where you may report bugs or propose wishes for new features.

A list of all open-source libraries that we used can be found under *Credits*.



Standard Properties

Some (expert) features can only be controlled in the *Standard Properties Sheet*.

AREA IDENTITY

Is an option of the *Color Calculation Method*. If this option is activated, the Area Series is mapped linearly to the color scale; this is useful in connection with the option *Show Area Absolute Values*.

ASCENDING SORT ORDER

Set this option to *true* in order to arrange the elements of the treemap in ascending order (largest elements at the bottom right). The default value is *false*, meaning values are displayed in descending order (largest elements at the top left), corresponding to the natural reading order as the main elements are shown first.

HIDE TREEMAP

If this option is set to *true*, only the title (incl. legend) is displayed. The treemap itself is not drawn.

SCHEMA

In addition to the color scales available on the *Additional Properties Sheet*, all scales of the ColorBrewers can be used. In order to do this, the name of the *ColorBrewer schema* together with the number of classes have to be entered separated by a comma (example: *Spectral, 4*).

SHOW AREA ABSOLUTE VALUES

If this option is set to *true*, the absolute values of the *Area Series* are taken to draw the treemap, ie negative values are represented as rectangle too.

USE FORMATTED DATA

This property allows using the preformatted data of the *Data Source* for cells of the treemap. This is possible only for those values that are passed from the *Data Source*. Values, which are calculated in the *graphomate treemaps* itself – as the deviations for *Color Calculation Based On Absolute* and *Percent* – are still formatted either in *basic* or *extended* number format.

CUSTOM COLOUR RANGES

This Property enables you to manually define colors for nodes whose color values are in a specific range. The color ranges will be applied, if the *Color Calculation Methode* is set to "custom". A valid property value is an array of strings, which consist of one or two conditions and one color, separated by "|".

Example: " $\geq -0.25 | \leq 0.25 | \#FFFFBF$ " - Sets the color of all nodes whose color series value lies between - .25 and .25 to Yellow

Another example is to use the colors red, yellow and green to mimic a traffic light.

Valid operators are: $>$, \geq , $<$, \leq . Valid Colors are 6-digit hex strings

COLOR SCALE BOUNDARIES

This property determines the boundaries of the color scale.

If the value "symmetrical" is selected, the scale is always created symmetrically. So if the data have a minimum value of -500 and a maximum value of 100, the color scale will range from -500 to +500. The limits are thus formed from the largest amount of the domain values. Even with exclusively positive values, the scale is mirrored at the zero crossing so that nodes with positive values doesn't get displayed in negative colors such as red.

If the value "real" was selected, the limits of the color scale correspond exactly to the domain. For a minimum value of -500 and a maximum value of 100, the color scale will range from -500 to 100.

THRESHOLD

This property sets the positive and the negative threshold of the color scale. It can be used to narrow the color scale to better identify smaller changes by excluding outliers from the color spectrum.

APPLY PERCENTAGE FORMAT TO AREA

If selected, the values of series 1 are scaled as percentage values. Note that a percentage format will not only show the "%" suffix, but also multiply the value by 100. The Scaling factor of an extended number format can be used to avoid this.

APPLY PERCENTAGE FORMAT TO COLOR

If selected, the values of series 2 are scaled as percentage values. Note that a percentage format will not only show the "%" suffix, but also multiply the value by 100. The Scaling factor of an extended number format can be used to avoid this.

Known Issues

- Data selections may only cover entire columns in order to be processed correctly by the component. Restricting the selection to certain row members is not supported.
- Structures or explicit hierarchies are supported, but only if they are the only dimensions of the Rows axis. The combination of structures or hierarchies with other dimensions is not supported.

Number Formatting

Possible Inputs for the numeral.js Format String

Floating point		
Number	Format String	Output
10000	'0,0.0000'	10.000,0000
10000.23	'0,0'	10
-10000	'0,0.0'	-10.000,0
-0.23	'.00'	-,23
-0.23	'(.00)'	(,23)
0.23	'0.00000'	0,23000
0.23	'0.0[0000]'	0,23
1230974	'0.0a'	1,2m
1460	'0 a'	1 k
1	'0o'	1 st
Currency		
Number	Format String	Output
1.000.234	'\$0,0.00'	\$1.000,23
1000.2	0,0[.]00 \$'	1.000,20 \$
1001	'\$ 0,0[.]00'	\$ 1.001
Percentage		
Number	Format String	Output
1	'0%'	100%
-0.43	'0 %'	-43%

Source

Possible input for the Extended Number format string

Percentages and numbers are formatted separately. For this two format string are used, which follow the same rules and structure. In general the strings of characters in the format string are simply replaced in the input number. Exceptions to this rule would be scaling factors and the number of decimal places.

The separate elements of the format string are separated by the "|" character. Format strings with less than 7 elements (i.e. 6 "|" characters) are considered invalid and therefore ignored. In this case the standard format without scaling and with one position after the decimal point is used.

The format string consists of the following elements, the ordering is obligatory and cannot be changed:
Negative|Prefix|Thousands|Decimal|Scaling|PointsAfterDecimal|Suffix

Negative	The prefix for negative numbers. The string "(" will result in the number will be placed inside parentheses, including eventual prefixes and suffixes. If <i>None</i> is used the absolute number will be shown.
Prefix	A prefixed string of characters, for example a currency symbol or an identifier for the scaling factor.
Thousands	A separator for thousands.
Decimal	A separator for the decimal places.
Scaling	The scaling factor for the value. This can be given as a whole number or a power, formatted as n^m . A whole number will be calculated as a power to the basis 10 internally. In case of a given power the value will be divided by it directly. Even negative numbers are possible, for example "-3" or "10^-3"; in this case the value will get adequately higher. Non numeric inputs are ignored. <i>Please note:</i> given a "0" or a "1" will result in a scaling factor of 1. The only way to achieve a scaling by 10 is to input "10^1".
PointsAfter Decimal	The number of places after the decimal.
Suffix	A suffix for the value, analogous to the prefix.

