dashBIRD

dashBIRD Berichtsentwicklung

Dokumentversion vom 06. 11. 2023

Inhalt

1 Einführung TRADUI Dashlet-Entwicklung 48
1.1 Was sind eigentlich Dashlets 48
1.2 Warum Dashlets? 48
1.3 BIRT-Plugin - Datenversorgung und Business Intelligence 48
1.4 Dashboards / dashBIRD - Eigenentwicklung oder fertige Lösung 48
2 C3-Charts 49
2.1 Einleitung 49
2.2 Ein C3-Chart-Dashlet erzeugen 49
2.3 Daten ans Dashlet übergeben 49
2.4 Hinzufügen von Optionen 49
2.5 Chart-Typen 49
2.5.1 Line Chart 49
2.5.2 Timeseries Chart 50
2.5.3 Spline Chart 50
2.5.4 Simple XY Line Chart 50
2.5.5 Multiple XY Line Chart 51
2.5.6 Line Chart with Regions 51
2.5.7 Step Chart 51
2.5.8 Area Step Chart 52
2.5.9 Area Chart 52
2.5.10 Area Spline Chart 52
2.5.11 Stacked Area Chart 53
2.5.12 Stacked Area Spline Chart 53
2.5.13 Bar Chart 54
2.5.14 Stacked Bar Chart 54
2.5.15 Scatter Plot 55
2.5.16 Pie Chart 55
2.5.17 Donut Chart 56
2.5.18 Gauge Chart 56
2.5.19 Combination Chart 57
2.6 Axis 57
2.6.1 Category Axis 57

2.6.2 Rotated Axis ... 57
2.6.3 Additional Y Axis ... 58
2.6.4 X Axis Tick Format ... 58
2.6.5 X Axis Tick Count ... 58
2.6.6 X, Y, Y2 Axis Tick Values ... 58
2.6.7 X Axis Tick Culling ... 59
2.6.8 X Axis Tick Fitting ... 59
2.6.9 X Axis Timezone ... 59
2.6.10 Rotate X Axis Tick Text ... 59
2.6.11 Y Axis Tick Format ... 59
2.6.12 Y Axis Tick Count ... 59
2.6.13 Y2 Axis Tick Format ... 59
2.6.14 Y2 Axis Tick Count ... 60
2.6.15 Padding for X Axis ... 60
2.6.16 Padding for Y Axis ... 60
2.6.17 Range for Y and Y2 Axis ... 60
2.6.18 Axis Label ... 60
2.6.19 X Axis Label Position ... 60
2.6.20 Y and Y2 Axis Label Position ... 61
2.7 # Data ... 61
2.7.1 Column Oriented Data ... 61
2.7.2 Row Oriented Data ... 61
2.7.3 JSON Data ... 61
2.7.4 Data from URL ... 62
2.7.5 Category Data ... 62
2.7.6 Load Data ... 62
2.7.7 Data Name ... 62
2.7.8 Data Color ... 62
2.7.9 Data Order ... 62
2.7.10 Data Label ... 62
2.7.11 Data Label Format ... 62
2.8 # Grid ... 63
2.8.1 Grid Lines ... 63
2.8.2 Optional X Grid Lines ... 63
2.8.3 Optional Y Grid Lines ... 63

2.9 # Region 64
2.9.1 Region 64
2.9.2 Region with Timeseries 64
2.10 # Interaction 64
2.10.1 Sub Chart 64
2.10.2 Zoom 64
2.11 # Legend 64
2.11.1 Hide Legend 64
2.11.2 Legend Position 65
2.11.3 Custom Legend 65
2.12 # Tooltip 65
2.12.1 Hide Tooltip 65
2.12.2 Tooltip Grouping 65
2.12.3 Tooltip Format 66
2.13 # Chart Options 66
2.13.1 Chart Size 66
2.13.2 Chart Padding 67
2.13.3 Color Pattern 67
2.13.4 Duration of Transition 67
2.14 # Line und Scatter Chart Options 68
2.14.1 Hide points 68
2.14.2 Point Size 68
2.15 # Pie Chart Options 68
2.15.1 Pie Label Format 68
2.16 # Gauge Chart Options 69
2.16.1 Gauge Label Format 69
2.16.2 Gauge Color Threshold Levels 70
2.16.3 Gauge Threshold Options 70
2.16.4 Gauge Min Max 71
2.17 # API 71
2.17.1 Flow 71
2.17.2 Data Name 71
2.17.3 Data Color 71
2.17.4 Axis Label 71
2.17.5 Axis Range 71

2.17.6 Resize ... 71
2.17.7 X Grid ... 71
2.18 # Transform ... 72
2.19 # Style ... 72
2.19.1 Style for Region ... 72
CSS: ... 73
Hinweis: ... 73
Beispiel: ... 73
2.19.2 Style for Grid ... 73
CSS: ... 74
Hinweis: ... 74
Beispiel: ... 74
3 ECharts ... 76
3.1 Einleitung ... 76
3.2 Ein EChart-Dashlet erzeugen ... 76
3.3 Daten ans Dashlet übergeben ... 76
3.4 Hinzufügen von Optionen ... 76
3.5 Generelle EChart-Optionen ... 76
3.5.1 chart.backgroundColor ... 76
3.5.2 chart.animation ... 77
3.5.3 chart.renderAsImage ... 77
3.5.4 chart.calculable ... 77
3.5.5 chart.symbolList ... 77
3.5.6 chart.calculableColor ... 77
3.5.7 chart.calculableHolderColor ... 78
3.5.8 chart.nameConnector ... 78
3.5.9 chart.valueConnector ... 78
3.5.10 chart.addDataAnimation ... 78
3.5.11 chart.animationThreshold ... 78
3.5.12 chart.animationDuration ... 78
3.5.13 chart.animationDurationUpdate ... 78
3.5.14 chart.animationEasing ... 78
3.6 Chart-Titel-Optionen ... 79
3.6.1 chart.title.show ... 79

3.6.2 chart.title.zlevel ... 79
3.6.3 chart.title.z ... 79
3.6.4 chart.title.text ... 79
3.6.5 chart.title.link ... 79
3.6.6 chart.title.target ... 80
3.6.7 chart.title.subtext ... 80
3.6.8 chart.title.sublink ... 80
3.6.9 chart.title.subtarget ... 80
3.6.10 chart.title.x ... 80
3.6.11 chart.title.y ... 80
3.6.12 chart.title.textAlign ... 80
3.6.13 chart.title.backgroundColor ... 80
3.6.14 chart.title.borderColor ... 81
3.6.15 chart.title.borderWidth ... 81
3.6.16 chart.title.padding ... 81
3.6.17 chart.title.itemGap ... 81
3.6.18 chart.title.textStyle.fontFamily ... 81
3.6.19 chart.title.textStyle.fontSize ... 81
3.6.20 chart.title.textStyle.fontStyle ... 81
3.6.21 chart.title.textStyle.fontWeight ... 81
3.6.22 chart.title.subtextStyle.fontFamily ... 82
3.6.23 chart.title.subtextStyle.fontSize ... 82
3.6.24 chart.title.subtextStyle.fontStyle ... 82
3.6.25 chart.title.subtextStyle.fontWeight ... 82
3.7 Chart-Toolbox-Optionen ... 82
3.7.1 chart.toolbox.show ... 82
3.7.2 chart.toolbox.zlevel ... 82
3.7.3 chart.toolbox.z ... 82
3.7.4 chart.toolbox.orient ... 83
3.7.5 chart.toolbox.x ... 83
3.7.6 chart.toolbox.y ... 83
3.7.7 chart.toolbox.backgroundColor ... 83
3.7.8 chart.toolbox.borderColor ... 83
3.7.9 chart.toolbox.borderWidth ... 83
3.7.10 chart.toolbox.padding ... 83
3.7.11 chart.toolbox.itemGap ... 83

3.7.12 chart.toolbox.itemSize ... 84
3.7.13 chart.toolbox.color ... 84
3.7.14 chart.toolbox.disableColor ... 84
3.7.15 chart.toolbox.effectiveColor ... 84
3.7.16 chart.toolbox.showTitle ... 84
3.7.17 chart.toolbox.textStyle.fontFamily ... 84
3.7.18 chart.toolbox.textStyle.fontSize ... 84
3.7.19 chart.toolbox.textStyle.fontStyle ... 84
3.7.20 chart.toolbox.textStyle.fontWeight ... 85
3.8 Chart-Toolbox-Feature Mark ... 85
3.8.1 chart.toolbox.feature.mark.show ... 85
3.8.2 chart.toolbox.feature.mark.title.mark ... 85
3.8.3 chart.toolbox.feature.mark.title.markUndo ... 85
3.8.4 chart.toolbox.feature.mark.title.markClear ... 85
3.8.5 chart.toolbox.feature.mark.lineStyle.color ... 85
3.8.6 chart.toolbox.feature.mark.lineStyle.type ... 85
3.8.7 chart.toolbox.feature.mark.lineStyle.width ... 86
3.8.8 chart.toolbox.feature.mark.lineStyle.shadowColor (nicht sichtbar) ... 86
3.8.9 chart.toolbox.feature.mark.lineStyle.shadowBlur (nicht sichtbar) ... 86
3.8.10 chart.toolbox.feature.mark.lineStyle.shadowOffsetX (nicht sichtbar) ... 86
3.8.11 chart.toolbox.feature.mark.lineStyle.shadowOffsetY (nicht sichtbar) ... 86
3.9 Chart-Toolbox-Feature dataZoom ... 86
3.9.1 chart.toolbox.feature.dataZoom.show ... 86
3.9.2 chart.toolbox.feature.dataZoom.title.dataZoom ... 86
3.9.3 chart.toolbox.feature.dataZoom.title.dataZoomReset ... 86
3.10 Chart-Toolbox-Feature dataView ... 87
3.10.1 chart.toolbox.feature.dataView.show ... 87
3.10.2 chart.toolbox.feature.dataView.title ... 87
3.10.3 chart.toolbox.feature.dataView.readOnly ... 87
3.10.4 chart.toolbox.feature.dataView.lang ... 87
3.10.5 chart.toolbox.feature.dataView.optionToContent (noch nicht implementiert) ... 87
3.10.6 chart.toolbox.feature.dataView.contentToOption (noch nicht implementiert) ... 87
3.11 Chart-Toolbox-Feature magicType ... 88
3.11.1 chart.toolbox.feature.magicType.show ... 88
3.11.2 chart.toolbox.feature.magicType.type ... 88

3.11.3 chart.toolbox.feature.magicType.title.line ... 88
3.11.4 chart.toolbox.feature.magicType.title.bar ... 88
3.11.5 chart.toolbox.feature.magicType.title.stack ... 88
3.11.6 chart.toolbox.feature.magicType.title.tiled ... 88
3.11.7 chart.toolbox.feature.magicType.title.force ... 88
3.11.8 chart.toolbox.feature.magicType.title.chord ... 89
3.11.9 chart.toolbox.feature.magicType.title.pie ... 89
3.11.10 chart.toolbox.feature.magicType.title.funnel ... 89
3.11.11 chart.toolbox.feature.magicType.option.line ... 89
3.11.12 chart.toolbox.feature.magicType.option.bar ... 89
3.11.13 chart.toolbox.feature.magicType.option.stack ... 89
3.11.14 chart.toolbox.feature.magicType.option.tiled ... 89
3.11.15 chart.toolbox.feature.magicType.option.force ... 89
3.11.16 chart.toolbox.feature.magicType.option.chord ... 89
3.11.17 chart.toolbox.feature.magicType.option.pie ... 90
3.11.18 chart.toolbox.feature.magicType.option.funnel ... 90
3.12 Chart-Toolbox-Feature Restore ... 90
3.12.1 chart.toolbox.feature.restore.show ... 90
3.12.2 chart.toolbox.feature.restore.title ... 90
3.13 Chart-Toolbox-Feature saveAsImage ... 90
3.13.1 chart.toolbox.feature.saveAsImage.show ... 90
3.13.2 chart.toolbox.feature.saveAsImage.title ... 90
3.13.3 chart.toolbox.feature.saveAsImage.type ... 90
3.13.4 chart.toolbox.feature.saveAsImage.name ... 91
3.13.5 chart.toolbox.feature.saveAsImage.lang ... 91
3.14 Chart-Tooltip-Optionen ... 91
3.14.1 chart.tooltip.show ... 91
3.14.2 chart.tooltip.zlevel ... 91
3.14.3 chart.tooltip.z ... 91
3.14.4 chart.tooltip.showContent ... 92
3.14.5 chart.tooltip.trigger ... 92
3.14.6 chart.tooltip.position ... 92
3.14.7 chart.tooltip.formatter (nicht getestet) ... 92
3.14.8 chart.tooltip.islandFormatter (nicht getestet) ... 93
3.14.9 chart.tooltip.showDelay ... 93

3.14.10 chart.tooltip.hideDelay ... 93
3.14.11 chart.tooltip.transitionDuration (nicht sichtbar) ... 93
3.14.12 chart.tooltip.enterable (nicht getestet) ... 94
3.14.13 chart.tooltip.backgroundColor ... 94
3.14.14 chart.tooltip.borderColor ... 94
3.14.15 chart.tooltip.borderRadius ... 94
3.14.16 chart.tooltip.borderWidth ... 94
3.14.17 chart.tooltip.padding ... 94
3.14.18 chart.tooltip.textStyle.fontFamily ... 94
3.14.19 chart.tooltip.textStyle.fontSize ... 94
3.14.20 chart.tooltip.textStyle.fontStyle ... 95
3.14.21 chart.tooltip.textStyle.fontWeight ... 95
3.14.22 chart.tooltip.axisPointer.type ... 95
3.14.23 chart.tooltip.axisPointer.lineStyle.color ... 95
3.14.24 chart.tooltip.axisPointer.lineStyle.type ... 95
3.14.25 chart.tooltip.axisPointer.lineStyle.width ... 95
3.14.26 chart.tooltip.axisPointer.lineStyle.shadowColor (nicht sichtbar) ... 95
3.14.27 chart.tooltip.axisPointer.lineStyle.shadowBlur (nicht sichtbar) ... 95
3.14.28 chart.tooltip.axisPointer.lineStyle.shadowOffsetX (nicht sichtbar) ... 96
3.14.29 chart.tooltip.axisPointer.lineStyle.shadowOffsetY (nicht sichtbar) ... 96
3.14.30 chart.tooltip.axisPointer.crossStyle.color ... 96
3.14.31 chart.tooltip.axisPointer.crossStyle.type ... 96
3.14.32 chart.tooltip.axisPointer.crossStyle.width ... 96
3.14.33 chart.tooltip.axisPointer.crossStyle.shadowColor (nicht sichtbar) ... 96
3.14.34 chart.tooltip.axisPointer.crossStyle.shadowBlur (nicht sichtbar) ... 96
3.14.35 chart.tooltip.axisPointer.crossStyle.shadowOffsetX (nicht sichtbar) ... 96
3.14.36 chart.tooltip.axisPointer.crossStyle.shadowOffsetY (nicht sichtbar) ... 97
3.14.37 chart.tooltip.axisPointer.shadowStyle.color (nicht sichtbar) ... 97
3.14.38 chart.tooltip.axisPointer.shadowStyle.type (nicht sichtbar) ... 97
3.14.39 chart.tooltip.axisPointer.textStyle.fontFamily (nicht testbar) ... 97
3.14.40 chart.tooltip.axisPointer.textStyle.fontSize (nicht testbar) ... 97
3.14.41 chart.tooltip.axisPointer.textStyle.fontStyle (nicht testbar) ... 97
3.14.42 chart.tooltip.axisPointer.textStyle.fontWeight (nicht testbar) ... 97
3.15 Chart-Legend-Optionen ... 98
3.15.1 chart.legend.show ... 98
3.15.2 chart.legend.zlevel ... 98

3.15.3 chart.legend.z ... 98
3.15.4 chart.legend.orient ... 98
3.15.5 chart.legend.x ... 98
3.15.6 chart.legend.y ... 99
3.15.7 chart.legend.backgroundColor ... 99
3.15.8 chart.legend.borderColor ... 99
3.15.9 chart.legend.borderWidth ... 99
3.15.10 chart.legend.padding ... 99
3.15.11 chart.legend.itemGap ... 99
3.15.12 chart.legend.itemWidth ... 99
3.15.13 chart.legend.itemHeight ... 99
3.15.14 chart.legend.textStyle.fontFamily ... 100
3.15.15 chart.legend.textStyle.fontSize ... 100
3.15.16 chart.legend.textStyle.fontStyle ... 100
3.15.17 chart.legend.textStyle.fontWeight ... 100
3.15.18 chart.legend.formatter (nicht getestet) ... 100
3.15.19 chart.legend.selectedMode ... 100
3.15.20 chart.legend.selected (nicht getestet) ... 100
3.16 Chart-dataRange-Optionen ... 101
3.16.1 chart.dataRange.show ... 101
3.16.2 chart.dataRange.zlevel ... 101
3.16.3 chart.dataRange.z ... 101
3.16.4 chart.dataRange.orient ... 101
3.16.5 chart.dataRange.x ... 101
3.16.6 chart.dataRange.y ... 102
3.16.7 chart.dataRange.backgroundColor ... 102
3.16.8 chart.dataRange.borderColor ... 102
3.16.9 chart.dataRange.borderWidth ... 102
3.16.10 chart.dataRange.padding ... 102
3.16.11 chart.dataRange.itemGap ... 102
3.16.12 chart.dataRange.itemWidth ... 102
3.16.13 chart.dataRange.itemHeight ... 102
3.16.14 chart.dataRange.min ... 103
3.16.15 chart.dataRange.max ... 103
3.16.16 chart.dataRange.precision ... 103
3.16.17 chart.dataRange.splitNumber ... 103

3.16.18 chart.dataRange.splitList ... 103
3.16.19 chart.dataRange.range (nicht getestet) ... 104
3.16.20 chart.dataRange.selectedMode ... 104
3.16.21 chart.dataRange.calculable ... 104
3.16.22 chart.dataRange.hoverLink (nicht getestet) ... 104
3.16.23 chart.dataRange.realtime (nicht testbar) ... 104
3.16.24 chart.dataRange.color ... 104
3.16.25 chart.dataRange.formatter (nicht getestet) ... 104
3.16.26 chart.dataRange.text ... 105
3.16.27 chart.dataRange.textStyle.fontFamily ... 105
3.16.28 chart.dataRange.textStyle.fontSize ... 105
3.16.29 chart.dataRange.textStyle.fontStyle ... 105
3.16.30 chart.dataRange.textStyle.fontWeight ... 105
3.17 Chart-dataZoom-Optionen ... 105
3.17.1 chart.dataZoom.zlevel ... 105
3.17.2 chart.dataZoom.z ... 106
3.17.3 chart.dataZoom.show ... 106
3.17.4 chart.dataZoom.orient ... 106
3.17.5 chart.dataZoom.x ... 106
3.17.6 chart.dataZoom.y ... 106
3.17.7 chart.dataZoom.width (funktioniert nicht) ... 106
3.17.8 chart.dataZoom.height ... 106
3.17.9 chart.dataZoom.backgroundColor ... 106
3.17.10 chart.dataZoom.dataBackgroundColor ... 107
3.17.11 chart.dataZoom.fillerColor ... 107
3.17.12 chart.dataZoom.handleColor ... 107
3.17.13 chart.dataZoom.handleSize ... 107
3.17.14 chart.dataZoom.xAxisIndex ... 107
3.17.15 chart.dataZoom.yAxisIndex (kein Beispiel vorhanden, konnte nicht getestet werden) ... 107
3.17.16 chart.dataZoom.start ... 107
3.17.17 chart.dataZoom.end ... 107
3.17.18 chart.dataZoom.showDetail ... 108
3.17.19 chart.dataZoom.realtime ... 108
3.17.20 chart.dataZoom.zoomLock ... 108
3.18 Chart-roamController-Optionen (nicht getestet, funktioniert nur bei Map) ... 108

3.18.1 chart.roamController.show ... 108
3.18.2 chart.roamController.zlevel ... 108
3.18.3 chart.roamController.z ... 108
3.18.4 chart.roamController.x ... 108
3.18.5 chart.roamController.y ... 109
3.18.6 chart.roamController.width ... 109
3.18.7 chart.roamController.height ... 109
3.18.8 chart.roamController.backgroundColor ... 109
3.18.9 chart.roamController.borderColor ... 109
3.18.10 chart.roamController.borderWidth ... 109
3.18.11 chart.roamController.padding ... 109
3.18.12 chart.roamController.fillerColor ... 109
3.18.13 chart.roamController.handleColor ... 110
3.18.14 chart.roamController.step ... 110
3.18.15 chart.roamController.mapTypeControl ... 110
3.19 Chart-Grid-Optionen ... 110
3.19.1 chart.grid.zlevel ... 110
3.19.2 chart.grid.z ... 110
3.19.3 chart.grid.x ... 111
3.19.4 chart.grid.y ... 111
3.19.5 chart.grid.x2 ... 111
3.19.6 chart.grid.y2 ... 111
3.19.7 chart.grid.width ... 111
3.19.8 chart.grid.height ... 111
3.19.9 chart.grid.backgroundColor ... 111
3.19.10 chart.grid.borderWidth ... 112
3.19.11 chart.grid.borderColor ... 112
3.20 Series-Bar ... 112
3.20.1 chart.series.i[].stack (Auch in line/area) ... 112
3.20.2 chart.series.i[].barGap ... 112
3.20.3 chart.series.i[].barCategoryGap ... 112
3.20.4 chart.series.i[].barMinHeight ... 112
3.20.5 chart.series.i[].barWidth ... 112
3.20.6 chart.series.i[].barMaxWidth ... 112
3.21 Series-Gauge ... 113

3.21.1 chart.series.i[].axisLabel.clickable ... 113
3.21.2 chart.series.i[].axisLabel.formatter ... 113
3.21.3 chart.series.i[].axisLabel.margin ... 113
3.21.4 chart.series.i[].axisLabel.rotate ... 113
3.21.5 chart.series.i[].axisLabel.show ... 113
3.21.6 chart.series.i[].axisLabel.textStyle.align ... 113
3.21.7 chart.series.i[].axisLabel.textStyle.baseline ... 113
3.21.8 chart.series.i[].axisLabel.textStyle.color ... 113
3.21.9 chart.series.i[].axisLabel.textStyle.fontFamily ... 114
3.21.10 chart.series.i[].axisLabel.textStyle.fontSize ... 114
3.21.11 chart.series.i[].axisLabel.textStyle.fontStyle ... 114
3.21.12 chart.series.i[].axisLabel.textStyle.fontWeight ... 114
3.21.13 chart.series.i[].splitNumber ... 114
3.21.14 chart.series.i[].axisLine.show ... 114
3.21.15 chart.series.i[].axisLine.lineStyle.color ... 114
3.21.16 chart.series.i[].axisLine.lineStyle.type ... 114
3.21.17 chart.series.i[].axisLine.lineStyle.width ... 115
3.21.18 chart.series.i[].axisLine.lineStyle.shadowColor ... 115
3.21.19 chart.series.i[].axisLine.lineStyle.shadowBlur ... 115
3.21.20 chart.series.i[].axisLine.lineStyle.shadowOffsetX ... 115
3.21.21 chart.series.i[].axisLine.lineStyle.shadowOffsetY ... 115
3.21.22 chart.series.i[].axisTick.show ... 115
3.21.23 chart.series.i[].axisTick.lineStyle.color ... 115
3.21.24 chart.series.i[].axisTick.lineStyle.type ... 115
3.21.25 chart.series.i[].axisTick.lineStyle.width ... 116
3.21.26 chart.series.i[].axisTick.lineStyle.shadowColor ... 116
3.21.27 chart.series.i[].axisTick.lineStyle.shadowBlur ... 116
3.21.28 chart.series.i[].axisTick.lineStyle.shadowOffsetX ... 116
3.21.29 chart.series.i[].axisTick.lineStyle.shadowOffsetY ... 116
3.21.30 chart.series.i[].axisTick.splitNumber ... 116
3.21.31 chart.series.i[].axisTick.length ... 116
3.21.32 chart.series.i[].detail.show ... 116
3.21.33 chart.series.i[].detail.backgroundColor ... 117
3.21.34 chart.series.i[].detail.borderWidth ... 117
3.21.35 chart.series.i[].detail.borderColor ... 117
3.21.36 chart.series.i[].detail.width ... 117

3.21.37 chart.series.i[].detail.height ... 117
3.21.38 chart.series.i[].detail.offsetCenter ... 117
3.21.39 chart.series.i[].detail.formatter ... 117
3.21.40 chart.series.i[].detail.textStyle ... 117
3.21.41 chart.series.i[].detail.textStyle.color ... 118
3.21.42 chart.series.i[].detail.textStyle.align ... 118
3.21.43 chart.series.i[].detail.textStyle.baseline ... 118
3.21.44 chart.series.i[].detail.textStyle.fontFamily ... 118
3.21.45 chart.series.i[].detail.textStyle.fontSize ... 118
3.21.46 chart.series.i[].detail.textStyle.fontStyle ... 118
3.21.47 chart.series.i[].detail.textStyle.fontWeight ... 118
3.21.48 chart.series.i[].startAngle ... 118
3.21.49 chart.series.i[].endAngle ... 119
3.21.50 chart.series.i[].pointer.length ... 119
3.21.51 chart.series.i[].pointer.width ... 119
3.21.52 chart.series.i[].pointer.color ... 119
3.21.53 chart.series.i[].splitLine.show ... 119
3.21.54 chart.series.i[].splitLine.length ... 119
3.21.55 chart.series.i[].splitLine.lineStyle.color ... 119
3.21.56 chart.series.i[].splitLine.lineStyle.shadowColor ... 119
3.21.57 chart.series.i[].splitLine.lineStyle.type ... 120
3.21.58 chart.series.i[].splitLine.lineStyle.width ... 120
3.21.59 chart.series.i[].splitLine.lineStyle.shadowBlur ... 120
3.21.60 chart.series.i[].splitLine.lineStyle.shadowOffsetX ... 120
3.21.61 chart.series.i[].splitLine.lineStyle.shadowOffsetY ... 120
3.21.62 chart.series.i[].title.show ... 120
3.21.63 chart.series.i[].title.offsetCenter ... 120
3.21.64 chart.series.i[].title.textStyle.color ... 120
3.21.65 chart.series.i[].title.textStyle.align ... 121
3.21.66 chart.series.i[].title.textStyle.baseline ... 121
3.21.67 chart.series.i[].title.textStyle.fontFamily ... 121
3.21.68 chart.series.i[].title.textStyle.fontSize ... 121
3.21.69 chart.series.i[].title.textStyle.fontStyle ... 121
3.21.70 chart.series.i[].title.textStyle.fontWeight ... 121
3.22 Series-Chord ... 121
3.22.1 chart.series.i[].data[] (Noch nicht implementiert) ... 121

3.22.2 chart.series.i[].data[].children (Noch nicht implementiert) ... 121
3.22.3 chart.series.i[].data[].evolution (Noch nicht implementiert) ... 121
3.22.4 chart.series.i[].data[].evolution.detail (Noch nicht implementiert) ... 122
3.22.5 chart.series.i[].data[].evolution.detail.img (Noch nicht implementiert) ... 122
3.22.6 chart.series.i[].data[].evolution.detail.link (Noch nicht implementiert) ... 122
3.22.7 chart.series.i[].data[].evolution.detail.text (Noch nicht implementiert) ... 122
3.22.8 chart.series.i[].data[].evolution.time (Noch nicht implementiert) ... 122
3.22.9 chart.series.i[].data[].evolution.value (Noch nicht implementiert) ... 122
3.22.10 chart.series.i[].data[].itemStyle (Noch nicht implementiert) ... 122
3.22.11 chart.series.i[].data[].name (Noch nicht implementiert) ... 122
3.22.12 chart.series.i[].data[].label (Noch nicht implementiert) ... 122
3.22.13 chart.series.i[].data[].value (Noch nicht implementiert) ... 122
3.22.14 chart.series.i[].data[].symbol (Noch nicht implementiert) ... 122
3.22.15 chart.series.i[].data[].symbolSize (Noch nicht implementiert) ... 123
3.22.16 chart.series.i[].data[].weight (Noch nicht implementiert) ... 123
3.22.17 chart.series.i[].data[].itemStyle.normal (Noch nicht implementiert) ... 123
3.22.18 chart.series.i[].data[].itemStyle.normal.color (Noch nicht implementiert) ... 123
3.22.19 chart.series.i[].data[].itemStyle.normal.chordStyle (Noch nicht implementiert) ... 123
3.22.20 chart.series.i[].data[].itemStyle.normal.chordStyle.width (Noch nicht implementiert) ... 123
3.22.21 chart.series.i[].data[].itemStyle.normal.chordStyle.color (Noch nicht implementiert) ... 123
3.22.22 chart.series.i[].data[].itemStyle.normal.chordStyle.borderWidth (Noch nicht implementiert) ... 123
3.22.23 chart.series.i[].data[].itemStyle.normal.chordStyle.borderColor (Noch nicht implementiert) ... 124
3.22.24 chart.series.i[].data[].itemStyle.normal.label (Noch nicht implementiert) ... 124
3.22.25 chart.series.i[].data[].itemStyle.normal.label.show (Noch nicht implementiert) ... 124
3.22.26 chart.series.i[].data[].itemStyle.normal.label.rotate (Noch nicht implementiert) ... 124
3.22.27 chart.series.i[].data[].itemStyle.normal.label.distance (Noch nicht implementiert) ... 124
3.22.28 chart.series.i[].data[].itemStyle.normal.label.formatter (Noch nicht implementiert) ... 124
3.22.29 chart.series.i[].data[].itemStyle.normal.label.textStyle (Noch nicht implementiert) ... 125
3.22.30 chart.series.i[].data[].itemStyle.normal.label.textStyle.color (Noch nicht implementiert) ... 125
3.22.31 chart.series.i[].data[].itemStyle.normal.label.textStyle.align (Noch nicht implementiert) ... 126
3.22.32 chart.series.i[].data[].itemStyle.normal.label.textStyle.baseline (Noch nicht implementiert) ... 126
3.22.33 chart.series.i[].data[].itemStyle.normal.label.textStyle.fontFamily (Noch nicht implementiert) ... 126
3.22.34 chart.series.i[].data[].itemStyle.normal.label.textStyle.fontSize (Noch nicht implementiert) ... 126
3.22.35 chart.series.i[].data[].itemStyle.normal.label.textStyle.fontStyle (Noch nicht implementiert) ... 126
3.22.36 chart.series.i[].data[].itemStyle.normal.label.textStyle.fontWeight (Noch nicht implementiert) ... 126
3.22.37 chart.series.i[].data[].itemStyle.emphasis (Noch nicht implementiert) ... 126

3.22.38 chart.series.i[].data[].itemStyle.emphasis.color (Noch nicht implementiert) ... 127
3.22.39 chart.series.i[].data[].itemStyle.emphasis.chordStyle (Noch nicht implementiert) ... 127
3.22.40 chart.series.i[].data[].itemStyle.emphasis.chordStyle.width (Noch nicht implementiert) ... 127
3.22.41 chart.series.i[].data[].itemStyle.emphasis.chordStyle.color (Noch nicht implementiert) ... 127
3.22.42 chart.series.i[].data[].itemStyle.emphasis.chordStyle.borderWidth (Noch nicht implementiert) ... 127
3.22.43 chart.series.i[].data[].itemStyle.emphasis.chordStyle.borderColor (Noch nicht implementiert) ... 127
3.22.44 chart.series.i[].data[].itemStyle.emphasis.label (Noch nicht implementiert) ... 127
3.22.45 chart.series.i[].data[].itemStyle.emphasis.label.show (Noch nicht implementiert) ... 128
3.22.46 chart.series.i[].data[].itemStyle.emphasis.label.rotate (Noch nicht implementiert) ... 128
3.22.47 chart.series.i[].data[].itemStyle.emphasis.label.distance (Noch nicht implementiert) ... 128
3.22.48 chart.series.i[].data[].itemStyle.emphasis.label.formatter (Noch nicht implementiert) ... 128
3.22.49 chart.series.i[].data[].itemStyle.emphasis.label.textStyle (Noch nicht implementiert) ... 129
3.22.50 chart.series.i[].data[].itemStyle.emphasis.label.textStyle.color (Noch nicht implementiert) ... 129
3.22.51 chart.series.i[].data[].itemStyle.emphasis.label.textStyle.align (Noch nicht implementiert) ... 129
3.22.52 chart.series.i[].data[].itemStyle.emphasis.label.textStyle.baseline (Noch nicht implementiert) ... 129
3.22.53 chart.series.i[].data[].itemStyle.emphasis.label.textStyle.fontFamily (Noch nicht implementiert) ... 130
3.22.54 chart.series.i[].data[].itemStyle.emphasis.label.textStyle.fontSize (Noch nicht implementiert) ... 130
3.22.55 chart.series.i[].data[].itemStyle.emphasis.label.textStyle.fontStyle (Noch nicht implementiert) ... 130
3.22.56 chart.series.i[].data[].itemStyle.emphasis.label.textStyle.fontWeight (Noch nicht implementiert) ... 130
3.22.57 chart.series.i[].symbol (Auch bei line/area, scatter, radar, force, tree) ... 130
3.22.58 chart.series.i[].symbolSize (Auch bei line/area, scatter, radar, force, tree) ... 130
3.22.59 chart.series.i[].clockWise (Auch bei pie) ... 130
3.22.60 chart.series.i[].categories.name (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 131
3.22.61 chart.series.i[].categories.symbol (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 131
3.22.62 chart.series.i[].categories.symbolSize (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 131
3.22.63 chart.series.i[].categories.itemStyle (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 131
3.22.64 chart.series.i[].categories.itemStyle.normal (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 131
3.22.65 chart.series.i[].categories.itemStyle.normal.color (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 131
3.22.66 chart.series.i[].categories.itemStyle.normal.chordStyle (Noch nicht implementiert) ... 132
3.22.67 chart.series.i[].categories.itemStyle.normal.chordStyle.width (Noch nicht implementiert) ... 132
3.22.68 chart.series.i[].categories.itemStyle.normal.chordStyle.color (Noch nicht implementiert) ... 132
3.22.69 chart.series.i[].categories.itemStyle.normal.chordStyle.borderWidth (Noch nicht implementiert) ... 132

3.22.70 chart.series.i[].categories.itemStyle.normal.chordStyle.borderColor (Noch nicht implementiert) 132

3.22.71 chart.series.i[].categories.itemStyle.normal.label (Noch nicht implementiert, auch bei force (noch nicht implementiert)) 132

3.22.72 chart.series.i[].categories.itemStyle.normal.label.show (Noch nicht implementiert, auch bei force (noch nicht implementiert)) 132

3.22.73 chart.series.i[].categories.itemStyle.normal.label.rotate (Noch nicht implementiert, auch bei force (noch nicht implementiert)) 133

3.22.74 chart.series.i[].categories.itemStyle.normal.label.distance (Noch nicht implementiert, auch bei force (noch nicht implementiert)) 133

3.22.75 chart.series.i[].categories.itemStyle.normal.label.formatter (Noch nicht implementiert, auch bei force (noch nicht implementiert)) 133

3.22.76 chart.series.i[].categories.itemStyle.normal.label.textStyle (Noch nicht implementiert, auch bei force (noch nicht implementiert)) 134

3.22.77 chart.series.i[].categories.itemStyle.normal.label.textStyle.color (Noch nicht implementiert, auch bei force (noch nicht implementiert)) 134

3.22.78 chart.series.i[].categories.itemStyle.normal.label.textStyle.align (Noch nicht implementiert, auch bei force (noch nicht implementiert)) 134

3.22.79 chart.series.i[].categories.itemStyle.normal.label.textStyle.baseline (Noch nicht implementiert, auch bei force (noch nicht implementiert)) 134

3.22.80 chart.series.i[].categories.itemStyle.normal.label.textStyle.fontFamily (Noch nicht implementiert, auch bei force (noch nicht implementiert)) 134

3.22.81 chart.series.i[].categories.itemStyle.normal.label.textStyle.fontSize (Noch nicht implementiert, auch bei force (noch nicht implementiert)) 135

3.22.82 chart.series.i[].categories.itemStyle.normal.label.textStyle.fontStyle (Noch nicht implementiert, auch bei force (noch nicht implementiert)) 135

3.22.83 chart.series.i[].categories.itemStyle.normal.label.textStyle.fontWeight (Noch nicht implementiert, auch bei force (noch nicht implementiert)) 135

3.22.84 chart.series.i[].categories.itemStyle.emphasis (Noch nicht implementiert, auch bei force (noch nicht implementiert)) 135

3.22.85 chart.series.i[].categories.itemStyle.emphasis.color (Noch nicht implementiert, auch bei force (noch nicht implementiert)) 135

3.22.86 chart.series.i[].categories.itemStyle.emphasis.chordStyle (Noch nicht implementiert) 135

3.22.87 chart.series.i[].categories.itemStyle.emphasis.chordStyle.width (Noch nicht implementiert) 135

3.22.88 chart.series.i[].categories.itemStyle.emphasis.chordStyle.color (Noch nicht implementiert) 136

3.22.89 chart.series.i[].categories.itemStyle.emphasis.chordStyle.borderWidth (Noch nicht implementiert) 136

3.22.90 chart.series.i[].categories.itemStyle.emphasis.chordStyle.borderColor (Noch nicht implementiert) 136

3.22.91 chart.series.i[].categories.itemStyle.emphasis.label (Noch nicht implementiert, auch bei force (noch nicht implementiert)) 136

3.22.92 chart.series.i[].categories.itemStyle.emphasis.label.show (Noch nicht implementiert, auch bei force (noch nicht implementiert)) 136

3.22.93 chart.series.i[].categories.itemStyle.emphasis.label.rotate (Noch nicht implementiert, auch bei force (noch nicht implementiert)) 136

3.22.94 chart.series.i[].categories.itemStyle.emphasis.label.distance (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 136
3.22.95 chart.series.i[].categories.itemStyle.emphasis.label.formatter (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 137
3.22.96 chart.series.i[].categories.itemStyle.emphasis.label.textStyle (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 138
3.22.97 chart.series.i[].categories.itemStyle.emphasis.label.textStyle.color (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 138
3.22.98 chart.series.i[].categories.itemStyle.emphasis.label.textStyle.align (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 138
3.22.99 chart.series.i[].categories.itemStyle.emphasis.label.textStyle.baseline (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 138
3.22.100 chart.series.i[].categories.itemStyle.emphasis.label.textStyle.fontFamily (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 138
3.22.101 chart.series.i[].categories.itemStyle.emphasis.label.textStyle.fontSize (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 138
3.22.102 chart.series.i[].categories.itemStyle.emphasis.label.textStyle.fontStyle (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 139
3.22.103 chart.series.i[].categories.itemStyle.emphasis.label.textStyle.fontWeight (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 139
3.22.104 chart.series.i[].matrix (Auch bei force) ... 139
3.22.105 chart.series.i[].minRadius (Auch bei force) ... 139
3.22.106 chart.series.i[].maxRadius (Auch bei force) ... 139
3.22.107 chart.series.i[].ribbonType ... 139
3.22.108 chart.series.i[].showScale ... 139
3.22.109 chart.series.i[].showScaleText ... 139
3.22.110 chart.series.i[].padding ... 140
3.22.111 chart.series.i[].sort (Auch bei funnel) ... 140
3.22.112 chart.series.i[].sortSub ... 140
3.22.113 chart.series.i[].nodes (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 140
3.22.114 chart.series.i[].nodes.name (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 140
3.22.115 chart.series.i[].nodes.label (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 140
3.22.116 chart.series.i[].nodes.value (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 140
3.22.117 chart.series.i[].nodes.symbol (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 140
3.22.118 chart.series.i[].nodes.symbolSize (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 141
3.22.119 chart.series.i[].nodes.itemStyle (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 141
3.22.120 chart.series.i[].nodes.itemStyle.normal (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 141

3.22.121 chart.series.i[].nodes.itemStyle.normal.color (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 141
3.22.122 chart.series.i[].nodes.itemStyle.normal.chordStyle (Noch nicht implementiert) ... 141
3.22.123 chart.series.i[].nodes.itemStyle.normal.chordStyle.width (Noch nicht implementiert) ... 141
3.22.124 chart.series.i[].nodes.itemStyle.normal.chordStyle.color (Noch nicht implementiert) ... 141
3.22.125 chart.series.i[].nodes.itemStyle.normal.chordStyle.borderWidth (Noch nicht implementiert) ... 142
3.22.126 chart.series.i[].nodes.itemStyle.normal.chordStyle.borderColor (Noch nicht implementiert) ... 142
3.22.127 chart.series.i[].nodes.itemStyle.normal.label (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 142
3.22.128 chart.series.i[].nodes.itemStyle.normal.label.show (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 142
3.22.129 chart.series.i[].nodes.itemStyle.normal.label.rotate (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 142
3.22.130 chart.series.i[].nodes.itemStyle.normal.label.distance (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 142
3.22.131 chart.series.i[].nodes.itemStyle.normal.label.formatter (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 143
3.22.132 chart.series.i[].nodes.itemStyle.noraml.label.textStyle (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 144
3.22.133 chart.series.i[].nodes.itemStyle.normal.label.textStyle.color (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 144
3.22.134 chart.series.i[].nodes.itemStyle.normal.label.textStyle.align (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 144
3.22.135 chart.series.i[].nodes.itemStyle.normal.label.textStyle.baseline (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 144
3.22.136 chart.series.i[].nodes.itemStyle.normal.label.textStyle.fontFamily (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 144
3.22.137 chart.series.i[].nodes.itemStyle.normal.label.textStyle.fontSize (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 144
3.22.138 chart.series.i[].nodes.itemStyle.normal.label.textStyle.fontStyle (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 144
3.22.139 chart.series.i[].nodes.itemStyle.normal.label.textStyle.fontWeight (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 145
3.22.140 chart.series.i[].nodes.itemStyle.emphasis (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 145
3.22.141 chart.series.i[].nodes.itemStyle.emphasis.color (Noch nicht implementiert, auch bei force (noch nicht implementiert)) ... 145
3.22.142 chart.series.i[].nodes.itemStyle.emphasis.chordStyle (Noch nicht implementiert) ... 145
3.22.143 chart.series.i[].nodes.itemStyle.emphasis.chordStyle.width (Noch nicht implementiert) ... 145
3.22.144 chart.series.i[].nodes.itemStyle.emphasis.chordStyle.color (Noch nicht implementiert) ... 145
3.22.145 chart.series.i[].nodes.itemStyle.emphasis.chordStyle.borderWidth (Noch nicht implementiert) ... 145

3.22.146 chart.series.i[].nodes.itemStyle.emphasis.chordStyle.borderColor (Noch nicht implementiert) 146
3.22.147 chart.series.i[].nodes.itemStyle.emphasis.label (Noch nicht implementiert, auch bei force (noch nicht implementiert)) 146
3.22.148 chart.series.i[].nodes.itemStyle.emphasis.label.show (Noch nicht implementiert, auch bei force (noch nicht implementiert)) 146
3.22.149 chart.series.i[].nodes.itemStyle.emphasis.label.rotate (Noch nicht implementiert, auch bei force (noch nicht implementiert)) 146
3.22.150 chart.series.i[].nodes.itemStyle.emphasis.label.distance (Noch nicht implementiert, auch bei force (noch nicht implementiert)) 146
3.22.151 chart.series.i[].nodes.itemStyle.emphasis.label.formatter (Noch nicht implementiert, auch bei force (noch nicht implementiert)) 146
3.22.152 chart.series.i[].nodes.itemStyle.emphasis.label.textStyle (Noch nicht implementiert, auch bei force (noch nicht implementiert)) 147
3.22.153 chart.series.i[].nodes.itemStyle.emphasis.label.textStyle.color (Noch nicht implementiert, auch bei force (noch nicht implementiert)) 148
3.22.154 chart.series.i[].nodes.itemStyle.emphasis.label.textStyle.align (Noch nicht implementiert, auch bei force (noch nicht implementiert)) 148
3.22.155 chart.series.i[].nodes.itemStyle.emphasis.label.textStyle.baseline (Noch nicht implementiert, auch bei force (noch nicht implementiert)) 148
3.22.156 chart.series.i[].nodes.itemStyle.emphasis.label.textStyle.fontFamily (Noch nicht implementiert, auch bei force (noch nicht implementiert)) 148
3.22.157 chart.series.i[].nodes.itemStyle.emphasis.label.textStyle.fontSize (Noch nicht implementiert, auch bei force (noch nicht implementiert)) 148
3.22.158 chart.series.i[].nodes.itemStyle.emphasis.label.textStyle.fontStyle (Noch nicht implementiert, auch bei force (noch nicht implementiert)) 148
3.22.159 chart.series.i[].nodes.itemStyle.emphasis.label.textStyle.fontWeight (Noch nicht implementiert, auch bei force (noch nicht implementiert)) 148
3.23 Series-Heatmap 149
3.23.1 chart.series.i[].blurSize 149
3.23.2 chart.series.i[].gradientColors 149
3.23.3 chart.series.i[].minAlpha (Noch nicht implementiert) 149
3.23.4 chart.series.i[].valueScale 149
3.23.5 chart.series.i[].opacity 149
3.24 Series-ItemStyle 149
3.24.1 ItemStyle.normal 149
chart.series.i[].itemStyle.normal.color 149
chart.series.i[].itemStyle.normal.label.distance (nur bei Chord) 149
chart.series.i[].itemStyle.normal.label.formatter 150
chart.series.i[].itemStyle.normal.label.position (Nur bei line/area, bar, scatter, pie) 151
chart.series.i[].itemStyle.normal.label.rotate (Nur bei Chord) 151

chart.series.i[].itemStyle.normal.label.show 151
chart.series.i[].itemStyle.normal.label.textStyle.color 151
chart.series.i[].itemStyle.normal.label.textStyle.align 151
chart.series.i[].itemStyle.normal.label.textStyle.baseline 151
chart.series.i[].itemStyle.normal.label.textStyle.fontFamily 151
chart.series.i[].itemStyle.normal.label.textStyle.fontSize 151
chart.series.i[].itemStyle.normal.label.textStyle.fontStyle 151
chart.series.i[].itemStyle.normal.label.textStyle.fontWeight 152
chart.series.i[].itemStyle.normal.label.x (Nur bei Treemap) 152
chart.series.i[].itemStyle.normal.label.y (Nur bei Treemap) 152
chart.series.i[].itemStyle.normal.labelLine.show (Nur bei Pie) 152
chart.series.i[].itemStyle.normal.labelLine.length (Nur bei Pie) 152
chart.series.i[].itemStyle.normal.labelLine.lineStyle.color (Nur bei Pie) 152
chart.series.i[].itemStyle.normal.labelLine.lineStyle.type (Nur bei Pie) 152
chart.series.i[].itemStyle.normal.labelLine.lineStyle.width (Nur bei Pie) 152
chart.series.i[].itemStyle.normal.labelLine.lineStyle.shadowColor (Nur bei Pie) 152
chart.series.i[].itemStyle.normal.labelLine.lineStyle.shadowBlur (Nur bei Pie) 153
chart.series.i[].itemStyle.normal.labelLine.lineStyle.shadowOffsetX (Nur bei Pie) 153
chart.series.i[].itemStyle.normal.labelLine.lineStyle.shadowOffsetY (Nur bei Pie) 153
chart.series.i[].itemStyle.normal.lineStyle.color (Nur bei line/area) 153
chart.series.i[].itemStyle.normal.lineStyle.type (Nur bei line/area) 153
chart.series.i[].itemStyle.normal.lineStyle.width (Nur bei line/area) 153
chart.series.i[].itemStyle.normal.lineStyle.shadowColor (Nur bei line/area) 153
chart.series.i[].itemStyle.normal.lineStyle.shadowBlur (Nur bei line/area) 153
chart.series.i[].itemStyle.normal.lineStyle.shadowOffsetX (Nur bei line/area) 153
chart.series.i[].itemStyle.normal.lineStyle.shadowOffsetY (Nur bei line/area) 154
chart.series.i[].itemStyle.normal.areaStyle.borderColor (Nur bei line/area, map (noch nicht implementiert)) 154
chart.series.i[].itemStyle.normal.areaStyle.borderWidth (Nur bei line/area, map (noch nicht implementiert)) 154
chart.series.i[].itemStyle.normal.areaStyle.color (Nur bei line/area, map) 154
chart.series.i[].itemStyle.normal.areaStyle.type (Nur bei line/area, map) 154
chart.series.i[].itemStyle.normal.barBorderColor (Nur bei bar) 154
chart.series.i[].itemStyle.normal.barBorderRadius (Nur bei bar) 154
chart.series.i[].itemStyle.normal.barBorderWidth (Nur bei bar) 154
chart.series.i[].itemStyle.normal.borderColor (Nur bei line/area, scatter, pie, map) 154

chart.series.i[].itemStyle.normal.borderWidth (Nur bei line/area, scatter, pie map) 155
chart.series.i[].itemStyle.normal.chordStyle.width (Nur bei Chord) 155
chart.series.i[].itemStyle.normal.chordStyle.color (Nur bei Chord) 155
chart.series.i[].itemStyle.normal.chordStyle.borderWidth (Nur bei Chord) 155
chart.series.i[].itemStyle.normal.chordStyle.borderColor (Nur bei Chord) 155
chart.series.i[].itemStyle.normal.textStyle.fontFamily (Nur bei WorldCloud) 155
chart.series.i[].itemStyle.normal.textStyle.fontSize (Nur bei WorldCloud) 155
chart.series.i[].itemStyle.normal.textStyle.fontStyle (Nur bei WorldCloud) 155
chart.series.i[].itemStyle.normal.textStyle.fontWeight (Nur bei WorldCloud) 155
chart.series.i[].itemStyle.normal.breadcrumb.show (Nur bei treemap (noch nicht implementiert)) 156
chart.series.i[].itemStyle.normal.breadcrumb.textStyle.color (Nur bei treemap (noch nicht implementiert)) 156
chart.series.i[].itemStyle.normal.breadcrumb.textStyle.align (Nur bei treemap (noch nicht implementiert)) 156
chart.series.i[].itemStyle.normal.breadcrumb.textStyle.baseline (Nur bei treemap (noch nicht implementiert)) 156
chart.series.i[].itemStyle.normal.breadcrumb.textStyle.fontFamily (Nur bei treemap (noch nicht implementiert)) 156
chart.series.i[].itemStyle.normal.breadcrumb.textStyle.fontSize (Nur bei treemap (noch nicht implementiert)) 156
chart.series.i[].itemStyle.normal.breadcrumb.textStyle.fontStyle (Nur bei treemap (noch nicht implementiert)) 156
chart.series.i[].itemStyle.normal.breadcrumb.textStyle.fontWeight (Nur bei treemap (noch nicht implementiert)) 157
chart.series.i[].itemStyle.normal.childBorderWidth (Nur bei treemap (noch nicht implementiert)) 157
chart.series.i[].itemStyle.normal.childBorderColor (Nur bei treemap (noch nicht implementiert)) 157
chart.series.i[].itemStyle.normal.linkStyle.normal (Nur bei force (noch nicht implementiert)) 157
chart.series.i[].itemStyle.normal.linkStyle.normal.type (Nur bei force (noch nicht implementiert)) 157
chart.series.i[].itemStyle.normal.linkStyle.normal.color (Nur bei force (noch nicht implementiert)) 157
chart.series.i[].itemStyle.normal.linkStyle.normal.width (Nur bei force (noch nicht implementiert)) 157
chart.series.i[].itemStyle.normal.linkStyle.emphasis (Nur bei force (noch nicht implementiert)) 157
chart.series.i[].itemStyle.normal.linkStyle.emphasis.type (Nur bei force (noch nicht implementiert)) 157
chart.series.i[].itemStyle.normal.linkStyle.emphasis.color (Nur bei force (noch nicht implementiert)) ... 158
chart.series.i[].itemStyle.normal.linkStyle.emphasis.width (Nur bei force (noch nicht implementiert)) .. 158
chart.series.i[].itemStyle.normal.nodeStyle.color (Nur bei force (noch nicht implementiert)) 158
chart.series.i[].itemStyle.normal.nodeStyle.borderColor (Nur bei force (noch nicht implementiert)) 158
chart.series.i[].itemStyle.normal.nodeStyle.borderWidth (Nur bei force (noch nicht implementiert)) 158
3.24.2 ItemStyle.emphasis 158

chart.series.i[].itemStyle.emphasis.color 158

chart.series.i[].itemStyle.emphasis.label.distance (Nur bei chord) 158

chart.series.i[].itemStyle.emphasis.label.formatter (Nur bei line/area, bar, scatter, pie, radar, chord, force, map, gauge, enentRiver, venn, treemap, tree) 159

chart.series.i[].itemStyle.emphasis.label.position (Nur bei line/area, bar, scatter, pie) 160

chart.series.i[].itemStyle.emphasis.label.rotate (Nur bei chord) 160

chart.series.i[].itemStyle.emphasis.label.show (Nur bei line/area, bar, scatter, pie, radar, chord, force, map, gauge, enentRiver, venn, treemap, tree) 160

chart.series.i[].itemStyle.emphasis.label.textStyle.color (Nur bei line/area, bar, scatter, pie, radar, chord, force, map, gauge, enentRiver, venn, treemap, tree) 160

chart.series.i[].itemStyle.emphasis.label.textStyle.align (Nur bei line/area, bar, scatter, pie, radar, chord, force, map, gauge, enentRiver, venn, treemap, tree) 160

chart.series.i[].itemStyle.emphasis.label.textStyle.baseline (Nur bei line/area, bar, scatter, pie, radar, chord, force, map, gauge, enentRiver, venn, treemap, tree) 160

chart.series.i[].itemStyle.emphasis.label.textStyle.fontFamily (Nur bei line/area, bar, scatter, pie, radar, chord, force, map, gauge, enentRiver, venn, treemap, tree) 160

chart.series.i[].itemStyle.emphasis.label.textStyle.fontSize (Nur bei line/area, bar, scatter, pie, radar, chord, force, map, gauge, enentRiver, venn, treemap, tree) 161

chart.series.i[].itemStyle.emphasis.label.textStyle.fontStyle (Nur bei line/area, bar, scatter, pie, radar, chord, force, map, gauge, enentRiver, venn, treemap, tree) 161

chart.series.i[].itemStyle.emphasis.label.textStyle.fontWeight (Nur bei line/area, bar, scatter, pie, radar, chord, force, map, gauge, enentRiver, venn, treemap, tree) 161

chart.series.i[].itemStyle.emphasis.label.x (Nur bei treemap) 161

chart.series.i[].itemStyle.emphasis.label.y (Nur bei treemap) 161

chart.series.i[].itemStyle.emphasis.labelLine.length (Nur bei pie) 161

chart.series.i[].itemStyle.emphasis.labelLine.show (Nur bei pie) 161

chart.series.i[].itemStyle.emphasis.labelLine.lineStyle.color (Nur bei pie) 161

chart.series.i[].itemStyle.emphasis.labelLine.lineStyle.type (Nur bei pie) 162

chart.series.i[].itemStyle.emphasis.labelLine.lineStyle.width (Nur bei pie) 162

chart.series.i[].itemStyle.emphasis.labelLine.lineStyle.shadowColor (Nur bei pie) 162

chart.series.i[].itemStyle.emphasis.labelLine.lineStyle.shadowBlur (Nur bei pie) 162

chart.series.i[].itemStyle.emphasis.labelLine.lineStyle.shadowOffsetX (Nur bei pie) 162

chart.series.i[].itemStyle.emphasis.labelLine.lineStyle.shadowOffsetY (Nur bei pie) 162

chart.series.i[].itemStyle.emphasis.lineStyle.color (Nur bei line/area) 162

chart.series.i[].itemStyle.emphasis.lineStyle.type (Nur bei line/area) 162

chart.series.i[].itemStyle.emphasis.lineStyle.width (Nur bei line/area) 162

chart.series.i[].itemStyle.emphasis.lineStyle.shadowColor (Nur bei line/area) 163

chart.series.i[].itemStyle.emphasis.lineStyle.shadowBlur (Nur bei line/area) 163

chart.series.i[].itemStyle.emphasis.lineStyle.shadowOffsetX (Nur bei line/area) 163

chart.series.i[].itemStyle.emphasis.lineStyle.shadowOffsetY (Nur bei line/area) 163
chart.series.i[].itemStyle.emphasis.textStyle.fontFamily (Nur bei WorldCloud) 163
chart.series.i[].itemStyle.emphasis.textStyle.fontSize (Nur bei WorldCloud) 163
chart.series.i[].itemStyle.emphasis.textStyle.fontStyle (Nur bei WordlCloud) 163
chart.series.i[].itemStyle.emphasis.textStyle.fontWeight (Nur bei WorldCloud) 163
chart.series.i[].itemStyle.emphasis.areaStyle.color (Nur bei line/area, force (noch nicht implementiert), map) 163
chart.series.i[].itemStyle.emphasis.areaStyle.type (Nur bei line/area, force (noch nicht implementiert), map) 164
chart.series.i[].itemStyle.emphasis.barBorderColor (Nur bei bar) 164
chart.series.i[].itemStyle.emphasis.barBorderRadius (Nur bei bar) 164
chart.series.i[].itemStyle.emphasis.barBorderWidth (Nur bei bar) 164
chart.series.i[].itemStyle.emphasis.borderColor (Nur bei line/area, scatter, pie, map) 164
chart.series.i[].itemStyle.emphasis.borderWidth (Nur bei line/area, scatter, pie, map) 164
chart.series.i[].itemStyle.emphasis.chordStyle.width (Nur bei chord) 164
chart.series.i[].itemStyle.emphasis.chordStyle.color (Nur bei chord) 164
chart.series.i[].itemStyle.emphasis.chordStyle.borderWidth (Nur bei chord) 165
chart.series.i[].itemStyle.emphasis.chordStyle.borderColor (Nur bei chord) 165
chart.series.i[].itemStyle.emphasis.breadcrumb (Nur bei treemap (noch nicht implementiert)) 165
chart.series.i[].itemStyle.emphasis.breadcrumb.show (Nur bei treemap (noch nicht implementiert)) 165
chart.series.i[].itemStyle.emphasis.breadcrumb.textStyle (Nur bei treemap (noch nicht implementiert)) 165
chart.series.i[].itemStyle.emphasis.breadcrumb.textStyle.color (Nur bei treemap (noch nicht implementiert)) 165
chart.series.i[].itemStyle.emphasis.breadcrumb.textStyle.align (Nur bei treemap (noch nicht implementiert)) 165
chart.series.i[].itemStyle.emphasis.breadcrumb.textStyle.baseline (Nur bei treemap (noch nicht implementiert)) 165
chart.series.i[].itemStyle.emphasis.breadcrumb.textStyle.fontFamily (Nur bei treemap (noch nicht implementiert)) 166
chart.series.i[].itemStyle.emphasis.breadcrumb.textStyle.fontSize (Nur bei treemap (noch nicht implementiert)) 166
chart.series.i[].itemStyle.emphasis.breadcrumb.textStyle.fontStyle (Nur bei treemap (noch nicht implementiert)) 166
chart.series.i[].itemStyle.emphasis.breadcrumb.textStyle.fontWeight (Nur bei treemap (noch nicht implementiert)) 166
chart.series.i[].itemStyle.emphasis.childBorderWidth (Nur bei treemap (noch nicht implementiert)) 166
chart.series.i[].itemStyle.emphasis.childBorderColor (Nur bei treemap (noch nicht implementiert)) 166
chart.series.i[].itemStyle.emphasis.linkStyle.normal.type (Nur bei force (noch nicht implementiert)) 166

chart.series.i[].itemStyle.emphasis.linkStyle.normal.color (Nur bei force (noch nicht implementiert)) ... 167
chart.series.i[].itemStyle.emphasis.linkStyle.normal.width (Nur bei force (noch nicht implementiert)) .. 167
chart.series.i[].itemStyle.emphasis.linkStyle.emphasis (Nur bei force (noch nicht implementiert)) 167
chart.series.i[].itemStyle.emphasis.linkStyle.emphasis.type (Nur bei force (noch nicht implementiert)) .167
chart.series.i[].itemStyle.emphasis.linkStyle.emphasis.color (Nur bei force (noch nicht implementiert)) 167
chart.series.i[].itemStyle.emphasis.linkStyle.emphasis.width (Nur bei force (noch nicht implementiert)) 167
chart.series.i[].itemStyle.emphasis.nodeStyle.color (Nur bei force (noch nicht implementiert)) 167
chart.series.i[].itemStyle.emphasis.nodeStyle.borderColor (Nur bei force (noch nicht implementiert)) .. 167
chart.series.i[].itemStyle.emphasis.nodeStyle.borderWidth (Nur bei force (noch nicht implementiert)) . 168

3.25 Series-MarkLine 168

3.25.1 chart.series.i[].markLine.clickable 168
3.25.2 chart.series.i[].markLine.symbol 168
3.25.3 chart.series.i[].markLine.symbolSize 168
3.25.4 chart.series.i[].markLine.symbolRotate 168
3.25.5 chart.series.i[].markLine.large 169
3.25.6 chart.series.i[].markLine.smooth 169
3.25.7 chart.series.i[].markLine.smoothness 169
3.25.8 chart.series.i[].markLine.precision 169
3.25.9 chart.series.i[].markLine.bundling.enable 169
3.25.10 chart.series.i[].markLine.bundling.maxTurningAngle 169
3.25.11 chart.series.i[].markLine.effect.show 169
3.25.12 chart.series.i[].markLine.effect.loop 169
3.25.13 chart.series.i[].markLine.effect.period 170
3.25.14 chart.series.i[].markLine.effect.scaleSize 170
3.25.15 chart.series.i[].markLine.effect.color 170
3.25.16 chart.series.i[].markLine.effect.shadowColor 170
3.25.17 chart.series.i[].markLine.effect.shadowBlur 170
3.25.18 chart.series.i[].markLine.itemStyle.normal 170
3.25.19 chart.series.i[].markLine.itemStyle.normal.color 170
3.25.20 chart.series.i[].markLine.itemStyle.normal.lineStyle 170
3.25.21 chart.series.i[].markLine.itemStyle.normal.lineStyle.color 170
3.25.22 chart.series.i[].markLine.itemStyle.normal.lineStyle.type 171
3.25.23 chart.series.i[].markLine.itemStyle.normal.lineStyle.width 171
3.25.24 chart.series.i[].markLine.itemStyle.normal.lineStyle.shadowColor 171

3.25.25 chart.series.i[].markLine.itemStyle.normal.lineStyle.shadowBlur ... 171
3.25.26 chart.series.i[].markLine.itemStyle.normal.lineStyle.shadowOffsetX ... 171
3.25.27 chart.series.i[].markLine.itemStyle.normal.lineStyle.shadowOffsetY ... 171
3.25.28 chart.series.i[].markLine.itemStyle.normal.label ... 171
3.25.29 chart.series.i[].markLine.itemStyle.normal.label.show ... 171
3.25.30 chart.series.i[].markLine.itemStyle.normal.label.formatter ... 172
3.25.31 chart.series.i[].markLine.itemStyle.normal.label.textStyle ... 173
3.25.32 chart.series.i[].markLine.itemStyle.normal.label.textStyle.color ... 173
3.25.33 chart.series.i[].markLine.itemStyle.normal.label.textStyle.align ... 173
3.25.34 chart.series.i[].markLine.itemStyle.normal.label.textStyle.baseline ... 173
3.25.35 chart.series.i[].markLine.itemStyle.normal.label.textStyle.fontFamily ... 173
3.25.36 chart.series.i[].markLine.itemStyle.normal.label.textStyle.fontSize ... 173
3.25.37 chart.series.i[].markLine.itemStyle.normal.label.textStyle.fontStyle ... 173
3.25.38 chart.series.i[].markLine.itemStyle.normal.label.textStyle.fontWeight ... 173
3.25.39 chart.series.i[].markLine.itemStyle.emphasis ... 173
3.25.40 chart.series.i[].markLine.itemStyle.emphasis.color ... 174
3.25.41 chart.series.i[].markLine.itemStyle.emphasis.lineStyle ... 174
3.25.42 chart.series.i[].markLine.itemStyle.emphasis.lineStyle.color ... 174
3.25.43 chart.series.i[].markLine.itemStyle.emphasis.lineStyle.type ... 174
3.25.44 chart.series.i[].markLine.itemStyle.emphasis.lineStyle.width ... 174
3.25.45 chart.series.i[].markLine.itemStyle.emphasis.lineStyle.shadowColor ... 174
3.25.46 chart.series.i[].markLine.itemStyle.emphasis.lineStyle.shadowBlur ... 174
3.25.47 chart.series.i[].markLine.itemStyle.emphasis.lineStyle.shadowOffsetX ... 174
3.25.48 chart.series.i[].markLine.itemStyle.emphasis.lineStyle.shadowOffsetY ... 175
3.25.49 chart.series.i[].markLine.itemStyle.emphasis.label ... 175
3.25.50 chart.series.i[].markLine.itemStyle.emphasis.label.show ... 175
3.25.51 chart.series.i[].markLine.itemStyle.emphasis.label.formatter ... 175
3.25.52 chart.series.i[].markLine.itemStyle.emphasis.label.textStyle ... 176
3.25.53 chart.series.i[].markLine.itemStyle.emphasis.label.textStyle.color ... 176
3.25.54 chart.series.i[].markLine.itemStyle.emphasis.label.textStyle.align ... 176
3.25.55 chart.series.i[].markLine.itemStyle.emphasis.label.textStyle.baseline ... 176
3.25.56 chart.series.i[].markLine.itemStyle.emphasis.label.textStyle.fontFamily ... 176
3.25.57 chart.series.i[].markLine.itemStyle.emphasis.label.textStyle.fontSize ... 177
3.25.58 chart.series.i[].markLine.itemStyle.emphasis.label.textStyle.fontStyle ... 177
3.25.59 chart.series.i[].markLine.itemStyle.emphasis.label.textStyle.fontWeight ... 177
3.25.60 chart.series.i[].markLine.data ... 177

3.26 Series-MarkPoint 177
3.26.1 chart.series.i[].markPoint.clickable 177
3.26.2 chart.series.i[].markPoint.symbol 177
3.26.3 chart.series.i[].markPoint.symbolSize 177
3.26.4 chart.series.i[].markPoint.symbolRotate 178
3.26.5 chart.series.i[].markPoint.large 178
3.26.6 chart.series.i[].markPoint.effect.show 178
3.26.7 chart.series.i[].markPoint.effect.type 178
3.26.8 chart.series.i[].markPoint.effect.loop 178
3.26.9 chart.series.i[].markPoint.effect.period 178
3.26.10 chart.series.i[].markPoint.effect.scaleSize 178
3.26.11 chart.series.i[].markPoint.effect.bounceDistance 179
3.26.12 chart.series.i[].markPoint.effect.color 179
3.26.13 chart.series.i[].markPoint.effect.shadowColor 179
3.26.14 chart.series.i[].markPoint.effect.shadowBlur 179
3.26.15 chart.series.i[].markPoint.itemStyle.normal.color 179
3.26.16 chart.series.i[].markPoint.itemStyle.normal.borderColor 179
3.26.17 chart.series.i[].markPoint.itemStyle.normal.borderWidth 179
3.26.18 chart.series.i[].markPoint.itemStyle.normal.label.show 179
3.26.19 chart.series.i[].markPoint.itemStyle.normal.label.formatter 180
3.26.20 chart.series.i[].markPoint.itemStyle.normal.label.textStyle.color 181
3.26.21 chart.series.i[].markPoint.itemStyle.normal.label.textStyle.align 181
3.26.22 chart.series.i[].markPoint.itemStyle.normal.label.textStyle.baseline 181
3.26.23 chart.series.i[].markPoint.itemStyle.normal.label.textStyle.fontFamily 181
3.26.24 chart.series.i[].markPoint.itemStyle.normal.label.textStyle.fontSize 181
3.26.25 chart.series.i[].markPoint.itemStyle.normal.label.textStyle.fontStyle 181
3.26.26 chart.series.i[].markPoint.itemStyle.normal.label.textStyle.fontWeight 181
3.26.27 chart.series.i[].markPoint.itemStyle.emphasis.color 181
3.26.28 chart.series.i[].markPoint.itemStyle.emphasis.borderColor 182
3.26.29 chart.series.i[].markPoint.itemStyle.emphasis.borderWidth 182
3.26.30 chart.series.i[].markPoint.itemStyle.emphasis.label.show 182
3.26.31 chart.series.i[].markPoint.itemStyle.emphasis.label.formatter 182
3.26.32 chart.series.i[].markPoint.itemStyle.emphasis.label.textStyle.color 183
3.26.33 chart.series.i[].markPoint.itemStyle.emphasis.label.textStyle.align 183
3.26.34 chart.series.i[].markPoint.itemStyle.emphasis.label.textStyle.baseline 183
3.26.35 chart.series.i[].markPoint.itemStyle.emphasis.label.textStyle.fontFamily 183

3.26.36 chart.series.i[].markPoint.itemStyle.emphasis.label.textStyle.fontSize ... 183
3.26.37 chart.series.i[].markPoint.itemStyle.emphasis.label.textStyle.fontStyle ... 184
3.26.38 chart.series.i[].markPoint.itemStyle.emphasis.label.textStyle.fontWeight ... 184
3.26.39 chart.series.i[].markPoint.data ... 184
3.27 Series-Tooltip ... 184
3.27.1 chart.series.i[].tooltip.show ... 184
3.27.2 chart.series.i[].tooltip.zlevel ... 184
3.27.3 chart.series.i[].tooltip.z ... 184
3.27.4 chart.series.i[].tooltip.showContent ... 184
3.27.5 chart.series.i[].tooltip.trigger ... 185
3.27.6 chart.series.i[].tooltip.position ... 185
3.27.7 chart.series.i[].tooltip.formatter ... 185
3.27.8 chart.series.i[].tooltip.islandFormatter ... 186
3.27.9 chart.series.i[].tooltip.showDelay ... 186
3.27.10 chart.series.i[].tooltip.hideDelay ... 186
3.27.11 chart.series.i[].tooltip.transitionDuration ... 186
3.27.12 chart.series.i[].tooltip.enterable ... 186
3.27.13 chart.series.i[].tooltip.backgroundColor ... 186
3.27.14 chart.series.i[].tooltip.borderColor ... 187
3.27.15 chart.series.i[].tooltip.borderRadius ... 187
3.27.16 chart.series.i[].tooltip.borderWidth ... 187
3.27.17 chart.series.i[].tooltip.padding ... 187
3.27.18 chart.series.i[].tooltip.axisPointer.type ... 187
3.27.19 chart.series.i[].tooltip.axisPointer.lineStyle.color ... 187
3.27.20 chart.series.i[].tooltip.axisPointer.lineStyle.type ... 187
3.27.21 chart.series.i[].tooltip.axisPointer.lineStyle.width ... 187
3.27.22 chart.series.i[].tooltip.axisPointer.lineStyle.shadowColor ... 188
3.27.23 chart.series.i[].tooltip.axisPointer.lineStyle.shadowBlur ... 188
3.27.24 chart.series.i[].tooltip.axisPointer.lineStyle.shadowOffsetX ... 188
3.27.25 chart.series.i[].tooltip.axisPointer.lineStyle.shadowOffsetY ... 188
3.27.26 chart.series.i[].tooltip.axisPointer.crossStyle.color ... 188
3.27.27 chart.series.i[].tooltip.axisPointer.crossStyle.type ... 188
3.27.28 chart.series.i[].tooltip.axisPointer.crossStyle.width ... 188
3.27.29 chart.series.i[].tooltip.axisPointer.crossStyle.shadowColor ... 188
3.27.30 chart.series.i[].tooltip.axisPointer.crossStyle.shadowBlur ... 189
3.27.31 chart.series.i[].tooltip.axisPointer.crossStyle.shadowOffsetX ... 189

3.27.32 chart.series.i[].tooltip.axisPointer.crossStyle.shadowOffsetY ... 189
3.27.33 chart.series.i[].tooltip.axisPointer.shadowStyle.color ... 189
3.27.34 chart.series.i[].tooltip.axisPointer.shadowStyle.type ... 189
3.27.35 chart.series.i[].tooltip.axisPointer.textStyle.color ... 189
3.27.36 chart.series.i[].tooltip.axisPointer.textStyle.decoration ... 189
3.27.37 chart.series.i[].tooltip.axisPointer.textStyle.align ... 189
3.27.38 chart.series.i[].tooltip.axisPointer.textStyle.baseline ... 189
3.27.39 chart.series.i[].tooltip.axisPointer.textStyle.fontFamily ... 190
3.27.40 chart.series.i[].tooltip.axisPointer.textStyle.fontSize ... 190
3.27.41 chart.series.i[].tooltip.axisPointer.textStyle.fontStyle ... 190
3.27.42 chart.series.i[].tooltip.axisPointer.textStyle.fontWeight ... 190
3.27.43 chart.series.i[].tooltip.textStyle.color ... 190
3.27.44 chart.series.i[].tooltip.textStyle.decoration ... 190
3.27.45 chart.series.i[].tooltip.textStyle.align ... 190
3.27.46 chart.series.i[].tooltip.textStyle.baseline ... 190
3.27.47 chart.series.i[].tooltip.textStyle.fontFamily ... 191
3.27.48 chart.series.i[].tooltip.textStyle.fontSize ... 191
3.27.49 chart.series.i[].tooltip.textStyle.fontStyle ... 191
3.27.50 chart.series.i[].tooltip.textStyle.fontWeight ... 191
3.27.51 chart.series.i[].tooltip.textStyle.textStyle.fontWeight ... 191
3.28 Series-WorldCloud ... 191
3.28.1 chart.series.i[].autoSize.enable ... 191
3.28.2 chart.series.i[].autoSize.minSize ... 191
3.28.3 chart.series.i[].size (Auch bei force, treemap) ... 191
3.28.4 chart.series.i[].textRotation (Noch nicht implementiert) ... 191
3.28.5 chart.series.i[].center (Auch bei treemap, gauge, pie, force) ... 192
3.29 xAxis-Optionen ... 192
3.29.1 Chart-xAxis-Optionen ... 192
chart.xAxis.[i].type ... 192
chart.xAxis.[i].show ... 192
chart.xAxis.[i].zlevel ... 192
chart.xAxis.[i].z ... 192
chart.xAxis.[i].position ... 192
chart.xAxis.[i].name ... 192
chart.xAxis.[i].nameLocation ... 193

chart.xAxis.[i].nameTextStyle ... 193
chart.xAxis.[i].axisLabel.textStyle ... 193
chart.xAxis.[i].axisLabel.textStyle.color ... 193
chart.xAxis.[i].axisLabel.textStyle.align ... 193
chart.xAxis.[i].axisLabel.textStyle.baseline ... 193
chart.xAxis.[i].axisLabel.textStyle.fontFamily ... 193
chart.xAxis.[i].axisLabel.textStyle.fontSize ... 193
chart.xAxis.[i].axisLabel.textStyle.fontStyle ... 193
chart.xAxis.[i].axisLabel.textStyle.fontWeight ... 194
chart.xAxis.[i].boundaryGap ... 194
chart.xAxis.[i].min ... 194
chart.xAxis.[i].max ... 194
chart.xAxis.[i].scale ... 194
chart.xAxis.[i].splitNumber ... 194
chart.xAxis.[i].logLabelBase ... 194
chart.xAxis.[i].logPositive ... 194
chart.xAxis.[i].axisLine.show ... 195
chart.xAxis.[i].axisLine.onZero ... 195
chart.xAxis.[i].axisLine.lineStyle.color ... 195
chart.xAxis.[i].axisLine.lineStyle.type ... 195
chart.xAxis.[i].axisLine.lineStyle.width ... 195
chart.xAxis.[i].axisLine.lineStyle.shadowColor ... 195
chart.xAxis.[i].axisLine.lineStyle.shadowBlur ... 195
chart.xAxis.[i].axisLine.lineStyle.shadowOffsetX ... 195
chart.xAxis.[i].axisLine.lineStyle.shadowOffsetY ... 195
chart.xAxis.[i].axisTick.show ... 196
chart.xAxis.[i].axisTick.interval ... 196
chart.xAxis.[i].axisTick.onGap ... 196
chart.xAxis.[i].axisTick.inside ... 196
chart.xAxis.[i].axisTick.length ... 196
chart.xAxis.[i].axisTick.lineStyle.color ... 196
chart.xAxis.[i].axisTick.lineStyle.type ... 196
chart.xAxis.[i].axisTick.lineStyle.width ... 196
chart.xAxis.[i].axisTick.lineStyle.shadowColor ... 197
chart.xAxis.[i].axisTick.lineStyle.shadowBlur ... 197
chart.xAxis.[i].axisTick.lineStyle.shadowOffsetX ... 197

chart.xAxis.[i].axisTick.lineStyle.shadowOffsetY ... 197
chart.xAxis.[i].axisLabel.show ... 197
chart.xAxis.[i].axisLabel.interval ... 197
chart.xAxis.[i].axisLabel.rotate ... 197
chart.xAxis.[i].axisLabel.margin ... 197
chart.xAxis.[i].axisLabel.clickable ... 198
chart.xAxis.[i].axisLabel.formatter ... 198
chart.xAxis.[i].axisLabel.textStyle.color ... 198
chart.xAxis.[i].axisLabel.textStyle.align ... 198
chart.xAxis.[i].axisLabel.textStyle.baseline ... 198
chart.xAxis.[i].axisLabel.textStyle.fontFamily ... 198
chart.xAxis.[i].axisLabel.textStyle.fontSize ... 198
chart.xAxis.[i].axisLabel.textStyle.fontStyle ... 198
chart.xAxis.[i].axisLabel.textStyle.fontWeight ... 198
chart.xAxis.[i].splitLine.show ... 199
chart.xAxis.[i].splitLine.onGap ... 199
chart.xAxis.[i].splitLine.lineStyle.color ... 199
chart.xAxis.[i].splitLine.lineStyle.type ... 199
chart.xAxis.[i].splitLine.lineStyle.width ... 199
chart.xAxis.[i].splitLine.lineStyle.shadowColor ... 199
chart.xAxis.[i].splitLine.lineStyle.shadowBlur ... 199
chart.xAxis.[i].splitLine.lineStyle.shadowOffsetX ... 199
chart.xAxis.[i].splitLine.lineStyle.shadowOffsetY ... 199
chart.xAxis.[i].splitArea.show ... 200
chart.xAxis.[i].splitArea.onGap ... 200
chart.xAxis.[i].splitArea.areaStyle.color ... 200
chart.xAxis.[i].splitArea.areaStyle.type ... 200
3.30 yAxis-Optionen ... 200
3.30.1 Chart-yAxis-Optionen ... 200
chart.yAxis.[i].type ... 200
chart.yAxis.[i].show ... 200
chart.yAxis.[i].zlevel ... 200
chart.yAxis.[i].z ... 200
chart.yAxis.[i].position ... 201
chart.yAxis.[i].name ... 201
chart.yAxis.[i].nameLocation ... 201

chart.yAxis.[i].nameTextStyle 201
chart.yAxis.[i].axisLabel.textStyle 201
chart.yAxis.[i].axisLabel.textStyle.color 201
chart.yAxis.[i].axisLabel.textStyle.align 201
chart.yAxis.[i].axisLabel.textStyle.baseline 201
chart.yAxis.[i].axisLabel.textStyle.fontFamily 202
chart.yAxis.[i].axisLabel.textStyle.fontSize 202
chart.yAxis.[i].axisLabel.textStyle.fontStyle 202
chart.yAxis.[i].axisLabel.textStyle.fontWeight 202
chart.yAxis.[i].boundaryGap 202
chart.yAxis.[i].min 202
chart.yAxis.[i].max 202
chart.yAxis.[i].scale 202
chart.yAxis.[i].splitNumber 203
chart.yAxis.[i].logLabelBase 203
chart.yAxis.[i].logPositive 203
chart.yAxis.[i].axisLine.show 203
chart.yAxis.[i].axisLine.onZero 203
chart.yAxis.[i].axisLine.lineStyle.color 203
chart.yAxis.[i].axisLine.lineStyle.type 203
chart.yAxis.[i].axisLine.lineStyle.width 203
chart.yAxis.[i].axisLine.lineStyle.shadowColor 204
chart.yAxis.[i].axisLine.lineStyle.shadowBlur 204
chart.yAxis.[i].axisLine.lineStyle.shadowOffsetX 204
chart.yAxis.[i].axisLine.lineStyle.shadowOffsetY 204
chart.yAxis.[i].axisTick.show 204
chart.yAxis.[i].axisTick.interval 204
chart.yAxis.[i].axisTick.onGap 204
chart.yAxis.[i].axisTick.inside 204
chart.yAxis.[i].axisTick.length 205
chart.yAxis.[i].axisTick.lineStyle.color 205
chart.yAxis.[i].axisTick.lineStyle.type 205
chart.yAxis.[i].axisTick.lineStyle.width 205
chart.yAxis.[i].axisTick.lineStyle.shadowColor 205
chart.yAxis.[i].axisTick.lineStyle.shadowBlur 205
chart.yAxis.[i].axisTick.lineStyle.shadowOffsetX 205

chart.yAxis.[i].axisTick.lineStyle.shadowOffsetY ... 205
chart.yAxis.[i].axisLabel.show ... 205
chart.yAxis.[i].axisLabel.interval ... 206
chart.yAxis.[i].axisLabel.rotate ... 206
chart.yAxis.[i].axisLabel.margin ... 206
chart.yAxis.[i].axisLabel.clickable ... 206
chart.yAxis.[i].axisLabel.formatter ... 206
chart.yAxis.[i].axisLabel.textStyle.color ... 206
chart.yAxis.[i].axisLabel.textStyle.align ... 206
chart.yAxis.[i].axisLabel.textStyle.baseline ... 206
chart.yAxis.[i].axisLabel.textStyle.fontFamily ... 207
chart.yAxis.[i].axisLabel.textStyle.fontSize ... 207
chart.yAxis.[i].axisLabel.textStyle.fontStyle ... 207
chart.yAxis.[i].axisLabel.textStyle.fontWeight ... 207
chart.yAxis.[i].splitLine.show ... 207
chart.yAxis.[i].splitLine.onGap ... 207
chart.yAxis.[i].splitLine.lineStyle.color ... 207
chart.yAxis.[i].splitLine.lineStyle.type ... 207
chart.yAxis.[i].splitLine.lineStyle.width ... 207
chart.yAxis.[i].splitLine.lineStyle.shadowColor ... 208
chart.yAxis.[i].splitLine.lineStyle.shadowBlur ... 208
chart.yAxis.[i].splitLine.lineStyle.shadowOffsetX ... 208
chart.yAxis.[i].splitLine.lineStyle.shadowOffsetY ... 208
chart.yAxis.[i].splitArea.show ... 208
chart.yAxis.[i].splitArea.onGap ... 208
chart.yAxis.[i].splitArea.areaStyle.color ... 208
chart.yAxis.[i].splitArea.areaStyle.type ... 208
3.31 Series-Force ... 209
3.31.1 chart.series.i[].data.name (Noch nicht implementiert) ... 209
3.31.2 chart.series.i[].data.label (Noch nicht implementiert) ... 209
3.31.3 chart.series.i[].data.value (Noch nicht implementiert) ... 209
3.31.4 chart.series.i[].data.ignore (Noch nicht implementiert) ... 209
3.31.5 chart.series.i[].data.symbol (Noch nicht implementiert) ... 209
3.31.6 chart.series.i[].data.symbolSize (Noch nicht implementiert) ... 209
3.31.7 chart.series.i[].data.category (Noch nicht implementiert) ... 209
3.31.8 chart.series.i[].data.itemStyle (Noch nicht implementiert) ... 209

3.31.9 chart.series.i[].data.itemStyle.normal (Noch nicht implementiert) ... 209
3.31.10 chart.series.i[].data.itemStyle.normal.color (Noch nicht implementiert) ... 210
3.31.11 chart.series.i[].data.itemStyle.normal.nodeStyle (Noch nicht implementiert) ... 210
3.31.12 chart.series.i[].data.itemStyle.normal.nodeStyle.color (Noch nicht implementiert) ... 210
3.31.13 chart.series.i[].data.itemStyle.normal.nodeStyle.borderColor (Noch nicht implementiert) ... 210
3.31.14 chart.series.i[].data.itemStyle.normal.nodeStyle.borderWidth (Noch nicht implementiert) ... 210
3.31.15 chart.series.i[].data.itemStyle.normal.linkStyle (Noch nicht implementiert) ... 210
3.31.16 chart.series.i[].data.itemStyle.normal.linkStyle.normal (Noch nicht implementiert) ... 210
3.31.17 chart.series.i[].data.itemStyle.normal.linkStyle.normal.type (Noch nicht implementiert) ... 211
3.31.18 chart.series.i[].data.itemStyle.normal.linkStyle.normal.color (Noch nicht implementiert) ... 211
3.31.19 chart.series.i[].data.itemStyle.normal.linkStyle.normal.width (Noch nicht implementiert) ... 211
3.31.20 chart.series.i[].data.itemStyle.normal.linkStyle.emphasis (Noch nicht implementiert) ... 211
3.31.21 chart.series.i[].data.itemStyle.normal.linkStyle.emphasis.type (Noch nicht implementiert) ... 211
3.31.22 chart.series.i[].data.itemStyle.normal.linkStyle.emphasis.color (Noch nicht implementiert) ... 211
3.31.23 chart.series.i[].data.itemStyle.normal.linkStyle.emphasis.width (Noch nicht implementiert) ... 211
3.31.24 chart.series.i[].data.itemStyle.normal.label (Noch nicht implementiert) ... 212
3.31.25 chart.series.i[].data.itemStyle.normal.label.show (Noch nicht implementiert) ... 212
3.31.26 chart.series.i[].data.itemStyle.normal.label.formatter (Noch nicht implementiert) ... 212
3.31.27 chart.series.i[].data.itemStyle.normal.label.textStyle (Noch nicht implementiert) ... 213
3.31.28 chart.series.i[].data.itemStyle.normal.label.textStyle.color (Noch nicht implementiert) ... 213
3.31.29 chart.series.i[].data.itemStyle.normal.label.textStyle.align (Noch nicht implementiert) ... 213
3.31.30 chart.series.i[].data.itemStyle.normal.label.textStyle.baseline (Noch nicht implementiert) ... 213
3.31.31 chart.series.i[].data.itemStyle.normal.label.textStyle.fontFamily (Noch nicht implementiert) ... 213
3.31.32 chart.series.i[].data.itemStyle.normal.label.textStyle.fontSize (Noch nicht implementiert) ... 214
3.31.33 chart.series.i[].data.itemStyle.normal.label.textStyle.fontStyle (Noch nicht implementiert) ... 214
3.31.34 chart.series.i[].data.itemStyle.normal.label.textStyle.fontWeight (Noch nicht implementiert) ... 214
3.31.35 chart.series.i[].data.itemStyle.emphasis (Noch nicht implementiert) ... 214
3.31.36 chart.series.i[].data.itemStyle.emphasis.color (Noch nicht implementiert) ... 214
3.31.37 chart.series.i[].data.itemStyle.emphasis.nodeStyle (Noch nicht implementiert) ... 214
3.31.38 chart.series.i[].data.itemStyle.emphasis.nodeStyle.color (Noch nicht implementiert) ... 214
3.31.39 chart.series.i[].data.itemStyle.emphasis.nodeStyle.borderColor (Noch nicht implementiert) ... 215
3.31.40 chart.series.i[].data.itemStyle.emphasis.nodeStyle.borderWidth (Noch nicht implementiert) ... 215
3.31.41 chart.series.i[].data.itemStyle.emphasis.linkStyle (Noch nicht implementiert) ... 215
3.31.42 chart.series.i[].data.itemStyle.emphasis.linkStyle.normal (Noch nicht implementiert) ... 215
3.31.43 chart.series.i[].data.itemStyle.emphasis.linkStyle.normal.type (Noch nicht implementiert) ... 215
3.31.44 chart.series.i[].data.itemStyle.emphasis.linkStyle.normal.color (Noch nicht implementiert) ... 215

3.31.45 chart.series.i[].data.itemStyle.emphasis.linkStyle.normal.width (Noch nicht implementiert) 215
3.31.46 chart.series.i[].data.itemStyle.emphasis.linkStyle.emphasis (Noch nicht implementiert) 216
3.31.47 chart.series.i[].data.itemStyle.emphasis.linkStyle.emphasis.type (Noch nicht implementiert) 216
3.31.48 chart.series.i[].data.itemStyle.emphasis.linkStyle.emphasis.color (Noch nicht implementiert) 216
3.31.49 chart.series.i[].data.itemStyle.emphasis.linkStyle.emphasis.width (Noch nicht implementiert) 216
3.31.50 chart.series.i[].data.itemStyle.emphasis.label (Noch nicht implementiert) 216
3.31.51 chart.series.i[].data.itemStyle.emphasis.label.show (Noch nicht implementiert) 216
3.31.52 chart.series.i[].data.itemStyle.emphasis.label.formatter (Noch nicht implementiert) 216
3.31.53 chart.series.i[].data.itemStyle.emphasis.label.textStyle (Noch nicht implementiert) 217
3.31.54 chart.series.i[].data.itemStyle.emphasis.label.textStyle.color (Noch nicht implementiert) 218
3.31.55 chart.series.i[].data.itemStyle.emphasis.label.textStyle.align (Noch nicht implementiert) 218
3.31.56 chart.series.i[].data.itemStyle.emphasis.label.textStyle.baseline (Noch nicht implementiert) 218
3.31.57 chart.series.i[].data.itemStyle.emphasis.label.textStyle.fontFamily (Noch nicht implementiert) 218
3.31.58 chart.series.i[].data.itemStyle.emphasis.label.textStyle.fontSize (Noch nicht implementiert) 218
3.31.59 chart.series.i[].data.itemStyle.emphasis.label.textStyle.fontStyle (Noch nicht implementiert) 218
3.31.60 chart.series.i[].data.itemStyle.emphasis.label.textStyle.fontWeight (Noch nicht implementiert) 218
3.31.61 chart.series.i[].data.initial (Noch nicht implementiert) 219
3.31.62 chart.series.i[].data.fixX (Noch nicht implementiert) 219
3.31.63 chart.series.i[].data.fixY (Noch nicht implementiert) 219
3.31.64 chart.series.i[].data.draggable (Noch nicht implementiert) 219
3.31.65 chart.series.i[].symbolSize (Auch bei line/area, scatter, radar, chord, tree) 219
3.31.66 chart.series.i[].large (Auch bei scatter) 219
3.31.67 chart.series.i[].center (Auch bei treemap, gauge, pie, worldcloud) 219
3.31.68 chart.series.i[].roam (Auch bei map, tree) 219
3.31.69 chart.series.i[].categories.name (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) 220
3.31.70 chart.series.i[].categories.symbol (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) 220
3.31.71 chart.series.i[].categories.symbolSize (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) 220
3.31.72 chart.series.i[].categories.itemStyle (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) 220
3.31.73 chart.series.i[].categories.itemStyle.normal (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) 220
3.31.74 chart.series.i[].categories.itemStyle.normal.color (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) 220
3.31.75 chart.series.i[].categories.itemStyle.normal.nodeStyle (Noch nicht implementiert) 221
3.31.76 chart.series.i[].categories.itemStyle.normal.nodeStyle.color (Noch nicht implementiert) 221

3.31.77 chart.series.i[].categories.itemStyle.normal.nodeStyle.borderColor (Noch nicht implementiert) 221
3.31.78 chart.series.i[].categories.itemStyle.normal.nodeStyle.borderWidth (Noch nicht implementiert) 221
3.31.79 chart.series.i[].categories.itemStyle.normal.linkStyle (Noch nicht implementiert) 221
3.31.80 chart.series.i[].categories.itemStyle.normal.linkStyle.normal (Noch nicht implementiert) 221
3.31.81 chart.series.i[].categories.itemStyle.normal.linkStyle.normal.type (Noch nicht implementiert) 221
3.31.82 chart.series.i[].categories.itemStyle.normal.linkStyle.normal.color (Noch nicht implementiert) 222
3.31.83 chart.series.i[].categories.itemStyle.normal.linkStyle.normal.width (Noch nicht implementiert) 222
3.31.84 chart.series.i[].categories.itemStyle.normal.linkStyle.emphasis (Noch nicht implementiert) 222
3.31.85 chart.series.i[].categories.itemStyle.normal.linkStyle.emphasis.type (Noch nicht implementiert) 222
3.31.86 chart.series.i[].categories.itemStyle.normal.linkStyle.emphasis.color (Noch nicht implementiert) 222
3.31.87 chart.series.i[].categories.itemStyle.normal.linkStyle.emphasis.width (Noch nicht implementiert) 222
3.31.88 chart.series.i[].categories.itemStyle.normal.label (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) 222
3.31.89 chart.series.i[].categories.itemStyle.normal.label.show (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) 223
3.31.90 chart.series.i[].categories.itemStyle.normal.label.rotate (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) 223
3.31.91 chart.series.i[].categories.itemStyle.normal.label.distance (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) 223
3.31.92 chart.series.i[].categories.itemStyle.normal.label.formatter (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) 223
3.31.93 chart.series.i[].categories.itemStyle.normal.label.textStyle (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) 224
3.31.94 chart.series.i[].categories.itemStyle.normal.label.textStyle.color (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) 224
3.31.95 chart.series.i[].categories.itemStyle.normal.label.textStyle.align (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) 224
3.31.96 chart.series.i[].categories.itemStyle.normal.label.textStyle.baseline (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) 224
3.31.97 chart.series.i[].categories.itemStyle.normal.label.textStyle.fontFamily (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) 225
3.31.98 chart.series.i[].categories.itemStyle.normal.label.textStyle.fontSize (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) 225
3.31.99 chart.series.i[].categories.itemStyle.normal.label.textStyle.fontStyle (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) 225
3.31.100 chart.series.i[].categories.itemStyle.normal.label.textStyle.fontWeight (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) 225
3.31.101 chart.series.i[].categories.itemStyle.emphasis (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) 225
3.31.102 chart.series.i[].categories.itemStyle.emphasis.color (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) 225

3.31.103 chart.series.i[].categories.itemStyle.emphasis.nodeStyle (Noch nicht implementiert) ... 225
3.31.104 chart.series.i[].categories.itemStyle.emphasis.nodeStyle.color (Noch nicht implementiert) ... 226
3.31.105 chart.series.i[].categories.itemStyle.emphasis.nodeStyle.borderColor (Noch nicht implementiert) ... 226
3.31.106 chart.series.i[].categories.itemStyle.emphasis.nodeStyle.borderWidth (Noch nicht implementiert) ... 226
3.31.107 chart.series.i[].categories.itemStyle.emphasis.linkStyle (Noch nicht implementiert) ... 226
3.31.108 chart.series.i[].categories.itemStyle.emphasis.linkStyle.normal (Noch nicht implementiert) ... 226
3.31.109 chart.series.i[].categories.itemStyle.emphasis.linkStyle.normal.type (Noch nicht implementiert) ... 226
3.31.110 chart.series.i[].categories.itemStyle.emphasis.linkStyle.normal.color (Noch nicht implementiert) ... 226
3.31.111 chart.series.i[].categories.itemStyle.emphasis.linkStyle.normal.width (Noch nicht implementiert) ... 227
3.31.112 chart.series.i[].categories.itemStyle.emphasis.linkStyle.emphasis (Noch nicht implementiert) ... 227
3.31.113 chart.series.i[].categories.itemStyle.emphasis.linkStyle.emphasis.type (Noch nicht implementiert) ... 227
3.31.114 chart.series.i[].categories.itemStyle.emphasis.linkStyle.emphasis.color (Noch nicht implementiert) ... 227
3.31.115 chart.series.i[].categories.itemStyle.emphasis.linkStyle.emphasis.width (Noch nicht implementiert) ... 227
3.31.116 chart.series.i[].categories.itemStyle.emphasis.label (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 227
3.31.117 chart.series.i[].categories.itemStyle.emphasis.label.show (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 228
3.31.118 chart.series.i[].categories.itemStyle.emphasis.label.rotate (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 228
3.31.119 chart.series.i[].categories.itemStyle.emphasis.label.distance (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 228
3.31.120 chart.series.i[].categories.itemStyle.emphasis.label.formatter (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 228
3.31.121 chart.series.i[].categories.itemStyle.emphasis.label.textStyle (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 229
3.31.122 chart.series.i[].categories.itemStyle.emphasis.label.textStyle.color (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 229
3.31.123 chart.series.i[].categories.itemStyle.emphasis.label.textStyle.align (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 230
3.31.124 chart.series.i[].categories.itemStyle.emphasis.label.textStyle.baseline (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 230
3.31.125 chart.series.i[].categories.itemStyle.emphasis.label.textStyle.fontFamily (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 230
3.31.126 chart.series.i[].categories.itemStyle.emphasis.label.textStyle.fontSize (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 230
3.31.127 chart.series.i[].categories.itemStyle.emphasis.label.textStyle.fontStyle (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 230
3.31.128 chart.series.i[].categories.itemStyle.emphasis.label.textStyle.fontWeight (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 230
3.31.129 chart.series.i[].links.source (Noch nicht implementiert) ... 230

3.31.130 chart.series.i[].links.target (Noch nicht implementiert) ... 231
3.31.131 chart.series.i[].links.weight (Noch nicht implementiert) ... 231
3.31.132 chart.series.i[].links.itemStyle (Noch nicht implementiert) ... 231
3.31.133 chart.series.i[].links.itemStyle.normal (Noch nicht implementiert) ... 231
3.31.134 chart.series.i[].links.itemStyle.normal.type (Noch nicht implementiert) ... 231
3.31.135 chart.series.i[].links.itemStyle.normal.color (Noch nicht implementiert) ... 231
3.31.136 chart.series.i[].links.itemStyle.normal.width (Noch nicht implementiert) ... 231
3.31.137 chart.series.i[].links.itemStyle.emphasis (Noch nicht implementiert) ... 231
3.31.138 chart.series.i[].links.itemStyle.emphasis.type (Noch nicht implementiert) ... 231
3.31.139 chart.series.i[].links.itemStyle.emphasis.color (Noch nicht implementiert) ... 232
3.31.140 chart.series.i[].links.itemStyle.emphasis.width (Noch nicht implementiert) ... 232
3.31.141 chart.series.i[].matrix (Auch bei chord) ... 232
3.31.142 chart.series.i[].size (Auch bei treemap, worldcloud) ... 232
3.31.143 chart.series.i[].minRadius (Auch bei chord) ... 232
3.31.144 chart.series.i[].maxRadius (Auch bei chord) ... 232
3.31.145 chart.series.i[].linkSymbol ... 232
3.31.146 chart.series.i[].linkSymbolSize ... 232
3.31.147 chart.series.i[].scaling ... 233
3.31.148 chart.series.i[].gravity ... 233
3.31.149 chart.series.i[].draggable ... 233
3.31.150 chart.series.i[].useWorker ... 233
3.31.151 chart.series.i[].steps ... 233
3.31.152 chart.series.i[].nodes (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 233
3.31.153 chart.series.i[].nodes.name (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 233
3.31.154 chart.series.i[].nodes.label (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 233
3.31.155 chart.series.i[].nodes.value (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 234
3.31.156 chart.series.i[].nodes.ignore (Noch nicht implementiert) ... 234
3.31.157 chart.series.i[].nodes.symbol (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 234
3.31.158 chart.series.i[].nodes.symbolSize (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 234
3.31.159 chart.series.i[].nodes.category (Noch nicht implementiert) ... 234
3.31.160 chart.series.i[].nodes.itemStyle (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 234
3.31.161 chart.series.i[].nodes.itemStyle.normal (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 234
3.31.162 chart.series.i[].nodes.itemStyle.normal.color (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 234

3.31.163 chart.series.i[].nodes.itemStyle.normal.nodeStyle (Noch nicht implementiert) ... 235
3.31.164 chart.series.i[].nodes.itemStyle.normal.nodeStyle.color (Noch nicht implementiert) ... 235
3.31.165 chart.series.i[].nodes.itemStyle.normal.nodeStyle.borderColor (Noch nicht implementiert) ... 235
3.31.166 chart.series.i[].nodes.itemStyle.normal.nodeStyle.borderWidth (Noch nicht implementiert) ... 235
3.31.167 chart.series.i[].nodes.itemStyle.normal.linkStyle (Noch nicht implementiert) ... 235
3.31.168 chart.series.i[].nodes.itemStyle.normal.linkStyle.normal (Noch nicht implementiert) ... 235
3.31.169 chart.series.i[].nodes.itemStyle.normal.linkStyle.normal.type (Noch nicht implementiert) ... 235
3.31.170 chart.series.i[].nodes.itemStyle.normal.linkStyle.normal.color (Noch nicht implementiert) ... 236
3.31.171 chart.series.i[].nodes.itemStyle.normal.linkStyle.normal.width (Noch nicht implementiert) ... 236
3.31.172 chart.series.i[].nodes.itemStyle.normal.linkStyle.emphasis (Noch nicht implementiert) ... 236
3.31.173 chart.series.i[].nodes.itemStyle.normal.linkStyle.emphasis.type (Noch nicht implementiert) ... 236
3.31.174 chart.series.i[].nodes.itemStyle.normal.linkStyle.emphasis.color (Noch nicht implementiert) ... 236
3.31.175 chart.series.i[].nodes.itemStyle.normal.linkStyle.emphasis.width (Noch nicht implementiert) ... 236
3.31.176 chart.series.i[].nodes.itemStyle.normal.label (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 236
3.31.177 chart.series.i[].nodes.itemStyle.normal.label.show (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 237
3.31.178 chart.series.i[].nodes.itemStyle.normal.label.rotate (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 237
3.31.179 chart.series.i[].nodes.itemStyle.normal.label.distance (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 237
3.31.180 chart.series.i[].nodes.itemStyle.normal.label.formatter (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 237
3.31.181 chart.series.i[].nodes.itemStyle.noraml.label.textStyle (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 238
3.31.182 chart.series.i[].nodes.itemStyle.normal.label.textStyle.color (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 238
3.31.183 chart.series.i[].nodes.itemStyle.normal.label.textStyle.align (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 238
3.31.184 chart.series.i[].nodes.itemStyle.normal.label.textStyle.baseline (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 238
3.31.185 chart.series.i[].nodes.itemStyle.normal.label.textStyle.fontFamily (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 239
3.31.186 chart.series.i[].nodes.itemStyle.normal.label.textStyle.fontSize (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 239
3.31.187 chart.series.i[].nodes.itemStyle.normal.label.textStyle.fontStyle (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 239
3.31.188 chart.series.i[].nodes.itemStyle.normal.label.textStyle.fontWeight (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 239

3.31.189 chart.series.i[].nodes.itemStyle.emphasis (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 239
3.31.190 chart.series.i[].nodes.itemStyle.emphasis.color (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 239
3.31.191 chart.series.i[].nodes.itemStyle.emphasis.nodeStyle (Noch nicht implementiert) ... 239
3.31.192 chart.series.i[].nodes.itemStyle.emphasis.nodeStyle.color (Noch nicht implementiert) ... 240
3.31.193 chart.series.i[].nodes.itemStyle.emphasis.nodeStyle.borderColor (Noch nicht implementiert) ... 240
3.31.194 chart.series.i[].nodes.itemStyle.emphasis.nodeStyle.borderWidth (Noch nicht implementiert) ... 240
3.31.195 chart.series.i[].nodes.itemStyle.emphasis.linkStyle (Noch nicht implementiert) ... 240
3.31.196 chart.series.i[].nodes.itemStyle.emphasis.linkStyle.normal (Noch nicht implementiert) ... 240
3.31.197 chart.series.i[].nodes.itemStyle.emphasis.linkStyle.normal.type (Noch nicht implementiert) ... 240
3.31.198 chart.series.i[].nodes.itemStyle.emphasis.linkStyle.normal.color (Noch nicht implementiert) ... 240
3.31.199 chart.series.i[].nodes.itemStyle.emphasis.linkStyle.normal.width (Noch nicht implementiert) ... 241
3.31.200 chart.series.i[].nodes.itemStyle.emphasis.linkStyle.emphasis (Noch nicht implementiert) ... 241
3.31.201 chart.series.i[].nodes.itemStyle.emphasis.linkStyle.emphasis.type (Noch nicht implementiert) ... 241
3.31.202 chart.series.i[].nodes.itemStyle.emphasis.linkStyle.emphasis.color (Noch nicht implementiert) ... 241
3.31.203 chart.series.i[].nodes.itemStyle.emphasis.linkStyle.emphasis.width (Noch nicht implementiert) ... 241
3.31.204 chart.series.i[].nodes.itemStyle.emphasis.label (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 241
3.31.205 chart.series.i[].nodes.itemStyle.emphasis.label.show (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 241
3.31.206 chart.series.i[].nodes.itemStyle.emphasis.label.rotate (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 242
3.31.207 chart.series.i[].nodes.itemStyle.emphasis.label.distance (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 242
3.31.208 chart.series.i[].nodes.itemStyle.emphasis.label.formatter (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 242
3.31.209 chart.series.i[].nodes.itemStyle.emphasis.label.textStyle (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 243
3.31.210 chart.series.i[].nodes.itemStyle.emphasis.label.textStyle.color (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 243
3.31.211 chart.series.i[].nodes.itemStyle.emphasis.label.textStyle.align (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 243
3.31.212 chart.series.i[].nodes.itemStyle.emphasis.label.textStyle.baseline (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 243
3.31.213 chart.series.i[].nodes.itemStyle.emphasis.label.textStyle.fontFamily (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 243
3.31.214 chart.series.i[].nodes.itemStyle.emphasis.label.textStyle.fontSize (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 244

3.31.215 chart.series.i[].nodes.itemStyle.emphasis.label.textStyle.fontStyle (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 244
3.31.216 chart.series.i[].nodes.itemStyle.emphasis.label.textStyle.fontWeight (Noch nicht implementiert, auch bei chord (noch nicht implementiert)) ... 244
3.31.217 chart.series.i[].nodes.initial (Noch nicht implementiert) ... 244
3.31.218 chart.series.i[].nodes.fixX (Noch nicht implementiert) ... 244
3.31.219 chart.series.i[].nodes.fixY (Noch nicht implementiert) ... 244
3.31.220 chart.series.i[].nodes.draggable (Noch nicht implementiert) ... 244
3.32 Series- Alle Charttypen ... 245
3.32.1 chart.series.i[].type ... 245
3.32.2 chart.series.i[].name ... 245
3.32.3 chart.series.i[].clickable ... 245
3.32.4 chart.series.i[].legendHoverLink ... 245
3.32.5 chart.series.i[].x ... 245
3.32.6 chart.series.i[].x2 ... 245
3.32.7 chart.series.i[].y ... 245
3.32.8 chart.series.i[].y2 ... 245
3.32.9 chart.series.i[].z ... 245
3.32.10 chart.series.i[].zlevel ... 246
3.33 Series-Line/Area ... 246
3.33.1 chart.series.i[].stack (Auch bei bar) ... 246
3.33.2 chart.series.i[].xAxisIndex (Auch bei bar, scatter, eventRiver) ... 246
3.33.3 chart.series.i[].yAxisIndex (Auch bei bar, scatter) ... 246
3.33.4 chart.series.i[].symbol (Auch bei scatter, radar, chord, force, tree) ... 246
3.33.5 chart.series.i[].symbolSize (Auch bei scatter, radar, chord, force, tree) ... 246
3.33.6 chart.series.i[].symbolRotate (Auch bei scatter, radar) ... 247
3.33.7 chart.series.i[].showAllSymbol ... 247
3.33.8 chart.series.i[].smooth ... 247
3.33.9 chart.series.i[].dataFilter ... 247
3.34 Series-Map ... 247
3.34.1 chart.series.i[].selectedMode (Noch nicht implementiert, auch bei pie) ... 247
3.34.2 chart.series.i[].mapType (Noch nicht implementiert) ... 247
3.34.3 chart.series.i[].hoverable (Noch nicht implementiert) ... 247
3.34.4 chart.series.i[].dataRangeHoverLink ... 248
3.34.5 chart.series.i[].mapLocation (Noch nicht implementiert) ... 248
3.34.6 chart.series.i[].mapValueCalculation (Noch nicht implementiert) ... 248

3.34.7 chart.series.i[].mapValuePrecision (Noch nicht implementiert) ... 248
3.34.8 chart.series.i[].showLegendSymbol (Noch nicht implementiert) ... 248
3.34.9 chart.series.i[].roam ... 248
3.34.10 chart.series.i[].scaleLimit (Noch nicht implementiert) ... 248
3.34.11 chart.series.i[].nameMap (Noch nicht implementiert) ... 248
3.34.12 chart.series.i[].textFixed (Noch nicht implementiert) ... 249
3.34.13 chart.series.i[].geoCoord (Noch nicht implementiert) ... 249
3.34.14 chart.series.i[].heatmap.blurSize (Noch nicht implementiert) ... 249
3.34.15 chart.series.i[].heatmap.gradientColors (Noch nicht implementiert) ... 249
3.34.16 chart.series.i[].heatmap.minAlpha (Noch nicht implementiert) ... 249
3.34.17 chart.series.i[].heatmap.valueScale (Noch nicht implementiert) ... 249
3.34.18 chart.series.i[].heatmap.opacity (Noch nicht implementiert) ... 249
3.34.19 chart.series.i[].heatmap.data (Noch nicht implementiert) ... 249
3.35 Series-Tree ... 250
3.35.1 chart.series.i[].symbol (Auch bei line/area, scatter, radar, chord, force) ... 250
3.35.2 chart.series.i[].symbolSize (Auch bei line/area, scatter, radar, chord, force) ... 250
3.35.3 chart.series.i[].roam ... 250
3.35.4 chart.series.i[].rootLocation (Noch nicht implementiert) ... 250
3.35.5 chart.series.i[].layerPadding ... 250
3.35.6 chart.series.i[].nodePadding (Noch nicht implementiert) ... 250
3.35.7 chart.series.i[].orient ... 250
3.35.8 chart.series.i[].direction ... 251
3.36 Series-Funnel ... 251
3.36.1 chart.series.i[].sort (Auch bei chord) ... 251
3.36.2 chart.series.i[].min (Auch bei gauge) ... 251
3.36.3 chart.series.i[].max (Auch bei gauge) ... 251
3.36.4 chart.series.i[].width ... 251
3.36.5 chart.series.i[].height ... 251
3.36.6 chart.series.i[].funnelAlign ... 251
3.36.7 chart.series.i[].minSize ... 251
3.36.8 chart.series.i[].maxSize ... 252
3.36.9 chart.series.i[].gap ... 252
3.37 Series-Scatter ... 252
3.37.1 chart.series.i[].xAxisIndex (Auch bei line/area, bar, eventRiver) ... 252
3.37.2 chart.series.i[].yAxisIndex (Auch bei line/area, bar) ... 252

3.37.3 chart.series.i[].symbol (Auch bei line/area, radar, chord, force, tree) ... 252
3.37.4 chart.series.i[].symbolSize (Auch bei line/area, radar, chord, force, tree) ... 252
3.37.5 chart.series.i[].symbolRotate (Auch bei line/area, radar) ... 253
3.37.6 chart.series.i[].largeThreshold ... 253
3.38 Series-Pie ... 253
3.38.1 chart.series.i[].radius (Auch bei gauge) ... 253
3.38.2 chart.series.i[].startAngle (Auch bei gauge) ... 253
3.38.3 chart.series.i[].minAngle ... 253
3.38.4 chart.series.i[].roseType ... 253
3.38.5 chart.series.i[].selectedOffset ... 253
3.38.6 chart.series.i[].selectedMode (Auch bei map (noch nicht implementiert)) ... 253
3.39 Series-Radar ... 254
3.39.1 chart.series.i[].symbol (Auch bei line/area, scatter, chord, force, tree) ... 254
3.39.2 chart.series.i[].symbolSize (Auch bei line/area, scatter, chord, force, tree) ... 254
3.39.3 chart.series.i[].symbolRotate (Auch bei line/area, scatter) ... 254
3.39.4 chart.series.i[].polarIndex ... 254
3.40 Series-Treemap ... 254
3.40.1 chart.series.i[].size (Auch bei force, worldCloud) ... 254
3.40.2 chart.series.i[].root (Noch nicht implementiert) ... 254
3.41 Series-EventRiver ... 255
3.41.1 chart.series.i[].xAxisIndex ... 255
3.41.2 chart.series.i[].weight ... 255
4 Dashlet-JavaScript-API ... 256
4.1 Die Dashlet-Application-API ... 256
4.1.1 showDashletDocument(params) ... 256
4.1.2 runDashletReport(params) ... 256
4.1.3 refreshDashletReport(params) ... 256
4.1.4 runAndExportReport(params) ... 257
4.1.5 downloadNativeDocument(params) ... 257
4.1.6 setDashletVisibility(params) ... 257
4.1.7 setDashletBoxCollapse(params) ... 257
4.1.8 changeDashletThemeSet(params) ... 257
5 Dashlet-DataTables ... 261
5.1 DataTables-AutoFill ... 261

5.1.1 table.autoFill.alwaysAsk (Noch nicht implementiert) ... 261
5.1.2 table.autoFill.columns (Noch nicht implementiert) ... 261
5.1.3 table.autoFill.editor (Noch nicht implementiert) ... 261
5.1.4 table.autoFill.enable (Noch nicht implementiert) ... 261
5.1.5 table.autoFill.focus (Noch nicht implementiert) ... 262
5.1.6 table.autoFill.update (Noch nicht implementiert) ... 262
5.1.7 table.autoFill (Noch nicht implementiert) ... 262
5.1.8 table.language.autoFill.button (Noch nicht implementiert) ... 262
5.1.9 table.language.autoFill.cancel (Noch nicht implementiert) ... 262
5.1.10 table.language.autoFill.fill (Noch nicht implementiert) ... 262
5.1.11 table.language.autoFill.fillHorizontal (Noch nicht implementiert) ... 263
5.1.12 table.language.autoFill.fillVertical (Noch nicht implementiert) ... 263
5.1.13 table.language.autoFill.increment (Noch nicht implementiert) ... 263
5.1.14 table.language.autoFill.info (Noch nicht implementiert) ... 263
5.1.15 table.language.autoFill (Noch nicht implementiert) ... 263
5.2 DataTables-Buttons ... 263
5.2.1 table.buttons.buttons.action (Noch nicht implementiert) ... 263
5.2.2 table.buttons.buttons.available (Noch nicht implementiert) ... 264
5.2.3 table.buttons.buttons.className (Noch nicht implementiert) ... 264
5.2.4 table.buttons.buttons.destroy (Noch nicht implementiert) ... 264
5.2.5 table.buttons.buttons.enabled (Noch nicht implementiert) ... 264
5.2.6 table.buttons.buttons.extend (Noch nicht implementiert) ... 265
5.2.7 table.buttons.buttons.init (Noch nicht implementiert) ... 265
5.2.8 table.buttons.buttons.key (Noch nicht implementiert) ... 265
5.2.9 table.buttons.buttons.name (Noch nicht implementiert) ... 265
5.2.10 table.buttons.buttons.namespace (Noch nicht implementiert) ... 265
5.2.11 table.buttons.buttons.text (Noch nicht implementiert) ... 266
5.3 DataTables-Callbacks ... 266
5.3.1 table.callback.createdRow (Noch nicht implementiert) ... 266
5.3.2 table.callback.drawCallback (Noch nicht implementiert) ... 266
5.3.3 table.callback.footerCallback (Noch nicht implementiert) ... 266
5.3.4 table.callback.formatNumber (Noch nicht implementiert) ... 266
5.3.5 table.callback.headerCallback (Noch nicht implementiert) ... 267
5.3.6 table.callback.infoCallback (Noch nicht implementiert) ... 267
5.3.7 table.callback.initComplete (Noch nicht implementiert) ... 267

5.3.8 table.callback.preDrawCallback (Noch nicht implementiert) ... 267
5.3.9 table.callback.rowCallback (Noch nicht implementiert) ... 267
5.3.10 table.callback.stateLoadCallback (Noch nicht implementiert) ... 268
5.3.11 table.callback.stateLoaded (Noch nicht implementiert) ... 268
5.3.12 table.callback.stateLoadParams (Noch nicht implementiert) ... 268
5.3.13 table.callback.stateSaveCallback (Noch nicht implementiert) ... 268
5.3.14 table.callback.stateSaveParams (Noch nicht implementiert) ... 268
5.4 DataTables-ColumnDef ... 269
5.5 DataTables-Columns ... 269
5.6 DataTables-Feature ... 269
5.6.1 table.feature.autoWidth ... 269
5.6.2 table.feature.deferRender ... 269
5.6.3 table.feature.info ... 269
5.6.4 table.feature.jQueryUI ... 269
5.6.5 table.feature.lengthChange ... 270
5.6.6 table.feature.ordering ... 270
5.6.7 table.feature.paging ... 270
5.6.8 table.feature.processing ... 270
5.6.9 table.feature.scrollX ... 270
5.6.10 table.feature.scrollY ... 271
5.6.11 table.feature.searching ... 271
5.6.12 table.feature.serverSide ... 271
5.6.13 table.feature.stateSave ... 271
5.7 DataTables-FixedColumns ... 272
5.8 DataTables-Internationalisation ... 272
5.8.1 table.language.aria.sortAscending ... 272
5.8.2 table.language.aria.sortDescending ... 272
5.8.3 table.language.aria ... 272
5.8.4 table.language.decimal ... 272
5.8.5 table.language.emptyTable ... 273
5.8.6 table.language.info ... 273
5.8.7 table.language.infoEmpty ... 273
5.8.8 table.language.infoFiltered ... 273
5.8.9 table.language.infoPostFix ... 273
5.8.10 table.language.lengthMenu ... 273

5.8.11 table.language.loadingRecords ... 274
5.8.12 table.language.paginate.first ... 274
5.8.13 table.language.paginate.last ... 274
5.8.14 table.language.paginate.next ... 274
5.8.15 table.language.paginate.previous ... 274
5.8.16 table.language.paginate ... 274
5.8.17 table.language.processing ... 275
5.8.18 table.language.search ... 275
5.8.19 table.language.searchPlaceholder ... 275
5.8.20 table.language.thousands ... 275
5.8.21 table.language.url ... 275
5.8.22 table.language.zeroRecords ... 275
5.9 DataTables-Options ... 276
5.9.1 table.deferLoading ... 276
5.9.2 table.destroy ... 276
5.9.3 table.displayStart ... 276
5.9.4 table.dom ... 276
5.9.5 table.lengthMenu ... 277
5.9.6 table.order ... 277
5.9.7 table.orderCellsTop ... 277
5.9.8 table.orderClasses ... 277
5.9.9 table.orderFixed ... 278
5.9.10 table.orderMulti ... 278
5.9.11 table.pageLength ... 278
5.9.12 table.pagingType ... 278
5.9.13 table.renderer ... 279
5.9.14 table.retrieve ... 279
5.9.15 table.rowId ... 279
5.9.16 table.scrollCollapse ... 279
5.9.17 table.search.caseInsensitive ... 280
5.9.18 table.search.regex ... 280
5.9.19 table.search.smart ... 280
5.9.20 table.search.search ... 280
5.9.21 table.searchCols ... 280
5.9.22 table.searchDelay ... 281
5.9.23 table.stateDuration ... 281

5.9.24 table.stripeClasses ... 281
5.9.25 table.tabIndex ... 281
5.10 DataTables-Responsive ... 282
5.11 DataTables-RowReorder ... 282
5.12 DataTables-Scroller ... 282
5.13 DataTables-Select ... 282
6 KPI-Dashlets ... 283
6.1 Einleitung ... 283
6.2 Ein KPI-Dashlet erzeugen ... 283
6.3 Daten ans Dashlet übergeben ... 283
6.4 Hinzufügen von Optionen ... 283
7 Verfügbare Funktionen in Dashlet Engine ... 284
7.1 Einleitung ... 284

1 Einführung TRADUI Dashlet-Entwicklung

1.1 Was sind eigentlich Dashlets

Unter Dashlets verstehen wir einzelne Visualisierungs-Elemente auf einem Dashboard. Das sind z.B. Charts, Tabellen, Textelemente oder andere interaktive Darstellungen.

Diese bestehen immer aus folgenden Elementen

- HTML5
- Cascading Style Sheets
- JavaScript

Erzeugt werden die Dashlets über die TRADUI-Dashlet-Engine. Diese übernimmt das Rendering mit aktuellen Daten.

1.2 Warum Dashlets?

Dashboards zu entwicklen und Framework-Entscheidungen zu treffen, die auf lange Zeit Bestand haben ist oft nicht einfach. Die Dashlet-Engine unterstützt daher verschiedene JavaScript- und Visualisierungs-Frameworks beim Rendering. Die Engine hat eine vereinfachte und standardisierte Schnittstelle für diese Frameworks. Meist muss man nur Darstellung auswählen und ein paar Optionen verändern. Das Erzeugen von HTML, CSS und Javascript übernimmt die Engine on-the-fly.

1.3 BIRT-Plugin - Datenversorgung und Business Intelligence

BIRT (Business Intelligence Reporting Tool) ist das optimale Tool um Dashlets mit Daten zu versorgen. Wir haben deshalb ein BIRT-Plugin geschaffen, welches unsere Dashlet-Engine nutzt, um mit BIRT sehr einfach Dashlets zu erzeugen. Bereits bestehende BIRT-Reports können um Dashlets erweitert werden.

1.4 Dashboards / dashBIRD - Eigenentwicklung oder fertige Lösung

Für die Nutzung von Dashlets auf einem Dashboard, können sie auf die fertige Lösung der TRADUI zurückgreifen, genannt dashBIRD als Bestandteil unseres Reporting-Servers servBIRD. BIRD ist hierbei die Abkürzung von Business Intelligence Reporting Desktop, um aus BIRT dem Tool eine einsatzbereite Gesamt-Lösung zu machen. Möchten man hingegen Dashlets in ein eigenes Portal integrieren, steht ihnen hier ein Integrations-Kit mit Dashlet-API und Restschnittstelle für unseren Server bereit.

Copyright © 2023 TRADUI Technologies GmbH

2 C3-Charts

2.1 Einleitung

Dieses Dokument zeigt welcher der C3-Chart-Funktionen derzeit schon von den TRADUI-Dashlets unterstützt werden.

2.2 Ein C3-Chart-Dashlet erzeugen

Erzeugung eines C3-Chart-Dashlets mit der Dashlet-Engine:

Dashlet erzeugen (Java)

```
DashletEngine engine = new DashletEngine();
Dashlet dashlet = engine.createDashlet(DashletType.C3_CHART);
```

2.3 Daten ans Dashlet übergeben

Daten werden ans Dashlet als DataTable übergeben, wie bei allen Dashlets. Der Aufbau der Daten-Tabelle wird weiter unten beschrieben. (Muss noch beschrieben werden)

```
dashlet.setData(table);
```

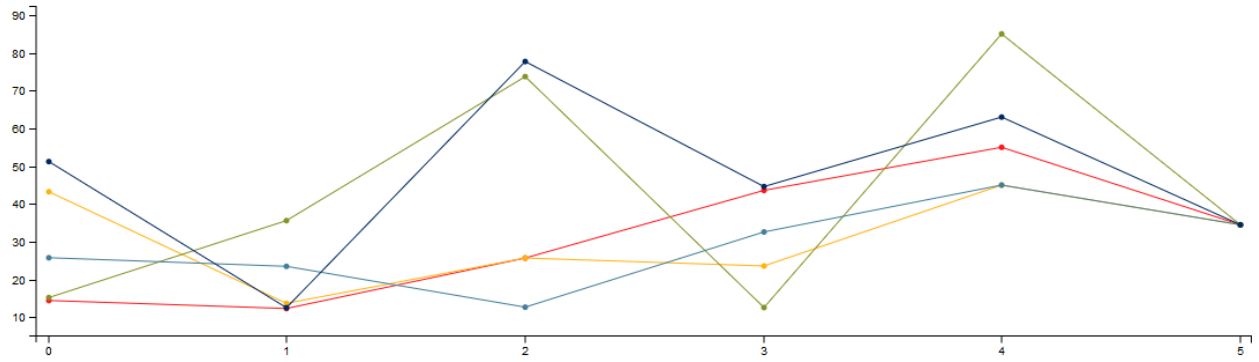
2.4 Hinzufügen von Optionen

Um das Verhalten der C3-Chart-Dashlets zu beeinflussen, können Optionen übergeben werden.
dashlet.addOption(OPTIONSKEY , OPTIONSVALUE);
Options-Keys und mögliche Optionswerte werden im Weiteren in diesem Dokument beschrieben.

2.5 Chart-Typen

2.5.1 Line Chart

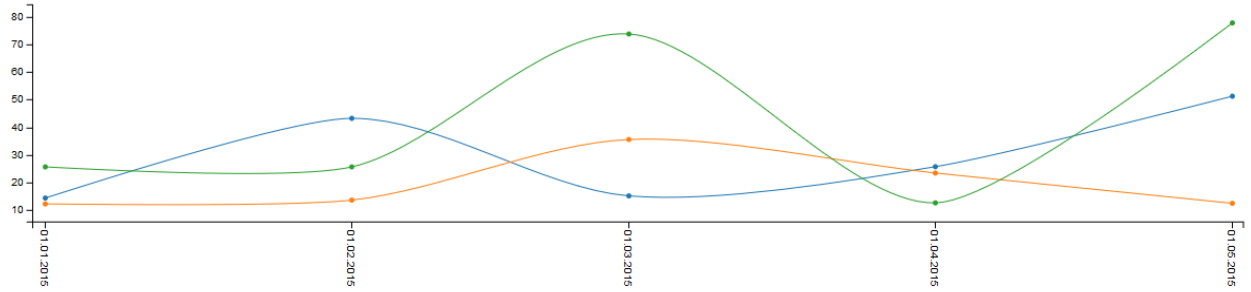
```
dashlet.addOption("chart.type", "line");
```



2.5.2 Timeseries Chart

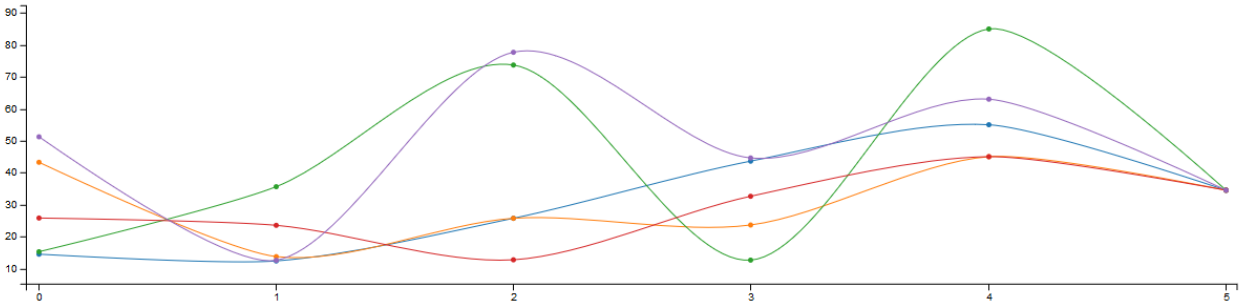
Timeseries Chart (Java)

```
dashlet.addOption("chart.timeseries", "series");
// oder
dashlet.addOption("chart.timeseries", "columns");
// Formatierung: (siehe auch D3-Formatierung)
dashlet.addOption("chart.axis.x.tick.format", "%d.%m.%Y");
```



2.5.3 Spline Chart

```
dashlet.addOption("chart.type", "spline");
```



2.5.4 Simple XY Line Chart

Um Werte für die X-Achse zu liefern werden diese als erster Datensatz mitgegeben.
Die festdefinierte Spalte **AxisAlias** ist der Bezug zu den X-Achsen-Daten.
ID,Series,AxisAlias,Daten1,Daten2,Daten3,Daten14,Daten5,Daten6
0,"xz", null, 2, 5, 8, 11, 14, 15
1,"Detlef", "xz", 14.46, 12.34, 25.7, 43.6, 55.0, 34.5
2,"Roland", "xz", 43.24, 13.7, 25.7, 23.6, 45.0, 34.5
3,"Felix", "xz", 15.23, 35.6, 73.7, 12.6, 85.0, 34.5
4,"Christian", "xz", 25.78, 23.5, 12.7, 32.6, 45.0, 34.5
9,"Jan", "xz", 51.23, 12.55, 77.7, 44.6, 63.0, 34.5

2.5.5 Multiple XY Line Chart

Multiple X-Achsendaten werden genau wie bei der einfachen Darstellung übergeben. Hier gibt es lediglich mehre Datensätze mit X-Achsen-Daten, also auch mehrere Alias-Namen für die X-Achse

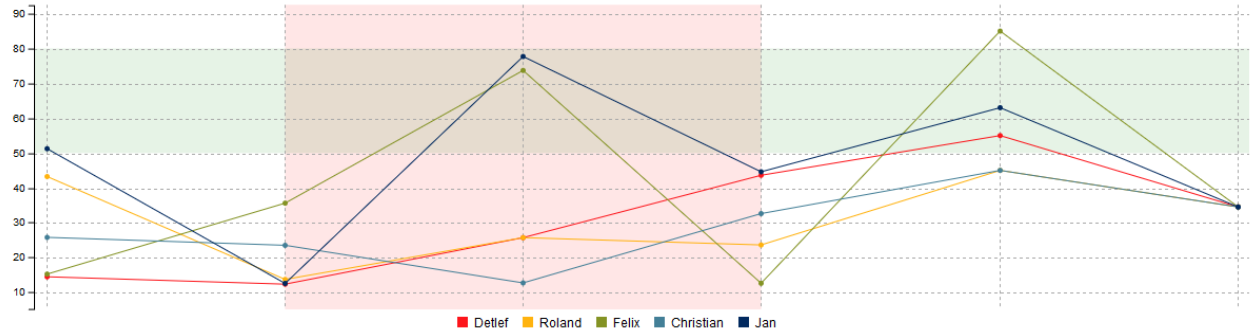
ID,Series,AxisAlias,Daten1,Daten2,Daten3,Daten14,Daten5,Daten6
0,"x1", null, 2, 5, 8, 11, 14, 15
0,"x2", null, 3, 4, 5, 7, 12, 16
1,"Detlef", "x1", 14.46, 12.34, 25.7, 43.6, 55.0, 34.5
2,"Roland", "x1", 43.24, 13.7, 25.7, 23.6, 45.0, 34.5
3,"Felix", "x2", 15.23, 35.6, 73.7, 12.6, 85.0, 34.5
4,"Christian", "x2", 25.78, 23.5, 12.7, 32.6, 45.0, 34.5
9,"Jan", "x2", 51.23, 12.55, 77.7, 44.6, 63.0, 34.5

2.5.6 Line Chart with Regions

Damit lassen sich Teile des Charts als gestrichelte Line (dashed) darstellen.
Hinweis: Bisher wird nur der Style 'dashed' unterstützt.

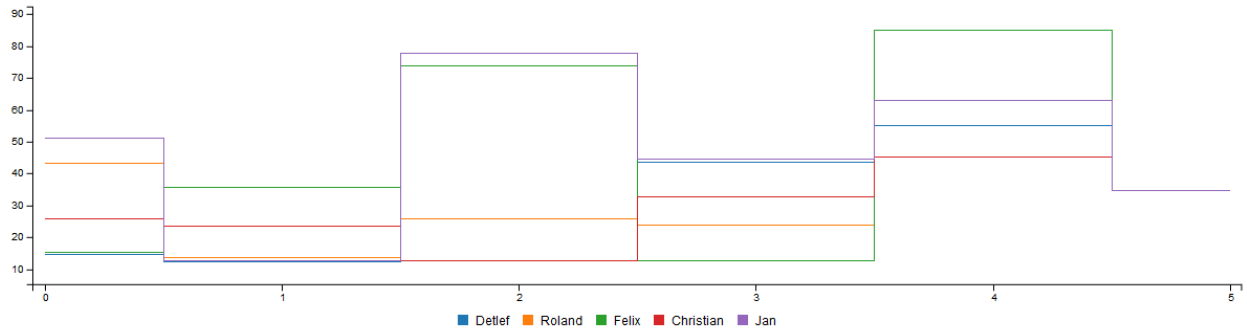
Line Chart with Regions

```
1  dashlet.addOption("chart.data.region", new ChartRegion("Christian", 1, 2,
   "dashed"));
2  dashlet.addOption("chart.data.region", new ChartRegion("Christian", 4,
   null, "dashed"));
3  dashlet.addOption("chart.data.region", new ChartRegion("Jan", "dashed"));
```



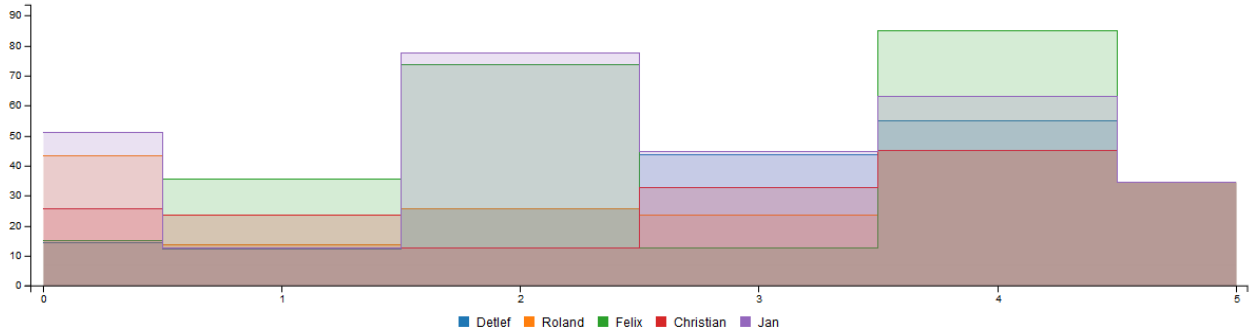
2.5.7 Step Chart

```
dashlet.addOption("chart.type", step");
```



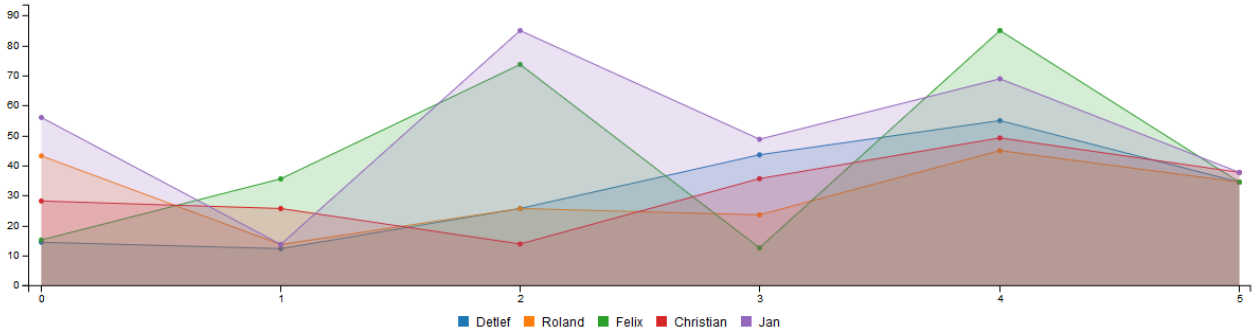
2.5.8 Area Step Chart

```
dashlet.addOption("chart.type", "area-step");
```



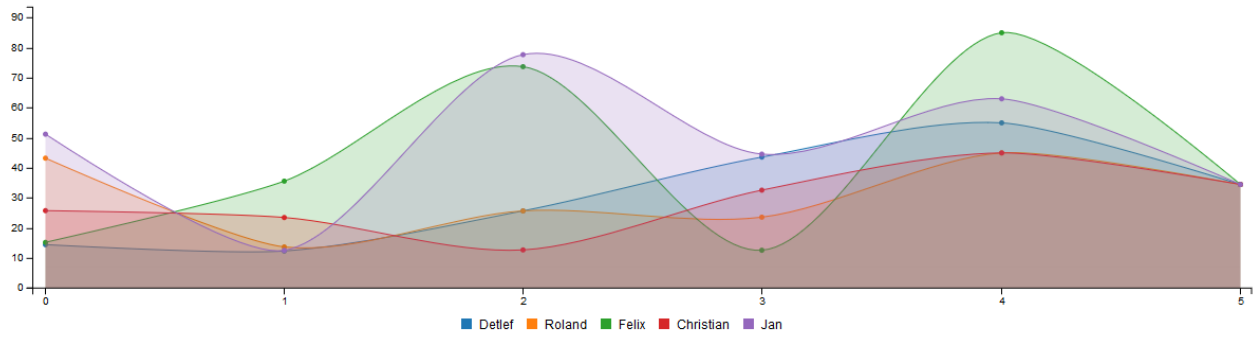
2.5.9 Area Chart

```
dashlet.addOption("chart.type", "area");
```



2.5.10 Area Spline Chart

```
dashlet.addOption("chart.type", "area-spline");
```

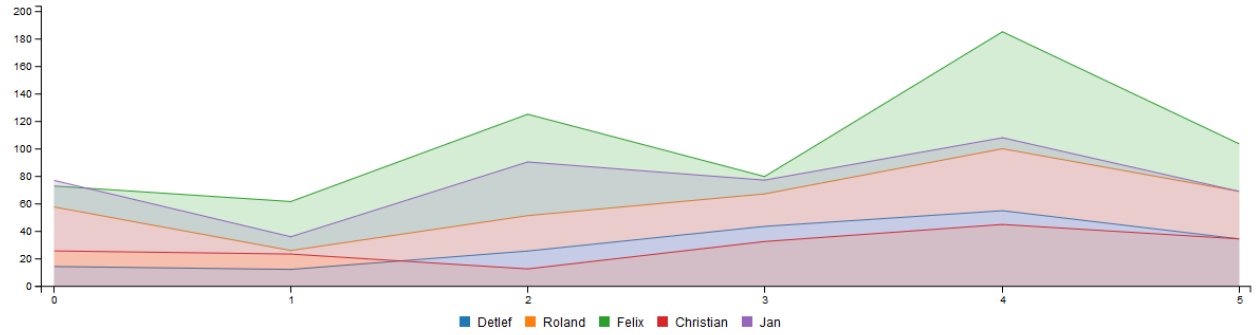


2.5.11 Stacked Area Chart

Hinweise -> siehe Stacked Bar Chart.

Stacked Area Chart (Java)

```
dashlet.addOption("chart.type", "area");
dashlet.addOption("chart.stacked", "groups");
// oder
dashlet.addOption("chart.stacked", "all");
```

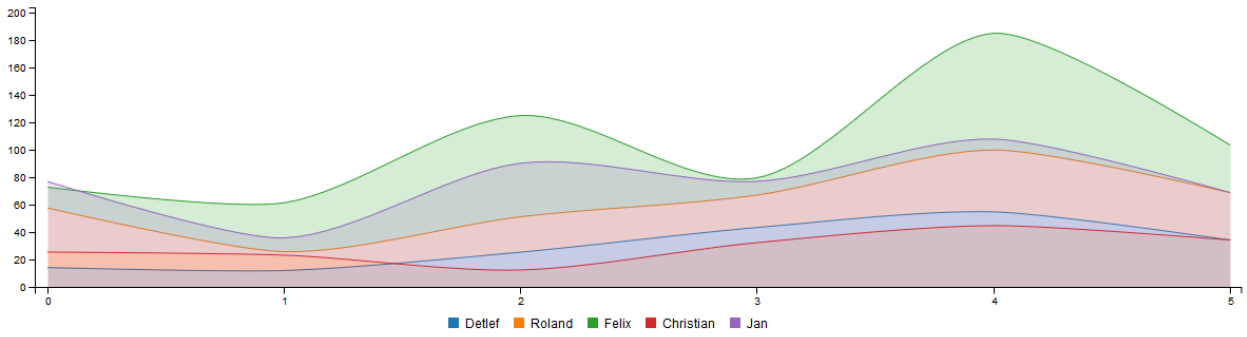


2.5.12 Stacked Area Spline Chart

Hinweise -> siehe Stacked Bar Chart.

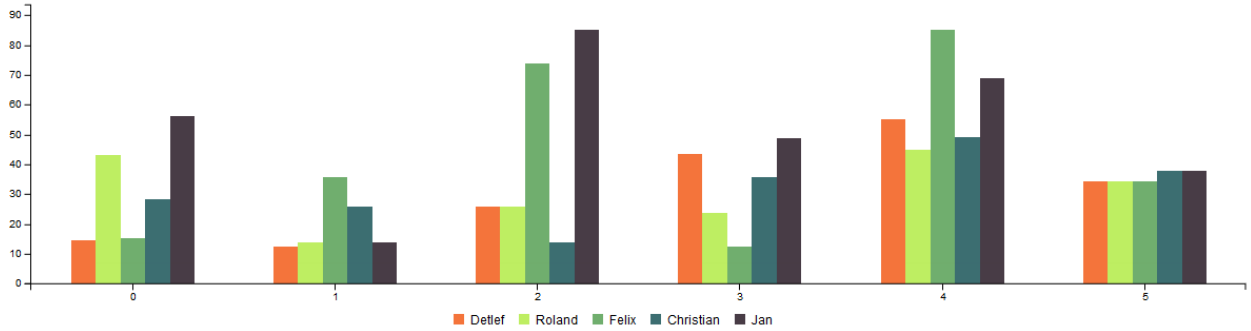
Stacked Area Spline Chart (Java)

```
dashlet.addOption("chart.type", "area-spline");
dashlet.addOption("chart.stacked", "groups");
// oder
dashlet.addOption("chart.stacked", "all");
```



2.5.13 Bar Chart

```
dashlet.addOption("chart.type", "bar");
```

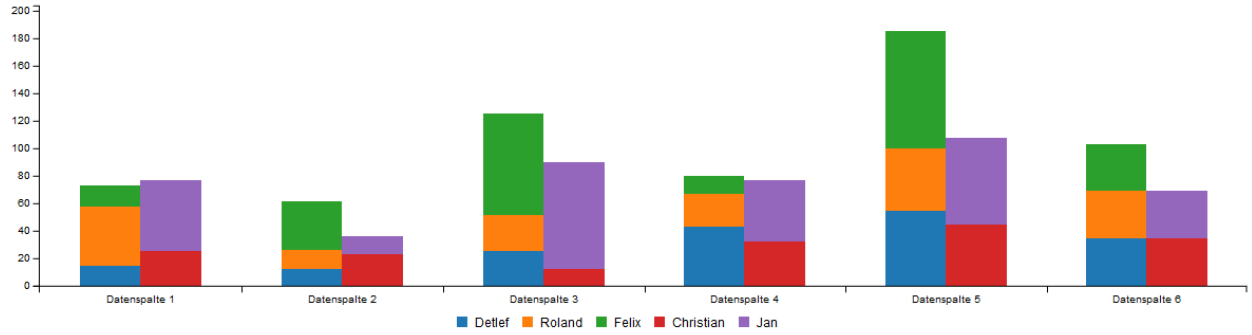


2.5.14 Stacked Bar Chart

Um Daten in Bar Charts übereinander zu stapeln, muss der Chart an der Datenspalte die Groups als „gruppenname" definieren, oder bei Series-Definition als Kategorie eine Spalte "Groups" übergeben werden, indem dort der "gruppenname" einzutragen ist. Alle Daten mit dem gleichen Gruppennamen werden dann aufeinander gestapelt. Außerdem müssen die Gruppen sichtbar gemacht werden.
Die Option „all" bewirkt das immer alles unabhängig von Gruppen übereinander gestapelt wird. Bei dieser Option werden keine Gruppen benötigt.

Stacked Bar Chart (Java)

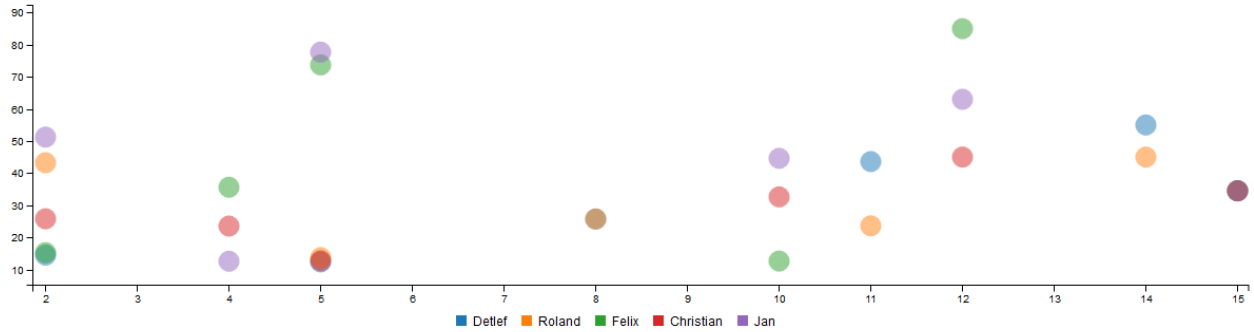
```
1  dashlet.addOption("chart.type", "bar");
2  dashlet.addOption("chart.stacked", "groups");
3  // oder
4  dashlet.addOption("chart.stacked", "all");
```



2.5.15 Scatter Plot

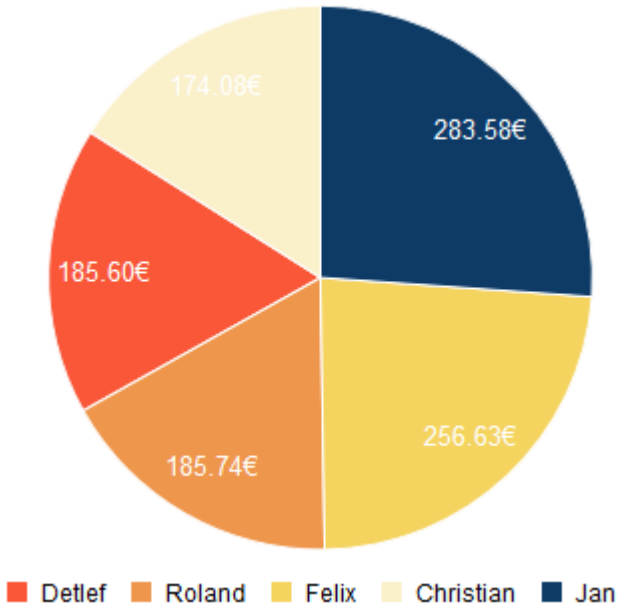
Datenübergabe wie beim Multiple XY Line Chart. Benötigt X-Achsen-Daten.

```
dashlet.addOption("chart.type", "scatter");
```



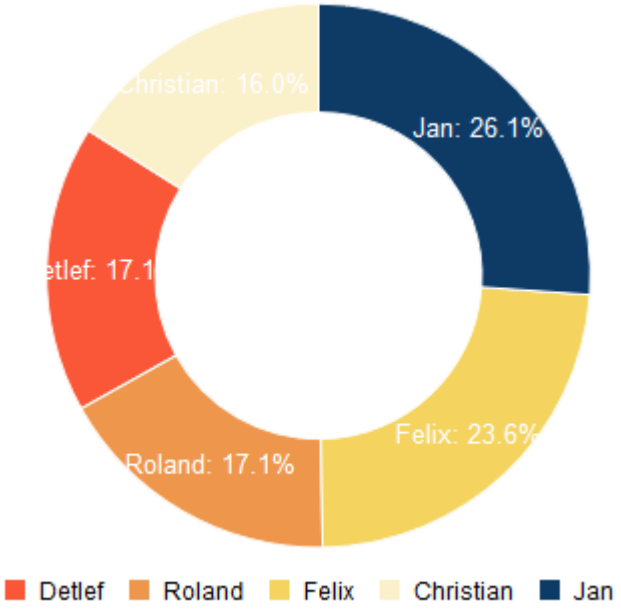
2.5.16 Pie Chart

```
dashlet.addOption("chart.type", "pie");
```



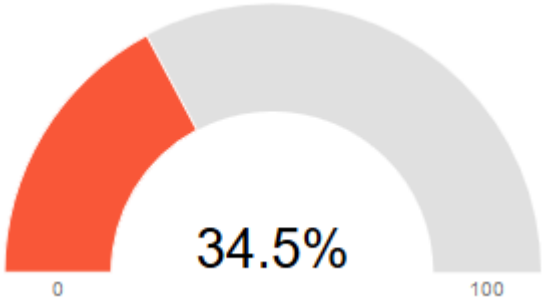
2.5.17 Donut Chart

```
dashlet.addOption("chart.type", "donut");
```



2.5.18 Gauge Chart

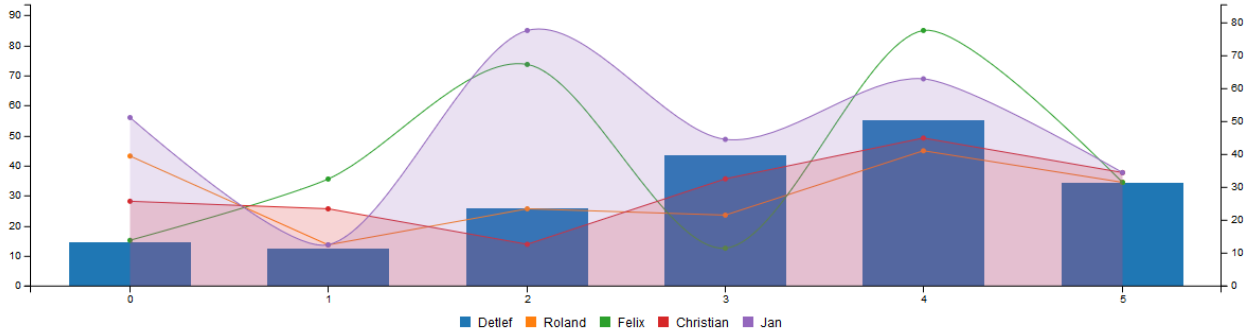
```
dashlet.addOption("chart.type", "gauge");
```



2.5.19 Combination Chart

Um verschiedene Chart-Typen in Kombination abzubilden, muss der Chart an der Datenspalte der Typ definiert sein, oder bei Series-Definition als Kategorie eine Spalte " Types" übergeben werden, indem der Chart-Typ einzutragen ist. Außerdem muss die Typ-Darstellung per Option sichtbar gemacht werden.

```
dashlet.addOption("chart.combination.show", "show");
```



2.6 Axis

2.6.1 Category Axis

Category Axis (Java)

```
dashlet.addOption("chart.categories", "series");
dashlet.addOption("chart.categories", "columns");
```

2.6.2 Rotated Axis

```
dashlet.addOption("chart.rotated", "true");
```

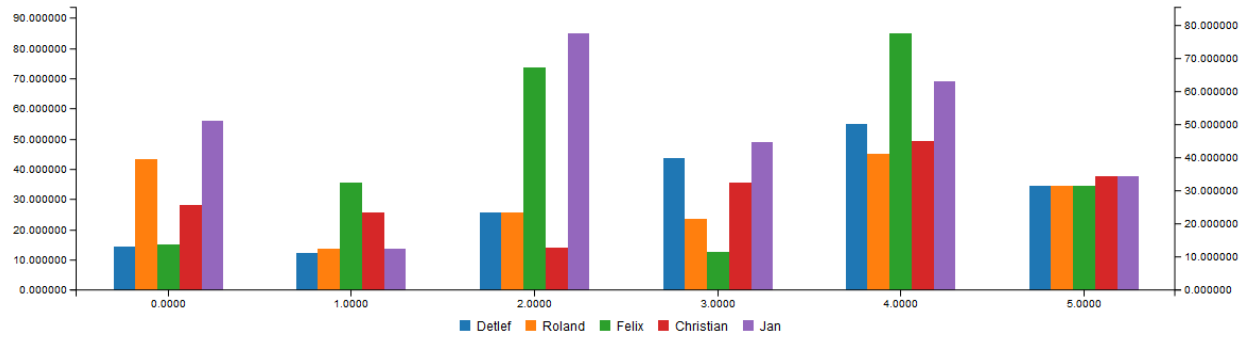
2.6.3 Additional Y Axis

Um Daten auf der zweiten Y-Achse Y2 abzubilden, muss der Chart an der Datenspalte die Axis als „y2" definiert sein, oder bei Series-Definition als Kategorie eine Spalte "Axis" übergeben werden, indem dort "y2" eingetragen ist. Außerdem muss die Y2-Achse sichtbar gemacht werden.

```
dashlet.addOption("chart.axis.y2.show", "show");
```

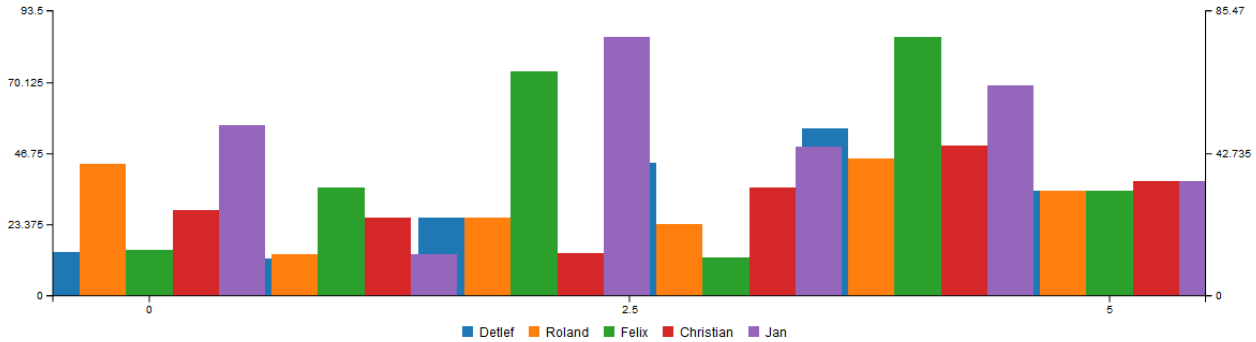
2.6.4 X Axis Tick Format

```
dashlet.addOption("chart.axis.x.tick.format", ".4f");
```



2.6.5 X Axis Tick Count

```
dashlet.addOption("chart.axis.x.tick.count", "3");
```



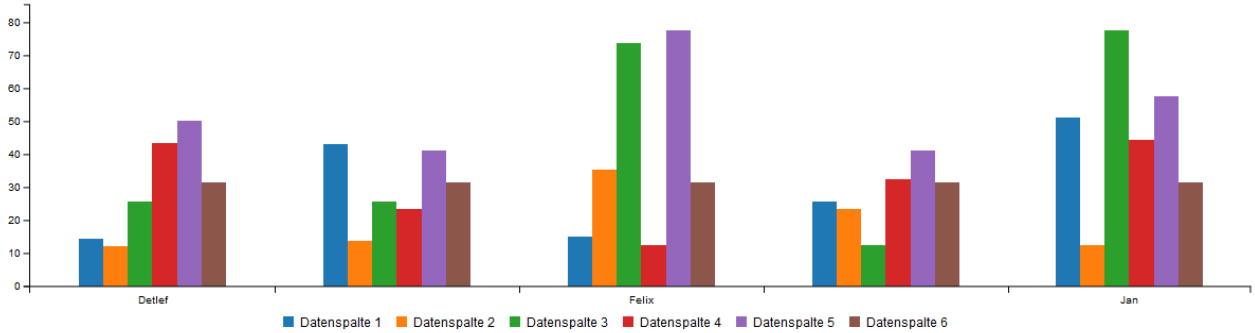
2.6.6 X, Y, Y2 Axis Tick Values

X, Y, Y2 Axis Tick Values (Java)

```
dashlet.addOption("chart.tick.values", new ChartTickValues("x", new Object[] {1,2.5,5}));
dashlet.addOption("chart.tick.values", new ChartTickValues("y", new Object[] {10,20,30,40,80}));
dashlet.addOption("chart.tick.values", new ChartTickValues("y2", new Object[] {0.2,0.3,0.4,0.7,0.9}));
```

2.6.7 X Axis Tick Culling

```
dashlet.addOption("chart.axis.x.tick.culling.max", "3");
```



2.6.8 X Axis Tick Fitting

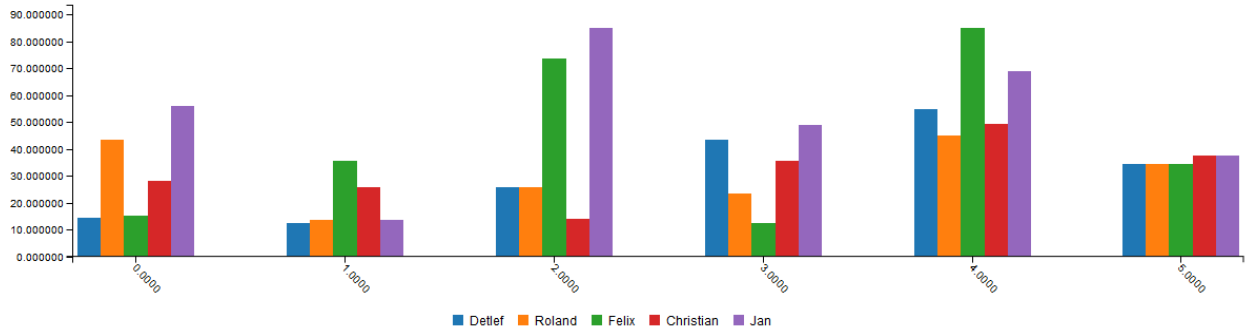
```
dashlet.addOption("chart.axis.x.tick.fitting", "true");
```

2.6.9 X Axis Timezone

```
dashlet.addOption("chart.localtime", "false");
```

2.6.10 Rotate X Axis Tick Text

```
dashlet.addOption("chart.axis.x.tick.rotate", "45");
```



2.6.11 Y Axis Tick Format

```
dashlet.addOption("chart.axis.y.tick.format", ".4f");
```

2.6.12 Y Axis Tick Count

```
dashlet.addOption("chart.axis.y.tick.count", "5");
```

2.6.13 Y2 Axis Tick Format

```
dashlet.addOption("chart.axis.y2.tick.format", ".4f");
```

Copyright © 2023 TRADUI Technologies GmbH

2.6.14 Y2 Axis Tick Count

```
dashlet.addOption("chart.axis.y2.tick.count", "5");
```

2.6.15 Padding for X Axis

Padding for X Axis (Java)

```
dashlet.addOption("chart.axis.x.padding.left", "2");
dashlet.addOption("chart.axis.x.padding.right", "4");
```

2.6.16 Padding for Y Axis

Padding for Y Axis

```
dashlet.addOption("chart.axis.y.padding.top", "100");
dashlet.addOption("chart.axis.y.padding.bottom", "200");
dashlet.addOption("chart.axis.y2.show", "show");
dashlet.addOption("chart.axis.y2.padding.top", "10");
dashlet.addOption("chart.axis.y2.padding.bottom", "20");
```

2.6.17 Range for Y and Y2 Axis

Range for Y and Y2 Axis (Java)

```
dashlet.addOption("chart.axis.y.min", "-20");
dashlet.addOption("chart.axis.y.max", "180");
dashlet.addOption("chart.axis.y2.min", "-80");
dashlet.addOption("chart.axis.y2.max", "250");
```

2.6.18 Axis Label

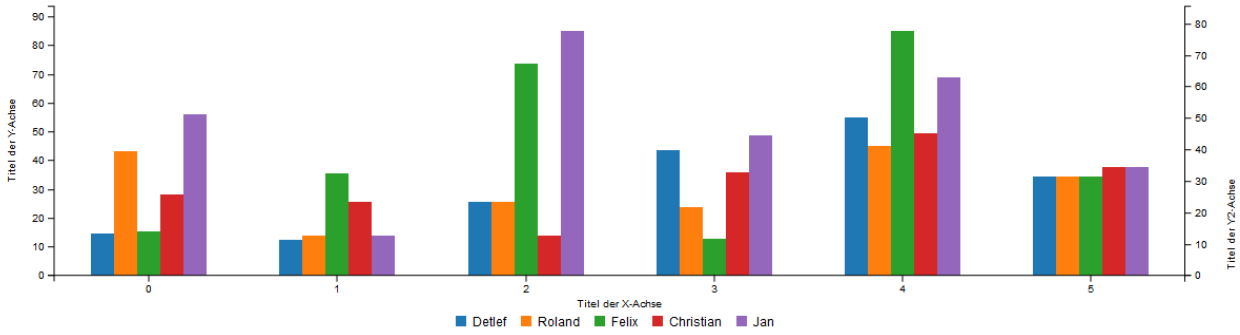
Axis Label (Java)

```
dashlet.addOption("chart.axis.y2.label", "Titel der Y2-Achse");
dashlet.addOption("chart.axis.y.label", "Titel der Y-Achse");
dashlet.addOption("chart.axis.x.label", "Titel der X-Achse");
```

2.6.19 X Axis Label Position

```
dashlet.addOption("chart.axis.x.label.position", "outer-center");
```

Copyright © 2023 TRADUI Technologies GmbH



// **inner-right** : default
// **inner-center**
// **inner-left**
// **outer-right**
// **outer-center**
// **outer-left**

2.6.20 Y and Y2 Axis Label Position

Y and Y2 Axis Label Position

```
dashlet.addOption("chart.axis.y.label.position", "outer-bottom");
dashlet.addOption("chart.axis.y2.label.position", "outer-middle");
```

// **inner-top** : default
// **inner-middle**
// **inner-bottom**
// **outer-top**
// **outer-middle**
// **outer-bottom**

2.7 # Data

2.7.1 Column Oriented Data

Column-oriented data can be used as input.

2.7.2 Row Oriented Data

Row-oriented data can be used as input.

2.7.3 JSON Data

JSON can be used as input.

2.7.4 Data from URL

Data from URL can be used as input.

2.7.5 Category Data

Load data with x values on category axis.

2.7.6 Load Data

Load data dynamically.

2.7.7 Data Name

Set name for each data.

2.7.8 Data Color

Um Daten mit Farbwerten zu verbinden, muss der Chart an der Datenspalte das Property Color als HTML-Hex-Farbwert definiert sein, oder bei Series-Definition als Kategorie eine Spalte "Color" übergeben werden, indem dort der HTML-Hex-Farbwert einzutragen ist. Außerdem muss die Daten-Farb-Verwendung sichtbar gemacht werden.

```
dashlet.addOption("chart.data.color", "show");
```

2.7.9 Data Order

Define data order. This will be used for stacked bar chart.

2.7.10 Data Label

Data Label (Java)

```
dashlet.addOption("chart.data.labels.show", "series");
dashlet.addOption("chart.data.labels.show", "columns");
```

2.7.11 Data Label Format

Das Datenformat für die Labels wird aus entweder aus der Spalte oder Serie übernommen.

Data Label Format (Java)

```
DataTableColumn formatcolumn = new DataTableColumn("DataLabelFormat",
DataTableColumn.DATA_TYPE_STRING, ".4f", DataTableColumn.ALIGN_LEFT );
formatcolumn.setVisible(false);
table.addTableColumn(formatcolumn);
```

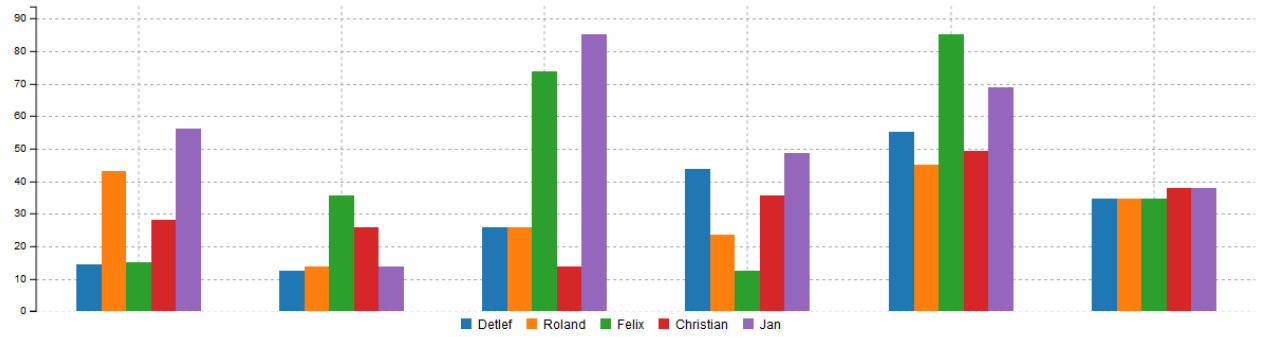
Copyright © 2023 TRADUI Technologies GmbH

2.8 # Grid

2.8.1 Grid Lines

Grid Lines (Java)

```
dashlet.addOption("chart.grid.x.show", "show");
dashlet.addOption("chart.grid.y.show", "show");
```



2.8.2 Optional X Grid Lines

Optional X Grid Lines (Java)

```
dashlet.addOption("chart.grid.line", new ChartGridLine("x", 20.8, "Line 1 bei 20", "start", "linestyle"));
dashlet.addOption("chart.grid.line", ChartGridLine("x", 60.5, "Line 2 bei 60", ""));
dashlet.addOption("chart.grid.line", new ChartGridLine("x", 80.7, "Line 3 bei 80", "middle", "linestyle"));
```

2.8.3 Optional Y Grid Lines

Optional Y Grid Lines (Java)

```
dashlet.addOption("chart.grid.line", new ChartGridLine("y", 20.8, "Line 1 bei 20", "start", "linestyle"));
dashlet.addOption("chart.grid.line", ChartGridLine("y", 60.5, "Line 2 bei 60", ""));
dashlet.addOption("chart.grid.line", new ChartGridLine("y", 80.7, "Line 3 bei 80", "middle", "linestyle"));
```

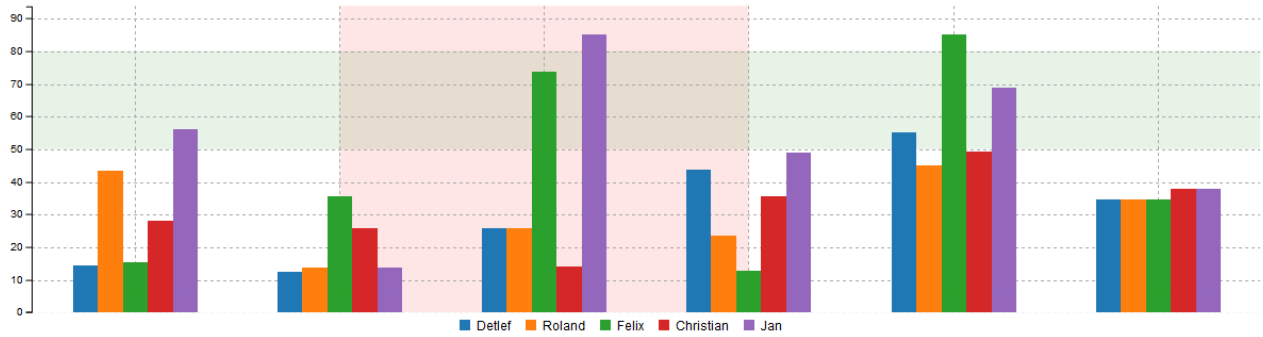
Copyright © 2023 TRADUI Technologies GmbH

2.9 # Region

2.9.1 Region

Region (Java)

```
dashlet.addOption("chart.grid.region", new ChartRegion("x", 1, 3, "regionstyle"));
dashlet.addOption("chart.grid.region", new ChartRegion("y", 50, 80, " regionstyle "));
```



2.9.2 Region with Timeseries

Show rects on timeseries chart.

2.10 # Interaction

2.10.1 Sub Chart

Show sub chart for zoom and selection range.

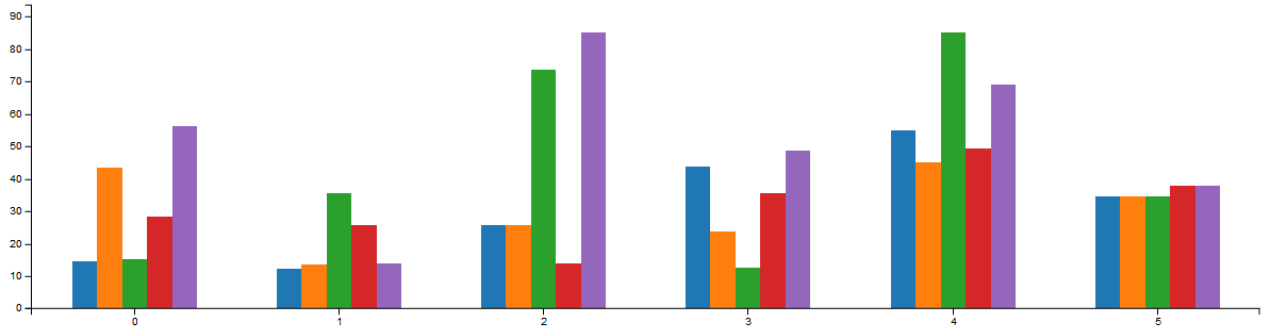
2.10.2 Zoom

Zoom by mouse wheel event and slide by drag.

2.11 # Legend

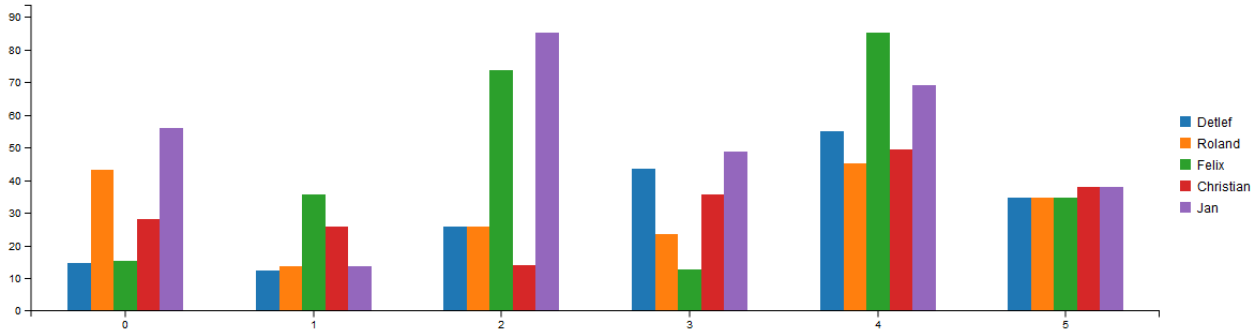
2.11.1 Hide Legend

```
dashlet.addOption("chart.legend.show", "hidden");
```



2.11.2 Legend Position

```
dashlet.addOption("chart.legend.position", "right");
```



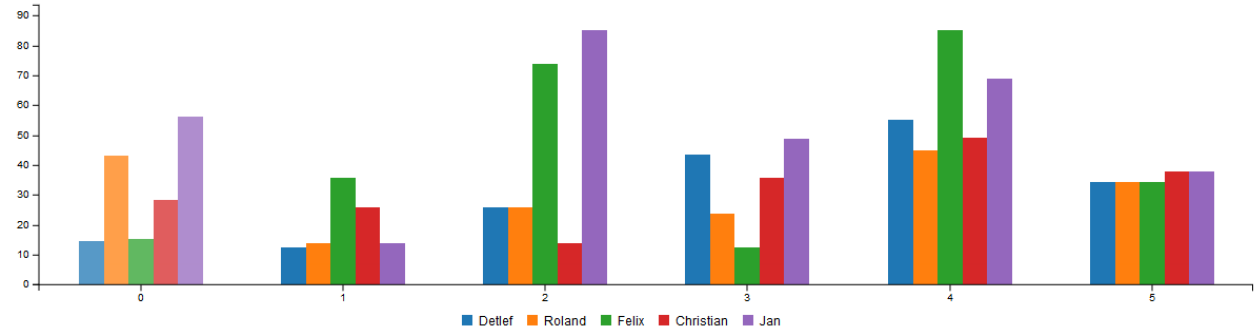
2.11.3 Custom Legend

Build custom legend

2.12 # Tooltip

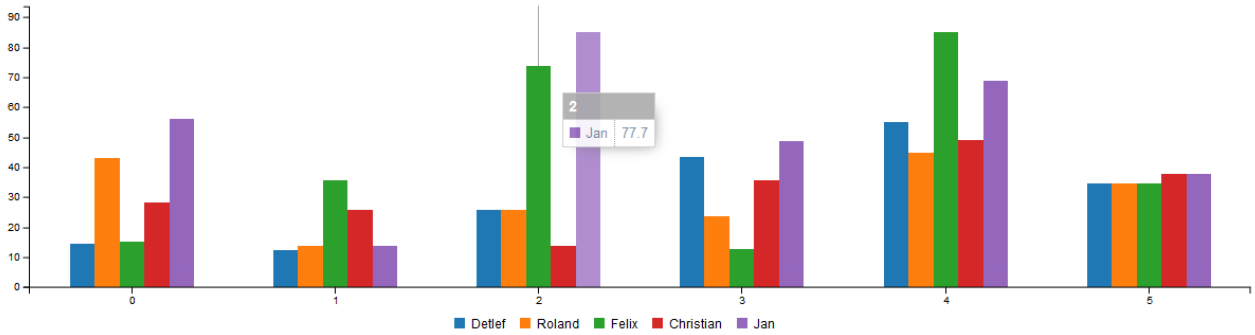
2.12.1 Hide Tooltip

```
dashlet.addOption("chart.tooltip.show", "hidden");
```



2.12.2 Tooltip Grouping

```
dashlet.addOption("chart.tooltip.grouped", "false");
```



2.12.3 Tooltip Format

Tooltip Format (Java)

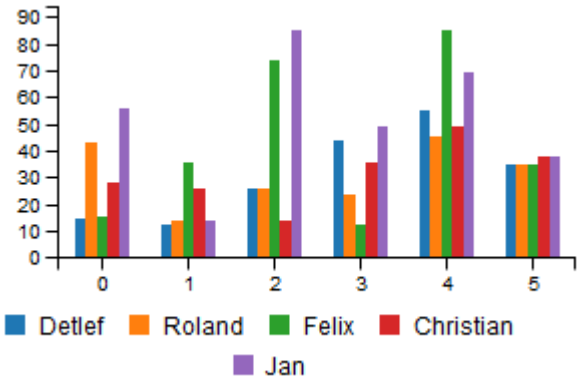
```
dashlet.addOption("chart.tooltip.format", ",.6f%");
dashlet.addOption("chart.tooltip.title", "Datenpunkt [value]");
```

2.13 # Chart Options

2.13.1 Chart Size

Chart Size (Java)

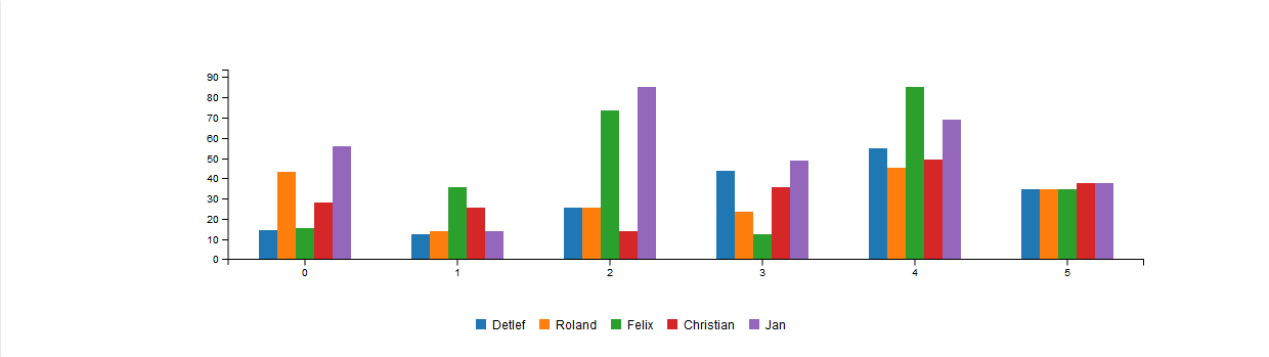
```
dashlet.addOption("chart.size.height", "200");
dashlet.addOption("chart.size.width", "300");
```



2.13.2 Chart Padding

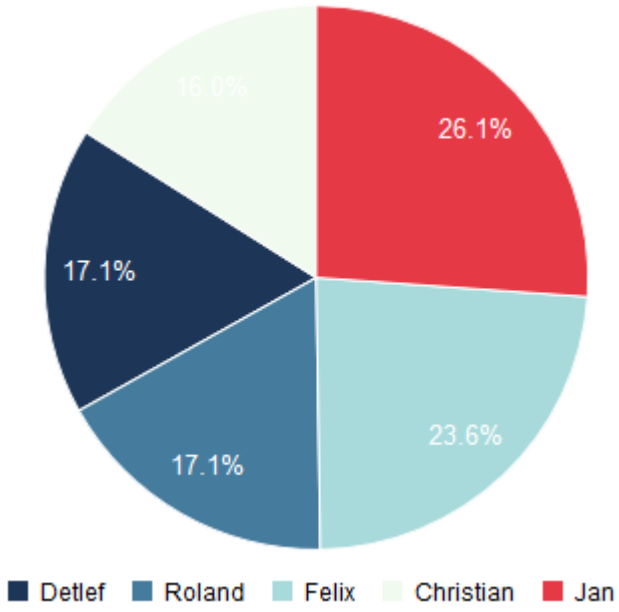
Chart Padding (Java)

```
dashlet.addOption("chart.padding.top", "50");
dashlet.addOption("chart.padding.bottom", "30");
dashlet.addOption("chart.padding.left", "200");
dashlet.addOption("chart.padding.right", "100");
```



2.13.3 Color Pattern

```
dashlet.addOption("chart.color.theme", "royale");
```



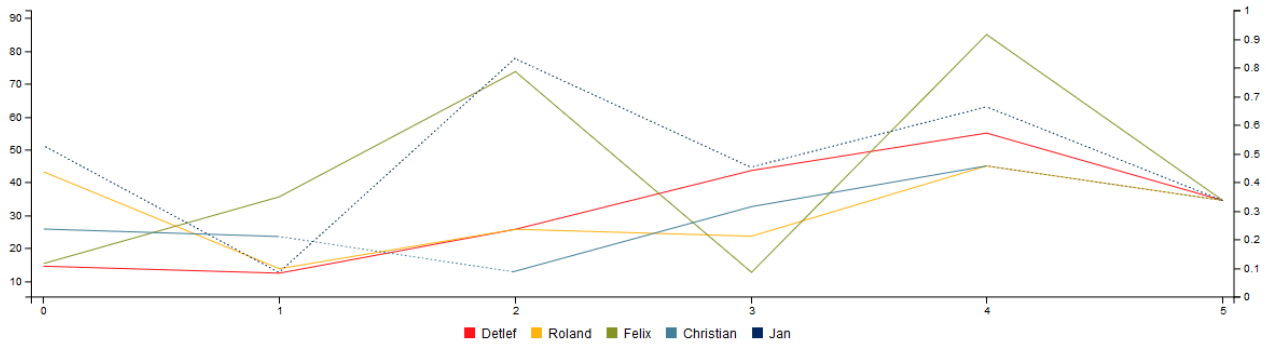
2.13.4 Duration of Transition

```
dashlet.addOption("chart.transition.duration", "300");
```

2.14 # Line und Scatter Chart Options

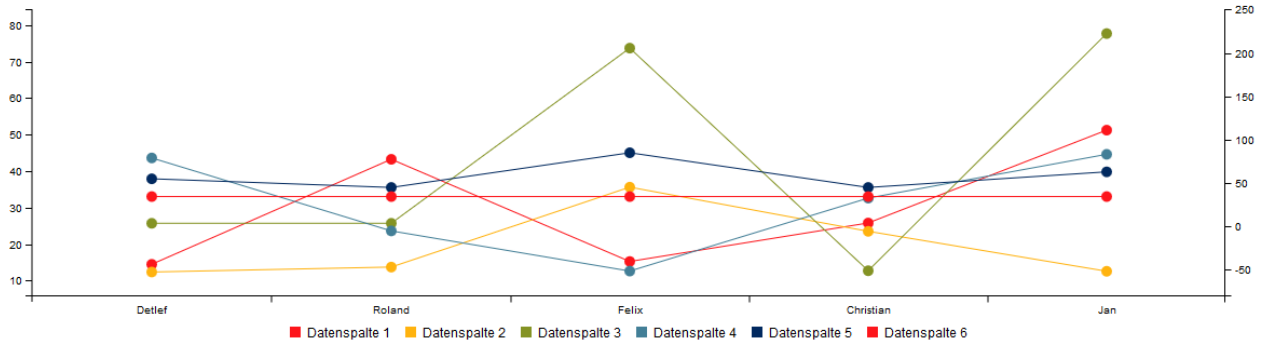
2.14.1 Hide points

```
dashlet.addOption("chart.point.show", "hidden");
```



2.14.2 Point Size

```
dashlet.addOption("chart.point.size", "10");
```

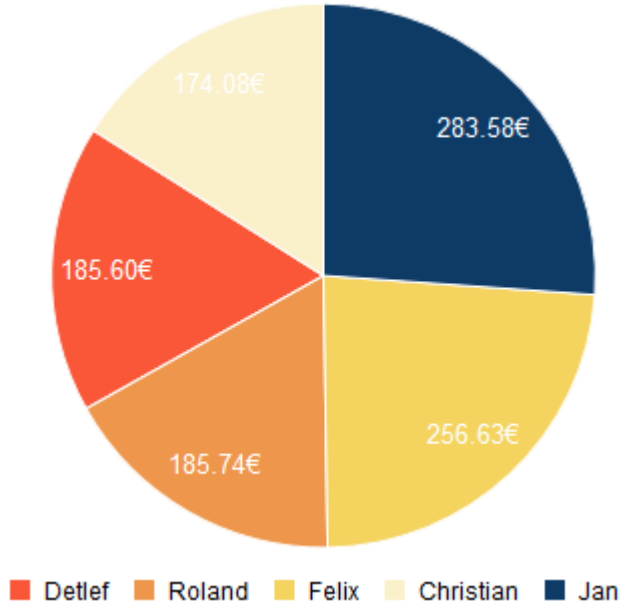


2.15 # Pie Chart Options

2.15.1 Pie Label Format

Pie Label Format (Java)

```
dashlet.addOption("chart.pie.label.format", "(d3.format(',.2f')(value) + 'u20AC')");
dashlet.addOption("chart.pie.label.format", "(id + ': ' + d3.format(',.1f')(ratio * 100) + '%')");
```

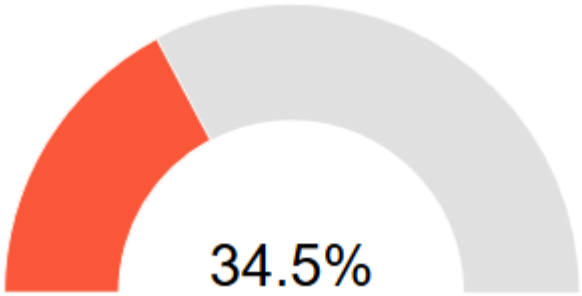


2.16 # Gauge Chart Options

2.16.1 Gauge Label Format

Gauge Label Format (Java)

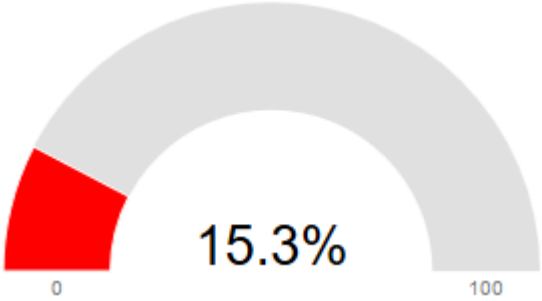
```
dashlet.addOption("chart.gauge.label.show", "hidden");
dashlet.addOption("chart.gauge.label.format", "value");
dashlet.addOption("chart.gauge.units", "in Prozent");
dashlet.addOption("chart.gauge.width", "15");
```



2.16.2 Gauge Color Threshold Levels

Guage Color Threshold Levels (Java)

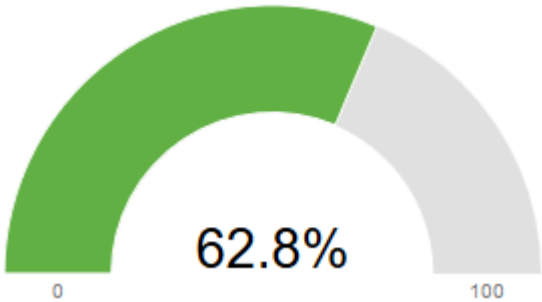
```
dashlet.addOption("chart.gauge.color.value", new ChartColorValue("#FF0000", 40));
dashlet.addOption("chart.gauge.color.value", new ChartColorValue("#F97600", 80));
dashlet.addOption("chart.gauge.color.value", new ChartColorValue("#F6C600", 140));
dashlet.addOption("chart.gauge.color.value", new ChartColorValue("#60B044", 200));
```



2.16.3 Gauge Threshold Options

Gauge Threshold Options (Java)

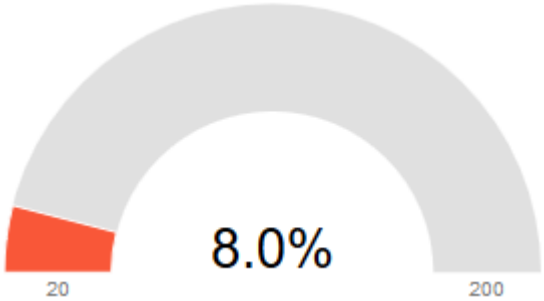
```
dashlet.addOption("chart.gauge.threshold.max", "45");
dashlet.addOption("chart.gauge.threshold.unit", "value"); // percentage is default
```



2.16.4 Gauge Min Max

Gauge Min Max (Java)

```
dashlet.addOption("chart.gauge.min", "20");
dashlet.addOption("chart.gauge.max", "200");
```



2.17 # API

2.17.1 Flow

Load/Unload data as flowing

2.17.2 Data Name

Update data names.

2.17.3 Data Color

Update data colors.

2.17.4 Axis Label

Update axis labels.

2.17.5 Axis Range

Update axis range.

2.17.6 Resize

Resize chart.

2.17.7 X Grid

Update custom x grids.

Copyright © 2023 TRADUI Technologies GmbH

2.18 # Transform

Das Transformieren der Charts wird per JavaScript unterstützt. Der Javascript-Aufruf lautet mit vorangestellter Dashlet-ID im Funktionsnamen:

Transform (JavaScript)

```
To Line Chart

dashletidTransform('pie');

To Spline Chart

dashletidTransform('spline');

To Bar Chart

dashletidTransform('bar');

To Area Chart

dashletidTransform('area');

To Area Spline Chart

dashletidTransform('spline');

To Scatter Plot

dashletidTransform('scatter');

To Pie Chart

dashletidTransform('pie');

To Donut Chart

dashletidTransform('donut');
```

2.19 # Style

2.19.1 Style for Region

Styles für Regionen sollten in einem externen CSS-File definiert werden, welches dem Dashlet hinzugefügt werden muss.
dashlet.addCustomCssFile("css/c3/c3-custom-styles.css", "1000");

CSS:

Style for Region

```
.c3-region.dashletregion1
{
fill: red;
}
.c3-region.dashletregion2
{
fill: blue;;
}
.c3-region.dashletregion3
{
fill: green;
}
.c3-region.dashletregion1 rect
{
fill-opacity: 0.5 !important;
}
.c3-region.dashletregion2 rect
{
fill-opacity: 0.5 !important;
}
.c3-region.dashletregion3 rect
{
fill-opacity: 0.5 !important;
}
```

Hinweis:

Wenn die Einstellungen nur für genau ein Dashlet gelten soll muss dem Stylenamen die Dashlet-ID und die Bezeichnung svg vorangestellt werden.

Beispiel:

Style for Region Dashlet bezogen

```
#dashletid svg .c3-region.dashletregion1
{
fill: red;
}
```

2.19.2 Style for Grid

Styles für Grids und Grid-Markerlinien sollten in einem externen CSS-File definiert werden, welches dem Dashlet hinzugefügt werden muss.
dashlet.addCustomCssFile("css/c3/c3-custom-styles.css", "1000");

CSS:

Style for Grid

```
.c3-xgrid-line.dashletgrid1 line
{
stroke: pink;
}
.c3-xgrid-line.dashletgrid1 text
{
fill: pink;
}
.c3-ygrid-line.dashletgrid2 line
{
stroke: green;
}
.c3-ygrid-line.dashletgrid2 text
{
fill: green;
}
```

Hinweis:

Wenn die Einstellungen nur für genau ein Dashlet gelten sollen, muss dem Style-Namen die Dashlet-ID und die Bezeichnung **svg** vorangestellt werden.

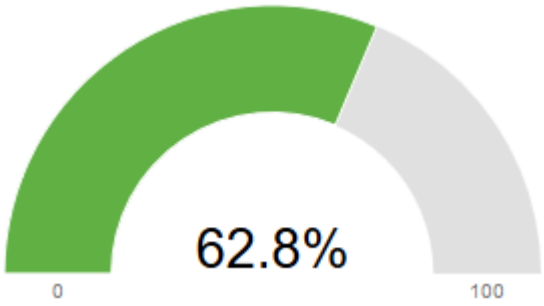
Beispiel:

Style for Grid Dashlet bezogen

```
#dashletid svg .c3-grid line
{
stroke: blue;
}

    1. dashletid svg .c3-grid text
    {
    fill: blue;
    }
```

Copyright © 2023 TRADUI Technologies GmbH



Copyright © 2023 TRADUI Technologies GmbH

3 ECharts

3.1 Einleitung

Dieses Dokument zeigt welcher der EChart-Funktionen derzeit schon von den TRADUI-Dashlets unterstützt werden.

3.2 Ein EChart-Dashlet erzeugen

Erzeugung eines EChart-Dashlets mit der Dashlet-Engine:

Dashlet erzeugen (Java)

```
DashletEngine engine = new DashletEngine();
Dashlet dashlet = engine.createDashlet(DashletType.E_CHART);
```

3.3 Daten ans Dashlet übergeben

Daten werden ans Dashlet als DataTable übergeben, wie bei allen Dashlets. Der Aufbau der Daten-Tabelle wird weiter unten beschrieben. (Muss noch beschrieben werden)

```
dashlet.setData(table);
```

3.4 Hinzufügen von Optionen

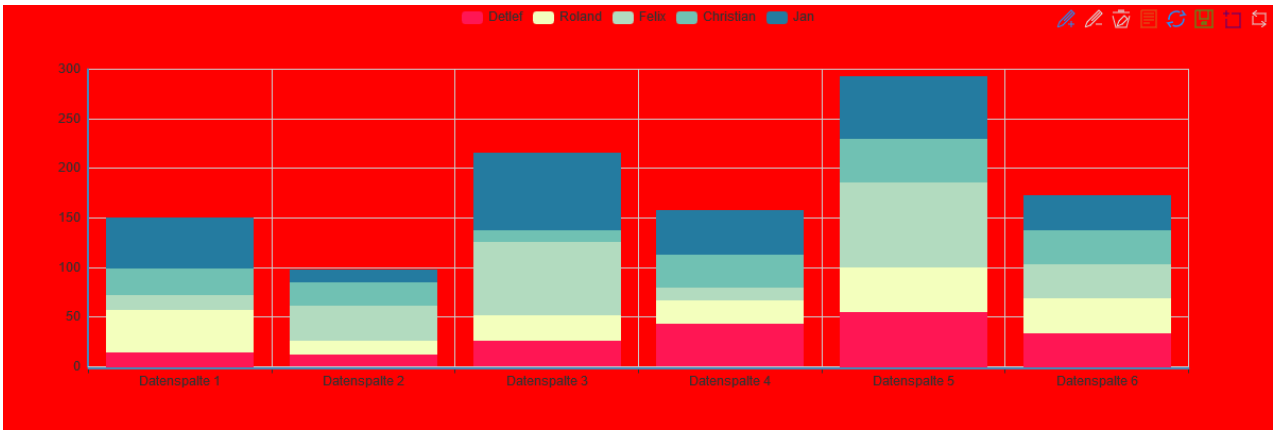
Um das Verhalten der EChart-Dashlets zu beeinflussen, können Optionen übergeben werden.
`dashlet.addOption( OPTIONSKEY , OPTIONSVALUE );`
Options-Keys und mögliche Optionswerte werden im Weiteren in diesem Dokument beschrieben.

3.5 Generelle EChart-Optionen

3.5.1 chart.backgroundColor

Type: **color**
Default Value: **'rgba(0,0,0,0)'**
The background color or gradient for the outer chart area. Support rgba and defaults to null, transprent.

```
dashlet.addOptionForUser("chart.backgroundColor", "#FF0000");
```



3.5.2 chart.animation

Type: **boolean**
Default Value: **true**
specifies whether the initial animation will be enabled, defaults to true. It is suggested to disable the initial animation in IE8-

3.5.3 chart.renderAsImage

Type: boolean, string
Default Value: false
Allows rendering as image in any browser but IE8-.supports render as image in non-IE8- browsers, can be set to true or specify image formats (png | jpeg). After rendered as image, the instance is still available (such as setOption, resize, etc.), but its various interactions will become invalid.

3.5.4 chart.calculable

Type: **boolean**
Default Value: **false**
Specifies whether the drag-recalculate feature will be enabled. Defaults to false. (See calculable, and other related properties like calculableColor, calculableHolderColor, nameConnector, valueConnector).

3.5.5 chart.symbolList

Type: **Array**
Default Value: ['circle', 'rectangle ', 'triangle ', 'diamond ', 'emptyCircle ', 'emptyRectangle ', 'emptyTriangle ', 'emptyDiamond ']
An array containing the default symbols. When all symbols are used, new symbols are pulled from the start again.

3.5.6 chart.calculableColor

Type: **color**
Default Value: **'rgba(255,165,0,0.6)'**
color of border around drag-recalculate

3.5.7 chart.calculableHolderColor

Type: color
Default Value: '#ccc'
color of calculable holder.

3.5.8 chart.nameConnector

Type: string
Default Value: '&'
a connector that links the names of data seriestogether after the combination of data series

3.5.9 chart.valueConnector

Type: string
Default Value: ':'
a connector that links the name of data series with value when island appears after the combination of data series

3.5.10 chart.addDataAnimation

Type: boolean
Default Value: true
specifies whether the dynamic data interface animation will be enabled, defaults to true. It is suggested to disable animation in IE8-

3.5.11 chart.animationThreshold

Type: number
Default Value: 2000
threshold of animated elements. No animation if the graphic elements generated are over 2500. It is suggested to disable animation in IE8-.

3.5.12 chart.animationDuration

Type: number
Default Value: 2000
duration of the enter animation, in ms.

3.5.13 chart.animationDurationUpdate

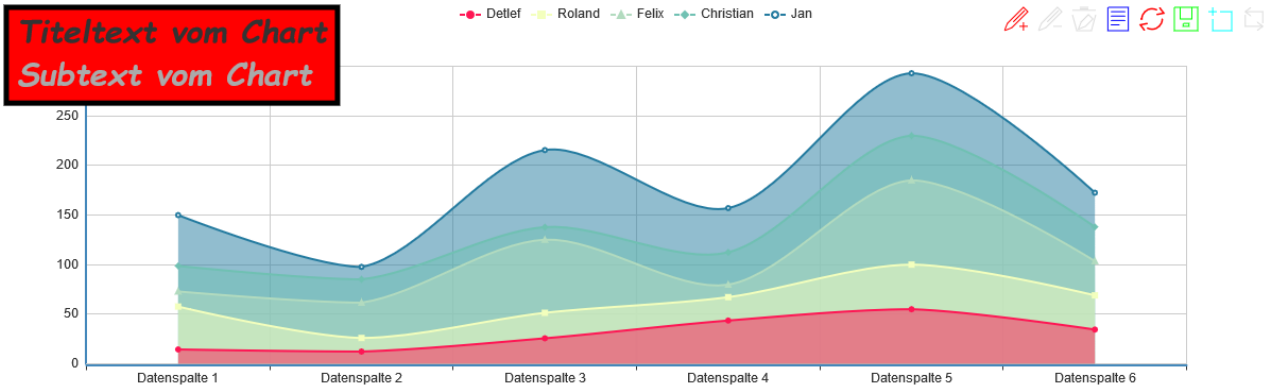
Type: number
Default Value: 500
duration of the update animation, in ms.

3.5.14 chart.animationEasing

Type: string
Default Value: 'ExponentialOut'
easing effect of the main element. Supports multi-level control. Seezrender.animation.easing, Possible values are: 'Linear','QuadraticIn', 'QuadraticOut', 'QuadraticInOut','CubicIn', 'CubicOut', 'CubicInOut','QuarticIn', 'QuarticOut', 'QuarticInOut','QuinticIn', 'QuinticOut', 'QuinticInOut','SinusoidalIn', 'SinusoidalOut', 'SinusoidalInOut','ExponentialIn', 'ExponentialOut', 'ExponentialInOut','CircularIn', 'CircularOut',

'CircularInOut','ElasticIn', 'ElasticOut', 'ElasticInOut','BackIn', 'BackOut', 'BackInOut','BounceIn', 'BounceOut', 'BounceInOut'

3.6 Chart-Titel-Optionen



3.6.1 chart.title.show

Type: boolean
Default Value: true
Specifies whether to show timeline. Valid values are: true (show) | false (hide).

3.6.2 chart.title.zlevel

Type: number
Default Value: 0
The first grade cascading control. Every zlevel will be allocated to a independent canvas, charts and components with the same zlevel will be rendered in the same canvas. The higher zlevel the closer to the top. More canvas dom will need more memory and performance, so never set too many zlevel. Most of the time, the second grade cascading control 'z' is recommended.

3.6.3 chart.title.z

Type: number
Default Value: 6
The second grade cascading control, In the same canvas (zlevel), the higher z the closer to the top.

3.6.4 chart.title.text

Type: string
Default Value: ''
title text, '\n' represents a line feed.

3.6.5 chart.title.link

Type: string
Default Value: ''
title text hyperlink.

3.6.6 chart.title.target

Type: string
Default Value: null
Specifies a window to open the title hyperlink. Can be one of 'self' | 'blank'. If unspecified, equal to 'blank' (a new window).

3.6.7 chart.title.subtext

Type: string
Default Value: ''
subtitle text, '\n' represents a line feed.

3.6.8 chart.title.sublink

Type: string
Default Value: ''
subtitle text hyperlink.

3.6.9 chart.title.subtarget

Type: string
Default Value: null
Specifies a window to open the subtitle hyperlink. Can be one of 'self' | 'blank'. If unspecified, equal to 'blank' (a new window).

3.6.10 chart.title.x

Type: string, number
Default Value: 'left'
horizontal position. Defaults to left. Valid values are: 'center' | 'left' | 'right' | {number}(x-coordinate, in px).

3.6.11 chart.title.y

Type: string, number
Default Value: 'top'
vertical position. Defaults to top. Valid values are: 'top' | 'bottom' | 'center' | {number}(y-coordinate, in px).

3.6.12 chart.title.textAlign

Type: string
Default Value: null
horizontal alignment. It is further determined by the x option by default. Valid values are: left' | 'right' | 'center.

3.6.13 chart.title.backgroundColor

Type: color
Default Value: 'rgba(0,0,0,0)'
The color of the title's background. Defaults to transparent.

Copyright © 2023 TRADUI Technologies GmbH

3.6.14 chart.title.borderColor

Type: string
Default Value: '#ccc'
The color of the drawn border around the title.

3.6.15 chart.title.borderWidth

Type: number
Default Value: 0
The width of the drawn border around the title, in px, defaults to 0 (no border).

3.6.16 chart.title.padding

Type: number, Array
Default Value: 5
The inner padding of the title, in px, defaults to 5. Can be set as array - [top, right, bottom, left], same as css, see image below.

3.6.17 chart.title.itemGap

Type: number
Default Value: 5
The vertical gap between each item in the title/subtitle, in px, defaults to 10.

3.6.18 chart.title.textStyle.fontFamily

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font family.

3.6.19 chart.title.textStyle.fontSize

Type: number
Default Value: 12
font size, in px.

3.6.20 chart.title.textStyle.fontStyle

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

3.6.21 chart.title.textStyle.fontWeight

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

Copyright © 2023 TRADUI Technologies GmbH

3.6.22 chart.title.subtextStyle.fontFamily

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font family.

3.6.23 chart.title.subtextStyle.fontSize

Type: number
Default Value: 12
font size, in px.

3.6.24 chart.title.subtextStyle.fontStyle

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

3.6.25 chart.title.subtextStyle.fontWeight

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

3.7 Chart-Toolbox-Optionen

Die EChart-Toolbox stellt verschiedene Features bereit. Zunächst werden allgemeine Einstellungen zur Toolbox aufgelistet. Danach folgen die einzelnen Features in separaten Abschnitten.

3.7.1 chart.toolbox.show

Type: boolean
Default Value: false
Specifies whether to show toolbox. Valid values are: true (show) | false (hide).

3.7.2 chart.toolbox.zlevel

Type: number
Default Value: 0
The first grade cascading control. Every zlevel will be allocated to a independent canvas, charts and components with the same zlevel will be rendered in the same canvas. The higher zlevel the closer to the top. More canvas dom will need more memory and performance, so never set too many zlevel. Most of the time, the second grade cascading control 'z' is recommended.

3.7.3 chart.toolbox.z

Type: number
Default Value: 6
The second grade cascading control, In the same canvas (zlevel), the higher z the closer to the top.

Copyright © 2023 TRADUI Technologies GmbH

3.7.4 chart.toolbox.orient

Type: string
Default Value: 'horizontal'
layout manner. Defaults to horizontal. Valid values are: 'horizontal' | 'vertical'.

3.7.5 chart.toolbox.x

Type: string, number
Default Value: 'right'
horizontal position. Defaults to center. Valid values are: 'center' | 'left' | 'right' | {number}(x-coordinate, in px).

3.7.6 chart.toolbox.y

Type: string, number
Default Value: 'top'
vertical position. Defaults to top. Valid values are: 'top' | 'bottom' | 'center' | {number}(y-coordinate, in px).

3.7.7 chart.toolbox.backgroundColor

Type: color
Default Value: 'rgba(0,0,0,0)'
The color of the toolbox's background. Defaults to transparent.

3.7.8 chart.toolbox.borderColor

Type: color
Default Value: '#ccc'
The color of the drawn border around the toolbox.

3.7.9 chart.toolbox.borderWidth

Type: number
Default Value: 0
The width of the drawn border around the toolbox, in px, defaults to 0 (no border).

3.7.10 chart.toolbox.padding

Type: number, Array
Default Value: 5
The inner padding of the toolbox, in px, defaults to 5. Can be set as array - [top, right, bottom, left], same as css, see image below.

3.7.11 chart.toolbox.itemGap

Type: number
Default Value: 10
The pixel gap between each item. Defaults to 10. It is horizontal in a toolbox with horizontal layout, and vertical in a toolbox with vertical layout. See image below.

Copyright © 2023 TRADUI Technologies GmbH

3.7.12 chart.toolbox.itemSize

Type: number
Default Value: 16
the size of toolbox icon, in px.

3.7.13 chart.toolbox.color

Type: Array
Default Value: ['#1e90ff','#22bb22','#4b0082','#d2691e']
An array containing the default colors for the toolbox's icons. When all colors are used, new colors are pulled from the start again. Default colors can also be set on specific features.

3.7.14 chart.toolbox.disableColor

Type: color
Default Value: '#ddd'
Specifies the disable color.

3.7.15 chart.toolbox.effectiveColor

Type: color
Default Value: 'red'
Specifies the effective color.

3.7.16 chart.toolbox.showTitle

Type: boolean
Default Value: true
Specifies whether the toolbox text will be showed. Defaults to true.

3.7.17 chart.toolbox.textStyle.fontFamily

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font family.

3.7.18 chart.toolbox.textStyle.fontSize

Type: number
Default Value: 12
font size, in px.

3.7.19 chart.toolbox.textStyle.fontStyle

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

Copyright © 2023 TRADUI Technologies GmbH

3.7.20 chart.toolbox.textStyle.fontWeight

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

3.8 Chart-Toolbox-Feature Mark

Mit dem Toolbox-Feature Mark lassen sich Markierungslinien in den Charts hinzufügen und wieder entfernen.

3.8.1 chart.toolbox.feature.mark.show

Type: boolean
Default Value: false
Zeigt das Mark-Feature an.

3.8.2 chart.toolbox.feature.mark.title.mark

Type: string
Default Value '辅助线开关'
Der Default-Titel des EChart-Frameworks ist Chinesisch. Im EChart-Dashlet, werden die chinesischen Titel automatisch durch deutsche Titel ersetzt. Kann mit dieser Option überschrieben werden.

3.8.3 chart.toolbox.feature.mark.title.markUndo

Type: string
Default Value: '删除辅助线'
Der Default-Titel des EChart-Frameworks ist Chinesisch. Im EChart-Dashlet, werden die chinesischen Titel automatisch durch deutsche Titel ersetzt. Kann mit dieser Option überschrieben werden.

3.8.4 chart.toolbox.feature.mark.title.markClear

Type: string
Default Value: '清空辅助线'
Der Default-Titel des EChart-Frameworks ist Chinesisch. Im EChart-Dashlet, werden die chinesischen Titel automatisch durch deutsche Titel ersetzt. Kann mit dieser Option überschrieben werden.

3.8.5 chart.toolbox.feature.mark.lineStyle.color

Type: color
Default Value: '#1e90ff'
color

3.8.6 chart.toolbox.feature.mark.lineStyle.type

Type: string
Default Value: 'dashed'
line style. Valid values are: 'solid' | 'dotted' | 'dashed', and can be 'curve' | 'broken' in tree structures.

Copyright © 2023 TRADUI Technologies GmbH

3.8.7 chart.toolbox.feature.mark.lineStyle.width

Type: number
Default Value: 2
width of the line.

3.8.8 chart.toolbox.feature.mark.lineStyle.shadowColor (nicht sichtbar)

Type: color
Default Value: 'rgba(0,0,0,0)'
applicable to the main line (in IE8+). Color of the shadow. Supports rgba.

3.8.9 chart.toolbox.feature.mark.lineStyle.shadowBlur (nicht sichtbar)

Type: number
Default Value: 5
applicable to the main line (in IE8+). Blur degree of the shadow. Valid values are positive.

3.8.10 chart.toolbox.feature.mark.lineStyle.shadowOffsetX (nicht sichtbar)

Type: number
Default Value: 3
applicable to the main line (in IE8+). Horizontal offset of the shadow. Right when positive, left when negative.

3.8.11 chart.toolbox.feature.mark.lineStyle.shadowOffsetY (nicht sichtbar)

Type: number
Default Value: 3
applicable to the main line (in IE8+). Vertical offset of the shadow. Down when positive, up when negative.

3.9 Chart-Toolbox-Feature dataZoom

Mit dem Toolbox-Feature dataZoom kann man in Daten-Ausschnitte des Charts hineinzoomen.

3.9.1 chart.toolbox.feature.dataZoom.show

Type: boolean
Default Value: false
Zeigt die dataZoom-Icons in der Toolbox an und aktiviert das Feature.

3.9.2 chart.toolbox.feature.dataZoom.title.dataZoom

Type: string
Default Value: '区域缩放'
Der Default-Titel des EChart-Frameworks ist Chinesisch. Im EChart-Dashlet, werden die chinesischen Titel automatisch durch deutsche Titel ersetzt. Kann mit dieser Option überschrieben werden.

3.9.3 chart.toolbox.feature.dataZoom.title.dataZoomReset

Type: string
Default Value: '区域缩放后退'

Der Default-Titel des EChart-Frameworks ist Chinesisch. Im EChart-Dashlet, werden die chinesischen Titel automatisch durch deutsche Titel ersetzt. Kann mit dieser Option überschrieben werden.

3.10 Chart-Toolbox-Feature dataView

Mit dem Toolbox-Feature dataView kann man die Daten-Ansicht des Charts aktivieren und diese wenn erlaubt verändern, kopieren und aktualisieren.

3.10.1 chart.toolbox.feature.dataView.show

Type: boolean
Default Value: false
Zeigt das dataView-Icon in der Toolbox an und aktiviert das Feature.

3.10.2 chart.toolbox.feature.dataView.title

Type: string
Default Value: '数据视图'
Der Default-Titel des EChart-Frameworks ist Chinesisch. Im EChart-Dashlet, werden die chinesischen Titel automatisch durch deutsche Titel ersetzt. Kann mit dieser Option überschrieben werden.

3.10.3 chart.toolbox.feature.dataView.readOnly

Type: boolean
Default Value: false
dataView is set to read-only by default. It can be edited when readOnly is false.

3.10.4 chart.toolbox.feature.dataView.lang

Type: Array
Default Value: ["数据视图", "关闭", "刷新"]
DataView has three default texts: ['Data View', 'close', 'refresh'], you can change it if you want.
Die Default-Titel des EChart-Frameworks ist Chinesisch. Im EChart-Dashlet, werden die chinesischen Titel automatisch durch deutsche Titel ersetzt. Kann mit dieser Option überschrieben werden.

3.10.5 chart.toolbox.feature.dataView.optionToContent (noch nicht implementiert)

Type: Function
Default Value:
custom display content of dataView.

3.10.6 chart.toolbox.feature.dataView.contentToOption (noch nicht implementiert)

Type: Function
Default Value:
When the readOnly of dataView is set to false, the refresh button will appear. You can set the content and the way to display it yourself if you want.

3.11 Chart-Toolbox-Feature magicType

Mit dem Toolbox-Feature magicType kann man den Typ des Charts auf „magische" Weise umschalten.

3.11.1 chart.toolbox.feature.magicType.show

Type: boolean
Default Value: false
Zeigt das magicType-Icon in der Toolbox an und aktiviert das Feature.

3.11.2 chart.toolbox.feature.magicType.type

Type: Array
Default Value: []
'line', 'bar', 'stack', 'tiled', 'force', 'chord', 'pie', 'funnel'

3.11.3 chart.toolbox.feature.magicType.title.line

Type: string
Default Value: '折线图切换'
Der Default-Titel des EChart-Frameworks ist Chinesisch. Im EChart-Dashlet, werden die chinesischen Titel automatisch durch deutsche Titel ersetzt. Kann mit dieser Option überschrieben werden.

3.11.4 chart.toolbox.feature.magicType.title.bar

Type: string
Default Value: '柱形图切换'
Der Default-Titel des EChart-Frameworks ist Chinesisch. Im EChart-Dashlet, werden die chinesischen Titel automatisch durch deutsche Titel ersetzt. Kann mit dieser Option überschrieben werden.

3.11.5 chart.toolbox.feature.magicType.title.stack

Type: string
Default Value: '堆积'
Der Default-Titel des EChart-Frameworks ist Chinesisch. Im EChart-Dashlet, werden die chinesischen Titel automatisch durch deutsche Titel ersetzt. Kann mit dieser Option überschrieben werden.

3.11.6 chart.toolbox.feature.magicType.title.tiled

Type: string
Default Value: '平铺'
Der Default-Titel des EChart-Frameworks ist Chinesisch. Im EChart-Dashlet, werden die chinesischen Titel automatisch durch deutsche Titel ersetzt. Kann mit dieser Option überschrieben werden.

3.11.7 chart.toolbox.feature.magicType.title.force

Type: string
Default Value: '力导向布局图切换'
Der Default-Titel des EChart-Frameworks ist Chinesisch. Im EChart-Dashlet, werden die chinesischen Titel automatisch durch deutsche Titel ersetzt. Kann mit dieser Option überschrieben werden.

Copyright © 2023 TRADUI Technologies GmbH

3.11.8 chart.toolbox.feature.magicType.title.chord

Type: string
Default Value: '和弦图切换'
Der Default-Titel des EChart-Frameworks ist Chinesisch. Im EChart-Dashlet, werden die chinesischen Titel automatisch durch deutsche Titel ersetzt. Kann mit dieser Option überschrieben werden.

3.11.9 chart.toolbox.feature.magicType.title.pie

Type: string
Default Value: '饼图切换'
Der Default-Titel des EChart-Frameworks ist Chinesisch. Im EChart-Dashlet, werden die chinesischen Titel automatisch durch deutsche Titel ersetzt. Kann mit dieser Option überschrieben werden.

3.11.10 chart.toolbox.feature.magicType.title.funnel

Type: string
Default Value: '漏斗图切换'
Der Default-Titel des EChart-Frameworks ist Chinesisch. Im EChart-Dashlet, werden die chinesischen Titel automatisch durch deutsche Titel ersetzt. Kann mit dieser Option überschrieben werden.

3.11.11 chart.toolbox.feature.magicType.option.line

Type: Object
Default Value

3.11.12 chart.toolbox.feature.magicType.option.bar

Type: Object
Default Value

3.11.13 chart.toolbox.feature.magicType.option.stack

Type: Object
Default Value

3.11.14 chart.toolbox.feature.magicType.option.tiled

Type: Object
Default Value

3.11.15 chart.toolbox.feature.magicType.option.force

Type: Object
Default Value

3.11.16 chart.toolbox.feature.magicType.option.chord

Type: Object
Default Value

Copyright © 2023 TRADUI Technologies GmbH

3.11.17 chart.toolbox.feature.magicType.option.pie

Type: Object
Default Value

3.11.18 chart.toolbox.feature.magicType.option.funnel

Type: Object
Default Value

3.12 Chart-Toolbox-Feature Restore

Mit dem Toolbox-Feature Restore lassen sich die ursprünglichen Einstellungen des Charts wiederherstellen, wenn man sie verändert hat.

3.12.1 chart.toolbox.feature.restore.show

Type: boolean
Default Value: false
Zeigt das Restore-Icon in der Toolbox an und aktiviert das Feature.

3.12.2 chart.toolbox.feature.restore.title

Type: string
Default Value: '还原'
Der Default-Titel des EChart-Frameworks ist Chinesisch. Im EChart-Dashlet, werden die chinesischen Titel automatisch durch deutsche Titel ersetzt. Kann mit dieser Option überschrieben werden.

3.13 Chart-Toolbox-Feature saveAsImage

Mit dem Toolbox-Feature saveAsImage lassen sich die Charts als Bild abspeichern.

3.13.1 chart.toolbox.feature.saveAsImage.show

Type: boolean
Default Value: false
Zeigt das saveAsImage -Icon in der Toolbox an und aktiviert das Feature.

3.13.2 chart.toolbox.feature.saveAsImage.title

Type: string
Default Value: '保存为图片'
Der Default-Titel des EChart-Frameworks ist Chinesisch. Im EChart-Dashlet, werden die chinesischen Titel automatisch durch deutsche Titel ersetzt. Kann mit dieser Option überschrieben werden.

3.13.3 chart.toolbox.feature.saveAsImage.type

Type: string
Default Value: 'png'
the default type to save image is 'png', change it to 'jpeg'.

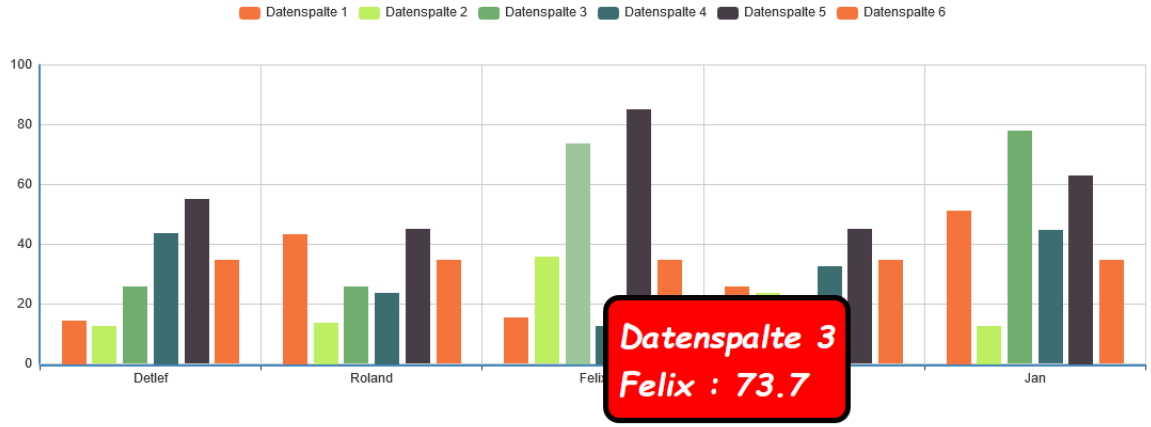
3.13.4 chart.toolbox.feature.saveAsImage.name

Type: string
Default Value: 'ECharts'
specifies name of the image. If unspecified, use the chart title. If there is no title, defaults to "ECharts".

3.13.5 chart.toolbox.feature.saveAsImage.lang

Type: Array
Default Value: ["点击保存"]
you can click to save it in non-IE browsers. Save text is supported. Defaults to "click Save", can be changed.
Der Default-Titel des EChart-Frameworks ist Chinesisch. Im EChart-Dashlet, werden die chinesischen Titel automatisch durch deutsche Titel ersetzt. Kann mit dieser Option überschrieben werden.

3.14 Chart-Tooltip-Optionen



3.14.1 chart.tooltip.show

Type: boolean
Default Value: true
Specifies whether to show tooltip. Valid values are: true (show) | false (hide).

3.14.2 chart.tooltip.zlevel

Type: number
Default Value: 1
The first grade cascading control. Every zlevel will be allocated to a independent canvas, charts and components with the same zlevel will be rendered in the same canvas. The higher zlevel the closer to the top. More canvas dom will need more memory and performance, so never set too many zlevel. Most of the time, the second grade cascading control 'z' is recommended.

3.14.3 chart.tooltip.z

Type: number
Default Value: 8
The second grade cascading control, In the same canvas (zlevel), the higher z the closer to the top.

3.14.4 chart.tooltip.showContent

Type: boolean
Default Value: true
Specifies whether to show the content of tooltip. Set it to false when you just need the tooltip to trigger events or show axisPointer.
Valid values are: true (show) | false (hide).

3.14.5 chart.tooltip.trigger

Type: string
Default Value: 'item'
Type of trigger. Defaults to 'item'. See image below. Valid values are: 'item' | 'axis'.

3.14.6 chart.tooltip.position

Type: Array, Function
Default Value: null
Specifies position, pass in {Array}, like [x, y], fixed position [x, y]; pass in {Function}, like function([x, y]) {return [newX,newY]}. The default displayed coordinates are input parameters, the new user-specified coordinates are output return values.

3.14.7 chart.tooltip.formatter (nicht getestet)

Type: string, Function
Default Value: null
content formatter: {string} (Template) | {Function}. Supports asynchronous callback. See below.
{string}, Template, its variables are:
{a} | {a0}
{b} | {b0}
{c} | {c0}
{d} | {d0} (some chart types do not have this variable)
{e} | {e0} (some chart types do not have this variable)
If there are multiple sets of data, there will be multiple sets of varaibles accordingly, like {a1}, {b1}, {c1}, {d1}, {a2}, {b2}, {c2}, {d2}, ...
Here are what variable a, b, c and d reprensent in different chart types:
line & area, column & bar, candlestick: a (series name), b(category value), c (value), d(null)
scatter & bubble: a (series name), b (data name), c (value array), d(null)
map: a (series name), b (area name), c(merged value), d(null)
pie, radar, gauge, funnel: a (series name), b (data item name), c(value), d (pie: percent | radar: indicator name)
force, chord :
node : a (series name), b (node name), c (node value), d (node index), e
link : a (series name), b (link name, default bigEndNodeName-smallEndNodeName), c(link.value), d(name or index of big end node), e(name or index of small end node)
{Function}, the list of parameters passed in is [params, ticket, callback], detail as follows:
params : array content and template varaibles are the same

Chart Tooltip Optionen (Java)

```
[
{
seriesIndex: 0,
seriesName: 'Sales',
name: 'Monday',
dataIndex: 0,
data: data,
name: name,
value: value,
percent: special, // pie
indicator: special, // radar, force, chord
value2: special2, // force, chord
indicator2: special2 // force, chord
},
{..},
...
]
```

ticket: asynchronous callback marker
callback: asynchronous callback, the callback needs two parameters, the first one is the ticket that we mentioned earlier, the second is the fill content html.

- When the function callbacks, 'this' pointer points to the current chart instance (myChart).

3.14.8 chart.tooltip.islandFormatter (nicht getestet)

Type: string, Function
Default Value: '{a} < br/>{b} : {c}'
island content formatter. For Drag-Recalculate only: {string} (Template) | {Function}. See below.

3.14.9 chart.tooltip.showDelay

Type: number
Default Value: 20
The number of milliseconds to wait until the tooltip is shown when the mouse moves across a point or chart. Placing a delay on a tooltip before it is displayed can prevent frequent switch, especially when the tooltip text is requested asynchronously.

3.14.10 chart.tooltip.hideDelay

Type: number
Default Value: 100
The number of milliseconds to wait until the tooltip is hidden when mouse out from a point or chart.

3.14.11 chart.tooltip.transitionDuration (nicht sichtbar)

Type: number
Default Value: 0.4
The duration in seconds of the animated transition. If you want the tooltip to follow the mouse as it moves across a

point or chart, setting showDelay to 0 is the key, but setting transitionDuration to 0 can also make a difference in interactive experiences.

3.14.12 chart.tooltip.enterable (nicht getestet)

Type: boolean
Default Value: false
Specifies whether to let the mouse go into the tip dom, default to false. Let it be true if you need interaction like link | button etc.

3.14.13 chart.tooltip.backgroundColor

Type: color
Default Value: 'rgba(0,0,0,0.7)'
The background color for the tooltip. Defaults to a black color (opacity to 30%).

3.14.14 chart.tooltip.borderColor

Type: string
Default Value: '#333'
The color of the tooltip border.

3.14.15 chart.tooltip.borderRadius

Type: number
Default Value: 4
The radius of the rounded border corners, in px, defaults to 4.

3.14.16 chart.tooltip.borderWidth

Type: number
Default Value: 0
The pixel width of the tooltip border, defaults to 0 (no border).

3.14.17 chart.tooltip.padding

Type: number, Array
Default Value: 5
The inner padding of the tooltip, in px, defaults to 5. Can be set as array - [top, right, bottom, left], same as css.

3.14.18 chart.tooltip.textStyle.fontFamily

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font family.

3.14.19 chart.tooltip.textStyle.fontSize

Type: number
Default Value: 12
font size, in px.

Copyright © 2023 TRADUI Technologies GmbH

3.14.20 chart.tooltip.textStyle.fontStyle

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

3.14.21 chart.tooltip.textStyle.fontWeight

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

3.14.22 chart.tooltip.axisPointer.type

Type: string
Default Value: 'line'
Possible value: 'line' | 'cross' | 'shadow' | 'none'. Each type has its corresponding style, see below.

3.14.23 chart.tooltip.axisPointer.lineStyle.color

Type: color
Default Value: '#1e90ff'
color

3.14.24 chart.tooltip.axisPointer.lineStyle.type

Type: string
Default Value: 'dashed'
line style. Valid values are: 'solid' | 'dotted' | 'dashed', and can be 'curve' | 'broken' in tree structures.

3.14.25 chart.tooltip.axisPointer.lineStyle.width

Type: number
Default Value: 2
width of the line.

3.14.26 chart.tooltip.axisPointer.lineStyle.shadowColor (nicht sichtbar)

Type: color
Default Value: 'rgba(0,0,0,0)'
applicable to the main line (in IE8+). Color of the shadow. Supports rgba.

3.14.27 chart.tooltip.axisPointer.lineStyle.shadowBlur (nicht sichtbar)

Type: number
Default Value: 5
applicable to the main line (in IE8+). Blur degree of the shadow. Valid values are positive.

Copyright © 2023 TRADUI Technologies GmbH

3.14.28 chart.tooltip.axisPointer.lineStyle.shadowOffsetX (nicht sichtbar)

Type: number
Default Value: 3
applicable to the main line (in IE8+). Horizontal offset of the shadow. Right when positive, left when negative.

3.14.29 chart.tooltip.axisPointer.lineStyle.shadowOffsetY (nicht sichtbar)

Type: number
Default Value: 3
applicable to the main line (in IE8+). Vertical offset of the shadow. Down when positive, up when negative.

3.14.30 chart.tooltip.axisPointer.crossStyle.color

Type: color
Default Value: '#1e90ff'
color

3.14.31 chart.tooltip.axisPointer.crossStyle.type

Type: string
Default Value: 'dashed'
line style. Valid values are: 'solid' | 'dotted' | 'dashed', and can be 'curve' | 'broken' in tree structures.

3.14.32 chart.tooltip.axisPointer.crossStyle.width

Type: number
Default Value: 2
width of the line.

3.14.33 chart.tooltip.axisPointer.crossStyle.shadowColor (nicht sichtbar)

Type: color
Default Value: 'rgba(0,0,0,0)'
applicable to the main line (in IE8+). Color of the shadow. Supports rgba.

3.14.34 chart.tooltip.axisPointer.crossStyle.shadowBlur (nicht sichtbar)

Type: number
Default Value: 5
applicable to the main line (in IE8+). Blur degree of the shadow. Valid values are positive.

3.14.35 chart.tooltip.axisPointer.crossStyle.shadowOffsetX (nicht sichtbar)

Type: number
Default Value: 3
applicable to the main line (in IE8+). Horizontal offset of the shadow. Right when positive, left when negative.

Copyright © 2023 TRADUI Technologies GmbH

3.14.36 chart.tooltip.axisPointer.crossStyle.shadowOffsetY (nicht sichtbar)

Type: number
Default Value: 3
applicable to the main line (in IE8+). Vertical offset of the shadow. Down when positive, up when negative.

3.14.37 chart.tooltip.axisPointer.shadowStyle.color (nicht sichtbar)

Type: color
Default Value: '#1e90ff'
color

3.14.38 chart.tooltip.axisPointer.shadowStyle.type (nicht sichtbar)

Type: string
Default Value: 'default'
Fill style, currently only supports 'default' (solid fill).

3.14.39 chart.tooltip.axisPointer.textStyle.fontFamily (nicht testbar)

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font family.

3.14.40 chart.tooltip.axisPointer.textStyle.fontSize (nicht testbar)

Type: number
Default Value: 12
font size, in px.

3.14.41 chart.tooltip.axisPointer.textStyle.fontStyle (nicht testbar)

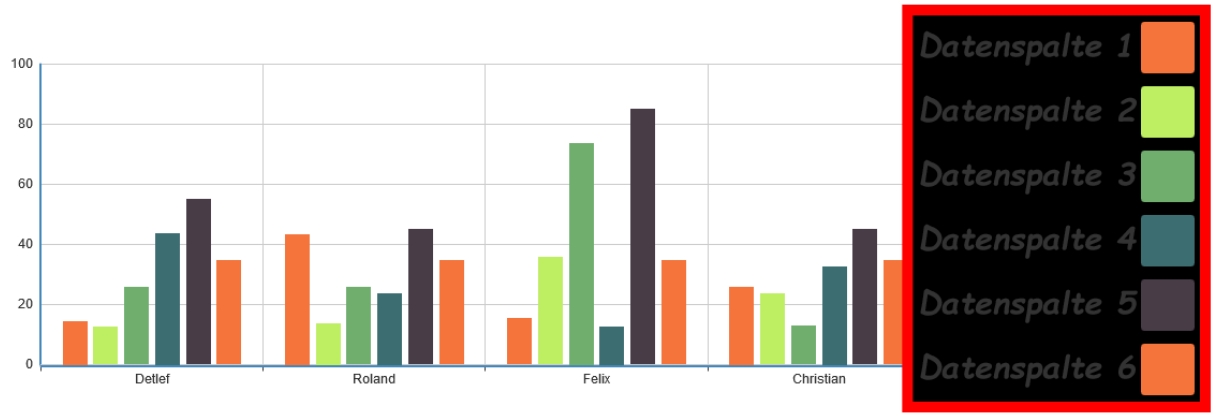
Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

3.14.42 chart.tooltip.axisPointer.textStyle.fontWeight (nicht testbar)

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

Copyright © 2023 TRADUI Technologies GmbH

3.15 Chart-Legend-Optionen



3.15.1 chart.legend.show

Type: boolean
Default Value: true
Specifies whether to show. Valid values are: true (show) | false (hide).

3.15.2 chart.legend.zlevel

Type: number
Default Value: 0
The first grade cascading control. Every zlevel will be allocated to a independent canvas, charts and components with the same zlevel will be rendered in the same canvas. The higher zlevel the closer to the top. More canvas dom will need more memory and performance, so never set too many zlevel. Most of the time, the second grade cascading control 'z' is recommended.

3.15.3 chart.legend.z

Type: number
Default Value: 4
The second grade cascading control, In the same canvas (zlevel), the higher z the closer to the top.

3.15.4 chart.legend.orient

Type: string
Default Value: 'horizontal'
layout manner, defaults to horizontal. Valid values are: 'horizontal' | 'vertical'.

3.15.5 chart.legend.x

Type: string, number
Default Value: 'center'
horizontal position. Defaults to center. Valid values are: 'center' | 'left' | 'right' | {number}}(x-coordinate, in px).

Copyright © 2023 TRADUI Technologies GmbH

3.15.6 chart.legend.y

Type: string, number
Default Value: 'top'
vertical position. Defaults to top. Valid values are: 'top' | 'bottom' | 'center' | {number}(y-coordinate, in px).

3.15.7 chart.legend.backgroundColor

Type: color
Default Value: 'rgba(0,0,0,0)'
The color of the legend's background. Defaults to transparent.

3.15.8 chart.legend.borderColor

Type: string
Default Value: '#ccc'
The color of the drawn border around the legend.

3.15.9 chart.legend.borderWidth

Type: number
Default Value: 0
The width of the drawn border around the legend, in px, defaults to 0 (no border).

3.15.10 chart.legend.padding

Type: number, Array
Default Value: 5
The inner padding of the legend, in px, defaults to 5. Can be set as array - [top, right, bottom, left], same as css, see image below.

3.15.11 chart.legend.itemGap

Type: number
Default Value: 10
The pixel gap between each item. Defaults to 10. It is horizontal in a legend with horizontal layout, and vertical in a legend with vertical layout. See image below.

3.15.12 chart.legend.itemWidth

Type: number
Default Value: 20
Legend shape's width.

3.15.13 chart.legend.itemHeight

Type: number
Default Value: 14
Legend shape's height.

Copyright © 2023 TRADUI Technologies GmbH

3.15.14 chart.legend.textStyle.fontFamily

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font family.

3.15.15 chart.legend.textStyle.fontSize

Type: number
Default Value: 12
font size, in px.

3.15.16 chart.legend.textStyle.fontStyle

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

3.15.17 chart.legend.textStyle.fontWeight

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

3.15.18 chart.legend.formatter (nicht getestet)

Type: string, Function
Default Value: null
text formatter: {string} (Template) | {Function}, template variable is '{name}', function paramater is name.

3.15.19 chart.legend.selectedMode

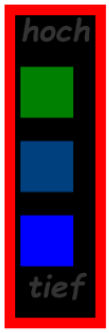
Type: boolean, string
Default Value: true
selection mode. Enable legend switch by default. Can be one of: single, multiple

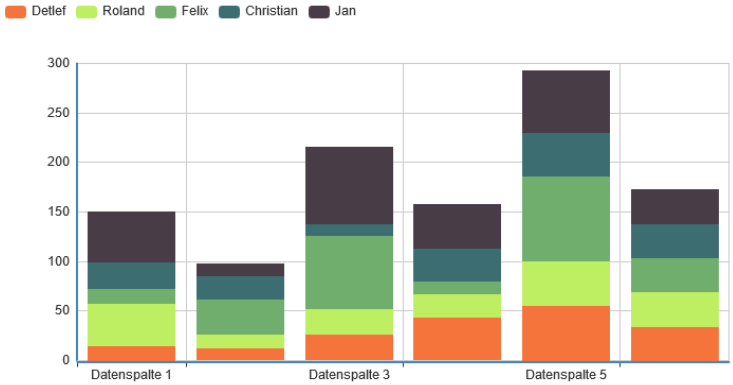
3.15.20 chart.legend.selected (nicht getestet)

Type: Object
Default Value: null
The default selected state, can be used to conduct dynamic data loading with LEGEND.SELECTED event.

Copyright © 2023 TRADUI Technologies GmbH

3.16 Chart-dataRange-Optionen





3.16.1 chart.dataRange.show

Type: boolean
Default Value: true
Specifies whether to show. Valid values are: true (show) | false (hide).

3.16.2 chart.dataRange.zlevel

Type: number
Default Value: 0
The first grade cascading control. Every zlevel will be allocated to a independent canvas, charts and components with the same zlevel will be rendered in the same canvas. The higher zlevel the closer to the top. More canvas dom will need more memory and performance, so never set too many zlevel. Most of the time, the second grade cascading control 'z' is recommended.

3.16.3 chart.dataRange.z

Type: number
Default Value: 4
The second grade cascading control, In the same canvas (zlevel), the higher z the closer to the top.

3.16.4 chart.dataRange.orient

Type: string
Default Value: 'vertical'
layout manner. Defaults to vertical. Valid values are: 'horizontal' | 'vertical'

3.16.5 chart.dataRange.x

Type: string, number
Default Value: 'left'
horizontal position. Defaults to left. Valid values are: 'center' | 'left' | 'right' | {number}}(x-coordinate, in px).

Copyright © 2023 TRADUI Technologies GmbH

3.16.6 chart.dataRange.y

Type: string, number
Default Value: 'bottom'
vertical position. Defaults to bottom. Valid values are: 'top' | 'bottom' | 'center' | {number}(y-coordinate, in px).

3.16.7 chart.dataRange.backgroundColor

Type: color
Default Value: 'rgba(0,0,0,0)'
The color of the dataRange's background. Defaults to transparent.

3.16.8 chart.dataRange.borderColor

Type: string
Default Value: '#ccc'
The color of the drawn border around the dataRange.

3.16.9 chart.dataRange.borderWidth

Type: number
Default Value: 0
The width of the drawn border around the dataRange, in px, defaults to 0 (no border).

3.16.10 chart.dataRange.padding

Type: number, Array
Default Value: 5
The inner padding of the dataRange, in px, defaults to 5. Can be set as array - [top, right, bottom, left], same as css, see image below.

3.16.11 chart.dataRange.itemGap

Type: number
Default Value: 10
The pixel gap between each item. Defaults to 10. It is horizontal in the dataRange with horizontal layout, and vertical in the dataRange with vertical layout. See image below.

3.16.12 chart.dataRange.itemWidth

Type: number
Default Value: 20
The width of dataRange item.

3.16.13 chart.dataRange.itemHeight

Type: number
Default Value: 14
The width of dataRange item.

Copyright © 2023 TRADUI Technologies GmbH

3.16.14 chart.dataRange.min

Type: number
Default Value: null
The specified minimum value, such as 0. Defaults to null. Required parameter. Only when splitList is used, min can be ignored.

3.16.15 chart.dataRange.max

Type: number
Default Value: null
The specified maximum value, such as 100. Defaults to null. Required parameter. Only when splitList is used, max can be ignored.

3.16.16 chart.dataRange.precision

Type: number
Default Value: 0
Decimal precision. Defaults to 0 (no decimal point). If the value of max - min is not divisible by splitNumber in the existing precision, the precision will be automatically increased for the sake of exact division. Inexact division is not supported.

3.16.17 chart.dataRange.splitNumber

Type: number
Default Value: 5
the number of segments. Defaults to 5. Linear gradient when set to 0. When calculable is true, it is equally split into 100 parts by default.

3.16.18 chart.dataRange.splitList

Type: Array
Default Value: null
Customizing that how to split dataRange. When splitList is specified, splitNumber is ignored.
splitList is an Array, which can not be empty. Each item of the Array is Object like:
{
start: 10 The start of the domain of this item.
end: 30 The end of the domain of this item.
'start' and 'end' can be set to the same value,
in which case this item is coresponding to the single value rather than an interval.
label: '10 to 30' text label of this item. If ignoring this attribute, Text label will be auto generated.
color: '#333' Color of this item. If ignoring attribute, color can be auto generated.
}
example:

Chart DataRange SplitList (Java)

```
dashlet.addOption("chart.dataRange.splitList", "[{start: 10, end: 30, label: '10 to
30', color: 'red'}, {start: 40, end: 70, label: '40 to 70', color: 'blue'}]")
```

10 to 30
40 to 70

3.16.19 chart.dataRange.range (nicht getestet)

Type: Object
Default Value: null
This option is used to set the initial selected range of dataRange. This option works when "calculable" is true.
The value of this option likes: {start: 10, end: 50}

3.16.20 chart.dataRange.selectedMode

Type: boolean, string
Default Value: true
selection mode. Enable dataRange switch by default. Can be one of: single, multiple

3.16.21 chart.dataRange.calculable

Type: boolean
Default Value: false
Specifies whether the scale roaming feature will be enabled. When enabled, ignore splitNumber and splitList, dataRange shows as linear gradient.

3.16.22 chart.dataRange.hoverLink (nicht getestet)

Type: boolean
Default Value: true
Hover link with the map.

3.16.23 chart.dataRange.realtime (nicht testbar)

Type: boolean
Default Value: true
Specifies whether the scale roaming feature will be displayed in real time. Default to false (CAN NOT be changed) in the explorer which does't support canvas, such as IE6/7/8.

3.16.24 chart.dataRange.color

Type: Array
Default Value: ["#1e90ff", "#f0ffff"]
color list of the dataRange. Length of the color array must be > = 2. The color shows the change from high value to low value; that is, the low color array represents the high value color. Alpha (rgba) is supported.

3.16.25 chart.dataRange.formatter (nicht getestet)

Type: string, Function
Default Value: null
content formatter：{string} (Template) | {Function}, the template's variables are {value}' and '{value2}', which stand

for start value and end value. There are two function parameters, whose meanings are the same as template variables. One parameter '{value}' only when calculable is true. Try this

3.16.26 chart.dataRange.text

Type: Array
Default Value: null
text of the dataRange, when splitNumber is valid, defaults to the calculated value. You can specify a text array with the length of 2 to show the brief dataRange text, like ['high', 'low']. '\n' represents a line feed.

3.16.27 chart.dataRange.textStyle.fontFamily

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font family.

3.16.28 chart.dataRange.textStyle.fontSize

Type: number
Default Value: 12
font size, in px.

3.16.29 chart.dataRange.textStyle.fontStyle

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

3.16.30 chart.dataRange.textStyle.fontWeight

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

3.17 Chart-dataZoom-Optionen

3.17.1 chart.dataZoom.zlevel

Type: number
Default Value: 0
The first grade cascading control. Every zlevel will be allocated to a independent canvas, charts and components with the same zlevel will be rendered in the same canvas. The higher zlevel the closer to the top. More canvas dom will need more memory and performance, so never set too many zlevel. Most of the time, the second grade cascading control 'z' is recommended.

3.17.2 chart.dataZoom.z

Type: number
Default Value: 4
The second grade cascading control, In the same canvas (zlevel), the higher z the closer to the top.

3.17.3 chart.dataZoom.show

Type: boolean
Default Value: false
Specifies whether to show dataZoom. If true, take over all the series of data using specified category axis. If unspecified, take over all the Cartesian coordinate system data.

3.17.4 chart.dataZoom.orient

Type: string
Default Value: 'horizontal'
layout manner, defaults to horizontal. Valid values are: 'horizontal' | 'vertical'.

3.17.5 chart.dataZoom.x

Type: number
Default Value: 自适应
Horizontal placement. By default it is adapted according to the grid's parameters. Vertical layout is left by default, can be specified. {number} (the x-coordinate, in pixels, of the upper-left corner.)

3.17.6 chart.dataZoom.y

Type: number
Default Value: 自适应
Vertical placement. By default it is adapted according to the grid's parameters. Vertical layout is bottom by default, can be specified. {number} (the y-coordinate, in pixels, of the upper-left corner.)

3.17.7 chart.dataZoom.width (funktioniert nicht)

Type: number
Default Value: 自适应 | 30
Specifies width. In honrizontal layout, it is adapted according to the grid's parameters by default. In vertical layout, defaults to 30, can be specified. {number} (width, in px)

3.17.8 chart.dataZoom.height

Type: number
Default Value: 自适应 | 30
Specifies height. In vertical layout, it is adapted according to the grid's parameters by default. In honrizontal layout, defaults to 30, can be specified. {number} (height, in px)

3.17.9 chart.dataZoom.backgroundColor

Type: color
Default Value: 'rgba(0,0,0,0)'
The color of the background. Defaults to transparent.

Copyright © 2023 TRADUI Technologies GmbH

3.17.10 chart.dataZoom.dataBackgroundColor

Type: color
Default Value: '#eee'
The color of the dataZoom's background. Only the first series would be rendered as data background.

3.17.11 chart.dataZoom.fillerColor

Type: color
Default Value: 'rgba(144,197,237,0.2)'
The fill color of the selected area.

3.17.12 chart.dataZoom.handleColor

Type: color
Default Value: 'rgba(70,130,180,0.8)'
The color of the control handle.

3.17.13 chart.dataZoom.handleSize

Type: number
Default Value: 8
The size of the control handle.

3.17.14 chart.dataZoom.xAxisIndex

Type: Array, number
Default Value: null
If unspecified, it controls by default all the x-axes. You can use array to specify multiple x-axes Index that needs to be controlled, and number to specify one x-axis Index that needs to be controlled.

3.17.15 chart.dataZoom.yAxisIndex (kein Beispiel vorhanden, konnte nicht getestet werden)

Type: Array, number
Default Value: null
If unspecified, it controls by default all the y-axes. You can use array to specify multiple y-axes Index that needs to be controlled, and number to specify one y-axis Index that needs to be controlled.

3.17.16 chart.dataZoom.start

Type: number
Default Value: 0
The start percent of dataZoom. Defaults to 0 (%), starts with the first data.

3.17.17 chart.dataZoom.end

Type: number
Default Value: 100
The end percent of dataZoom. Defaults to 100 (%), ends with the last data.

Copyright © 2023 TRADUI Technologies GmbH

3.17.18 chart.dataZoom.showDetail

Type: boolean
Default Value: true
Show detail when dragging.

3.17.19 chart.dataZoom.realtime

Type: boolean
Default Value: false
Specifies whether dataZoom will be displayed in real time. It is suggested to disable the realtime effect when your browser is slow or data sets are large and complex. Default to false (CAN NOT be changed) in the explorer which does't support canvas, such as IE6/7/8.

3.17.20 chart.dataZoom.zoomLock

Type: boolean
Default Value: false
DataZoom lock. Defaults to false. When set to true, the selected area cannot be zoomed, i.e. (end - start) value remains unchanged; only data roaming is applicable.

3.18 Chart-roamController-Optionen (nicht getestet, funktioniert nur bei Map)

3.18.1 chart.roamController.show

Type: boolean
Default Value: true
Specifies whether to show. Valid values are: true (show) | false (hide).

3.18.2 chart.roamController.zlevel

Type: number
Default Value: 0
The first grade cascading control. Every zlevel will be allocated to a independent canvas, charts and components with the same zlevel will be rendered in the same canvas. The higher zlevel the closer to the top. More canvas dom will need more memory and performance, so never set too many zlevel. Most of the time, the second grade cascading control 'z' is recommended.

3.18.3 chart.roamController.z

Type: number
Default Value: 4
The second grade cascading control, In the same canvas (zlevel), the higher z the closer to the top.

3.18.4 chart.roamController.x

Type: string, number
Default Value: 'left'
horizontal position. Defaults to left. Valid values are: 'center' | 'left' | 'right' | {number}(x-coordinate, in px).

3.18.5 chart.roamController.y

Type: string, number
Default Value: 'top'
vertical position. Defaults to top. Valid values are: 'top' | 'bottom' | 'center' | {number}(y-coordinate, in px).

3.18.6 chart.roamController.width

Type: number
Default Value: 80
Specifies width which also define the size of the roam disk, can be {number}(in px)

3.18.7 chart.roamController.height

Type: number
Default Value: 120
Specifies height of the whold controller boundary box, the zoom button would be placed at the bottom, can be {number}(in px)

3.18.8 chart.roamController.backgroundColor

Type: color
Default Value: 'rgba(0,0,0,0)'
The color of the background. Defaults to transparent.

3.18.9 chart.roamController.borderColor

Type: string
Default Value: '#ccc'
Color of the border.

3.18.10 chart.roamController.borderWidth

Type: number
Default Value: 0
Width of the border.

3.18.11 chart.roamController.padding

Type: number, Array
Default Value: 5
The inner padding, in px, defaults to 5. Can be set as array - [top, right, bottom, left], same as css, see image below.

3.18.12 chart.roamController.fillerColor

Type: color
Default Value: '#fff'
The fill color of the text in the controller.

Copyright © 2023 TRADUI Technologies GmbH

3.18.13 chart.roamController.handleColor

Type: color
Default Value: '#6495ed'
The main color of the control handle.

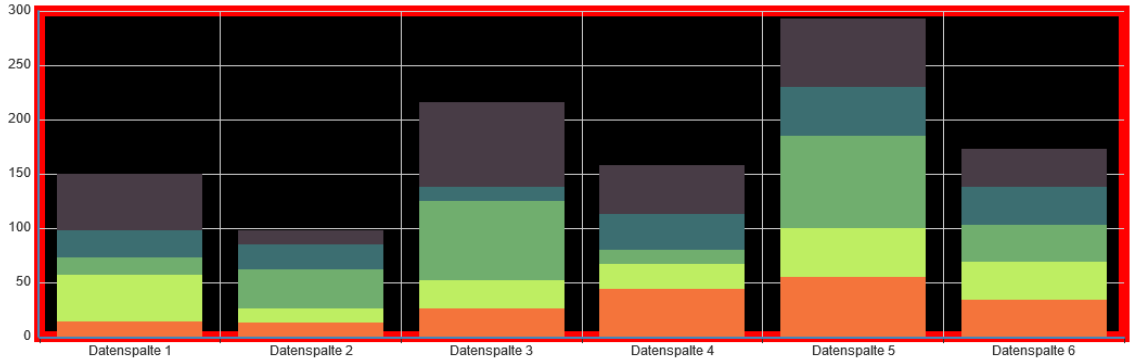
3.18.14 chart.roamController.step

Type: number
Default Value: 15
moving step of the 4 direction roam in px

3.18.15 chart.roamController.mapTypeControl

Type: Object
Default Value: null
Required. Specifies the mapType which should be under control, as: { china: true }
You can specify every single mapType when multiple map in a chart at the same time, such as: { china: false, world: true}

3.19 Chart-Grid-Optionen



3.19.1 chart.grid.zlevel

Type: number
Default Value: 0
The first grade cascading control. Every zlevel will be allocated to a independent canvas, charts and components with the same zlevel will be rendered in the same canvas. The higher zlevel the closer to the top. More canvas dom will need more memory and performance, so never set too many zlevel. Most of the time, the second grade cascading control 'z' is recommended.

3.19.2 chart.grid.z

Type: number
Default Value: 0
The second grade cascading control, In the same canvas (zlevel), the higher z the closer to the top.

3.19.3 chart.grid.x

Type: number, string
Default Value: 80
Set abscissa on the upper left corner of the grid in px, or in percent (string) such as '50%' (horizontal center of the display area).

3.19.4 chart.grid.y

Type: number, string
Default Value: 60
Set ordinate on the upper left corner of the grid in px, or in percent (string). Defaults to null. The position varies with y2, such as '50%' (vertical center of the display area).

3.19.5 chart.grid.x2

Type: number, string
Default Value: 80
Set abscissa on the lower right corner of the grid in px, or in percent (string) such as '50%' (horizontal center of the display area).

3.19.6 chart.grid.y2

Type: number, string
Default Value: 60
Set ordinate on the lower right corner of the grid in px, or in percent (string) such as '50%' (vertical center of the display area).

3.19.7 chart.grid.width

Type: number
Default Value: 自适应
Width of the grid (without axes), defaults to the total width - x - x2, in px, ignore x2 after specifying width. See the image below.
Can also be set in percent (string), such as '50%' (half the width of the display area).

3.19.8 chart.grid.height

Type: number
Default Value: 自适应
Height of the grid (without axes), defaults to the total height - y - y2, in px, ignore y2 after specifying height. See the image below.
Can also be set in percent (string), such as '50%' (half the height of the display area).

3.19.9 chart.grid.backgroundColor

Type: color
Default Value: 'rgba(0,0,0,0)'
Background color, defaults to transparent.

Copyright © 2023 TRADUI Technologies GmbH

3.19.10 chart.grid.borderWidth

Type: number
Default Value: 1
Width of the border.

3.19.11 chart.grid.borderColor

Type: color
Default Value: '#ccc'
Color of the border.

3.20 Series-Bar

3.20.1 chart.series.i[].stack (Auch in line/area)

Type: string
Default Value: null
name of the stack, USELESS in irregular line|bar. It is used in stacked charts with multiple groups of data. For example, stack:'group1' means to stack the values of the data whose stack value is 'group1' in the series array.

3.20.2 chart.series.i[].barGap

Type: number, string
Default Value: '30%'
the distance between each bar. Defaults to barWidth * 30%. Can also set to be a fixed value.

3.20.3 chart.series.i[].barCategoryGap

Type: number, string
Default Value: '20%'
the distance between each bar cetegory. Defaults to barCategoryWidth * 20%. Can also set to be a fixed value.

3.20.4 chart.series.i[].barMinHeight

Type: number
Default Value: 0
The minimum height for the bar. Used to prevent tiny item values from affecting interaction.

3.20.5 chart.series.i[].barWidth

Type: number
Default Value: @TODO: 自适应
the width of the bar/ candlestick. Adaptive when not specified.

3.20.6 chart.series.i[].barMaxWidth

Type: number
Default Value: @TODO: 自适应
the maximum width of the bar (candlestick). Adaptive when not specified.

3.21 Series-Gauge

3.21.1 chart.series.i[].axisLabel.clickable

Type: boolean
Default Value: true
Specifies data graphic clickable or not, default to true, recommend to false when you do not have a click event handler.

3.21.2 chart.series.i[].axisLabel.formatter

Type: string, Function
Default Value: null
split name formatter: **{string}** (Template) | **{Function}**

3.21.3 chart.series.i[].axisLabel.margin

Type: number
Default Value: 8
The pixel distance between the axisLabel and the axis.Defaults to 8.

3.21.4 chart.series.i[].axisLabel.rotate

Type: number
Default Value: 0
The angle of rotation, defaults to 0 (not rotate). Positive value for counterclockwise rotation, negative value for clockwise rotation. Valid values are: -90 ~ 90.

3.21.5 chart.series.i[].axisLabel.show

Type: boolean
Dafault Value: true
Specifies whether to show axisLabel. Defaults to true, which is a prerequisite for the following properties.

3.21.6 chart.series.i[].axisLabel.textStyle.align

Type: string
Default Value: @TODO:各异
horizontal alignment. Valid values are: 'left' | 'right' | 'center'.

3.21.7 chart.series.i[].axisLabel.textStyle.baseline

Type: string
Default Value: @TODO: 各异
vertical alignment. Valid values are: 'top' | 'bottom' | 'middle'.

3.21.8 chart.series.i[].axisLabel.textStyle.color

Type: color
Default Value: @TODO: 各异
color

Copyright © 2023 TRADUI Technologies GmbH

3.21.9 chart.series.i[].axisLabel.textStyle.fontFamily

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font family.

3.21.10 chart.series.i[].axisLabel.textStyle.fontSize

Type: number
Default Value: 12
font size, in px.

3.21.11 chart.series.i[].axisLabel.textStyle.fontStyle

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

3.21.12 chart.series.i[].axisLabel.textStyle.fontWeight

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

3.21.13 chart.series.i[].splitNumber

Type: number
Default Value: 10
The number of segments. Defaults to 10.

3.21.14 chart.series.i[].axisLine.show

Type: boolean
Default Value: true
The property "show" specifies whether to show axisLine or not.

3.21.15 chart.series.i[].axisLine.lineStyle.color

Type: Array
Default Value: [[0.2, "#228b22"], [0.8, "#48b"], [1, "#ff4500"]]
it is a two-dimensional array that can be used to divide the gauge axis into several parts, and designate a specific color to each part on this form: [[percent, color value], [...]].

3.21.16 chart.series.i[].axisLine.lineStyle.type

Type: string
Default Value: 'solid'
line style. Valid values are: 'solid' | 'dotted' | 'dashed', and can be 'curve' | 'broken' in tree structures.

Copyright © 2023 TRADUI Technologies GmbH

3.21.17 chart.series.i[].axisLine.lineStyle.width

Type: number
Default Value: 30
width of the line.

3.21.18 chart.series.i[].axisLine.lineStyle.shadowColor

Type: color
Default Value: 'rgba(0,0,0,0)'
applicable to the main line (in IE8+). Color of the shadow. Supports rgba.

3.21.19 chart.series.i[].axisLine.lineStyle.shadowBlur

Type: number
Default Value: 5
applicable to the main line (in IE8+). Blur degree of the shadow. Valid values are positive.

3.21.20 chart.series.i[].axisLine.lineStyle.shadowOffsetX

Type: number
Default Value: 3
applicable to the main line (in IE8+). Horizontal offset of the shadow. Right when positive, left when negative.

3.21.21 chart.series.i[].axisLine.lineStyle.shadowOffsetY

Type: number
Default Value: 3
applicable to the main line (in IE8+). Vertical offset of the shadow. Down when positive, up when negative.

3.21.22 chart.series.i[].axisTick.show

Type: boolean
Defaul Value: true
The property "show" specifies whether to show axisLine or not.

3.21.23 chart.series.i[].axisTick.lineStyle.color

Type: color
Default Value: '#eee'
color

3.21.24 chart.series.i[].axisTick.lineStyle.type

Type: string
Default Value: 'solid'
line style. Valid values are: 'solid' | 'dotted' | 'dashed', and can be 'curve' | 'broken' in tree structures.

Copyright © 2023 TRADUI Technologies GmbH

3.21.25 chart.series.i[].axisTick.lineStyle.width

Type: number
Default Value: 1
width of the line.

3.21.26 chart.series.i[].axisTick.lineStyle.shadowColor

Type: color
Default Value: 'rgba(0,0,0,0)'
applicable to the main line (in IE8+). Color of the shadow. Supports rgba.

3.21.27 chart.series.i[].axisTick.lineStyle.shadowBlur

Type: number
Default Value: 5
applicable to the main line (in IE8+). Blur degree of the shadow. Valid values are positive.

3.21.28 chart.series.i[].axisTick.lineStyle.shadowOffsetX

Type: number
Default Value: 3
applicable to the main line (in IE8+). Horizontal offset of the shadow. Right when positive, left when negative.

3.21.29 chart.series.i[].axisTick.lineStyle.shadowOffsetY

Type: number
Default Value: 3
applicable to the main line (in IE8+). Vertical offset of the shadow. Down when positive, up when negative.

3.21.30 chart.series.i[].axisTick.splitNumber

Type: number
Default Value: 5
The property "splitNumber" controls the number of segments;

3.21.31 chart.series.i[].axisTick.length

Type: number
Default Value: 8
The property "length" controls line length for axisTick.

3.21.32 chart.series.i[].detail.show

Type: boolean
Default Value: true
The property "show" specifies whether to show detail or not.

Copyright © 2023 TRADUI Technologies GmbH

3.21.33 chart.series.i[].detail.backgroundColor

Type: color
Default Value: 'rgba(0,0,0,0)'
background color

3.21.34 chart.series.i[].detail.borderWidth

Type: number
Default Value: 0
border width

3.21.35 chart.series.i[].detail.borderColor

Type: color
Default Value: '#ccc'
border color

3.21.36 chart.series.i[].detail.width

Type: number
Default Value: 100
width of detail

3.21.37 chart.series.i[].detail.height

Type: number
Default Value: 40
height of detail

3.21.38 chart.series.i[].detail.offsetCenter

Type: Array
Default Value: [0, "-40%"]
The property "offsetCenter" is used to locate title. Offset to the center coordinates if the array is x-axis. If in percent, it is relative to the outer radius of the gauge.

3.21.39 chart.series.i[].detail.formatter

Type: Function
Default Value: null
formatter of detail.

3.21.40 chart.series.i[].detail.textStyle

Type: Object
Default Value: {...}
textStyle

Copyright © 2023 TRADUI Technologies GmbH

3.21.41 chart.series.i[].detail.textStyle.color

Type: color
Default Value: 'auto'
color

3.21.42 chart.series.i[].detail.textStyle.align

Type: string
Default Value: @TODO: 各异
horizontal alignment. Valid values are: 'left' | 'right' | 'center'.

3.21.43 chart.series.i[].detail.textStyle.baseline

Type: string
Default Value: @TODO: 各异
vertical alignment. Valid values are: 'top' | 'bottom' | 'middle'.

3.21.44 chart.series.i[].detail.textStyle.fontFamily

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font family.

3.21.45 chart.series.i[].detail.textStyle.fontSize

Type:number
Default Value: 30
font size, in px.

3.21.46 chart.series.i[].detail.textStyle.fontStyle

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

3.21.47 chart.series.i[].detail.textStyle.fontWeight

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

3.21.48 chart.series.i[].startAngle

Typ: number
Default Value: 225
start angle, pie (90), gauge (225). Valid range is: [-360,360]

Copyright © 2023 TRADUI Technologies GmbH

3.21.49 chart.series.i[].endAngle

Type: number
Default Value: -45
end angle. Valid range is: [-360,360], ensure that startAngle - endAngle is positive.

3.21.50 chart.series.i[].pointer.length

Type: string, number
Default Value: 80%
The property "length" controls line length for pointer. If in percent, it is relative to the outer radius of the gauge.

3.21.51 chart.series.i[].pointer.width

Type: number
Default Value: 8
The property "width" controls the widest point of the pointer.

3.21.52 chart.series.i[].pointer.color

Type: color
Default Value: 'auto'
The property "color" controls the color of the pointer.

3.21.53 chart.series.i[].splitLine.show

Type: boolean
Default Value: true
The property "show" specifies whether to show splitLine or not.

3.21.54 chart.series.i[].splitLine.length

Type: number
Default Value: 30
The property "length" controls line length for splitLine.

3.21.55 chart.series.i[].splitLine.lineStyle.color

Type: color
Default Value: '#eee'
color

3.21.56 chart.series.i[].splitLine.lineStyle.shadowColor

Type: color
Default Value: 'rgba(0,0,0,0)'
applicable to the main line (in IE8+). Color of the shadow. Supports rgba.

Copyright © 2023 TRADUI Technologies GmbH

3.21.57 chart.series.i[].splitLine.lineStyle.type

Type: string
Default Value: 'solid'
line style. Valid values are: 'solid' | 'dotted' | 'dashed', and can be 'curve' | 'broken' in tree structures.

3.21.58 chart.series.i[].splitLine.lineStyle.width

Type: number
Default Value: 2
width of the line.

3.21.59 chart.series.i[].splitLine.lineStyle.shadowBlur

Type: number
Default Value: 5
applicable to the main line (in IE8+). Blur degree of the shadow. Valid values are positive.

3.21.60 chart.series.i[].splitLine.lineStyle.shadowOffsetX

Type: number
Default Value: 3
applicable to the main line (in IE8+). Horizontal offset of the shadow. Right when positive, left when negative.

3.21.61 chart.series.i[].splitLine.lineStyle.shadowOffsetY

Type: number
Default Value: 3
applicable to the main line (in IE8+). Vertical offset of the shadow. Down when positive, up when negative.

3.21.62 chart.series.i[].title.show

Type: boolean
Default Value: true
The property "show" specifies whether to show title or not.

3.21.63 chart.series.i[].title.offsetCenter

Type: Array
Dafault Value: [0, "-40%"]
The property "offsetCenter" is used to locate title. Offset to the center coordinates if the array is x-axis. If in percent, it is relative to the outer radius of the gauge.

3.21.64 chart.series.i[].title.textStyle.color

Type: color
Default Value: '#333'
color

Copyright © 2023 TRADUI Technologies GmbH

3.21.65 chart.series.i[].title.textStyle.align

Type: string
Default Value: @TODO: 各异
horizontal alignment. Valid values are: 'left' | 'right' | 'center'.

3.21.66 chart.series.i[].title.textStyle.baseline

Type: string
Default Value: @TODO: 各异
vertical alignment. Valid values are: 'top' | 'bottom' | 'middle'.

3.21.67 chart.series.i[].title.textStyle.fontFamily

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font family.

3.21.68 chart.series.i[].title.textStyle.fontSize

Type: number
Default Value: 15
font size, in px.

3.21.69 chart.series.i[].title.textStyle.fontStyle

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

3.21.70 chart.series.i[].title.textStyle.fontWeight

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

3.22 Series-Chord

3.22.1 chart.series.i[].data[] (Noch nicht implementiert)

Type: Object
The same as series[i].nodes

3.22.2 chart.series.i[].data[].children (Noch nicht implementiert)

(Nicht in original Doku vorhanden, nur bei treemap)

3.22.3 chart.series.i[].data[].evolution (Noch nicht implementiert)

(Nicht in Original Doku vorhanden, nur bei EventRiver)

Copyright © 2023 TRADUI Technologies GmbH

3.22.4 chart.series.i[].data[].evolution.detail (Noch nicht implementiert)

(Nicht in Original Doku vorhanden, nur bei EventRiver)

3.22.5 chart.series.i[].data[].evolution.detail.img (Noch nicht implementiert)

(Nicht in Original Doku vorhanden, nur bei EventRiver)

3.22.6 chart.series.i[].data[].evolution.detail.link (Noch nicht implementiert)

(Nicht in Original Doku vorhanden, nur bei EventRiver)

3.22.7 chart.series.i[].data[].evolution.detail.text (Noch nicht implementiert)

(Nicht in Original Doku vorhanden, nur bei EventRiver)

3.22.8 chart.series.i[].data[].evolution.time (Noch nicht implementiert)

(Nicht in Original Doku vorhanden, nur bei EventRiver)

3.22.9 chart.series.i[].data[].evolution.value (Noch nicht implementiert)

(Nicht in Original Doku vorhanden, nur bei EventRiver)

3.22.10 chart.series.i[].data[].itemStyle (Noch nicht implementiert)

Type: Object
item style see itemStyle. Notice that itemStyle in a single node of force has no nodeStyle under the normal(emphasis) option. Instead itemStyle.normal(emphasis).color|borderWidth|borderColor will be used

3.22.11 chart.series.i[].data[].name (Noch nicht implementiert)

Type: string
Default Value: ' '
Name of node

3.22.12 chart.series.i[].data[].label (Noch nicht implementiert)

Type: string
Default Value: ' '
Label name of the node, name(above) as default.

3.22.13 chart.series.i[].data[].value (Noch nicht implementiert)

Type: number
Value of name

3.22.14 chart.series.i[].data[].symbol (Noch nicht implementiert)

Type: string
Default Value: 'circle'
See series (Cartesian)

Copyright © 2023 TRADUI Technologies GmbH

3.22.15 chart.series.i[].data[].symbolSize (Noch nicht implementiert)

Type: number, Array
Size of symbol

3.22.16 chart.series.i[].data[].weight (Noch nicht implementiert)

(Nicht in Original Doku vorhanden)

3.22.17 chart.series.i[].data[].itemStyle.normal (Noch nicht implementiert)

Type: Object
normal item style

3.22.18 chart.series.i[].data[].itemStyle.normal.color (Noch nicht implementiert)

Type: color, Function
Default Value: @TODO: 图表各异
color, dominant color, function paramater is {seriesIndex:x, series:xxx, dataIndex:y, data:yyy}

3.22.19 chart.series.i[].data[].itemStyle.normal.chordStyle (Noch nicht implementiert)

Type: Object
Default Value: @TODO: 图表各异
chord style.

3.22.20 chart.series.i[].data[].itemStyle.normal.chordStyle.width (Noch nicht implementiert)

Type: number
Default Value: 1
Width of bezier curve, Only available when ribbonType is false

3.22.21 chart.series.i[].data[].itemStyle.normal.chordStyle.color (Noch nicht implementiert)

Type: string
Default Value: 'black'
Color of bezier curve, Only available when ribbonType is false

3.22.22 chart.series.i[].data[].itemStyle.normal.chordStyle.borderWidth (Noch nicht implementiert)

Type: number
Default Value: 1
Border width of ribbon shape, Only available when ribbonType is true

Copyright © 2023 TRADUI Technologies GmbH

3.22.23 chart.series.i[].data[].itemStyle.normal.chordStyle.borderColor (Noch nicht implementiert)

Type: string
Default Value: '#999'
Border color of ribbon shape, Only available when ribbonType is true

3.22.24 chart.series.i[].data[].itemStyle.normal.label (Noch nicht implementiert)

Type: Object
Default Value: { "show": true, "position": "outer" }
In pie, displayed on the outside by default. labelLine.length controls the label's distance from the pie. In funnel, displayed on the right by default. labelLine.length controls the label's distance from the funnel. In map, the location of labels cannot be specified. position can be specified in line, column, candlestick, scatter. See below.

3.22.25 chart.series.i[].data[].itemStyle.normal.label.show (Noch nicht implementiert)

Type: boolean
Default Value: true
Specifies whether to show label. Valid values are: true (show) | false (hide).

3.22.26 chart.series.i[].data[].itemStyle.normal.label.rotate (Noch nicht implementiert)

Type: boolean
Default Value: false
applicable to chord. Label rotates automatically.

3.22.27 chart.series.i[].data[].itemStyle.normal.label.distance (Noch nicht implementiert)

Type: number
Default Value: 10
In chord, Label's distance from chord after it rotates automatically. In pie, this distance means the factor of label distance and pie radius(In ring chart, is sum of inner and outer radius), default to **0.5**.

3.22.28 chart.series.i[].data[].itemStyle.normal.label.formatter (Noch nicht implementiert)

Type: string, Function
Default Value: null
content formatter: **{string}** (Template) | **{Function}**. Supports asynchronous callback. See below.

- **{string}**, Template, its variables are:
  - {a} | {a0}
  - {b} | {b0}
  - {c} | {c0}
  - {d} | {d0} (some chart types do not have this variable)
  - {e} | {e0} (some chart types do not have this variable)
  - If there are multiple sets of data, there will be multiple sets of varaibles accordingly, like {a1}, {b1}, {c1}, {d1}, {a2}, {b2}, {c2}, {d2}, ...

- Here are what variable a, b, c and d reprensent in different chart types:
  - **line & area**, **column & bar, candlestick**: a (series name), b(category value), c (value), d(null)
  - **scatter & bubble**: a (series name), b (data name), c (value array), d(null)
  - **map**: a (series name), b (area name), c(merged value), d(null)
  - **pie**, **radar**, **gauge**, **funnel**: a (series name), b (data item name), c(value), d (pie: percent | radar: indicator name)
  - **force, chord** :
    - node : a (series name), b (node name), c (node value), d (node index), e
    - link : a (series name), b (link name, default bigEndNodeName-smallEndNodeName), c(link.value), d(name or index of big end node), e(name or index of small end node)

- **{Function}**, the list of parameters passed in is [params, ticket, callback], detail as follows:
  - params : array content and template varaibles are the same

```
[
    {

        seriesIndex: 0,
        seriesName: 'Sales',
        name: 'Monday',
        dataIndex: 0,
        data: data,
        name: name,
        value: value,
        percent: special,       // pie
        indicator: special,     // radar, force, chord
        value2: special2,       // force, chord
        indicator2: special2    // force, chord
    },
    {..},
    ...
]
```

  - ticket: asynchronous callback marker
  - callback: asynchronous callback, the callback needs two parameters, the first one is the ticket that we mentioned earlier, the second is the fill content html.
  - ** When the function callbacks, 'this' pointer points to the current chart instance (myChart).*

label text formatter, universal, same as Tooltip.formatter, Supports template, method callback. Does not support asynchronous callback.

3.22.29 chart.series.i[].data[].itemStyle.normal.label.textStyle (Noch nicht implementiert)

Type: Object
Default Value: null
style for label text.

3.22.30 chart.series.i[].data[].itemStyle.normal.label.textStyle.color (Noch nicht implementiert)

Type: color
Default Value: @TODO: 各异
color

3.22.31 chart.series.i[].data[].itemStyle.normal.label.textStyle.align (Noch nicht implementiert)

Type: string
Default Value: @TODO: 各异
horizontal alignment. Valid values are: 'left' | 'right' | 'center'.

3.22.32 chart.series.i[].data[].itemStyle.normal.label.textStyle.baseline (Noch nicht implementiert)

Type: string
Default Value: @TODO: 各异
vertical alignment. Valid values are: 'top' | 'bottom' | 'middle'.

3.22.33 chart.series.i[].data[].itemStyle.normal.label.textStyle.fontFamily (Noch nicht implementiert)

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font family.

3.22.34 chart.series.i[].data[].itemStyle.normal.label.textStyle.fontSize (Noch nicht implementiert)

Type: number
Default Value: 12
font size, in px.

3.22.35 chart.series.i[].data[].itemStyle.normal.label.textStyle.fontStyle (Noch nicht implementiert)

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

3.22.36 chart.series.i[].data[].itemStyle.normal.label.textStyle.fontWeight (Noch nicht implementiert)

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

3.22.37 chart.series.i[].data[].itemStyle.emphasis (Noch nicht implementiert)

Type: Object
emphasis item style (when hovered)

Copyright © 2023 TRADUI Technologies GmbH

3.22.38 chart.series.i[].data[].itemStyle.emphasis.color (Noch nicht implementiert)

Type: color, Function
Default Value: @TODO: 图表各异
color, dominant color, function paramater is {seriesIndex:x, series:xxx, dataIndex:y, data:yyy}

3.22.39 chart.series.i[].data[].itemStyle.emphasis.chordStyle (Noch nicht implementiert)

Type: Object
Default Value: @TODO: 图表各异
chord style.

3.22.40 chart.series.i[].data[].itemStyle.emphasis.chordStyle.width (Noch nicht implementiert)

Type: number
Default Value: 1
Width of bezier curve, Only available when ribbonType is false

3.22.41 chart.series.i[].data[].itemStyle.emphasis.chordStyle.color (Noch nicht implementiert)

Type: string
Default Value: 'black'
Color of bezier curve, Only available when ribbonType is false

3.22.42 chart.series.i[].data[].itemStyle.emphasis.chordStyle.borderWidth (Noch nicht implementiert)

Type: number
Default Value: 1
Border width of ribbon shape, Only available when ribbonType is true

3.22.43 chart.series.i[].data[].itemStyle.emphasis.chordStyle.borderColor (Noch nicht implementiert)

Type: string
Default Value: '#999'
Border color of ribbon shape, Only available when ribbonType is true

3.22.44 chart.series.i[].data[].itemStyle.emphasis.label (Noch nicht implementiert)

Type: Object
Default Value: { "show": true, "position": "outer" }
In pie, displayed on the outside by default. labelLine.length controls the label's distance from the pie. In funnel, displayed on the right by default. labelLine.length controls the label's distance from the funnel. In map, the location of labels cannot be specified. position can be specified in line, column, candlestick, scatter. See below.

Copyright © 2023 TRADUI Technologies GmbH

3.22.45 chart.series.i[].data[].itemStyle.emphasis.label.show (Noch nicht implementiert)

Type: boolean
Default Value: true
Specifies whether to show label. Valid values are: true (show) | false (hide).

3.22.46 chart.series.i[].data[].itemStyle.emphasis.label.rotate (Noch nicht implementiert)

Type: boolean
Default Value: false
applicable to chord. Label rotates automatically.

3.22.47 chart.series.i[].data[].itemStyle.emphasis.label.distance (Noch nicht implementiert)

Type: number
Default Value: 10
In chord, Label's distance from chord after it rotates automatically. In pie, this distance means the factor of label distance and pie radius(In ring chart, is sum of inner and outer radius), default to **0.5**.

3.22.48 chart.series.i[].data[].itemStyle.emphasis.label.formatter (Noch nicht implementiert)

Type: string, Function
Default Value: null
content formatter: **{string}** (Template) | **{Function}**. Supports asynchronous callback. See below.

- **{string}**, Template, its variables are:
  - {a} | {a0}
  - {b} | {b0}
  - {c} | {c0}
  - {d} | {d0} (some chart types do not have this variable)
  - {e} | {e0} (some chart types do not have this variable)
  - If there are multiple sets of data, there will be multiple sets of varaibles accordingly, like {a1}, {b1}, {c1}, {d1}, {a2}, {b2}, {c2}, {d2}, ...
  - Here are what variable a, b, c and d reprensent in different chart types:
    - **line & area**, **column & bar, candlestick**: a (series name), b(category value), c (value), d(null)
    - **scatter & bubble**: a (series name), b (data name), c (value array), d(null)
    - **map**: a (series name), b (area name), c(merged value), d(null)
    - **pie**, **radar**, **gauge**, **funnel**: a (series name), b (data item name), c(value), d (pie: percent | radar: indicator name)
    - **force, chord** :
      - node : a (series name), b (node name), c (node value), d (node index), e
      - link : a (series name), b (link name, default bigEndNodeName-smallEndNodeName), c(link.value), d(name or index of big end node), e(name or index of small end node)
- **{Function}**, the list of parameters passed in is [params, ticket, callback], detail as follows:
  - params : array content and template varaibles are the same

```
[
    {
```

```
            seriesIndex: 0,
            seriesName: 'Sales',
            name: 'Monday',
            dataIndex: 0,
            data: data,
            name: name,
            value: value,
            percent: special,       // pie
            indicator: special,     // radar, force, chord
            value2: special2,       // force, chord
            indicator2: special2    // force, chord
        },
        {..},
        ...
    ]
```

- ticket: asynchronous callback marker
- callback: asynchronous callback, the callback needs two parameters, the first one is the ticket that we mentioned earlier, the second is the fill content html.
- ** When the function callbacks, 'this' pointer points to the current chart instance (myChart).*

label text formatter, universal, same as Tooltip.formatter, Supports template, method callback. Does not support asynchronous callback.

3.22.49 chart.series.i[].data[].itemStyle.emphasis.label.textStyle (Noch nicht implementiert)

Type: Object
Default Value: null
style for label text.

3.22.50 chart.series.i[].data[].itemStyle.emphasis.label.textStyle.color (Noch nicht implementiert)

Type: color
Default Value: @TODO: 各异
color

3.22.51 chart.series.i[].data[].itemStyle.emphasis.label.textStyle.align (Noch nicht implementiert)

Type: string
Default Value: @TODO: 各异
horizontal alignment. Valid values are: 'left' | 'right' | 'center'.

3.22.52 chart.series.i[].data[].itemStyle.emphasis.label.textStyle.baseline (Noch nicht implementiert)

Type: string
Default Value: @TODO: 各异
vertical alignment. Valid values are: 'top' | 'bottom' | 'middle'.

3.22.53 chart.series.i[].data[].itemStyle.emphasis.label.textStyle.fontFamily (Noch nicht implementiert)

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font family.

3.22.54 chart.series.i[].data[].itemStyle.emphasis.label.textStyle.fontSize (Noch nicht implementiert)

Type: number
Default Value: 12
font size, in px.

3.22.55 chart.series.i[].data[].itemStyle.emphasis.label.textStyle.fontStyle (Noch nicht implementiert)

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

3.22.56 chart.series.i[].data[].itemStyle.emphasis.label.textStyle.fontWeight (Noch nicht implementiert)

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

3.22.57 chart.series.i[].symbol (Auch bei line/area, scatter, radar, force, tree)

Type: string
Default Value: null
A predefined shape or symbol for the marker. The default symbol is automatically selected from (When all symbols are used, new symbols are pulled from the start again. Set it to'none' if you don't want to show the symbol): 'circle' | 'rectangle' | 'triangle' | 'diamond' | 'emptyCircle' | 'emptyRectangle' | 'emptyTriangle' | 'emptyDiamond' Other possible values are 'heart', 'droplet', 'pin', 'arrow' and 'star'. Although They are not included in the regular eight symbols above, you can use them at both series and data level. In addition, you can use 'star' + n (n>=3) to get N-pointed star, like 'star6' for six-pointed star. Since 1.3.5, ECharts added support for setting custom image as symbol. The URL to an image can be given on this form:'image://' + 'URL', such as 'image://../asset/ico/favicon.png'. See example 》

3.22.58 chart.series.i[].symbolSize (Auch bei line/area, scatter, radar, force, tree)

Type: number
Size of symbol, Only available if ribbonType is false

3.22.59 chart.series.i[].clockWise (Auch bei pie)

Type: boolean
Default Value: false
Specifies whether it is displayed in clockwise direction.

3.22.60 chart.series.i[].categories.name (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: string
Default Value: ' '
Name of category

3.22.61 chart.series.i[].categories.symbol (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: string
Default Value: 'circle'
A predefined shape or symbol for the marker. The default symbol is automatically selected from (When all symbols are used, new symbols are pulled from the start again. Set it to'none' if you don't want to show the symbol): 'circle' | 'rectangle' | 'triangle' | 'diamond' | 'emptyCircle' | 'emptyRectangle' | 'emptyTriangle' | 'emptyDiamond' Other possible values are 'heart', 'droplet', 'pin', 'arrow' and 'star'. Although They are not included in the regular eight symbols above, you can use them at both series and data level. In addition, you can use 'star' + n (n>=3) to get N-pointed star, like 'star6' for six-pointed star. Since 1.3.5, ECharts added support for setting custom image as symbol. The URL to an image can be given on this form:'image://' + 'URL', such as 'image://../asset/ico/favicon.png'. See example 》
See series (Cartesian)

3.22.62 chart.series.i[].categories.symbolSize (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: number, Array
Size of symbol

3.22.63 chart.series.i[].categories.itemStyle (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: Object
item style. see itemStyle. Notice that itemStyle in thecategory of force has no nodeStyle under the normal(emphasis) option. Instead itemStyle.normal(emphasis).color|borderWidth|borderColor will be used

3.22.64 chart.series.i[].categories.itemStyle.normal (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: Object
normal item style

3.22.65 chart.series.i[].categories.itemStyle.normal.color (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: color, Function
Default Value: @TODO: 图表各异
color, dominant color, function paramater is {seriesIndex:x, series:xxx, dataIndex:y, data:yyy}

3.22.66 chart.series.i[].categories.itemStyle.normal.chordStyle (Noch nicht implementiert)

Type: Object
Default Value: @TODO: 图表各异
chord style.

3.22.67 chart.series.i[].categories.itemStyle.normal.chordStyle.width (Noch nicht implementiert)

Type: number
Default Value: 1
Width of bezier curve, Only available when ribbonType is false

3.22.68 chart.series.i[].categories.itemStyle.normal.chordStyle.color (Noch nicht implementiert)

Type: string
Default Value: 'black'
Color of bezier curve, Only available when ribbonType is false

3.22.69 chart.series.i[].categories.itemStyle.normal.chordStyle.borderWidth (Noch nicht implementiert)

Type: number
Default Value: 1
Border width of ribbon shape, Only available when ribbonType is true

3.22.70 chart.series.i[].categories.itemStyle.normal.chordStyle.borderColor (Noch nicht implementiert)

Type: string
Default Value: '#999'
Border color of ribbon shape, Only available when ribbonType is true

3.22.71 chart.series.i[].categories.itemStyle.normal.label (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: Object
Default Value: { "show": true, "position": "outer" }
In pie, displayed on the outside by default. labelLine.length controls the label's distance from the pie. In funnel, displayed on the right by default. labelLine.length controls the label's distance from the funnel. In map, the location of labels cannot be specified. position can be specified in line, column, candlestick, scatter. See below.

3.22.72 chart.series.i[].categories.itemStyle.normal.label.show (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: boolean
Default Value: true
Specifies whether to show label. Valid values are: true (show) | false (hide).

Copyright © 2023 TRADUI Technologies GmbH

3.22.73 chart.series.i[].categories.itemStyle.normal.label.rotate (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: boolean
Default Value: false
applicable to chord. Label rotates automatically.

3.22.74 chart.series.i[].categories.itemStyle.normal.label.distance (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: number
Default Value: 10
In chord, Label's distance from chord after it rotates automatically. In pie, this distance means the factor of label distance and pie radius(In ring chart, is sum of inner and outer radius), default to **0.5**.

3.22.75 chart.series.i[].categories.itemStyle.normal.label.formatter (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: string, Function
Default Value: null
content formatter: **{string}** (Template) | **{Function}**. Supports asynchronous callback. See below.

- **{string}**, Template, its variables are:
  - {a} | {a0}
  - {b} | {b0}
  - {c} | {c0}
  - {d} | {d0} (some chart types do not have this variable)
  - {e} | {e0} (some chart types do not have this variable)
  - If there are multiple sets of data, there will be multiple sets of varaibles accordingly, like {a1}, {b1}, {c1}, {d1}, {a2}, {b2}, {c2}, {d2}, ...
  - Here are what variable a, b, c and d reprensent in different chart types:
    - **line & area**, **column & bar, candlestick**: a (series name), b(category value), c (value), d(null)
    - **scatter & bubble**: a (series name), b (data name), c (value array), d(null)
    - **map**: a (series name), b (area name), c(merged value), d(null)
    - **pie**, **radar**, **gauge**, **funnel**: a (series name), b (data item name), c(value), d (pie: percent | radar: indicator name)
    - **force, chord** :
      - node : a (series name), b (node name), c (node value), d (node index), e
      - link : a (series name), b (link name, default bigEndNodeName-smallEndNodeName), c(link.value), d(name or index of big end node), e(name or index of small end node)
- **{Function}**, the list of parameters passed in is [params, ticket, callback], detail as follows:
  - params : array content and template varaibles are the same

```
[
    {

        seriesIndex: 0,
        seriesName: 'Sales',
        name: 'Monday',
        dataIndex: 0,
        data: data,
        name: name,
        value: value,
        percent: special,       // pie
```

```
            indicator: special,     // radar, force, chord
            value2: special2,       // force, chord
            indicator2: special2    // force, chord
        },
        {..},
        ...
    ]
```

- ticket: asynchronous callback marker
- callback: asynchronous callback, the callback needs two parameters, the first one is the ticket that we mentioned earlier, the second is the fill content html.
- ** When the function callbacks, 'this' pointer points to the current chart instance (myChart).*

label text formatter, universal, same as Tooltip.formatter, Supports template, method callback. Does not support asynchronous callback.

3.22.76 chart.series.i[].categories.itemStyle.normal.label.textStyle (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: Object
Default Value: null
style for label text.

3.22.77 chart.series.i[].categories.itemStyle.normal.label.textStyle.color (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: color
Default Value: @TODO: 各异
color

3.22.78 chart.series.i[].categories.itemStyle.normal.label.textStyle.align (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: string
Default Value: @TODO: 各异
horizontal alignment. Valid values are: 'left' | 'right' | 'center'.

3.22.79 chart.series.i[].categories.itemStyle.normal.label.textStyle.baseline (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: string
Default Value: @TODO: 各异
vertical alignment. Valid values are: 'top' | 'bottom' | 'middle'.

3.22.80 chart.series.i[].categories.itemStyle.normal.label.textStyle.fontFamily (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font family.

3.22.81 chart.series.i[].categories.itemStyle.normal.label.textStyle.fontSize (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: number
Default Value: 12
font size, in px.

3.22.82 chart.series.i[].categories.itemStyle.normal.label.textStyle.fontStyle (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

3.22.83 chart.series.i[].categories.itemStyle.normal.label.textStyle.fontWeight (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

3.22.84 chart.series.i[].categories.itemStyle.emphasis (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: Object
emphasis item style (when hovered)

3.22.85 chart.series.i[].categories.itemStyle.emphasis.color (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: color, Function
Default Value: @TODO: 图表各异
color, dominant color, function paramater is {seriesIndex:x, series:xxx, dataIndex:y, data:yyy}

3.22.86 chart.series.i[].categories.itemStyle.emphasis.chordStyle (Noch nicht implementiert)

Type: Object
Default Value: @TODO: 图表各异
chord style.

3.22.87 chart.series.i[].categories.itemStyle.emphasis.chordStyle.width (Noch nicht implementiert)

Type: number
Default Value: 1
Width of bezier curve, Only available when ribbonType is false

3.22.88 chart.series.i[].categories.itemStyle.emphasis.chordStyle.color (Noch nicht implementiert)

Type: string
Default Value: 'black'
Color of bezier curve, Only available when ribbonType is false

3.22.89 chart.series.i[].categories.itemStyle.emphasis.chordStyle.borderWidth (Noch nicht implementiert)

Type: number
Default Value: 1
Border width of ribbon shape, Only available when ribbonType is true

3.22.90 chart.series.i[].categories.itemStyle.emphasis.chordStyle.borderColor (Noch nicht implementiert)

Type: string
Default Value: '#999'
Border color of ribbon shape, Only available when ribbonType is true

3.22.91 chart.series.i[].categories.itemStyle.emphasis.label (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: Object
Default Value: { "show": true, "position": "outer" }
In pie, displayed on the outside by default. labelLine.length controls the label's distance from the pie. In funnel, displayed on the right by default. labelLine.length controls the label's distance from the funnel. In map, the location of labels cannot be specified. position can be specified in line, column, candlestick, scatter. See below.

3.22.92 chart.series.i[].categories.itemStyle.emphasis.label.show (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: boolean
Default Value: true
Specifies whether to show label. Valid values are: true (show) | false (hide).

3.22.93 chart.series.i[].categories.itemStyle.emphasis.label.rotate (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: boolean
Default Value: false
applicable to chord. Label rotates automatically.

3.22.94 chart.series.i[].categories.itemStyle.emphasis.label.distance (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: number
Default Value: 10

Copyright © 2023 TRADUI Technologies GmbH

In chord, Label's distance from chord after it rotates automatically. In pie, this distance means the factor of label distance and pie radius(In ring chart, is sum of inner and outer radius), default to **0.5**.

3.22.95 chart.series.i[].categories.itemStyle.emphasis.label.formatter (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: string, Function
Default Value: null
content formatter: **{string}** (Template) | **{Function}**. Supports asynchronous callback. See below.

- **{string}**, Template, its variables are:
  - {a} | {a0}
  - {b} | {b0}
  - {c} | {c0}
  - {d} | {d0} (some chart types do not have this variable)
  - {e} | {e0} (some chart types do not have this variable)
  - If there are multiple sets of data, there will be multiple sets of varaibles accordingly, like {a1}, {b1}, {c1}, {d1}, {a2}, {b2}, {c2}, {d2}, ...
  - Here are what variable a, b, c and d reprensent in different chart types:
    - **line & area**, **column & bar, candlestick**: a (series name), b(category value), c (value), d(null)
    - **scatter & bubble**: a (series name), b (data name), c (value array), d(null)
    - **map**: a (series name), b (area name), c(merged value), d(null)
    - **pie**, **radar**, **gauge**, **funnel**: a (series name), b (data item name), c(value), d (pie: percent | radar: indicator name)
    - **force, chord** :
      - node : a (series name), b (node name), c (node value), d (node index), e
      - link : a (series name), b (link name, default bigEndNodeName-smallEndNodeName), c(link.value), d(name or index of big end node), e(name or index of small end node)
- **{Function}**, the list of parameters passed in is [params, ticket, callback], detail as follows:
  - params : array content and template varaibles are the same

```
[
    {

        seriesIndex: 0,
        seriesName: 'Sales',
        name: 'Monday',
        dataIndex: 0,
        data: data,
        name: name,
        value: value,
        percent: special,       // pie
        indicator: special,     // radar, force, chord
        value2: special2,       // force, chord
        indicator2: special2    // force, chord
    },
    {..},
    ...
]
```

  - ticket: asynchronous callback marker
  - callback: asynchronous callback, the callback needs two parameters, the first one is the ticket that we mentioned earlier, the second is the fill content html.
  - ** When the function callbacks, 'this' pointer points to the current chart instance (myChart).*

label text formatter, universal, same as Tooltip.formatter, Supports template, method callback. Does not support asynchronous callback.

3.22.96 chart.series.i[].categories.itemStyle.emphasis.label.textStyle (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: Object
Default Value: null
style for label text.

3.22.97 chart.series.i[].categories.itemStyle.emphasis.label.textStyle.color (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: color
Default Value: @TODO: 各异
color

3.22.98 chart.series.i[].categories.itemStyle.emphasis.label.textStyle.align (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: string
Default Value: @TODO: 各异
horizontal alignment. Valid values are: 'left' | 'right' | 'center'.

3.22.99 chart.series.i[].categories.itemStyle.emphasis.label.textStyle.baseline (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: string
Default Value: @TODO: 各异
vertical alignment. Valid values are: 'top' | 'bottom' | 'middle'.

3.22.100 chart.series.i[].categories.itemStyle.emphasis.label.textStyle.fontFamily (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font family.

3.22.101 chart.series.i[].categories.itemStyle.emphasis.label.textStyle.fontSize (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: number
Default Value: 12
font size, in px.

Copyright © 2023 TRADUI Technologies GmbH

3.22.102 chart.series.i[].categories.itemStyle.emphasis.label.textStyle.fontStyle (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

3.22.103 chart.series.i[].categories.itemStyle.emphasis.label.textStyle.fontWeight (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

3.22.104 chart.series.i[].matrix (Auch bei force)

Type: Array
Default Value: []
Adjacency matrix. Only one of links and matrix can exist. Matrix is a two-dimensional array, where [i][j] means data relationship from i to j

3.22.105 chart.series.i[].minRadius (Auch bei force)

Type: number
Default Value: 10
Minimum radius after mapping to symbol size . Only available if ribbonType is false

3.22.106 chart.series.i[].maxRadius (Auch bei force)

Type: number
Default Value: 20
Maximum radius after mapping to symbol size, . Only available if ribbonType is true

3.22.107 chart.series.i[].ribbonType

Type: boolean
Default Value: true
Chord with ribbon type is drawed with sector and ribbon shapes. Each ribbon can represent the weight of edge with the size of two ends. Chord without ribbon type is drawed with symbols and bezier curves. Which can't represent edge weight.

3.22.108 chart.series.i[].showScale

Type: boolean
Default Value: false
Specifies whether the scale will be showed. Only available if ribbonType is true

3.22.109 chart.series.i[].showScaleText

Type: boolean
Default Value: false
Specifies whether the scale text will be showed. Only available if ribbonType is true

Copyright © 2023 TRADUI Technologies GmbH

3.22.110 chart.series.i[].padding

Type: number
Default Value: 2
the distance between each sector (in degrees).

3.22.111 chart.series.i[].sort (Auch bei funnel)

Type: string
Default Value: 'none'
Data sorting. Can be one of none, ascending, descending.

3.22.112 chart.series.i[].sortSub

Type: string
Default Value: 'none'
Data sorting (chord). Can be one of none, ascending, descending.

3.22.113 chart.series.i[].nodes (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: Object
The same as series[i].data

3.22.114 chart.series.i[].nodes.name (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: string
Default Value: ' '
Name of node

3.22.115 chart.series.i[].nodes.label (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: string
Default Value: ' '
Label name of the node, name(above) as default.

3.22.116 chart.series.i[].nodes.value (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: number
Value of name

3.22.117 chart.series.i[].nodes.symbol (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: string
Default Value: 'circle'
See series (Cartesian)

Copyright © 2023 TRADUI Technologies GmbH

3.22.118 chart.series.i[].nodes.symbolSize (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: number, Array
Size of symbol

3.22.119 chart.series.i[].nodes.itemStyle (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: Object
item style. see itemStyle. Notice that itemStyle in a single node of force has no nodeStyle under the normal(emphasis) option. Instead itemStyle.normal(emphasis).color|borderWidth|borderColor will be used

3.22.120 chart.series.i[].nodes.itemStyle.normal (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: Object
normal item style

3.22.121 chart.series.i[].nodes.itemStyle.normal.color (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: color, Function
Default Value: @TODO: 图表各异
color, dominant color, function paramater is {seriesIndex:x, series:xxx, dataIndex:y, data:yyy}

3.22.122 chart.series.i[].nodes.itemStyle.normal.chordStyle (Noch nicht implementiert)

Type: Object
Default Value: @TODO: 图表各异
chord style.

3.22.123 chart.series.i[].nodes.itemStyle.normal.chordStyle.width (Noch nicht implementiert)

Type: number
Default Value: 1
Width of bezier curve, Only available when ribbonType is false

3.22.124 chart.series.i[].nodes.itemStyle.normal.chordStyle.color (Noch nicht implementiert)

Type: string
Default Value: 'black'
Color of bezier curve, Only available when ribbonType is false

Copyright © 2023 TRADUI Technologies GmbH

3.22.125 chart.series.i[].nodes.itemStyle.normal.chordStyle.borderWidth (Noch nicht implementiert)

Type: number
Default Value: 1
Border width of ribbon shape, Only available when ribbonType is true

3.22.126 chart.series.i[].nodes.itemStyle.normal.chordStyle.borderColor (Noch nicht implementiert)

Type: string
Default Value: '#999'
Border color of ribbon shape, Only available when ribbonType is true

3.22.127 chart.series.i[].nodes.itemStyle.normal.label (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: Object
Default Value: { "show": true, "position": "outer" }
In pie, displayed on the outside by default. labelLine.length controls the label's distance from the pie. In funnel, displayed on the right by default. labelLine.length controls the label's distance from the funnel. In map, the location of labels cannot be specified. position can be specified in line, column, candlestick, scatter. See below.

3.22.128 chart.series.i[].nodes.itemStyle.normal.label.show (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: boolean
Default Value: true
Specifies whether to show label. Valid values are: true (show) | false (hide).

3.22.129 chart.series.i[].nodes.itemStyle.normal.label.rotate (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: boolean
Default Value: false
applicable to chord. Label rotates automatically.

3.22.130 chart.series.i[].nodes.itemStyle.normal.label.distance (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: number
Default Value: 10
In chord, Label's distance from chord after it rotates automatically. In pie, this distance means the factor of label distance and pie radius(In ring chart, is sum of inner and outer radius), default to **0.5**.

Copyright © 2023 TRADUI Technologies GmbH

3.22.131 chart.series.i[].nodes.itemStyle.normal.label.formatter (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: string, Function
Default Value: null
content formatter: **{string}** (Template) | **{Function}**. Supports asynchronous callback. See below.

- **{string}**, Template, its variables are:
  - {a} | {a0}
  - {b} | {b0}
  - {c} | {c0}
  - {d} | {d0} (some chart types do not have this variable)
  - {e} | {e0} (some chart types do not have this variable)
  - If there are multiple sets of data, there will be multiple sets of varaibles accordingly, like {a1}, {b1}, {c1}, {d1}, {a2}, {b2}, {c2}, {d2}, ...
  - Here are what variable a, b, c and d reprensent in different chart types:
    - **line & area**, **column & bar, candlestick**: a (series name), b(category value), c (value), d(null)
    - **scatter & bubble**: a (series name), b (data name), c (value array), d(null)
    - **map**: a (series name), b (area name), c(merged value), d(null)
    - **pie**, **radar**, **gauge**, **funnel**: a (series name), b (data item name), c(value), d (pie: percent | radar: indicator name)
    - **force, chord** :
      - node : a (series name), b (node name), c (node value), d (node index), e
      - link : a (series name), b (link name, default bigEndNodeName-smallEndNodeName), c(link.value), d(name or index of big end node), e(name or index of small end node)
- **{Function}**, the list of parameters passed in is [params, ticket, callback], detail as follows:
  - params : array content and template varaibles are the same

```
[
    {

        seriesIndex: 0,
        seriesName: 'Sales',
        name: 'Monday',
        dataIndex: 0,
        data: data,
        name: name,
        value: value,
        percent: special,       // pie
        indicator: special,     // radar, force, chord
        value2: special2,       // force, chord
        indicator2: special2    // force, chord
    },
    {..},
    ...
]
```

  - ticket: asynchronous callback marker
  - callback: asynchronous callback, the callback needs two parameters, the first one is the ticket that we mentioned earlier, the second is the fill content html.
  - ** When the function callbacks, 'this' pointer points to the current chart instance (myChart).*

label text formatter, universal, same as Tooltip.formatter, Supports template, method callback. Does not support asynchronous callback.

3.22.132 chart.series.i[].nodes.itemStyle.noraml.label.textStyle (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: Object
Default Value: null
style for label text.

3.22.133 chart.series.i[].nodes.itemStyle.normal.label.textStyle.color (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: color
Default Value: @TODO: 各异
color

3.22.134 chart.series.i[].nodes.itemStyle.normal.label.textStyle.align (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: string
Default Value: @TODO: 各异
horizontal alignment. Valid values are: 'left' | 'right' | 'center'.

3.22.135 chart.series.i[].nodes.itemStyle.normal.label.textStyle.baseline (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: string
Default Value: @TODO: 各异
vertical alignment. Valid values are: 'top' | 'bottom' | 'middle'.

3.22.136 chart.series.i[].nodes.itemStyle.normal.label.textStyle.fontFamily (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font family.

3.22.137 chart.series.i[].nodes.itemStyle.normal.label.textStyle.fontSize (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: number
Default Value: 12
font size, in px.

3.22.138 chart.series.i[].nodes.itemStyle.normal.label.textStyle.fontStyle (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

Copyright © 2023 TRADUI Technologies GmbH

3.22.139 chart.series.i[].nodes.itemStyle.normal.label.textStyle.fontWeight (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

3.22.140 chart.series.i[].nodes.itemStyle.emphasis (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: Object
emphasis item style (when hovered)

3.22.141 chart.series.i[].nodes.itemStyle.emphasis.color (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: color, Function
Default Value: @TODO: 图表各异
color, dominant color, function paramater is {seriesIndex:x, series:xxx, dataIndex:y, data:yyy}

3.22.142 chart.series.i[].nodes.itemStyle.emphasis.chordStyle (Noch nicht implementiert)

Type: Object
Default Value: @TODO: 图表各异
chord style.

3.22.143 chart.series.i[].nodes.itemStyle.emphasis.chordStyle.width (Noch nicht implementiert)

Type: number
Default Value: 1
Width of bezier curve, Only available when ribbonType is false

3.22.144 chart.series.i[].nodes.itemStyle.emphasis.chordStyle.color (Noch nicht implementiert)

Type: string
Default Value: 'black'
Color of bezier curve, Only available when ribbonType is false

3.22.145 chart.series.i[].nodes.itemStyle.emphasis.chordStyle.borderWidth (Noch nicht implementiert)

Type: number
Default Value: 1
Border width of ribbon shape, Only available when ribbonType is true

Copyright © 2023 TRADUI Technologies GmbH

3.22.146 chart.series.i[].nodes.itemStyle.emphasis.chordStyle.borderColor (Noch nicht implementiert)

Type: string
Default Value: '#999'
Border color of ribbon shape, Only available when ribbonType is true

3.22.147 chart.series.i[].nodes.itemStyle.emphasis.label (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: Object
Default Value: { "show": true, "position": "outer" }
In pie, displayed on the outside by default. labelLine.length controls the label's distance from the pie. In funnel, displayed on the right by default. labelLine.length controls the label's distance from the funnel. In map, the location of labels cannot be specified. position can be specified in line, column, candlestick, scatter. See below.

3.22.148 chart.series.i[].nodes.itemStyle.emphasis.label.show (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: boolean
Default Value: true
Specifies whether to show label. Valid values are: true (show) | false (hide).

3.22.149 chart.series.i[].nodes.itemStyle.emphasis.label.rotate (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: boolean
Default Value: false
applicable to chord. Label rotates automatically.

3.22.150 chart.series.i[].nodes.itemStyle.emphasis.label.distance (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: number
Default Value: 10
In chord, Label's distance from chord after it rotates automatically. In pie, this distance means the factor of label distance and pie radius(In ring chart, is sum of inner and outer radius), default to **0.5**.

3.22.151 chart.series.i[].nodes.itemStyle.emphasis.label.formatter (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: string, Function
Default Value: null
content formatter: **{string}** (Template) | **{Function}**. Supports asynchronous callback. See below.

- **{string}**, Template, its variables are:
  - {a} | {a0}
  - {b} | {b0}
  - {c} | {c0}
  - {d} | {d0} (some chart types do not have this variable)
  - {e} | {e0} (some chart types do not have this variable)

- If there are multiple sets of data, there will be multiple sets of varaibles accordingly, like {a1}, {b1}, {c1}, {d1}, {a2}, {b2}, {c2}, {d2}, ...
- Here are what variable a, b, c and d reprensent in different chart types:
  - **line & area**, **column & bar, candlestick**: a (series name), b(category value), c (value), d(null)
  - **scatter & bubble**: a (series name), b (data name), c (value array), d(null)
  - **map**: a (series name), b (area name), c(merged value), d(null)
  - **pie**, **radar**, **gauge**, **funnel**: a (series name), b (data item name), c(value), d (pie: percent | radar: indicator name)
  - **force, chord** :
    - node : a (series name), b (node name), c (node value), d (node index), e
    - link : a (series name), b (link name, default bigEndNodeName-smallEndNodeName), c(link.value), d(name or index of big end node), e(name or index of small end node)

- **{Function}**, the list of parameters passed in is [params, ticket, callback], detail as follows:
  - params : array content and template varaibles are the same

```
[
    {

        seriesIndex: 0,
        seriesName: 'Sales',
        name: 'Monday',
        dataIndex: 0,
        data: data,
        name: name,
        value: value,
        percent: special,       // pie
        indicator: special,     // radar, force, chord
        value2: special2,       // force, chord
        indicator2: special2    // force, chord
    },
    {..},
    ...
]
```

  - ticket: asynchronous callback marker
  - callback: asynchronous callback, the callback needs two parameters, the first one is the ticket that we mentioned earlier, the second is the fill content html.
  - ** When the function callbacks, 'this' pointer points to the current chart instance (myChart).*

label text formatter, universal, same as Tooltip.formatter, Supports template, method callback. Does not support asynchronous callback.

3.22.152 chart.series.i[].nodes.itemStyle.emphasis.label.textStyle (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: Object
Default Value: null
style for label text.

3.22.153 chart.series.i[].nodes.itemStyle.emphasis.label.textStyle.color (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: color
Default Value: @TODO: 各异
color

3.22.154 chart.series.i[].nodes.itemStyle.emphasis.label.textStyle.align (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: string
Default Value: @TODO: 各异
horizontal alignment. Valid values are: 'left' | 'right' | 'center'.

3.22.155 chart.series.i[].nodes.itemStyle.emphasis.label.textStyle.baseline (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: string
Default Value: @TODO: 各异
vertical alignment. Valid values are: 'top' | 'bottom' | 'middle'.

3.22.156 chart.series.i[].nodes.itemStyle.emphasis.label.textStyle.fontFamily (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font family.

3.22.157 chart.series.i[].nodes.itemStyle.emphasis.label.textStyle.fontSize (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: number
Default Value: 12
font size, in px.

3.22.158 chart.series.i[].nodes.itemStyle.emphasis.label.textStyle.fontStyle (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

3.22.159 chart.series.i[].nodes.itemStyle.emphasis.label.textStyle.fontWeight (Noch nicht implementiert, auch bei force (noch nicht implementiert))

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

Copyright © 2023 TRADUI Technologies GmbH

3.23 Series-Heatmap

3.23.1 chart.series.i[].blurSize

Type: number
Default Value: 30
Blur size of a data point.

3.23.2 chart.series.i[].gradientColors

Type: Array
Default Value: ["blue", "cyan", "lime", "yellow", "red"]
It could be an array of offset and color like [{ offset: 0.2, color: 'blue' }, { offset 0.8, color: 'cyan' }] or an array of color strings like ['blue', 'cyan', 'lime', 'yellow', 'red'], with which the color will transform evenly.

3.23.3 chart.series.i[].minAlpha (Noch nicht implementiert)

Type: number
Default Value: 0.05
If the unified value is less than minAlpha, it will be set to minAlpha to ensure small data value can also be visible on the chart.

3.23.4 chart.series.i[].valueScale

Type: number
Default Value: 1
All data values will multiply this value.

3.23.5 chart.series.i[].opacity

Type: number
Default Value: 1
Opacity of the heatmap.

3.24 Series-ItemStyle

3.24.1 ItemStyle.normal

chart.series.i[].itemStyle.normal.color

Type: color, Function
Default Value: @TODO: 图表各异
color, dominant color, function paramater is {seriesIndex:x, series:xxx, dataIndex:y, data:yyy}

chart.series.i[].itemStyle.normal.label.distance (nur bei Chord)

Type: number
Default Vallue: 10
In chord, Label's distance from chord after it rotates automatically. In pie, this distance means the factor of label distance and pie radius(In ring chart, is sum of inner and outer radius), default to **0.5**.

Copyright © 2023 TRADUI Technologies GmbH

chart.series.i[].itemStyle.normal.label.formatter

Type: string, Function
Default Value: null
content formatter: **{string}** (Template) | **{Function}**. Supports asynchronous callback. See below.

- **{string}**, Template, its variables are:
  - {a} | {a0}
  - {b} | {b0}
  - {c} | {c0}
  - {d} | {d0} (some chart types do not have this variable)
  - {e} | {e0} (some chart types do not have this variable)
  - If there are multiple sets of data, there will be multiple sets of varaibles accordingly, like {a1}, {b1}, {c1}, {d1}, {a2}, {b2}, {c2}, {d2}, ...
  - Here are what variable a, b, c and d reprensent in different chart types:
    - **line & area**, **column & bar, candlestick**: a (series name), b(category value), c (value), d(null)
    - **scatter & bubble**: a (series name), b (data name), c (value array), d(null)
    - **map**: a (series name), b (area name), c(merged value), d(null)
    - **pie**, **radar**, **gauge**, **funnel**: a (series name), b (data item name), c(value), d (pie: percent | radar: indicator name)
    - **force, chord** :
      - node : a (series name), b (node name), c (node value), d (node index), e
      - link : a (series name), b (link name, default bigEndNodeName-smallEndNodeName), c(link.value), d(name or index of big end node), e(name or index of small end node)
- **{Function}**, the list of parameters passed in is [params, ticket, callback], detail as follows:
  - params : array content and template varaibles are the same

```
[
    {

        seriesIndex: 0,
        seriesName: 'Sales',
        name: 'Monday',
        dataIndex: 0,
        data: data,
        name: name,
        value: value,
        percent: special,       // pie
        indicator: special,     // radar, force, chord
        value2: special2,       // force, chord
        indicator2: special2    // force, chord
    },
    {..},
    ...
]
```

  - ticket: asynchronous callback marker
  - callback: asynchronous callback, the callback needs two parameters, the first one is the ticket that we mentioned earlier, the second is the fill content html.
  - ** When the function callbacks, 'this' pointer points to the current chart instance (myChart).*

label text formatter, universal, same as Tooltip.formatter, Supports template, method callback. Does not support asynchronous callback.

chart.series.i[].itemStyle.normal.label.position (Nur bei line/area, bar, scatter, pie)

Type: string
Default Value: 'top' | 'right'
Position of label. The default value is 'top' (for vertical layout) or 'right' (for horizontal layout). Can also be one of: 'inside' | 'left' | 'bottom';

chart.series.i[].itemStyle.normal.label.rotate (Nur bei Chord)

Type: boolean
Default Value: false
applicable to chord. Label rotates automatically.

chart.series.i[].itemStyle.normal.label.show

Type: boolean
Default Value: true
Specifies whether to show label. Valid values are: true (show) | false (hide).

chart.series.i[].itemStyle.normal.label.textStyle.color

Type: color
Default Value: @TODO: 各异
color

chart.series.i[].itemStyle.normal.label.textStyle.align

Type: string
Default Value: @TODO: 各异
horizontal alignment. Valid values are: 'left' | 'right' | 'center'.

chart.series.i[].itemStyle.normal.label.textStyle.baseline

Type: string
Default Value: @TODO: 各异
vertical alignment. Valid values are: 'top' | 'bottom' | 'middle'.

chart.series.i[].itemStyle.normal.label.textStyle.fontFamily

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font family.

chart.series.i[].itemStyle.normal.label.textStyle.fontSize

Type: number
Default Value: 12
font size, in px.

chart.series.i[].itemStyle.normal.label.textStyle.fontStyle

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

Copyright © 2023 TRADUI Technologies GmbH

chart.series.i[].itemStyle.normal.label.textStyle.fontWeight

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

chart.series.i[].itemStyle.normal.label.x (Nur bei Treemap)

Type: number
Default Value: @TODO: 各异
only in treemap, x-coordinate in px of text

chart.series.i[].itemStyle.normal.label.y (Nur bei Treemap)

Type: number
Default Value: @TODO: 各异
only in treemap, y-coordinate in px of text

chart.series.i[].itemStyle.normal.labelLine.show (Nur bei Pie)

Type: boolean
Default Value: true
Specifies whether to show labelLine. Valid values are: true (show) | false (hide).

chart.series.i[].itemStyle.normal.labelLine.length (Nur bei Pie)

Type: number
Default Value: 40
length of the line. Starts from the outer edge. Can be negative. In funnel, length can be 'auto'

chart.series.i[].itemStyle.normal.labelLine.lineStyle.color (Nur bei Pie)

Type: color
Default Value: @TODO: 各异
color

chart.series.i[].itemStyle.normal.labelLine.lineStyle.type (Nur bei Pie)

Type: string
Default Value: 'solid'
line style. Valid values are: 'solid' | 'dotted' | 'dashed', and can be 'curve' | 'broken' in tree structures.

chart.series.i[].itemStyle.normal.labelLine.lineStyle.width (Nur bei Pie)

Type: number
Default Value: @TODO: 各异
width of the line.

chart.series.i[].itemStyle.normal.labelLine.lineStyle.shadowColor (Nur bei Pie)

Type: color
Default Value: 'rgba(0,0,0,0)'
applicable to the main line (in IE8+). Color of the shadow. Supports rgba.

Copyright © 2023 TRADUI Technologies GmbH

chart.series.i[].itemStyle.normal.labelLine.lineStyle.shadowBlur (Nur bei Pie)

Type: number
Default Value: 5
applicable to the main line (in IE8+). Blur degree of the shadow. Valid values are positive.

chart.series.i[].itemStyle.normal.labelLine.lineStyle.shadowOffsetX (Nur bei Pie)

Type: number
Default Value: 3
applicable to the main line (in IE8+). Horizontal offset of the shadow. Right when positive, left when negative.

chart.series.i[].itemStyle.normal.labelLine.lineStyle.shadowOffsetY (Nur bei Pie)

Type: number
Default Value: 3
applicable to the main line (in IE8+). Vertical offset of the shadow. Down when positive, up when negative.

chart.series.i[].itemStyle.normal.lineStyle.color (Nur bei line/area)

Type: color
Default Value: @TODO: 各异
color

chart.series.i[].itemStyle.normal.lineStyle.type (Nur bei line/area)

Type: string
Default Value: 'solid'
line style. Valid values are: 'solid' | 'dotted' | 'dashed', and can be 'curve' | 'broken' in tree structures.

chart.series.i[].itemStyle.normal.lineStyle.width (Nur bei line/area)

Type: number
Default Value: @TODO: 各异
width of the line.

chart.series.i[].itemStyle.normal.lineStyle.shadowColor (Nur bei line/area)

Type: color
Default Value: 'rgba(0,0,0,0)'
applicable to the main line (in IE8+). Color of the shadow. Supports rgba.

chart.series.i[].itemStyle.normal.lineStyle.shadowBlur (Nur bei line/area)

Type: number
Default Value: 5
applicable to the main line (in IE8+). Blur degree of the shadow. Valid values are positive.

chart.series.i[].itemStyle.normal.lineStyle.shadowOffsetX (Nur bei line/area)

Type: number
Default Value: 3
applicable to the main line (in IE8+). Horizontal offset of the shadow. Right when positive, left when negative.

Copyright © 2023 TRADUI Technologies GmbH

chart.series.i[].itemStyle.normal.lineStyle.shadowOffsetY (Nur bei line/area)

Type: number
Default Value: 3
applicable to the main line (in IE8+). Vertical offset of the shadow. Down when positive, up when negative.

chart.series.i[].itemStyle.normal.areaStyle.borderColor (Nur bei line/area, map (noch nicht implementiert))

In Original Doku nicht vorhanden

chart.series.i[].itemStyle.normal.areaStyle.borderWidth (Nur bei line/area, map (noch nicht implementiert))

In Original Doku nicht vorhanden

chart.series.i[].itemStyle.normal.areaStyle.color (Nur bei line/area, map)

Type: color
Default Value: @TODO: 各异
color

chart.series.i[].itemStyle.normal.areaStyle.type (Nur bei line/area, map)

Type: string
Default Value: 'default'
Fill style, currently only supports 'default' (solid fill).

chart.series.i[].itemStyle.normal.barBorderColor (Nur bei bar)

Type: string
Default Value: '#fff'
color of border.

chart.series.i[].itemStyle.normal.barBorderRadius (Nur bei bar)

Type: number, Array
Default Value: 0
radius of bar (column) border, in px, defaults to 0. can be Array to assign a radius to the 4 corners, such as [5, 5, 0, 0] (clockwise as left-top, right-top, right-bottom, left-bottom)

chart.series.i[].itemStyle.normal.barBorderWidth (Nur bei bar)

Type: number
Default Value: 0
width of bar (column) border, in px, defaults to 0.

chart.series.i[].itemStyle.normal.borderColor (Nur bei line/area, scatter, pie, map)

Type: string
Default Value: @TODO: 各异
color of border.

Copyright © 2023 TRADUI Technologies GmbH

chart.series.i[].itemStyle.normal.borderWidth (Nur bei line/area, scatter, pie map)

Type: number
Default Value: @TODO: 各异
width of column border, in px.

chart.series.i[].itemStyle.normal.chordStyle.width (Nur bei Chord)

Type: number
Default Value: 1
Width of bezier curve, Only available when ribbonType is false

chart.series.i[].itemStyle.normal.chordStyle.color (Nur bei Chord)

Type: string
Default Value: 'black'
Color of bezier curve, Only available when ribbonType is false

chart.series.i[].itemStyle.normal.chordStyle.borderWidth (Nur bei Chord)

Type: number
Default Value: 1
Border width of ribbon shape, Only available when ribbonType is true

chart.series.i[].itemStyle.normal.chordStyle.borderColor (Nur bei Chord)

Type: string
Default Value: '#999'
Border color of ribbon shape, Only available when ribbonType is true

chart.series.i[].itemStyle.normal.textStyle.fontFamily (Nur bei WorldCloud)

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font family.

chart.series.i[].itemStyle.normal.textStyle.fontSize (Nur bei WorldCloud)

Type: number
Default Value: 12
font size, in px.

chart.series.i[].itemStyle.normal.textStyle.fontStyle (Nur bei WorldCloud)

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

chart.series.i[].itemStyle.normal.textStyle.fontWeight (Nur bei WorldCloud)

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

Copyright © 2023 TRADUI Technologies GmbH

chart.series.i[].itemStyle.normal.breadcrumb.show (Nur bei treemap (noch nicht implementiert))

Type: boolean
Default Value: true
show or not.

chart.series.i[].itemStyle.normal.breadcrumb.textStyle.color (Nur bei treemap (noch nicht implementiert))

Type: color
Default Value: @TODO: 各异
color

chart.series.i[].itemStyle.normal.breadcrumb.textStyle.align (Nur bei treemap (noch nicht implementiert))

Type: string
Default Value: @TODO: 各异
horizontal alignment. Valid values are: 'left' | 'right' | 'center'.

chart.series.i[].itemStyle.normal.breadcrumb.textStyle.baseline (Nur bei treemap (noch nicht implementiert))

Type: string
Default Value: @TODO: 各异
vertical alignment. Valid values are: 'top' | 'bottom' | 'middle'.

chart.series.i[].itemStyle.normal.breadcrumb.textStyle.fontFamily (Nur bei treemap (noch nicht implementiert))

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font family.

chart.series.i[].itemStyle.normal.breadcrumb.textStyle.fontSize (Nur bei treemap (noch nicht implementiert))

Type: number
Default Value: 12
font size, in px.

chart.series.i[].itemStyle.normal.breadcrumb.textStyle.fontStyle (Nur bei treemap (noch nicht implementiert))

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

Copyright © 2023 TRADUI Technologies GmbH

chart.series.i[].itemStyle.normal.breadcrumb.textStyle.fontWeight (Nur bei treemap (noch nicht implementiert))

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

chart.series.i[].itemStyle.normal.childBorderWidth (Nur bei treemap (noch nicht implementiert))

Type: number
Default Value: 1
width of second level border.

chart.series.i[].itemStyle.normal.childBorderColor (Nur bei treemap (noch nicht implementiert))

Type: color
Default Value: ' '
color of second level border.

chart.series.i[].itemStyle.normal.linkStyle.normal (Nur bei force (noch nicht implementiert))

Type: Object

chart.series.i[].itemStyle.normal.linkStyle.normal.type (Nur bei force (noch nicht implementiert))

Type: string
Default Value: 'line'
line style, can be: 'curve' | 'line'

chart.series.i[].itemStyle.normal.linkStyle.normal.color (Nur bei force (noch nicht implementiert))

Type: color
Default Value: '#5182ab'
color of the line

chart.series.i[].itemStyle.normal.linkStyle.normal.width (Nur bei force (noch nicht implementiert))

Type: number
Default Value: 1
width of the line

chart.series.i[].itemStyle.normal.linkStyle.emphasis (Nur bei force (noch nicht implementiert))

Type: Object

chart.series.i[].itemStyle.normal.linkStyle.emphasis.type (Nur bei force (noch nicht implementiert))

Type: string
Default Value: 'line'
line style, can be: 'curve' | 'line'

Copyright © 2023 TRADUI Technologies GmbH

chart.series.i[].itemStyle.normal.linkStyle.emphasis.color (Nur bei force (noch nicht implementiert))

Type: color
Default Value: '#5182ab'
color of the line

chart.series.i[].itemStyle.normal.linkStyle.emphasis.width (Nur bei force (noch nicht implementiert))

Type: number
Default Value: 1
width of the line

chart.series.i[].itemStyle.normal.nodeStyle.color (Nur bei force (noch nicht implementiert))

Type: color
Default Value: '#f08c2e'
fill color.

chart.series.i[].itemStyle.normal.nodeStyle.borderColor (Nur bei force (noch nicht implementiert))

Type: color
Default Value: '#5182ab'
stroke color.

chart.series.i[].itemStyle.normal.nodeStyle.borderWidth (Nur bei force (noch nicht implementiert))

Type: number
Default Value: 1
width of the stroke line.

3.24.2 ItemStyle.emphasis

chart.series.i[].itemStyle.emphasis.color

Type: color, Function
Default Value: @TODO: 图表各异

color, dominant color, function paramater is {seriesIndex:x, series:xxx, dataIndex:y, data:yyy}

chart.series.i[].itemStyle.emphasis.label.distance (Nur bei chord)

Type: number
Default Value: 10
In chord, Label's distance from chord after it rotates automatically. In pie, this distance means the factor of label distance and pie radius(In ring chart, is sum of inner and outer radius), default to **0.5**.

chart.series.i[].itemStyle.emphasis.label.formatter (Nur bei line/area, bar, scatter, pie, radar, chord, force, map, gauge, enentRiver, venn, treemap, tree)

Type: string, Function
Default Value: null
content formatter: **{string}** (Template) | **{Function}**. Supports asynchronous callback. See below.

- **{string}**, Template, its variables are:
  - {a} | {a0}
  - {b} | {b0}
  - {c} | {c0}
  - {d} | {d0} (some chart types do not have this variable)
  - {e} | {e0} (some chart types do not have this variable)
  - If there are multiple sets of data, there will be multiple sets of varaibles accordingly, like {a1}, {b1}, {c1}, {d1}, {a2}, {b2}, {c2}, {d2}, ...
  - Here are what variable a, b, c and d reprensent in different chart types:
    - **line & area**, **column & bar, candlestick**: a (series name), b(category value), c (value), d(null)
    - **scatter & bubble**: a (series name), b (data name), c (value array), d(null)
    - **map**: a (series name), b (area name), c(merged value), d(null)
    - **pie**, **radar**, **gauge**, **funnel**: a (series name), b (data item name), c(value), d (pie: percent | radar: indicator name)
    - **force, chord** :
      - node : a (series name), b (node name), c (node value), d (node index), e
      - link : a (series name), b (link name, default bigEndNodeName-smallEndNodeName), c(link.value), d(name or index of big end node), e(name or index of small end node)
- **{Function}**, the list of parameters passed in is [params, ticket, callback], detail as follows:
  - params : array content and template varaibles are the same

```
[
    {

        seriesIndex: 0,
        seriesName: 'Sales',
        name: 'Monday',
        dataIndex: 0,
        data: data,
        name: name,
        value: value,
        percent: special,       // pie
        indicator: special,     // radar, force, chord
        value2: special2,       // force, chord
        indicator2: special2    // force, chord
    },
    {..},
    ...
]
```

  - ticket: asynchronous callback marker
  - callback: asynchronous callback, the callback needs two parameters, the first one is the ticket that we mentioned earlier, the second is the fill content html.
  - ** When the function callbacks, 'this' pointer points to the current chart instance (myChart).*

label text formatter, universal, same as Tooltip.formatter, Supports template, method callback. Does not support asynchronous callback.

chart.series.i[].itemStyle.emphasis.label.position (Nur bei line/area, bar, scatter, pie)

Type: string
Default Value: 'top' | 'right'
Position of label. The default value is 'top' (for vertical layout) or 'right' (for horizontal layout). Can also be one of: 'inside' | 'left' | 'bottom';

chart.series.i[].itemStyle.emphasis.label.rotate (Nur bei chord)

Type: boolean
Default Value: false
applicable to chord. Label rotates automatically.

chart.series.i[].itemStyle.emphasis.label.show (Nur bei line/area, bar, scatter, pie, radar, chord, force, map, gauge, enentRiver, venn, treemap, tree)

Type: boolean
Default Value: true
Specifies whether to show label. Valid values are: true (show) | false (hide).

chart.series.i[].itemStyle.emphasis.label.textStyle.color (Nur bei line/area, bar, scatter, pie, radar, chord, force, map, gauge, enentRiver, venn, treemap, tree)

Type: color
Default Value: @TODO: 各异
color

chart.series.i[].itemStyle.emphasis.label.textStyle.align (Nur bei line/area, bar, scatter, pie, radar, chord, force, map, gauge, enentRiver, venn, treemap, tree)

Type: string
Default Value: @TODO: 各异
horizontal alignment. Valid values are: 'left' | 'right' | 'center'.

chart.series.i[].itemStyle.emphasis.label.textStyle.baseline (Nur bei line/area, bar, scatter, pie, radar, chord, force, map, gauge, enentRiver, venn, treemap, tree)

Type: string
Default Value: @TODO: 各异
vertical alignment. Valid values are: 'top' | 'bottom' | 'middle'.

chart.series.i[].itemStyle.emphasis.label.textStyle.fontFamily (Nur bei line/area, bar, scatter, pie, radar, chord, force, map, gauge, enentRiver, venn, treemap, tree)

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font family.

Copyright © 2023 TRADUI Technologies GmbH

chart.series.i[].itemStyle.emphasis.label.textStyle.fontSize (Nur bei line/area, bar, scatter, pie, radar, chord, force, map, gauge, enentRiver, venn, treemap, tree)

Type: number
Default Value: 12
font size, in px.

chart.series.i[].itemStyle.emphasis.label.textStyle.fontStyle (Nur bei line/area, bar, scatter, pie, radar, chord, force, map, gauge, enentRiver, venn, treemap, tree)

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

chart.series.i[].itemStyle.emphasis.label.textStyle.fontWeight (Nur bei line/area, bar, scatter, pie, radar, chord, force, map, gauge, enentRiver, venn, treemap, tree)

Type: string, numer
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

chart.series.i[].itemStyle.emphasis.label.x (Nur bei treemap)

Type: number
Default Value: @TODO: 各异
only in treemap, x-coordinate in px of text

chart.series.i[].itemStyle.emphasis.label.y (Nur bei treemap)

Type: number
Default Value: @TODO: 各异
only in treemap, y-coordinate in px of text

chart.series.i[].itemStyle.emphasis.labelLine.length (Nur bei pie)

Type: number
Default Value: 40
length of the line. Starts from the outer edge. Can be negative. In funnel, length can be 'auto'

chart.series.i[].itemStyle.emphasis.labelLine.show (Nur bei pie)

Type: boolean
Default Value: true
Specifies whether to show labelLine. Valid values are: true (show) | false (hide).

chart.series.i[].itemStyle.emphasis.labelLine.lineStyle.color (Nur bei pie)

Type: color
Default Value: @TODO: 各异
color

Copyright © 2023 TRADUI Technologies GmbH

chart.series.i[].itemStyle.emphasis.labelLine.lineStyle.type (Nur bei pie)

Type: string
Default Value: 'solid'
line style. Valid values are: 'solid' | 'dotted' | 'dashed', and can be 'curve' | 'broken' in tree structures.

chart.series.i[].itemStyle.emphasis.labelLine.lineStyle.width (Nur bei pie)

Type: number
Default Value: @TODO: 各异
width of the line.

chart.series.i[].itemStyle.emphasis.labelLine.lineStyle.shadowColor (Nur bei pie)

Type: color
Default Value: 'rgba(0,0,0,0)'
applicable to the main line (in IE8+). Color of the shadow. Supports rgba.

chart.series.i[].itemStyle.emphasis.labelLine.lineStyle.shadowBlur (Nur bei pie)

Type: number
Default Value: 5
applicable to the main line (in IE8+). Blur degree of the shadow. Valid values are positive.

chart.series.i[].itemStyle.emphasis.labelLine.lineStyle.shadowOffsetX (Nur bei pie)

Type: number
Default Value: 3
applicable to the main line (in IE8+). Horizontal offset of the shadow. Right when positive, left when negative.

chart.series.i[].itemStyle.emphasis.labelLine.lineStyle.shadowOffsetY (Nur bei pie)

Type: number
Default Value: 3
applicable to the main line (in IE8+). Horizontal offset of the shadow. Right when positive, left when negative.

chart.series.i[].itemStyle.emphasis.lineStyle.color (Nur bei line/area)

Type: color
Default Value: @TODO: 各异
color

chart.series.i[].itemStyle.emphasis.lineStyle.type (Nur bei line/area)

Type: string
Default Value: 'solid'
line style. Valid values are: 'solid' | 'dotted' | 'dashed', and can be 'curve' | 'broken' in tree structures.

chart.series.i[].itemStyle.emphasis.lineStyle.width (Nur bei line/area)

Type: number
Default Value: @TODO: 各异
width of the line.

Copyright © 2023 TRADUI Technologies GmbH

chart.series.i[].itemStyle.emphasis.lineStyle.shadowColor (Nur bei line/area)

Type: color
Default Value: 'rgba(0,0,0,0)'
applicable to the main line (in IE8+). Color of the shadow. Supports rgba.

chart.series.i[].itemStyle.emphasis.lineStyle.shadowBlur (Nur bei line/area)

Type: number
Default Value: 5
applicable to the main line (in IE8+). Blur degree of the shadow. Valid values are positive.

chart.series.i[].itemStyle.emphasis.lineStyle.shadowOffsetX (Nur bei line/area)

Type: number
default Value: 3
applicable to the main line (in IE8+). Horizontal offset of the shadow. Right when positive, left when negative.

chart.series.i[].itemStyle.emphasis.lineStyle.shadowOffsetY (Nur bei line/area)

Type: number
Default Value: 3
applicable to the main line (in IE8+). Vertical offset of the shadow. Down when positive, up when negative.

chart.series.i[].itemStyle.emphasis.textStyle.fontFamily (Nur bei WorldCloud)

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font family.

chart.series.i[].itemStyle.emphasis.textStyle.fontSize (Nur bei WorldCloud)

Type: number
Default Value: 12
font size, in px.

chart.series.i[].itemStyle.emphasis.textStyle.fontStyle (Nur bei WordlCloud)

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

chart.series.i[].itemStyle.emphasis.textStyle.fontWeight (Nur bei WorldCloud)

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

chart.series.i[].itemStyle.emphasis.areaStyle.color (Nur bei line/area, force (noch nicht implementiert), map)

Type: color
Default Value: @TODO: 各异
color

chart.series.i[].itemStyle.emphasis.areaStyle.type (Nur bei line/area, force (noch nicht implementiert), map)

Type: string
Default Value: 'default'
Fill style, currently only supports 'default' (solid fill).

chart.series.i[].itemStyle.emphasis.barBorderColor (Nur bei bar)

Type: string
Default Value: '#fff'
color of border.

chart.series.i[].itemStyle.emphasis.barBorderRadius (Nur bei bar)

Type: number, Array
Default Value: 0
radius of bar (column) border, in px, defaults to 0. can be Array to assign a radius to the 4 corners, such as [5, 5, 0, 0] (clockwise as left-top, right-top, right-bottom, left-bottom)

chart.series.i[].itemStyle.emphasis.barBorderWidth (Nur bei bar)

Type: number
Default Value: 0
width of bar (column) border, in px, defaults to 0.

chart.series.i[].itemStyle.emphasis.borderColor (Nur bei line/area, scatter, pie, map)

Type: string
Default Value: @TODO: 各异
color of border.

chart.series.i[].itemStyle.emphasis.borderWidth (Nur bei line/area, scatter, pie, map)

Type: number
Default Value: @TODO: 各异
width of column border, in px.

chart.series.i[].itemStyle.emphasis.chordStyle.width (Nur bei chord)

Type: number
Default Value: 1
Width of bezier curve, Only available when ribbonType is false

chart.series.i[].itemStyle.emphasis.chordStyle.color (Nur bei chord)

Type: string
Default Value: 'black'
Color of bezier curve, Only available when ribbonType is false

Copyright © 2023 TRADUI Technologies GmbH

chart.series.i[].itemStyle.emphasis.chordStyle.borderWidth (Nur bei chord)

Type: number
Default Value: 1
Border width of ribbon shape, Only available when ribbonType is true

chart.series.i[].itemStyle.emphasis.chordStyle.borderColor (Nur bei chord)

Type: string
Default Value: '#999'
Border color of ribbon shape, Only available when ribbonType is true

chart.series.i[].itemStyle.emphasis.breadcrumb (Nur bei treemap (noch nicht implementiert))

Type: Object
breadcrumb

chart.series.i[].itemStyle.emphasis.breadcrumb.show (Nur bei treemap (noch nicht implementiert))

Type: boolean
Default Value: true
show or not.

chart.series.i[].itemStyle.emphasis.breadcrumb.textStyle (Nur bei treemap (noch nicht implementiert))

Type: Object

chart.series.i[].itemStyle.emphasis.breadcrumb.textStyle.color (Nur bei treemap (noch nicht implementiert))

Type: color
Default Value: @TODO: 各异
color

chart.series.i[].itemStyle.emphasis.breadcrumb.textStyle.align (Nur bei treemap (noch nicht implementiert))

Type: string
Default Value: @TODO: 各异
horizontal alignment. Valid values are: 'left' | 'right' | 'center'.

chart.series.i[].itemStyle.emphasis.breadcrumb.textStyle.baseline (Nur bei treemap (noch nicht implementiert))

Type: string
Default Value: @TODO: 各异
vertical alignment. Valid values are: 'top' | 'bottom' | 'middle'.

Copyright © 2023 TRADUI Technologies GmbH

chart.series.i[].itemStyle.emphasis.breadcrumb.textStyle.fontFamily (Nur bei treemap (noch nicht implementiert))

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font family.

chart.series.i[].itemStyle.emphasis.breadcrumb.textStyle.fontSize (Nur bei treemap (noch nicht implementiert))

Type: number
Default Value: 12
font size, in px.

chart.series.i[].itemStyle.emphasis.breadcrumb.textStyle.fontStyle (Nur bei treemap (noch nicht implementiert))

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

chart.series.i[].itemStyle.emphasis.breadcrumb.textStyle.fontWeight (Nur bei treemap (noch nicht implementiert))

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

chart.series.i[].itemStyle.emphasis.childBorderWidth (Nur bei treemap (noch nicht implementiert))

Type: number
Default Value: 1
width of second level border.

chart.series.i[].itemStyle.emphasis.childBorderColor (Nur bei treemap (noch nicht implementiert))

Type: color
Default Value: ' '
color of second level border.

chart.series.i[].itemStyle.emphasis.linkStyle.normal.type (Nur bei force (noch nicht implementiert))

Type: string
Default Value: 'line'
line style, can be: 'curve' | 'line'

Copyright © 2023 TRADUI Technologies GmbH

chart.series.i[].itemStyle.emphasis.linkStyle.normal.color (Nur bei force (noch nicht implementiert))

Type: color
Default Value: '#5182ab'
color of the line

chart.series.i[].itemStyle.emphasis.linkStyle.normal.width (Nur bei force (noch nicht implementiert))

Type: number
Default Value: 1
width of the line

chart.series.i[].itemStyle.emphasis.linkStyle.emphasis (Nur bei force (noch nicht implementiert))

Type: Object

chart.series.i[].itemStyle.emphasis.linkStyle.emphasis.type (Nur bei force (noch nicht implementiert))

Type: string
Default Value: 'line'
line style, can be: 'curve' | 'line'

chart.series.i[].itemStyle.emphasis.linkStyle.emphasis.color (Nur bei force (noch nicht implementiert))

Type: color
Default Value: '#5182ab'
color of the line

chart.series.i[].itemStyle.emphasis.linkStyle.emphasis.width (Nur bei force (noch nicht implementiert))

Type: number
Default Value: 1
width of the line

chart.series.i[].itemStyle.emphasis.nodeStyle.color (Nur bei force (noch nicht implementiert))

Type: color
Default Value: '#f08c2e'
fill color.

chart.series.i[].itemStyle.emphasis.nodeStyle.borderColor (Nur bei force (noch nicht implementiert))

Type: color
Default Value: '#5182ab'
stroke color.

Copyright © 2023 TRADUI Technologies GmbH

chart.series.i[].itemStyle.emphasis.nodeStyle.borderWidth (Nur bei force (noch nicht implementiert))

Type: number
Default Value: 1
width of the stroke line.

3.25 Series-MarkLine

3.25.1 chart.series.i[].markLine.clickable

Type: boolen
Default Value: true
Specifies data graphic clickable or not, default to true, recommend to false when you do not have a click event handler.

3.25.2 chart.series.i[].markLine.symbol

Type: string
Default Value: ["circle", "arrow"]
A predefined shape or symbol for the marker. The default symbol is automatically selected from (When all symbols are used, new symbols are pulled from the start again. Set it to'none' if you don't want to show the symbol): 'circle' | 'rectangle' | 'triangle' | 'diamond' | 'emptyCircle' | 'emptyRectangle' | 'emptyTriangle' | 'emptyDiamond'. Other possible values are 'heart', 'droplet', 'pin', 'arrow' and 'star'. Although They are not included in the regular eight symbols above, you can use them at both series and data level. In addition, you can use 'star' + n (n>=3) to get N-pointed star, like 'star6' for six-pointed star. Since 1.3.5, ECharts added support for setting custom image as symbol. The URL to an image can be given on this form:'image://' + 'URL', such as 'image://../asset/ico/favicon.png'. See example 》
symbols of the start and end of markLine. If the two are the same, string can be directly passed in. Same as series' symbol.

3.25.3 chart.series.i[].markLine.symbolSize

Type: number, Function, Array
Default Value: [2, 4]
size of the symbol. It is usually a number. But if you use an image as symbol. To prevent stretching of image caused by same width and height, you can use an array as symbolSize which first element is width and the second is height. When "calculable" is enabled, it's recommended to increase symbolSize for better interactive experience. When implementing a bubble chart, symbolSize should be a Function, and the bubble size depends on the return value of the method, the parameter passed in is the current data item (value array). sizes of the start and end symbols of markLine. Half-width (radius) parameters. If the two are the same, number or function can be directly passed in. Same as series' symbolSize.

3.25.4 chart.series.i[].markLine.symbolRotate

Type: number
Default Value: null
the angle by which the symbol is rotated. Valid range is: [-180,180]. angles by which the start and end markLine symbols rotate, same as series' symbolRotate.

3.25.5 chart.series.i[].markLine.large

Type: boolean
Default Value: false
If enable optimization for large scale markline. Large scale means line number >= 2k. Optimization will batch lines drawing. Lines of series will force to have same style. And symbols of two end of line will be ignore.

3.25.6 chart.series.i[].markLine.smooth

Type: boolean
Default Value: false
smoothed line, while smooth is true, lineStyle.type can not be dashed.

3.25.7 chart.series.i[].markLine.smoothness

Type: number
Default Value: 0.2
Line smoothness. Only available when smooth is true

3.25.8 chart.series.i[].markLine.precision

Type: number
Default Value: 2
precision, use for the average markLine.

3.25.9 chart.series.i[].markLine.bundling.enable

Type: boolean
Default Value: false
If enable edge bundling.

3.25.10 chart.series.i[].markLine.bundling.maxTurningAngle

Type: number
Default Value: 45
maxTurningAngle: Max turning angle of bundled edge, ranges from 0 degree to 90 degree. Tip：Edge bundling use algorithm from "Multilevel Agglomerative Edge Bundling for Visualizing Large Graphs"

3.25.11 chart.series.i[].markLine.effect.show

Type: boolean
Default Value: false
specifies whether to show glow effect or not. Defaults to false.

3.25.12 chart.series.i[].markLine.effect.loop

Type: boolean
Default Value: true
specifies whether to play the animation in a loop. Defaults to true.

Copyright © 2023 TRADUI Technologies GmbH

3.25.13 chart.series.i[].markLine.effect.period

Type: number
Default Value: 15
animation period. No units. The larger the value, the slower. Defaults to 15.

3.25.14 chart.series.i[].markLine.effect.scaleSize

Type: number
Default Value: 2
magnification. Based on markLine lineWidth.

3.25.15 chart.series.i[].markLine.effect.color

Type: color
Default Value: null
color of the glow. The default value is pulled from the markLine itemStyle array.

3.25.16 chart.series.i[].markLine.effect.shadowColor

Type: color
Default Value: null
color of the shadow. The default value is pulled from the color array.

3.25.17 chart.series.i[].markLine.effect.shadowBlur

Type: number
Default Value: null
blur degree of shadow. The default value is based on scaleSize.

3.25.18 chart.series.i[].markLine.itemStyle.normal

Type: Objekt
normal item style

3.25.19 chart.series.i[].markLine.itemStyle.normal.color

Type: color, Function
Default Value: @TODO: 图表各异
color, dominant color, function paramater is {seriesIndex:x, series:xxx, dataIndex:y, data:yyy}

3.25.20 chart.series.i[].markLine.itemStyle.normal.lineStyle

Type: Object
Default Value: @TODO: 图表各异
line style.

3.25.21 chart.series.i[].markLine.itemStyle.normal.lineStyle.color

Type: color
Default Value: @TODO: 各异
color

Copyright © 2023 TRADUI Technologies GmbH

3.25.22 chart.series.i[].markLine.itemStyle.normal.lineStyle.type

Type: string
Default Value: 'solid'
line style. Valid values are: 'solid' | 'dotted' | 'dashed', and can be 'curve' | 'broken' in tree structures.

3.25.23 chart.series.i[].markLine.itemStyle.normal.lineStyle.width

Type: number
Default Value: @TODO: 各异
width of the line.

3.25.24 chart.series.i[].markLine.itemStyle.normal.lineStyle.shadowColor

Typ: color
Default Value: 'rgba(0,0,0,0)'
applicable to the main line (in IE8+). Color of the shadow. Supports rgba.

3.25.25 chart.series.i[].markLine.itemStyle.normal.lineStyle.shadowBlur

Type: number
Default Value: 5
applicable to the main line (in IE8+). Blur degree of the shadow. Valid values are positive.

3.25.26 chart.series.i[].markLine.itemStyle.normal.lineStyle.shadowOffsetX

Type: number
Default Value: 3
applicable to the main line (in IE8+). Horizontal offset of the shadow. Right when positive, left when negative.

3.25.27 chart.series.i[].markLine.itemStyle.normal.lineStyle.shadowOffsetY

Type: number
Default Value: 3
applicable to the main line (in IE8+). Vertical offset of the shadow. Down when positive, up when negative.

3.25.28 chart.series.i[].markLine.itemStyle.normal.label

Type: Object
Default Value: { "show": true, "position": "outer" }
In pie, displayed on the outside by default. labelLine.length controls the label's distance from the pie. In funnel, displayed on the right by default. labelLine.length controls the label's distance from the funnel. In map, the location of labels cannot be specified. position can be specified in line, column, candlestick, scatter. See below.

3.25.29 chart.series.i[].markLine.itemStyle.normal.label.show

Type: boolean
Default Value: true
Specifies whether to show label. Valid values are: true (show) | false (hide).

Copyright © 2023 TRADUI Technologies GmbH

3.25.30 chart.series.i[].markLine.itemStyle.normal.label.formatter

Type: string, Function
Default Value: null
content formatter: **{string}** (Template) | **{Function}**. Supports asynchronous callback. See below.

- **{string}**, Template, its variables are:
  - {a} | {a0}
  - {b} | {b0}
  - {c} | {c0}
  - {d} | {d0} (some chart types do not have this variable)
  - {e} | {e0} (some chart types do not have this variable)
  - If there are multiple sets of data, there will be multiple sets of varaibles accordingly, like {a1}, {b1}, {c1}, {d1}, {a2}, {b2}, {c2}, {d2}, ...
  - Here are what variable a, b, c and d reprensent in different chart types:
    - **line & area**, **column & bar, candlestick**: a (series name), b(category value), c (value), d(null)
    - **scatter & bubble**: a (series name), b (data name), c (value array), d(null)
    - **map**: a (series name), b (area name), c(merged value), d(null)
    - **pie**, **radar**, **gauge**, **funnel**: a (series name), b (data item name), c(value), d (pie: percent | radar: indicator name)
    - **force, chord** :
      - node : a (series name), b (node name), c (node value), d (node index), e
      - link : a (series name), b (link name, default bigEndNodeName-smallEndNodeName), c(link.value), d(name or index of big end node), e(name or index of small end node)
- **{Function}**, the list of parameters passed in is [params, ticket, callback], detail as follows:
  - params : array content and template varaibles are the same

```
[
    {

        seriesIndex: 0,
        seriesName: 'Sales',
        name: 'Monday',
        dataIndex: 0,
        data: data,
        name: name,
        value: value,
        percent: special,       // pie
        indicator: special,     // radar, force, chord
        value2: special2,       // force, chord
        indicator2: special2    // force, chord
    },
    {..},
    ...
]
```

  - ticket: asynchronous callback marker
  - callback: asynchronous callback, the callback needs two parameters, the first one is the ticket that we mentioned earlier, the second is the fill content html.
  - ** When the function callbacks, 'this' pointer points to the current chart instance (myChart).*

label text formatter, universal, same as Tooltip.formatter, Supports template, method callback. Does not support asynchronous callback.

3.25.31 chart.series.i[].markLine.itemStyle.normal.label.textStyle

Type: Object
Default Value: null
style for label text.

3.25.32 chart.series.i[].markLine.itemStyle.normal.label.textStyle.color

Type: color
Default Value: @TODO: 各异
color

3.25.33 chart.series.i[].markLine.itemStyle.normal.label.textStyle.align

Type: string
Default Value: @TODO: 各异
horizontal alignment. Valid values are: 'left' | 'right' | 'center'.

3.25.34 chart.series.i[].markLine.itemStyle.normal.label.textStyle.baseline

Type: string
Default Value: @TODO: 各异
vertical alignment. Valid values are: 'top' | 'bottom' | 'middle'.

3.25.35 chart.series.i[].markLine.itemStyle.normal.label.textStyle.fontFamily

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font family.

3.25.36 chart.series.i[].markLine.itemStyle.normal.label.textStyle.fontSize

Type: number
Default Value: 12
font size, in px.

3.25.37 chart.series.i[].markLine.itemStyle.normal.label.textStyle.fontStyle

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

3.25.38 chart.series.i[].markLine.itemStyle.normal.label.textStyle.fontWeight

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

3.25.39 chart.series.i[].markLine.itemStyle.emphasis

Type: Object
emphasis item style (when hovered)

Copyright © 2023 TRADUI Technologies GmbH

3.25.40 chart.series.i[].markLine.itemStyle.emphasis.color

Type: color, Function
Default Value: @TODO: 图表各异
color, dominant color, function paramater is {seriesIndex:x, series:xxx, dataIndex:y, data:yyy}

3.25.41 chart.series.i[].markLine.itemStyle.emphasis.lineStyle

Type: Object
Default Value: @TODO: 图表各异
line style.

3.25.42 chart.series.i[].markLine.itemStyle.emphasis.lineStyle.color

Type: color
Default Value: @TODO: 各异
color

3.25.43 chart.series.i[].markLine.itemStyle.emphasis.lineStyle.type

Type: string
Default Value: 'solid'
line style. Valid values are: 'solid' | 'dotted' | 'dashed', and can be 'curve' | 'broken' in tree structures.

3.25.44 chart.series.i[].markLine.itemStyle.emphasis.lineStyle.width

Type: number
Default Value: @TODO: 各异
width of the line.

3.25.45 chart.series.i[].markLine.itemStyle.emphasis.lineStyle.shadowColor

Typ: color
Default Value: 'rgba(0,0,0,0)'
applicable to the main line (in IE8+). Color of the shadow. Supports rgba.

3.25.46 chart.series.i[].markLine.itemStyle.emphasis.lineStyle.shadowBlur

Type: number
Default Value: 5
applicable to the main line (in IE8+). Blur degree of the shadow. Valid values are positive.

3.25.47 chart.series.i[].markLine.itemStyle.emphasis.lineStyle.shadowOffsetX

Type: number
Default Value: 3
applicable to the main line (in IE8+). Horizontal offset of the shadow. Right when positive, left when negative.

Copyright © 2023 TRADUI Technologies GmbH

3.25.48 chart.series.i[].markLine.itemStyle.emphasis.lineStyle.shadowOffsetY

Type: number
Default Value: 3
applicable to the main line (in IE8+). Vertical offset of the shadow. Down when positive, up when negative.

3.25.49 chart.series.i[].markLine.itemStyle.emphasis.label

Type: Object
Default Value: { "show": true, "position": "outer" }
In pie, displayed on the outside by default. labelLine.length controls the label's distance from the pie. In funnel, displayed on the right by default. labelLine.length controls the label's distance from the funnel. In map, the location of labels cannot be specified. position can be specified in line, column, candlestick, scatter. See below.

3.25.50 chart.series.i[].markLine.itemStyle.emphasis.label.show

Type: boolean
Default Value: true
Specifies whether to show label. Valid values are: true (show) | false (hide).

3.25.51 chart.series.i[].markLine.itemStyle.emphasis.label.formatter

Type: string, Function
Default Value: null
content formatter: **{string}** (Template) | **{Function}**. Supports asynchronous callback. See below.

- **{string}**, Template, its variables are:
  - {a} | {a0}
  - {b} | {b0}
  - {c} | {c0}
  - {d} | {d0} (some chart types do not have this variable)
  - {e} | {e0} (some chart types do not have this variable)
  - If there are multiple sets of data, there will be multiple sets of varaibles accordingly, like {a1}, {b1}, {c1}, {d1}, {a2}, {b2}, {c2}, {d2}, ...
  - Here are what variable a, b, c and d reprensent in different chart types:
    - **line & area**, **column & bar, candlestick**: a (series name), b(category value), c (value), d(null)
    - **scatter & bubble**: a (series name), b (data name), c (value array), d(null)
    - **map**: a (series name), b (area name), c(merged value), d(null)
    - **pie**, **radar**, **gauge**, **funnel**: a (series name), b (data item name), c(value), d (pie: percent | radar: indicator name)
    - **force, chord** :
      - node : a (series name), b (node name), c (node value), d (node index), e
      - link : a (series name), b (link name, default bigEndNodeName-smallEndNodeName), c(link.value), d(name or index of big end node), e(name or index of small end node)
- **{Function}**, the list of parameters passed in is [params, ticket, callback], detail as follows:
  - params : array content and template varaibles are the same

```
[
    {

        seriesIndex: 0,
        seriesName: 'Sales',
        name: 'Monday',
        dataIndex: 0,
```

```
            data: data,
            name: name,
            value: value,
            percent: special,       // pie
            indicator: special,     // radar, force, chord
            value2: special2,       // force, chord
            indicator2: special2    // force, chord
        },
        {..},
        ...
    ]
```

- ticket: asynchronous callback marker
- callback: asynchronous callback, the callback needs two parameters, the first one is the ticket that we mentioned earlier, the second is the fill content html.
- ** When the function callbacks, 'this' pointer points to the current chart instance (myChart).*

label text formatter, universal, same as Tooltip.formatter, Supports template, method callback. Does not support asynchronous callback.

3.25.52 chart.series.i[].markLine.itemStyle.emphasis.label.textStyle

Type: Object
Default Value: null
style for label text.

3.25.53 chart.series.i[].markLine.itemStyle.emphasis.label.textStyle.color

Type: color
Default Value: @TODO: 各异
color

3.25.54 chart.series.i[].markLine.itemStyle.emphasis.label.textStyle.align

Type: string
Default Value: @TODO: 各异
horizontal alignment. Valid values are: 'left' | 'right' | 'center'.

3.25.55 chart.series.i[].markLine.itemStyle.emphasis.label.textStyle.baseline

Type: string
Default Value: @TODO: 各异
vertical alignment. Valid values are: 'top' | 'bottom' | 'middle'.

3.25.56 chart.series.i[].markLine.itemStyle.emphasis.label.textStyle.fontFamily

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font family.

3.25.57 chart.series.i[].markLine.itemStyle.emphasis.label.textStyle.fontSize

Type: number
Default Value: 12
font size, in px.

3.25.58 chart.series.i[].markLine.itemStyle.emphasis.label.textStyle.fontStyle

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

3.25.59 chart.series.i[].markLine.itemStyle.emphasis.label.textStyle.fontWeight

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

3.25.60 chart.series.i[].markLine.data

Type: Array
Default Value: []
data of markLine

3.26 Series-MarkPoint

3.26.1 chart.series.i[].markPoint.clickable

Type: boolean
Default Value: true
Specifies data graphic clickable or not, default to true, recommend to false when you do not have a click event handler.

3.26.2 chart.series.i[].markPoint.symbol

Type: string
Default Value: 'pin'
A predefined shape or symbol for the marker. The default symbol is automatically selected from (When all symbols are used, new symbols are pulled from the start again. Set it to'none' if you don't want to show the symbol): 'circle' | 'rectangle' | 'triangle' | 'diamond' | 'emptyCircle' | 'emptyRectangle' | 'emptyTriangle' | 'emptyDiamond' Other possible values are 'heart', 'droplet', 'pin', 'arrow' and 'star'. Although They are not included in the regular eight symbols above, you can use them at both series and data level. In addition, you can use 'star' + n (n>=3) to get N-pointed star, like 'star6' for six-pointed star. Since 1.3.5, ECharts added support for setting custom image as symbol. The URL to an image can be given on this form:'image://' + 'URL', such as 'image://../asset/ico/favicon.png'. See example 》
the symbol of markPoint, same as series' symbol.

3.26.3 chart.series.i[].markPoint.symbolSize

Type: number, Function, Array
Default Value: 10
size of the symbol. It is usually a number. But if you use an image as symbol. To prevent stretching of image caused

by same width and height, you can use an array as symbolSize which first element is width and the second is height. When "calculable" is enabled, it's recommended to increase symbolSize for better interactive experience. When implementing a bubble chart, symbolSize should be a Function, and the bubble size depends on the return value of the method, the parameter passed in is the current data item (value array). the size of markPoint symbol, same as series' symbolSize.

3.26.4 chart.series.i[].markPoint.symbolRotate

Type: number
Default Value: null
the angle by which the symbol is rotated. Valid range is: [-180,180]. the angle by which markPoint symbol rotates, same as series' symbolRotate.

3.26.5 chart.series.i[].markPoint.large

Type: boolean
Default Value: false
Specifies whether the large scale markPoint mode will be enabled.

3.26.6 chart.series.i[].markPoint.effect.show

Type: boolean
Default Value: false
specifies whether to show glow effect or not. Defaults to false.

3.26.7 chart.series.i[].markPoint.effect.type

Type: string
Default Value: 'scale'
effect type, default to 'scale', can be 'bounce'

3.26.8 chart.series.i[].markPoint.effect.loop

Type: boolean
Default Value: true
specifies whether to play the animation in a loop. Defaults to true.

3.26.9 chart.series.i[].markPoint.effect.period

Type: number
Default Value: 15
animation period. No units. The larger the value, the slower. Defaults to 15.

3.26.10 chart.series.i[].markPoint.effect.scaleSize

Type: number
Default Value: 2
magnification. Based on markPoint symbolSize, available when type is scale.

Copyright © 2023 TRADUI Technologies GmbH

3.26.11 chart.series.i[].markPoint.effect.bounceDistance

Type: number
Default Value: 10
bouncing distance, in px, available when type is bounce.

3.26.12 chart.series.i[].markPoint.effect.color

Type: color
Default Vaue: null
color of the glow. The default value is pulled from the markPoint itemStyle array.

3.26.13 chart.series.i[].markPoint.effect.shadowColor

Type: color
Default Value: null
color of the shadow. The default value is pulled from the color array.

3.26.14 chart.series.i[].markPoint.effect.shadowBlur

Type: number
Default Value: 0
blur degree of shadow. Defaults to 0.

3.26.15 chart.series.i[].markPoint.itemStyle.normal.color

Type: color, Function
Default Value: @TODO: 图表各异
color, dominant color, function paramater is {seriesIndex:x, series:xxx, dataIndex:y, data:yyy}

3.26.16 chart.series.i[].markPoint.itemStyle.normal.borderColor

Type: string
Default Value: @TODO: 各异
color of border.

3.26.17 chart.series.i[].markPoint.itemStyle.normal.borderWidth

Type: number
Default Value: @TODO: 各异
width of column border, in px.

3.26.18 chart.series.i[].markPoint.itemStyle.normal.label.show

Type: boolean
Default Value: true
Specifies whether to show label. Valid values are: true (show) | false (hide).

Copyright © 2023 TRADUI Technologies GmbH

3.26.19 chart.series.i[].markPoint.itemStyle.normal.label.formatter

Type: string, Function
Default Value: null
content formatter: **{string}** (Template) | **{Function}**. Supports asynchronous callback. See below.

- **{string}**, Template, its variables are:
  - {a} | {a0}
  - {b} | {b0}
  - {c} | {c0}
  - {d} | {d0} (some chart types do not have this variable)
  - {e} | {e0} (some chart types do not have this variable)
  - If there are multiple sets of data, there will be multiple sets of varaibles accordingly, like {a1}, {b1}, {c1}, {d1}, {a2}, {b2}, {c2}, {d2}, ...
  - Here are what variable a, b, c and d reprensent in different chart types:
    - **line & area**, **column & bar, candlestick**: a (series name), b(category value), c (value), d(null)
    - **scatter & bubble**: a (series name), b (data name), c (value array), d(null)
    - **map**: a (series name), b (area name), c(merged value), d(null)
    - **pie**, **radar**, **gauge**, **funnel**: a (series name), b (data item name), c(value), d (pie: percent | radar: indicator name)
    - **force, chord** :
      - node : a (series name), b (node name), c (node value), d (node index), e
      - link : a (series name), b (link name, default bigEndNodeName-smallEndNodeName), c(link.value), d(name or index of big end node), e(name or index of small end node)
- **{Function}**, the list of parameters passed in is [params, ticket, callback], detail as follows:
  - params : array content and template varaibles are the same

```
[
    {

        seriesIndex: 0,
        seriesName: 'Sales',
        name: 'Monday',
        dataIndex: 0,
        data: data,
        name: name,
        value: value,
        percent: special,       // pie
        indicator: special,     // radar, force, chord
        value2: special2,       // force, chord
        indicator2: special2    // force, chord
    },
    {..},
    ...
]
```

  - ticket: asynchronous callback marker
  - callback: asynchronous callback, the callback needs two parameters, the first one is the ticket that we mentioned earlier, the second is the fill content html.
  - ** When the function callbacks, 'this' pointer points to the current chart instance (myChart).*

label text formatter, universal, same as Tooltip.formatter, Supports template, method callback. Does not support asynchronous callback.

3.26.20 chart.series.i[].markPoint.itemStyle.normal.label.textStyle.color

Type: color
Default Value: @TODO: 各异
color

3.26.21 chart.series.i[].markPoint.itemStyle.normal.label.textStyle.align

Type: string
Default Value: @TODO: 各异
horizontal alignment. Valid values are: 'left' | 'right' | 'center'.

3.26.22 chart.series.i[].markPoint.itemStyle.normal.label.textStyle.baseline

Type: string
Default Value: @TODO: 各异
vertical alignment. Valid values are: 'top' | 'bottom' | 'middle'.

3.26.23 chart.series.i[].markPoint.itemStyle.normal.label.textStyle.fontFamily

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font family.

3.26.24 chart.series.i[].markPoint.itemStyle.normal.label.textStyle.fontSize

Type: number
Default Value: 12
font size, in px.

3.26.25 chart.series.i[].markPoint.itemStyle.normal.label.textStyle.fontStyle

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

3.26.26 chart.series.i[].markPoint.itemStyle.normal.label.textStyle.fontWeight

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

3.26.27 chart.series.i[].markPoint.itemStyle.emphasis.color

Type: color, Function
Default Value: @TODO: 图表各异
color, dominant color, function paramater is {seriesIndex:x, series:xxx, dataIndex:y, data:yyy}

Copyright © 2023 TRADUI Technologies GmbH

3.26.28 chart.series.i[].markPoint.itemStyle.emphasis.borderColor

Type: string
Default Value: @TODO: 各异
color of border.

3.26.29 chart.series.i[].markPoint.itemStyle.emphasis.borderWidth

Type: number
Default Value: @TODO: 各异
width of column border, in px.

3.26.30 chart.series.i[].markPoint.itemStyle.emphasis.label.show

Type: boolean
Default Value: true
Specifies whether to show label. Valid values are: true (show) | false (hide).

3.26.31 chart.series.i[].markPoint.itemStyle.emphasis.label.formatter

Type: string. Function
Default Value: null
content formatter: **{string}** (Template) | **{Function}**. Supports asynchronous callback. See below.

- **{string}**, Template, its variables are:
  - {a} | {a0}
  - {b} | {b0}
  - {c} | {c0}
  - {d} | {d0} (some chart types do not have this variable)
  - {e} | {e0} (some chart types do not have this variable)
  - If there are multiple sets of data, there will be multiple sets of varaibles accordingly, like {a1}, {b1}, {c1}, {d1}, {a2}, {b2}, {c2}, {d2}, ...
  - Here are what variable a, b, c and d reprensent in different chart types:
    - **line & area**, **column & bar, candlestick**: a (series name), b(category value), c (value), d(null)
    - **scatter & bubble**: a (series name), b (data name), c (value array), d(null)
    - **map**: a (series name), b (area name), c(merged value), d(null)
    - **pie**, **radar**, **gauge**, **funnel**: a (series name), b (data item name), c(value), d (pie: percent | radar: indicator name)
    - **force, chord** :
      - node : a (series name), b (node name), c (node value), d (node index), e
      - link : a (series name), b (link name, default bigEndNodeName-smallEndNodeName), c(link.value), d(name or index of big end node), e(name or index of small end node)
- **{Function}**, the list of parameters passed in is [params, ticket, callback], detail as follows:
  - params : array content and template varaibles are the same

```
[
    {

        seriesIndex: 0,
        seriesName: 'Sales',
        name: 'Monday',
        dataIndex: 0,
        data: data,
        name: name,
```

```
            value: value,
            percent: special,       // pie
            indicator: special,     // radar, force, chord
            value2: special2,       // force, chord
            indicator2: special2    // force, chord
        },
        {..},
        ...
    ]
```

- ticket: asynchronous callback marker
- callback: asynchronous callback, the callback needs two parameters, the first one is the ticket that we mentioned earlier, the second is the fill content html.
- ** When the function callbacks, 'this' pointer points to the current chart instance (myChart).*

label text formatter, universal, same as Tooltip.formatter, Supports template, method callback. Does not support asynchronous callback.

3.26.32 chart.series.i[].markPoint.itemStyle.emphasis.label.textStyle.color

Type: color
Default Value: @TODO: 各异
color

3.26.33 chart.series.i[].markPoint.itemStyle.emphasis.label.textStyle.align

Type: string
Default Value: @TODO: 各异
horizontal alignment. Valid values are: 'left' | 'right' | 'center'.

3.26.34 chart.series.i[].markPoint.itemStyle.emphasis.label.textStyle.baseline

Type: string
Default Value: @TODO: 各异
vertical alignment. Valid values are: 'top' | 'bottom' | 'middle'.

3.26.35 chart.series.i[].markPoint.itemStyle.emphasis.label.textStyle.fontFamily

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font family.

3.26.36 chart.series.i[].markPoint.itemStyle.emphasis.label.textStyle.fontSize

Type: number
Default Value: 12
font size, in px.

Copyright © 2023 TRADUI Technologies GmbH

3.26.37 chart.series.i[].markPoint.itemStyle.emphasis.label.textStyle.fontStyle

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

3.26.38 chart.series.i[].markPoint.itemStyle.emphasis.label.textStyle.fontWeight

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

3.26.39 chart.series.i[].markPoint.data

Type: Array
Default Value: []
data of markPoint

3.27 Series-Tooltip

3.27.1 chart.series.i[].tooltip.show

Type: boolean
Default Value: true
Specifies whether to show tooltip. Valid values are: true (show) | false (hide).

3.27.2 chart.series.i[].tooltip.zlevel

Type: number
Default Value: 1
The first grade cascading control. Every zlevel will be allocated to a independent canvas, charts and components with the same zlevel will be rendered in the same canvas. The higher zlevel the closer to the top. More canvas dom will need more memory and performance, so never set too many zlevel. Most of the time, the second grade cascading control 'z' is recommended.

3.27.3 chart.series.i[].tooltip.z

Type: number
Default Value: 8
The second grade cascading control, In the same canvas (zlevel), the higher z the closer to the top.

3.27.4 chart.series.i[].tooltip.showContent

Type: boolean
Default Value: true
Specifies whether to show the content of tooltip. Set it to false when you just need the tooltip to trigger events or show axisPointer. Valid values are: true (show) | false (hide).

Copyright © 2023 TRADUI Technologies GmbH

3.27.5 chart.series.i[].tooltip.trigger

Type: string
Default Value: 'item'
Type of trigger. Defaults to 'item'. See image below. Valid values are: 'item' | 'axis'.

3.27.6 chart.series.i[].tooltip.position

Type: Array, Function
Default Value: null
Specifies position, pass in **{Array}**, like [x, y], fixed position [x, y]; pass in **{Function}**, like function([x, y]) {return [newX,newY]}. The default displayed coordinates are input parameters, the new user-specified coordinates are output return values.

3.27.7 chart.series.i[].tooltip.formatter

Type: string, Function
Default Value: null
content formatter: **{string}** (Template) | **{Function}**. Supports asynchronous callback. See below.

- **{string}**, Template, its variables are:
  - {a} | {a0}
  - {b} | {b0}
  - {c} | {c0}
  - {d} | {d0} (some chart types do not have this variable)
  - {e} | {e0} (some chart types do not have this variable)
  - If there are multiple sets of data, there will be multiple sets of varaibles accordingly, like {a1}, {b1}, {c1}, {d1}, {a2}, {b2}, {c2}, {d2}, ...
  - Here are what variable a, b, c and d reprensent in different chart types:
    - **line & area**, **column & bar, candlestick**: a (series name), b(category value), c (value), d(null)
    - **scatter & bubble**: a (series name), b (data name), c (value array), d(null)
    - **map**: a (series name), b (area name), c(merged value), d(null)
    - **pie**, **radar**, **gauge**, **funnel**: a (series name), b (data item name), c(value), d (pie: percent | radar: indicator name)
    - **force, chord** :
      - node : a (series name), b (node name), c (node value), d (node index), e
      - link : a (series name), b (link name, default bigEndNodeName-smallEndNodeName), c(link.value), d(name or index of big end node), e(name or index of small end node)
- **{Function}**, the list of parameters passed in is [params, ticket, callback], detail as follows:
  - params : array content and template varaibles are the same

```
[
    {

        seriesIndex: 0,
        seriesName: 'Sales',
        name: 'Monday',
        dataIndex: 0,
        data: data,
        name: name,
        value: value,
        percent: special,       // pie
        indicator: special,     // radar, force, chord
        value2: special2,       // force, chord
        indicator2: special2    // force, chord
```

```
        },
        {..},
        ...
    ]
```

- ticket: asynchronous callback marker
- callback: asynchronous callback, the callback needs two parameters, the first one is the ticket that we mentioned earlier, the second is the fill content html.
- ** When the function callbacks, 'this' pointer points to the current chart instance (myChart).*

3.27.8 chart.series.i[].tooltip.islandFormatter

Type: string, Function
Default Value: '{a} < br/>{b} : {c}'
island content formatter. For Drag-Recalculate only: **{string}** (Template) | **{Function}**. See below.

3.27.9 chart.series.i[].tooltip.showDelay

Type: number
Default Value: 20
The number of milliseconds to wait until the tooltip is shown when the mouse moves across a point or chart. Placing a delay on a tooltip before it is displayed can prevent frequent switch, especially when the tooltip text is requested asynchronously.

3.27.10 chart.series.i[].tooltip.hideDelay

Type: number
Default Value: 100
The number of milliseconds to wait until the tooltip is hidden when mouse out from a point or chart.

3.27.11 chart.series.i[].tooltip.transitionDuration

Type: number
Default Value. 0.4
The duration in seconds of the animated transition. If you want the tooltip to follow the mouse as it moves across a point or chart, setting showDelay to 0 is the key, but setting transitionDuration to 0 can also make a difference in interactive experiences.

3.27.12 chart.series.i[].tooltip.enterable

Type: boolean
Default Value: false
Specifies whether to let the mouse go into the tip dom, default to false. Let it be true if you need interaction like link | button etc.

3.27.13 chart.series.i[].tooltip.backgroundColor

Type: color
Default Value: 'rgba(0,0,0,0.7)'
The background color for the tooltip. Defaults to a black color (opacity to 30%).

3.27.14 chart.series.i[].tooltip.borderColor

Type: string
Default Value:'#333'
The color of the tooltip border.

3.27.15 chart.series.i[].tooltip.borderRadius

Type: number
Default Value: 4
The radius of the rounded border corners, in px, defaults to 4.

3.27.16 chart.series.i[].tooltip.borderWidth

Type: number
Default Value: 0
The pixel width of the tooltip border, defaults to 0 (no border).

3.27.17 chart.series.i[].tooltip.padding

Type: number, Array
Default Value: 5
The inner padding of the tooltip, in px, defaults to 5. Can be set as array - [top, right, bottom, left], same as css.

3.27.18 chart.series.i[].tooltip.axisPointer.type

Type: string
Default Value: 'line'
Possible value: 'line' | 'cross' | 'shadow' | 'none'. Each type has its corresponding style, see below.

3.27.19 chart.series.i[].tooltip.axisPointer.lineStyle.color

Type: color
Default Value: @TODO: 各异
color

3.27.20 chart.series.i[].tooltip.axisPointer.lineStyle.type

Type: string
Default Value: 'solid'
line style. Valid values are: 'solid' | 'dotted' | 'dashed', and can be 'curve' | 'broken' in tree structures.

3.27.21 chart.series.i[].tooltip.axisPointer.lineStyle.width

Type: number
Default Value: @TODO: 各异
width of the line

Copyright © 2023 TRADUI Technologies GmbH

3.27.22 chart.series.i[].tooltip.axisPointer.lineStyle.shadowColor

Type: color
Default Value: 'rgba(0,0,0,0)'
applicable to the main line (in IE8+). Color of the shadow. Supports rgba.

3.27.23 chart.series.i[].tooltip.axisPointer.lineStyle.shadowBlur

Type: number
Default Value: 5
applicable to the main line (in IE8+). Blur degree of the shadow. Valid values are positive.

3.27.24 chart.series.i[].tooltip.axisPointer.lineStyle.shadowOffsetX

Type: number
Default Value: 3
applicable to the main line (in IE8+). Horizontal offset of the shadow. Right when positive, left when negative.

3.27.25 chart.series.i[].tooltip.axisPointer.lineStyle.shadowOffsetY

Type: number
Default Value: 3
applicable to the main line (in IE8+). Vertical offset of the shadow. Down when positive, up when negative.

3.27.26 chart.series.i[].tooltip.axisPointer.crossStyle.color

Type: color
Default Value: @TODO: 各异
color

3.27.27 chart.series.i[].tooltip.axisPointer.crossStyle.type

Type: string
Default Value: 'solid'
line style. Valid values are: 'solid' | 'dotted' | 'dashed', and can be 'curve' | 'broken' in tree structures.

3.27.28 chart.series.i[].tooltip.axisPointer.crossStyle.width

Type: number
Default Value: @TODO: 各异
width of the line

3.27.29 chart.series.i[].tooltip.axisPointer.crossStyle.shadowColor

Type: color
Default Value: 'rgba(0,0,0,0)'
applicable to the main line (in IE8+). Color of the shadow. Supports rgba.

Copyright © 2023 TRADUI Technologies GmbH

3.27.30 chart.series.i[].tooltip.axisPointer.crossStyle.shadowBlur

Type: number
Default Value: 5
applicable to the main line (in IE8+). Blur degree of the shadow. Valid values are positive.

3.27.31 chart.series.i[].tooltip.axisPointer.crossStyle.shadowOffsetX

Type: number
Default Value: 3
applicable to the main line (in IE8+). Horizontal offset of the shadow. Right when positive, left when negative.

3.27.32 chart.series.i[].tooltip.axisPointer.crossStyle.shadowOffsetY

Type: number
Default Value: 3
applicable to the main line (in IE8+). Vertical offset of the shadow. Down when positive, up when negative.

3.27.33 chart.series.i[].tooltip.axisPointer.shadowStyle.color

Type: color
Default Value: @TODO: 各异
color

3.27.34 chart.series.i[].tooltip.axisPointer.shadowStyle.type

Type: string
Default Value: 'default'
Fill style, currently only supports 'default' (solid fill).

3.27.35 chart.series.i[].tooltip.axisPointer.textStyle.color

Type: color
Default Value: @TODO: 各异
color

3.27.36 chart.series.i[].tooltip.axisPointer.textStyle.decoration

Nicht in Original Doku vorhanden

3.27.37 chart.series.i[].tooltip.axisPointer.textStyle.align

Type: string
Default Value: @TODO: 各异
horizontal alignment. Valid values are: 'left' | 'right' | 'center'.

3.27.38 chart.series.i[].tooltip.axisPointer.textStyle.baseline

Type: string
Default Value: @TODO: 各异
vertical alignment. Valid values are: 'top' | 'bottom' | 'middle'.

Copyright © 2023 TRADUI Technologies GmbH

3.27.39 chart.series.i[].tooltip.axisPointer.textStyle.fontFamily

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font family.

3.27.40 chart.series.i[].tooltip.axisPointer.textStyle.fontSize

Type: number
Default Value: 12
font size, in px.

3.27.41 chart.series.i[].tooltip.axisPointer.textStyle.fontStyle

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

3.27.42 chart.series.i[].tooltip.axisPointer.textStyle.fontWeight

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

3.27.43 chart.series.i[].tooltip.textStyle.color

Type: color
Default Value: @TODO: 各异
color

3.27.44 chart.series.i[].tooltip.textStyle.decoration

Type: string
Default Value: 'none'
decoration. Applicable to tooltip.textStyle only.

3.27.45 chart.series.i[].tooltip.textStyle.align

Type: string
Default Value: @TODO: 各异
horizontal alignment. Valid values are: 'left' | 'right' | 'center'.

3.27.46 chart.series.i[].tooltip.textStyle.baseline

Type: string
Default Value: @TODO: 各异
vertical alignment. Valid values are: 'top' | 'bottom' | 'middle'.

Copyright © 2023 TRADUI Technologies GmbH

3.27.47 chart.series.i[].tooltip.textStyle.fontFamily

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font family.

3.27.48 chart.series.i[].tooltip.textStyle.fontSize

Type: number
Default Value: 12
font size, in px.

3.27.49 chart.series.i[].tooltip.textStyle.fontStyle

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

3.27.50 chart.series.i[].tooltip.textStyle.fontWeight

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

3.27.51 chart.series.i[].tooltip.textStyle.textStyle.fontWeight

nicht in Original Doku vorhanden

3.28 Series-WorldCloud

3.28.1 chart.series.i[].autoSize.enable

Nicht in Original Doku vorhanden

3.28.2 chart.series.i[].autoSize.minSize

Nicht in Original Doku vorhanden

3.28.3 chart.series.i[].size (Auch bei force, treemap)

Type: Array
Default Value: ["40%", "40%"]
Center of the word cloud. Can be relatively percent or absolutely pixel.

3.28.4 chart.series.i[].textRotation (Noch nicht implementiert)

Type: Array
Default Value: [0, 90]
List of optional text rotation. In default it has two options: horizontal(0) and vertical(90).

Copyright © 2023 TRADUI Technologies GmbH

3.28.5 chart.series.i[].center (Auch bei treemap, gauge, pie, force)

Type: Array
Default Value: ["50%", "50%"]
Center of the word cloud. Can be relatively percent or absolutely pixel.

3.29 xAxis-Optionen

3.29.1 Chart-xAxis-Optionen

Es kann mehrere x-Achsen geben. Daher steht der Platzhalter **[i]** für die Nummer der X-Achse. Also bei nur einer Achse wäre dies die 1.

chart.xAxis.[i].type

Type: string
Default Value: 'category'
type of the coordinate axis. By default, x-axis is 'category', y-axis is 'value'.

chart.xAxis.[i].show

Type: boolean
Default Value: true
Specifies whether to show axis. Valid values are: true (show) | false (hide).

chart.xAxis.[i].zlevel

Type: number
Default Value: 0

chart.xAxis.[i].z

Type: number
Default Value: 0
The second grade cascading control, In the same canvas (zlevel), the higher z the closer to the top.

chart.xAxis.[i].position

Type: string
Default Value: 'bottom' | 'left'
position of the coordinate axis. For x-axis ('category'), defaults to 'bottom'. For y-axis ('value'), defaults to 'left'. Valid values are: 'bottom' | 'top' | 'left' | 'right'.

chart.xAxis.[i].name

Type: string
Default Value: ''
name of the value axis. Defaults to null.

Copyright © 2023 TRADUI Technologies GmbH

chart.xAxis.[i].nameLocation

Type: string
Default Value: 'end'
name location of the value axis. Defaults to 'end'. Valid values are: 'start' | 'end'.

chart.xAxis.[i].nameTextStyle

Type: Object
Default Value: {}
text style for the name of the value axis. Defaults to global options, takes the main color of axisLine. Can be specified

chart.xAxis.[i].axisLabel.textStyle

Type: Object
Default Value: {}
style for the loading text.

chart.xAxis.[i].axisLabel.textStyle.color

Type: color
Default Value: '#333'
main title text color.

chart.xAxis.[i].axisLabel.textStyle.align

Type: string
Default Value: null
horizontal alignment. Valid values are: 'left' | 'right' | 'center'.

chart.xAxis.[i].axisLabel.textStyle.baseline

Type: string
Default Value:null
vertical alignment. Valid values are: 'top' | 'bottom' | 'middle'.

chart.xAxis.[i].axisLabel.textStyle.fontFamily

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font family.

chart.xAxis.[i].axisLabel.textStyle.fontSize

Type: number
Default Value: 12
font size, in px.

chart.xAxis.[i].axisLabel.textStyle.fontStyle

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

Copyright © 2023 TRADUI Technologies GmbH

chart.xAxis.[i].axisLabel.textStyle.fontWeight

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

chart.xAxis.[i].boundaryGap

Type: Array
Default Value: [0, 0]
blank border on each side of the axis. the value within the array represents percentage, [the difference between the original minimum value and the final minimum value, the difference between the original maximum value and the final maximum value].

chart.xAxis.[i].min

Type: number
Default Value: null
The pecified minimum value, such as 0. Defaults to null. Further determined by specific values. Ignores boundaryGap[0] when specified.

chart.xAxis.[i].max

Type: number
Default Value: null
The pecified maximum value, such as 100. Defaults to null. Further determined by specific values. Ignores boundaryGap[1] when specified.

chart.xAxis.[i].scale

Type: boolean
Default Value: false
If false, the value axis must start with 0. If true, you can set the minimum and maximum value of value axis as the starting and ending value of your value axis.

chart.xAxis.[i].splitNumber

Type: number
Default Value: null
Number of segments, defaults to auto split along with the min/max.

chart.xAxis.[i].logLabelBase

Type: number
Default Value: null
This option works when axis.type === 'log'. If specified, axisLabel is drawn as exponent style. For example, when logLabelBase = 4, axisLabel is drawn like 4^2, 4^3. If not specified, axisLabel is drawn like 1,000,000 as usual.

chart.xAxis.[i].logPositive

Type: Object
Default Value: null
This option works when axis.type === 'log'. If set to false, negative value is supported. It is self-adapting by default, which means that if all of the data is negative, logPositive will be set to false, otherwise true.

Copyright © 2023 TRADUI Technologies GmbH

chart.xAxis.[i].axisLine.show

Type: boolean
Default Value: true
Specifies whether to show axisLine. Defaults to true, which is a prerequisite for the following properties.

chart.xAxis.[i].axisLine.onZero

Type: boolean
Default Value: true
Locates to the coordinates whose vertical axis value is 0.

chart.xAxis.[i].axisLine.lineStyle.color

Type: color
Default Value: '#1e90ff'
color

chart.xAxis.[i].axisLine.lineStyle.type

Type: string
Default Value: 'dashed'
line style. Valid values are: 'solid' | 'dotted' | 'dashed', and can be 'curve' | 'broken' in tree structures.

chart.xAxis.[i].axisLine.lineStyle.width

Type: number
Default Value: 2
width of the line.

chart.xAxis.[i].axisLine.lineStyle.shadowColor

Type: color
Default Value: 'rgba(0,0,0,0)'
applicable to the main line (in IE8+). Color of the shadow. Supports rgba.

chart.xAxis.[i].axisLine.lineStyle.shadowBlur

Type: number
Default Value: 5
applicable to the main line (in IE8+). Blur degree of the shadow. Valid values are positive.

chart.xAxis.[i].axisLine.lineStyle.shadowOffsetX

Type: number
Default Value: 3
applicable to the main line (in IE8+). Horizontal offset of the shadow. Right when positive, left when negative.

chart.xAxis.[i].axisLine.lineStyle.shadowOffsetY

Type: number
Default Value: 3
applicable to the main line (in IE8+). Vertical offset of the shadow. Down when positive, up when negative.

Copyright © 2023 TRADUI Technologies GmbH

chart.xAxis.[i].axisTick.show

Type: boolean
Default Value: false（数值轴和时间轴）
 true（类目轴）
Specifies whether to show axisTick. Defaults to false. The following properties will be set by default values When it is true.

chart.xAxis.[i].axisTick.interval

Type: string, number, Function
Default Value: 'auto'
specifies the interval between axisTicks. Defaults to 'auto'. Valid values are:
'auto' (automatically hide those that cannot be displayed) | 0 (unhide all) |
{number}(specified by the user).
{function}callback, paramater is [index, data[index]], return true to show and false to hide

chart.xAxis.[i].axisTick.onGap

Type: boolean
Default Value: null
Specifies whether the axisTick will be positioned at regular intervals. Defaults to boundaryGap.

chart.xAxis.[i].axisTick.inside

Type: boolean
Default Value: false
Specifies whether the axisTick will be displayed inside the grid. Defaults to outside.

chart.xAxis.[i].axisTick.length

Type: number
Default Value: 5
controls the length of the line.

chart.xAxis.[i].axisTick.lineStyle.color

Type: color
Default Value: '#1e90ff'
color

chart.xAxis.[i].axisTick.lineStyle.type

Type: string
Default Value: 'dashed'
line style. Valid values are: 'solid' | 'dotted' | 'dashed', and can be 'curve' | 'broken' in tree structures.

chart.xAxis.[i].axisTick.lineStyle.width

Type: number
Default Value: 2
width of the line.

Copyright © 2023 TRADUI Technologies GmbH

chart.xAxis.[i].axisTick.lineStyle.shadowColor

Type: color
Default Value: 'rgba(0,0,0,0)'
applicable to the main line (in IE8+). Color of the shadow. Supports rgba.

chart.xAxis.[i].axisTick.lineStyle.shadowBlur

Type: number
Default Value: 5
applicable to the main line (in IE8+). Blur degree of the shadow. Valid values are positive.

chart.xAxis.[i].axisTick.lineStyle.shadowOffsetX

Type: number
Default Value: 3
applicable to the main line (in IE8+). Horizontal offset of the shadow. Right when positive, left when negative.

chart.xAxis.[i].axisTick.lineStyle.shadowOffsetY

Type: number
Default Value: 3
applicable to the main line (in IE8+). Vertical offset of the shadow. Down when positive, up when negative.

chart.xAxis.[i].axisLabel.show

Type: boolean
Default Value: true
Specifies whether to show axisLabel. Defaults to true, which is a prerequisite for the following properties.

chart.xAxis.[i].axisLabel.interval

Type: string, number, Function
Default Value: 'auto'
specifies the interval between axisLabels. Defaults to 'auto'. Valid values are:
'auto' (automatically hide those that cannot be displayed) | 0 (unhide all) |
{number}(specified by the user).
{function}callback, paramater is [index, data[index]], return true to show and false to hide

chart.xAxis.[i].axisLabel.rotate

Type: number
Default Value: 0
The angle of rotation, defaults to 0 (not rotate). Positive value for counterclockwise rotation, negative value for clockwise rotation. Valid values are: -90 ~ 90.

chart.xAxis.[i].axisLabel.margin

Type: number
Default Value: 8
The pixel distance between the axisLabel and the axis.Defaults to 8.

Copyright © 2023 TRADUI Technologies GmbH

chart.xAxis.[i].axisLabel.clickable

Type: boolean
Default Value: false
Specifies whether the axisLabel is clickable.

chart.xAxis.[i].axisLabel.formatter

Type: string, Function
Default Value: null
split name formatter: {string} (Template) | {Function}

chart.xAxis.[i].axisLabel.textStyle.color

Type: color
Default Value: 各异
color

chart.xAxis.[i].axisLabel.textStyle.align

Type: string
Default Value: 各异
horizontal alignment. Valid values are: 'left' | 'right' | 'center'.

chart.xAxis.[i].axisLabel.textStyle.baseline

Type: string
Default Value: 各异
vertical alignment. Valid values are: 'top' | 'bottom' | 'middle'.

chart.xAxis.[i].axisLabel.textStyle.fontFamily

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font family.

chart.xAxis.[i].axisLabel.textStyle.fontSize

Type: number
Default Value: 12
font size, in px.

chart.xAxis.[i].axisLabel.textStyle.fontStyle

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

chart.xAxis.[i].axisLabel.textStyle.fontWeight

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

Copyright © 2023 TRADUI Technologies GmbH

chart.xAxis.[i].splitLine.show

Type: boolean
Default Value: true
Specifies whether to show splitLine. Defaults to true, which is a prerequisite for the following properties.

chart.xAxis.[i].splitLine.onGap

Type: boolean
Default Value: null
Specifies whether the splitLine will be positioned at regular intervals. Equal to boundaryGap by default.

chart.xAxis.[i].splitLine.lineStyle.color

Type: color
Default Value: '#1e90ff'
color

chart.xAxis.[i].splitLine.lineStyle.type

Type: string
Default Value: 'dashed'
line style. Valid values are: 'solid' | 'dotted' | 'dashed', and can be 'curve' | 'broken' in tree structures.

chart.xAxis.[i].splitLine.lineStyle.width

Type: number
Default Value: 2
width of the line.

chart.xAxis.[i].splitLine.lineStyle.shadowColor

Type: color
Default Value: 'rgba(0,0,0,0)'
applicable to the main line (in IE8+). Color of the shadow. Supports rgba.

chart.xAxis.[i].splitLine.lineStyle.shadowBlur

Type: number
Default Value: 5
applicable to the main line (in IE8+). Blur degree of the shadow. Valid values are positive.

chart.xAxis.[i].splitLine.lineStyle.shadowOffsetX

Type: number
Default Value: 3
applicable to the main line (in IE8+). Horizontal offset of the shadow. Right when positive, left when negative.

chart.xAxis.[i].splitLine.lineStyle.shadowOffsetY

Type: number
Default Value: 3
applicable to the main line (in IE8+). Vertical offset of the shadow. Down when positive, up when negative.

Copyright © 2023 TRADUI Technologies GmbH

chart.xAxis.[i].splitArea.show

Type: boolean
Default Value: false
universal

chart.xAxis.[i].splitArea.onGap

Type: boolean
Default Value: null
category axis

chart.xAxis.[i].splitArea.areaStyle.color

Type: color
Default Value: 各异
color

chart.xAxis.[i].splitArea.areaStyle.type

Type: string
Default Value: 'default'
Fill style, currently only supports 'default' (solid fill).

3.30 yAxis-Optionen

3.30.1 Chart-yAxis-Optionen

Es kann mehrere Y-Achsen geben. Daher steht der Platzhalter **[i]** für die Nummer der X-Achse. Also bei nur einer Achse wäre dies die 1.

chart.yAxis.[i].type

Type: string
Default Value: 'category'
type of the coordinate axis. By default, x-axis is 'category', y-axis is 'value'.

chart.yAxis.[i].show

Type: boolean
Default Value: true
Specifies whether to show axis. Valid values are: true (show) | false (hide).

chart.yAxis.[i].zlevel

Type: number
Default Value: 0

chart.yAxis.[i].z

Type: number
Default Value: 0
The second grade cascading control, In the same canvas (zlevel), the higher z the closer to the top.

chart.yAxis.[i].position

Type: string
Default Value: 'bottom' | 'left'
position of the coordinate axis. For x-axis ('category'), defaults to 'bottom'. For y-axis ('value'), defaults to 'left'. Valid values are: 'bottom' | 'top' | 'left' | 'right'.

chart.yAxis.[i].name

Type: string
Default Value: ''
name of the value axis. Defaults to null.

chart.yAxis.[i].nameLocation

Type: string
Default Value: 'end'
name location of the value axis. Defaults to 'end'. Valid values are: 'start' | 'end'.

chart.yAxis.[i].nameTextStyle

Type: Object
Default Value: {}
text style for the name of the value axis. Defaults to global options, takes the main color of axisLine. Can be specified

chart.yAxis.[i].axisLabel.textStyle

Type: Object
Default Value: {}
style for the loading text.

chart.yAxis.[i].axisLabel.textStyle.color

Type: color
Default Value: '#333'
main title text color.

chart.yAxis.[i].axisLabel.textStyle.align

Type: string
Default Value: null
horizontal alignment. Valid values are: 'left' | 'right' | 'center'.

chart.yAxis.[i].axisLabel.textStyle.baseline

Type: string
Default Value:null
vertical alignment. Valid values are: 'top' | 'bottom' | 'middle'.

Copyright © 2023 TRADUI Technologies GmbH

chart.yAxis.[i].axisLabel.textStyle.fontFamily

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font family.

chart.yAxis.[i].axisLabel.textStyle.fontSize

Type: number
Default Value: 12
font size, in px.

chart.yAxis.[i].axisLabel.textStyle.fontStyle

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

chart.yAxis.[i].axisLabel.textStyle.fontWeight

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

chart.yAxis.[i].boundaryGap

Type: Array
Default Value: [0, 0]
blank border on each side of the axis. the value within the array represents percentage, [the difference between the original minimum value and the final minimum value, the difference between the original maximum value and the final maximum value].

chart.yAxis.[i].min

Type: number
Default Value: null
The pecified minimum value, such as 0. Defaults to null. Further determined by specific values. Ignores boundaryGap[0] when specified.

chart.yAxis.[i].max

Type: number
Default Value: null
The pecified maximum value, such as 100. Defaults to null. Further determined by specific values. Ignores boundaryGap[1] when specified.

chart.yAxis.[i].scale

Type: boolean
Default Value: false
If false, the value axis must start with 0. If true, you can set the minimum and maximum value of value axis as the starting and ending value of your value axis.

Copyright © 2023 TRADUI Technologies GmbH

chart.yAxis.[i].splitNumber

Type: number
Default Value: null
Number of segments, defaults to auto split along with the min/max.

chart.yAxis.[i].logLabelBase

Type: number
Default Value: null
This option works when axis.type === 'log'. If specified, axisLabel is drawn as exponent style. For example, when logLabelBase = 4, axisLabel is drawn like 4^2, 4^3. If not specified, axisLabel is drawn like 1,000,000 as usual.

chart.yAxis.[i].logPositive

Type: Object
Default Value: null
This option works when axis.type === 'log'. If set to false, negative value is supported. It is self-adapting by default, which means that if all of the data is negative, logPositive will be set to false, otherwise true.

chart.yAxis.[i].axisLine.show

Type: boolean
Default Value: true
Specifies whether to show axisLine. Defaults to true, which is a prerequisite for the following properties.

chart.yAxis.[i].axisLine.onZero

Type: boolean
Default Value: true
Locates to the coordinates whose vertical axis value is 0.

chart.yAxis.[i].axisLine.lineStyle.color

Type: color
Default Value: '#1e90ff'
color

chart.yAxis.[i].axisLine.lineStyle.type

Type: string
Default Value: 'dashed'
line style. Valid values are: 'solid' | 'dotted' | 'dashed', and can be 'curve' | 'broken' in tree structures.

chart.yAxis.[i].axisLine.lineStyle.width

Type: number
Default Value: 2
width of the line.

Copyright © 2023 TRADUI Technologies GmbH

chart.yAxis.[i].axisLine.lineStyle.shadowColor

Type: color
Default Value: 'rgba(0,0,0,0)'
applicable to the main line (in IE8+). Color of the shadow. Supports rgba.

chart.yAxis.[i].axisLine.lineStyle.shadowBlur

Type: number
Default Value: 5
applicable to the main line (in IE8+). Blur degree of the shadow. Valid values are positive.

chart.yAxis.[i].axisLine.lineStyle.shadowOffsetX

Type: number
Default Value: 3
applicable to the main line (in IE8+). Horizontal offset of the shadow. Right when positive, left when negative.

chart.yAxis.[i].axisLine.lineStyle.shadowOffsetY

Type: number
Default Value: 3
applicable to the main line (in IE8+). Vertical offset of the shadow. Down when positive, up when negative.

chart.yAxis.[i].axisTick.show

Type: boolean
Default Value: false（数值轴和时间轴）
 true（类目轴）
Specifies whether to show axisTick. Defaults to false. The following properties will be set by default values When it is true.

chart.yAxis.[i].axisTick.interval

Type: string, number, Function
Default Value: 'auto'
specifies the interval between axisTicks. Defaults to 'auto'. Valid values are:
'auto' (automatically hide those that cannot be displayed) | 0 (unhide all) |
{number}(specified by the user).
{function}callback, paramater is [index, data[index]], return true to show and false to hide

chart.yAxis.[i].axisTick.onGap

Type: boolean
Default Value: null
Specifies whether the axisTick will be positioned at regular intervals. Defaults to boundaryGap.

chart.yAxis.[i].axisTick.inside

Type: boolean
Default Value: false
Specifies whether the axisTick will be displayed inside the grid. Defaults to outside.

Copyright © 2023 TRADUI Technologies GmbH

chart.yAxis.[i].axisTick.length

Type: number
Default Value: 5
controls the length of the line.

chart.yAxis.[i].axisTick.lineStyle.color

Type: color
Default Value: '#1e90ff'
color

chart.yAxis.[i].axisTick.lineStyle.type

Type: string
Default Value: 'dashed'
line style. Valid values are: 'solid' | 'dotted' | 'dashed', and can be 'curve' | 'broken' in tree structures.

chart.yAxis.[i].axisTick.lineStyle.width

Type: number
Default Value: 2
width of the line.

chart.yAxis.[i].axisTick.lineStyle.shadowColor

Type: color
Default Value: 'rgba(0,0,0,0)'
applicable to the main line (in IE8+). Color of the shadow. Supports rgba.

chart.yAxis.[i].axisTick.lineStyle.shadowBlur

Type: number
Default Value: 5
applicable to the main line (in IE8+). Blur degree of the shadow. Valid values are positive.

chart.yAxis.[i].axisTick.lineStyle.shadowOffsetX

Type: number
Default Value: 3
applicable to the main line (in IE8+). Horizontal offset of the shadow. Right when positive, left when negative.

chart.yAxis.[i].axisTick.lineStyle.shadowOffsetY

Type: number
Default Value: 3
applicable to the main line (in IE8+). Vertical offset of the shadow. Down when positive, up when negative.

chart.yAxis.[i].axisLabel.show

Type: boolean
Default Value: true
Specifies whether to show axisLabel. Defaults to true, which is a prerequisite for the following properties.

Copyright © 2023 TRADUI Technologies GmbH

chart.yAxis.[i].axisLabel.interval

Type: string, number, Function
Default Value: 'auto'
specifies the interval between axisLabels. Defaults to 'auto'. Valid values are:
'auto' (automatically hide those that cannot be displayed) | 0 (unhide all) |
{number}(specified by the user).
{function}callback, paramater is [index, data[index]], return true to show and false to hide

chart.yAxis.[i].axisLabel.rotate

Type: number
Default Value: 0
The angle of rotation, defaults to 0 (not rotate). Positive value for counterclockwise rotation, negative value for clockwise rotation. Valid values are: -90 ~ 90.

chart.yAxis.[i].axisLabel.margin

Type: number
Default Value: 8
The pixel distance between the axisLabel and the axis.Defaults to 8.

chart.yAxis.[i].axisLabel.clickable

Type: boolean
Default Value: false
Specifies whether the axisLabel is clickable.

chart.yAxis.[i].axisLabel.formatter

Type: string, Function
Default Value: null
split name formatter: {string} (Template) | {Function}

chart.yAxis.[i].axisLabel.textStyle.color

Type: color
Default Value: 各异
color

chart.yAxis.[i].axisLabel.textStyle.align

Type: string
Default Value: 各异
horizontal alignment. Valid values are: 'left' | 'right' | 'center'.

chart.yAxis.[i].axisLabel.textStyle.baseline

Type: string
Default Value: 各异
vertical alignment. Valid values are: 'top' | 'bottom' | 'middle'.

Copyright © 2023 TRADUI Technologies GmbH

chart.yAxis.[i].axisLabel.textStyle.fontFamily

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font family.

chart.yAxis.[i].axisLabel.textStyle.fontSize

Type: number
Default Value: 12
font size, in px.

chart.yAxis.[i].axisLabel.textStyle.fontStyle

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

chart.yAxis.[i].axisLabel.textStyle.fontWeight

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

chart.yAxis.[i].splitLine.show

Type: boolean
Default Value: true
Specifies whether to show splitLine. Defaults to true, which is a prerequisite for the following properties.

chart.yAxis.[i].splitLine.onGap

Type: boolean
Default Value: null
Specifies whether the splitLine will be positioned at regular intervals. Equal to boundaryGap by default.

chart.yAxis.[i].splitLine.lineStyle.color

Type: color
Default Value: '#1e90ff'
color

chart.yAxis.[i].splitLine.lineStyle.type

Type: string
Default Value: 'dashed'
line style. Valid values are: 'solid' | 'dotted' | 'dashed', and can be 'curve' | 'broken' in tree structures.

chart.yAxis.[i].splitLine.lineStyle.width

Type: number
Default Value: 2
width of the line.

Copyright © 2023 TRADUI Technologies GmbH

chart.yAxis.[i].splitLine.lineStyle.shadowColor

Type: color
Default Value: 'rgba(0,0,0,0)'
applicable to the main line (in IE8+). Color of the shadow. Supports rgba.

chart.yAxis.[i].splitLine.lineStyle.shadowBlur

Type: number
Default Value: 5
applicable to the main line (in IE8+). Blur degree of the shadow. Valid values are positive.

chart.yAxis.[i].splitLine.lineStyle.shadowOffsetX

Type: number
Default Value: 3
applicable to the main line (in IE8+). Horizontal offset of the shadow. Right when positive, left when negative.

chart.yAxis.[i].splitLine.lineStyle.shadowOffsetY

Type: number
Default Value: 3
applicable to the main line (in IE8+). Vertical offset of the shadow. Down when positive, up when negative.

chart.yAxis.[i].splitArea.show

Type: boolean
Default Value: false
universal

chart.yAxis.[i].splitArea.onGap

Type: boolean
Default Value: null
category axis

chart.yAxis.[i].splitArea.areaStyle.color

Type: color
Default Value: 各异
color

chart.yAxis.[i].splitArea.areaStyle.type

Type: string
Default Value: 'default'
Fill style, currently only supports 'default' (solid fill).

Copyright © 2023 TRADUI Technologies GmbH

3.31 Series-Force

3.31.1 chart.series.i[].data.name (Noch nicht implementiert)

Type: string
Default Value: ' '
Name of node

3.31.2 chart.series.i[].data.label (Noch nicht implementiert)

Type: string
Default Value: ' '
Label name of the node, name(above) as default.

3.31.3 chart.series.i[].data.value (Noch nicht implementiert)

Type: number
Value of name

3.31.4 chart.series.i[].data.ignore (Noch nicht implementiert)

Type: boolean
Default Value: false
Specifies whether to neglect the node.

3.31.5 chart.series.i[].data.symbol (Noch nicht implementiert)

Type: string
Default Value: 'circle'
See series (Cartesian)

3.31.6 chart.series.i[].data.symbolSize (Noch nicht implementiert)

Type: number, Array
Size of symbol

3.31.7 chart.series.i[].data.category (Noch nicht implementiert)

Type: number
Category index of the node.

3.31.8 chart.series.i[].data.itemStyle (Noch nicht implementiert)

Type: Object
item style. see itemStyle. Notice that itemStyle in a single node of force has no nodeStyle under the normal(emphasis) option. Instead itemStyle.normal(emphasis).color|borderWidth|borderColor will be used

3.31.9 chart.series.i[].data.itemStyle.normal (Noch nicht implementiert)

Type: Object
normal item style

Copyright © 2023 TRADUI Technologies GmbH

3.31.10 chart.series.i[].data.itemStyle.normal.color (Noch nicht implementiert)

Type: color
Default Value: @TODO: 图表各异
color, dominant color, function paramater is {seriesIndex:x, series:xxx, dataIndex:y, data:yyy}

3.31.11 chart.series.i[].data.itemStyle.normal.nodeStyle (Noch nicht implementiert)

Type: Object
Default Value: @TODO: 图表各异
node style.

3.31.12 chart.series.i[].data.itemStyle.normal.nodeStyle.color (Noch nicht implementiert)

Type: color
Default Value: '#f08c2e'
fill color.

3.31.13 chart.series.i[].data.itemStyle.normal.nodeStyle.borderColor (Noch nicht implementiert)

Type: color
Default Value: '#5182ab'
stroke color.

3.31.14 chart.series.i[].data.itemStyle.normal.nodeStyle.borderWidth (Noch nicht implementiert)

Type: number
Default Value: 1
width of the stroke line.

3.31.15 chart.series.i[].data.itemStyle.normal.linkStyle (Noch nicht implementiert)

Type: Object
Default Value: @TODO: 图表各异
link style.

3.31.16 chart.series.i[].data.itemStyle.normal.linkStyle.normal (Noch nicht implementiert)

Type: Object

Copyright © 2023 TRADUI Technologies GmbH

3.31.17 chart.series.i[].data.itemStyle.normal.linkStyle.normal.type (Noch nicht implementiert)

Type: string
Default Value: 'line'
line style, can be: 'curve' | 'line'

3.31.18 chart.series.i[].data.itemStyle.normal.linkStyle.normal.color (Noch nicht implementiert)

Type: color
Default Value: '#5182ab'
color of the line

3.31.19 chart.series.i[].data.itemStyle.normal.linkStyle.normal.width (Noch nicht implementiert)

Type: number
Default Value: 1
width of the line

3.31.20 chart.series.i[].data.itemStyle.normal.linkStyle.emphasis (Noch nicht implementiert)

Type: Object

3.31.21 chart.series.i[].data.itemStyle.normal.linkStyle.emphasis.type (Noch nicht implementiert)

Type: string
Default Value: 'line'
line style, can be: 'curve' | 'line'

3.31.22 chart.series.i[].data.itemStyle.normal.linkStyle.emphasis.color (Noch nicht implementiert)

Type: color
Default Value: '#5182ab'
color of the line

3.31.23 chart.series.i[].data.itemStyle.normal.linkStyle.emphasis.width (Noch nicht implementiert)

Type: number
Default Value: 1
width of the line

Copyright © 2023 TRADUI Technologies GmbH

3.31.24 chart.series.i[].data.itemStyle.normal.label (Noch nicht implementiert)

Type: Object
Default Value: { "show": true, "position": "outer" }
In pie, displayed on the outside by default. labelLine.length controls the label's distance from the pie. In funnel, displayed on the right by default. labelLine.length controls the label's distance from the funnel. In map, the location of labels cannot be specified. position can be specified in line, column, candlestick, scatter. See below.

3.31.25 chart.series.i[].data.itemStyle.normal.label.show (Noch nicht implementiert)

Type: boolean
Default Value: true
Specifies whether to show label. Valid values are: true (show) | false (hide).

3.31.26 chart.series.i[].data.itemStyle.normal.label.formatter (Noch nicht implementiert)

Type: string, Function
Default Value: null
content formatter: **{string}** (Template) | **{Function}**. Supports asynchronous callback. See below.

- **{string}**, Template, its variables are:
  - {a} | {a0}
  - {b} | {b0}
  - {c} | {c0}
  - {d} | {d0} (some chart types do not have this variable)
  - {e} | {e0} (some chart types do not have this variable)
  - If there are multiple sets of data, there will be multiple sets of varaibles accordingly, like {a1}, {b1}, {c1}, {d1}, {a2}, {b2}, {c2}, {d2}, ...
  - Here are what variable a, b, c and d reprensent in different chart types:
    - **line & area**, **column & bar, candlestick**: a (series name), b(category value), c (value), d(null)
    - **scatter & bubble**: a (series name), b (data name), c (value array), d(null)
    - **map**: a (series name), b (area name), c(merged value), d(null)
    - **pie**, **radar**, **gauge**, **funnel**: a (series name), b (data item name), c(value), d (pie: percent | radar: indicator name)
    - **force, chord** :
      - node : a (series name), b (node name), c (node value), d (node index), e
      - link : a (series name), b (link name, default bigEndNodeName-smallEndNodeName), c(link.value), d(name or index of big end node), e(name or index of small end node)
- **{Function}**, the list of parameters passed in is [params, ticket, callback], detail as follows:
  - params : array content and template varaibles are the same

```
[
    {

        seriesIndex: 0,
        seriesName: 'Sales',
        name: 'Monday',
        dataIndex: 0,
        data: data,
        name: name,
        value: value,
        percent: special,       // pie
```

```
            indicator: special,     // radar, force, chord
            value2: special2,       // force, chord
            indicator2: special2    // force, chord
        },
        {..},
        ...
    ]
```

- ticket: asynchronous callback marker
- callback: asynchronous callback, the callback needs two parameters, the first one is the ticket that we mentioned earlier, the second is the fill content html.
- ** When the function callbacks, 'this' pointer points to the current chart instance (myChart).*

label text formatter, universal, same as Tooltip.formatter, Supports template, method callback. Does not support asynchronous callback.

3.31.27 chart.series.i[].data.itemStyle.normal.label.textStyle (Noch nicht implementiert)

Type: Object
Default Value: null
style for label text.

3.31.28 chart.series.i[].data.itemStyle.normal.label.textStyle.color (Noch nicht implementiert)

Type: color
Default Value: @TODO: 各异
color

3.31.29 chart.series.i[].data.itemStyle.normal.label.textStyle.align (Noch nicht implementiert)

Type: string
Default Value: @TODO: 各异
horizontal alignment. Valid values are: 'left' | 'right' | 'center'.

3.31.30 chart.series.i[].data.itemStyle.normal.label.textStyle.baseline (Noch nicht implementiert)

Type: string
Default Value: @TODO: 各异
vertical alignment. Valid values are: 'top' | 'bottom' | 'middle'.

3.31.31 chart.series.i[].data.itemStyle.normal.label.textStyle.fontFamily (Noch nicht implementiert)

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font famliy.

3.31.32 chart.series.i[].data.itemStyle.normal.label.textStyle.fontSize (Noch nicht implementiert)

Type: number
Default Value: 12
font size, in px.

3.31.33 chart.series.i[].data.itemStyle.normal.label.textStyle.fontStyle (Noch nicht implementiert)

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

3.31.34 chart.series.i[].data.itemStyle.normal.label.textStyle.fontWeight (Noch nicht implementiert)

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

3.31.35 chart.series.i[].data.itemStyle.emphasis (Noch nicht implementiert)

Type: Object
emphasis item style (when hovered)

3.31.36 chart.series.i[].data.itemStyle.emphasis.color (Noch nicht implementiert)

Type: color
Default Value: @TODO: 图表各异
color, dominant color, function paramater is {seriesIndex:x, series:xxx, dataIndex:y, data:yyy}

3.31.37 chart.series.i[].data.itemStyle.emphasis.nodeStyle (Noch nicht implementiert)

Type: Object
Default Value: @TODO: 图表各异
node style.

3.31.38 chart.series.i[].data.itemStyle.emphasis.nodeStyle.color (Noch nicht implementiert)

Type: color
Default Value: '#f08c2e'
fill color.

3.31.39 chart.series.i[].data.itemStyle.emphasis.nodeStyle.borderColor (Noch nicht implementiert)

Type: color
Default Value: '#5182ab'
stroke color.

3.31.40 chart.series.i[].data.itemStyle.emphasis.nodeStyle.borderWidth (Noch nicht implementiert)

Type: number
Default Value: 1
width of the stroke line.

3.31.41 chart.series.i[].data.itemStyle.emphasis.linkStyle (Noch nicht implementiert)

Type: Object
Default Value: @TODO: 图表各异
link style.

3.31.42 chart.series.i[].data.itemStyle.emphasis.linkStyle.normal (Noch nicht implementiert)

Type: Object

3.31.43 chart.series.i[].data.itemStyle.emphasis.linkStyle.normal.type (Noch nicht implementiert)

Type: string
Default Value: 'line'
line style, can be: 'curve' | 'line'

3.31.44 chart.series.i[].data.itemStyle.emphasis.linkStyle.normal.color (Noch nicht implementiert)

Type: color
Default Value: '#5182ab'
color of the line

3.31.45 chart.series.i[].data.itemStyle.emphasis.linkStyle.normal.width (Noch nicht implementiert)

Type: number
Default Value: 1
width of the line

Copyright © 2023 TRADUI Technologies GmbH

3.31.46 chart.series.i[].data.itemStyle.emphasis.linkStyle.emphasis (Noch nicht implementiert)

Type: Object

3.31.47 chart.series.i[].data.itemStyle.emphasis.linkStyle.emphasis.type (Noch nicht implementiert)

Type: string
Default Value: 'line'
line style, can be: 'curve' | 'line'

3.31.48 chart.series.i[].data.itemStyle.emphasis.linkStyle.emphasis.color (Noch nicht implementiert)

Type: color
Default Value: '#5182ab'
color of the line

3.31.49 chart.series.i[].data.itemStyle.emphasis.linkStyle.emphasis.width (Noch nicht implementiert)

Type: number
Default Value: 1
width of the line

3.31.50 chart.series.i[].data.itemStyle.emphasis.label (Noch nicht implementiert)

Type: Object
Default Value: { "show": true, "position": "outer" }
In pie, displayed on the outside by default. labelLine.length controls the label's distance from the pie. In funnel, displayed on the right by default. labelLine.length controls the label's distance from the funnel. In map, the location of labels cannot be specified. position can be specified in line, column, candlestick, scatter. See below.

3.31.51 chart.series.i[].data.itemStyle.emphasis.label.show (Noch nicht implementiert)

Type: boolean
Default Value: true
Specifies whether to show label. Valid values are: true (show) | false (hide).

3.31.52 chart.series.i[].data.itemStyle.emphasis.label.formatter (Noch nicht implementiert)

Type: string, Function
Default Value: null
content formatter: **{string}** (Template) | **{Function}**. Supports asynchronous callback. See below.

- **{string}**, Template, its variables are:
  - {a} | {a0}
  - {b} | {b0}

- {c} | {c0}
- {d} | {d0} (some chart types do not have this variable)
- {e} | {e0} (some chart types do not have this variable)
- If there are multiple sets of data, there will be multiple sets of varaibles accordingly, like {a1}, {b1}, {c1}, {d1}, {a2}, {b2}, {c2}, {d2}, ...
- Here are what variable a, b, c and d reprensent in different chart types:
  - **line & area**, **column & bar, candlestick**: a (series name), b(category value), c (value), d(null)
  - **scatter & bubble**: a (series name), b (data name), c (value array), d(null)
  - **map**: a (series name), b (area name), c(merged value), d(null)
  - **pie**, **radar**, **gauge**, **funnel**: a (series name), b (data item name), c(value), d (pie: percent | radar: indicator name)
  - **force, chord** :
    - node : a (series name), b (node name), c (node value), d (node index), e
    - link : a (series name), b (link name, default bigEndNodeName-smallEndNodeName), c(link.value), d(name or index of big end node), e(name or index of small end node)

- **{Function}**, the list of parameters passed in is [params, ticket, callback], detail as follows:
  - params : array content and template varaibles are the same

```
[
    {

        seriesIndex: 0,
        seriesName: 'Sales',
        name: 'Monday',
        dataIndex: 0,
        data: data,
        name: name,
        value: value,
        percent: special,       // pie
        indicator: special,     // radar, force, chord
        value2: special2,       // force, chord
        indicator2: special2    // force, chord
    },
    {..},
    ...
]
```

  - ticket: asynchronous callback marker
  - callback: asynchronous callback, the callback needs two parameters, the first one is the ticket that we mentioned earlier, the second is the fill content html.
  - ** When the function callbacks, 'this' pointer points to the current chart instance (myChart).*

label text formatter, universal, same as Tooltip.formatter, Supports template, method callback. Does not support asynchronous callback.

3.31.53 chart.series.i[].data.itemStyle.emphasis.label.textStyle (Noch nicht implementiert)

Type: Object
Default Value: null
style for label text.

3.31.54 chart.series.i[].data.itemStyle.emphasis.label.textStyle.color (Noch nicht implementiert)

Type: color
Default Value: @TODO: 各异
color

3.31.55 chart.series.i[].data.itemStyle.emphasis.label.textStyle.align (Noch nicht implementiert)

Type: string
Default Value: @TODO: 各异
horizontal alignment. Valid values are: 'left' | 'right' | 'center'.

3.31.56 chart.series.i[].data.itemStyle.emphasis.label.textStyle.baseline (Noch nicht implementiert)

Type: string
Default Value: @TODO: 各异
vertical alignment. Valid values are: 'top' | 'bottom' | 'middle'.

3.31.57 chart.series.i[].data.itemStyle.emphasis.label.textStyle.fontFamily (Noch nicht implementiert)

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font famliy.

3.31.58 chart.series.i[].data.itemStyle.emphasis.label.textStyle.fontSize (Noch nicht implementiert)

Type: number
Default Value: 12
font size, in px.

3.31.59 chart.series.i[].data.itemStyle.emphasis.label.textStyle.fontStyle (Noch nicht implementiert)

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

3.31.60 chart.series.i[].data.itemStyle.emphasis.label.textStyle.fontWeight (Noch nicht implementiert)

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

Copyright © 2023 TRADUI Technologies GmbH

3.31.61 chart.series.i[].data.initial (Noch nicht implementiert)

Type: Array
Forces to specify the initial value of the node. When not specified, it will be randomly generated within a certain range.

3.31.62 chart.series.i[].data.fixX (Noch nicht implementiert)

Type: boolean
Default Value: false
Specifies whether to fix the X-value of the node. Works with initial.

3.31.63 chart.series.i[].data.fixY (Noch nicht implementiert)

Type: boolean
Default Value: false
Specifies whether to fix the Y-value of the node. Works with initial.

3.31.64 chart.series.i[].data.draggable (Noch nicht implementiert)

Type: boolean
Specifies whether the node is draggable.

3.31.65 chart.series.i[].symbolSize (Auch bei line/area, scatter, radar, chord, tree)

Type: number, Array
size of the node.

3.31.66 chart.series.i[].large (Auch bei scatter)

Type: boolean
Default Value: false
it is suggested that you set it to true on a chart with over 500 vertices, use Barnes-Hut simulation; meanwhile, open useWorker and increase the value of steps.

3.31.67 chart.series.i[].center (Auch bei treemap, gauge, pie, worldcloud)

Type: Array
Default Value: ["50%", "50%"]
Center of the word cloud. Can be relatively percent or absolutely pixel.

3.31.68 chart.series.i[].roam (Auch bei map, tree)

Type: boolean, string
Default Value: false
Specifies whether the wheel zoom and drag roam will be enabled. Default to false (boolean), can be true (boolean) and 'scale' (string) for just enable the zoom(in/out), 'move' (string) for just enable the move.

Copyright © 2023 TRADUI Technologies GmbH

3.31.69 chart.series.i[].categories.name (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: string
Default Value: ' '
Name of category

3.31.70 chart.series.i[].categories.symbol (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: string
Default Value: 'circle'
A predefined shape or symbol for the marker. The default symbol is automatically selected from (When all symbols are used, new symbols are pulled from the start again. Set it to'none' if you don't want to show the symbol): 'circle' | 'rectangle' | 'triangle' | 'diamond' | 'emptyCircle' | 'emptyRectangle' | 'emptyTriangle' | 'emptyDiamond' Other possible values are 'heart', 'droplet', 'pin', 'arrow' and 'star'. Although They are not included in the regular eight symbols above, you can use them at both series and data level. In addition, you can use 'star' + n (n>=3) to get N-pointed star, like 'star6' for six-pointed star. Since 1.3.5, ECharts added support for setting custom image as symbol. The URL to an image can be given on this form:'image://' + 'URL', such as 'image://../asset/ico/favicon.png'. See example 》
See series (Cartesian)

3.31.71 chart.series.i[].categories.symbolSize (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: number, Array
Size of symbol

3.31.72 chart.series.i[].categories.itemStyle (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: Object
item style. see itemStyle. Notice that itemStyle in thecategory of force has no nodeStyle under the normal(emphasis) option. Instead itemStyle.normal(emphasis).color|borderWidth|borderColor will be used

3.31.73 chart.series.i[].categories.itemStyle.normal (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: Object
normal item style

3.31.74 chart.series.i[].categories.itemStyle.normal.color (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: color, Function
Default Value: @TODO: 图表各异
color, dominant color, function paramater is {seriesIndex:x, series:xxx, dataIndex:y, data:yyy}

3.31.75 chart.series.i[].categories.itemStyle.normal.nodeStyle (Noch nicht implementiert)

Type: Object
Default Value: @TODO: 图表各异
node style.

3.31.76 chart.series.i[].categories.itemStyle.normal.nodeStyle.color (Noch nicht implementiert)

Type: color
Default Value: '#f08c2e'
fill color.

3.31.77 chart.series.i[].categories.itemStyle.normal.nodeStyle.borderColor (Noch nicht implementiert)

Type: color
Default Value: '#5182ab'
stroke color.

3.31.78 chart.series.i[].categories.itemStyle.normal.nodeStyle.borderWidth (Noch nicht implementiert)

Type: number
Default Value: 1
width of the stroke line.

3.31.79 chart.series.i[].categories.itemStyle.normal.linkStyle (Noch nicht implementiert)

Type: Object
Default Value: @TODO: 图表各异
linkt style.

3.31.80 chart.series.i[].categories.itemStyle.normal.linkStyle.normal (Noch nicht implementiert)

Type: Object

3.31.81 chart.series.i[].categories.itemStyle.normal.linkStyle.normal.type (Noch nicht implementiert)

Type: string
Default Value: 'line'
line style, can be: 'curve' | 'line'

Copyright © 2023 TRADUI Technologies GmbH

3.31.82 chart.series.i[].categories.itemStyle.normal.linkStyle.normal.color (Noch nicht implementiert)

Type: color
Default Value: '#5182ab'
color of the line

3.31.83 chart.series.i[].categories.itemStyle.normal.linkStyle.normal.width (Noch nicht implementiert)

Type: number
Default Value: 1
width of the line

3.31.84 chart.series.i[].categories.itemStyle.normal.linkStyle.emphasis (Noch nicht implementiert)

Type: Object

3.31.85 chart.series.i[].categories.itemStyle.normal.linkStyle.emphasis.type (Noch nicht implementiert)

Type: string
Default Value: 'line'
line style, can be: 'curve' | 'line'

3.31.86 chart.series.i[].categories.itemStyle.normal.linkStyle.emphasis.color (Noch nicht implementiert)

Type: color
Default Value: '#5182ab'
color of the line

3.31.87 chart.series.i[].categories.itemStyle.normal.linkStyle.emphasis.width (Noch nicht implementiert)

Type: number
Default Value: 1
width of the line

3.31.88 chart.series.i[].categories.itemStyle.normal.label (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: Object
Default Value: { "show": true, "position": "outer" }
In pie, displayed on the outside by default. labelLine.length controls the label's distance from the pie. In funnel, displayed on the right by default. labelLine.length controls the label's distance from the funnel. In map, the location of labels cannot be specified. position can be specified in line, column, candlestick, scatter. See below.

3.31.89 chart.series.i[].categories.itemStyle.normal.label.show (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: boolean
Default Value: true
Specifies whether to show label. Valid values are: true (show) | false (hide).

3.31.90 chart.series.i[].categories.itemStyle.normal.label.rotate (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: boolean
Default Value: false
applicable to chord. Label rotates automatically.

3.31.91 chart.series.i[].categories.itemStyle.normal.label.distance (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: number
Default Value: 10
In chord, Label's distance from chord after it rotates automatically. In pie, this distance means the factor of label distance and pie radius(In ring chart, is sum of inner and outer radius), default to **0.5**.

3.31.92 chart.series.i[].categories.itemStyle.normal.label.formatter (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: string, Function
Default Value: null
content formatter: **{string}** (Template) | **{Function}**. Supports asynchronous callback. See below.

- **{string}**, Template, its variables are:
  - {a} | {a0}
  - {b} | {b0}
  - {c} | {c0}
  - {d} | {d0} (some chart types do not have this variable)
  - {e} | {e0} (some chart types do not have this variable)
  - If there are multiple sets of data, there will be multiple sets of varaibles accordingly, like {a1}, {b1}, {c1}, {d1}, {a2}, {b2}, {c2}, {d2}, ...
  - Here are what variable a, b, c and d reprensent in different chart types:
    - **line & area**, **column & bar, candlestick**: a (series name), b(category value), c (value), d(null)
    - **scatter & bubble**: a (series name), b (data name), c (value array), d(null)
    - **map**: a (series name), b (area name), c(merged value), d(null)
    - **pie**, **radar**, **gauge**, **funnel**: a (series name), b (data item name), c(value), d (pie: percent | radar: indicator name)
    - **force, chord** :
      - node : a (series name), b (node name), c (node value), d (node index), e
      - link : a (series name), b (link name, default bigEndNodeName-smallEndNodeName), c(link.value), d(name or index of big end node), e(name or index of small end node)
- **{Function}**, the list of parameters passed in is [params, ticket, callback], detail as follows:
  - params : array content and template varaibles are the same

```
[
    {
```

```
            seriesIndex: 0,
            seriesName: 'Sales',
            name: 'Monday',
            dataIndex: 0,
            data: data,
            name: name,
            value: value,
            percent: special,       // pie
            indicator: special,     // radar, force, chord
            value2: special2,       // force, chord
            indicator2: special2    // force, chord
        },
        {..},
        ...
    ]
```

- ticket: asynchronous callback marker
- callback: asynchronous callback, the callback needs two parameters, the first one is the ticket that we mentioned earlier, the second is the fill content html.
- ** When the function callbacks, 'this' pointer points to the current chart instance (myChart).*

label text formatter, universal, same as Tooltip.formatter, Supports template, method callback. Does not support asynchronous callback.

3.31.93 chart.series.i[].categories.itemStyle.normal.label.textStyle (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: Object
Default Value: null
style for label text.

3.31.94 chart.series.i[].categories.itemStyle.normal.label.textStyle.color (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: color
Default Value: @TODO: 各异
color

3.31.95 chart.series.i[].categories.itemStyle.normal.label.textStyle.align (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: string
Default Value: @TODO: 各异
horizontal alignment. Valid values are: 'left' | 'right' | 'center'.

3.31.96 chart.series.i[].categories.itemStyle.normal.label.textStyle.baseline (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: string
Default Value: @TODO: 各异
vertical alignment. Valid values are: 'top' | 'bottom' | 'middle'.

3.31.97 chart.series.i[].categories.itemStyle.normal.label.textStyle.fontFamily (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font family.

3.31.98 chart.series.i[].categories.itemStyle.normal.label.textStyle.fontSize (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: number
Default Value: 12
font size, in px.

3.31.99 chart.series.i[].categories.itemStyle.normal.label.textStyle.fontStyle (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

3.31.100 chart.series.i[].categories.itemStyle.normal.label.textStyle.fontWeight (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

3.31.101 chart.series.i[].categories.itemStyle.emphasis (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: Object
emphasis item style (when hovered)

3.31.102 chart.series.i[].categories.itemStyle.emphasis.color (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: color, Function
Default Value: @TODO: 图表各异
color, dominant color, function paramater is {seriesIndex:x, series:xxx, dataIndex:y, data:yyy}

3.31.103 chart.series.i[].categories.itemStyle.emphasis.nodeStyle (Noch nicht implementiert)

Type: Object
Default Value: @TODO: 图表各异
node style.

3.31.104 chart.series.i[].categories.itemStyle.emphasis.nodeStyle.color (Noch nicht implementiert)

Type: color
Default Value: '#f08c2e'
fill color.

3.31.105 chart.series.i[].categories.itemStyle.emphasis.nodeStyle.borderColor (Noch nicht implementiert)

Type: color
Default Value: '#5182ab'
stroke color.

3.31.106 chart.series.i[].categories.itemStyle.emphasis.nodeStyle.borderWidth (Noch nicht implementiert)

Type: number
Default Value: 1
width of the stroke line.

3.31.107 chart.series.i[].categories.itemStyle.emphasis.linkStyle (Noch nicht implementiert)

Type: Object
Default Value: @TODO: 图表各异
linkt style.

3.31.108 chart.series.i[].categories.itemStyle.emphasis.linkStyle.normal (Noch nicht implementiert)

Type: Object

3.31.109 chart.series.i[].categories.itemStyle.emphasis.linkStyle.normal.type (Noch nicht implementiert)

Type: string
Default Value: 'line'
line style, can be: 'curve' | 'line'

3.31.110 chart.series.i[].categories.itemStyle.emphasis.linkStyle.normal.color (Noch nicht implementiert)

Type: color
Default Value: '#5182ab'
color of the line

Copyright © 2023 TRADUI Technologies GmbH

3.31.111 chart.series.i[].categories.itemStyle.emphasis.linkStyle.normal.width (Noch nicht implementiert)

Type: number
Default Value: 1
width of the line

3.31.112 chart.series.i[].categories.itemStyle.emphasis.linkStyle.emphasis (Noch nicht implementiert)

Type: Object

3.31.113 chart.series.i[].categories.itemStyle.emphasis.linkStyle.emphasis.type (Noch nicht implementiert)

Type: string
Default Value: 'line'
line style, can be: 'curve' | 'line'

3.31.114 chart.series.i[].categories.itemStyle.emphasis.linkStyle.emphasis.color (Noch nicht implementiert)

Type: color
Default Value: '#5182ab'
color of the line

3.31.115 chart.series.i[].categories.itemStyle.emphasis.linkStyle.emphasis.width (Noch nicht implementiert)

Type: number
Default Value: 1
width of the line

3.31.116 chart.series.i[].categories.itemStyle.emphasis.label (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: Object
Default Value: { "show": true, "position": "outer" }
In pie, displayed on the outside by default. labelLine.length controls the label's distance from the pie. In funnel, displayed on the right by default. labelLine.length controls the label's distance from the funnel. In map, the location of labels cannot be specified. position can be specified in line, column, candlestick, scatter. See below.

3.31.117 chart.series.i[].categories.itemStyle.emphasis.label.show (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: boolean
Default Value: true
Specifies whether to show label. Valid values are: true (show) | false (hide).

3.31.118 chart.series.i[].categories.itemStyle.emphasis.label.rotate (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: boolean
Default Value: false
applicable to chord. Label rotates automatically.

3.31.119 chart.series.i[].categories.itemStyle.emphasis.label.distance (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: number
Default Value: 10
In chord, Label's distance from chord after it rotates automatically. In pie, this distance means the factor of label distance and pie radius(In ring chart, is sum of inner and outer radius), default to **0.5**.

3.31.120 chart.series.i[].categories.itemStyle.emphasis.label.formatter (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: string, Function
Default Value: null
content formatter: **{string}** (Template) | **{Function}**. Supports asynchronous callback. See below.

- **{string}**, Template, its variables are:
  - {a} | {a0}
  - {b} | {b0}
  - {c} | {c0}
  - {d} | {d0} (some chart types do not have this variable)
  - {e} | {e0} (some chart types do not have this variable)
  - If there are multiple sets of data, there will be multiple sets of varaibles accordingly, like {a1}, {b1}, {c1}, {d1}, {a2}, {b2}, {c2}, {d2}, ...
  - Here are what variable a, b, c and d reprensent in different chart types:
    - **line & area**, **column & bar, candlestick**: a (series name), b(category value), c (value), d(null)
    - **scatter & bubble**: a (series name), b (data name), c (value array), d(null)

- **map**: a (series name), b (area name), c(merged value), d(null)
- **pie**, **radar**, **gauge**, **funnel**: a (series name), b (data item name), c(value), d (pie: percent | radar: indicator name)
- **force, chord** :
  - node : a (series name), b (node name), c (node value), d (node index), e
  - link : a (series name), b (link name, default bigEndNodeName-smallEndNodeName), c(link.value), d(name or index of big end node), e(name or index of small end node)

- **{Function}**, the list of parameters passed in is [params, ticket, callback], detail as follows:
  - params : array content and template varaibles are the same

```
[
    {

        seriesIndex: 0,
        seriesName: 'Sales',
        name: 'Monday',
        dataIndex: 0,
        data: data,
        name: name,
        value: value,
        percent: special,       // pie
        indicator: special,     // radar, force, chord
        value2: special2,       // force, chord
        indicator2: special2    // force, chord
    },
    {..},
    ...
]
```

  - ticket: asynchronous callback marker
  - callback: asynchronous callback, the callback needs two parameters, the first one is the ticket that we mentioned earlier, the second is the fill content html.
  - ** When the function callbacks, 'this' pointer points to the current chart instance (myChart).*

label text formatter, universal, same as Tooltip.formatter, Supports template, method callback. Does not support asynchronous callback.

3.31.121 chart.series.i[].categories.itemStyle.emphasis.label.textStyle (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: Object
Default Value: null
style for label text.

3.31.122 chart.series.i[].categories.itemStyle.emphasis.label.textStyle.color (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: color
Default Value: @TODO: 各异
color

3.31.123 chart.series.i[].categories.itemStyle.emphasis.label.textStyle.align (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: string
Default Value: @TODO: 各异
horizontal alignment. Valid values are: 'left' | 'right' | 'center'.

3.31.124 chart.series.i[].categories.itemStyle.emphasis.label.textStyle.baseline (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: string
Default Value: @TODO: 各异
vertical alignment. Valid values are: 'top' | 'bottom' | 'middle'.

3.31.125 chart.series.i[].categories.itemStyle.emphasis.label.textStyle.fontFamily (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font family.

3.31.126 chart.series.i[].categories.itemStyle.emphasis.label.textStyle.fontSize (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: number
Default Value: 12
font size, in px.

3.31.127 chart.series.i[].categories.itemStyle.emphasis.label.textStyle.fontStyle (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

3.31.128 chart.series.i[].categories.itemStyle.emphasis.label.textStyle.fontWeight (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

3.31.129 chart.series.i[].links.source (Noch nicht implementiert)

Type: string, number
Default Value: ' '
Index or name of source node

3.31.130 chart.series.i[].links.target (Noch nicht implementiert)

Type: string, number
Default Value: ' '
Index or name of target node

3.31.131 chart.series.i[].links.weight (Noch nicht implementiert)

Type: number
Default Value: 1
Weight of link

3.31.132 chart.series.i[].links.itemStyle (Noch nicht implementiert)

Type: Object
see linkStyle

3.31.133 chart.series.i[].links.itemStyle.normal (Noch nicht implementiert)

Type: Object

3.31.134 chart.series.i[].links.itemStyle.normal.type (Noch nicht implementiert)

Type: string
Default Value: 'line'
line style, can be: 'curve' | 'line'

3.31.135 chart.series.i[].links.itemStyle.normal.color (Noch nicht implementiert)

Type: color
Default Value: '#5182ab'
color of the line

3.31.136 chart.series.i[].links.itemStyle.normal.width (Noch nicht implementiert)

Type: number
Default Value: 1
width of the line

3.31.137 chart.series.i[].links.itemStyle.emphasis (Noch nicht implementiert)

Type: Object

3.31.138 chart.series.i[].links.itemStyle.emphasis.type (Noch nicht implementiert)

Type: string
Default Value: 'line'
line style, can be: 'curve' | 'line'

Copyright © 2023 TRADUI Technologies GmbH

3.31.139 chart.series.i[].links.itemStyle.emphasis.color (Noch nicht implementiert)

Type: color
Default Value: '#5182ab'
color of the line

3.31.140 chart.series.i[].links.itemStyle.emphasis.width (Noch nicht implementiert)

Type: number
Default Value: 1
width of the line

3.31.141 chart.series.i[].matrix (Auch bei chord)

Type: Array
Default Value: []
Adjacency matrix. Only one of links and matrix can exist. Matrix is a two-dimensional array, where [i][j] means data relationship from i to j

3.31.142 chart.series.i[].size (Auch bei treemap, worldcloud)

Type: number
Default Value: 100%
size of the layout, in pixels or in percent.

3.31.143 chart.series.i[].minRadius (Auch bei chord)

Type: number
Default Value: 10
the minimum radius after the vertex data is mapped to the circle radius.

3.31.144 chart.series.i[].maxRadius (Auch bei chord)

Type: number
Default Value: 20
the maximum radius after the vertex data is mapped to the circle radius.

3.31.145 chart.series.i[].linkSymbol

Type: Array
Default Value: 'none'
An array containing the default symbols. When all symbols are used, new symbols are pulled from the start again.
symbol of force's link. Can be specified as 'arrow'.

3.31.146 chart.series.i[].linkSymbolSize

Type: Array
Default Value: [10, 15]
the size of force's link symbol.

Copyright © 2023 TRADUI Technologies GmbH

3.31.147 chart.series.i[].scaling

Type: number
Default Value: 1
layout scaling factor, not entirely accurate, achieves similar effect with the layout size.

3.31.148 chart.series.i[].gravity

Type: number
Default Value: 1
centripetal force coefficient. The greater the coefficient, the closer the node is to the center.

3.31.149 chart.series.i[].draggable

Type: boolean
Default Value: true
specifies whether the node is draggable.

3.31.150 chart.series.i[].useWorker

Type: boolean
Default Value: false
specifies whether to put layout calculation into web worker when the browser supports web worker.

3.31.151 chart.series.i[].steps

Type: number
Default Value: 1
the number of iterations of each frame layout calculation. Since it takes much more time to render each frame than layout, so when using web worker you can increase the value of steps to balance the two, so as to achieve optimum efficiency.

3.31.152 chart.series.i[].nodes (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: Object
The same as series[i].data

3.31.153 chart.series.i[].nodes.name (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: string
Default Value: ' '
Name of node

3.31.154 chart.series.i[].nodes.label (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: string
Default Value: ' '
Label name of the node, name(above) as default.

Copyright © 2023 TRADUI Technologies GmbH

3.31.155 chart.series.i[].nodes.value (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: number
Value of name

3.31.156 chart.series.i[].nodes.ignore (Noch nicht implementiert)

Type: boolean
Default Vaue: false
Specifies whether to neglect the node.

3.31.157 chart.series.i[].nodes.symbol (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: string
Default Value: 'circle'
See series (Cartesian)

3.31.158 chart.series.i[].nodes.symbolSize (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: number, Array
Size of symbol

3.31.159 chart.series.i[].nodes.category (Noch nicht implementiert)

Type: number
Category index of the node.

3.31.160 chart.series.i[].nodes.itemStyle (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: Object
item style. see itemStyle. Notice that itemStyle in a single node of force has no nodeStyle under the normal(emphasis) option. Instead itemStyle.normal(emphasis).color|borderWidth|borderColor will be used

3.31.161 chart.series.i[].nodes.itemStyle.normal (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: Object
normal item style

3.31.162 chart.series.i[].nodes.itemStyle.normal.color (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: color, Function
Default Value: @TODO: 图表各异
color, dominant color, function paramater is {seriesIndex:x, series:xxx, dataIndex:y, data:yyy}

3.31.163 chart.series.i[].nodes.itemStyle.normal.nodeStyle (Noch nicht implementiert)

Type: Object
Default Value: @TODO: 图表各异
node style.

3.31.164 chart.series.i[].nodes.itemStyle.normal.nodeStyle.color (Noch nicht implementiert)

Type: color
Default Value: '#f08c2e'
fill color.

3.31.165 chart.series.i[].nodes.itemStyle.normal.nodeStyle.borderColor (Noch nicht implementiert)

Type: color
Default Value: '#5182ab'
stroke color.

3.31.166 chart.series.i[].nodes.itemStyle.normal.nodeStyle.borderWidth (Noch nicht implementiert)

Type: number
Default Value: 1
width of the stroke line.

3.31.167 chart.series.i[].nodes.itemStyle.normal.linkStyle (Noch nicht implementiert)

Type: Object
Default Value: @TODO: 图表各异
link style.

3.31.168 chart.series.i[].nodes.itemStyle.normal.linkStyle.normal (Noch nicht implementiert)

Type: Object

3.31.169 chart.series.i[].nodes.itemStyle.normal.linkStyle.normal.type (Noch nicht implementiert)

Type: string
Default Value: 'line'
line style, can be: 'curve' | 'line'

Copyright © 2023 TRADUI Technologies GmbH

3.31.170 chart.series.i[].nodes.itemStyle.normal.linkStyle.normal.color (Noch nicht implementiert)

Type: color
Default Value: '#5182ab'
color of the line

3.31.171 chart.series.i[].nodes.itemStyle.normal.linkStyle.normal.width (Noch nicht implementiert)

Type: number
Default Value: 1
width of the line

3.31.172 chart.series.i[].nodes.itemStyle.normal.linkStyle.emphasis (Noch nicht implementiert)

Type: Object

3.31.173 chart.series.i[].nodes.itemStyle.normal.linkStyle.emphasis.type (Noch nicht implementiert)

Type: string
Default Value: 'line'
line style, can be: 'curve' | 'line'

3.31.174 chart.series.i[].nodes.itemStyle.normal.linkStyle.emphasis.color (Noch nicht implementiert)

Type: color
Default Value: '#5182ab'
color of the line

3.31.175 chart.series.i[].nodes.itemStyle.normal.linkStyle.emphasis.width (Noch nicht implementiert)

Type: number
Default Value: 1
width of the line

3.31.176 chart.series.i[].nodes.itemStyle.normal.label (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: Object
Default Value: { "show": true, "position": "outer" }
In pie, displayed on the outside by default. labelLine.length controls the label's distance from the pie. In funnel, displayed on the right by default. labelLine.length controls the label's distance from the funnel. In map, the location of labels cannot be specified. position can be specified in line, column, candlestick, scatter. See below.

3.31.177 chart.series.i[].nodes.itemStyle.normal.label.show (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: boolean
Default Value: true
Specifies whether to show label. Valid values are: true (show) | false (hide).

3.31.178 chart.series.i[].nodes.itemStyle.normal.label.rotate (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: boolean
Default Value: false
applicable to chord. Label rotates automatically.

3.31.179 chart.series.i[].nodes.itemStyle.normal.label.distance (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: number
Default Value: 10
In chord, Label's distance from chord after it rotates automatically. In pie, this distance means the factor of label distance and pie radius(In ring chart, is sum of inner and outer radius), default to **0.5**.

3.31.180 chart.series.i[].nodes.itemStyle.normal.label.formatter (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: string, Function
Default Value: null
content formatter: **{string}** (Template) | **{Function}**. Supports asynchronous callback. See below.

- **{string}**, Template, its variables are:
  - {a} | {a0}
  - {b} | {b0}
  - {c} | {c0}
  - {d} | {d0} (some chart types do not have this variable)
  - {e} | {e0} (some chart types do not have this variable)
  - If there are multiple sets of data, there will be multiple sets of varaibles accordingly, like {a1}, {b1}, {c1}, {d1}, {a2}, {b2}, {c2}, {d2}, ...
  - Here are what variable a, b, c and d reprensent in different chart types:
    - **line & area**, **column & bar, candlestick**: a (series name), b(category value), c (value), d(null)
    - **scatter & bubble**: a (series name), b (data name), c (value array), d(null)
    - **map**: a (series name), b (area name), c(merged value), d(null)
    - **pie**, **radar**, **gauge**, **funnel**: a (series name), b (data item name), c(value), d (pie: percent | radar: indicator name)
    - **force, chord** :
      - node : a (series name), b (node name), c (node value), d (node index), e
      - link : a (series name), b (link name, default bigEndNodeName-smallEndNodeName), c(link.value), d(name or index of big end node), e(name or index of small end node)
- **{Function}**, the list of parameters passed in is [params, ticket, callback], detail as follows:
  - params : array content and template varaibles are the same

```
[
    {
```

```
            seriesIndex: 0,
            seriesName: 'Sales',
            name: 'Monday',
            dataIndex: 0,
            data: data,
            name: name,
            value: value,
            percent: special,       // pie
            indicator: special,     // radar, force, chord
            value2: special2,       // force, chord
            indicator2: special2    // force, chord
        },
        {..},
        ...
    ]
```

- ticket: asynchronous callback marker
- callback: asynchronous callback, the callback needs two parameters, the first one is the ticket that we mentioned earlier, the second is the fill content html.
- ** When the function callbacks, 'this' pointer points to the current chart instance (myChart).*

label text formatter, universal, same as Tooltip.formatter, Supports template, method callback. Does not support asynchronous callback.

3.31.181 chart.series.i[].nodes.itemStyle.noraml.label.textStyle (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: Object
Default Value: null
style for label text.

3.31.182 chart.series.i[].nodes.itemStyle.normal.label.textStyle.color (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: color
Default Value: @TODO: 各异
color

3.31.183 chart.series.i[].nodes.itemStyle.normal.label.textStyle.align (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: string
Default Value: @TODO: 各异
horizontal alignment. Valid values are: 'left' | 'right' | 'center'.

3.31.184 chart.series.i[].nodes.itemStyle.normal.label.textStyle.baseline (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: string
Default Value: @TODO: 各异
vertical alignment. Valid values are: 'top' | 'bottom' | 'middle'.

3.31.185 chart.series.i[].nodes.itemStyle.normal.label.textStyle.fontFamily (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font family.

3.31.186 chart.series.i[].nodes.itemStyle.normal.label.textStyle.fontSize (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: number
Default Value: 12
font size, in px.

3.31.187 chart.series.i[].nodes.itemStyle.normal.label.textStyle.fontStyle (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

3.31.188 chart.series.i[].nodes.itemStyle.normal.label.textStyle.fontWeight (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

3.31.189 chart.series.i[].nodes.itemStyle.emphasis (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: Object
emphasis item style (when hovered)

3.31.190 chart.series.i[].nodes.itemStyle.emphasis.color (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: color, Function
Default Value: @TODO: 图表各异
color, dominant color, function paramater is {seriesIndex:x, series:xxx, dataIndex:y, data:yyy}

3.31.191 chart.series.i[].nodes.itemStyle.emphasis.nodeStyle (Noch nicht implementiert)

Type: Object
Default Value: @TODO: 图表各异
node style.

3.31.192 chart.series.i[].nodes.itemStyle.emphasis.nodeStyle.color (Noch nicht implementiert)

Type: color
Default Value: '#f08c2e'
fill color.

3.31.193 chart.series.i[].nodes.itemStyle.emphasis.nodeStyle.borderColor (Noch nicht implementiert)

Type: color
Default Value: '#5182ab'
stroke color.

3.31.194 chart.series.i[].nodes.itemStyle.emphasis.nodeStyle.borderWidth (Noch nicht implementiert)

Type: number
Default Value: 1
width of the stroke line.

3.31.195 chart.series.i[].nodes.itemStyle.emphasis.linkStyle (Noch nicht implementiert)

Type: Object
Default Value: @TODO: 图表各异
link style.

3.31.196 chart.series.i[].nodes.itemStyle.emphasis.linkStyle.normal (Noch nicht implementiert)

Type: Object

3.31.197 chart.series.i[].nodes.itemStyle.emphasis.linkStyle.normal.type (Noch nicht implementiert)

Type: string
Default Value: 'line'
line style, can be: 'curve' | 'line'

3.31.198 chart.series.i[].nodes.itemStyle.emphasis.linkStyle.normal.color (Noch nicht implementiert)

Type: color
Default Value: '#5182ab'
color of the line

Copyright © 2023 TRADUI Technologies GmbH

3.31.199 chart.series.i[].nodes.itemStyle.emphasis.linkStyle.normal.width (Noch nicht implementiert)

Type: number
Default Value: 1
width of the line

3.31.200 chart.series.i[].nodes.itemStyle.emphasis.linkStyle.emphasis (Noch nicht implementiert)

Type: Object

3.31.201 chart.series.i[].nodes.itemStyle.emphasis.linkStyle.emphasis.type (Noch nicht implementiert)

Type: string
Default Value: 'line'
line style, can be: 'curve' | 'line'

3.31.202 chart.series.i[].nodes.itemStyle.emphasis.linkStyle.emphasis.color (Noch nicht implementiert)

Type: color
Default Value: '#5182ab'
color of the line

3.31.203 chart.series.i[].nodes.itemStyle.emphasis.linkStyle.emphasis.width (Noch nicht implementiert)

Type: number
Default Value: 1
width of the line

3.31.204 chart.series.i[].nodes.itemStyle.emphasis.label (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: Object
Default Value: { "show": true, "position": "outer" }
In pie, displayed on the outside by default. labelLine.length controls the label's distance from the pie. In funnel, displayed on the right by default. labelLine.length controls the label's distance from the funnel. In map, the location of labels cannot be specified. position can be specified in line, column, candlestick, scatter. See below.

3.31.205 chart.series.i[].nodes.itemStyle.emphasis.label.show (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: boolean
Default Value: true
Specifies whether to show label. Valid values are: true (show) | false (hide).

3.31.206 chart.series.i[].nodes.itemStyle.emphasis.label.rotate (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: boolean
Default Value: false
applicable to chord. Label rotates automatically.

3.31.207 chart.series.i[].nodes.itemStyle.emphasis.label.distance (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: number
Default Value: 10
In chord, Label's distance from chord after it rotates automatically. In pie, this distance means the factor of label distance and pie radius(In ring chart, is sum of inner and outer radius), default to **0.5**.

3.31.208 chart.series.i[].nodes.itemStyle.emphasis.label.formatter (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: string, Function
Default Value: null
content formatter: **{string}** (Template) | **{Function}**. Supports asynchronous callback. See below.

- **{string}**, Template, its variables are:
  - {a} | {a0}
  - {b} | {b0}
  - {c} | {c0}
  - {d} | {d0} (some chart types do not have this variable)
  - {e} | {e0} (some chart types do not have this variable)
  - If there are multiple sets of data, there will be multiple sets of varaibles accordingly, like {a1}, {b1}, {c1}, {d1}, {a2}, {b2}, {c2}, {d2}, ...
  - Here are what variable a, b, c and d reprensent in different chart types:
    - **line & area**, **column & bar, candlestick**: a (series name), b(category value), c (value), d(null)
    - **scatter & bubble**: a (series name), b (data name), c (value array), d(null)
    - **map**: a (series name), b (area name), c(merged value), d(null)
    - **pie**, **radar**, **gauge**, **funnel**: a (series name), b (data item name), c(value), d (pie: percent | radar: indicator name)
    - **force, chord** :
      - node : a (series name), b (node name), c (node value), d (node index), e
      - link : a (series name), b (link name, default bigEndNodeName-smallEndNodeName), c(link.value), d(name or index of big end node), e(name or index of small end node)
- **{Function}**, the list of parameters passed in is [params, ticket, callback], detail as follows:
  - params : array content and template varaibles are the same

```
[
    {

        seriesIndex: 0,
        seriesName: 'Sales',
        name: 'Monday',
        dataIndex: 0,
        data: data,
        name: name,
        value: value,
        percent: special,       // pie
```

```
            indicator: special,     // radar, force, chord
            value2: special2,       // force, chord
            indicator2: special2    // force, chord
        },
        {..},
        ...
    ]
```

- ticket: asynchronous callback marker
- callback: asynchronous callback, the callback needs two parameters, the first one is the ticket that we mentioned earlier, the second is the fill content html.
- ** When the function callbacks, 'this' pointer points to the current chart instance (myChart).*

label text formatter, universal, same as Tooltip.formatter, Supports template, method callback. Does not support asynchronous callback.

3.31.209 chart.series.i[].nodes.itemStyle.emphasis.label.textStyle (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: Object
Default Value: null
style for label text.

3.31.210 chart.series.i[].nodes.itemStyle.emphasis.label.textStyle.color (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: color
Default Value: @TODO: 各异
color

3.31.211 chart.series.i[].nodes.itemStyle.emphasis.label.textStyle.align (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: string
Default Value: @TODO: 各异
horizontal alignment. Valid values are: 'left' | 'right' | 'center'.

3.31.212 chart.series.i[].nodes.itemStyle.emphasis.label.textStyle.baseline (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: string
Default Value: @TODO: 各异
vertical alignment. Valid values are: 'top' | 'bottom' | 'middle'.

3.31.213 chart.series.i[].nodes.itemStyle.emphasis.label.textStyle.fontFamily (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: string
Default Value: 'Arial, Verdana, sans-serif'
font family.

3.31.214 chart.series.i[].nodes.itemStyle.emphasis.label.textStyle.fontSize (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: number
Default Value: 12
font size, in px.

3.31.215 chart.series.i[].nodes.itemStyle.emphasis.label.textStyle.fontStyle (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: string
Default Value: 'normal'
font style. Valid values are: 'normal' | 'italic' | 'oblique'.

3.31.216 chart.series.i[].nodes.itemStyle.emphasis.label.textStyle.fontWeight (Noch nicht implementiert, auch bei chord (noch nicht implementiert))

Type: string, number
Default Value: 'normal'
font weight. Valid values are: 'normal' | 'bold' | 'bolder' | 'lighter' | 100 | 200 |... | 900.

3.31.217 chart.series.i[].nodes.initial (Noch nicht implementiert)

Type: Array
Forces to specify the initial value of the node. When not specified, it will be randomly generated within a certain range.

3.31.218 chart.series.i[].nodes.fixX (Noch nicht implementiert)

Type: boolean
Default Value: false
Specifies whether to fix the X-value of the node. Works with initial.

3.31.219 chart.series.i[].nodes.fixY (Noch nicht implementiert)

Type: boolean
Default Value: false
Specifies whether to fix the Y-value of the node. Works with initial.

3.31.220 chart.series.i[].nodes.draggable (Noch nicht implementiert)

Type: boolean
Specifies whether the node is draggable.

3.32 Series- Alle Charttypen

3.32.1 chart.series.i[].type

Type: string
chart type. Required parameter. If null or unsupported type, the data of this series will not be shown. Valid values are: 'line' | 'bar' | 'scatter' | 'candlestick' | 'pie' | 'radar' | 'chord' | 'force' | 'map'

3.32.2 chart.series.i[].name

Type: string
Default Value: null
name of series. if the legend is enabled, the value will be indexed to legend.data.

3.32.3 chart.series.i[].clickable

Type: boolean
Default Value: true
Specifies data graphic clickable or not, default to true, recommend to false when you do not have a click event handler.

3.32.4 chart.series.i[].legendHoverLink

Type: boolean
Default Value: true
Enables legend hover link to the chart.

3.32.5 chart.series.i[].x

In Original Doku nicht vorhanden

3.32.6 chart.series.i[].x2

In Original Doku nicht vorhanden

3.32.7 chart.series.i[].y

In Original Doku nicht vorhanden

3.32.8 chart.series.i[].y2

In Original Doku nicht vorhanden

3.32.9 chart.series.i[].z

Type: number
Default Value: 2
The second grade cascading control, In the same canvas (zlevel), the higher z the closer to the top.

Copyright © 2023 TRADUI Technologies GmbH

3.32.10 chart.series.i[].zlevel

Type: number
Default Value 0
The first grade cascading control. Every zlevel will be allocated to a independent canvas, charts and components with the same zlevel will be rendered in the same canvas. The higher zlevel the closer to the top. More canvas dom will need more memory and performance, so never set too many zlevel. Most of the time, the second grade cascading control 'z' is recommended.

3.33 Series-Line/Area

3.33.1 chart.series.i[].stack (Auch bei bar)

Type: string
Default Value: null
name of the stack, USELESS in irregular line|bar. It is used in stacked charts with multiple groups of data. For example, stack:'group1' means to stack the values of the data whose stack value is 'group1' in the series array.

3.33.2 chart.series.i[].xAxisIndex (Auch bei bar, scatter, eventRiver)

Type: number
Default Value: 0
xAxis the index of the axis array. Specifies the horizontal axis that the data series uses.

3.33.3 chart.series.i[].yAxisIndex (Auch bei bar, scatter)

Type: number
Default Value: 0
yAxis the index of the axis array. Specifies the vertical axis that the data series uses.

3.33.4 chart.series.i[].symbol (Auch bei scatter, radar, chord, force, tree)

Type: string
Default Value: null
A predefined shape or symbol for the marker. The default symbol is automatically selected from (When all symbols are used, new symbols are pulled from the start again. Set it to'none' if you don't want to show the symbol): 'circle' | 'rectangle' | 'triangle' | 'diamond' | 'emptyCircle' | 'emptyRectangle' | 'emptyTriangle' | 'emptyDiamond' Other possible values are 'heart', 'droplet', 'pin', 'arrow' and 'star'. Although They are not included in the regular eight symbols above, you can use them at both series and data level. In addition, you can use 'star' + n (n>=3) to get N-pointed star, like 'star6' for six-pointed star. Since 1.3.5, ECharts added support for setting custom image as symbol. The URL to an image can be given on this form:'image://' + 'URL', such as 'image://../asset/ico/favicon.png'. See example 》

3.33.5 chart.series.i[].symbolSize (Auch bei scatter, radar, chord, force, tree)

Type: number, Function, Array
Default Value: 2 | 4
size of the symbol. It is usually a number. But if you use an image as symbol. To prevent stretching of image caused by same width and height, you can use an array as symbolSize which first element is width and the second is height. When "calculable" is enabled, it's recommended to increase symbolSize for better interactive experience. When implementing a bubble chart, symbolSize should be a Function, and the bubble size depends on the return value of the method, the parameter passed in is the current data item (value array).

3.33.6 chart.series.i[].symbolRotate (Auch bei scatter, radar)

Type: number
Default Value: null
the angle by which the symbol is rotated. Valid range is: [-180,180]

3.33.7 chart.series.i[].showAllSymbol

Type: boolean
Default Value: false
By default, a symbol will show only when its corresponding axis label does. set it to true if you need to show all the symbols.

3.33.8 chart.series.i[].smooth

Type: boolean
Default Value: false
smoothed line, while smooth is true, lineStyle.type can not be dashed.

3.33.9 chart.series.i[].dataFilter

Type: boolean
Default Value: 'nearst'
ECharts will optimize for the situation when data number is much larger than viewport width. It will filter the data showed in one pixel width. And this option is for data filtering strategy. Options: 'nearest', 'min', 'max', 'average'.

3.34 Series-Map

3.34.1 chart.series.i[].selectedMode (Noch nicht implementiert, auch bei pie)

Type: boolean, string
Default Value: null
selected mode. Defaults to false. Can be one of single, multiple.

3.34.2 chart.series.i[].mapType (Noch nicht implementiert)

Type: string
Default Value: 'china'
@TODO: map type, including World Map, China Map, Map of China Provinces & Cities. For the mapType of China Provinces & Cities, just use simplified Chinese. 新疆, 西藏, 内蒙古, 青海, 四川, 黑龙江, 甘肃, 云南, 广西, 湖南, 陕西, 广东, 吉林, 河北, 湖北, 贵州, 山东, 江西, 河南, 辽宁, 山西, 安徽, 福建, 浙江, 江苏, 重庆, 宁夏, 海南, 台湾, 北京, 天津, 上海, 香港, 澳门' Supports sub-region mode, which develops sub-region maps from the main map types. The format is 'main map type | sub-region name', like 'world|Brazil', 'china|广东'. See example 》

3.34.3 chart.series.i[].hoverable (Noch nicht implementiert)

Type: boolean
Default Value: true
Non-numerical display (eg: only used to display markPoints and markLines). Set it to false to turn off regional hover highlight.

3.34.4 chart.series.i[].dataRangeHoverLink

Type: boolean
Default Value: true
enables dataRange hover link to the chart

3.34.5 chart.series.i[].mapLocation (Noch nicht implementiert)

Type: Object
Default Value: { "x": "center", "y": "center" }
Map location. x, y, width, height are configurable. Valid values for x and y are 'top', 'bottom', 'left', 'right', 'center', or absolute pixel value. Width and height can be set in pixel values. If any parameter is null, it will be adaptive in accordance with other parameters.

3.34.6 chart.series.i[].mapValueCalculation (Noch nicht implementiert)

Type: string
Default Value: 'sum'
Calculation of map value. Defaults to sum. Valid values are: 'sum' | 'average'

3.34.7 chart.series.i[].mapValuePrecision (Noch nicht implementiert)

Type: number
Default Value: 0
Decimal precision of map value calculation. Valid when mapValueCalculation is set to average. Defaults to rounding.If decimal precision is needed, set an integer greater than 0.

3.34.8 chart.series.i[].showLegendSymbol (Noch nicht implementiert)

Type: boolean
Default Value: true
Color of show legend symbol (small dots symbolizing different series). Valid when there is a legend.

3.34.9 chart.series.i[].roam

Type: boolean, string
Default Value: fasle
Specifies whether the wheel zoom and drag roam will be enabled. Default to false (boolean), can be true (boolean) and 'scale' (string) for just enable the zoom(in/out), 'move' (string) for just enable the move.

3.34.10 chart.series.i[].scaleLimit (Noch nicht implementiert)

Type: Object
Default Value: null
Controls the limit of wheel zoom. You can specify {max:number, min:number}, max is magnification coefficient, whose valid value should be greater than 1; min is reduction coefficient, whose valid value should be less than 1.

3.34.11 chart.series.i[].nameMap (Noch nicht implementiert)

Type: Object
Default Value. null
name mapping of the custom region, like {'China' : '中国'}.

Copyright © 2023 TRADUI Technologies GmbH

3.34.12 chart.series.i[].textFixed (Noch nicht implementiert)

Type: Object
Default Value: null
fixed location of the name text of a region in px. When the value is positive, move to the lower left; When the value is negative, move to the upper right. Such as {'China' : [10, -10]}.

3.34.13 chart.series.i[].geoCoord (Noch nicht implementiert)

Type: Object
Default Value: null
Specifies the location of the name text of a region by absolute latitude and longitude. For example, {'Islands': [113.95, 22.26]}: the name of Hong Kong's Islands District is located at 22.26°N and 113.95°E.

3.34.14 chart.series.i[].heatmap.blurSize (Noch nicht implementiert)

Type: number
Default Value: 30
Blur size of a data point.

3.34.15 chart.series.i[].heatmap.gradientColors (Noch nicht implementiert)

Type: Array
Default Value: ["blue", "cyan", "lime", "yellow", "red"]
It could be an array of offset and color like [{ offset: 0.2, color: 'blue' }, { offset 0.8, color: 'cyan' }] or an array of color strings like ['blue', 'cyan', 'lime', 'yellow', 'red'], with which the color will transform evenly.

3.34.16 chart.series.i[].heatmap.minAlpha (Noch nicht implementiert)

Type: number
Default Value: 0.05
If the unified value is less than minAlpha, it will be set to minAlpha to ensure small data value can also be visible on the chart.

3.34.17 chart.series.i[].heatmap.valueScale (Noch nicht implementiert)

Type: number
Default Value: 1
All data values will multiply this value.

3.34.18 chart.series.i[].heatmap.opacity (Noch nicht implementiert)

Type: number
Default Value: 1
Opacity of the heatmap.

3.34.19 chart.series.i[].heatmap.data (Noch nicht implementiert)

Type: Array
2-dimension array, like [100, 80, 0.67]. The first two items indicate location, and the third item indicates heat, which is between 0 and 1.

Copyright © 2023 TRADUI Technologies GmbH

3.35 Series-Tree

3.35.1 chart.series.i[].symbol (Auch bei line/area, scatter, radar, chord, force)

Type: string
Default Value: null
A predefined shape or symbol for the marker. The default symbol is automatically selected from (When all symbols are used, new symbols are pulled from the start again. Set it to'none' if you don't want to show the symbol): 'circle' | 'rectangle' | 'triangle' | 'diamond' | 'emptyCircle' | 'emptyRectangle' | 'emptyTriangle' | 'emptyDiamond' Other possible values are 'heart', 'droplet', 'pin', 'arrow' and 'star'. Although They are not included in the regular eight symbols above, you can use them at both series and data level. In addition, you can use 'star' + n (n>=3) to get N-pointed star, like 'star6' for six-pointed star. Since 1.3.5, ECharts added support for setting custom image as symbol. The URL to an image can be given on this form:'image://' + 'URL', such as 'image://../asset/ico/favicon.png'. See example 》

3.35.2 chart.series.i[].symbolSize (Auch bei line/area, scatter, radar, chord, force)

Type: number, Array
Default Value: 20
size of nodes

3.35.3 chart.series.i[].roam

Type: boolean, string
Default Value: false
Specifies whether the wheel zoom and drag roam will be enabled. Default to false (boolean), can be true (boolean) and 'scale' (string) for just enable the zoom(in/out), 'move' (string) for just enable the move.

3.35.4 chart.series.i[].rootLocation (Noch nicht implementiert)

Type: Object
Default Value: @TODO: 各异
coordinate of the location of the root, in pixels, percent or keywords. { x: 'center' | 'left' | 'right' | 'x%' | {number}, y: 'center' | 'top' | 'bottom' | 'y%' | {number}}

3.35.5 chart.series.i[].layerPadding

Type: number
Default Value: 100
padding of layers

3.35.6 chart.series.i[].nodePadding (Noch nicht implementiert)

Type: number
Default Value: 30
padding of brothers

3.35.7 chart.series.i[].orient

Type: string
Default Value: 'vertical'
direction of the tree, 'vertical' | 'horizontal' | 'radial'

3.35.8 chart.series.i[].direction

Type: string
Default Value: ' '
to inverse the direction of the tree, can be: 'inverse'

3.36 Series-Funnel

3.36.1 chart.series.i[].sort (Auch bei chord)

Type: string
Default Value: 'descending'
Data sorting, can be one of ascending, descending

3.36.2 chart.series.i[].min (Auch bei gauge)

Type: number
Default Value: 0
The specified minimum value.

3.36.3 chart.series.i[].max (Auch bei gauge)

Type: number
Default Value: 100
The specified maximum value.

3.36.4 chart.series.i[].width

Type: number, string
Default Value: null
The total width, defaults to the drawing area's total width - x - x2, in px, ignore x2 after specifying height. Can also be set in percent (string), such as '50%' (half the width of the display area).

3.36.5 chart.series.i[].height

Type: number, string
Default Value: null
The total height, defaults to the drawing area's total height - y - y2, in px, ignore y2 after specifying height. Can also be set in percent (string), such as '50%' (half the height of the display area).

3.36.6 chart.series.i[].funnelAlign

Type: string
Default Value: 'center'
Horizontal layout style, default to 'center', other option can be: 'left' | 'right' | 'center'

3.36.7 chart.series.i[].minSize

Type: string
Default Value: '0%'

Copyright © 2023 TRADUI Technologies GmbH

The proportion of the minimum value 'min' to the total width. If the symbol of required min is not sharp triangle, you can set minSize to achieve it.

3.36.8 chart.series.i[].maxSize

Type: string
Default Value: '100%'
The proportion of the maximum value 'max' to the total width.

3.36.9 chart.series.i[].gap

Type: number
Default Value: 0
The distance between data symbols.

3.37 Series-Scatter

3.37.1 chart.series.i[].xAxisIndex (Auch bei line/area, bar, eventRiver)

Type: number
Default VAlue: 0
xAxis the index of the axis array. Specifies the horizontal axis that the data series uses.

3.37.2 chart.series.i[].yAxisIndex (Auch bei line/area, bar)

Type: number
Default Value: 0
yAxis the index of the axis array. Specifies the vertical axis that the data series uses.

3.37.3 chart.series.i[].symbol (Auch bei line/area, radar, chord, force, tree)

Type: string
Default Value: null
A predefined shape or symbol for the marker. The default symbol is automatically selected from (When all symbols are used, new symbols are pulled from the start again. Set it to'none' if you don't want to show the symbol): 'circle' | 'rectangle' | 'triangle' | 'diamond' | 'emptyCircle' | 'emptyRectangle' | 'emptyTriangle' | 'emptyDiamond' Other possible values are 'heart', 'droplet', 'pin', 'arrow' and 'star'. Although They are not included in the regular eight symbols above, you can use them at both series and data level. In addition, you can use 'star' + n (n>=3) to get N-pointed star, like 'star6' for six-pointed star. Since 1.3.5, ECharts added support for setting custom image as symbol. The URL to an image can be given on this form:'image://' + 'URL', such as 'image://../asset/ico/favicon.png'. See example 》

3.37.4 chart.series.i[].symbolSize (Auch bei line/area, radar, chord, force, tree)

Type: number, Function, Array
Default Value: 2 | 4
size of the symbol. It is usually a number. But if you use an image as symbol. To prevent stretching of image caused by same width and height, you can use an array as symbolSize which first element is width and the second is height. When "calculable" is enabled, it's recommended to increase symbolSize for better interactive experience. When implementing a bubble chart, symbolSize should be a Function, and the bubble size depends on the return value of the method, the parameter passed in is the current data item (value array).

3.37.5 chart.series.i[].symbolRotate (Auch bei line/area, radar)

Type: number
Default Value: null
the angle by which the symbol is rotated. Valid range is: [-180,180]

3.37.6 chart.series.i[].largeThreshold

Type: number
Default Value: 2000
the threshold of large scale scatter anto-switch. Valid when set to true.

3.38 Series-Pie

3.38.1 chart.series.i[].radius (Auch bei gauge)

Type: number, Array
Default Value: [0, "75%"]
radius, in pixels or in percent. The formula for calculating percent: min(width, height) / 2 * 75%. If the array [inner radius, outer radius] is passed in, it wil become a doughnut.

3.38.2 chart.series.i[].startAngle (Auch bei gauge)

Type: number
Default Value: 90
start angle, pie (90), gauge (225). Valid range is [-360,360].

3.38.3 chart.series.i[].minAngle

Type: number
Default Value: 0
The minimum angle for the pie. Used to prevent tiny item values from affecting interaction.

3.38.4 chart.series.i[].roseType

Type: string
Default Value: null
>Nightingale's rose diagram. 'radius' | 'area'

3.38.5 chart.series.i[].selectedOffset

Type: number
Default Value: 10
Offset of the sector when selected.

3.38.6 chart.series.i[].selectedMode (Auch bei map (noch nicht implementiert))

Type: boolean, string
Default Value: null
selected mode. Defaults to false. Can be one of single, multiple.

Copyright © 2023 TRADUI Technologies GmbH

3.39 Series-Radar

3.39.1 chart.series.i[].symbol (Auch bei line/area, scatter, chord, force, tree)

Type: string
Default Value: null
A predefined shape or symbol for the marker. The default symbol is automatically selected from (When all symbols are used, new symbols are pulled from the start again. Set it to'none' if you don't want to show the symbol): 'circle' | 'rectangle' | 'triangle' | 'diamond' | 'emptyCircle' | 'emptyRectangle' | 'emptyTriangle' | 'emptyDiamond' Other possible values are 'heart', 'droplet', 'pin', 'arrow' and 'star'. Although They are not included in the regular eight symbols above, you can use them at both series and data level. In addition, you can use 'star' + n (n>=3) to get N-pointed star, like 'star6' for six-pointed star. Since 1.3.5, ECharts added support for setting custom image as symbol. The URL to an image can be given on this form:'image://' + 'URL', such as 'image://../asset/ico/favicon.png'. See example 》

3.39.2 chart.series.i[].symbolSize (Auch bei line/area, scatter, chord, force, tree)

Type: number, Function, Array
Default Value: 2 | 4
size of the symbol. It is usually a number. But if you use an image as symbol. To prevent stretching of image caused by same width and height, you can use an array as symbolSize which first element is width and the second is height. When "calculable" is enabled, it's recommended to increase symbolSize for better interactive experience. When implementing a bubble chart, symbolSize should be a Function, and the bubble size depends on the return value of the method, the parameter passed in is the current data item (value array).

3.39.3 chart.series.i[].symbolRotate (Auch bei line/area, scatter)

Type: number
Default Value: null
the angle by which the symbol is rotated. Valid range is: [-180,180]

3.39.4 chart.series.i[].polarIndex

Type: number
Default Value: 0
polar coordinates index.

3.40 Series-Treemap

3.40.1 chart.series.i[].size (Auch bei force, worldCloud)

Type: Array
Default Value: ["80%", "80%"]
size of the chart, in pixels or percent.

3.40.2 chart.series.i[].root (Noch nicht implementiert)

Type: string
Default Value: ' '
Name of root node

3.41 Series-EventRiver

3.41.1 chart.series.i[].xAxisIndex

Type: number
Default Value: 0
xAxis the index of the axis array. Specifies the horizontal axis that the data series uses.

3.41.2 chart.series.i[].weight

Type: number
Default Value: 1
Specifies the weight of this event's series.

Copyright © 2023 TRADUI Technologies GmbH

4 Dashlet-JavaScript-API

zu Die Dashlet-JavaScript-API dient der Kommunikation der Dashlets untereinander sowie zur Kommunikation mit der Applikation, die die Dashlets verwendet. Es wird deshalb unterschieden in

- Dashlet-Application-API
- Dashlet-Communication-API

4.1 Die Dashlet-Application-API

Die Dashlet-Application-API beschreibt verschiedene JavaScript-Funktion, die von der Web-Applikation die die Dashlets darstellt unterstützt werden müssen. Unter dashBIRD auf servBIRD sind diese Funktionen bereits implementiert.

Im folgenden werden die einzelnen Funktionen beschrieben:

4.1.1 showDashletDocument(params)

Diese Funktion dient dem Aufruf anderer Dashlet-Dokumente. Dashlet-Dokumente sind immer verschiede Dashlets, die als eine HTML-Seite gerendert wurden. Deshalb wird hier auch ein Dokumentname als Parameter erwartet. Die Dokumentendung .html kann dabei weggelassen werden.

```
showDashletDocument([{name: 'dashletdocument', value: 'Dokumentname.html'}])
```

4.1.2 runDashletReport(params)

Diese Funktion führt einen Dashletreport aus und kann optional ein bestimmtes Dashlet-Dokument zur Anzeige aufrufen. Dazu müssen der Funktion weitere Parameter übergeben werden, damit die Applikation weiß, welcher Report in welcher Report-Gruppe gemeint ist. Der Syntax sieht dann wie im folgenden beschrieben aus:

```
runDashletReport([{name: 'dashletgroup', value: 'Gruppenname'},
                  {name: 'maindashletreporttitle', value: 'ReportTitleProperty'},
                  {name: 'dashletreportfile', value: 'dateiname.rptdesign'},
                  {name: 'dashletdocument', value: 'Dokumentname.html'}, // optional
(wenn das Default-Dokument des Reports angezeigt werden soll)
                  {name: 'dashletparameters', value: '{"parameter1": "value1",
                                                       "parameter2": "value2",
                                                       "parameter3": "value3"}'}]) //
optional (Parameter, mit denen der Dashlet-Report erzeugt werden soll)
```

4.1.3 refreshDashletReport(params)

Weist die Applikation an den aktuellen Dashlet-Report mit Daten zu aktualisieren. Gegebenfalls sind geänderte bzw. neue Parameter für den Report zu übergeben.

```
refreshDashletReport([{name: 'dashletparameters', value: '{"parameter1": "value1",
                                                           "parameter2": "value2",
                                                           "parameter3": "value3"}'}]
) // optional (Parameter, mit denen der Dashlet-Report erzeugt werden soll)
```

Copyright © 2023 TRADUI Technologies GmbH

4.1.4 runAndExportReport(params)

Mit der Funktion soll ein AdHoc-Report ausgeführt und ein Ausgabe-Dokument als Download bereitgestellt werden können.

```
runAndExportReport([{name: 'reportfolder', value: '/categorypath'},
                    {name: 'reportname', value: '/adhoc-report.rptdesign'},
                    {name: 'outputformat', value: 'PDF'},
                    {name: 'reportparameters', value: '{"parameter1": "value1",
                                                        "parameter2": "value2",
                                                        "parameter3": "value3"}' }]) /
/ optional (Parameter, mit denen der AdHoc-Report erzeugt werden soll)
```

4.1.5 downloadNativeDocument(params)

Mit der Funktion soll, wenn native Dokumente vorhanden, diese zum Download bereitgestellt werden.

```
downloadNativeDocument([{name: 'filename', value: 'NATIVE_EXCEL_FILE.xlsx'},
                    {name: 'reportjobid', value: '99'}])
```

4.1.6 setDashletVisibility(params)

Dient dazu ein Dashlet des aktuellen Dokuments ein- bzw auszublenden. Die Applikation sollte sich den Zustand über das Flag am DashletFragment (DashletReader-Projekt) merken.

```
setDashletVisibility([{name: 'dashletid', value: 'Dashlet-Id'},{name:
'dashletvisibility', value: true|false}]);
```

4.1.7 setDashletBoxCollapse(params)

Dient dazu die Anzeige-Box, sofern unterstützt, eines Dashlets auf- oder zuzuklappen. Die Applikation sollte sich den Zustand über das Flag am DashletFragment (DashletReader-Projekt) merken.

```
setDashletBoxCollapse([{name: 'dashletid', value: 'Dashlet-Id'}]);
```

4.1.8 changeDashletThemeSet(params)

Dient dazu ein Theme oder mehrere Themes verschiedener JavaScript- und CSS-Frameworks umzuschalten. Die Applikation sollte sich merken, welche Themes aktuell verwendet werden..

```
changeDashletThemeSet([{name: 'dashletc3theme', value: 'infographic'},
                       {name: 'dashletechartstheme', value: 'infographic'}]);
```

Hier folgt eine Aufstellung der derzeit unterstützten Themes verschiedener Frameworks.

Copyright © 2023 TRADUI Technologies GmbH

Framework	Theme-Key	mögliche Theme-Values

C3-Charts	`dashletc3theme`	`infographic`
		`shine`
		`sakura`
		`roma`
		`dark`
		`mint`
		`macarons2`
		`macarons`
		`helianthus`
		`green`
		`gray`
		`red`
		`blue`

ECharts	`dashletechartstheme`	`infographic`
		`shine`
		`sakura`
		`roma`
		`dark`
		`mint`
		`macarons2`
		`macarons`
		`helianthus`
		`green`
		`gray`
		`red`
		`blue`

Copyright © 2023 TRADUI Technologies GmbH

Framework	Theme-Key	mögliche Theme-Values

Bootswatch for Bootstrap	`dashletbootstraptheme`	`bootstrap`
		`bootstrap_bootswatch_cerulan`
		`bootstrap_bootswatch_cosmo`
		`bootstrap_bootswatch_cyborg`
		`bootstrap_bootswatch_darkly`
		`bootstrap_bootswatch_journal`
		`bootstrap_bootswatch_lumen`
		`bootstrap_bootswatch_paper`
		`bootstrap_bootswatch_readable`
		`bootstrap_bootswatch_sandstone`
		`bootstrap_bootswatch_simplex`
		`bootstrap_bootswatch_slate`
		`bootstrap_bootswatch_spacelab`
		`bootstrap_bootswatch_superhero`
		`bootstrap_bootswatch_united`
		`bootstrap_bootswatch_yeti`
		`bootstrap_bootswatch_flatly`

Copyright © 2023 TRADUI Technologies GmbH

Framework	Theme-Key	mögliche Theme-Values

Tradui-Dashlets	`dashletscolortheme`	`dashlets.color.cerulan`
		`dashlets.color.cosmo`
		`dashlets.color.cyborg`
		`dashlets.color.darkly`
		`dashlets.color.journal`
		`dashlets.color.lumen`
		`dashlets.color.paper`
		`dashlets.color.readable`
		`dashlets.color.sandstone`
		`dashlets.color.simplex`
		`dashlets.color.slate`
		`dashlets.color.spacelab`
		`dashlets.color.superhero`
		`dashlets.color.united`
		`dashlets.color.yeti`
		`dashlets.color.flatly`

Copyright © 2023 TRADUI Technologies GmbH

5 Dashlet-DataTables

Inhalt

5.1 DataTables-AutoFill

5.1.1 table.autoFill.alwaysAsk (Noch nicht implementiert)

Type: boolean
Default Value. false
Description: When an auto fill is completed by the end user, AutoFill will determine how many data fill plug-ins can be used for the data that has been dragged over. If there are multiple options it will display a list of those options to the end user allowing them to select the action to perform - a cancel button is also shown. By default, if there is only a single data fill plug-in that can operate on the data, it will be immediately acted upon, no confirmation required by the end user and no option to cancel the action. This option can be used to force AutoFill to confirm the action to take, even if there is only one option. This provides the end user with the ability to cancel the action if they made a mistake during the drag.
Always ask the end user if an action should be taken or not

5.1.2 table.autoFill.columns (Noch nicht implementiert)

Type: colnumn-selector
Default Value: ' '
Description: When working with complex tables it is not uncommon to have certain columns that you might not wish to allow a data fill on - for example a checkbox column that is used for row selection or a column with editing action buttons. This option can be used to define which columns can be auto filled. If a column cannot be auto filled, cells in it will not show the fill drag handle and will not be selectable during a drag. Please note that only visible cells can be auto filled, so the columns defined by this option are automatically merged with the visible columns in the table. Non-visible columns will cannot be auto filled.
Select the columns that can be auto filled

5.1.3 table.autoFill.editor (Noch nicht implementiert)

Type: DataTable.editor
Default Value: null
Description: When completing a AutoFill action you will typically wish to update the data source to reflect the changes made by the end user. This can be done with AutoFill's integration with Editor. This option can specify an Editor instance that should be used to submit the changes to the server. Upon drag completion the changes will be immediately submitted to the server and the table redrawn with the updated data retrieved from the server.
Attach an Editor instance for database updating

5.1.4 table.autoFill.enable (Noch nicht implementiert)

Type: boolean
Default Value: true
Description: It can be useful to disable AutoFill's user input controls at certain times, depending on the state of your application. This option provides that ability when the table is initially created, while the `autoFill().enable()` and `autoFill().disable()` methods provide the option to enabling the user interaction after the table has been created.
Initial enablement state of AutoFill

5.1.5 table.autoFill.focus (Noch nicht implementiert)

Type: string
Default Value: null
Description: The click-to-drag auto fill drag handle can be attached to the table cells using a number of different methods, depending on how you wish the end user to interact with your table. This option provides the ability to switch between the various options which are detailed below.
Action that will cause the auto fill drag handle to appear in a cell

5.1.6 table.autoFill.update (Noch nicht implementiert)

Type: boolean
Default Value: true
Description: At the end of a fill action you will typically wish to take some action to reflect the change from the data fill. By default AutoFill will write the filled values to the DataTable and then redraw the table to account for any changes in ordering. This action is not always desirable, particularly if you are using server-side processing or which to have an external process update the data. In such circumstances this option can be used to disable the automatic data update and draw.
Control automatic of data when a fill drag is completed

5.1.7 table.autoFill (Noch nicht implementiert)

Type: boolean, object
Default Value: undefined
Description: This option provides the ability to enable and configure AutoFill for DataTables. In its simplest form as the boolean `true` it will enable AutoFill with the default configuration options (as defined by `$.fn.dataTable.AutoFill.defaults`). It can also be used as an object to provide custom configuration options as described below. Please note that as with all other configuration options for AutoFill, this option is an extension to the default set of DataTables options. This property should be set in the DataTables initialisation object.
Enable and configure the AutoFill extension for DataTables

5.1.8 table.language.autoFill.button (Noch nicht implementiert)

Type: string
Default Value: >
Description: This option gives the ability to customise the text shown in the multi-fill selector buttons. The end user will click these buttons to select what fill type to use when multiple fills are available.
Multi-fill selector button text

5.1.9 table.language.autoFill.cancel (Noch nicht implementiert)

Type: string
Default Value: cancel
Description: If the multi-fill selector is shown it will always contain a cancel option that the end user can select to have no update performed on the table. This option provides the ability to customise the cancel message shown in the selector.
Multi-fill selector cancel option message

5.1.10 table.language.autoFill.fill (Noch nicht implementiert)

Type: string
Default Value: `Fill all cells with >i<{data}>/i<`
Description: The full fill type will copy the data from the cell where the fill originated to all other cells. This option

sets the message that will be shown when this fill type is available (which is always!).
Multi-fill selector message for the *full fill* fill type

5.1.11 table.language.autoFill.fillHorizontal (Noch nicht implementiert)

Type: string
Default Value: `Fill cells horizontally`
Description: The horizontal fill type will copy the data from the first column in the selected cells to the other cells horizontally. This option sets the message that will be shown when this fill type is available.
Multi-fill selector message for the *horizontal fill* fill type

5.1.12 table.language.autoFill.fillVertical (Noch nicht implementiert)

Type: string
Default Value: `Fill cells vertically`
Description: The vertical fill type will copy the data from the first row in the selected cells to the other cells vertically. This option sets the message that will be shown when this fill type is available.
Multi-fill selector message for the *vertical fill* fill type

5.1.13 table.language.autoFill.increment (Noch nicht implementiert)

Type: string
Default Value: `Increment / decrement each cell by: >input type='number' value='1'<`
Description: This option provides the ability to customise the message that the increment fill type will show when it is available for a data fill.
Multi-fill selector message for the *increment* fill type

5.1.14 table.language.autoFill.info (Noch nicht implementiert)

Type: string
Default Value: `''`
AutoFill's fill selector box (shown when there are multiple fill options available) has the ability to show an information message at the top of the box. This can be used to explain to the end user why the box is being shown and what action they should take.
Information message shown at the top of the fill type selector

5.1.15 table.language.autoFill (Noch nicht implementiert)

Type: object
Description: AutoFill make use of a number of strings that are displayed to the end user to explain what action will be performed based on the user input. A parameters of this object provide the ability to customise those strings to suit your needs. This can be a full translation or a tweak of just one or two strings to match the styling of your site / app.
Container object for language strings used by AutoFill

5.2 DataTables-Buttons

5.2.1 table.buttons.buttons.action (Noch nicht implementiert)

Type: function
Default Value: `Default action depends upon the button type. Please refer to the button type documentation`

Description: This function defined the action that the button will take when activated by the end user. This will normally be to perform some operation on the DataTable, but can be absolutely anything since the function can be defined by yourself.
Action to take when the button is activated

5.2.2 table.buttons.buttons.available (Noch nicht implementiert)

Type: function
Default Value: ''
Description: A number of buttons types, particularly the file export buttons, can depend upon certain APIs being available in the web-browser and external third-party libraries having been loaded. For example the `pdfHtml5` button type depends upon the `FileReader` API and also the `makePdf` library, while the `pdfFlash` button depends upon Adobe Flash being available. This method provides button developers with the ability to ensure that any dependencies their buttons might have are satisfied before the button is shown to the end user. If the function returns `false`, the button is simply not shown to the end user.
Ensure that any requirements have been satisfied before initialising a button

5.2.3 table.buttons.buttons.className (Noch nicht implementiert)

Type: string
Default Value. undefined
Description: The ability to set a class name for a button can be useful both for styling the button and also for selection and manipulation via the API (`buttons()` for example). This option provides exactly that ability. The class name defined here is added to the class name defined by `buttons.buttons.className`, so the button will likely have two or more class names assigned to it. Please note that most buttons will define their own class name so they will be style-able without being required to add a custom class name using you are extending an existing button. Please refer to the documentation for each button type for their default class name. Additionally, if you extend a button that has its own class name defined, any classes you set will automatically be appended to the extended button's class. It will *not* overwrite the original class.
Set the class name for the button

5.2.4 table.buttons.buttons.destroy (Noch nicht implementiert)

Type: function
Default Value: ''
Description: This function is provided to allow button authors to be able to assign custom events to the host DataTable or the button's DOM node, and then cleanly remove those events when a button is destroyed, ensuring that there are no memory leaks. Of particular interest when using this method and the `buttons.buttons.init` option to attach events, will be the `buttons.buttons.namespace` option (accessible as the `namespace` parameter of the third parameter passed into this function). The `namespace` option is a unique namespacing string for every button, allowing events to be correctly removed without accidentally also removing other events.
Function that is called when the button is destroyed

5.2.5 table.buttons.buttons.enabled (Noch nicht implementiert)

Type: boolean
Default Value: true
Description: This option provides the ability to set the initial enabled state of a button using a boolean value.
Set a button's initial enabled state

5.2.6 table.buttons.buttons.extend (Noch nicht implementiert)

Type: string
Default Value: undefined
Description: When creating buttons you may often wish to customise an existing button type, for example modifying the display text, while still making use of the functionality that the button offers, without rewriting a whole lot of code! This `extends` option provides exactly that ability. The buttons created can extend multiple existing button types - for example a customised button might extend any built in button type, and that customised button might itself be extended during initialisation to set the button text. The `extends` option is not required, however, if it is not provided, the `buttons.buttons.text` and `buttons.buttons.action` options must be defined.
Define which button type the button should be based on

5.2.7 table.buttons.buttons.init (Noch nicht implementiert)

Type: function
Default Value: ''
Description: This function provides the ability to button plug-in authors to run custom code when a button is initialised. This can be useful for attaching event handlers to the host DataTable that will update the button. For example, the buttons provided by the *Select* extension for DataTables make use of this to create buttons types which are only active when there is one or more item in the DataTable selected.
Initialisation function that can be used to add events specific to this button

5.2.8 table.buttons.buttons.key (Noch nicht implementiert)

Type: string, object
Default Value: undefined
Description: Buttons has the built in ability to activate buttons through keyboard key presses and key combinations. This is to aid accessibility and provide complete keyboard navigation of your table. For example, with using Editor, records can be added, edited and deleted without touching the mouse! Key presses are only processed when the document has no element that is actively focused. This means that typing into an `input` element will not accidentally trigger a button's action! Please note that for Adobe Flash based buttons, key activation simply will not work! This is a security feature implemented by Adobe - such buttons *must* be clicked on to activate them.
Define an activation key for a button

5.2.9 table.buttons.buttons.name (Noch nicht implementiert)

Type: string
Default Value: undefined
Description: The `buttons()` and `button()` API methods provide the ability to use a `button-selector` to be able to select one or more buttons in one or more Buttons instances. In turn the button selector provides an option to easily select a button by using a predefined name (thus removing any complexities that might occur if using index selection). This option provides the ability to set that name. Multiple buttons can share the same name, they need not be unique, but please be aware that they will always be selected together if using a name selector.
Set a name for each selection

5.2.10 table.buttons.buttons.namespace (Noch nicht implementiert)

Type: string
Default Value: null
Description: It can often be useful for authors to be able to add events to a button or DataTable when using the `buttons.buttons.init` option to take some action when a trigger is activated on the page. In order to be able to

cleanly remove the attached events and prevent possible memory leaks form these event listeners, Buttons will automatically generate a namespace string for each button. The `buttons.buttons.destroy` option can be used to define a function that will remove the event listeners added. It is possible to define your own namepsace, however it is **strongly** recommended that you *do not* provide a custom value, instead allowing Buttons to assign its own value. There is no benefit in assigning your own namespace! This parameter is documented primarily so you know that it exists and is automatically generated by Buttons.
Unique namespace for every button

5.2.11 table.buttons.buttons.text (Noch nicht implementiert)

Type: string
Default Value: `Dependent on button type`
Description: Being able to let your users know what will happen when they activate a button is obviously fundamentally important to the Buttons extension and this option provides exactly that ability.
The text to show in the button

5.3 DataTables-Callbacks

5.3.1 table.callback.createdRow (Noch nicht implementiert)

Type: Function
Description: This callback is executed when a `TR` element is created (and all `TD` child elements have been inserted), or registered if using a DOM source, allowing manipulation of the `TR` element. This is particularly useful when using deferred rendering (`deferRender`) or server-side processing (`serverSide`) so you can add events, class name information or otherwise format the row when it is created.
Callback for whenever a TR element is created for the table's body.

5.3.2 table.callback.drawCallback (Noch nicht implementiert)

Type: Function
Description: It can be useful to take an action on every draw event of the table - for example you might want to update an external control with the newly displayed data, or with server-side processing is enabled you might want to assign events to the newly created elements. This callback is designed for exactly that purpose and will execute on every draw.
Function that is called every time DataTables performs a draw.

5.3.3 table.callback.footerCallback (Noch nicht implementiert)

Type: Function
Description: Identical to `headerCallback` but for the table footer this function allows you to modify the table footer on every 'draw' event. Note that if the table does not have a `tfoot` element, it this callback will not be fired.
Footer display callback function.

5.3.4 table.callback.formatNumber (Noch nicht implementiert)

Type: Function
Default Value: Shows numbers grouped as thousands with a comma separator
Description: DataTables will display numbers in a few different locations when drawing information about a table, for example in the table's information element and the pagination controls. When working with large numbers it is often useful to format it for readability by separating the thousand units - for example 1 million is rendered as "1,000,000", allowing the user to see at a glance what order of magnitude the number shows. This function allows complete control over how that formatting is performed. By default DataTables will use the character specified in

`language.thousands` (in turn, that, by default, is a comma) as the thousands separator.
Number formatting callback function.

5.3.5 table.callback.headerCallback (Noch nicht implementiert)

Type: Function
Description: This function is called on every 'draw' event (i.e. when a filter, sort or page event is initiated by the end user or the API), and allows you to dynamically modify the header row. This can be used to calculate and display useful information about the table.
Header display callback function.

5.3.6 table.callback.infoCallback (Noch nicht implementiert)

Type: Function
Description: The information element can be used to convey information about the current state of the table. Although the internationalisation options presented by DataTables are quite capable of dealing with most customisations, there may be times where you wish to customise the string further. This callback allows you to do exactly that. Please note that if the `info` option id disable in the initialisation, this callback function is not fired.
Table summary information display callback.

5.3.7 table.callback.initComplete (Noch nicht implementiert)

Type: Function
Description: It can often be useful to know when your table has fully been initialised, data loaded and drawn, particularly when using an `ajax` data source. In such a case, the table will complete its initial run before the data has been loaded (Ajax is asynchronous after all!) so this callback is provided to let you know when the data is fully loaded. Additionally the callback is passed in the JSON data received from the server when Ajax loading data, which can be useful for configuring components connected to your table, for example Editor fields.
Initialisation complete callback.

5.3.8 table.callback.preDrawCallback (Noch nicht implementiert)

Type: Function
Description: The partner of the `drawCallback` callback, this function is called at the very start of each table draw. It can therefore be used to update or clean the display before each draw (for example removing events), and additionally can be used to cancel the draw by returning `false`. Any other return (including `undefined`) results in the full draw occurring.
Pre-draw callback.

5.3.9 table.callback.rowCallback (Noch nicht implementiert)

Type: Function
Description: This callback allows you to 'post process' each row after it have been generated for each table draw, but *before* it is rendered into the document. This means that the contents of the row might not have dimensions (`$().width()` for example) if it is not already in the document. This function might be used for setting the row class name or otherwise manipulating the row's `tr` element (although note that `createdRow` can often be more efficient).
Row draw callback.

5.3.10 table.callback.stateLoadCallback (Noch nicht implementiert)

Type: Function
Description: With this callback you can define where, and how, the state of a table is loaded from. By default DataTables will load from `localStorage` or `sessionStrorage`, but for more permanent storage, you can store state in a server-side database. Prior to DataTables 1.10.13 this method had to act synchronously, i.e. the state would be returned by the function. As of 1.10.13 it is possible to load state asynchronously via Ajax or any other async method and execute the callback function, passing in the loaded state. To maintain backwards compatibility the state can still be returned synchronously. To use the callback method, simply don't return a value from your `stateLoadCallback` function. See below for examples of both use cases. Note that this callback works hand-in-hand with `stateSaveCallback`. This callback loads the state from storage when the table is reloaded while `stateSaveCallback` saves it.
Callback that defines where and how a saved state should be loaded.

5.3.11 table.callback.stateLoaded (Noch nicht implementiert)

Type: Function
Description: Callback that is fired once the state has been loaded (`stateLoadCallback`) and the saved data manipulated (if required - `stateLoadParams`). This callback is useful if you simply wish to know information from the saved state, without getting into the inner workings of where and how the state information has been saved. For example it can be useful for populating custom filter inputs.
State loaded callback.

5.3.12 table.callback.stateLoadParams (Noch nicht implementiert)

Type: Function
Description: Callback which allows modification of the saved state prior to loading that state. This callback is called when the table is loading state from the stored data, but prior to the settings object being modified by the saved state. Note that the `stateLoadCallback` option is used to define where and how to load the state, while this function is used to manipulate the data once it has been retrieved from storage. Further note that for plug-in authors, you should use the `stateLoadParams` event to load parameters for a plug-in.
State loaded - data manipulation callback

5.3.13 table.callback.stateSaveCallback (Noch nicht implementiert)

Type: Function
Description: DataTables can save the state of the table (paging, filtering etc) when the `stateSave` option is enabled, and by default it will use HTML5's `localStorage` to save the state into. This callback method allows you to change where the state is saved (for example you might wish to use a server-side database or cookies). Note that additional extensions can add extra information to this structure, or you may use `stateSaveParams` or `stateSaveParams` to add your own parameters. Also the information stored is type sensitive - that is, the data type of the data given by DataTables must be preserved. For example the `start` parameter must be a `number` data type. This method is required only to store the data given to it. The `stateSaveParams` method is used to manipulate the data that is to actually be saved. This callback works hand-in-hand with `stateLoadCallback`. This method saves the state while `stateSaveCallback` will load it from where this callback has saved it.
Callback that defines how the table state is stored and where.

5.3.14 table.callback.stateSaveParams (Noch nicht implementiert)

Type: Function
Description: Callback which allows modification of the parameters to be saved for the DataTables state saving (`stateSave`), prior to the data actually being saved. This callback is called every time DataTables requests that the state be saved. For the format of the data that is stored, please refer to the `stateSaveCallback` documentation.

Note that the `stateSaveCallback` option is used to define where and how to store the state, while this function is used to manipulate the data before it is entered into storage. Further note that for plug-in authors, you should use the `stateSaveParams` event to add extra parameters to the state storage object if required.
State save - data manipulation callback

5.4 DataTables-ColumnDef

5.5 DataTables-Columns

5.6 DataTables-Feature

5.6.1 table.feature.autoWidth

Type: boolean
Default Value: true
Description: Enable or disable automatic column width calculation. This can be disabled as an optimisation (it takes a finite amount of time to calculate the widths) if the tables widths are passed in using `columns.width`.
Feature control DataTables' smart column width handling

5.6.2 table.feature.deferRender

Type: boolean
Default Value: false
Description: By default, when DataTables loads data from an Ajax or Javascript data source (`ajax` and `data` respectively) it will create all HTML elements needed up-front. When working with large data sets, this operation can take a not-insignificant amount of time, particularly in older browsers such as IE6-8. This option allows DataTables to create the nodes (rows and cells in the table body) only when they are needed for a draw. As an example to help illustrate this, if you load a data set with 10,000 rows, but a paging display length of only 10 records, rather than create all 10,000 rows, when deferred rendering is enabled, DataTables will create only 10. When the end user then sorts, pages or filters the data the rows needed for the next display will be created automatically. This effectively spreads the load of creating the rows across the life time of the page. Note that when enabled, it goes without saying that not all nodes will always be available in the table, so when working with API methods such as `columns().nodes()` you must take this into account. Below shows an example of how to use jQuery delegated events to handle such a situation.
Feature control deferred rendering for additional speed of initialisation.

5.6.3 table.feature.info

Type: boolean
Default Value: true
Description: When this option is enabled, Datatables will show information about the table including information about filtered data if that action is being performed. This option allows that feature to be enabled or disabled. Note that by default the information display is shown below the table on the left, but this can be controlled using `dom` and CSS).
Feature control table information display field

5.6.4 table.feature.jQueryUI

Type: boolean

nicht in Original Doku vorhanden

5.6.5 table.feature.lengthChange

Type: boolean
Default Value: true
Description: When pagination is enabled, this option will control the display of an option for the end user to change number of records to be shown per page. The options shown in the list are controlled by the `lengthMenu` configuration option. Note that by default the control is shown at the top left of the table. That can be controlled using `dom` and CSS. If this option is disabled (`false`) the length change input control is removed - although the `page.len()` method can still be used if you wish to programmatically change the page size and `pageLength` can be used to specify the initial page length. Paging itself is not effected. dditionally, if pagination is disabled using the `paging` option, this option is automatically disabled since it has no relevance when the is no pagination.
Feature control the end user's ability to change the paging display length of the table.

5.6.6 table.feature.ordering

Type: boolean
Default Value: true
Description: Enable or disable ordering of columns - it is as simple as that! DataTables, by default, allows end users to click on the header cell for each column, ordering the table by the data in that column. The ability to order data can be disabled using this option. Note that the ability to add or remove sorting of individual columns can be disabled by the `columns.orderable` option for each column. This parameter is a global option - when disabled, there are no sorting actions applied by DataTables at all.
Feature control ordering (sorting) abilities in DataTables.

5.6.7 table.feature.paging

Type: boolen
Default Value: true
Description: DataTables can split the rows in tables into individual pages, which is an efficient method of showing a large number of records in a small space. The end user is provided with controls to request the display of different data as the navigate through the data. This feature is enabled by default, but if you wish to disable it, you may do so with this parameter.
Enable or disable table pagination.

5.6.8 table.feature.processing

Type: boolean
Default Value: false
Description: Enable or disable the display of a 'processing' indicator when the table is being processed (e.g. a sort). This is particularly useful for tables with large amounts of data where it can take a noticeable amount of time to sort the entries.
Feature control the processing indicator.

5.6.9 table.feature.scrollX

Type: boolean
Default Value: false
Description: Enable horizontal scrolling. When a table is too wide to fit into a certain layout, or you have a large number of columns in the table, you can enable horizontal (x) scrolling to show the table in a viewport, which can be scrolled. This property can be `true` which will allow the table to scroll horizontally when needed (recommended), or any CSS unit, or a number (in which case it will be treated as a pixel measurement).
Horizontal scrolling

5.6.10 table.feature.scrollY

Type: string
Description: Enable vertical scrolling. Vertical scrolling will constrain the DataTable to the given height, and enable scrolling for any data which overflows the current viewport. This can be used as an alternative to paging to display a lot of data in a small area (although paging and scrolling can both be enabled at the same time if desired). The value given here can be any CSS unit, or a number (in which case it will be treated as a pixel measurement) and is applied to the table body (i.e. it does not take into account the header or footer height directly).
Vertical scrolling

5.6.11 table.feature.searching

Type: boolean
Default Value: true
Description: This option allows the search abilities of DataTables to be enabled or disabled. Searching in DataTables is "smart" in that it allows the end user to input multiple words (space separated) and will match a row containing those words, even if not in the order that was specified (this allow matching across multiple columns). Please be aware that technically the search in DataTables is actually a filter, since it is subtractive, removing data from the data set as the input becomes more complex. It is named "search" here, and else where in the DataTables API for consistency and to ensure there are no conflicts with other methods of a similar name (specific the `filter()` API method). Note that if you wish to use the search abilities of DataTables this must remain `true` - to remove the default search input box whilst retaining searching abilities (for example you might use the `search()` method), use the `dom` option.
Feature control search (filtering) abilities

5.6.12 table.feature.serverSide

Type: boolean
Default Value: false
Description: DataTables has two fundamental modes of operation:

- Client-side processing - where filtering, paging and sorting calculations are all performed in the web-browser.
- Server-side processing - where filtering, paging and sorting calculations are all performed by a server.

By default DataTables operates in client-side processing mode, but can be switched to server-side processing mode using this option. Server-side processing is useful when working with large data sets (typically >50'000 records) as it means a database engine can be used to perform the sorting etc calculations - operations that modern database engines are highly optimised for, allowing use of DataTables with massive data sets (millions of rows).

When operating in server-side processing mode, DataTables will send parameters to the server indicating what data it needs (what page, what filters are applied etc), and also expects certain parameters back in order that it has all the information required to display the table. The client-server communication protocol DataTables uses is detailed in the DataTables documentation.
Feature control DataTables' server-side processing mode.

5.6.13 table.feature.stateSave

Type: boolean
Default Value: false
Description: Enable or disable state saving. When enabled aDataTables will store state information such as pagination position, display length, filtering and sorting. When the end user reloads the page the table's state will be altered to match what they had previously set up. Data storage for the state information in the browser is performed by use of the `localStorage` or `sessionStorage` HTML5 APIs. The `stateDuration` indicated to DataTables which API should be used (`localStorage`: 0 or greater, or `sessionStorage`: -1). To be able to uniquely

identify each table's state data, information is stored using a combination of the table's DOM `id` and the current page's pathname. If the table's `id` changes, or the page URL changes, the state information will be lost. Please note that the use of the HTML5 APIs for data storage means that the built in state saving option **will not work with IE6/7** as these browsers do not support these APIs. Alternative options of using cookies or saving the state on the server through Ajax can be used through the `stateSaveCallback` and `stateLoadCallback` options.
State saving - restore table state on page reload

5.7 DataTables-FixedColumns

5.8 DataTables-Internationalisation

5.8.1 table.language.aria.sortAscending

Type: string
Default Value: `: activate to sort column ascending`
Description: ARIA label that is added to the table headers when the column may be sorted ascending by activating the column (click or return when focused). Note that the column header text is prefixed to this string.
Language strings used for WAI-ARIA specific attributes

5.8.2 table.language.aria.sortDescending

Type: string
Default Value: `: activate to sort column descending`
Description: ARIA label that is added to the table headers when the column may be sorted descending by activing the column (click or return when focused). Note that the column header text is prefixed to this string.
Language strings used for WAI-ARIA specific attributes

5.8.3 table.language.aria

Type: string
nicht in Original Doku vorhanden

5.8.4 table.language.decimal

Type: string
Default Value: `''`
Description: A dot (`.`) is used to mark the decimal place in Javascript, however, many parts of the world use a comma (`,`) and other characters such as the Unicode decimal separator (◤) or a dash (–) are often used to show the decimal place in a displayed number. When reading such numbers, Javascript won't automatically recognise them as numbers, however, DataTables' type detection and sorting methods can be instructed through the `language.decimal` option which character is used as the decimal place in your numbers. This will be used to correctly adjust DataTables' type detection and sorting algorithms to sort numbers in your table. This option is a little unusual as DataTables will never display a formatted, floating point number (it has no need to!) so this option only effects how it parses the read data (none of the other language options have this ability). Any character can be set as the decimal place using this option, although the decimal place used in a single table must be consistent (i.e. numbers with a dot decimal place and comma decimal place cannot both appear in the same table as the two types are ambiguous). Different tables on the same page can use different decimal characters if required. When given as an empty string (as this parameter is by default) a dot (`.`) is assumed to be the character used for the decimal place.
Decimal place character

5.8.5 table.language.emptyTable

Type: string
Default Value: `No data available in table`
Description: This string is shown in preference to `language.zeroRecords` when the table is empty of data (regardless of filtering) - i.e. there are zero records in the table. Note that this is an optional parameter. If it is not given, the value of `language.zeroRecords` will be used instead (either the default or given value).
Table has no records string

5.8.6 table.language.info

Type: string
Default Value: `Showing _START_ to _END_ of _TOTAL_ entries`
Description: This string gives information to the end user about the information that is current on display on the page. The following tokens can be used in the string and will be dynamically replaced as the table display updates.
Table summary information display string

5.8.7 table.language.infoEmpty

Type: string
Default Value: `Showing 0 to 0 of 0 entries`
Description: Display information string for when the table is empty. Typically the format of this string should match `info`.
Table summary information string used when the table is empty of records

5.8.8 table.language.infoFiltered

Type: string
Default Value: `(filtered from _MAX_ total entries)`
Description: When a user filters the information in a table, this string is appended to the information (`info`) to give an idea of how strong the filtering is. The token `_MAX_` is dynamically updated - see `language.info` for information about all available tokens.
Appended string to the summary information when the table is filtered

5.8.9 table.language.infoPostFix

Type: string
Default Value: *`Empty string`*
Description: If can be useful to append extra information to the info string at times, and this variable does exactly that. This string will be appended to the `language.info` (`language.infoEmpty` and `language.infoFiltered` in whatever combination they are being used) at all times.
String to append to all other summary information strings

5.8.10 table.language.lengthMenu

Type: string
Default Value: `Show _MENU_ entries`
Description: Detail the action that will be taken when the drop down menu for the pagination length option is changed. The token `_MENU_` is replaced with a default select list of 10, 25, 50 and 100 (or any other value specified by `lengthMenu`), and can be replaced with a custom select list if required.
Page length options string

5.8.11 table.language.loadingRecords

Type: string
Default Value: `Loading...`
Description: When using Ajax sourced data and during the first draw when DataTables is gathering the data, this message is shown in an empty row in the table to indicate to the end user the the data is being loaded. Note that this parameter is not used when loading data by server-side processing, just Ajax sourced data with client-side processing.
Loading information display string - shown when Ajax loading data

5.8.12 table.language.paginate.first

Type: string
Default Value: `First`
Description: Text used when the pagination control shows the button to take the user to the first page. Note that DataTables writes this property to the document as HTML, so you can use HTML tags in the language string, but HTML entities such as < and > should be encoded as `<` and `>` respectively.
Pagination 'first' button string

5.8.13 table.language.paginate.last

Type: string
Default Value: `Last`
Description: Text used when the pagination control shows the button to take the user to the last page. Note that DataTables writes this property to the document as HTML, so you can use HTML tags in the language string, but HTML entities such as < and > should be encoded as `<` and `>` respectively.
Pagination 'last' button string

5.8.14 table.language.paginate.next

Type: string
Default Value: `Next`
Description: Text to use for the 'next' pagination button (to take the user to the next page). Note that DataTables writes this property to the document as HTML, so you can use HTML tags in the language string, but HTML entities such as < and > should be encoded as `<` and `>` respectively.
Pagination 'next' button string

5.8.15 table.language.paginate.previous

Type: string
Default Value: `Previous`
Description: Text to use for the 'previous' pagination button (to take the user to the previous page). Note that DataTables writes this property to the document as HTML, so you can use HTML tags in the language string, but HTML entities such as < and > should be encoded as `<` and `>` respectively.
Pagination 'previous' button string

5.8.16 table.language.paginate

Type: string

nicht in Original Doku vorhanden als string

5.8.17 table.language.processing

Type: string
Default Value: `Processing...`
Description: Text that is displayed when the table is processing a user action (usually a sort command or similar).
Processing indicator string

5.8.18 table.language.search

Type: string
Default Value: `Search:`
Description: Sets the string that is used for DataTables filtering input control. The token `_INPUT_`, if used in the string, is replaced with the HTML text box for the filtering input allowing control over where it appears in the string. If `_INPUT_` is not given then the input box is appended to the string automatically.
Search input string

5.8.19 table.language.searchPlaceholder

Type: string
Default Value: `''`
Description: HTML 5 introduces a placeholder attribute for `input type="text"` elements to provide informational text for an input control when it has no value. This parameter can be used to set a value for the placeholder attribute in a DataTable's search input.
Search input element placeholder attribute

5.8.20 table.language.thousands

Type: string
Default Value: ,
Description: DataTables' built in number formatter (`formatNumber`) is used to format large numbers that are used in the table information. By default a comma is used, but this can be trivially changed to any other character you wish with this parameter, suitable for any locality, or set to an empty string if you do not which to have a thousands separator character. Please note that unlike the `language.decimal` option, the thousands separator option is used for output of information only (specifically the `info` option). Changing it does not effect how DataTables reads numeric data.
Thousands separator

5.8.21 table.language.url

Type: string
Description: All of the language options DataTables provides can be stored in a file on the server, which DataTables will look up if this parameter is passed. The file must be a valid JSON file, and the object it contains has the same properties as the `language` object in the initialiser object. There are a wide range of translations readily available on this site, in the internationalisation plug-ins. Note that when this parameter is set, DataTables' initialisation will be asynchronous due to the Ajax data load. That is to say that the table will not be drawn until the Ajax request as completed. As such, any actions that require the table to have completed its initialisation should be placed into the `initComplete` callback.
Load language information from remote file

5.8.22 table.language.zeroRecords

Type: string
Default Value: `No matching records found`

Decription: Text shown inside the table records when the is no information to be displayed after filtering. Note that `language.emptyTable` is shown when there is simply no information in the table at all (regardless of filtering), while this parameter is used for when the table is empty due to filtering.
Table empty as a result of filtering string

5.9 DataTables-Options

5.9.1 table.deferLoading

Type: integer, array
Default Value: null
Description: When using server-side processing, the default mode of operation for DataTables is to simply throw away any data that currently exists in the table and make a request to the server to get the first page of data to display. This is fine for an empty table, but if you already have the first page of data displayed in the plain HTML, it is a waste of resources. As such, this option exists to allow you to instruct DataTables to not make that initial request, rather it will use the data already on the page (no sorting etc will be applied to it). `deferLoading` is used to indicate that deferred loading is required, but it is also used to tell DataTables how many records there are in the full table (allowing the information element and pagination to be displayed correctly). In the case where a filtering is applied to the table on initial load, this can be indicated by giving the parameter as an array, where the first element is the number of records available after filtering and the second element is the number of records without filtering (allowing the table information element to be shown correctly). Note that this option only has effect when `serverSide` is enabled. It does not have any effect when using client-side processing.
Delay the loading of server-side data until second draw

5.9.2 table.destroy

Type: boolean
Default Value: false
Description: Initialise a new DataTable as usual, but if there is an existing DataTable which matches the selector, it will be destroyed and replaced with the new table. This can be useful if you want to change a property of the table which cannot be altered through the API. Note that if you are not changing the configuration of the table, but just altering the data displayed by the table, it is far more efficient to use the `ajax.reload()` method (or `rows.add()` etc).
Destroy any existing table matching the selector and replace with the new options.

5.9.3 table.displayStart

Type: integer
Default Value: 0
Description: Define the starting point for data display when using DataTables with pagination (`paging`). Note that this parameter is the number of records (counting from 0), rather than the page number, so if you have 10 records per page and want to start on the third page, it should be `20` rather than 2 or 3.
Initial paging start point

5.9.4 table.dom

Type: string
Default Value: `lfrtip`
Description: DataTables will add a number of elements around the table to both control the table and show additional information about it. The position of these elements on screen are controlled by a combination of their order in the document (DOM) and the CSS applied to the elements. This parameter is used to control their ordering and additional mark-up surrounding them in the DOM. Each table control element in DataTables has a single letter

associated with it, and that letter it used in this dom configuration option to indicate where that element will appear in the document order.
Define the table control elements to appear on the page and in what order

5.9.5 table.lengthMenu

Type: array
Defaul Value: [10, 25, 50, 100]
Description: This parameter allows you to readily specify the entries in the length drop down `select` list that DataTables shows when pagination is enabled. It can be either:

- 1D array of integer values which will be used for both the displayed option and the value to use for the display length, or
- 2D array which will use the first inner array as the page length values and the second inner array as the displayed options. This is useful for language strings such as 'All').

The page length values must always be integer values > `0`, with the sole exception of `-1`. When `-1` is used as a value this tells DataTables to disable pagination (i.e. display all rows). Note that the `pageLength` property will be automatically set to the first value given in this array, unless `pageLength` is also provided.
Change the options in the page length `select` list.

5.9.6 table.order

Type: array
Default Value: `[[0, 'asc']]`
Description: If ordering is enabled (`ordering`), then DataTables will perform a first pass order during initialisation. Using this parameter you can define which column(s) the order is performed upon, and the ordering direction. The `order` must be an array of arrays, each inner array comprised of two elements:

- Column index to order upon
- Direction so order to apply (`asc` for ascending order or `desc` for descending order).

This 2D array structure allows a multi-column order to be defined as the initial state should it be required.
Initial order (sort) to apply to the table

5.9.7 table.orderCellsTop

Type: boolean
Default Value: false
Description: Allows control over whether DataTables should use the top (`true`) unique cell that is found for a single column, or the bottom (`false` - default) to attach the default order listener. This is useful when using complex headers. In this case, when `orderCellsTop` is `false` (default) the cells `1` and `2.2` will have the order event listener applied to them. If `orderCellsTop` is `true` then `1` and `2.1` will have the order event listeners applied to them.
Control which cell the order event handler will be applied to in a column

5.9.8 table.orderClasses

Type: boolean
Default Value: true
Description: DataTables highlight the columns which are used to order the content in the table's body by adding a class to the cells in that column, which in turn has CSS applied to those classes to highlight those cells. This is done by the addition of the classes `sorting_1`, `sorting_2` and `sorting_3` to the columns which are currently being ordered on. The integer value indicates the level of sorting when mutli-column sorting. If more than 3 columns are being ordered upon, the `sorting_3` class is repeated. Please note that this feature can affect performance, particularly in old browsers and when there are a lot of rows to be displayed as it is manipulating a large number of

DOM elements. As such, this option is available as a feature switch to allow this feature to be disabled with working with old browsers or large data sets.
Highlight the columns being ordered in the table's body

5.9.9 table.orderFixed

Type: array, object
Description: The option works in tandem with the `order` option which provides an initial ordering state for the table which can then be modified by the user clicking on column headings, while the ordering specified by this option will always be applied to the table, regardless of user interaction. This fixed ordering can be applied before (`pre`) or after (`post`) the user's own ordering criteria using the two different forms of this option (array or object) described below. The values that are used to describe the ordering conditions for the table are given as two element arrays:

- Column index to order upon
- Direction so order to apply (`asc` for ascending order or `desc` for descending order).

It is also possible to give a set of nested arrays (i.e. arrays in arrays) to allow multi-column ordering to be assigned. This option can be useful if you a column (visible or hidden) which must always be sorted upon first - a priority order or index column for example, or for grouping similar rows together.
Ordering to always be applied to the table

5.9.10 table.orderMulti

Type: boolean
Default Value: true
Description: When ordering is enabled (`ordering`), by default DataTables allows users to sort multiple columns by shift clicking upon the header cell for each column. Although this can be quite useful for users, it can also increase the complexity of the order, potentiality increasing the processing time of ordering the data. Therefore, this option is provided to allow this shift-click multiple column ability. Note that disabling this ability does not impede your ability as a developer to do multiple column ordering using `columns.orderData`, `order` or `order()`, it just disallows the user from performing their own multi-column order.
Multiple column ordering ability control.

5.9.11 table.pageLength

Type: integer
Default Value: 10
Description: Number of rows to display on a single page when using pagination. If `lengthChange` is feature enabled (it is by default) then the end user will be able to override the value set here to a custom setting using a pop-up menu (see `lengthMenu`).
Change the initial page length (number of rows per page)

5.9.12 table.pagingType

Type: string
Default Value: `simple_numbers`
Description: The pagination option of DataTables will display a pagination control below the table (by default, its position can be changed using `dom` and CSS) with buttons that the end user can use to navigate the pages of the table. Which buttons are shown in the pagination control are defined by the option given here. DataTables has six built-in paging button arrangements:

- `numbers` - Page number buttons only (1.10.8)
- `simple` - 'Previous' and 'Next' buttons only
- `simple_numbers` - 'Previous' and 'Next' buttons, plus page numbers

- `full` - 'First', 'Previous', 'Next' and 'Last' buttons
- `full_numbers` - 'First', 'Previous', 'Next' and 'Last' buttons, plus page numbers
- `first_last_numbers` - 'First' and 'Last' buttons, plus page numbers

Further methods can be added using plug-ins.
Pagination button display options

5.9.13 table.renderer

Type: string, object
Description: DataTables adds some complex components to your HTML page, such as the pagination control. The business logic used to calculate information should be displayed (what buttons in the case of the pagination buttons) is core to DataTables and generally doesn't vary how the buttons are actually displayed does based on the styling requirements of the page. For example the pagination buttons might be displayed as `li` elements in a `ul` list, or simply as a collection of `a` buttons. This ability to use different renderers, while maintaining the same core business logic, is fundamental to how DataTables provides integration options for CSS frameworks such as Bootstrap, Foundation and jQuery UI, customising the HTML it uses to fit the requirements of each framework. This parameter controls which renderers will be used. The value given will be used is such a renderer exists, otherwise the default renderer will be used. Additional renderers can be added by plug-ins. DataTables currently supports two different types of renderers:

- `header` - header cell renderer
- `pageButton` - pagination buttons

This list will likely expand significantly in future versions of DataTables!
Display component renderer types

5.9.14 table.retrieve

Type: boolean
Default Value: false
Description: Retrieve the DataTables object for the given selector. Note that if the table has already been initialised, this parameter will cause DataTables to simply return the object that has already been set up - it will not take account of any changes you might have made to the initialisation object passed to DataTables (setting this parameter to true is an acknowledgement that you understand this!). The `destroy` option can be used to reinitialise a table with different options if required.
Retrieve an existing DataTables instance

5.9.15 table.rowId

Type: string
Default Value: `DT_RowId`
Description: It can often be useful to have a `id` attribute on each `tr` element in a DataTable for row selection and data source identification, particularly when using events. DataTables will attempt to automatically read an id value from the data source for each row using the property defined by this option. By default it is `DT_RowId` but can be set to any other name. As with `columns.data` it can also read from a nested JSON data source by using Javascript dotted object notation (e.g. `DT_RowId: 'image.id'`). If no id value for the row is found, the `id` property will not be automatically set.
Data property name that DataTables will use to set `tr` element DOM IDs

5.9.16 table.scrollCollapse

Type: boolean
Default Value: false
Description: When vertical (y) scrolling is enabled through the use of the `scrollY` option, DataTables will force the

height of the table's viewport to the given height at all times (useful for layout). However, this can look odd when filtering data down to a small data set, and the footer is left "floating" further down. This parameter (when enabled) will cause DataTables to collapse the table's viewport down when the result set will fit within the given Y height.
Allow the table to reduce in height when a limited number of rows are shown.

5.9.17 table.search.caseInsensitive

Type: boolean
Default Value: true
Description: Flag to indicate if the filtering should be case insensitive or not.
Control case-sensitive filtering option.

5.9.18 table.search.regex

Type: boolean
Default Value: false
Description: Regular expressions can be used to build fantastically complex filtering terms, but also it is perfectly valid for users to enter characters such as `*` into the filter, so a decision needs to be made if you wish to escape regular expression special characters or not. This option controls that ability in DataTables. It is simply a flag to indicate if the search term should be interpreted as a regular expression (`true`) or not (`false`) and therefore and special regex characters escaped.
Enable / disable escaping of regular expression characters in the search term.

5.9.19 table.search.smart

Type: boolean
Default Value: true
Description: DataTables' built-in filtering is "smart" in that it breaks the user's input into individual words and then matches those words in any position and in any order in the table (rather than simple doing a simple string compare). Although this can significantly enhance usability of the filtering feature, it uses a complex regular expression to perform this task, and as such it can interfere with a custom regular expression input if you enable that option (`search.regex`). As such, this option is provided to disable this smart filtering ability.
Enable / disable DataTables' smart filtering

5.9.20 table.search.search

Type: object
Description: The `search` option allows the way DataTables performs filtering to be set during the initialisation, and to set an initial global filter.
Set an initial filter in DataTables and / or filtering options.

5.9.21 table.searchCols

Type: array
Description: Basically the same as the `search` option, but in this case for individual columns, rather than the global filter, this option defined the filtering to apply to the table during initialisation. The array must be of the same size as the number of columns, and each element be an object with the parameters `search` and `escapeRegex` (the latter is optional). 'null' is also accepted and the default will be used.
Define an initial search for individual columns.

5.9.22 table.searchDelay

Type: integer
Default Value: null
Description: The built in DataTables global search (by default at the top right of every DataTable)will instantly search the table on every keypress when in client-side processing mode and reduce the search call frequency automatically to 400mS when in server-side processing mode. This call frequency (throttling) can be controlled using the `searchDelay` parameter for both client-side and server-side processing. Being able to control the call frequency has a number of uses:

- Older browsers and slower computers can have their processing load reduced by reducing the search frequency
- Fewer table redraws while search can be less distracting for the user
- Reduce the load on the server when using server-side processing by making fewer calls
- Conversely, you can speed up the search when using server-side processing by reducing the default of 400mS to instant (`0`).

As with many other parts of DataTables, it is up to yourself how you configure it to suit your needs! The value given for `searchDelay` is in milliseconds (mS). Please note that this option effects only the built in global search box that DataTables provides. It does not effect the `search()` or `column().search()` methods at all. If you wish to be able to throttle calls to those API methods use the utility method `$.fn.dataTable.util.throttle()`.
Set a throttle frequency for searching

5.9.23 table.stateDuration

Type: integer
Default Value: 7200
Description: Duration for which the saved state information is considered valid. After this period has elapsed the state will be returned to the default. This option is also used to indicate to DataTables if `localStorage` or `sessionStorage` should be used for storing the table's state. When set to `-1` `sessionStorage` will be used, while for 0 or greater `localStorage` will be used. The difference between the two storage APIs is that `sessionStorage` retains data only for the current session (i..e the current browser window). For more information on these two HTML APIs please refer to the Mozilla Storage documentation. Please note that the value is given in seconds. The value `0` is a *special* value as it indicates that the state can be stored and retrieved indefinitely with no time limit.
Saved state validity duration

5.9.24 table.stripeClasses

Type: array
Description: An array of CSS classes that should be applied to displayed rows, in sequence. This array may be of any length, and DataTables will apply each class sequentially, looping when required. Note that by default this option will take the values determined by the `$.fn.dataTable.ext.classes.stripe*` options (these are `odd` and `even` by default).
Set the zebra stripe class names for the rows in the table.

5.9.25 table.tabIndex

Type: integer
Default Value: 0
Description: By default DataTables allows keyboard navigation of the table (sorting, paging, and filtering) by adding a `tabindex` attribute to the required elements. This allows the end user to tab through the controls and press the enter key to activate them, allowing the table controls to be accessible without a mouse. The default tabindex is 0, meaning that the tab follows the flow of the document. You can overrule this using this parameter if you wish. Use a

value of -1 to disable built-in keyboard navigation, although this is not recommended for accessibility reasons.
Tab index control for keyboard navigation

5.10 DataTables-Responsive

5.11 DataTables-RowReorder

5.12 DataTables-Scroller

5.13 DataTables-Select

6 KPI-Dashlets

6.1 Einleitung

Dieses Dokument zeigt welcher der KPI-Funktionen derzeit schon von den TRADUI-Dashlets unterstützt werden.

6.2 Ein KPI-Dashlet erzeugen

Erzeugung eines KPI-Dashlets mit der Dashlet-Engine:

EChart Dashlet erzeugen

```
// DashletManager erzeugen
ExpertsDashlet.createDashletManager(reportContext, "DashletManager");

// Dashlet erzeugen
dashlet = DashletManager.addDashlet("DashletKpiBox", "KPI-Bar", "KPI-Bar", false,
null);
```

6.3 Daten ans Dashlet übergeben

Daten werden ans Dashlet als MemoryBuffer (DataTable) übergeben, wie bei allen Dashlets.

EChart Dashlet erzeugen

```
// Dashlet erzeugen mit mbr
dashlet = DashletManager.addDashlet("DashletKpiBox", "KPI-Bar", "KPI-Bar", false,
mbr);
```

6.4 Hinzufügen von Optionen

Um das Verhalten der KPI-Dashlets zu beeinflussen, können Optionen übergeben werden.
`dashlet.addOption( OPTIONSKEY , OPTIONSVALUE );`
Options-Keys und mögliche Optionswerte werden im Weiteren in diesem Dokument beschrieben (siehe Dashlet Engine KPI).

Copyright © 2023 TRADUI Technologies GmbH

7 Verfügbare Funktionen in Dashlet Engine

7.1 Einleitung

Der gibt viele Funktionen, die in der Dashlet Engine zu Verfügung stehen. Diese Optionen können im Dashboard verwendet werden, um die Anzeige der Dashlets zu steuern, um Werte in einem Dashlet anzupassen und vieles mehr.

Rückgabetyp	Funktion
void	**dashlet_Alert(dashletObj [, infotext] [, options])** Öffnet die Alert-Box eines Dashlets **Details**

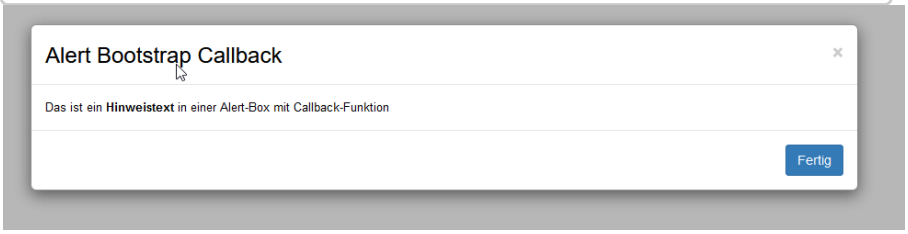
Parameter

Parameter	Typ	Beschreibung	Optional
dashletObj	String	die Dashlet-Id	nein
infoText	String	Anzeige in das Alert Box	Ja
options	JSON-Objekt	Ein JSON-Object mit viele Parameters wie Title des Alert-Boxes, closebutton - Titel, closebuttonclick-Funktion	Ja

Definiert in: dashlets-bootstrap-api.js

Beispiel

```
// Alert Box
dashlet_Alert('chart4', 'Das ist ein <b>Hinweistext</b>
in einer Alert-Box', {title: 'Alert Bootstrap Callback',
closebuttontitle: 'Fertig', closebuttonclick: function()
{alert('Callback Fertig-Button')}});
```



Rückgabetyp	Funktion
Boolean	**dashlet_ArrayContains(_array, _obj)** Prüft ob Wert im Array enthalten ist **Details** Parameter: **Parameter** | **Typ** | **Beschreibung** | **Optional** _array | Array | Array | nein _obj | String / Integer | Wert zu suchen | nein Definiert in: dashlets-bootstrap-api.js
Integer	**dashlet_checkDataTableColumn(tableid, foundx)** Prüft den Index einer DataTable-Spalte **Details** Parameter: **Parameter** | **Typ** | **Beschreibung** | **Optional** tableid | String | Array von Button-Namen | nein foundx | String / Integer | Spalte Namen / Index | nein Definiert in: dashlets-bootstrap-api.js
Integer Array	**dashlet_checkReorderData(tableid, dim)** Prüfen in welcher Reihenfolge die Spalten angeordnet sind. **Details** Parameter: **Parameter** | **Typ** | **Beschreibung** | **Optional** tableid | String | Dashlet-Id des Tables | nein dim | String Array | Name der Spalten in den DataTable | ja Definiert in: dashlets-datatables-functions.js

Rückgabetyp	Funktion
Object	**dashlet_clipRectByRect(params, rect)** Support clipRectByRect eCharts-Functions **Details** Parameter: **Parameter** | **Typ** | **Beschreibung** | **Optional** params | Object | Echarts params Object | nein rect | Object | Gantt Chart rect Object | nein Definiert in: dashlets-bootstrap-api.js
Object	**dashlet_cloneObject(obj)** Kopieren der Javaskript Variablen für cyclic reference **Details** Parameter: **Parameter** | **Typ** | **Beschreibung** | **Optional** obj | Object | Name des Objects, der geklont / kopiert werden soll | nein Definiert in: dashlets-bootstrap-api.js Beispiel:

```
// Macht ein Kopie
var clone_dashlet1 = dashlet_cloneObject(dashlet1);
```

Rückgabetyp	Funktion
Integer	**dashlet_compareValue(a, b [, sortorder])** Vergleicht zwei values **Details** Parameter **Parameter** | **Typ** | **Beschreibung** | **Optional** a | String / Integer | Wert 1 | nein b | String / Integer | Wert 2 | nein sortorder | String | Sort Order (desc oder asc) | Ja Definiert in: dashlets-bootstrap-api.js
boolean	**dashlet_compare(value1, value2, compareOperator)** Vergleich den Wert abhänging an den CompareOperator **Details** Parameter **Parameter** | **Typ** | **Beschreibung** | **Optional** value1 | String / Integer | Vergleich Wert 1 | nein value2 | String / Integer | Vergleich Wert 2 | nein compareOperator | String | Definition für positioneren eines Dashlets | nein Definiert in: dashlets-datatables-functions.js

Rückgabetyp	Funktion
void	**dashlet_Confirm(dashletObj [, infotext] [, options])** Öffnet die Confirm-Box eines Dashlets **Details**

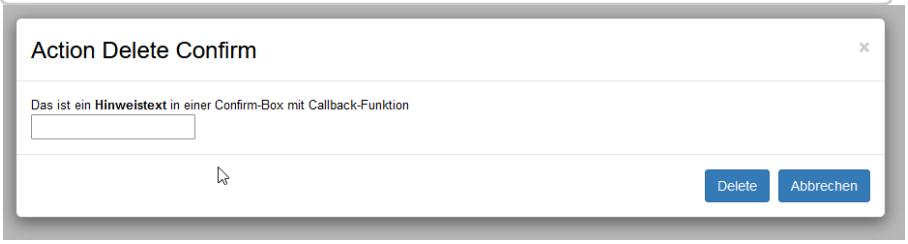
Parameter

Parameter	Typ	Beschreibung	Optional
dashletObj	String	die Dashlet-Id	nein
infoText	String	Anzeige in das Confirm Box	Ja
options	JSON-Objekt	Ein JSON-Object mit viele Parameters wie Title des Alert-Boxes, closebutton - Titel, closebuttonclick-Funktion, cancelbutton - Titel ,cancelbutton - Funktion	Ja

Definiert in: dashlets-bootstrap-api.js

Beispiel

```
// Confirm Box mit Input Text
dashlet_Confirm('chart4', 'Das ist ein <b>Hinweistext</b>
in einer Confirm-Box mit Callback-Funktion<br/><input
type=text id=inputfieldid name=inputfieldname>', {title:
'Action Delete Confirm', closebuttontitle: 'Delete',
closebuttonclick: function(){alert('Callback Delete-
Button')}, cancelbuttontitle: 'Abbrechen',
cancelbuttonclick: function(){alert('Callback Cancel-
Button')}});
```



Rückgabetyp	Funktion
void	**dashlet_DataTable_initColumnSearch(tableid [, searchcolumns] [, opt])** Erzeugt den Suchheader in dem DataTables-Dashlet wenn aktiviert **Details**

Parameter	

Parameter	Typ	Beschreibung	Optional
tableid	String	Dashlet-Id des Tables	nein
searchcolumns	Array of String	Name die Splaten, die ein Search-Feld darstellten sollten	ja
opt	String	Options (zb. flexselect aktivieren oder nicht)	ja

Definiert in	dashlets-datatables-functions.js

Rückgabetyp	Funktion
void	**dashlet_DataTable_initComplexHeader(tableid, header [, opt])** Fügt einen zusätzlichen Header über der Tabelle ein **Details**

Parameter

Parameter	Typ	Beschreibung	Optional
tableid	String	Dashlet-Id des Tables	nein
header	String	Header Information für den Tabelle	nein
opt	String	Weitere Optionen	ja

Definiert in: dashlets-datatables-functions.js

Beispiel

```
dashlet_DataTable_initComplexHeader('table1', '<th colspan="3"><div style="margin-bottom: 10px; height: 40px; background-color: red;"><b>Cars Trains Ships and more</b></div></th><th colspan="5"><div style="margin-bottom: 10px; height: 40px; background-color: yellow;"><b>Classic Models</b></div></th>', null);
```

Der DataTable sieht als folgendes :

void	**dashlet_DataTable_clearColumnSearch(tableid)** Enfernt alle Suchbedingungen der Spaltensuche **Details**

Parameter

Parameter	Typ	Beschreibung	Optional
tableid	String	Dashlet-Id des Tables	nein

Definiert in: dashlets-datatables-functions.js

Rückgabetyp	Funktion
void	**dashlet_DataTable_clearSelectedColumnSearch(tableid)** Löscht alle selektieren Suchbedingungen der Spaltensuche **Details** Parameter: | Parameter | Typ | Beschreibung | Optional | | tableid | String | Dashlet-Id des Tables | nein | Definiert in: dashlets-datatables-functions.js
String Array	**dashlet_DataTable_group(tableid, dataSet, group_columns, dim, agg)** Gruppiert eine DataTable nach Gruppierungsspalten und liefert die gruppierten Daten zurück **Details** Parameter: | Parameter | Typ | Beschreibung | Optional | | tableid | String | Dashlet-Id des Tables | nein | | dataSet | String | JSON DataSet Array für die Grüppierung | nein | | group_columns | String Array | Array mit den grüppierte Spaltennamen | nein | | dim | String | Dimension | nein | | agg | String | Aggregate Funktion (zb. DIM, FIRST, LAST, SUM) | nein | Definiert in: dashlets-datatables-functions.js
void	**dashlet_DataTable_removeColumnSearch(tableid)** Gelöscht den Suchheader in dem DataTables-Dashlet **Details** Parameter: | Parameter | Typ | Beschreibung | Optional | | tableid | String | Dashlet-Id des Tables | nein | Definiert in: dashlets-datatables-functions.js

Rückgabetyp	Funktion
void	**dashlet_DataTable_restoreVisibiliyColumns(tableid)** Lädt die gespeicherte Sichtbarkeit zurück auf die Tabelle **Details** Parameter: | Parameter | Typ | Beschreibung | Optional | | tableid | String | Dashlet-Id des Tables | nein | Definiert in: dashlets-datatables-functions.js
void	**dashlet_DataTable_saveVisibiliyColumns(tableid)** Speicheren die Spalten Liste **Details** Parameter: | Parameter | Typ | Beschreibung | Optional | | tableid | String | Dashlet-Id des Tables | nein | Definiert in: dashlets-datatables-functions.js
void	**dashlet_DataTable_setColumnSearchValue(tableid, columntitle, value)** Setzt die Suchedingung auf die richtige Spalte, selbst wenn die Spalten per colReorder neu angeordnet wurden **Details** Parameter: | Parameter | Typ | Beschreibung | Optional | | tableid | String | Dashlet-Id des Tables | nein | | columntitle | String | Name des Splatens | nein | | value | String | Such Werte | nein | Definiert in: dashlets-datatables-functions.js

Rückgabetyp	Funktion
void	**dashlet_DataTable_setVisibiliyColumnName(tableid, colname, visibiliy)** Lädt die gespeicherte Sichtbarkeit zurück auf die Tabelle **Details** Parameter: **Parameter** | **Typ** | **Beschreibung** | **Optional** tableid | String | Dashlet-Id des Tables | nein colname | String | Name des Spaltes | nein visibiliy | boolean | true: Spalte ist sichbar | nein Definiert in: dashlets-datatables-functions.js
String Array	**dashlet_getAggForArray(arr [, aggfunc] [, opt])** Führt auf das übergebene Array die Agg-Funktion aus und liefert das Result **Details** Parameter: **Parameter** | **Typ** | **Beschreibung** | **Optional** arr | String Array | Array | nein aggfunc | String | Aggregate Funktion (zb. DIM, FIRST, LAST, SUM) | ja opt | String Array | Array mit den grüppierte Spaltennamen | ja Definiert in: dashlets-datatables-functions.js

Rückgabetyp	Funktion
Object	**dashlet_getDataRowsFromDataDef(_dataSet,_dataSet_dim [, search1] [, found1] [, search2] [, found2] [, search3] [, found3] [, search4] [, found4])** Filtert das Dataset nur mit den Zeilen, die mit den Suchwerten übereinstimmen **Details**

Parameter

Parameter	Typ	Beschreibung	Optional
_dataSet	Object	Name des Datasets	nein
_dataSet_dim	Array of Strings	Namen von die Splaten des Datasets	nein
search1	Array / String / Integer	1. Filter Wert	Ja
found1	String / Integer	1. Filter Spalte-Index / Spalte-Name	Ja
search2	Array / String / Integer	2. Filter Wert	Ja
found2	String / Integer	2. Filter Spalte-Index/ Spalte-Name	Ja
search3	Array / String / Integer	3. Filter Wert	Ja
found3	String / Integer	3. Filter Spalte-Index/ Spalte-Name	Ja
search4	Array / String / Integer	4. Filter Wert	Ja
found4	String / Integer	4. Filter Spalte-Index/ Spalte-Name	Ja

Definiert in: dashlets-bootstrap-api.js

Beispiel

```
// Zeilen aus chart1_dataset, der in Spalte 'country' das
Wert 'USA' hat
```

Rückgabetyp	Funktion
	`dashlet_getDataRowsFromDataDef(chart1_dataSet, chart1_dataSet_dim, 'USA', 'Country')`

Rückgabetyp	Funktion
Object	**dashlet_getDataRowsFromDataSet(_dataSet [, search1] [, found1] [, search2] [, found2] [, search3] [, found3] [, search4] [, found4])** Filtert das Dataset nur mit den Zeilen, die mit den Suchwerten übereinstimmen **Details**

Parameter

Parameter	Typ	Beschreibung	Optional
_dataSet	Object	Name des Datasets	nein
search1	Array / String / Integer	1. Filter Wert	Ja
found1	Integer	1. Filter Spalte-Index	Ja
search2	Array / String / Integer	2. Filter Wert	Ja
found2	Integer	2. Filter Spalte-Index	Ja
search3	Array / String / Integer	3. Filter Wert	Ja
found3	Integer	3. Filter Spalte-Index	Ja
search4	Array / String / Integer	4. Filter Wert	Ja
found4	Integer	4. Filter Spalte-Index	Ja

Definiert in: dashlets-bootstrap-api.js

Beispiel

```
// Zeilen aus chart1_dataset, der in Spalte 2 das Wert 'France' hat
dashlet_getDataRowsFromDataSet(chart1_dataSet, 'France', 2);
```

Rückgabetyp	Funktion
Array of Strings	**dashlet_getDistinctValuesFromDataSet(_dataSet,distinctColumn [, search1] [, found1] [, search2] [, found2] [, search3] [, found3] [, search4] [, found4])** Eindeutige (Unique) Werte aus dem Datensatz erhalten **Details**

Parameter

Parameter	Typ	Beschreibung	Optional
_dataSet	Object	Name des Datasets	nein
distinctColumn	Integer	Spalten-Index in dem der eindeutige (unique) werte gesucht werden soll	nein
search1	Array / String / Integer	1. Filter Wert	Ja
found1	Integer	1. Filter Spalte-Index	Ja
search2	Array / String / Integer	2. Filter Wert	Ja
found2	Integer	2. Filter Spalte-Index	Ja
search3	Array / String / Integer	3. Filter Wert	Ja
found3	Integer	3. Filter Spalte-Index	Ja
search4	Array / String / Integer	4. Filter Wert	Ja
found4	Integer	4. Filter Spalte-Index	Ja

Definiert in: dashlets-bootstrap-api.js

Beispiel

```
// Liste der Buttons, die darstellt werden soll in
'topbar2btngrp',  mit unique Werte inSpalte 3 in
table1_dataSet
dashlet_setButtonsVisibility(dashlet_getDistinctValuesFro
mDataSet(table1_dataSet, 3), 'topbar2btngrp');
```

Rückgabetyp	Funktion
	`// Liste der Buttons, die darstellt werden soll in 'topbar2btngrp', mit unique Werte in Spalte 3 in table1_dataSet, wobei der table1_dataset ist auch gefiltert mit 'Vintage Cars' in Spalte 0` `dashlet_setButtonsVisibility(dashlet_getDistinctValuesFromDataSet(table1_dataSet, 3, 'Vintage Cars', 0), 'topbar2btngrp');`
Object	**dashlet_getCopyHierachy(_hdataorg)** Erzeugt ein Kopiertes Objekt der Hierachy **Details** Parameter: _hdataorg – Object – Object Name , der ein Kopie erstellt werden soll – nein Definiert in: dashlets-bootstrap-api.js
Integer	**dashlet_getIndexFromDim(_dataSet_dim,found)** Liefert die Index eines Suchwertes in ein Array von Strings **Details** Parameter: _dataSet_dim – Array of Strings – Namen von die Splaten eines Datasets – nein; found – String – Name des Spaltes – Ja Definiert in: dashlets-bootstrap-api.js Beispiel: `// Zeilen aus chart1_dataset, der in Spalte 'country' das Wert 'USA' hat` `dashlet_getDataRowsFromDataDef(chart1_dataSet, chart1_dataSet_dim, 'USA', 'Country')`

Rückgabetyp	Funktion
boolean	**dashlet_onlyUnique(value, index, self)** Liefert eindeutige (unique) Werte in eine Liste **Details**

Parameter	

Parameter	Typ	Beschreibung	Optional
value	String	Dashlet-Id des Tables	nein
index			
self			

Definiert in	dashlets.js
Beispiel	

```
// obtain only unique values in selectedList
selectedList = selectedList.filter( dashlet_onlyUnique );
```

Rückgabetyp	Funktion
void	**dashlet_openModalPanel(dashletObj)** Öffnet ein Dashlet in einer ModalPanel-DashletBox **Details**

Parameter	

Parameter	Typ	Beschreibung	Optional
dashletObj	String	die Dashlet-Id	nein

Definiert in	dashlets-bootstrap-api.js
Beispiel	

```
// Offenen der Tabelle als Popup
dashlet_openModalPanel('table1') ;
```

Rückgabetyp	Funktion
Object	**dashlet_sortDataDef(_dataset,_dataset_dim [, idx1] [, so1] [, idx2] [, so2] [, idx3] [, so3] [, idx4] [, so4])** Sortiert das Dataset nach den angegeben Spalten **Details**

Parameter

Parameter	Typ	Beschreibung	Optional
_dataSet	Object	Name des Datasets	nein
_dataSet_dim	Array of Strings	Namen von die Splaten des Datasets	nein
idx1	String / Integer	1. Filter Spalte Name/Index	Ja
so1	String	Sort Order (desc oder asc)	Ja
idx2	String / Integer	2. Filter Spalte Name/ Index	Ja
so2	String	Sort Order (desc oder asc)	Ja
idx3	String / Integer	3. Filter Spalte Name / Index	Ja
so3	String	Sort Order (desc oder asc)	Ja
idx4	String / Integer	4. Filter Spalte Name/ Index	Ja
so4	String	Sort Order (desc oder asc)	Ja

Definiert in: dashlets-bootstrap-api.js

Beispiel

```
// Sortiert den Data in chart1_dataSet_selected mit
Spalte 'Ordernumber' in der angegebene order
```

Rückgabetyp	Funktion

```
dashlet_sortDataDef(chart1_dataSet_selected,
chart1_dataSet_dim, 'Ordernumber', 'desc');
dashlet_sortDataDef(chart1_dataSet_selected,
chart1_dataSet_dim, 'Ordernumber', 'asc');
```

Rückgabetyp	Funktion
Object	**dashlet_sortDataSet(_dataset [, idx1] [, so1] [, idx2] [, so2] [, idx3] [, so3] [, idx4] [, so4])** Sortiert das Dataset nach den angegeben Spalten Indexes **Details**

Parameter

Parameter	Typ	Beschreibung	Optional
_dataSet	Object	Name des Datasets	nein
_dataSet_dim	Array of Strings	Namen von die Splaten des Datasets	nein
idx1	Integer	1. Filter Spalte Index	Ja
so1	String	Sort Order (desc oder asc)	Ja
idx2	Integer	2. Filter Spalte Index	Ja
so2	String	Sort Order (desc oder asc)	Ja
idx3	Integer	3. Filter Spalte Index	Ja
so3	String	Sort Order (desc oder asc)	Ja
idx4	Integer	4. Filter Spalte Index	Ja
so4	String	Sort Order (desc oder asc)	Ja

Definiert in: dashlets-bootstrap-api.js

Beispiel

```
// Sortiert den Data in chart1_dataSet_selected mit
angegebene Spalten-Indexes in der angegebene order
dashlet_sortDataSet(chart1_dataSet, 2, 'desc', 2, 'desc',
2, 'desc', 1, 'asc');
```

Rückgabetyp	Funktion
Array	**dashlet_searchHierachy(data, found1, search1, found2, search2, found3, search3)** Sucht in den drei Leveln einer Hierachie und liefert reduzierte Hierachie zurück **Details**

Parameter

Parameter	Typ	Beschreibung	Optional
data	JSON Array	zb. Series Object in Echarts	nein
found1	String	Spalte Name1	ja
search1	String / Array	Such Wert1	ja
found2	String	Spalte Name2	ja
search2	String / Array	Such Wert2	ja
found3	String	Spalte Name3	ja
search3	String / Array	Such Wert3	ja

Definiert in: dashlets-bootstrap-api.js

Beispiel

```
// Level 1 Suche (Sunburst chart)
var newdata =
dashlet_searchHierachy(chart1_predefined_series.data1[0].
data, 'NAME', ['Trains', 'Ships']);

// Level 2 Suche (Sunburst chart)
var newdata =
dashlet_searchHierachy(chart1_predefined_series.data1[0].
data, null, null, 'NAME', '1:32');

// Level 3 Suche (Sunburst chart)
var newdata =
dashlet_searchHierachy(chart1_predefined_series.data1[0].
data, null, null, null, null, 'NAME', ['S700_1138',
'S700_1938']);
```

Rückgabetyp	Funktion
void	**dashlet_setButtonsVisibility(textbutton, buttongrp [, displayflag])** Öffnet ein Dashlet in einer ModalPanel-DashletBox. **Details**

Parameter

Parameter	Typ	Beschreibung	Optional
textbutton	Array of Strings	Array von Button-Namen	nein
buttongrp	String	Die HTML ID der Button-Group	nein
displayflag	Boolean	Falls true werden immer alle Buttons dargestellt Falls false werden nur die Buttons aus dem textbutton Parameter dargestellt	ja. Default = true

Definiert in: dashlets-bootstrap-api.js

Beispiel

```
// Liste der Buttons, die dargestellt werden sollen.
dashlet_setButtonsVisibility(dashlet_getDistinctValuesFro
mDataSet(table1_dataSet, 3), 'topbar2btngrp');
```

Rückgabetyp	Funktion
void	**dashlet_setFixedPosition(_dashletid, _status [, _prop])** Fixiert das Position eines Dashlets sodass es beim Skroll immer angezeigt wird **Details**

Parameter

Parameter	Typ	Beschreibung	Optional
_dashletid	String	Name des Dashlets, der fixiert werden soll	nein
_status	Boolean	true : Dashlet fixiern beim Skrollen false: Dashlet nicht fixern	nein
_prop	JSON	Definition für positioneren eines Dashlets	ja

Definiert in: dashlets-bootstrap-api.js

Beispiel

```
// Fixieren ein Dashlet beim Scroll
dashlet_setFixedPosition('dashletbox-dashlet1',true,{'z-index':999,left:300});
```

Rückgabetyp	Funktion
boolean	**exists_DataTable(tableid)** Prüft ob der DataTable existiert oder nicht **Details**

Parameter

Parameter	Typ	Beschreibung	Optional
tableid	String	Dashlet-Id des Tables	nein

Definiert in: dashlets-datatables-functions.js

Rückgabetyp	Funktion
String Array	**getSeriesSelection(sourceList, filterList)** Funktion zum Zurückgeben der gefiltertenQuellenliste basierend auf dem booleschen Array filterList. Diese Boolean array ist die gleiche größe wie der sourceList. **Details**

Parameter

Parameter	Typ	Beschreibung	Optional
sourceList	String Array	Die Liste für filtering	nein
filterList	Boolean Array	Boolean array mit true / false.	nein

Definiert in: dashlets.js

Beispiel

```
// Sort the chart1_xAxis_data based on the array of boolean values (chart1_filterList)
chart1_xAxis_data = getSeriesSelection(chart1_xAxis,chart1_filterList);
```

void

setDashletBoxExpand(params)

Dashlet-API-Funktion / Muss von der Integrations-Umgebung zur Verfügung gestellt werden

Details

Parameter

Parameter	Typ	Beschreibung	Optional
params	JSON Array	Definition mit Parameters 'dashledid' und 'dashletresponsive'	nein

Definiert in: dashlets-bootstrap-api.js

Beispiel

```
// Box Expand / Compress Function call
setDashletBoxExpand( [{name: 'dashletid', value: 'dashlet1'}, {name: 'dashletrseponsive', value: 'col-md-12 col-sm-12 col-xs-12'}] );
```

Rückgabetyp	Funktion
void	**<TABLE_ID>_lazyLoadTable([, search1] [, found1] [, search2] [, found2] [, search3] [, found3] [, search4] [, found4][, search5] [, found5])** Loads the dataTable Dashlet with the filtered values. **Details**

Parameter

Parameter	Typ	Beschreibung	Optional
search1	Array / String / Integer	1. Filter Wert	Ja
found1	Integer / String	1. Filter Spalte-Index / Spalten-Name	Ja
search2	Array / String / Integer	2. Filter Wert	Ja
found2	Integer / String	2. Filter Spalte-Index / Spalten-Name	Ja
search3	Array / String / Integer	3. Filter Wert	Ja
found3	Integer / String	3. Filter Spalte-Index / Spalten-Name	Ja
search4	Array / String / Integer	4. Filter Wert	Ja
found4	Integer / String	4. Filter Spalte-Index / Spalten-Name	Ja
search5	Array / String / Integer	5. Filter Wert	Ja
found5	Integer / String	5. Filter Spalte-Index / Spalten-Name	Ja

Definiert in: dashlets.js

Beispiel

```
// Sort the chart1_xAxis_data based on the array of
boolean values (chart1_filterList)
```

Rückgabetyp	Funktion
	`table1_lazyLoadTable('USA','Country' , 'Ordered', 'Status');`