



dashBIRD

dashBIRD Handbuch

Dokumentversion vom 06. 11. 2023

Inhalt

1	Über diese Dokumentation	10
1.1	Änderungshistorie.....	10
1.2	Hinweise und Konventionen	10
1.3	Inhalt.....	10
2	Grundlagen.....	12
2.1	Produktübersicht	12
2.1.1	Einführung.....	12
2.2	Plugins downloaden und Umgebung vorbereiten	12
2.2.1	Einführung.....	12
2.3	Architektur und Funktionsweise	14
2.3.1	Einführung.....	14
2.4	Schnelleinstieg.....	15
2.4.1	Einführung.....	15
3	Daten und Parameter	18
3.1	Einführung.....	18
3.2	Arten von Daten	18
3.2.1	Einführung.....	18
3.3	Daten in das richtige Format bringen	23
3.3.1	Einführung.....	23
3.4	Datenstrategien	23
3.4.1	Einführung.....	23
3.5	Datenquellen.....	25
3.5.1	Einführung.....	25
4	Dashlets erstellen.....	26
4.1	Fortsetzung Schnellstart	26
4.1.1	Einführung.....	26
4.2	Fehlerbehandlung.....	26
4.2.1	Einführung.....	26
	Try-catch Blocks.....	26
	Output Log.....	27
	Browser Console (F12-Key).....	27

4.3	Anatomie eines Dashlets	28
4.3.1	Einführung.....	28
4.3.2	Erklärung	30
	Box Area.....	30
	Top / Bottom Button Bar	30
	Top / Bottom HTML Content	30
	Chart Area.....	31
4.4	Charting Engines	31
4.4.1	Einführung.....	31
4.5	Chart-/Dashlettypen	34
4.5.1	3D Weltkarte	34
	Einführung.....	34
4.5.2	Balkendiagramm.....	34
	Einführung.....	34
	Dashlets erzeugen	36
	Dashlet Optionen	37
	FAQ.....	37
4.5.3	Baumdiagramm (Tree Diagram)	39
	Einführung.....	39
4.5.4	Blasen-/Kuchendiagramm (Bubble Pie)	40
	Einführung.....	40
4.5.5	Blasendiagramm (Bubble Chart).....	41
	Einführung.....	41
4.5.6	Box-Plot	42
	Einführung.....	42
4.5.7	Data Dashlet	43
	Einführung.....	43
	Dashlets erzeugen	43
	Dashlet Optionen	44
4.5.8	Dynamische Tabellen (DataTables)	44
	Einführung.....	44
	Dashlets erzeugen	45
	Dashlet Optionen	45
	FAQ (DataTable)	46
4.5.9	Flächen-/Stufendiagramm (Area Chart / Area-Step).....	66

Einführung.....	66
Dashlets erzeugen.....	67
Dashlet Optionen.....	68
4.5.10 Gantt Diagramm.....	68
Einführung.....	68
Dashlets erzeugen.....	69
FAQ.....	70
4.5.11 Graph / Chord Diagramm.....	77
Einführung.....	77
4.5.12 Gruppen-Dashlet.....	78
Einführung.....	78
Dashlets erzeugen.....	78
FAQ.....	79
4.5.13 Heatmap.....	81
Einführung.....	82
4.5.14 HTML Dashlet.....	82
Einführung.....	82
Dashlets erzeugen.....	83
4.5.15 Illustrationsdiagramm (Piktogramm).....	84
Einführung.....	84
4.5.16 Kacheldiagramm (Treemap).....	85
Einführung.....	85
Dashlets erzeugen.....	85
4.5.17 Kartendiagramm (Geo/Map Diagram).....	86
Einführung.....	86
Dashlets erzeugen.....	87
Dashlet Optionen.....	87
4.5.18 Karten Diagramm (Leaflet).....	87
Einführung.....	87
Dashlets erzeugen.....	88
Dashlet Optionen.....	89
FAQ (Leaflet).....	89
Leaflet Markers.....	104
Leaflet Routing.....	108
4.5.19 Kerzendiagramm (Candlestick).....	116
Einführung.....	117

4.5.20 KPI-Box / -Leiste	117
Einführung	117
Dashlets erzeugen	119
Dashlet Optionen	119
FAQ	123
4.5.21 Kuchen- und Donutdiagramm (Pie Chart)	125
Einführung	125
Dashlets erzeugen	126
Dashlet Optionen	127
FAQ	127
4.5.22 Linien-/Kurvendiagramm	130
Einführung	131
Dashlets erzeugen	132
Dashlet Optionen	133
FAQ	133
4.5.23 Messzeiger (Gauge) Diagramm	134
Einführung	134
Dashlets erzeugen	135
Dashlet Optionen	136
4.5.24 Organisational Chart	136
Einführung	136
Dashlets erzeugen	137
Dashlet Optionen	137
4.5.25 Parameter Dashlet	138
Einführung	138
Dashlets erzeugen	138
Dashlet Optionen	138
Beispiel Report	139
4.5.26 PivotTable	139
Einführung	139
Dashlets erzeugen	140
Dashlet Optionen	141
FAQ	141
4.5.27 Sankey Diagramm	151
Einführung	151
4.5.28 Spinnennetzdiagramm / Netzdiagramm (Radar chart)	152

Einführung	152
Dashlets erzeugen	152
Dashlet Optionen	152
4.5.29 Streamgraph/Flussdiagramm	153
Einführung	153
4.5.30 Streudiagramm (Scatter Chart)	154
Einführung	154
Dashlets erzeugen	155
FAQ	156
4.5.31 Sunburst Diagramm	157
Einführung	157
Dashlets erzeugen	157
Dashlet Optionen	158
FAQ	159
4.5.32 SVG Visualisierungen/-Karten	161
Einführung	161
4.5.33 Tabellen	162
Einführung	162
Dashlets erzeugen	162
Dashlet Optionen	163
4.5.34 Trichterdiagramm (Funnel Chart)	163
Einführung	163
Dashlets erzeugen	164
4.5.35 Venn Diagramm	165
Einführung	165
4.5.36 Wortwolke (Word-Cloud)	166
Einführung	166
4.6 Optionen setzen	167
4.6.1 Anreicherung mit Dokumentation	167
Einführung	167
4.6.2 Komponenten	169
Buttons	169
Multiselect List	171
Mutiselect Boxen	172
Single-select Boxen	175

5	Dashboards designen	177
5.1	Best Practice.....	177
5.1.1	Einführung.....	177
5.2	Positionierung, Größe & Responsive Design	177
5.2.1	Positionierung.....	177
	Einführung.....	177
5.2.2	Positionierung.....	177
	Einführung.....	177
5.2.3	Responsive Design	178
	Einführung.....	178
5.3	Mehrsprachigkeit/Lokalisierung	179
5.3.1	Einführung.....	179
6	Interaktivität.....	180
6.1	Alert-Box Anzeigen	180
6.1.1	Einführung.....	180
6.1.2	Optionen.....	180
6.2	Confirm-Box Anzeigen	181
6.2.1	Einführung.....	181
6.2.2	Optionen.....	181
	Einfache Confirm-Box	181
	Confirm-Box mit ein Input Feld	182
6.3	Drilldowns	183
6.3.1	Einführung.....	183
6.3.2	Drilldown erzeugen	183
6.4	Eigene CSS-Datei für jede Dashlet hinzufügen	184
6.4.1	Einführung.....	184
6.4.2	Optionen.....	184
6.5	Eigene Scripte erstellen	185
6.5.1	Einführung.....	185
6.5.2	Optionen.....	185
	Variant 1 (Echarts) :	185
	Variant 2 (HTML Dashlet) :	186
	Variant 3 (DataTable Dashlet) :	186
6.6	Expand Button anzeigen.....	187

6.6.1	Einführung.....	187
6.6.2	Optionen.....	187
	Expand Button anzeigen.....	187
6.6.3	FAQ.....	188
6.7	Gruppieren verschiedene Dashlets zusammen.....	188
6.7.1	Einführung.....	188
6.7.2	Optionen.....	188
6.7.3	FAQ.....	190
6.8	HTML Content Rendern	193
6.8.1	Einführung.....	193
6.8.2	Optionen.....	193
6.9	Info-Box Anzeigen	194
6.9.1	Einführung.....	194
6.9.2	Optionen.....	194
	Info Box als ein Pop-up anzeigen	195
	Info box als Modal dialog anzeigen	196
6.10	Kombination von meherere Dashlets in ein Box	196
6.10.1	Einführung.....	196
6.10.2	Optionen.....	196
6.10.3	FAQ.....	197
6.11	Legend - Toggle Visibility.....	198
6.11.1	Einführung.....	198
6.11.2	Toggle Visibilty erzeugen.....	199
6.12	Native Dokumente als Download anbieten.....	200
6.12.1	Einführung.....	200
6.12.2	Drilldown erzeugen	200
6.13	Unterberichte starten	202
6.13.1	Einführung.....	202
6.13.2	Unterberichte starten Skripte	202
6.13.3	FAQ.....	202
7	Bereitstellung und Weiterführendes.....	205
7.1	Grundlagen in der servBIRD Administration.....	205
7.1.1	Einführung.....	205
	Weitere Dokumentation	205

7.2	Bereitstellung/Deployment.....	205
7.2.1	Einführung.....	205
	Weitere Dokumentation	206
7.3	Caching und Vor-Generierung.....	206
7.3.1	Einführung.....	206
	Caching (Ausführungsbeschränkungen).....	206
	Vor-Genereierung.....	207
	Weitere Dokumentation	207
7.4	Rollen und Rechte.....	207
7.4.1	Einführung.....	207
7.4.2	Weiter Dokumentation	208
7.5	Snapshots/Archive	208
7.5.1	Einführung.....	208
	Weitere Dokumentation	208

1 Über diese Dokumentation

1.1 Änderungshistorie

Datum	Version	Änderungsbeschreibung	Änderung von
07.05.2021	v1.0	Initialer Stand der dashBIRD 1.0 Dokumentation	Team TRADUI

1.2 Hinweise und Konventionen



Tipp

Dieser Block hebt Informationen hervor, die bspw. zeit- oder ressourcenschonende Verfahren oder Best Practices erläutern.



Information

In diesem Block werden Informationen von besonderer Bedeutung oder besonderem Interesse hervorgehoben.



Hinweis

Dieser Block weist auf zu beachtende Informationen hin oder warnt vor Stolperfallen. Bitte lesen Sie diese Blöcke aufmerksam!



Achtung

Diese Hinweise deuten auf Informationen hin, die bei Missachtung oder falscher Nutzung zu Fehlfunktion der Software oder Löschung von relevanten Daten führen können.

Beispiel

In diesem Block wird ein Beispiel zum jeweiligen Kontext aufgeführt.

1.3

Inhalt

1. Grundlagen

a. **Produktübersicht**

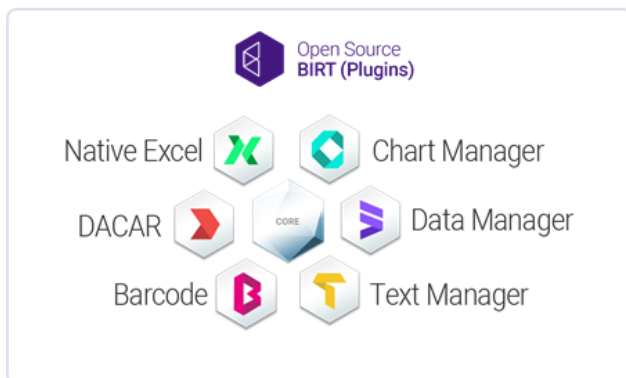
- b. Plugins downloaden und Umgebung vorbereiten
 - c. Architektur und Funktionsweise
 - d. Schnelleinstieg
- 2. **Daten und Parameter**
 - a. Arten von Daten
 - b. Daten in das richtige Format bringen
 - c. Datenstrategien
 - d. Datenquellen
- 3. **Dashlets erstellen**
 - a. Ein erstes Dashlet erzeugen
 - b. Chart Typen
 - i. Bar
 - ii. Area
 - iii. Candlestick
 - iv. DataTables
 - v. ...
 - c. Optionen setzen, übergeben, auslagern
 - i. Charts formatieren
 - ii. Metainformationen an Dashlets hinterlegen
- 4. **Dashboards designen**
 - a. Gutes Dashboard Design mit Corporate Identity, Farben und Themes
 - b. Position, Größe, Responsiviness
 - c. Mehrsprachigkeit
- 5. **Interaktivität**
 - a. Drill Down
 - b. Native Dokumente hinterlegen
 - c. Filter
- 6. **Bereitstellung und Weiterführendes**
 - a. Deployment
 - b. Pre-Caching im servBIRD
 - c. Rollen und Rechte
 - d. Gesamte Dashboards archivieren und versenden
- 7. **Erweitern der Engine**
 - a. Anbindung von externen Web-Services
 - b. Eigene Charting-Libraries einbauen/nutzen
 - c. Eigene Dashlets entwickeln

2 Grundlagen

2.1 Produktübersicht

2.1.1 Einführung

Ein Übersicht von die enizele Plugins und der **servBIRD** server ist dargestellt.



2.2 Plugins downloaden und Umgebung vorbereiten

2.2.1 Einführung

Die Schritte für das Umgebung vorzubereiten ist in diesem Kapitel erklärt.

1. Eclipse herunterladen

Eclipse unterladen und lokal installieren

<http://download.eclipse.org/birt/downloads/build.php?build=R-R1-4.6.0-201606072112>

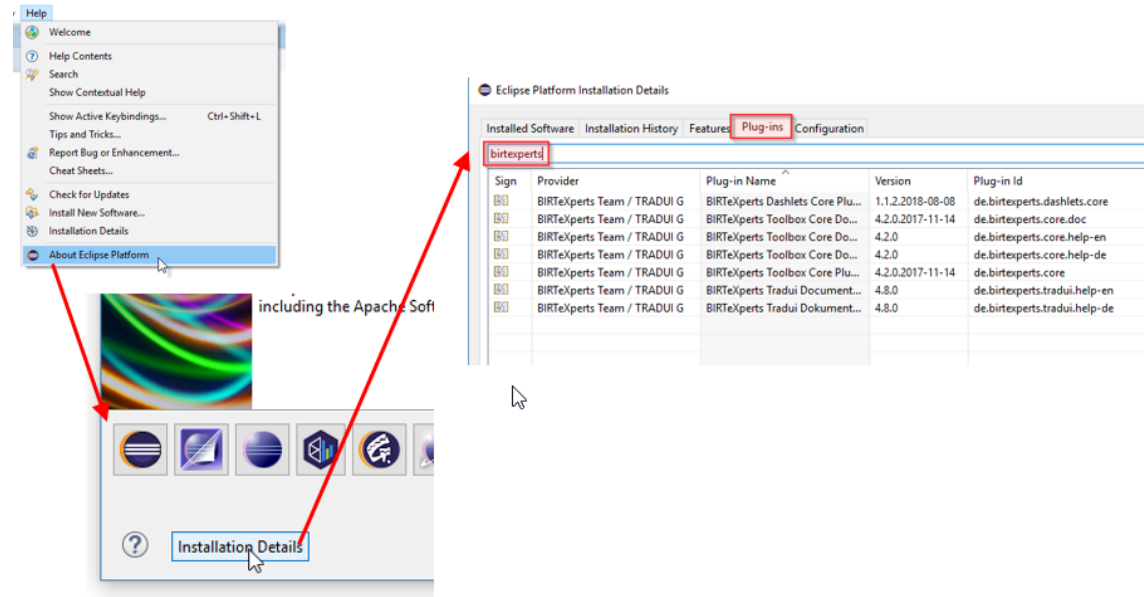
2. Plugins herunterladen

Die Plugins unterladen und in den „plugins“ Verzeichnis von Eclipse kopieren

3. Prüfen ob die Plugins korrekt geladen sind

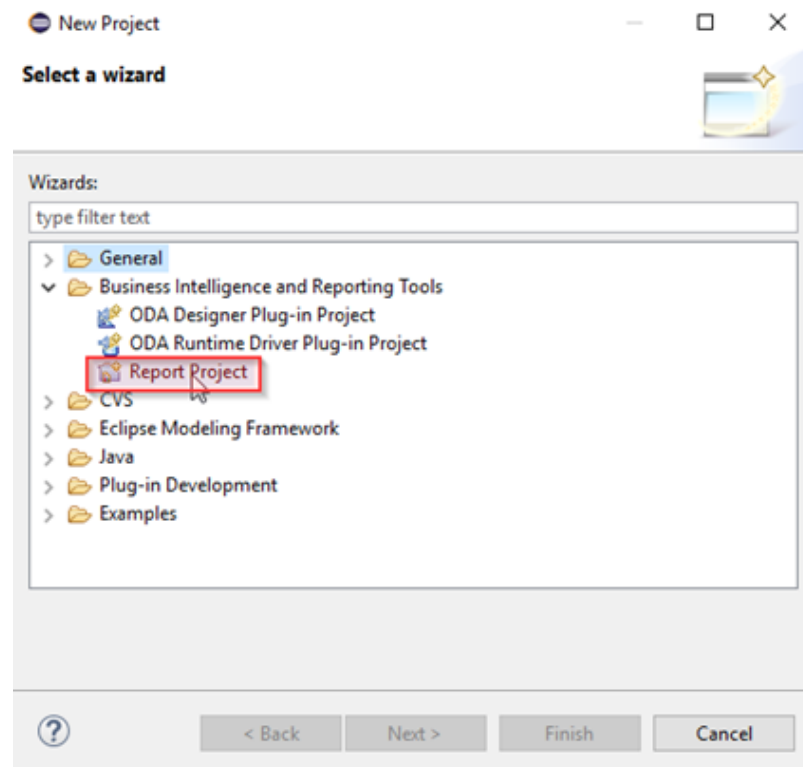
Für diese Schritte zu machen, gehen Sie in Eclipse nach 'About Eclipse Platform' und dann in 'Installation Details'. In das Fenster, gehen Sie zu dem Tab 'Plugins'. Hier ist die alle Liste von Plugins angezeigt. die Plugins von Tradui

können Sie suchen mit dem Begriff 'birtexperts'.



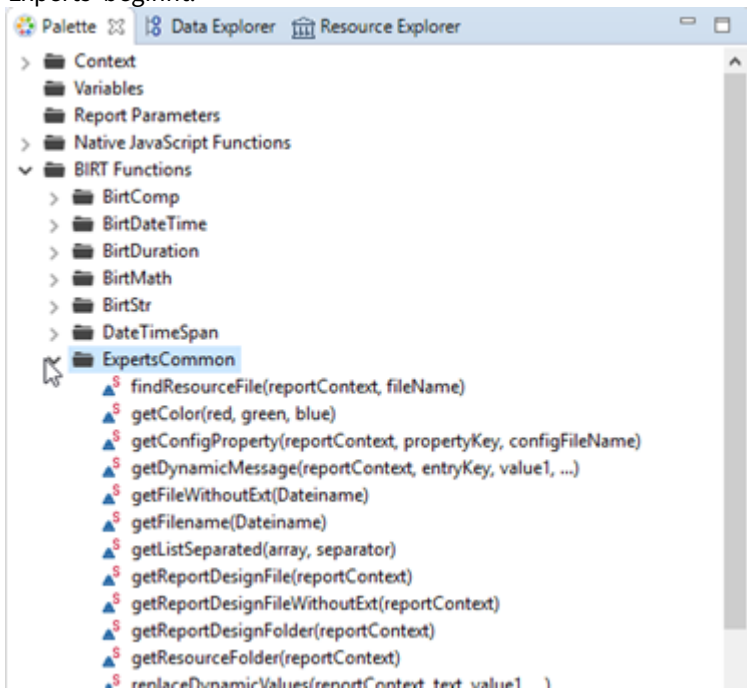
4. Neue Projekte erstellen

Für ein neue Projekte zu erstellen, gehen Sie bitte 'New Projekt' und dann 'Report Projekt' wählen.



5. Prüfen ob die Plugins korrekt geladen sind (andere Methode)

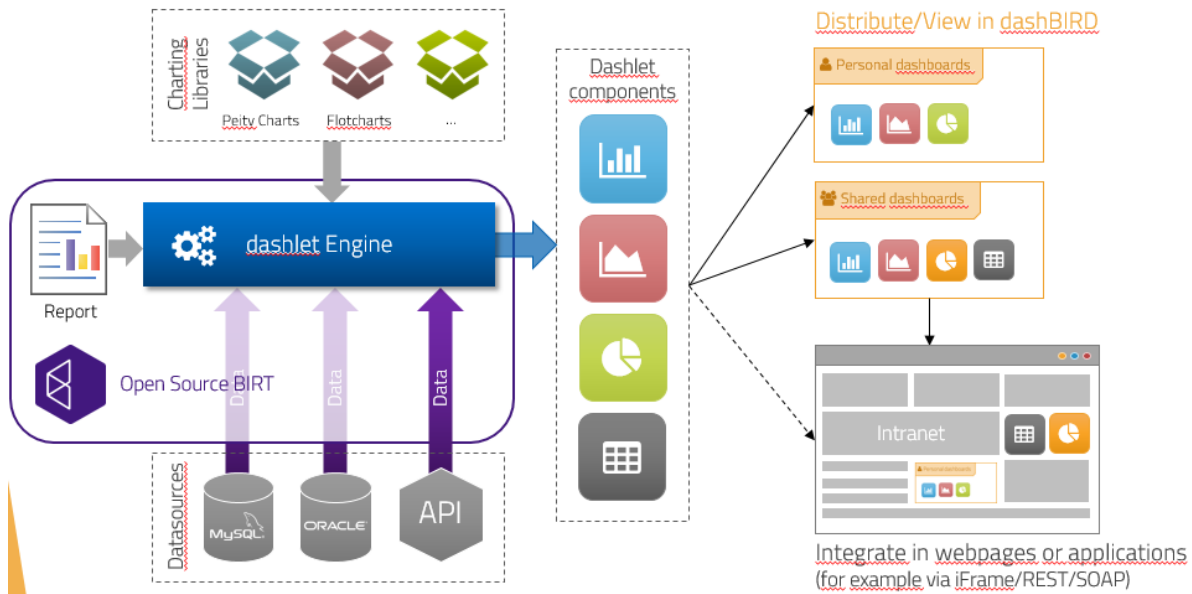
Sie können in die 'Palette', die Unter-Titel 'BIRT Functions' ausklappen. Hier sehen Sie manche Optionen, die mit 'Experts' beginnt.



2.3 Architektur und Funktionsweise

2.3.1 Einführung

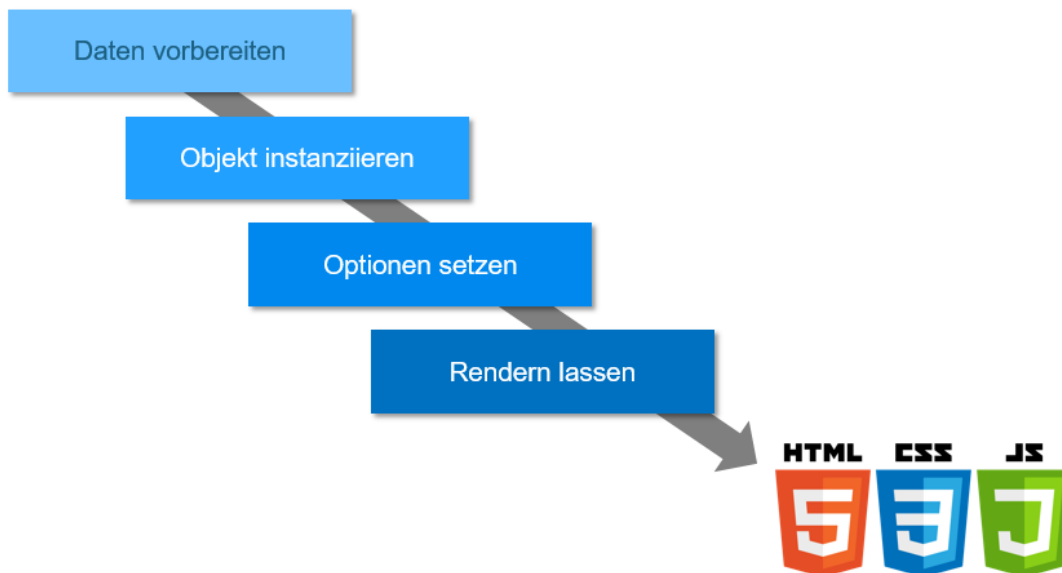
Das **dashBIRD** Architekturübersicht ist in das folgende Bild dargestellt.



2.4 Schnelleinstieg

2.4.1 Einführung

Das Workflow für ein **dashBIRD** zu erstellen ist wie dargestellt.



Die einzelne Schritt sind hier in Detail erklärt.

Daten vorbereiten

Die erste Schritt ist das Daten zu vorbereiten.

- Ein MemoryBuffer sollte gefüllt sein mit die Daten.
- Die Splate-Namen in der MemoryBuffer sollte angepasst sein (falls notwendig, z.b. für Pie-Charts, für Charts mit 'Series' Werte, usw).
- Die Splate, die nicht relevant sind sollte ausgeblendet sein.

Objekt instanzieren

Die DashletManager ist erstellt für die Dashlet Objekt zu instanzieren, und dann ist die einzelne Dashlets erstellt.

Dashlet erzeugen

```
// DashletManager erzeugen
ExpertsDashlet.createDashletManager(reportContext, "DashletManager");

// Dashlet erzeugen
var dashlet = DashletManager.addDashletEChart("line", "Wertpapier-Entwicklung", "Line
Chart", true, mbr);
```

Optionen setzen

Die Optionen für die Dashlets ist gesetzt mit der Function `addOption("Option", "value");`

Dashlet Optionen

```
//Option für Kurvendiagramm
dashlet.addOption("chart.series.smooth", "true");
// Ermöglicht es, alle Komponenten wie in einem "Stacked-chart" zu stapeln.
dashlet.addOption("chart.stacked", "all");
//Zeigt die Legende an
dashlet.addOption("chart.legend.show", "true");
```

Rendern lasen

Die Dashlets sind gerendert mit die folgende Code-Block

Render function

```
DashletManager.renderAll();
if (LogInfo.getRunOnServer() == false)
{
    DashletManager.getEngine().extractResources();
}
```

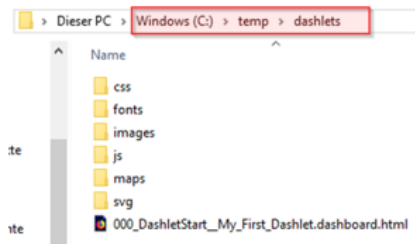
Output Datein

Die Ergebnisse (HTML Datei, CSS Datei oder JS Datei) ist in der Lokale Umgebung in 'C:/temp/dashlets' Verzeichnis gespeichert.

Wenn der Dashboard in der Server Umgebung ausgeführt ist, ist der Output-Datein in einem speziellen Job-Ordner gespeichert.



Lokale Umgebung



Server Umgebung

- ▶ Der Inhalt wird in einem speziellen Job-Ordner gespeichert

3 Daten und Parameter

3.1 Einführung

Es gibt verschiedene Klassifikationen von Daten, wie in das Bild dargestellt ist.

Quantitativ	Ordinal	Nominal
Numerische Werte	Numerische oder nicht numerische Werte	Nicht numerische Werte
Berechnungen möglich	Definierte Ordnung	Keine definierte Ordnung
Beispiele: Umsatz, Kennzahlen, Größe	Beispiele: Monate, Schulnoten, Steuerklassen	Beispiele: Geschlecht, Familienstand, Farben

Andere Klassifikationen von Daten ist wie folgende.

	Quantitativ	Ordinal
Interne Daten	<ul style="list-style-type: none"> •Tägliche Verkaufszahlen •Interne E-Mails •Tagesaktuelle Kosten •u.v.a 	<ul style="list-style-type: none"> •Geschäftskennzahlen •Jahresberichte •Kundenanalyse •u.v.a.
Externe Daten	<ul style="list-style-type: none"> •Neue Patente •Kundenanfragen •Aktienkurse •Kommerzielle Newsticker •u.v.a. 	<ul style="list-style-type: none"> •Marktanalysen •Branchenliteratur •Kommerzielle Datenquellen (Hoppenstedt, Reuters, etc.) •u.v.a.

3.2 Arten von Daten


3.2.1 Einführung

Die Verschiedene Datentypen ist weiter erklärt.

1-dimensional

Der erste Art von Daten ist 1-Dimensionale Daten. Es ist immer ein einfache Listen und is in ein single Dimension.

1D




ANZAHL
5844033
3439570
1915517
1744036
1003828
732251
450792

2-dimensional

Der nächste Art von Daten ist 2-Dimensionale Daten. Es ist immer ein einfache Tabellen Form und is in zwei Dimension.

2D

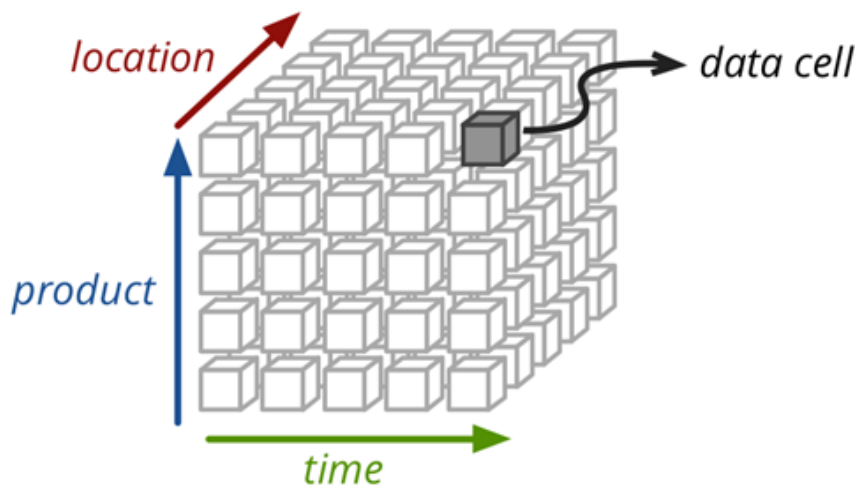


PRODUKTLINIE	MENGE	UMSATZ	GEWINN
Classic Cars	5844033	3853923.0	109085.0
Vintage Cars	3439570	1797560.0	51480.0
Motorcycles	1915517	1121427.0	31349.0
Planes	1744036	954638.0	26990.0
Trucks and Buses	1003828	1024114.0	28555.0
Ships	732251	663999.0	19050.0
Trains	450792	188533.0	5439.0

n-dimensional

Ein andere Art von Daten ist n-Dimensionale Daten. Es ist immer ein complexes Datenwürfel Form und is in multi-Dimension.

Es ist Üblicherweise als Gruppierungen dargestellt und hat ein Dimension, ein Fakt und ein Wert.



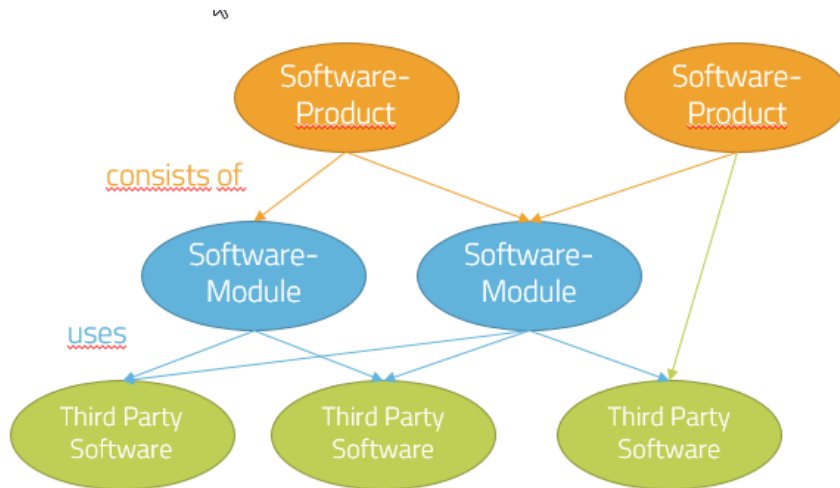
3D

LAND	KUNDE	UMSATZ
Australia	Anna's Decorations, Ltd	107800.0
Australia	Australian Collectables, Ltd	60300.0
Australia	Australian Collectors, Co.	117300.0
Australia	Australian Gift Network, Co	51600.0
Australia	Souvenirs And Things Co.	93300.0
Austria	Mini Auto Werke	45300.0
Austria	Salzburg Collectables	71700.0
Belgium	Petit Auto	79900.0
Belgium	Royale Belge	23500.0
Canada	Canadian Gift Exchange Network	90300.0
...		
USA	Toys4GrownUps.com	90700.0
USA	Vitachrome Inc.	76400.0
USA	West Coast Collectables Co.	55400.0

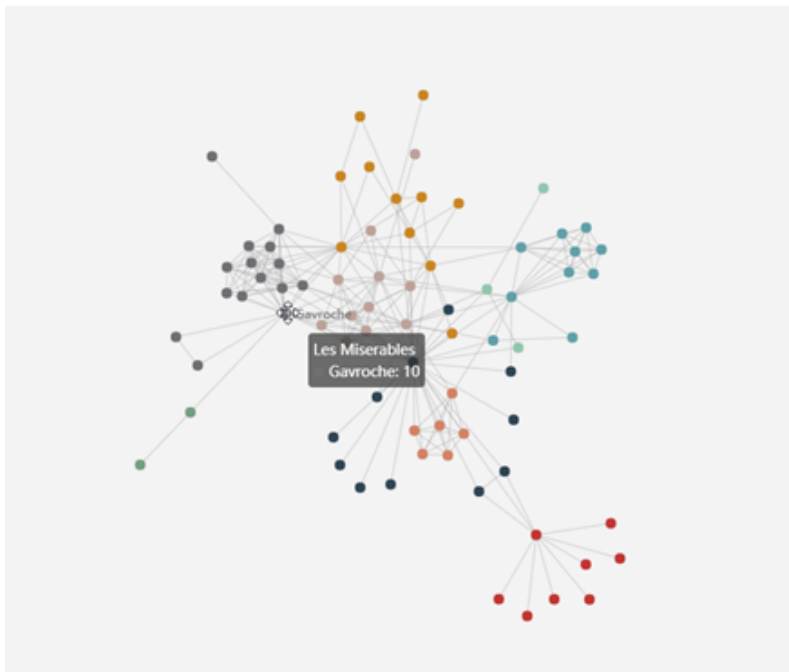
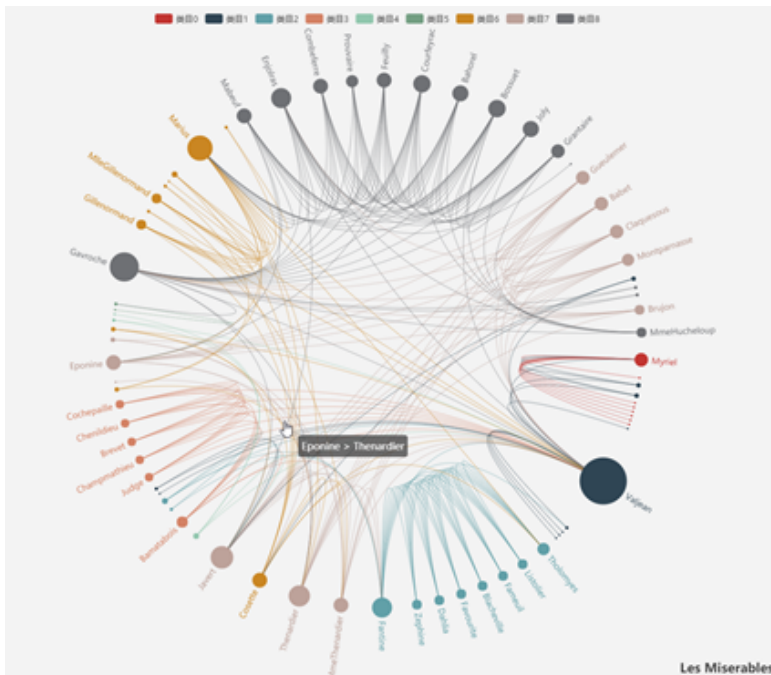
Graphenbasiert

Ein andere Art von Daten ist Graphenbasiert Daten. Es ist ein Hirachische oder relationale Daten und bestehend aus Knoten und Kanten. Die Korrelationen sollte definieret sein.

Es ist in Spezielle Datenbanken, z.B. "neo4j" benutzt und ist normalewiese als JSON- oder XML-orientiert.



Beispiele Charts mit diesen Datentypen ist folgende dargestellt in Echarts.



Vektorinformationen & Geodaten

Ein andere Art von Daten ist Vektorinformationen und Geodaten.

Diese Variante für Daten benutzen wir normalerweise mapping of Ländern/Städten. Es ist XML basiert, z.B. aus OpenStreetMaps.

Es enthält auch der GPS-Daten/Informationen oder der Geokodierte Daten, z.B. Adressen und gruppiert zu logischen Clustern. Es enthält auch SVG-Daten für Bereiche (areas).

Beispiele Charts mit diesen Datentypen ist folgende dargestellt.



3.3 Daten in das richtige Format bringen

3.3.1 Einführung

Es gibt verschiedene Schritte sodass die Daten in das richtige Format für das Dashlet vorbereitet sind.

Einige der Optionen sind unten aufgeführt.

- Memory Cube oder SQL Funktionen wie „pivot“
- Spalten Namen ändern (z.b. für Pie Charts, Radar Charts, Geo Charts, usw).
- MetaType setzen (z.b. Geo Charts)
- Nicht relevante Spalten ausblenden

3.4 Datenstrategien

3.4.1 Einführung

Der gibt verschiedene Möglichkeit der Output-Dateien zu speichern

Default Variant

HTML, CSS, JS & Daten werden in einem Dokument gespeichert

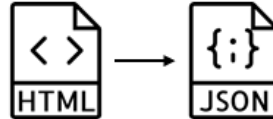


+ Leichterer Aufbau und alle Ressource sind gebündelt

- Datei kann aufgeblasen werden

„Excluded Variant“

HTML, CSS & JS -> Daten ist ausgelagert in ein Datei

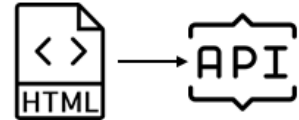


+ Minimale Dateigröße (schnellere Response-Time), Daten werden asynchron geladen

- Rewrite-Effort beim Speichern in ein Archiv/Snapshot

Webservice Variant

HTML, CSS & JS -> Daten werden von einem externen Webservice zur Verfügung gestellt



+ Minimale Dateigröße (schnellere Response-Time), Daten werden asynchron geladen

- Kein archiving/snapshots Möglichkeit

Default Variant

In diesem Fall, die HTML, CSS, JS und Daten sind in ein HTML Dokument gespeichert. Die Vorteil ist das es hat leichter Aufbau weil alle Ressourcen sind gebündelt. Aber der Nachteil ist das der HTML Datei kann sehr groß werden und deswegen aufgeblasen werden.

Excluded Variant

In diesem Variant, die HTML, CSS und JS Information sind in ein HTML Dokument gespeichert, während das Daten ist ausgelagert in ein externe Datei. Die Vorteil ist das der HTML Datei hat ein Minimale Dateigröße und deswegen ein schnellere Response-Time. Die Daten werden asynchron geladen. Der Nachteil ist ein Rewrite-Aufwand beim Speichern in ein Archiv/snapshot.

Webservice Variant

In die dritte Variant, die HTML, CSS und JS Information sind in ein HTML Dokument gespeichert, während das Daten ist von einem externen Webservice zur Verfügung gestellt. Die Vorteil ist gleich wie die 'Excluded Variant' und der Nachteil ist kein archiving oder snapshots sind möglich.



Tipp

Denken Sie daran:

- Browser/Clients sind heute in der Lage, Datenmengen zu verarbeiten...
- ABER: Es ist nie eine gute Idee 4 Mio. Datenzeilen aus der Datenbank in den Browser zu laden!

Best Practice:

- Versuchen Sie, Daten auf der ersten Ebene zu aggregieren (Datenbankschema/Sql-Abfragen)
- Stellen Sie detaillierte Daten (per Drilldown) nur dann zur Verfügung, wenn dies notwendig ist und die Datenmenge praktikable ist, oder
- Stellen Sie detaillierte Daten auf Anfrage zur Verfügung (Laden nach Klick)
- Verwenden Sie Caching-Mechanismen

3.5 Datenquellen

3.5.1 Einführung

BIRT unterstützt eine Vielzahl von Datenquellen:

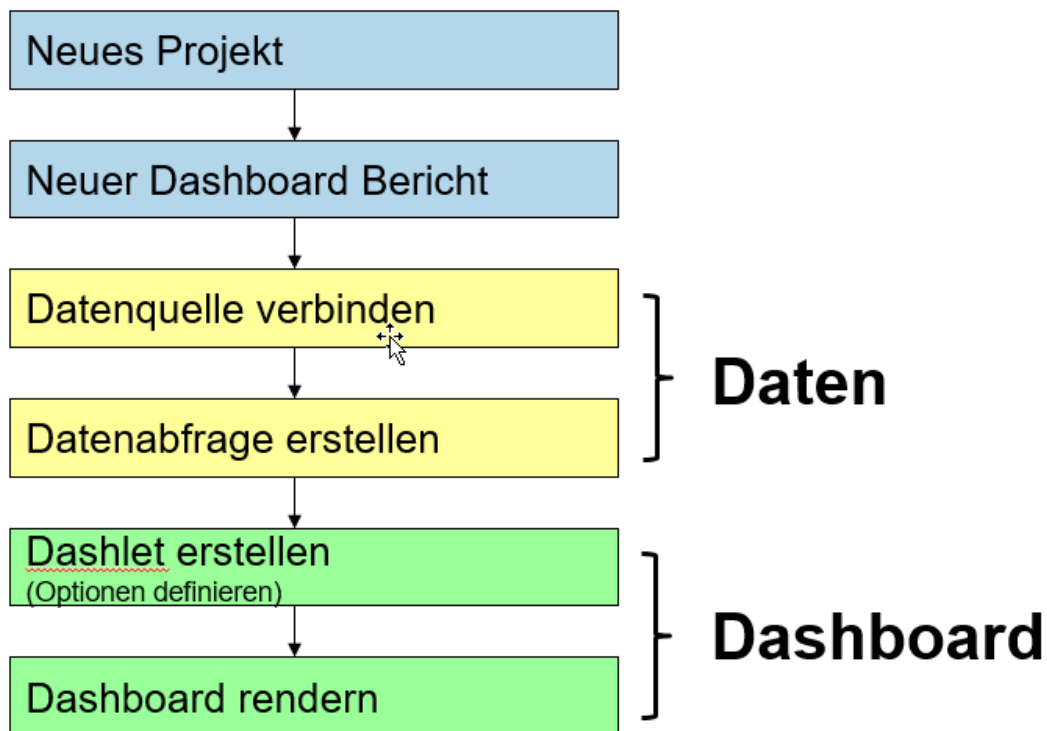
- JDBC-Datenquellen
- XML-Dateien
- Web Services
- Java Beans
- Flat Files
- OLAP-Cubes
- ...

4 Dashlets erstellen

4.1 Fortsetzung Schnellstart

4.1.1 Einführung

Der Workflow für ein Dashboard zu erstellen ist als folgendes:



Mehr Informationen zu der einzelnen Schritt, finden Sie hier: [Schnelleinstieg](#)

4.2 Fehlerbehandlung

4.2.1 Einführung

Die Fehlerbehandlung in Dashboards kann der Entwickler mit die folgende Optionen machen.

Try-catch Blocks

Verwenden try-catch Blocks in Report und bei Custom Skripte.

onFetch() Method on DataSet (JavaScript)

```
1 try { mbrUmsatz.addRowToBuffer(row) } catch (e) {};
```

Output Log

An ende die HTML Dashboard ist die Log angezeigt. Hier ist manche Fehler aufgelistet (z.b. Unique Name für alle Dashlets).

```
Engine Log:
06.08.2018 13:47:47:181 INFO-ENGINE DashletEngine: - Engine start rending all!
06.08.2018 13:47:47:181 INFO-ENGINE DashletEngine: - Rendering start time: 06.08.2018 13:47:47:181
06.08.2018 13:47:47:181 INFO-ENGINE DashletEngine: - Dashlet document file 'c:\temp\dashlets\000_DashletStart_My_First_Dashlet.dashboard.html'
06.08.2018 13:47:47:181 INFO-ENGINE DashletEngine:Dash1 - Render html dashlet 'Dash1'
06.08.2018 13:47:47:197 INFO-ENGINE DashletPage: - Render script dashlet footer!
06.08.2018 13:47:47:197 INFO-ENGINE DashletPage:Dash1 - Render java script
06.08.2018 13:47:47:197 INFO-ENGINE DashletPage:Dash1 - Render user options
06.08.2018 13:47:47:197 INFO-ENGINE DashletPage: - Engine finish rending all!
06.08.2018 13:47:47:197 INFO-ENGINE DashletPage: - Rendering finish time: 06.08.2018 13:47:47:197
06.08.2018 13:47:47:197 INFO-ENGINE DashletPage: - Render time total: 0 min 0 sec 16 msec
```

Mit die folgende Option in Dashboard, kann der Log mit mehr detaillierte Information angezeigt werden.

```
// DashletManager initialisieren
ExpertsDashlet.createDashletManager(reportContext, "DashletManager", "Erste
Dashboard");

// Logging Option
DashletManager.getEngine().getLogger().addLogType("ALL");
```

```
Engine Log:
07.06.2021 13:22:29:413 INFO-ENGINE DashletEngine: - Engine start rending all!
07.06.2021 13:22:29:413 INFO-ENGINE DashletEngine: - Rendering start time: 07.06.2021 13:22:29:413
07.06.2021 13:22:29:413 INFO-ENGINE DashletEngine: - Dashlet document file 'c:\temp\dashlets\Dashlets-EChart_KPI.html'
07.06.2021 13:22:29:413 INFO-ENGINE DashletEngine:KpiBox1 - Render html dashlet 'KpiBox1'
07.06.2021 13:22:29:413 DEBUG-OPTIONS DashletKpiBox:KpiBox1 - List of options
07.06.2021 13:22:29:413 DEBUG-OPTIONS DashletKpiBox:KpiBox1 - 'kpi.box.count': 5
07.06.2021 13:22:29:413 DEBUG-OPTIONS DashletKpiBox:KpiBox1 - 'kpi.box.i[0].responsive': 2-4-6-8
07.06.2021 13:22:29:413 DEBUG-OPTIONS DashletKpiBox:KpiBox1 - 'kpi.box.i[0].type': trendbox
07.06.2021 13:22:29:413 DEBUG-OPTIONS DashletKpiBox:KpiBox1 - 'kpi.box.i[0].value': E 139
07.06.2021 13:22:29:413 DEBUG-OPTIONS DashletKpiBox:KpiBox1 - 'kpi.box.i[0].footer': Gesamtanzahl
07.06.2021 13:22:29:413 DEBUG-OPTIONS DashletKpiBox:KpiBox1 - 'kpi.box.i[0].title': All Years
07.06.2021 13:22:29:413 DEBUG-OPTIONS DashletKpiBox:KpiBox1 - 'kpi.box.i[0].trendvalue': 1
07.06.2021 13:22:29:413 DEBUG-OPTIONS DashletKpiBox:KpiBox1 - 'kpi.box.i[0].trendvalueformat': #0.0%
07.06.2021 13:22:29:413 DEBUG-OPTIONS DashletKpiBox:KpiBox1 - 'kpi.box.i[0].icon': fa fa-heart
07.06.2021 13:22:29:413 DEBUG-OPTIONS DashletKpiBox:KpiBox1 - 'kpi.box.i[1].responsive': 2-4-6-8
07.06.2021 13:22:29:413 DEBUG-OPTIONS DashletKpiBox:KpiBox1 - 'kpi.box.i[1].type': trendbox
07.06.2021 13:22:29:413 DEBUG-OPTIONS DashletKpiBox:KpiBox1 - 'kpi.box.i[1].value': Ø 18%
07.06.2021 13:22:29:413 DEBUG-OPTIONS DashletKpiBox:KpiBox1 - 'kpi.box.i[1].footer': Absolute 25
07.06.2021 13:22:29:413 DEBUG-OPTIONS DashletKpiBox:KpiBox1 - 'kpi.box.i[1].title': 2012
07.06.2021 13:22:29:413 DEBUG-OPTIONS DashletKpiBox:KpiBox1 - 'kpi.box.i[1].trendvalue': 1
```

Browser Console (F12-Key)

Die Browser-Konsole kann mit der F12 Key angezeigt werden. Hier ist die HTML Fehler angezeigt.

```

Uncaught TypeError: this._connected[r].connectedEventHandler is not a function
    at i.__onevent (echarts.js:1)
    at i.__onevent (echarts.js:1)
    at t.e.dispatchWithContext (echarts.js:8)
    at t.messageCenter.dispatch (echarts.js:1)
    at t.__onmousemove (echarts.js:4)
    at U.l.__onmousemove (echarts.js:4)
    at U.e.dispatch (echarts.js:8)
    at U.e.dispatchAgency (echarts.js:8)
    at U.mousemove (echarts.js:8)
    at HTMLDivElement.<anonymous> (echarts.js:8)

```

4.3 Anatomie eines Dashlets

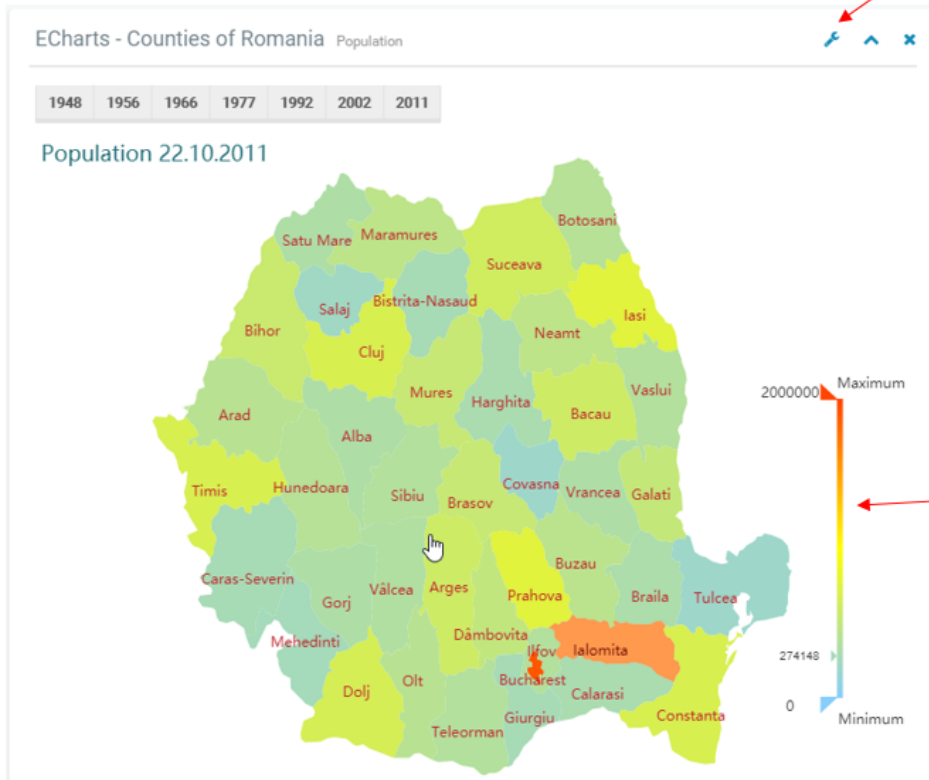
4.3.1 Einführung

Jede Dashlet hat die folgende Anatomie und jede Dashlet ist in einzelnen Box eingepackt.

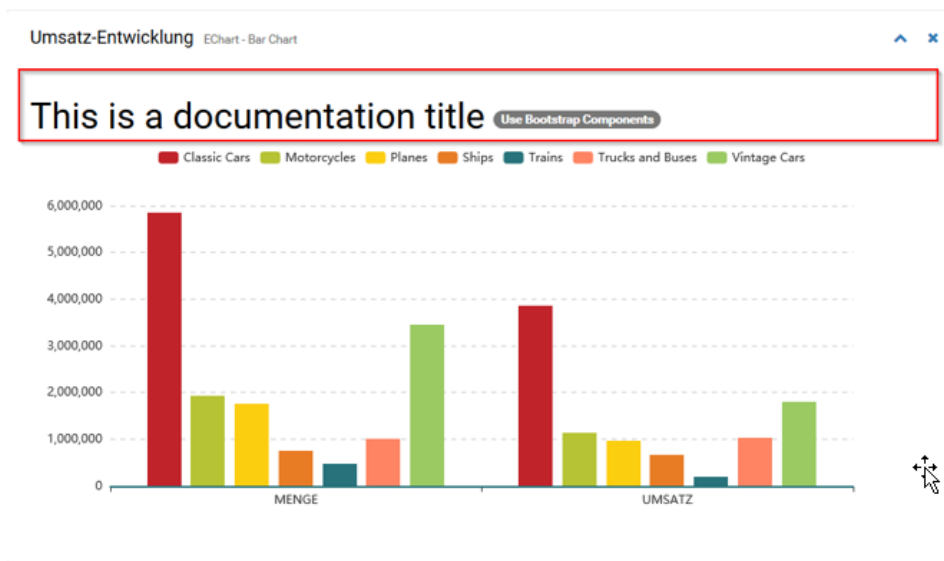


Das Box kann mehr Optionen haben (z.b. InfoBox, Excel Download funktionalität).

**Additional documents and
documentation**



Es gibt auch die Möglichkeit verschiedene HTML Content zu anzeigen



HTML Content

4.3.2 Erklärung

Box Area

Box Titel

Der Titel des Boxes ist hier angezeigt.

Box-Subtitel

Der Unter-Titel (Subtitel) des Boxes ist hier angezeigt.

Weitere Box Optionen

Jede Box hat weitere Optionen wie

- Ausklappen und Einklappen der Box
- Schließen der Box
- Weitere Dokumentaion hinzufügen (z.b. Popup oder Modal Dialog)
- Weitere Externe-Dokumente herunterladen möglichkeit (z.b. Native Excel Download)

Top / Bottom Button Bar

Hier gibt die Möglichkeit Buttons für die Charts zu anzeigen oder der Multiselect Boxes zu anzeigen.

Top / Bottom HTML Content

Hier gibt die Möglichkeit Verscheidene HTML Content zu anzeigen.

Chart Area

Chart Titel

Der Titel des Charts ist hier angezeigt.

Chart-Subtitel

Der Unter-Titel (Subtitel) des Charts ist hier angezeigt.

Toolbox Optionen

Verschiedene Toolbox Optionen sind hier angezeigt (z.b. Bar/ Stacked-Bar Anzeige, Line/Stacked-Line Anzeige, Legende aus-und einblenden, usw).

Legend Liste

Die Liste der Legende ist hier angezeigt.

Weitere Chart Optionen

Jede Chart hat weitere Optionen wie

- Range Selector anzeige
- DataZoom Selektor anzeige



Tipp

- Jede Dashlet sollte ein Unique ID haben.
- Die ID kann der Entwickler Manuel setzen oder es ist von der Plugin erstellt mit ein Incrementing Nummer.
- Diese ID von ein Dashlet benötigt für die Interaktivität in der Dashlets.

4.4 Charting Engines

4.4.1 Einführung

dashBIRD implementiert die folgende Charting Engines

- ECharts
- C3 Charts
- PivotTable
- jquery DataTables
- FlotCharts
- Piety- Sparklines
- OpenStreetMaps (Leaflet)
- Custom Components



C3.js D3-based reusable chart library

PivotTable.js

jQuery DataTables

OpenStreetMaps (Leaflet)

Custom Components

Peity - Sparklines

FlotCharts

OrgCharts

Die Version für die einzelne Engines sind in der folgende Tabelle dargestellt.

Nr.	Platform	Version
1	Nicescroll	3.2.0
2	Peity jQuery	3.2.0
3	ECharts5	5.1.0
4	D3 Charts	3.5.17
5	DataTables	1.10.18
5a	JSZip	2.5.0
5b	pdfmake	0.1.36
5c	Buttons	1.5.2
5d	Column visibility	1.5.2
5f	HTML5 export	1.5.2
5g	Print view	1.5.2
5h	ColReorder	1.5.0
5i	FixedColumns	3.2.5
5j	FixedHeader	3.1.4
5k	KeyTable	2.4.0
5l	Responsive	2.2.2

5m	RowGroup	1.0.3
5n	RowReorder	1.2.4
5o	Scroller	1.5.0
5p	Select	1.2.6
6	Moment.js	2.9.0
7	C3 Charts	0.4.11
8	FlotChart	0.8.3
9	Time	1.0
10	Canvas	1.0
11	Pace	1.0.0
12	html2canvas	1.0.0-rc.5
13	Chart.js	2.0.0-beta2
14	OrgChart	2.1.9
15	Sparkline	jquery.sparkline 2.1.2
16	Leaflet	Leaflet 1.1.0+Detached
17	bootstrap-progressbar	v0.8.4
18	iCheck	v1.0.2
19	Bootstrap	v3.3.7
20	PivotTable	Version 2.23.0
21	Bootstrap Multiselect	(?)
22	PivotTable - Subtotal	V1 11.0

4.5 Chart-/Dashlettypen

4.5.1 3D Weltkarte

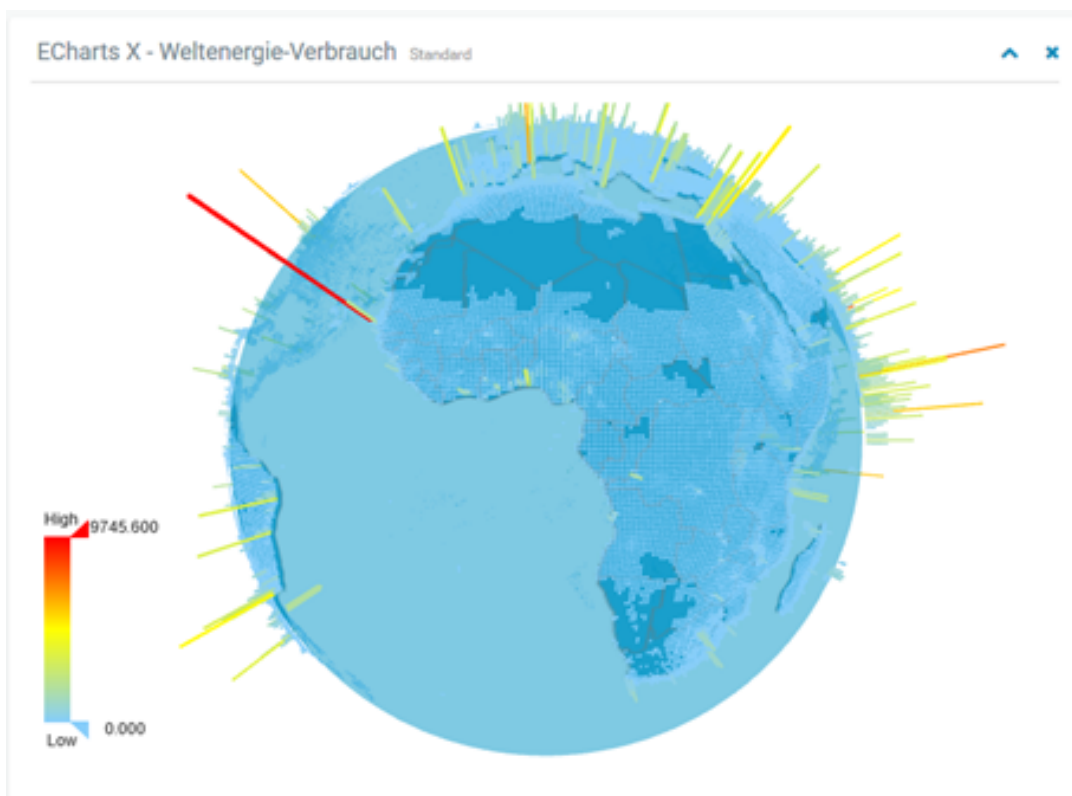


Information

Dieser Charttyp ist derzeit noch nicht im Dashlet-Plugin verfügbar. Bitte kontaktieren Sie unseren [Support](#), wenn Sie an der Nutzung dieser Darstellungsform interessiert sind.

Einführung

Die 3D Weltkarte ist eine EChart Erweiterung. Sie ist gleich wie das Kartendiagramm, nur in 3D.



4.5.2 Balkendiagramm

Einführung

Das Balkendiagramm Dashlet könnte mit das C3Charts oder das eCharts platform entwickelt werden.

Variant 1: ECharts

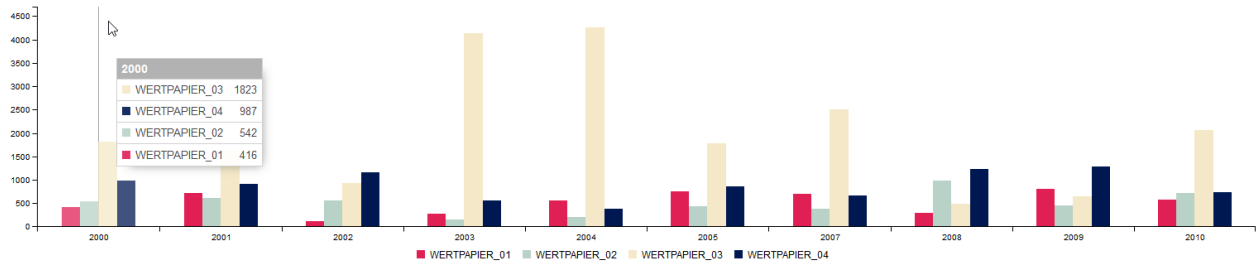
Beispiel



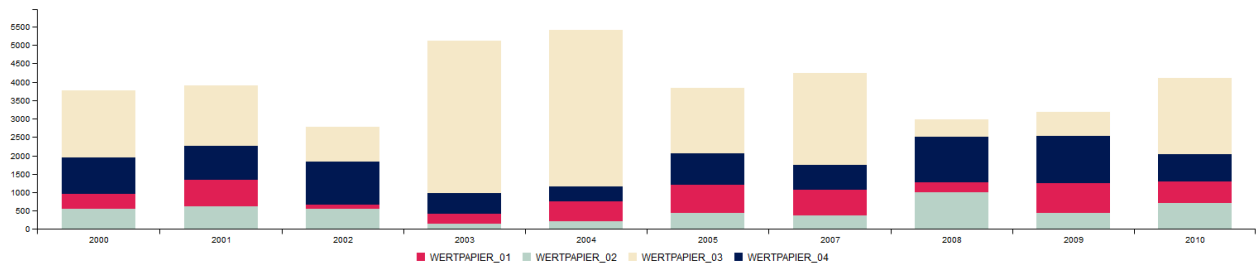
Variant 2:C3 Charts

Beispiel

Wertpapier-Entwicklung C3 Chart - Bar Chart



Wertpapier-Entwicklung C3 Chart - Stacked Bar Chart



Dashlets erzeugen

Echart Chart

Die Balkendiagramm Dashlet ist mit die folgende Code Block erstellt.

EChart Dashlet erzeugen

```
// DashletManager erzeugen
ExpertsDashlet.createDashletManager(reportContext, "DashletManager");

// Series Spalte definieren
mbr.getColumnDefinition("ORDERYEAR").setDisplayNames("Series");

// Dashlet erzeugen
var dashlet = DashletManager.addDashletEChart("bar", "Wertpapier-Entwicklung", "C3
Chart-Bar Chart", true, mbr);
```

C3 Chart

Die Balkendiagramm Dashlet ist mit die folgende Code Block erstellt.

EChart Dashlet erzeugen

```
// DashletManager erzeugen
ExpertsDashlet.createDashletManager(reportContext, "DashletManager");

// Series Spalte definieren
mbr.getColumnDefinition("ORDERYEAR").setDisplayNames("Series");

// Dashlet erzeugen
var dashlet = DashletManager.addDashletC3Chart("line", "Wertpapier-Entwicklung", "C3
Chart-Bar Chart", true, mbr);
```

Dashlet Optionen

Durch das Hinzufügen von Optionen können Dashlets individuell angepasst werden.

Dashlet Optionen

```
// Ermöglicht es, alle Komponenten wie in einem "Stacked-chart" zu stapeln.
dashlet.addOption("chart.stacked", "all");
//Zeigt die Legende an
dashlet.addOption("chart.legend.show", "true");
```

FAQ

Wie kann der Tooltip bei ein gestappte BalkenDiagram angepasst werden?

Der Tooltip für ein gestapptes Balkendiagramm kann mit die folgende Optionen angepasst werden.

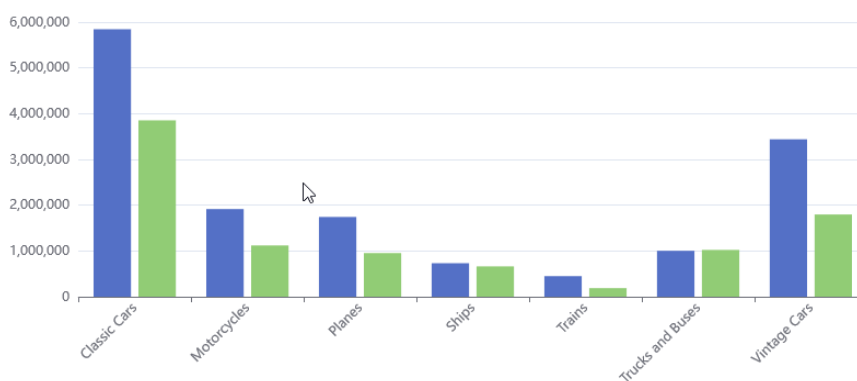
```
// the tooltip displays only the values of selected legend elements.
dashlet.addOption("chart.tooltip.show" , "true");
dashlet.addOption("chart.tooltip.trigger" , "axis");
dashlet.addOption("chart.tooltip.formatter.object", function(params, ticket,
callback)
{
    var test = "";
    test += params[0].axisValueLabel + "<br>";
    for(var i = 0; i < params.length; i++)
    {
        if(params[i].value != 0)
        {
            test += "<svg height=\"7\" width=\"7\"> <rect width=\"5\"
height=\"5\" style=\"fill:" + params[i].color + ";stroke-width:3;\" /></svg> ";
            test += params[i].seriesName + ": " + params[i].value + "<br>";
        }
    }
    return test;
});
```

Wie kann der X-Axis Label angepasst werden?

Der X-Axis Label kann mit die folgende Optionen angepasst werden.

```
dashlet.addOption("chart.xAxis.axisLabel.rotate", "45"); // x-Axis Label
anpassen
dashlet.addOption("chart.grid.height" , "250px"); // Höhe des Grids anpassen
dashlet.addOption("chart.grid.bottom" , "50px"); // Abstand zwischen des
Grids und der Box-unten
```

Umsatz-Entwicklung EChart - Line Chart



Wie kann der Farbe für ein Series angepasst werden?

Der Farbe für ein Series (Gruppe) kann mit der folgende Option angepasst werden.

```
// Mit Series Name (z.b. "Umsatz")  
dashlet.addOption("chart.series.n[Umsatz].itemStyle.color", "#FF0000");
```

Verschiedene Anzeige Formate sind möglich für diese Dashlet (siehe [ECharts \(4\) \[Options\]](#)).

Für mehr Information für C3 Charts, siehe [C3-Charts](#).

4.5.3 Baumdiagramm (Tree Diagram)



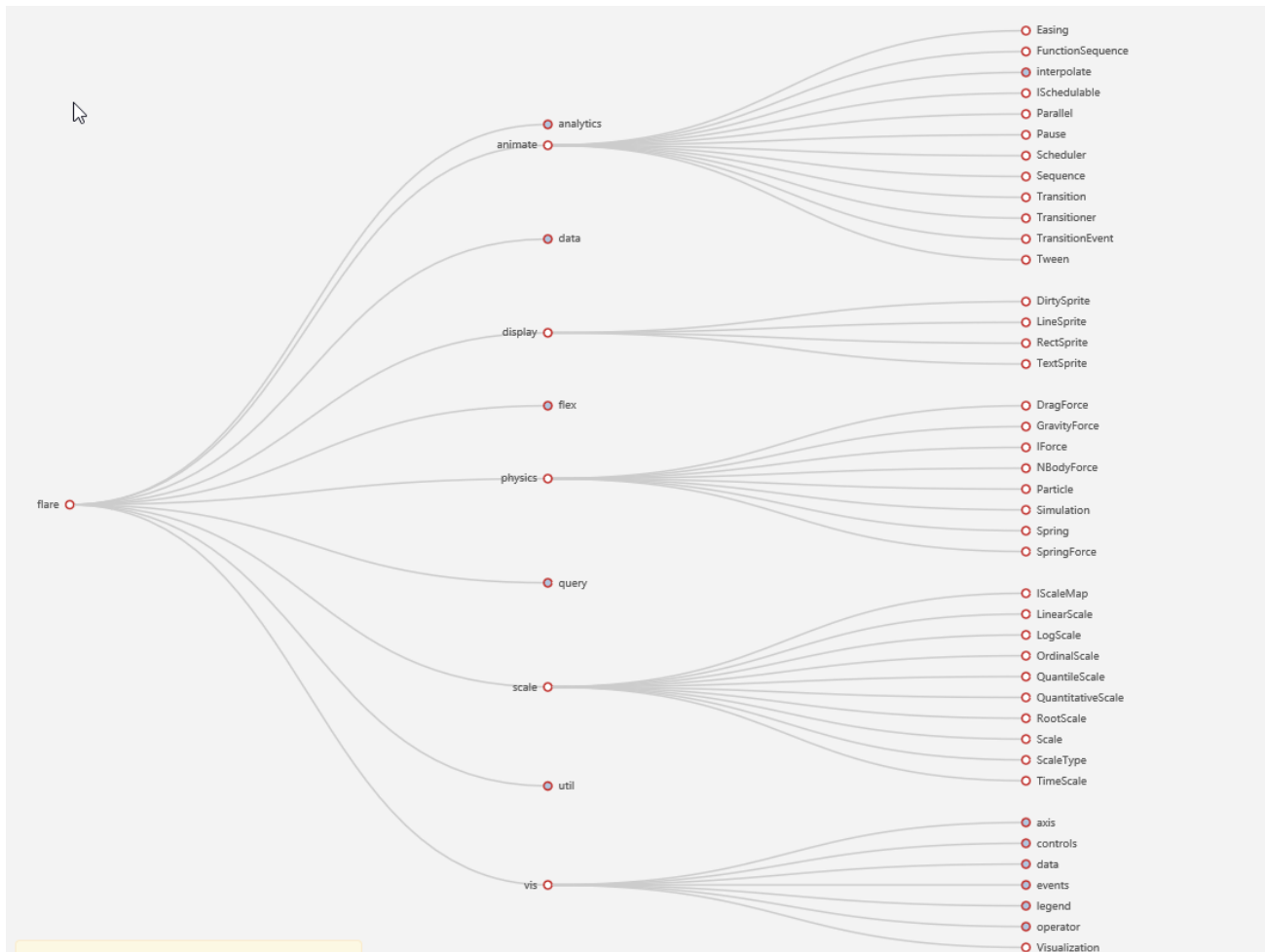
Information

Dieser Charttyp ist derzeit noch nicht im Dashlet-Plugin verfügbar. Bitte kontaktieren Sie unseren [Support](#), wenn Sie an der Nutzung dieser Darstellungsform interessiert sind.

Einführung

Das Baumdiagramm (Tree Diagram) wird hauptsächlich zur Visualisierung der Baumdatenstruktur verwendet.

Es wird die Abhängigkeiten darstellen. Die Dateneingabe sollte ein JSON oder XML sein.



4.5.4 Blasen-/Kuchendiagramm (Bubble Pie)

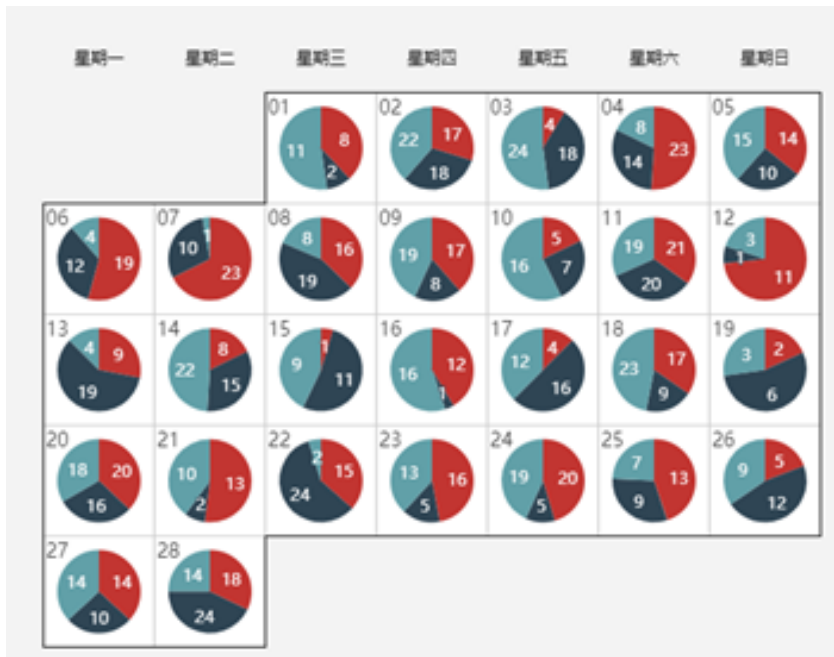


Information

Dieser Charttyp ist derzeit noch nicht im Dashlet-Plugin verfügbar. Bitte kontaktieren Sie unseren [Support](#), wenn Sie an der Nutzung dieser Darstellungsform interessiert sind.

Einführung

Diese Chart-Typ ist eine Kombination aus Blasendiagramm und Tortendiagramm. Es ist nützlich, wenn Sie das Verhältnis auf 3-Dimensionen anzeigen möchten.



4.5.5 Blasendiagramm (Bubble Chart)

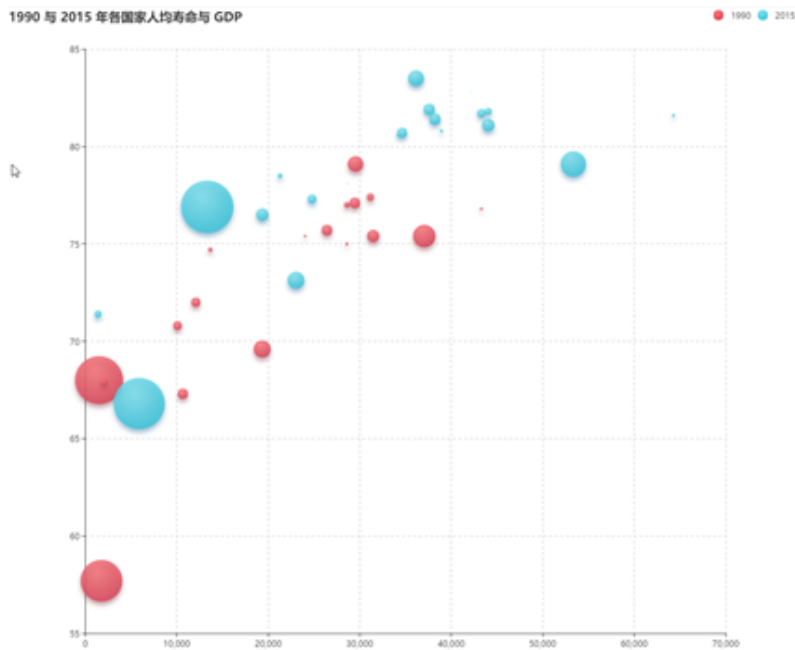


Information

Dieser Charttyp ist derzeit noch nicht im Dashlet-Plugin verfügbar. Bitte kontaktieren Sie unseren [Support](#), wenn Sie an der Nutzung dieser Darstellungsform interessiert sind.

Einführung

Der Blasendiagramm (Bubble Chart) ist ein Streudiagramm (scatter chart) mit gewichtetem Wert. Es ist nützlich, wenn man 3-dimensionale Werte hat.



4.5.6 Box-Plot

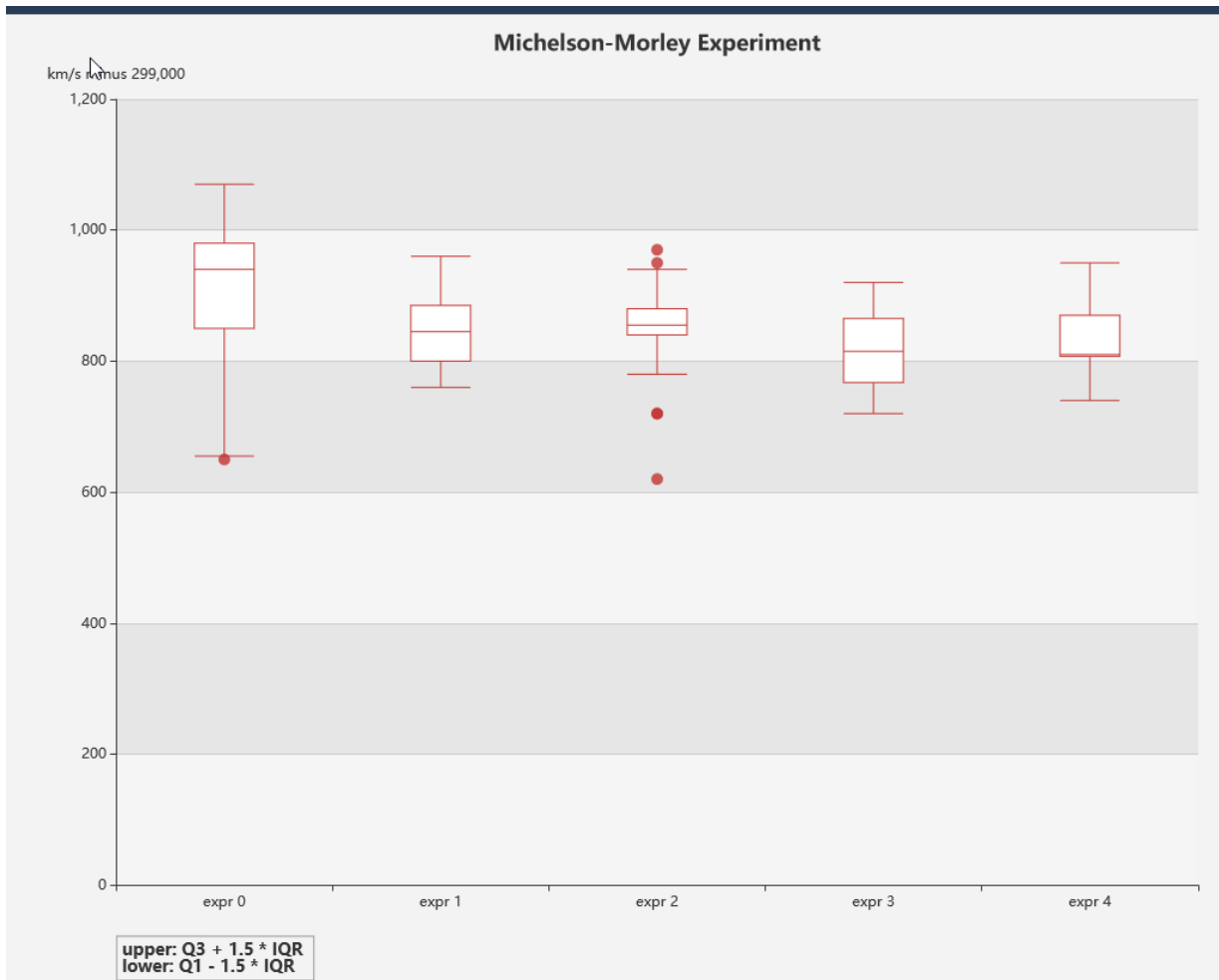


Information

Dieser Charttyp ist derzeit noch nicht im Dashlet-Plugin verfügbar. Bitte kontaktieren Sie unseren [Support](#), wenn Sie an der Nutzung dieser Darstellungsform interessiert sind.

Einführung

Es wird oft für statistische Visualisierungen und bei der Erkennung von Anomalien (maschinelles Lernen) verwendet. Es ist ähnlich wie ein Candlestick Diagramm und zeigt Median, Quartil, Bereich, Mittelbereich, Ausreißer



4.5.7 Data Dashlet

Einführung

Das Data-Dashlet kann die Daten von ein MemoryBuffer in ein JSON-Format in die Dashboard renderen.

Es ist benutzt wenn die Filter-Condition verwendet wird, zb. wie in die folgende Use-case

Ein Dashboard enthält KPI-Wert und auch ein Bar-Chart. Beim ein Click auf ein Bar in der Bar-Chart, sollte die KPI-Werte aktualisiert sein.

Dashlets erzeugen

Die Linien-/Kurvendiagramm Dynamische Tabellen

EChart Dashlet erzeugen

```
// DashletManager erzeugen
ExpertsDashlet.createDashletManager(reportContext, "DashletManager");

// Dashlet erzeugen
var dashlet = DashletManager.addDashlet("DashletData", "", "", true, mbr);
```

Dashlet Optionen

Durch das Hinzufügen von Optionen können Dashlets individuell angepasst werden.

Dashlet Optionen

```
// set Id for the Dashlet
dashlet.setId("dataDashletAUD_KPI");

// option to make the chart small so that no space is occupied for this dashlet on
the dashbaord screen.
dashlet.addOption("chart.height","0px");

// option to render the memory buffer contents in a JSON array within the HTML file
dashlet.addOption("data.script.mode","table");
```



Tipp

Der JSON-Data-Dashlet ist im HTML-Datei gerendert mit der Name als <dashletId>_table1

4.5.8 Dynamische Tabellen (DataTables)

Einführung

Das Dynamische Tabellen Dashlet könnte mit das Dynamic Table platform entwickelt werden.

Dynamische Datentabelle bietet die möglichkeit mit mehreren Funktionen an, wie:

- Sortierung
- Filterung (Suche)
- Paginierung (Kontrolle der Menge der angezeigten Zeilen)
- Exportieren nach CSV, Excel, PDF, ...
- Reaktionsfähiger Spaltenumbruch mit Anzeige von Unterelementen
- Spaltenreihenfolge

Details ⬇ ⬆ ✕

CSV EXCEL

Search:

CUSTOMERNUMBER	CUSTOMERNAME	ORDERNUMBER	ORDERDATE	QUANTITYORDERED	PRICEEACH	COUNTRY	PRODUCTLINE	PRODUCTSCALE	PRODUCTVENDOR	CC_UMSATZ
124	Mini Gifts Distributors Ltd.	10142	2011-08-08	24	122.08	USA	Classic Cars	1:18	Min Lin Diecast	2929.92
124	Mini Gifts Distributors Ltd.	10142	2011-08-08	49	74.35	USA	Vintage Cars	1:24	Min Lin Diecast	3643.15
205	Toys4GrownUps.com	10145	2011-08-25	45	76.56	USA	Motorcycles	1:10	Min Lin Diecast	3445.2
320	Mini Creations Ltd.	10143	2011-08-10	46	70.54	USA	Ships	1:18	Min Lin Diecast	3244.84
320	Mini Creations Ltd.	10143	2011-08-10	36	86.77	USA	Planes	1:700	Min Lin Diecast	3123.72

Showing 1 to 5 of 5 entries

PREVIOUS 1 NEXT

Dashlets erzeugen

Die Linien-/Kurvendiagramm Dynamische Tabellen

EChart Dashlet erzeugen

```
// DashletManager erzeugen
ExpertsDashlet.createDashletManager(reportContext, "DashletManager");

// Dashlet erzeugen
var dashlet = DashletManager.addDashletTableDynamic("Details", "", true, mbr);
```

Dashlet Optionen

Durch das Hinzufügen von Optionen können Dashlets individuell angepasst werden.

Dashlet Optionen

```
//styles setzen
dashlet.addOption("table.style", "table table-striped table-bordered stripe compact row-borders");

// splate ausblenden
dashlet.addOption("table.columnDef.t[0,11].visible", "false");
```

Verscheide Anzeige Format sind möglich für diese Dashlet (siehe [DataTables \[Options\]](#))

FAQ (DataTable)

Anzeige Anpassungen

Wie kann die Anzahl des Data-Records in eine Seite angepasst werden soll?

Die Anzahl des Data-Records kann mit die folgende Option angepasst werden.

```
dashlet.addOption("table.feature.paging", "true"); // Paging Option darstellen
dashlet.addOption("table.pageLength", "12"); // Anzahl der Zeile in jede Page
```

Wie kann der Spalten Sorting-Option angepasst werden?

Der Spalten Sorting Option kann mit die folgende Option angepasst werden.

```
dashlet.addOption("table.feature.ordering", "true"); // Sorting funtion ein-
ausblenden
```

Wie kann Spalten drag-und-drop aktiviert werden?

Die Option für aktivierung des Drag-und-Drop Options des Spaltens ist folgendes:

```
// reorder columns
// tableDashlet1.addOption("table.colReorder.enable" , "true");
```

Wie kann ein Zeile (Row) Drag-und-Drop Option aktiviert werden?

Die Option für aktivierung des Drag-und-Drop Options des Rows ist folgendes:

```
// Verschiebbarkeit der zZeilen aktivieren
tableDashlet1.addOption("table.rowReorder", "true");
```

Wie kann die Zielen Drag-und-Drop Option nur in vertikal-Richtung gesetzt werden?te

Die Drag-und-Drop eines Zieles kann nur in die Vertikale-Richtung mit die folgende Option gesetzt werden.

```
// Einrastfunktion, damit die Zeile nicht nach Links und Rechts gezogen werden kann
tableDashlet1.addOption("table.rowReorder.snapX", "true");
```

Filter Optionen

Wie kann DataTable gefiltert werden?

Der DataTable kann mit die folgende Funktions-Aufruf gefiltert werden.

```
// MemoryBuffer Splate als HTML erstellen
table1_lazyLoadTable('searchValue1', 'Column1', 'searchValue2', 'Column2',
'searchValue3', 'Column3', 'searchValue4', 'Column4', 'searchValue5', 'Column5');
```



- Bis zum 5 Such-Wert kann einmal gemacht werden.
- Der Funktions-Name ist <table_ID>_lazyLoadTable

Die folgende Werte kann für die *searchOperator* Optionen benutzt werden. Der Default Such-Operation ist "==" .

- == oder equals
- != oder notequals
- < oder greaterthan
- <= oder greathanorequalto
- > oder lessthan
- <= oder lessthanorequalto
- contains oder includes
- notcontains oder notincludes
- startswith
- notstartswith
- endswith
- notendswith

```
// Such Operators (bis 5)
dashlet.addOption("data.script.searchOperator1","contains");
dashlet.addOption("data.script.searchOperator2","notcontains");
dashlet.addOption("data.script.searchOperator3","==");
dashlet.addOption("data.script.searchOperator4","!=");
dashlet.addOption("data.script.searchOperator5",">=");
```

Wie kann Splate als HTML definiert werden?

Ein Splate kann mit die folgende Option als HTML-Spalte gesetzt werden.

```
// MemoryBuffer Splate als HTML erstellen
mbr.getColumnDefinition("html_column"
).addAdditionalInfo(DashletManager.MBR_DASHLET_HTML_COLUMN, "true");
```

Falls in der Spalten-Werte Single-Quotation-Marks vorhanden sind, dann sind diese Single-Quotes 'escaped'.

Wenn diese Fall nicht gewünscht ist, dann bitte die folgende Option nutzen.

```
// MemoryBuffer Splate als HTML erstellen
mbr.getColumnDefinition("html_column"
).addAdditionalInfo(DashletManager.MBR_DASHLET_HTML_COLUMN_DATATABLE, "true");
```

Wie kann die Spalten in ein DataTable ausgeblendet werden?

Die Spalten kann mit die folgende Option in ein DataTable ausgeblendet werden.

```
//Hide columns 0 and 11
dashlet.addOption("table.columnDef.t[0,11].visible", "false");

//Using column names
dashlet.addOption("table.columns.n[ID].visible", "false");
dashlet.addOption("table.columns.n[PRODUCTSCALE].visible", "false");
```

Wie kann ein click-Event für ein Zeile in DataTable gesetzt werden?

Die Click-Event für ein Zeile kann mit die folgende Option gesetzt werden.

```
// Click-event for a row
dashlet.addOption("table.customScript", "$(document).ready(function(){\n" +
    "    var table = $('#table1').DataTable();\n" +
    "    table.$('tr').css(\"cursor\", \"pointer\");\n" +
    "    $('#table1 tbody').on( 'click', 'tr', function () {\n" +
    "        alert('Click on table row: ' +
table.rows(this).indexes()[0] + '\\ndata: [' + table.rows(this).data()[0] + ']);\n" +
    "    } );\n" +
    "});");
```

Wie kann ein Custom-Javascript für DataTable erstellt werden?

Eigene Javascript Code hinzufügen in ein DataTable Dashlet.

```
// Varaint 1 :
// Custom Skript aus ein externe js Datei
tableDashlet1.addOption("table.customScript", readResourceFileAsString("dashlet_table
_planning_stock_custom_scripts.js"));

// Varaint 2 :
// Custom Skript in den .rptdesign-Datei definieren
dashlet.addOption("table.customScript",
    "$(document).ready(function(){\n" +
    "    var table = $('#table1').DataTable();\n" +
    "    table.$('tr').css(\"cursor\", \"pointer\");\n" +
    "    $('#table1 tbody').on( 'click', 'tr', function () {\n" +
    "        alert('Click on table row: ' + table.rows(this).indexes()[0] + '\\ndata: ['
+ table.rows(this).data()[0] + ']);\n" +
    "    } );\n" +
    "});");
```

```
//Funktion definition
//die ganze Pfad sollte berücksichtigt werden
glResourceFolderReport = "/customScript/";
function readResourceFileAsString(filename)
{
    try
    {
        return "" +
        ToolboxMasterObject.getResourceHelper().readResourceFileUTF8(reportContext,
        glResourceFolderReport + filename);
    } catch (er) {
        return "";
    }
}
```

Style Anpassungen

Wie kann ein Spalten-Breite in DataTable gesetzt werden?

Die Spalten-Breite kann mit die folgende Option gesetzt werden.

```
//Option 1: Column Index
tableDashlet1.addOption("table.columnDef.t[3,4].width", "10%");
//Option 2: Column Name
tableDashlet1.addOption("table.columns.n[Beanstandungsgrund].width","10%");

// Option 3 : ein SyleClass für diese Spalten zu definieren:
tableDashlet1.addOption("table.columns.n[Beanstandungsgrund].className", "px200");
tableDashlet1.addOption("table.columns.n[Kontrollart].className", "px200");
tableDashlet1.addOption("table.columns.n[Behörde].className", "px200");

//Und dann in der HTML-Dashlet, kann diese Style definieret werden:
customHTML += "<style>                                \n";
customHTML += ".px200 {                                \n"; // Name des Styles
customHTML += "    width: 200px;                        \n";
customHTML += "    max-width: 200px;                    \n";
customHTML += "    word-wrap: break-word;                    \n";
customHTML += "}"                                           \n";
customHTML += "</style>"                                \n";
jsDashlet.addOption("html",customHTML);
```

Falls der "scrollX" Option (*table.feature.scrollX*) gesetzt wird, dann kann der Spalten-Breite gesetzt werden, wenn noch ein Style an der Tabelle definiert wird.

```
//Und dann in der HTML-Dashlet, kann diese Style definiert werden:
customHTML += "<style>                                \n";
customHTML += ".px200 {                                \n"; // Name des Styles
customHTML += "    width: 200px;                        \n";
customHTML += "    max-width: 200px;                    \n";
customHTML += "    word-wrap: break-word;              \n";
customHTML += "}                                        \n";
customHTML += "#table1 , #table2{                       \n"; //Name die Tabellen
customHTML += "    table-layout: fixed !important;        \n";
customHTML += "    word-wrap: break-word;                  \n";
customHTML += "}                                        \n";
customHTML += "</style>                                \n";
jsDashlet.addOption("html",customHTML);
```

Wie kann ein Custom-Style für DataTable erstellt werden?

Custom Style für DataTable Dashlet kann als folgendes erstellt werden.

```
// Varaint 1 :
// Custom Skript aus ein externe js Datei
// Custom-Style-Sheets für DataTable-Dashlet verwenden / Ressourcen-File -->
dashlet_table_style.css
tableDashlet1.addOption("dashlet.custom.styles", readResourceFileAsString("dashlet_table_style.css"));

// Varaint 2 :
// Custom Skript in den .rptdesign-Datei definieren
tableDashlet1.addOption("dashlet.custom.styles", "div .dataTables_scrollBody {border-left: 0px none #fff !important }");
```

```
//Funktion definition
//die ganze Pfad sollte berücksichtigt werden
glResourceFolderReport = "/customScript/";
function readResourceFileAsString(filename)
{
    try
    {
        return "" +
        ToolboxMasterObject.getResourceHelper().readResourceFileUTF8(reportContext,
        glResourceFolderReport + filename);
    } catch (er) {
        return "";
    }
}
```

Wie kann ein Zeile in DataTable farblich hervorheben basiert an Splate-Wert?

Die Highlighting ein Zeile kann mit die folgende Option gesetzt werden.

```
// Highlight row in lightblue color if value in column[5]=USA
dashlet.addOption("table.callback.rowCallback" , " if ( data[5] == 'USA' ) { $
('td', row).css('background-color', 'lightblue'); } ");
```

Das Table-Dashlet sieht folgende aus.

Customers Classic Models

Show 10 entries

Search:

Customer Number	Customer Name	First name	Last name	City	Country	Adressline 1	Adressline 2	Phone	Postal Code	Sales number	Creditlimit
103	Atelier graphique	Carine	Schmitt	Nantes	France	54, rue Royale		40 32 2555	44000	1370	21000
112	Signal Gift Stores	Jean	King	Las Vegas	USA	8489 Strong St.		7025551838	83030	1166	71800
114	Australian Collectors, Co.	Peter	Ferguson	Melbourne	Australia	636 St Kilda Road	Level 3	03 9520 4555	3004	1611	117300
119	La Rochelle Gifts	Janine	Labrune	Nantes	France	67, rue des Cinquante Otages		40 67 8555	44000	1370	118200
121	Baane Mini Imports	Jonas	Bergulfsen	Stavem	Norway	Erling Skakkes gate 78		07-98 9555	4110	1504	81700
124	Mini Gifts Distributors Ltd.	Susan	Nelson	San Rafael	USA	5677 Strong St.		4155551450	97562	1165	210500
125	Havel & Zbyszek Co	Zbyszek	Piestrzeniewicz	Warszawa	Poland	ul. Filitrowa 68		(26) 642-7555	01-012		0
128	Blauer See Auto, Co.	Roland	Keitel	Frankfurt	Germany	Lyonerstr. 34		+49 69 66 90 2555	60528	1504	59700
129	Mini Wheels Co.	Julie	Murphy	San Francisco	USA	5557 North Pendale Street		6505555787	94217	1165	64600
131	Land of Toys Inc.	Kwai	Lee	NYC	USA	897 Long Airport Avenue		2125557818	10022	1323	114900

Showing 1 to 10 of 122 entries

Previous 1 2 3 4 5 ... 13 Next

Wie kann ein Spalte in DataTable farblich hervorheben basiert an Splate-Wert?

Die Highlighting ein Spalte kann mit die folgende Option gesetzt werden.

```
// Using Column-Indexes
//Set the Color of Country if Flag1 == 1
var rowCallback = " if ( data[47] == 1 ) { $('td', row).eq(2).css('background-color',
'red'); } \n " +
tableDashlet1.addOption("table.callback.rowCallback" , rowCallback);
```

Möglichkeit wäre falls der Spalten ein/ausblendet werden kann sind folgendes:

```
// Set the Color of Country if Flag1 == 1
tableDashlet1.addOption("table.callback.rowCallback" , " // get index of visible
columns only\n" +
    "     var colnamesVisible = [];\n" +
    "     var colnames = [];\n" +
    "     $( '#table1').DataTable().columns().every( function () {\n" +
    "         if (this.visible()) {colnamesVisible.push($
(this.header()).text() ); }\n" +
    "         colnames.push($(this.header()).text());\n" +
    "     } );\n" +
    "     indexVisibleColList =colnamesVisible.indexOf('Country');\n" +
    "     index = colnames.indexOf('Flag1');\n" +
    "     if (indexVisibleColList > -1) {\n" +
    "         if( data[index] == 1 ) { \n" +
    "             $('td', row).eq(indexVisibleColList).css('background-
color', 'red');\n" +
    "         }\n" +
    "     }\n" ;
```

Wie kann die Spalten in ein DataTable als Auswahl ein/ausblendet werden?

Die Spalten kann mit die folgende Option in ein DataTable ein/ausblendet (als Auswahl) werden.

```
//Column List to hide / show the columns
dashlet.addOption("table.buttons.buttons", "['colvis']");
```


Details

Column visibility

ID				
CUSTOMERNUMBER		CUSTOMERNAME		ORDERNUMBER
CUSTOMERNAME				ORDERDATE
ORDERNUMBER				QUANTITYORDERED
ORDERDATE				PRICEEACH
QUANTITYORDERED		Mini Gifts Distributors Ltd.	10182	2011-11-11
PRICEEACH		Mini Gifts Distributors Ltd.	10182	2011-11-11
COUNTRY		Mini Gifts Distributors Ltd.	10182	2011-11-11
PRODUCTLINE		Saveley & Henriot, Co.	10194	2011-11-11
PRODUCTSCALE		Saveley & Henriot, Co.	10194	2011-11-11
PRODUCTVENDOR		King Kong Collectables, Co.	10187	2011-11-11
ORDERYEAR		King Kong Collectables, Co.	10187	2011-11-11
CC_UMSATZ		Enaco Distributors	10197	2011-11-11
211		Enaco Distributors	10197	2011-11-11
216				
216				

Wie kann nur manche Spalten in die ColumnVisiblilty Liste darstellt werden und wie kann nur manche Spalten in das Export-Datei berücksichtigt werden?

Die Splaten, die mit ein ClassName definiert sind, kann in die ColumnVisiblilty Liste und auch in der Export-Funktion als folgendes berücksichtigt werden:

```
//ClassName für die Spalten, die nicht berücksichtigt werden soll, definieren
tableDashlet2.addOption("table.columns.n[checkpoints_fine].className", "noVis");

//ClassName berücksichtigen bei der BUTTONS definition
tableDashlet2.addOption("table.buttons.buttons" , "[\n" +
"{extend:'colvis', columns : ':not(.noVis)}',\n" +
"{extend:'csv',title:'OutputData' , exportOptions: { columns: ':not(.noVis)}'},\n" +
"{extend:'excel',title:'OutputData' , exportOptions: { columns: ':not(.noVis)}'}\n" +
"]");
```

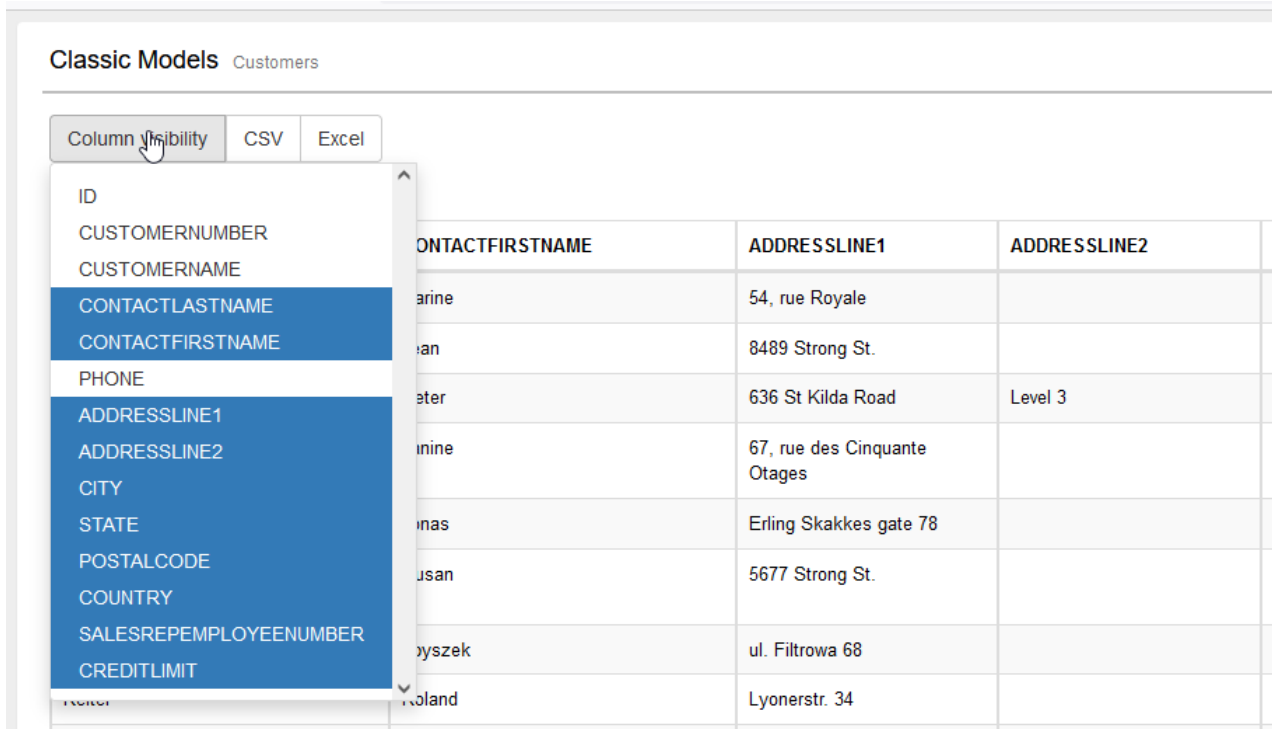
Wie kann manche Liste von Anfang ausblendet werden, aber bei bedarf darstellt werden können?

Die Spalten, die nicht darstellt werden soll kann mit der "visible" Option ausgeblendet werden.

```
dashlet.addOption("table.columnDef.t[0,1,2,5].visible", "false");
```

Wenn der Colvis (Column-Visiblity) button dargestellt ist, dann sind die alle Spalten in die Liste. Die Spalten, die als "visible=false" gesetzt wird, sind auch drin.

```
//Buttons
dashlet.addOption("table.buttons.buttons" ,  "["colvis','csv','excel']");
```



Wie kann die Formatierung hinterlegen werden?

Die Formatierung für Spalten kann mit die folgende Optionen gesetzt werden in verschiedene Darstellungen.

```
//Case 1: keine Nachkommastellen und Tausender-Trennpunkte für Spalte 10 und 11
dashlet.addOption("table.columnDef.t[10,11].render",
"$$.fn.dataTable.render.number('.', ',', 0, '')"); // 123,000

//Case 2: keine Nachkommastellen und Tausender-Trennpunkte für 11 mit Currency-Sign
am vorne
dashlet.addOption("table.columnDef.t[11].render", "$$.fn.dataTable.render.number('.',
',', 0, '$ ')"); // $ 123,000

//Case 3: keine Nachkommastellen und Tausender-Trennpunkte für 10 mit Currency-Sign
am Ende
dashlet.addOption("table.columnDef.t[10].render", "$$.fn.dataTable.render.number('.',
',', 0, '', '$')"); // 123,000 $
```

Wie kann eine Horizontal-Scrollbar gesetzt werden?

Eine Horizontal-Scrollbar kann mit die folgende Option gesetzt werden.



ScrollX kommt nur wenn die Spalten in der Breite fix sind sodass er es nicht zusammenschieben kann.

```
dashlet.addOption("table.feature.scrollX","true");
```

Details



CSV

Excel

Search:

CUSTOMERNUMBER	CUSTOMERNAME	ORDERNUMBER	ORDERDATE	QUANTITYORDERED	PRICEEACH	COUNTRY
141	Euro+ Shopping Channel	10133	2011-06-27	24	76.73	Spain
151	Muscle Machine Inc	10127	2011-06-03	25	126.39	USA
198	Auto-Moto Classics Inc.	10130	2011-06-16	40	68.82	USA
324	Stylish Desk Decors, Co.	10129	2011-06-12	41	81.43	UK



Showing 1 to 4 of 4 entries

Previous

1

Next

Wie kann die Höhe des DataTables angepasst werden?

Option 1:

Die Höhe des DataTables kann mit die folgende Option angepasst werden.

```
dashlet.addOption("table.feature.scrolly","200px");
```

Option 2:

Es ist auch möglich die Höhe mit ein HTML Dashlet anzupassen. Die Skripte für die HTML-Dashlet ist folgendes.

```
var jsDashlet = DashletManager.addDashlet("DashletHTML", "", "", false, null);
// Die Table Dashlet hat die ID als 'table1'.
// Die Höhe des DataTables ist als 1000px gesetzt.
var customJS =
"  $(document).ready(function(){"+
"  $('#dashletbox-table1 div.dash-content').css('max-height','1000px').css('overflow-y','scroll');"+
"  });\n\n";

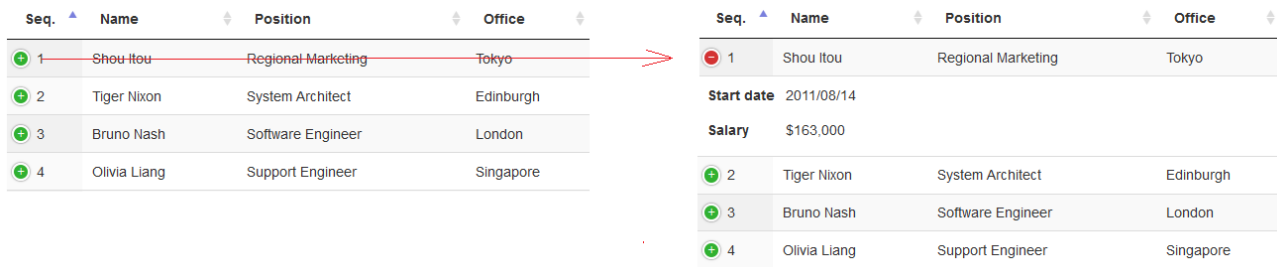
jsDashlet.addOption("javascript", customJS);
```

Wie kann der DataTables als Responsive gesetzt werden?

Der DataTables kann mit die folgende Option als responsive gesetzt werden.

Responsive passt die Sichtbarkeit von Spalten in Ihren Tabellen automatisch an, damit das Layout der Informationen unabhängig von der Bildschirmgröße gut dargestellt wird.

```
// Table responsive Feature  
tableDashlet.addOption("table.responsive", "true");
```



Seq.	Name	Position	Office
1	Shou Itou	Regional Marketing	Tokyo
2	Tiger Nixon	System Architect	Edinburgh
3	Bruno Nash	Software Engineer	London
4	Olivia Liang	Support Engineer	Singapore

Seq.	Name	Position	Office
1	Shou Itou	Regional Marketing	Tokyo
Start date 2011/08/14			
Salary \$163,000			
2	Tiger Nixon	System Architect	Edinburgh
3	Bruno Nash	Software Engineer	London
4	Olivia Liang	Support Engineer	Singapore

Such-Optionen

Wie kann ein Custom Such Funktion erstellt werden?

Ein Custom Such Funktion kann mit die folgende Optionen erstellt werden.

```
// Custom Such Funktion aktivieren
dashlet.addOption("table.ext.search", "true");

// Custom Search Funktionen
dashlet.addOption("table.ext.search.custom1", "if (dataIndex == 0)
console.log(options);\nfor(var i=0;i<data.length;i++)\n{\nif ('Frankfurt' ==
String(data[i])) return true;\n}");
dashlet.addOption("table.ext.search.custom2", "for(var i=0;i<data.length;i++)\n{\nif
('USA' == String(data[i])) return true;\n}");

dashlet.addOption("table.ext.search.startSearchMode", "exact"); // welche custom
search sollte als Erstes darstellt werden
dashlet.addOption("table.ext.search.searchModeList",
"['like','exact','smart','custom1','custom2']"); // serach Button Liste
dashlet.addOption("table.ext.search.searchModeButtonNames", "['ähnliche
Suche','exakte Suche','Smarte Suche','Eigene Suche Frankfurt','Eigene Suche USA']"); /
/ Eigene Name für die search Button Liste
dashlet.addOption("table.ext.search.searchModeButtonVisible", "true"); // Aus-
einblenden des Search-Buttons
dashlet.addOption("table.ext.search.caseinsensitiveSearch", "true"); // Aktivieren
der Case-Sensitive Search
dashlet.addOption("table.ext.search.caseinsensitiveCheckBoxVisible", "true"); //
Checkbox für der Case-Sensitive Serach
dashlet.addOption("table.ext.search.caseinsensitiveText", "Groß- Klein Egal"); //
Eigene Name für der Case-Sensitive Search

// Min und Max Suche Aktivieren
dashlet.addOption("table.ext.search.inputBarShow", "true");
dashlet.addOption("table.ext.search.inputBarMax", "true");
dashlet.addOption("table.ext.search.inputBarMaxtext", "Maximum:");
dashlet.addOption("table.ext.search.inputBarMin", "true");
dashlet.addOption("table.ext.search.inputBarMintext", "Minimum:");

dashlet.addOption("table.ext.search.minmaxColumnIndex", "2");
dashlet.addOption("table.ext.search.minmaxColumnType", "other"); // DataType der
Column
dashlet.addOption("table.ext.search.minmaxColumnName", "Customer Name");

dashlet.addOption("table.ext.search.hiddenSearch", "true"); // Aus-einblenden des
Custom Search, falls nur Min-Max darstellt werden soll
```

Wie kann Suchfelder für Spalten DataTable gesetzt werden?

Die Suchfelder für Spalten kann mit die folgende Option gesetzt werden.

```
// Column Search Filter
dashlet.addOption("data.script.columnSearch","true"); // option to indicate if the
search fields are to be displayed or not
dashlet.addOption("data.script.columnSearchColumns","['CUSTOMERNUMBER',
'CUSTOMERNAME', 'ORDERNUMBER']"); // list of columns for which search fields should
be displayed
dashlet.addOption("data.script.flexselect","true"); // option to indicate if an
Autocomplete search fields are displayed or not
```

147

Search:

CUSTOMERNUMBER	CUSTOMERNAME	ORDERNUMBER	ORDERDATE	PRICEEACH	PRODUCTLINE	PRODUCTSCALE	PRODUCTVENDOR	CC_UMSATZ
242								
242	Alpha Cognac	10178	2011-11-08	131.92	Classic Cars	1:10	Motor City Art Classics	3166.08
242	Alpha Cognac	10178	2011-11-08	91.74	Planes	1:700	Motor City Art Classics	2018.28
242	Alpha Cognac	10178	2011-11-08	41.71	Planes	1:72	Motor City Art Classics	1876.95

Showing 1 to 3 of 3 entries (filtered from 26 total entries)

Previous 1 Next

Wie kann die Multi-select Boxen für alle Spalten gesetzt werden?

Die Multi-select Boxen für alle Spalten kann mit die folgende Option gesetzt werden.

```
dashlet.addOption("table.callback.initComplete", "    var table_footer = $('<tfoot></tfoot>').appendTo(this);\n" +
"    var table_footer_row = $('<tr></tr>').appendTo(table_footer);\n" +
"    \n" +
"    this.api().columns().every( function () {\n" +
"    var column = this;\n" +
"    var table_footer_col = $('<th></th>').appendTo(table_footer_row);\n" +
"    var select = $('<select><option value=\"\"></option></select>')\n" +
"    .appendTo( table_footer_col )\n" +
"    .on( 'change', function () {\n" +
"    var val = $.fn.dataTable.util.escapeRegex(\n" +
"    $(this).val()\n" +
"    );\n" +
"    \n" +
"    column\n" +
"    .search( val ? '^'+val+'$' : '', true, false )\n" +
"    .draw();\n" +
"    } );\n" +
"    \n" +
"    column.data().unique().sort().each( function ( d, j ) {\n" +
"    select.append( '<option value=\"'+d+'\">'+d+'</option>' )\n" +
"    } );\n" +
"    \n" +
"    } );");
```

Das Table-Dashlet sieht folgende aus.

Customers Classic Models

Show 10 entries

Customer Number	Customer Name	First name	Last name	City	Country	Adressline 1	Adressline 2	Phone	Postal Code	Sales number	Creditlimit
103	Atelier graphique	Celine	Schmitt	Nantes	France	54, rue Royale		40 32 2555	44000	1370	21000
112	Signal Gift Stores	Jean	King	Las Vegas	USA	8489 Strong St.		7025551838	83030	1166	71800
114	Australian Collectors, Co.	Peter	Ferguson	Melbourne	Australia	636 St Kilda Road	Level 3	03 9520 4555	3004	1611	117300
119	La Rochelle Gifts	Janine	Labruno	Nantes	France	67, rue des Cinquante Otages		40 67 8555	44000	1370	118200
121	Blaene Mini Imports	Jonas	Bergulfen	Stavern	Norway	Erling Skakkes gate 78		07 98 9555	4110	1584	81700
124	Mini Gifts Distributors Ltd.	Susan	Nelson	San Rafael	USA	5677 Strong St.		4155551450	97562	1165	210500
125	Havel & Zbyszek Co.	Zbyszek	Piestrzeniewicz	Warszawa	Poland	ul. Filitowa 68		(26) 642-7555	01-012		0
128	Blauer See Auto. Co.	Roland	Keitel	Frankfurt	Germany	Lyonerstr. 34		+49 69 66 90 2555	60528	1504	59700
129	Mini Wheels Co.	Julie	Murphy	San Francisco	USA	5557 North Pendale Street		6505555787	94217	1165	64600
131	Land of Toys Inc.	Kwai	Lee	NYC	USA	897 Long Airport Avenue		2125557818	10022	1323	114900

Showing 1 to 10 of 122 entries

Previous 1 2 3 4 5 ... 13 Next

Button Darstellungen

Wie kann der erste Zeile eines Excel-Datei angepasst werden?

Der erste-Zeile von kann Excel-Datei kann mit den folgenden Option angepasst werden.

```
table.buttons.buttons=[{extend:'csv', title:'MyCSVExport'}, {extend:'excel', autoFilter:
:true, title:'MyExcelExport', customize: function (xlsx) { var sheet =
xlsx.xl.worksheets['sheet1.xml']; $('row:first c', sheet).attr('s', '30'); /*Style
30 - Normal text, grey background, thin black border*/} }]
```

A1

Mein Excel Export

	A	B	C	D	E	F	G	H	I
1	Mein Excel Export								
2	ID	ReferenceType	Status	Startzeit	Endzeit	Dauer	Anz. Fehler	start_date	
3	1	3		17:18:28	17:18:34	0 Std. 0 Min. 6 Sek.		0 16.07.23	
4	2	3		17:18:25	17:18:28	0 Std. 0 Min. 3 Sek.		0 16.07.23	
5	3	3		17:18:12	17:18:13	0 Std. 0 Min. 1 Sek.		0 16.07.23	
6	4	3		17:18:34	17:18:38	0 Std. 0 Min. 4 Sek.		0 16.07.23	
7	5	3		17:18:34	17:18:38	0 Std. 0 Min. 4 Sek.		0 16.07.23	
8	6	3		17:18:38	17:18:41	0 Std. 0 Min. 3 Sek.		0 16.07.23	
9	7	3		17:18:48	17:18:50	0 Std. 0 Min. 2 Sek.		0 16.07.23	
10	8	3		17:18:28	17:18:34	0 Std. 0 Min. 6 Sek.		0 16.07.23	
11	9	3		17:18:42	17:18:45	0 Std. 0 Min. 3 Sek.		0 16.07.23	
12	10	3		17:18:28	17:18:34	0 Std. 0 Min. 6 Sek.		0 16.07.23	
13	11	3		17:18:18	17:18:19	0 Std. 0 Min. 1 Sek.		0 16.07.23	
14	12	3		17:18:13	17:18:16	0 Std. 0 Min. 3 Sek.		0 16.07.23	

Mehr Info bezüglich Styles finden Sie auch in den folgenden Link: <https://datatables.net/reference/button/excelHtml5>

Wie kann Buttons hinzugefügt werden?

Die Buttons kann mit die folgende Option hinzugefügt werden.

```
//Buttons
dashlet.addOption("table.buttons.buttons" , ["'csv','excel'"]);
```

Das Table-Dashlet sieht folgende aus.

Details

[CSV](#)
[Excel](#)

CUSTOMERNUMBER	CUSTOMERNAME	ORDERNUMBER
146	Saveley & Henriot, Co.	10194
146	Saveley & Henriot, Co.	10194
167	Herkku Gifts	10181
175	Gift Depot Inc.	10172
216	Enaco Distributors	10197
216	Enaco Distributors	10197
216	Enaco Distributors	10197
242	Alpha Cognac	10178
319	Mini Classics	10195
320	Mini Creations Ltd.	10185

Showing 1 to 10 of 27 entries

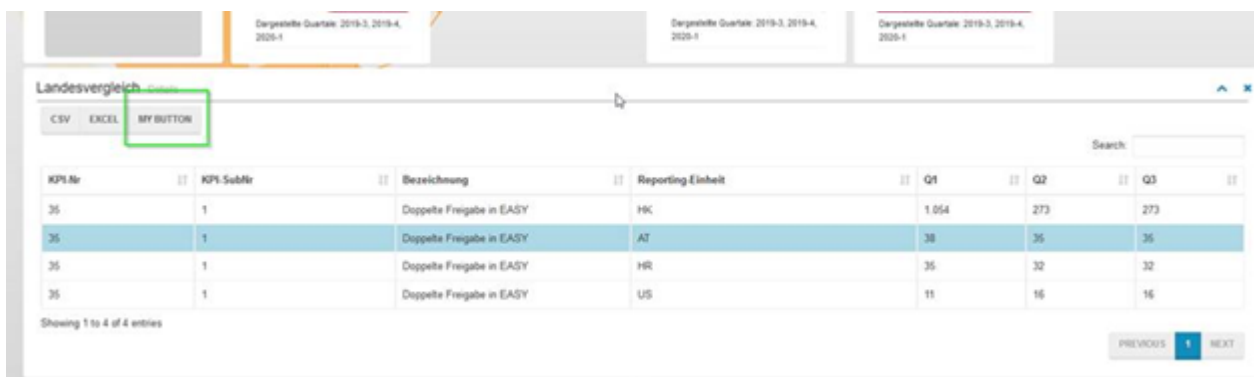
Wie kann customized Button hinzugefügt werden?

Die customized-Button kann mit die folgende Option hinzugefügt werden.

```
//customize Buttons
var myButton = "{\n" +
  "text: 'My button',\n" +
  "action: function ( e, dt, node, config ) {\n" +
  "  alert( 'Button activated' );\n" +
  "  }\n" +
  " }";

tableDashlet.addOption("table.buttons.buttons" , "[{extend:'csv',
title:'OutputData'},{extend:'excel', title:'OutputData'}, " + myButton + "]");
```

Das Table-Dashlet sieht folgende aus.



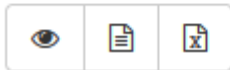
KPI.Nr	KPI.SubNr	Bezeichnung	Reporting Einheit	Q1	Q2	Q3
35	1	Doppelte Freigabe in EASY	HK	1.054	273	273
35	1	Doppelte Freigabe in EASY	AT	36	36	36
35	1	Doppelte Freigabe in EASY	HR	35	32	32
35	1	Doppelte Freigabe in EASY	US	11	16	16

Wie kann Icons für die Buttons gesetzt werden?

Ein Icon kann an die Button als folgendes gesetzt werden.

```
dashlet.addOption("table.buttons.buttons", "[\n" +
    "{extend:'colvis' , text: '<i class=\"fa fa-eye\"></i>' }, \n" +
    "{extend:'csv' , title:'OutputData' , text: '<i class=\"fa fa-file-text-o\"></i>' }, \n" +
    "{extend:'excel' , title:'OutputData' , text: '<i class=\"fa fa-file-excel-o\"></i>' } \n" +
    "]);
```

Details



PRODUCTSCALE ↑↓	PRODUCTVENDO

Wie kann die ausgeblendete Spalten beim Export rausgelasst werden?

Die Spalten, die ausgeblendet sind, kann mit die folgende Option beim Export nicht berücksichtigt werden.

```
// splate ausblenden
dashlet.addOption("table.columnDef.t[0,12].visible", "false");

//Buttons
dashlet.addOption("table.buttons.buttons" , "[\n" +
    "{extend:'pdf', title:'ClassicModels_Customers' , orientation: 'landscape',
    pageSize:'LEGAL', exportOptions: { columns: ':visible'}},\n" +
    "{extend:'csv',title:'ClassicModels_Customers' , exportOptions:
    { columns: ':visible'}},\n" +
    "{extend:'excel',title:'ClassicModels_Customers' , exportOptions:
    { columns: ':visible'}}\n" +
    "]);
```

Wie kann manche Spalten beim Export nicht mitgegeben werden?

Die Spalten, die beim Export berücksichtigt werden kann in zwei Variante gegeben werden.

Variant 1 : Eine ClassName für die Spalten definieren

```
//set className
dashlet.addOption("table.columns.n[ID].className", "noVis");

//Buttons
dashlet.addOption("table.buttons.buttons" , "[\n" +
    "{extend:'csv',title:'ClassicModels_Customers' , exportOptions:
    { columns:':not(.noVis)'}},\n" +
    "{extend:'excel',title:'ClassicModels_Customers' , exportOptions:
    { columns:':not(.noVis)'}},\n" +
    "]" );
```

Variante 2 : Die Spalten-Indexes mitgeben.

```
//Buttons
dashlet.addOption("table.buttons.buttons" , "[\n" +
    "{extend:'csv',title:'ClassicModels_Customers' , exportOptions: { columns:[ 0, 1,
    2, 5 ]}},\n" +
    "{extend:'excel',title:'ClassicModels_Customers' , exportOptions: { columns:[ 0,
    1, 2, 5 ]}},\n" +
    "]" );
```

Wie kann der Delimiter für CSV-Dateien geändert werden?

Der Delimiter für CSV-Datei Export kann als folgendes angepasst werden.

```
// CSV delimiter anpassen
dashlet.addOption("table.buttons.buttons" , "[{extend:'csv',filename:'OutputData' ,
    fieldSeparator: ';'}, 'excel'}]");
```

Wie kann ein DruckOption gesetzt werden?

Der Druck Option kann als folgendes gesetzt werden.

```
dashlet.addOption("table.buttons.buttons", "['pdf']"); // Create and save a PDF file
// that contains the data from the table
dashlet.addOption("table.buttons.buttons", "['print']"); //Button show a printable
// view of the table's data
```

Es gibt auch die Möglichkeit die Print-Option anzupassen, wie folgendes:

```
dashlet.addOption("table.buttons.buttons" , "[ {extend:'pdf', orientation:
    'landscape', pageSize: 'LEGAL', exportOptions: { columns: ':visible'}} ]");
```

Wie kann die RAW-Daten exportiert werden und nicht die Formatierte-Daten?

Mit der folgende Option kann nur die RAW-Daten, die für Datatable benutzt wird in der Export-Datei dargestellt werden. Bei default ist die Rendered-Daten (Formatierte-Daten) exportiert.

```
dashlet.addOption("table.buttons.buttons" , "[\n" +
    "{extend:'csv', exportOptions: { orthogonal: 'export' }}, \n" +
    "{extend:'excel', exportOptions: { orthogonal: 'export' }}\n" +
    "]);
```

Wie kann customized Bootstrap Buttons erstellt werden?

Für ein customized Bootstrap Button zu erstellen, muss die DataTables Buttons erzeugt werden, aber **"unsichtbar"** gesetzt werden soll, sodass die CSV, Excel-Export , usw. Funktionen kann direkt von DataTables benutzt werden.

```
//id für dataTable
dashlet.setId("table1");

// DataTable Buttons erstellen
dashlet.addOption("table.buttons.buttons" , "['csv','excel']");

// Buttons unsichtbar machen mit custom Script
dashlet.addOption("dashlet.custom.styles", "#table1_wrapper .dt-buttons {display: none; }");

// Bar erzeugen mit id 'topbarTable' Position TOP als Klasse DashletBootstrapBar
dashlet.addOption("add.bar", "topbarTable,TOP,DashletBootstrapBar");
// Bootstrap-Buttongröße auf XS verringern // Gilt für alle Elemente der Bar
dashlet.addOption("bootstrap.button.size", "xs");

// Button erzeugen mit id 'simpleButton1' als Klasse BootstrapButton
dashlet.addOption("bar.n[topbarTable].add.component",
    "BootstrapButton,simpleButtonCSV");
dashlet.addOption("bar.n[topbarTable].component.n[simpleButtonCSV].button.name",
    "CSV");
dashlet.addOption("bar.n[topbarTable].component.n[simpleButtonCSV].button.size", "xs")
;
dashlet.addOption("bar.n[topbarTable].component.n[simpleButtonCSV].button.onclick",
    "$('#table1_wrapper .dt-buttons .buttons-csv').click();");

// Button erzeugen mit id 'simpleButton1' als Klasse BootstrapButton
dashlet.addOption("bar.n[topbarTable].add.component",
    "BootstrapButton,simpleButtonEXCEL");
dashlet.addOption("bar.n[topbarTable].component.n[simpleButtonEXCEL].button.name",
    "EXCEL");
dashlet.addOption("bar.n[topbarTable].component.n[simpleButtonEXCEL].button.size",
    "xs");
dashlet.addOption("bar.n[topbarTable].component.n[simpleButtonEXCEL].button.onclick",
    "$('#table1_wrapper .dt-buttons .buttons-excel').click();");
```



Sonsitges

Wie kann bei der DataTables ein Gruppierung Option gesetzt werden?

Ein Gruppierung Option kann mit die folgende Option als responsive gesetzt werden.

In der MemoryBuffer sollten der Spalten-attribute "aggregation" gesetzt werden, sodass die Spalte sind mit diese Aggregation-Value dargestellt.

Für die Spalten, die kein "aggregation" gesetzt sind, sind als Default NULL gesetzt.

```
//give the column aggregation type in the memoryBuffer
mbrData.getColumnDefinition("ANZ_ARTIKEL"
).addAdditionalInfo(DashletManager.MBR_DASHLET_AGGREGATION ,
DashletManager.MBR_AGG_SUM); //SUM
mbrData.getColumnDefinition("STUECK"
).addAdditionalInfo(DashletManager.MBR_DASHLET_AGGREGATION ,
DashletManager.MBR_AGG_COUNT); //COUNT
```

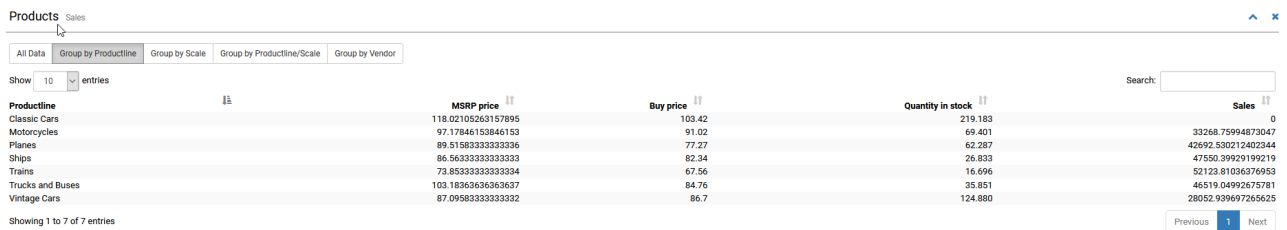
Die Gruppierung ist mit dem Buttons erstellt.

```
//Buttons on top
buttonbar = dashlet.getDashlet().createDashletBar("DashletButtonBar", "TOP");
btn = buttonbar.addButton("All Data", false);
btn.addOption("button.onclick", "table3_dataSet_group_columns =
[];table3_lazyLoadTable();");
btn = buttonbar.addButton("Group by Productline", false);
btn.addOption("button.onclick", "table3_dataSet_group_columns =
['Productline'];table3_lazyLoadTable();");
btn = buttonbar.addButton("Group by Scale", false);
btn.addOption("button.onclick", "table3_dataSet_group_columns =
['Scale'];table3_lazyLoadTable();");
btn = buttonbar.addButton("Group by Productline/Scale", false);
btn.addOption("button.onclick", "table3_dataSet_group_columns =
['Productline','Scale'];table3_lazyLoadTable();");
btn = buttonbar.addButton("Group by Vendor", false);
btn.addOption("button.onclick", "table3_dataSet_group_columns =
['Vendor'];table3_lazyLoadTable();");
```

Hinweis:

- Der Variable "<DataTableID>_dataSet_group_columns" definiert die Spalten-Namen für die Gruppierung.
- Der Funktion "<DataTableID>_lazyLoadTable()" lädt die DataTable neue mit die Gruppierung.

Die Ergebnis sieht so aus :



Productline	MSRP price	Buy price	Quantity in stock	Sales
Classic Cars	118.02105263157895	103.42	219.183	0
Motorcycles	97.17846153846153	91.02	69.401	33268.75994873047
Planes	89.51583333333336	77.27	62.287	42692.530212402344
Ships	86.56333333333333	82.34	26.833	47550.39929199219
Trains	73.85333333333334	67.56	16.696	52123.81036376953
Trucks and Buses	103.18363636363637	84.76	35.851	46519.04992675781
Vintage Cars	87.09583333333332	86.7	124.880	28052.939697265625

Wie kann der DataTable als LazyLoad gesetzt werden?

Der DataTable kann mit die folgende Optionen als LazyLoad gesetzt werden.

```
//LazyLoad
dashlet.addOption("table.lazy.load", "true");
dashlet.addOption("table.lazy.load.time", "5");
```

Wie kann der DataTable als Popup darstellt werden?

Für der DataTable als Popup darzustellen muss die folgende Optionen definiert werden.

```
var tableDashlet1 = DashletManager.addDashletTableDynamic("table1" , "" , true ,
mbr);
tableDashlet1.setId("table1");
tableDashlet1.addOption("modal.box", "true");
tableDashlet1.addOption("box.modal.size","xl"); // die größe der Drilldown ist
dann groß als die Bildschirm Fenster
```

In die Click-Event Function (von ein Chart oder Link)muss die Definition als folgendes sein

```
// Wenn der DataTable als LazyLoad definiert ist, dann der Such-Funktion heißt
<DataTable_Id>_lazyLoadTable
dashlet.addOption("chart.custom.function", "function table1_load(param)
{table1_lazyLoadTable(param.name,4); openDashletModalPanel('table1')};\n");

// Wenn der DataTable NICHT als LazyLoad definiert ist, dann der Such-Funktion heißt
<DataTable_Id>_loadDataTable
dashlet.addOption("chart.custom.function", "function table1_load(param)
{table1_loadDataTable(); openDashletModalPanel('table1')};\n");

dashlet.addOption("chart.event.click", "table1_load");
```

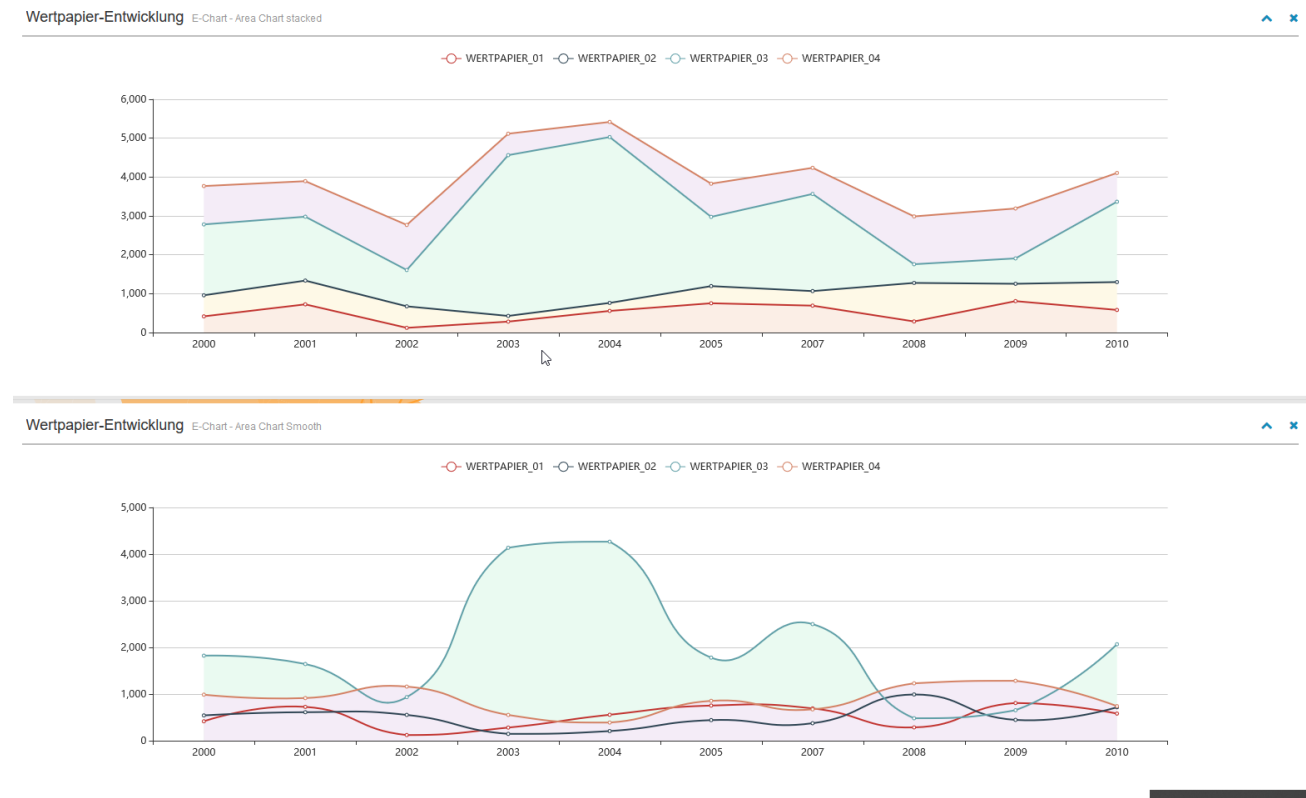
4.5.9 Flächen-/Stufendiagramm (Area Chart / Area-Step)

Einführung

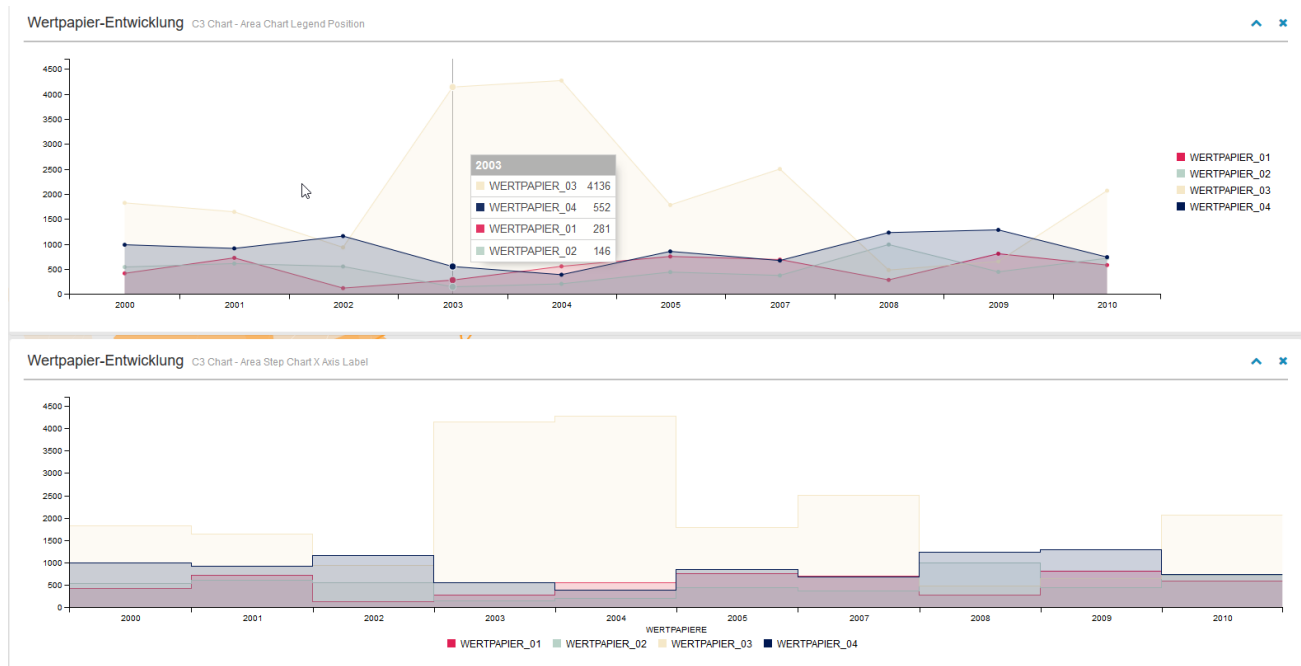
Das Flächen-/Stufendiagramm Dashlet könnte mit das C3Charts oder das ECharts platform entwickelt werden.

Es ist ähnlich wie Liniendiagramm, nur ausgefüllt . Das Stufendiagramm ist auch ähnlich wie Histogramm.

ECharts Beispiel



C3 Charts Beispiel



Dashlets erzeugen

Echart Chart

Die Flächen-/Stufendiagram Dashlet ist mit die folgende Code Block erstellt.

EChart Dashlet erzeugen

```
// DashletManager erzeugen
ExpertsDashlet.createDashletManager(reportContext, "DashletManager");

// Dashlet erzeugen
var dashlet = DashletManager.addDashletEChart("line", "Wertpapier-Entwicklung", "C3
Chart-Area Chart", true, mbr);
```

C3 Chart

Die Flächen-/Stufendiagram Dashlet ist mit die folgende Code Block erstellt.

EChart Dashlet erzeugen

```
// DashletManager erzeugen
ExpertsDashlet.createDashletManager(reportContext, "DashletManager");

// Parameter Dashlet erzeugen
var dashlet = DashletManager.addDashletC3Chart("area", "Wertpapier-Entwicklung", "C3
Chart-Area Chart", true, mbr);
```

Dashlet Optionen

Durch das Hinzufügen von Optionen können Dashlets individuell angepasst werden.

Dashlet Optionen

```
//Option für Flächen-/Stufendiagramm in Echarts
dashletObjArea.addOption("chart.series.i[1].areaStyle.color", "#fbee6");
dashletObjArea.addOption("chart.series.areaStyle.shadowBlur", "0");
dashletObjArea.addOption("chart.series.areaStyle.shadowOffsetX", "0");
dashletObjArea.addOption("chart.series.areaStyle.shadowOffsetY", "0");
dashletObjArea.addOption("chart.series.areaStyle.shadowColor", "#ccc");
dashletObjArea.addOption("chart.series.areaStyle.opacity", "1");
```

Verscheide Anzeige Format sind möglich für diese Dashlet (siehe [ECharts \(4\) \[Options\]](#)).

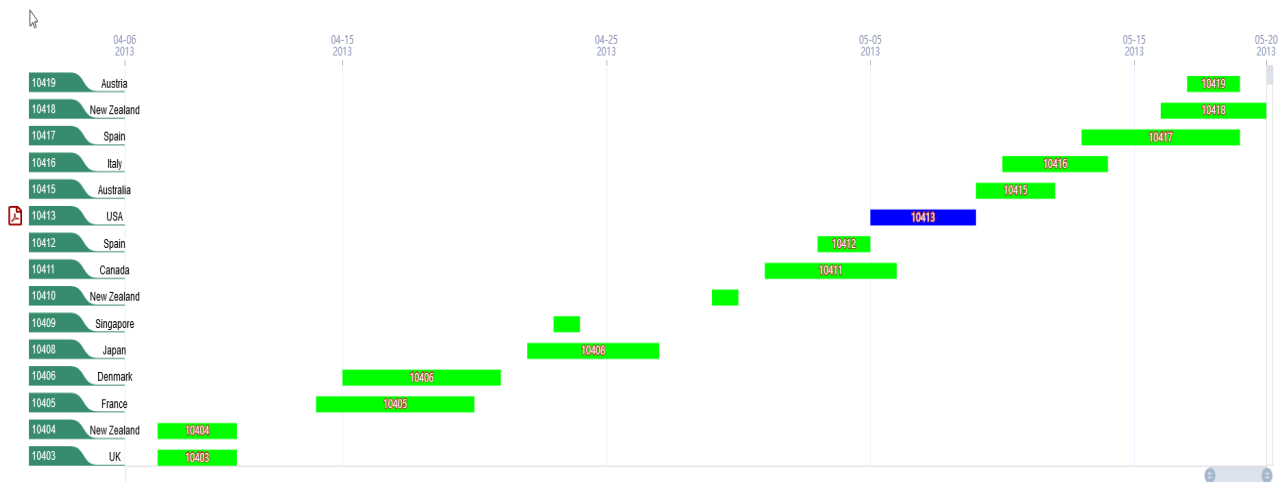
Für mehr Information für C3 Charts, siehe [C3-Charts](#).

4.5.10 Gantt Diagramm

Einführung

Das Gantt Diagramm ist mit der ECharts platform entwickelt werden.

Es wird verwendet, um zeitbasierte Serien und Aufgaben anzuzeigen Der Anwendungsfall: Projektmanagement



Dashlets erzeugen

Die Gantt Dashlet ist mit die folgende Code Block erstellt.

EChart Dashlet erzeugen

```
// DashletManager erzeugen
ExpertsDashlet.createDashletManager(reportContext, "DashletManager");

// Set the dataType of the start and end date columns
mbr.getColumnDefinition("audit_start_date").setDataType("timestampsec");
mbr.getColumnDefinition("audit_end_date").setDataType("timestampsec");

//Set the keyname attribute for the columns
mbr.getColumnDefinition("counter"
).addAdditionalInfo(DashletManager.MBR_DASHLET_KEYNAME,"category-index");
mbr.getColumnDefinition("audit_start_date"
).addAdditionalInfo(DashletManager.MBR_DASHLET_KEYNAME,"time-start");
mbr.getColumnDefinition("audit_end_date"
).addAdditionalInfo(DashletManager.MBR_DASHLET_KEYNAME,"time-end");
mbr.getColumnDefinition("color"
).addAdditionalInfo(DashletManager.MBR_DASHLET_KEYNAME,"Color");

// Dashlet erzeugen
var dashlet = DashletManager.addDashlet("gantt", "", "", true, mbr);
```

Dashlet Optionen

Das Gantt-chart kann mit die folgende Optionen angepasst werden.

```
dashlet.addOption("chart.series.function","[]");
dashlet.addOption("chart.predefined.xAxis","[]");
dashlet.addOption("chart.custom.init.function" , "chart1_LazyLoad(chart1_dataSet,
chart1_dataSet_dim);");
dashlet.addOption("chart.gantt.column.categoryname", "2"); //Spalte für CategoryName
(Information angezeigt auf das Linke Grüne-Tab)
dashlet.addOption("chart.gantt.column.categorygroup", "3"); //Spalte für
CategoryGroup (Information angezeigt neben das Linke Grüne-Tab)
dashlet.addOption("chart.gantt.column.barinfo", "1"); // Splate-Information ist auf
das Balken angezeigt
dashlet.addOption("chart.gantt.column.barcolor", "6"); //Spalte-Information für die
Farben des Balkens anzuzeigen
```

Hinweis : Für die folgende Splate, sollten die Index in das DataSet von 0 beginnen

- chart.gantt.column.barcolor
- chart.gantt.column.barinfo

, während für die andere Spalte, sollten die Index in das DataSet von 1 beginnen

- chart.gantt.column.categoryname
- chart.gantt.column.categorygroup
- chart.gantt.column.iconinfo

FAQ

Wie kann das Gantt-chart als ModalBox (popup) geöffnet werden?

Das Gantt-chart kann mit die folgende Optionen als ModalBox (popup Fenster) geöffnet werden.

Schritt 1 : GanttChart Dashlet erstellen

```
var dashlet = DashletManager.addDashletEChart("gantt" , "" , "" , true ,
mbrGantt); //Dashlet (GanttChart) erstellen
dashlet.setId("chart1"); // ID des Charts

// Modal Box options
dashlet.addOption("modal.box", "true");
dashlet.addOption("box.modal.size","xl");

//ModalBox Function
dashlet.addOption("chart.custom.openpanel.function" , "var _temp_dataSet =
chart1_dataSet;\n" +
    "if (typeof chart1_filtered_DataSet !== 'undefined' &&
chart1_filtered_DataSet !== null) _temp_dataSet = chart1_filtered_DataSet;\n" +
    "chart1_LazyLoad(_temp_dataSet, chart1_dataSet_dim);");

//init function should be disabled
// dashlet.addOption("chart.custom.init.function" , "chart1_LazyLoad(chart1_dataSet,
chart1_dataSet_dim);");
```

Schritt 2 : GanttChart Anruf definieren

Und in ein andere Dashlet, von wo die Gantt-Chart geöffnet werden soll (z.b. ein Button, oder ein Link), sollte die folgende Option haben.

```
// button for data
buttonbar = customDashlet.getDashlet().createDashletBar("DashletButtonBar", "TOP");
btn = buttonbar.addButton("All Data", false);
btn.addOption("button.onclick", "try{chart1.dispose();}catch(e)
{};chart1_filtered_DataSet = dashlet_sortDataSet(chart1_dataSet, 3,
'desc');openDashletModalPanel('chart1');");
btn = buttonbar.addButton("Data USA");
btn.addOption("button.onclick", "try{chart1.dispose();}catch(e)
{};chart1_filtered_DataSet = dashlet_getDataRowsFromDataDef(chart1_dataSet,
chart1_dataSet_dim, 'USA', 'COUNTRY');openDashletModalPanel('chart1');");
btn = buttonbar.addButton("Data Germany");
btn.addOption("button.onclick", "try{chart1.dispose();}catch(e)
{};chart1_filtered_DataSet = dashlet_getDataRowsFromDataDef(chart1_dataSet,
chart1_dataSet_dim, 'Germany', 'COUNTRY');openDashletModalPanel('chart1');");
```

Wie kann Tooltip für das Gantt-chart angepasst werden?

Die tooltip kann mit die folgende Optionen angepasst werden.

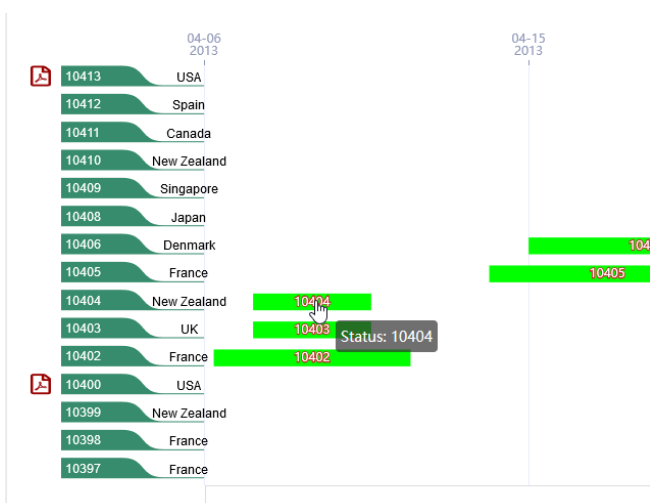
```
// tooltip options
dashlet.addOption("chart.tooltip.show" , "true");
dashlet.addOption("chart.tooltip.formatter.object" ,
    function (params) {
        var test = '';
        // display tooltip only for the StatusBars inside the

Gantt chart

        if (params.seriesName == 'cmganttdiagramm') {
            test += "Status: " + params.value[1] +

"<br>";

        }
        return test;
    }
);
```



Wie kann Datums-Wert für das Gantt-chart formatiert werden?

Die Datums-Wert kann mit die folgende Optionen formatiert werden.

```
dashlet.addOption("chart.xAxis.0.axisLabel.color" , "#929ABA");
dashlet.addOption("chart.xAxis.0.axisLabel.inside" , "false");
dashlet.addOption("chart.xAxis.0.axisLabel.align" , "center");
dashlet.addOption("chart.xAxis.0.axisLabel.formatter.object" , function (val) {
    var test = '';
    //
    Display the format as DD.MM.YYYY
    var d = new
    Date(val);
    test +=
    d.getDate() + '.' + (d.getMonth()+1) + '.' + d.getFullYear();
    return test;
});
```

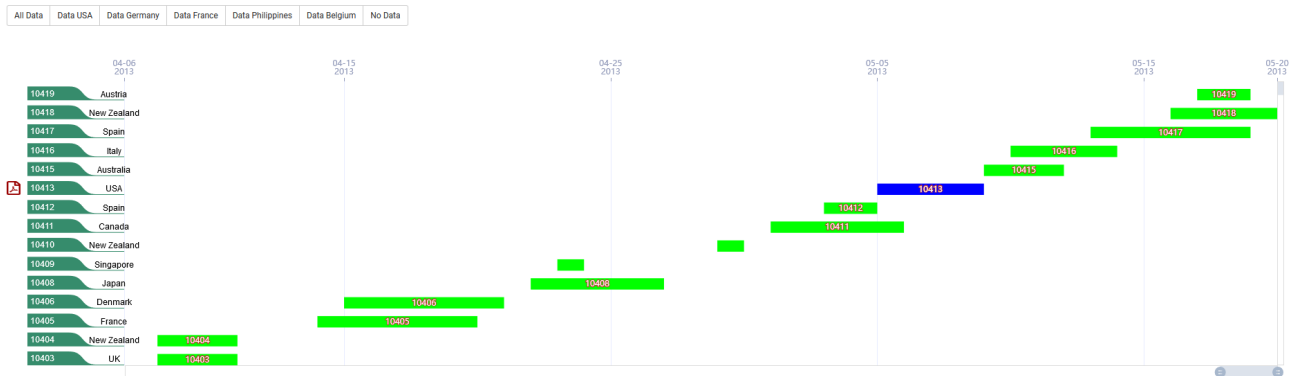
Wie kann Buttons für das Gantt-chart erstellt werden?

Die Buttons kann mit die folgende Optionen erstellt werden.

```
// Create buttons
buttonbar = mapDashlet.getDashlet().createDashletBar("DashletButtonBar", "TOP");
btn = buttonbar.addButton("All Data", false);
btn.addOption("button.onclick", "chart1_LazyLoad(chart1_dataSet,
chart1_dataSet_dim);");
btn = buttonbar.addButton("Data USA");
btn.addOption("button.onclick",
"chart1_LazyLoad(dashlet_getDataRowsFromDataSet(chart1_dataSet, 'USA', 2),
chart1_dataSet_dim);");
btn = buttonbar.addButton("Data Germany");
btn.addOption("button.onclick",
"chart1_LazyLoad(dashlet_getDataRowsFromDataSet(chart1_dataSet, 'Germany', 2),
chart1_dataSet_dim);");
btn = buttonbar.addButton("Data France");
btn.addOption("button.onclick",
"chart1_LazyLoad(dashlet_getDataRowsFromDataSet(chart1_dataSet, 'France', 2),
chart1_dataSet_dim);");
btn = buttonbar.addButton("Data Philippines");
btn.addOption("button.onclick",
"chart1_LazyLoad(dashlet_getDataRowsFromDataSet(chart1_dataSet, 'Philippines', 2),
chart1_dataSet_dim);");
btn = buttonbar.addButton("Data Belgium");
btn.addOption("button.onclick",
"chart1_LazyLoad(dashlet_getDataRowsFromDataSet(chart1_dataSet, 'Belgium', 2),
chart1_dataSet_dim);");
btn = buttonbar.addButton("No Data");
btn.addOption("button.onclick",
"chart1_LazyLoad(dashlet_getDataRowsFromDataSet(chart1_dataSet, 'No Data', 2),
chart1_dataSet_dim);");
```

Hinweis : Die onclick Function für jede Button hat die folgende Struktur

```
<CHART_NAME>_LazyLoad(dashlet_getDataRowsFromDataSet(<CHART_NAME>_dataSet, 'No Data', 2),
<CHART_NAME>_dataSet_dim);
```



Wie kann ein Sort Button erstellt werden in das Gantt-chart?

Ein Sort Button kann mit die folgende Optionen erstellt werden.

```
// Button for sorting
buttonbar = dashlet.getDashlet().createDashletBar("DashletButtonBar", "TOP");
btn = buttonbar.addButton('Sortierung nach Sparte, Land, Prüfungsstart');
btn.addOption("button.onclick",
"ganttChart_LazyLoad(dashlet_sortDataDef(gantChart_dataSet_selected,
gantChart_dataSet_dim, 'division', 'desc', 'name_de', 'desc', 'audit_start_date',
'desc'), gantChart_dataSet_dim); " );
```

Hinweis : Die Sort-Function für jede Button hat die folgende Struktur und kann bis 4 Spalte sortieren.

```
<CHART_NAME>_LazyLoad(dashlet_sortDataDef(<CHART_NAME>_dataSet_selected, <CHART_NAME>_dataSet_dim,
'<1COLUMN_NAME>', '<1COLUMN_SORTINGORDER>', '<2COLUMN_NAME>', '<2COLUMN_SORTINGORDER>',
'<3COLUMN_NAME>', '<3COLUMN_SORTINGORDER>', '<4COLUMN_NAME>', '<4COLUMN_SORTINGORDER>'),
<CHART_NAME>_dataSet_dim);
```

Wie kann ein PDF-Icon angezeigt werden für ein Zeile in das Gantt-chart?

Ein PDF-Icon kann angezeigt werden für die Zeilen in Gantt-chart, die ein bestimmt Condition hat, mit die folgende Optionen.

```
// Splate-Nummer (Diese Spalte ist benutzt für die Entscheidung ob die Icon angezeigt
werden soll oder nicht)
dashlet.addOption("chart.gantt.column.iconinfo", "3");

// Wert versucht in diese Spalte für die Entscheidung ob die Icon angezeigt werden
soll oder nicht
dashlet.addOption("chart.gantt.iconlink.check", "'USA'");

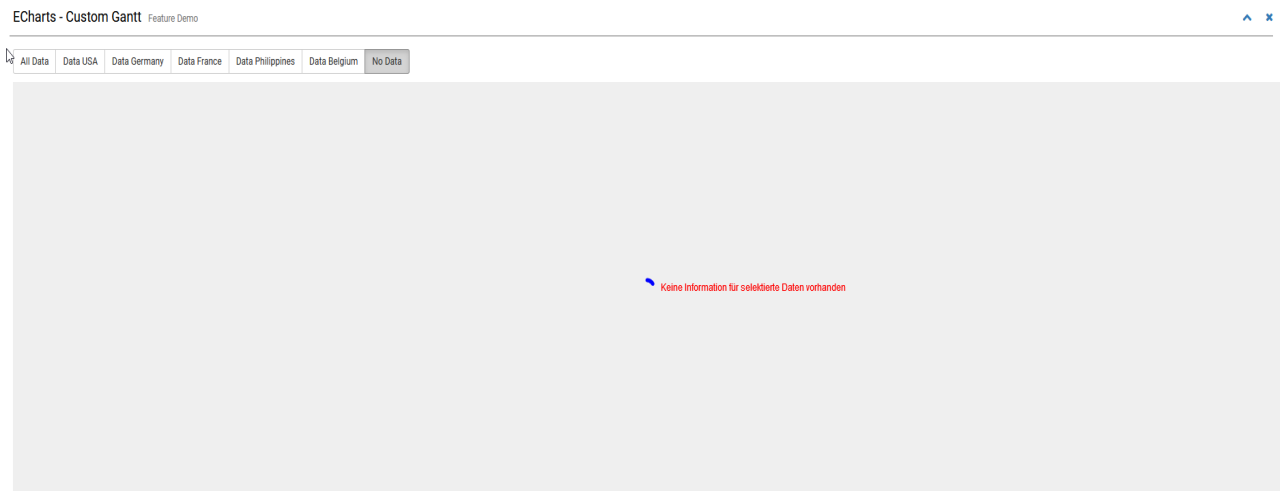
// SVG für die Icon
dashlet.addOption("chart.gantt.iconlink.path", "M369.9 97.9L286 14C277 5 264.8-.1
252.1-.1H48C21.5 0 0 21.5 0 48v416c0 26.5 21.5 48 48h288c26.5 0 48-21.5
48-48V131.9c0-12.7-5.1-25-14.1-34zM332.1 128H256V51.9l76.1 76.1zM48 464V48h160v104c0
13.3 10.7 24 24
24h104v288H48zm250.2-143.7c-12.2-12-47-8.7-64.4-6.5-17.2-10.5-28.7-25-36.8-46.3
3.9-16.1 10.1-40.6 5.4-56-4.2-26.2-37.8-23.6-42.6-5.9-4.4 16.1-.4 38.5 7 67.1-10
23.9-24.9 56-35.4 74.4-20 10.3-47 26.2-51 46.2-3.3 15.8 26 55.2 76.1-31.2 22.4-7.4
46.8-16.5 68.4-20.1 18.9 10.2 41 17 55.8 17 25.5 0 28-28.2 17.5-38.7zm-198.1
77.8c5.1-13.7 24.5-29.5 30.4-35-19 30.3-30.4 35.7-30.4 35zm81.6-190.6c7.4 0 6.7 32.1
1.8 40.8-4.4-13.9-4.3-40.8-1.8-40.8zm-24.4 136.6c9.7-16.9 18-37 24.7-54.7 8.3 15.1
18.9 27.2 30.1 35.5-20.8 4.3-38.9 13.1-54.8 19.2zm131.6-5s-5 6-37.3-7.8c35.1-2.6 40.9
5.4 37.3 7.8z");

// Farbe für die Icon
dashlet.addOption("chart.gantt.icon.color", "#990000");
```

Wie kann eine Information in das Gantt-Chart angezeigt werden, falls kein Daten vorhanden ist?

Eine Message kann angezeigt werden falls kein Daten vorhanden ist (z.b. für ein Filter / Button ist kein Daten zu zeigen in das Gantt-Chart).

```
dashlet.addOption("chart.gantt.nodata.text","Keine Information für selektierte Daten
vorhanden");
dashlet.addOption("chart.gantt.nodata.color","#0000ff");
dashlet.addOption("chart.gantt.nodata.textColor","#ff0000");
dashlet.addOption("chart.gantt.nodata.maskColor","#efefef");
dashlet.addOption("chart.gantt.nodata.zlevel","0");
// dashlet.addOption("chart.gantt.nodata.fontSize","12");
dashlet.addOption("chart.gantt.nodata.showSpinner","true");
dashlet.addOption("chart.gantt.nodata.spinnerRadius","20");
dashlet.addOption("chart.gantt.nodata.lineWidth","8");
```



Wie kann die Axen des Gantt-charts unterschiedet werden (z.b. für ein Click event)?

Die unterschiedliche Axen kann Namen gegeben werden, und beim Click-Funktion geprüft werden.

```
dashlet.addOption("chart.series.n[ganttitem].name", "cmganttdiagramm"); // Namen für
die Ganttitems (Balken in das Gantt-Bereich)
dashlet.addOption("chart.series.n[ganttyaxis1].name", "axisLabels"); //Namen für die
Axen-Labels
dashlet.addOption("chart.series.n[ganttyaxis2].name", "pdficon"); //Namen für die
PDF Icons Axis
```

Wie kann ein Onclick Funktion für das Gantt-chart erstellt werden?

Ein Onclick-Funktion kann mit die folgende Optionen gesetzt werden.

```
dashlet.addOption("chart.custom.function", "function click_gantt_test(param)
{console.log(param);if(param.seriesName == 'pdficon') alert('Click PDF Icon: ' +
param.data);}");
dashlet.addOption("chart.event.click", "click_gantt_test");
```

Wie kann die Balken in das Gantt-chart in custom-Farben angezeigt werden?

Die Balken können mit die verschiedene Farben angezeigt werden, wenn ein Spalte für die Farben in das MemoryBuffer definieret sind.

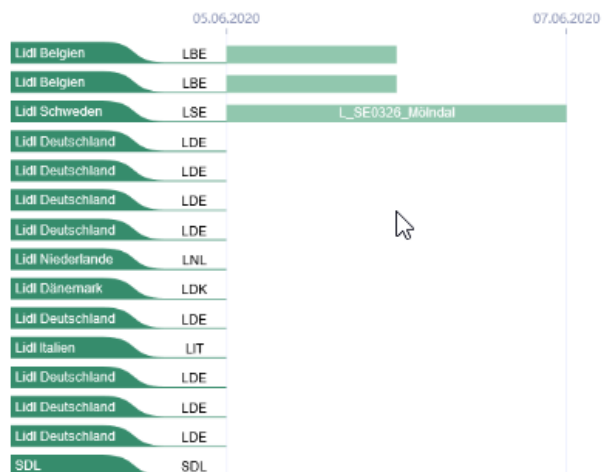
```
//KeyName für die Spalte definieren
mbr.getColumnDefinition("color"
).addAdditionalInfo(DashletManager.MBR_DASHLET_KEYNAME,"Color");

//Spalte-Information für die Farben des Balkens anzuzeigen
dashlet.addOption("chart.gantt.column.barcolor", "6");
```

Wie kann die Axis Label in das Gantt-chart angepasst werden?

Die AxisLabel können mit die verschiedene Option angepasst werden.

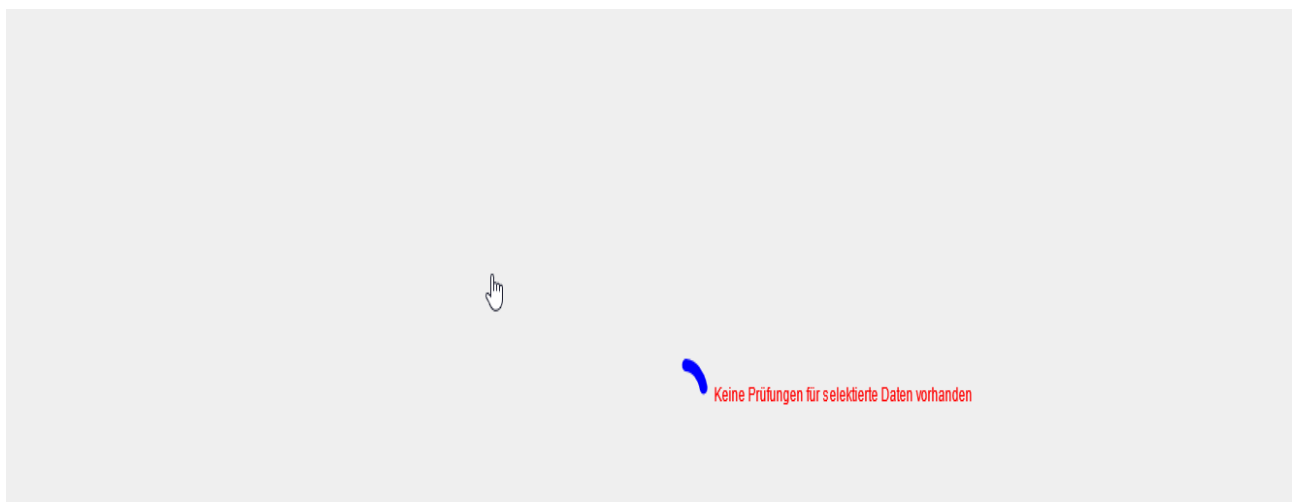
```
//Adjusting the GreenLabel and Grid positions
dashlet.addOption("chart.gantt.axis.width", "200"); // Axis width, to make the Green
Label bigger
dashlet.addOption("chart.gantt.axis.categorygroup.text.style.x", "170"); //
dep_short_name starting position
dashlet.addOption("chart.gantt.grid.left", "230"); //grid left position, so as to
shift the grid
```



Was muss bei Echarts 5 Migration berücksichtigt werden?

Bei der NoData Darstellung, der folgende Option **raus** nehmen.

```
dashlet.addOption("chart.gantt.nodata.zlevel","0");
```



4.5.11 Graph / Chord Diagramm

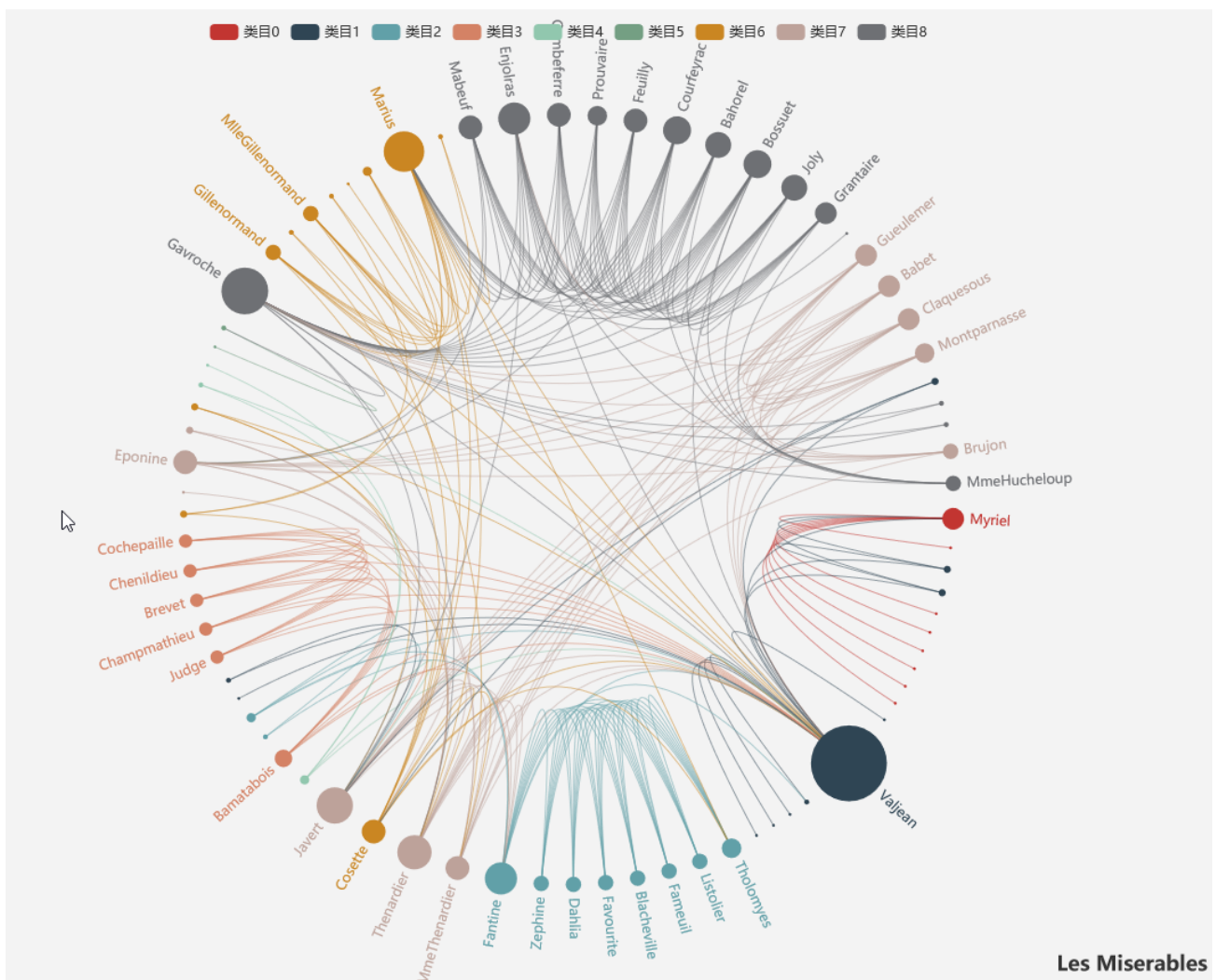


Information

Dieser Charttyp ist derzeit noch nicht im Dashlet-Plugin verfügbar. Bitte kontaktieren Sie unseren [Support](#), wenn Sie an der Nutzung dieser Darstellungsform interessiert sind.

Einführung

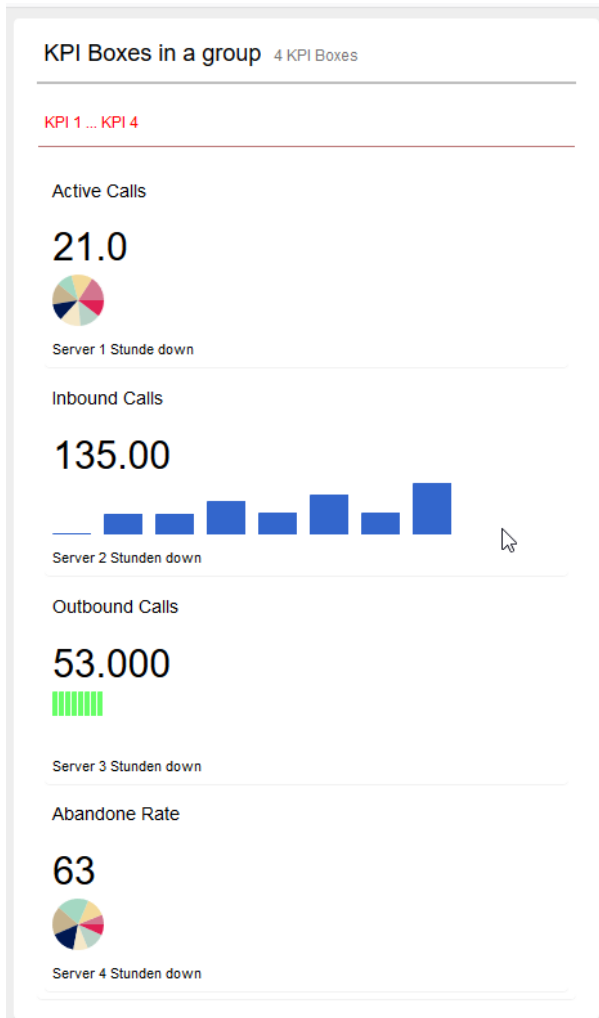
Es zeigt die Abhängigkeiten und Korrelationen an. Die Anwendungsfall: Anzeige von Produkt- oder Kundeninformationen und Abhängigkeiten



4.5.12 Gruppen-Dashlet

Einführung

Das Gruppen-Dashlet ermöglicht die Möglichkeit für mehr Dashlets in eine Gruppe darzustellen.



Dashlets erzeugen

Die Gruppen-Dashlet ist mit die folgende Code Block erstellt.

EChart Dashlet erzeugen

Dashlets erzeugen

```
// DashletManager erzeugen
ExpertsDashlet.createDashletManager(reportContext, "DashletManager");

// Dashlet erzeugen
var dashlet = DashletManager.addDashlet("DashletHtmlGroup", "Group 1", "", false,
null);

// child dashlets erzeugen
dashlet.addOption("childid", "kpi_boxes31");
```

Durch das Hinzufügen von Optionen können Dashlets individuell angepasst werden.

Dashlet Optionen

Optionen

```
// Breite des Dashlets setzen
dashlet.setResponsiveGrid("2-4-6-12");
```

FAQ

Wie kann eine Gruppen-Dashlet in ein Haupt-Gruppe gesetzt werden?

Die Verpackung von einer Gruppe in eine andere Gruppe kann mit der folgenden Option gesetzt werden.

Die Childs werden mit rekursiven Aufrufen gerendert. Damit kann man Child-Dashlets über mehrere Ebenen erzeugen.

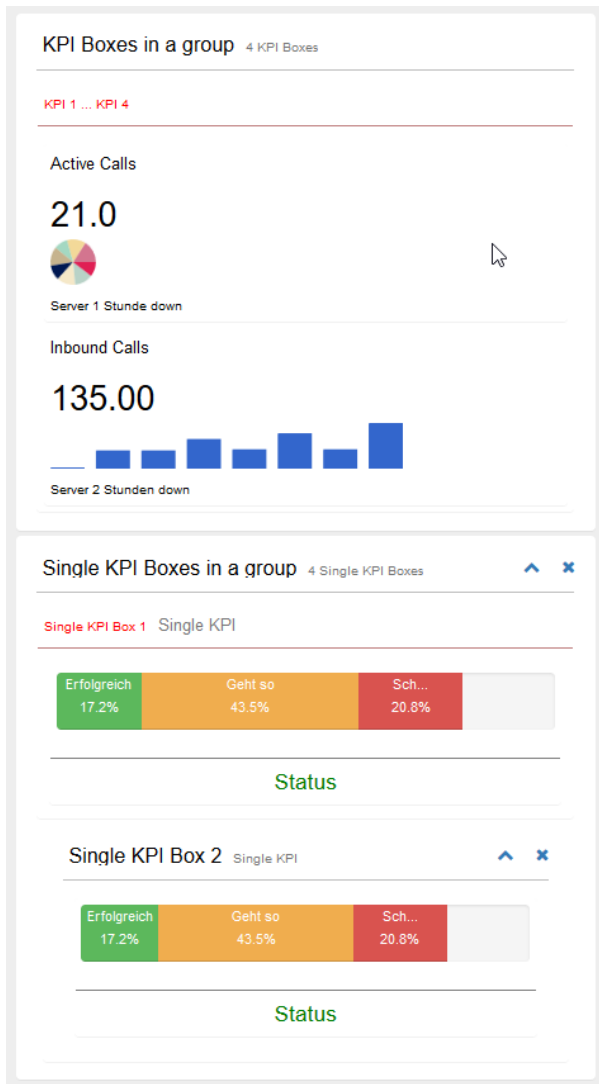
Bitte vorsichtig und keine Zirkel-Beziehungen bauen!

```
// Option für die Viewpoint zu setzen
dashlet = DashletManager.addDashlet("DashletHtmlGroup", "Main Group 1", "", false,
null);
dashlet.setId("kpi_main_group");
dashlet.setResponsiveGrid("3-4-6-12");
dashlet.addOption("childid","kpi_group2");
dashlet.addOption("childid","kpi_group1");

dashlet = DashletManager.addDashlet("DashletHtmlGroup", "Group 1", "", false, null);
dashlet.setId("kpi_group1");
dashlet.setResponsiveGrid("12-12-12-12");
dashlet.addOption("childid","kpi_boxes4");
dashlet.addOption("title","KPI Boxes in a group");
dashlet.addOption("subtitle","4 KPI Boxes");
dashlet.addOption("box.content.style","padding: 0px;");
dashlet.addOption("box.closelink.show", "hidden");
dashlet.addOption("box.collapselink.show", "hidden");
dashlet.addOption("child.box","true");

dashlet = DashletManager.addDashlet("DashletHtmlGroup", "Group 2", "", false, null);
dashlet.setId("kpi_group2");
dashlet.setResponsiveGrid("12-12-12-12");
dashlet.addOption("childid","kpi_box_sb");
dashlet.addOption("childid","kpi_box_sb_2");
dashlet.addOption("childid","kpi_box_sb_3");
dashlet.addOption("childid","kpi_box_sb_4");
dashlet.addOption("title","Single KPI Boxes in a group");
dashlet.addOption("subtitle","4 Single KPI Boxes");
dashlet.addOption("box.content.style","padding: 0px;");
dashlet.addOption("child.box","true");
```

Somit, können die Dashlets unter- einander angezeigt werden.



4.5.13 Heatmap

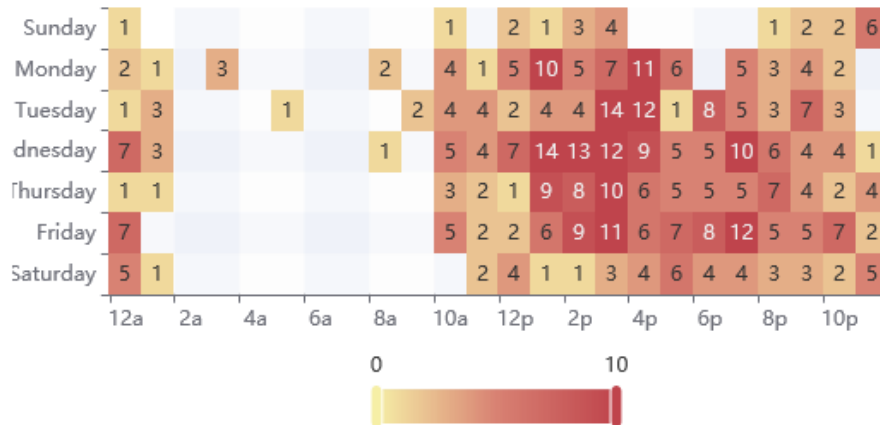


Information

Dieser Charttyp ist derzeit noch nicht im Dashlet-Plugin verfügbar. Bitte kontaktieren Sie unseren [Support](#), wenn Sie an der Nutzung dieser Darstellungsform interessiert sind.

Einführung

Heatmap verwendet hauptsächlich Farben, um Werte darzustellen.



In ein Geo-Map, ist es benutzt für:

- Anzeige des kumulierten Auftretens eines Wertes auf einer geografischen Karte.
- Anwendungsfall: grafische Darstellung von Kundenadressdaten
- Voraussetzung: Geocodierte Daten (Zuordnung von Adresse zu Geokoordinaten)



4.5.14 HTML Dashlet

Einführung

Mit ein HTML Dashlet, kann der Report-Developer eigene HTML oder Javascript Code in das Dashboard rendern.

Dashlets erzeugen

Das HTML Dashlet ist mit die folgende Code Block erstellt.

HTML Dashlet erzeugen

```
// DashletManager erzeugen
ExpertsDashlet.createDashletManager(reportContext, "DashletManager");

// Dashlet erzeugen
var dashlet= DashletManager.addDashlet("DashletHTML", "", "", false, null);
```

Dashlet Optionen

Das Gantt-chart kann mit die folgende Optionen angepasst werden.

```
// HTML Code
var customHTML = "";
customHTML += "<table id=\"Dash2\" class=\"table table-striped table-bordered stripe compact row-borders\" width=\"100%\"></table>";
// render the Code in HTML File
dashlet.addOption("html", customHTML);

// Javascript Code
var customJS = "";
// function to call the adhoc report from dashboard
customJS += "    function callAdhocBericht() {\n";
customJS += "        paramDateFrom = startDate ; \n";
customJS += "        paramDateTo = endDate; \n";
customJS += "        try{openJobCreatorDialog([\n";
customJS += "            {name:'--__report', value:'/kurzcheck/Kumulschadenfälle.rptdesign'},\n";
customJS += "            {name:'--__outputFormat',\n";
customJS += "value:'PDF,HTML'},\n";
customJS += "            {name:'--paramDateFrom',\n";
customJS += "value:paramDateFrom},\n";
customJS += "            {name:'--paramDateTo',\n";
customJS += "value:paramDateTo}}]); \n";
customJS += "        }catch(e){};\n";
customJS += "    } \n\n";
// render the Code in HTML File
dashlet.addOption("javascript", customJS + "\n\n");
```

4.5.15 Illustrationsdiagramm (Piktogramm)



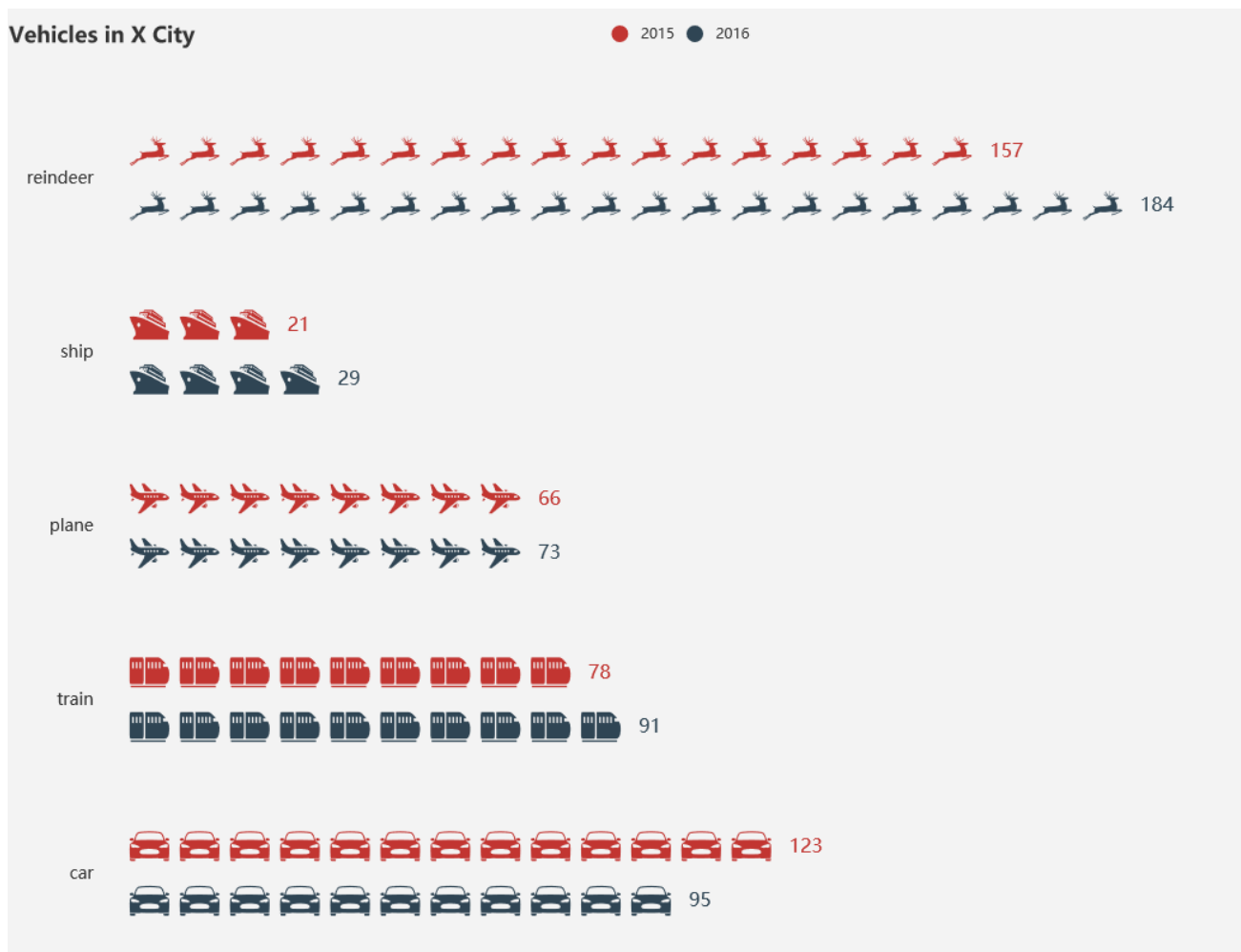
Information

Dieser Charttyp ist derzeit noch nicht im Dashlet-Plugin verfügbar. Bitte kontaktieren Sie unseren [Support](#), wenn Sie an der Nutzung dieser Darstellungsform interessiert sind.

Einführung

Das Piktogramm ist eine Art Balkendiagramm, bei dem anstelle eines rechteckigen Balkens benutzerdefinierte Glyphen (wie Bilder, SVG PathData) verwendet werden können.

Diese Art von Diagramm wird normalerweise in Infografiken verwendet.

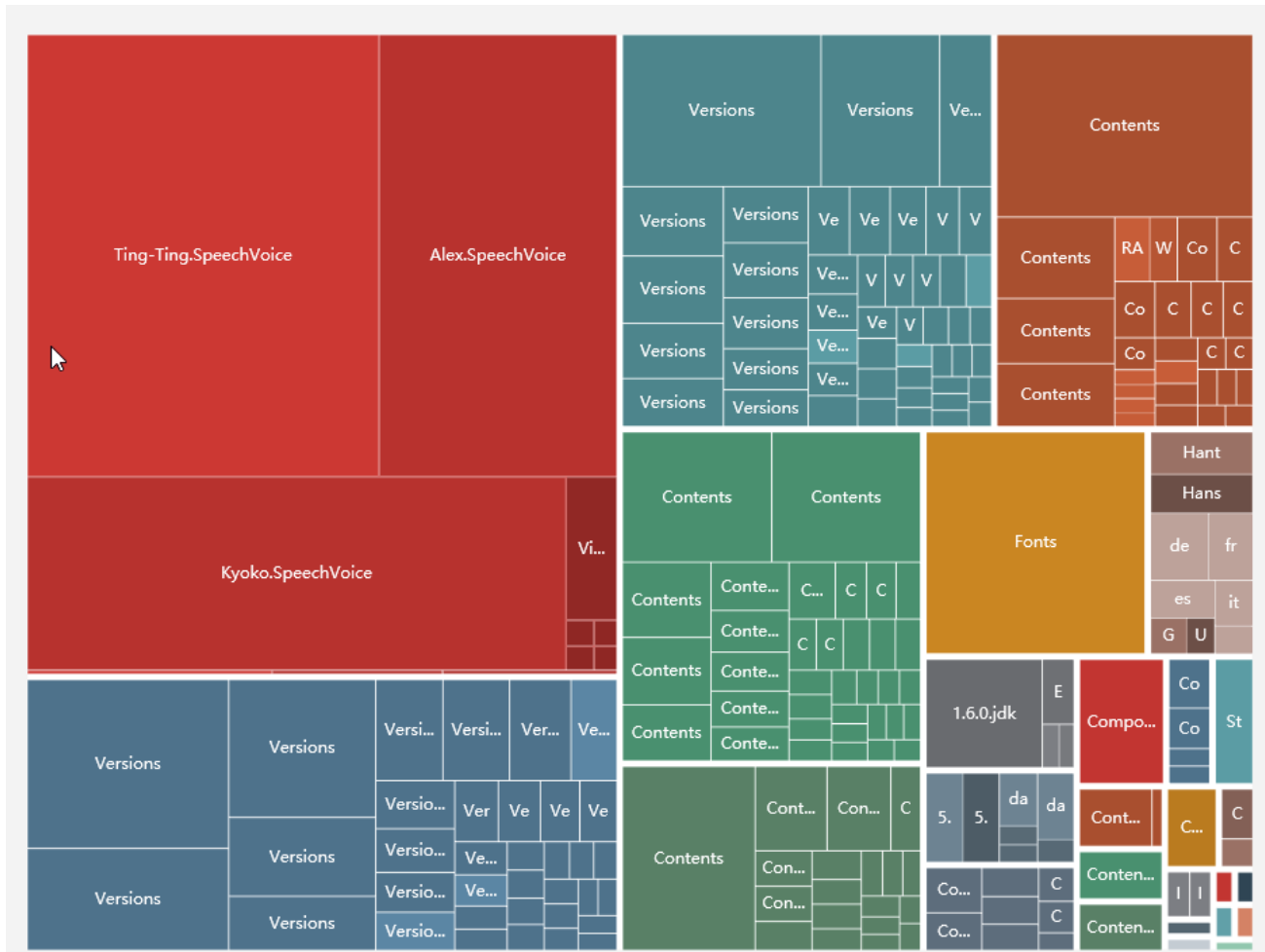


4.5.16 Kacheldiagramm (Treemap)

Einführung

Treemap ist eine gebräuchliche Methode, um "hierarchische Daten" oder "Baumdaten" darzustellen.

Die Größe repräsentiert proportionale Werte. Die Gruppierung erfolgt durch Farben und es besteht die Möglichkeit eines Drilldowns.



Dashlets erzeugen

Die Treemap Dashlet ist mit die folgende Code Block erstellt.

EChart Dashlet erzeugen

```
// DashletManager erzeugen
ExpertsDashlet.createDashletManager(reportContext, "DashletManager");

// Rename the displayName
mbr.getColumnDefinition("dep_id").setDisplayName("hierachy-id");
mbr.getColumnDefinition("parentId").setDisplayName("parent-id");
mbr.getColumnDefinition("depName").setDisplayName("NAME");

//Set the keyname attribute for the above columns
mbr.getColumnDefinition("depName").addAdditionalInfo("keyname","NAME");
mbr.getColumnDefinition("parentId").addAdditionalInfo("keyname","parent-id");
mbr.getColumnDefinition("dep_id").addAdditionalInfo("keyname","hierachy-id");

// Dashlet erzeugen
var dashlet = DashletManager.addDashlet("treemap", "", "", true, mbr);
```

4.5.17 Kartendiagramm (Geo/Map Diagram)

Einführung

Das Kartendiagramm Dashlet könnte mit das ECharts platform entwickelt werden. Mehrere Gebiete mit Drilldown möglich (Welt, EU, Amerika, Asien, Afrika, Australien, Deutschland, deutsche Bundesländer, deutsche Landkreise, ...)

Hinweis: Die Daten/Ländercode-Zuordnung muss übereinstimmen! Andernfalls werden keine Daten angezeigt.

World Map Classic Models Customer Count



Dashlets erzeugen

Echart Chart

Die Kartendiagramm Dashlet ist mit die folgende Code Block erstellt.

EChart Dashlet erzeugen

```
// DashletManager erzeugen
ExpertsDashlet.createDashletManager(reportContext, "DashletManager");

// Pie Dashlet erzeugen
var dashlet = DashletManager.addDashletEChart("map" , "World Map" , "Classic Models
CustomerCount", true , mbr);
```

Dashlet Optionen

Durch das Hinzufügen von Optionen können Dashlets individuell angepasst werden.

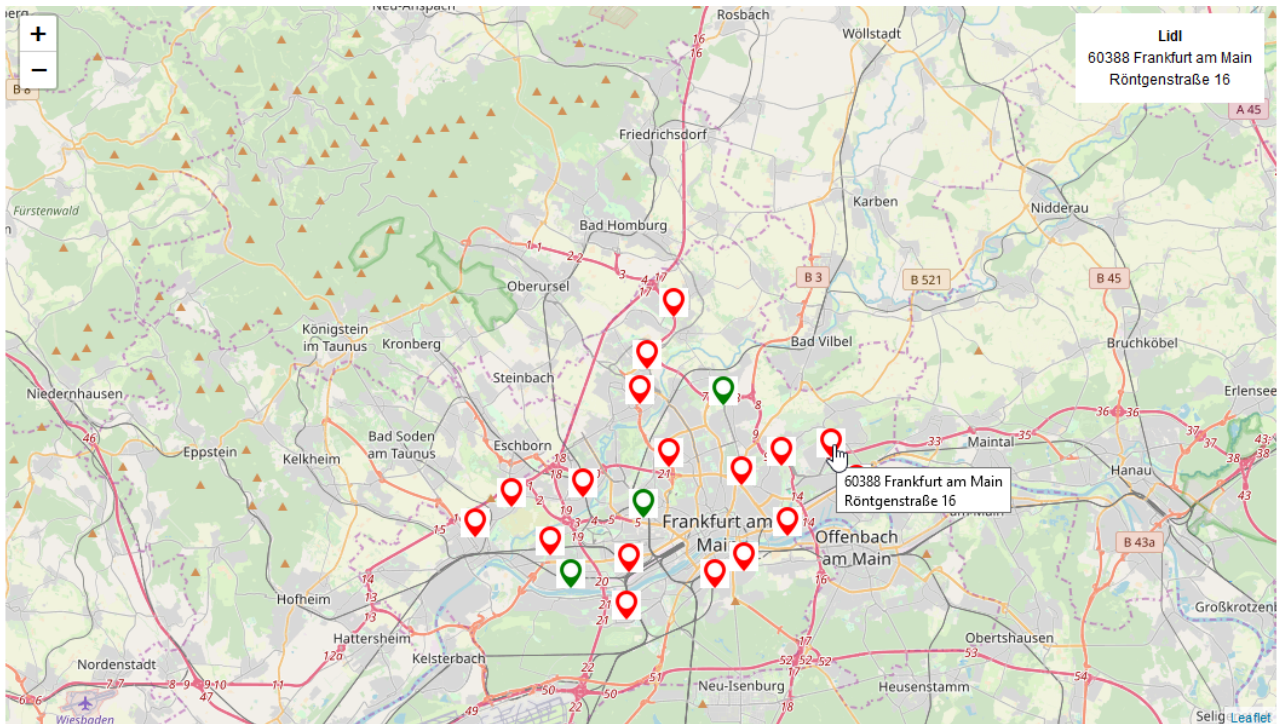
Dashlet Optionen

```
//Option für Donut chart in ECharts
dashlet.addOption("chart.type", "map");
dashlet.addOption("chart.map.type", "world");
```

4.5.18 Karten Diagramm (Leaflet)

Einführung

Das Geodiagramm ermöglicht das Zeichnen der Standorte auf einer Karte (z. B. aus OpenStreetMaps).



Dashlets erzeugen

Hinweis: Die Spalten-Namen sollte angepasst werden und die Metatyp definieren.

```
mbr.getColumnDefinition("X_COORDINATE_L").setDisplayName("map-lat"); //latitude
mbr.getColumnDefinition("Y_COORDINATE_B").setDisplayName("map-lng"); //longitude

// Set the metaType of the data (mbrMapDashlet) as 'mapmarkerarray', which is used to
// render the data in mbrMapDashlet as a Marker for the MapDashlet
mbr.addAdditionalInfo(DashletManager.MBR_DASHLET_METATYPE,"mapmarkerarray");
```

Durch das Hinzufügen von Optionen können Dashlets individuell angepasst werden.

Die GeoMap Dashlet ist mit die folgende Code Block erstellt.

EChart Dashlet erzeugen

```
// DashletManager erzeugen
ExpertsDashlet.createDashletManager(reportContext, "DashletManager");

// Dashlet erzeugen
var dashlet = DashletManager.addDashlet("DashletLeaflet", "Karte", "", true, mbr);
```

Dashlet Optionen

Durch das Hinzufügen von Optionen können Dashlets individuell angepasst werden.

Dashlet Optionen

```
// Option für Höhe der Außer-Box
dashlet.addOption("box.height", "700");
// Option für Höhe der Leaflet
dashlet.addOption("leaflet.height", "600");
// Option für die Aufruf der OpenStreetMap
dashlet.addOption("leaflet.map.tileLayer", "https://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png");
```

FAQ (Leaflet)

Wie kann die Markers von Anfang darstellt werden?

Die Markers kann mit die folgende Option von Anfang des Dashboards angezeigt werden:

```
dashlet.addOption("leaflet.markers.autoload" , "true");
```

Wie kann die Viewpoint des GeoMaps gesetzt werden?

Die Viewpoint kann mit die folgende Option gesetzt werden:

```
// Option für die Viewpoint zu setzen
dashlet.addOption("leaflet.map.viewpoint", "[50.101694,8.631921]");
```

Wie kann die ZoomLevel des GeoMaps gesetzt werden?

Die ZoomLevel kann mit die folgende Option gesetzt werden:

```
// Option für die Zoom-level des Charts
dashlet.addOption("leaflet.map.zoom", "5");
```

Wie kann ein Click-Event in das GeoMap gesetzt werden?

Ein Click-Event kann mit die folgende Option gesetzt werden.

```
// Option für Click-Event
dashlet.addOption("leaflet.marker.event.click", "alert('Click-Test');");
```

Wie kann die Icons des GeoMaps gesetzt werden?

Belibige Icons kann mit die folgende Option gesetzt werden. Die Icon für die Active und Inactive Punkte könnte auch gesetzt werden.

```
// Optionen für Icons zu setzen
// green marker
dashlet.addOption("leaflet.marker.icon.1.iconUrl", "data:image/
png;base64,iVBORw0KGgoAAAANSUheUgAAABAAAAAQCAAAAAf8/9hAAAACXBIWXMAAA7EAAAOxAGVKw4bAA
ABS0LEQVQ4jX3SsUtXURQH8I/
i0BSIj4Y31ia04V3DtSVoCJHIQIjgt4aQc0PgX9Dk4hYVRAGS4hYPpMmhlgiHS8glmiRCwuF3n753fbzvcuB7
vt/vPee9M6NA3VRz2MBjLGX6G3bxJob0t6ufKcy38A13y+CMY9yPIZ20xGzx8sds/olVLGAeD/
MUS9irm+pG65vrpK9jGScIMaTTTu9d3VQH+IJFTLDdmwBruW4VZhBD+oMXhbYX0H6wz9dXv0Tba7W9gP8jxhL
nQwFHua6MGNve16GA3Vxf1U1Vlc66qW7idaG9uoP8G7/jNn5gE/
umq93L5kX8wp0Y0lkvIic8wPuRFeBpDGLnaAUxpA9402Lex06XmB0QPTM9phKn+fUeeS0ghvTb9HTPOvQ/
PIohxVI/NIEY0hGeuLqN5zGkwyHtKOqmmtRN9XJMcwEU+llNYjeGUAAAAABJRUErkJggg==");
dashlet.addOption("leaflet.marker.icon.1.iconSize", "[24, 24]");
dashlet.addOption("leaflet.marker.icon.1.iconAnchor", "[12, 12]");
dashlet.addOption("leaflet.marker.icon.1.popupAnchor", "[-3, -76]");
// dashlet.addOption("leaflet.marker.icon.1.shadowUrl", "");
// dashlet.addOption("leaflet.marker.icon.1.shadowSize", "");
// dashlet.addOption("leaflet.marker.icon.1.shadowAnchor", "");

//red marker
dashlet.addOption("leaflet.marker.icon.2.iconUrl", "data:image/
png;base64,iVBORw0KGgoAAAANSUheUgAAABAAAAAQCAAAAAf8/9hAAAACXBIWXMAAA7EAAAOxAGVKw4bAA
ABAULEQVQ4jZWSvWpCQRCFP8ItJEgQEQkWPkUeQFKnELGQkCIEiyBiLSqv5B0kSpVCFC1ErERSBbFKkSKoOSk
yF5bL3NUsDAT7znd29gcyQ5AIHgVvgk+rkWAgKGT9WbgqGAuUU3NBPQ90bCcJ1oK0oCwoCVqCpWkLtxNB1wzv
gqqjl4KQJy/g1cS7yBFvzDPzxK2Jl5GAgnl26dpZoP/
kgc7YewETm68jYKpNvfZugxeo0PqFPaMEPS8gEazMsBK0DSra5S1M+xCcu/
0JmpFP1NZ95IggGEbgF0Xpv4CyfaYsvBHUjvFpyJXgK4C/BY2T4CCKLThYwM0/
4CCkL3i0eX4BbI64pl9t1I8AAAAASUVORK5CYII=");
dashlet.addOption("leaflet.marker.icon.2.iconSize", "[24, 24]");
dashlet.addOption("leaflet.marker.icon.2.iconAnchor", "[12, 12]");
dashlet.addOption("leaflet.marker.icon.2.popupAnchor", "[-3, -76]");

// create icons blue
```

```
dashlet.addOption("leaflet.marker.icon.3.iconUrl", " data:image/
png;base64,iVBORw0KGgoAAAANSUHEUgAAADAAAAAwCAYAAABXAvmHAAAAABmJLR0QA/wD/
AP+gvaeTAAAFBkLEQVRogdWZTWxUZRSgn3NvW2ojgmFhAEFo00MDqIFpLUaiRhCFxBijxYUJQaFMS/
kRNYKJMWbcmEjYKNIhdmFMWBQWwgJMIGAgUFum1UCK2E75C1BxYUFSpS1zjwugmNI7c+7MYOK7uznvOed9833
3+7kX/
ueQXBWq2JKYRB4LFZ2LMg0YDIwFFOUKwjLrbVeRw8nB5N6fVpddyKXfrA1U1CdeVHq98DTgGNM80IPi8NmxF
H92FTP2Mat4R8DT2QjAGgSRz/
K1EhgA5FY1xjU2yLwRiYnFYWoflNQ4K0+sqzSwqC8IOTK+q5QEm83UBZInREKJzUpL7fVlSasOWYDkVjXo6Le
PuChjNSZId1e0pvXVhf+xcS2kGZ9cWqCm+82Aw9npc0K5ZKbR2VzdehCOMraVSMSi+e7+e4u/
ivxAMKEZJKdz248mJeOmn7Z0zEfAuW50BUQldfGT/
wgHSnlFIReuaLer8ChQEanxC0AUf29xf2nQUYdb1oCp70V2Q5MDNArb589cJNtdMu+hFSD5F667GL71dhXWt
3aYyN4g2LtQPtVY365ZmeRA3KZqDAULPoBu77wFo/gu8IPNVwanT/
oHsRGG0TLwtbo6UHDVxmb00854juxWb16kBR38TjSx7vHSno+w70DzgLsIlH0bet4gHaaksPoPKukT6mok9ov
l/Q/yV2n0eNDU60/hbaZuQ0Yeq4kq3cnFppIcoLfjF/A6qzLMVV90sR5nxa7FgsSZQGwW8e84uLMMBUS/
E8lSx0k7LPSCzxC/
gbEB6wVP4r0XDeK0IuJcQ4cc5IFdAvkGojS1oqjy10Mz6S3+g15w76BXwNPKPK7ScSg08ko4i4k82Wykeqrxd
AoKap4aHW1WoEmHN9p1qKEeCISQKyvKpRXa0QIVQ1qqvCchNZ9KhfyH8EHDls1DLzTE+ixsgdwumezjqB6Rau
5/lr8TVw5boeEOgxqVE2V9R3zTNxgYptHfNFZZOR/kf/
uAHfXd7XQGJNqN+DncYmBYq3p3xrYlWq6VTVqG4klrFGPdkD5Jsqi+xoXzxjwDecKjcS66gUlR9Nje6g/
eYOK/
v+9vrPATznJjqioguAZdZpMwT1Ku0101r8wmnX4fL6jsMgcwM1zR00xWtCz6QipL+RefJ5zuQEhEda98S0E5b
Xdx4FnsxaUTC0xK0lcxDRVCTTp0BBN+RGkx0qsiGdeDaa0FYTPgTsylqVHd9ZL0jWj7Go0GuAEa910Uaf4ybf
sZLNBlqjJedV+DQzTXYoFNJSXXbGyjbABjdfWET0BRYLR1x50rmIAmBz/IVX3UUqyM/
Y7zwB0Cv4zizw1aUdARJCjQCAMdWhk+LyHtB89JBVNYFFQ9Z/OAor+/
cDbyUaf4wfb+Pl16yLJvDEXgEbsN13WqQ7kzz70C6Xdddmol4yMJAc3XxZQd9FfA9KRow6HnJ15uriy9nWiBj
AwAtNaEmBP0aPRwqrG1b0c16cRoR0fnNWh7rbEB5K2Dnb+PR0JJse2c1AreR3lu4CogHSInfX5C3Ihe9c/
aj0xI70V7Ib0J5JBVP4WKeyxzL7yMLcjICAK3R6d2usgi4koL2p3ruolyJhxwaAGiuCZ1EeYWRV6ZB0NfaVhY
fz2XPnBoA1NeGfLDRN4F/f7H2VHRpvCZs/
ZhrRs4NALRGw9tVWHXrURWta42Gt9+LXvcUkVjnhkis857e5v4Bd0GivJEoJ+IAAAAAASUVORK5CYII=");
dashlet.addOption("leaflet.marker.icon.3.iconSize", "[24, 24]");
dashlet.addOption("leaflet.marker.icon.3.iconAnchor", "[12, 12]");
dashlet.addOption("leaflet.marker.icon.3.popupAnchor", "[-3, -76]");

dashlet.addOption("leaflet.marker.icon.active", "leaflet1_icon1"); // active Markers
highlighted in green
dashlet.addOption("leaflet.marker.icon.active2", "leaflet1_icon3"); // blue
dashlet.addOption("leaflet.marker.icon.default", "leaflet1_icon2"); // inactive
Markers highlighted in red
```

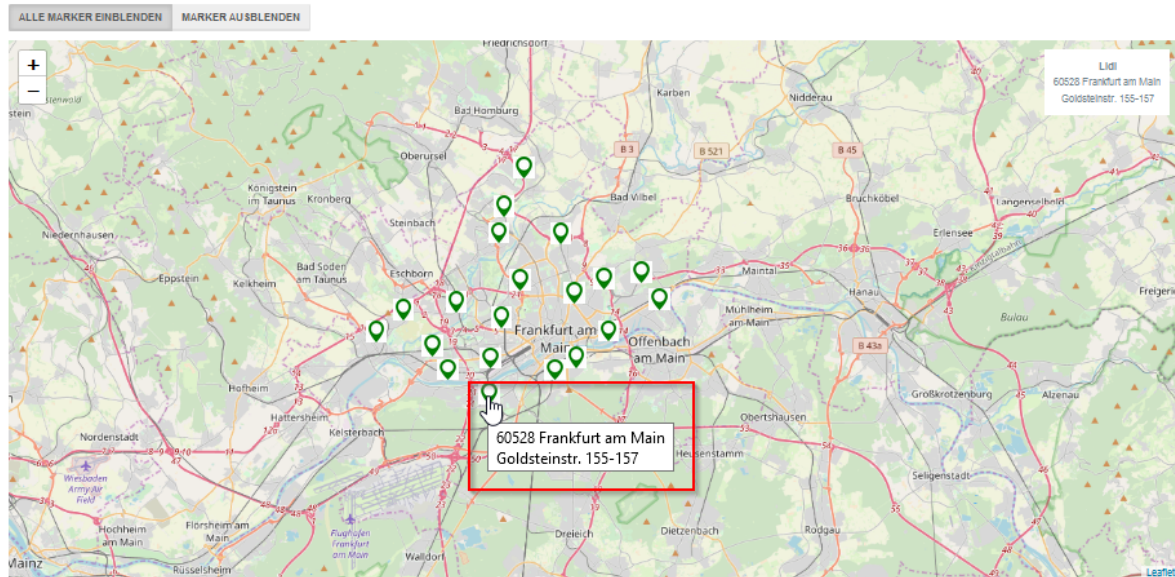
Wie kann die Marker Titles des GeoMaps gesetzt werden?

Die Marker Tile (d.h. die Anzeige wenn der Mouse über die Marker fährt) kann mit die folgende Option gesetzt werden:

'mo' ist der Marker Objekt, und die Attributes von die Marker Objekt (z.b. Postleitzahl, Stadt, Anschrift , usw) könnte selektiert werden für die Anzeige.

```
//marker options (mo ist der Marker Objekt)
dashlet.addOption("leaflet.marker.options.title", "mo['Postleitzahl'] + ' ' +
mo['Stadt'] + '\\n' + mo['Anschrift']");
```


Leaflet Beginner Feature Demo



Wie kann ein Info Box in der GeoMaps Anzeigen werden?

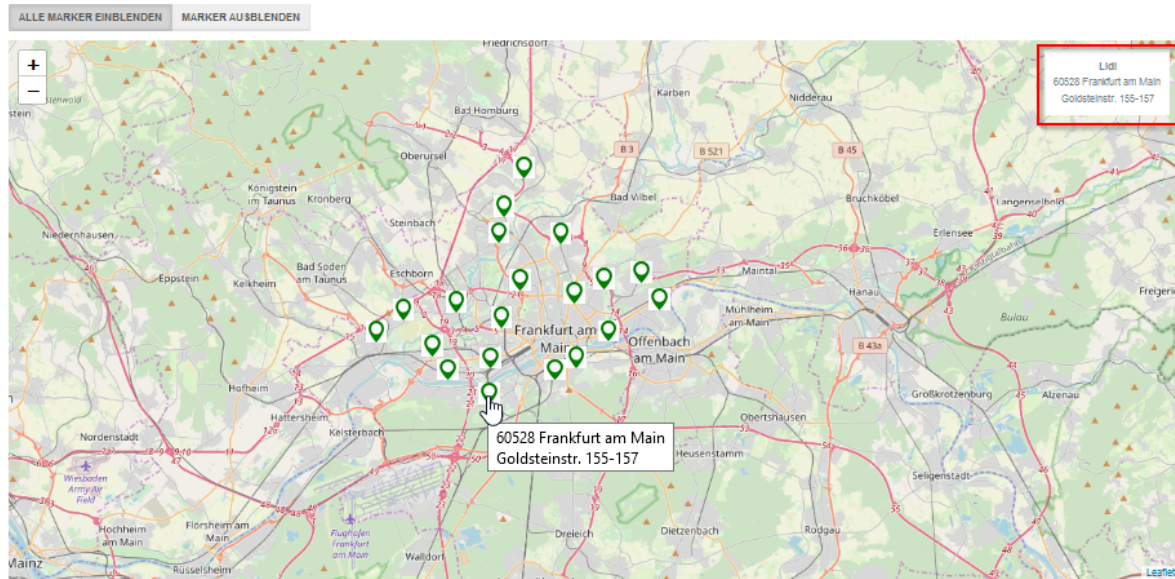
Das Infobox (d.h. die Anzeige wenn der Mouse über die Marker fährt in ein Box Format) kann mit die folgende Option gesetzt werden:

'mo' ist der Marker Objekt, und die Attributes von die Marker Objekt (z.b. Postleitzahl, Stadt, Anschrift , usw) könnte selektiert werden für die Anzeige.

```
// Optionen für Mouseover und Mouseout event
dashlet.addOption("leaflet.marker.event.mouseover",
"leaflet1_info.update(param.target.options.data)");
dashlet.addOption("leaflet.marker.event.mouseout", "leaflet1_info.update(null)");

//info box details
dashlet.addOption("leaflet.infobox.style", "background-color: white; padding: 10px;
text-align: center;" );
dashlet.addOption("leaflet.infobox.style.default", "background-color: yellow;
padding: 10px; width: 500px; text-align: center;" );
dashlet.addOption("leaflet.infobox.html.active", "'<b>' + data[\"Sparte\"] + '</b><br />' + data[\"Postleitzahl\"] + ' ' + data[\"Stadt\"] + '<br/>' +
data[\"Anschrift\"]" );
dashlet.addOption("leaflet.infobox.html.default", "''" );
```


Leaflet Beginner Feature Demo



Wie kann ein Click-Event für die Marker in GeoMap erstellt werden ?

Ein Click-Event kann mit der folgende Option gesetzt werden.

Der Logik ist mit der Latitude und Longitude Wert des Markers, die richtige DataRecord zu identifizieren und dann ein DataTable (zum Beispiel) zu filtern.

```
//Click Event
dashlet.addOption("leaflet.marker.event.click", " // console.log(param.latlng);
\n "+
                                " var selectedData =
leaflet1_markers_json.data;
\n "+
                                " for (var i = 0 ; i <
selectedData.length ; i++)
{
                                \n "+
                                "     if ( selectedData[i][\"map-
lat\"] == param.latlng.lat && selectedData[i][\"map-lng\"] == param.latlng.lng )
{
                                \n "+
                                "         selectedKunde =
selectedData[i][\"Name\"];
\n "+
                                "     }
\n "+
                                " }
\n "+
                                " if (typeof
table1_lazyLoadTable!== 'undefined')
{
                                \n "+
                                "
table1_lazyLoadTable(param.latlng.lng,7, param.latlng.lat,8, selectedKunde, 3);
\n "+
                                "     }
\n
                                );
```

Wie kann Buttons in GeoMap erstellt werden für alle Marker aus und einzublenden?

Die Buttons für alle Markers aus und einzublenden könnte mit die folgende Optionen gesetzt werden.

Hinweis :

- Die Dashlet hat die ID '*leaflet1*', und deswegen die JSON array heißt '*leaflet1_geojson*'.
- Die Funktion *leaflet1_addMarkersToMap()* ist in HTML File gerendert und macht die Aufgabe alle Markers, die in JSON array enthalten sind, einzublenden.
 - Die Parameters für diese Funktion sind folgende
 - i. *data_markers* - Die JSON array mit alle marker Informationen (z.b. *leaflet1_markers_json.data*)
 - ii. *idkey* - Der Name des Keys, die in der JSON array gesucht werden soll.
 - iii. *idarray* - Der List für der Keys.
- Die Funktion *leaflet1_removeMarkersFromMap()* ist in HTML File gerendert und macht die Aufgabe alle Markers entfernen.

leaflet1_addMarkersToMap(data_markers, idkey, idarray, idarray2)

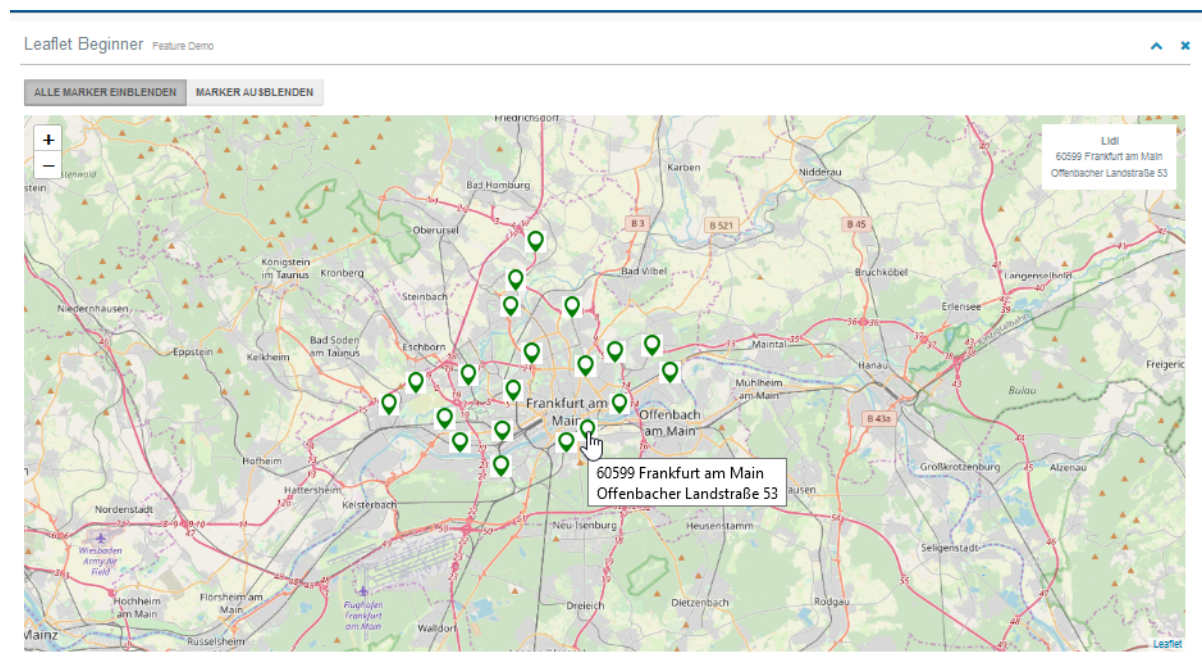
- i. *data_markers* : Die Array mit die Markers, die anzeigen soll.
- ii. *Idkey* : Die Id-Spalte
- iii. *Idarray* : Die erste Array mit ID-Werte
- iv. *Idarray2* : Die zweite Array mit ID-Werte

Funktion-Aufruf	Ergebnis
<code>leaflet1_addMarkersToMap(leaflet1_markers_json.data);</code>	Alle icons in Active farbe
<code>leaflet1_addMarkersToMap(leaflet1_markers_json.data, 'ID', ['DE3339','DE6998','DE5372']);</code>	Die Filterete Were (3 IDs) in Active Farbe und die anderen in Default Farbe
<code>leaflet1_addMarkersToMap(leaflet1_markers_json.data, 'ID', ['DE5045'], ['DE3339','DE6998','DE5372']);</code>	Die erste Array in Active Farbe, die zweite array in Default Farbe und die alle anderen in Active2 Farbe

```
// Create buttons
buttonbar = dashlet.getDashlet().createDashletBar("DashletButtonBar", "TOP");

// Buttons alle Marker einzublenden
btn = buttonbar.addButton("Alle Marker einblenden", false);
btn.addOption("button.onclick",
"leaflet1_addMarkersToMap(leaflet1_markers_json.data); console.log('Marker ein');");

// Buttons alle Marker auszublenden
btn = buttonbar.addButton("Marker auszublenden", true);
btn.addOption("button.onclick", "leaflet1_removeMarkersFromMap(); console.log('Marker aus');");
```



Wie kann Bild von ein (oder mehr) Dashlets zu ein Bericht weiterleitet werden?

Das Bild von ein (oder mehr) Dashlets kann als folgendes in ein BIRT-Bericht weitergelitet werden.

Schritt 1 : In ein HTML-Dashlet kann ein Button erstellt werden.

- Der onclick-event für das Button würde ein Funktion "exportDashlets" anrufen. Diese Funktion ist automatisch in das Dashboard gerendert.
- Die Optionen für die Liste von Chart-Ids, die exportiert werden soll kann mit der Option "exportChart.dashletId" gegeben werden.
- Die servBIRD-Bericht Aufruf kann mit der Option "exportChart.javascript" definiert werden.

```
// HTML Dashlet
var jsDashlet1 = DashletManager.addDashlet("DashletHTML", "", "", false, null);


// HTML Script
var html = "<div class=\"btn btn-primary\" onclick=\"\" +
\"exportDashlets();\"+
\">Bericht</div>\";
jsDashlet1.addOption("html", html);

// Export options
jsDashlet1.addOption("exportChart.dashletId", "chart1;table1;leaflet1"); // Semi-
colon seperated list
jsDashlet1.addOption("exportChart.javascript",
"openJobCreatorDialog( [
"                                {name:'--__report',      value: '/test/
Export_Report.rptdesign'}      \n" +
"                                ,{name:'--__outputFormat', value: 'PDF,HTML'}
\n" +
"                                ,{name:'--paramImageMap',  value:
leaflet1_img.split(',')[1] }  \n" +
"                                ,{name:'--paramImageChart', value:
chart1_img.split(',')[1] }    \n" +
"                                ,{name:'--paramImageTable', value:
table1_img.split(',')[1] }    \n" +
"                                ]); ");
```

Schritt 2 : Der BIRT-Bericht sollte definiert werden mit diese Parameters.

- Image-Element sollte definiert werden in der Report, und der Expression ist als folgendes:

```
importPackage(Packages.org.apache.commons.codec.binary);
this.data=Base64.decodeBase64(params["paramImageMap"].value);
```

 Edit Image Item ✕

Select Image from:

- ☐ URI
- ☐ Image file in shared resources
- ☐ Embedded image
- ☒ Dynamic image

Retrieve image from data set (databases)

Enter dynamic image expression:

Select Image Data...



Beim Leaflet-Dashlet, bitte bei der folgende Option aufpassen.

```
// Map-Starteinstellungen setzen
dashlet.addOption("leaflet.map.tileLayer", "https://
{s}.tile.openstreetmap.org/{z}/{x}/{y}.png");
```

Wie kann Buttons in GeoMap erstellt werden für nur manche Marker aus und einzublenden?

Die Buttons für nur manche Markers aus und einzublenden könnte mit die folgende Optionen gesetzt werden.

Hinweis :

- Die Dashlet hat die ID '*leaflet1*' , und deswegen die JSON array heißt '*leaflet1_geojson*'.
- Die Funktion *leaflet1_addMarkersToMap()* ist in HTML File gerendert und macht die Aufgabe alle Markers, die in JSON array enthalten sind, einzublenden.
 - Die Parameters für diese Funktion sind folgende
 - i. *data_markers* - Die JSON array mit alle marker Informationen (z.b. *leaflet1_markers_json.data*)
 - ii. *idkey* - Der Name des Keys, die in der JSON array gesucht werden soll. (z.b. ID Feld)
 - iii. *idarray* - Der List für der Keys.

```
// Create buttons
buttonbar = dashlet.getDashlet().createDashletBar("DashletButtonBar", "TOP");

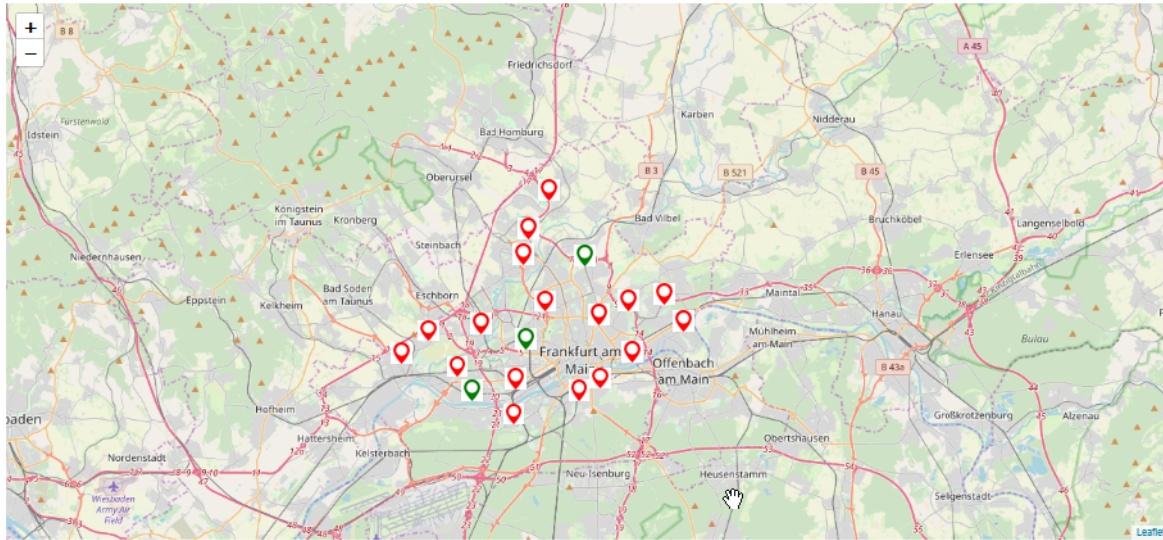
// Button nur 3 Marker einzublenden
btn = buttonbar.addButton("3 Marker einblenden", false);
btn.addOption("button.onclick", "leaflet1_addMarkersToMap(leaflet1_markers_json.data,
'ID', ['DE3339','DE6998','DE5372']); console.log('Marker ein');");

// Buttons nur 1 von 3 Marker einzublenden
btn = buttonbar.addButton("1 von 3 Markern einblenden", false);
btn.addOption("button.onclick",
"leaflet1_addMarkersToMap(leaflet1_searchMarkersWithId(leaflet1_markers_json.data,
'ID', ['DE3339','DE6998','DE5372']), 'ID', ['DE6998']); console.log('Marker ein');");
```

Die erste Bild zeigt die Ergebnis wenn Mann auf der Button "3 Marker einblenden" gedrückt haben und die zweite Bild zeigt die Ergebnis für den Button "1 von 3 Markern ausblenden".

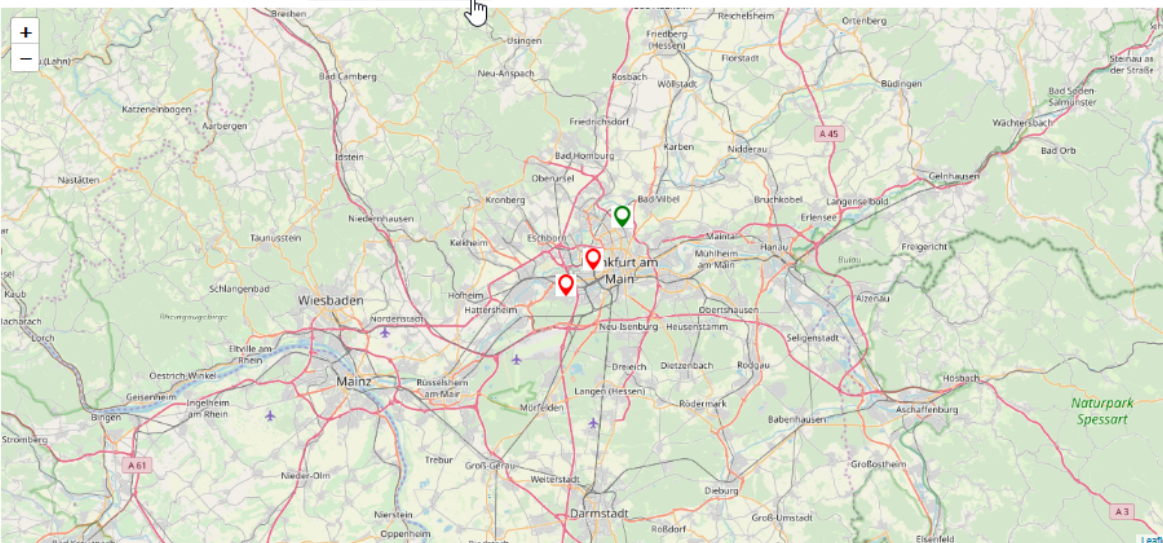
Leaflet Beginner Feature Demo

ALLE MARKER EINBLENDEN 3 MARKER EINBLENDEN 1 VON 3 MARKERN EINBLENDEN MARKER AUSBLENDEN



Leaflet Beginner Feature Demo

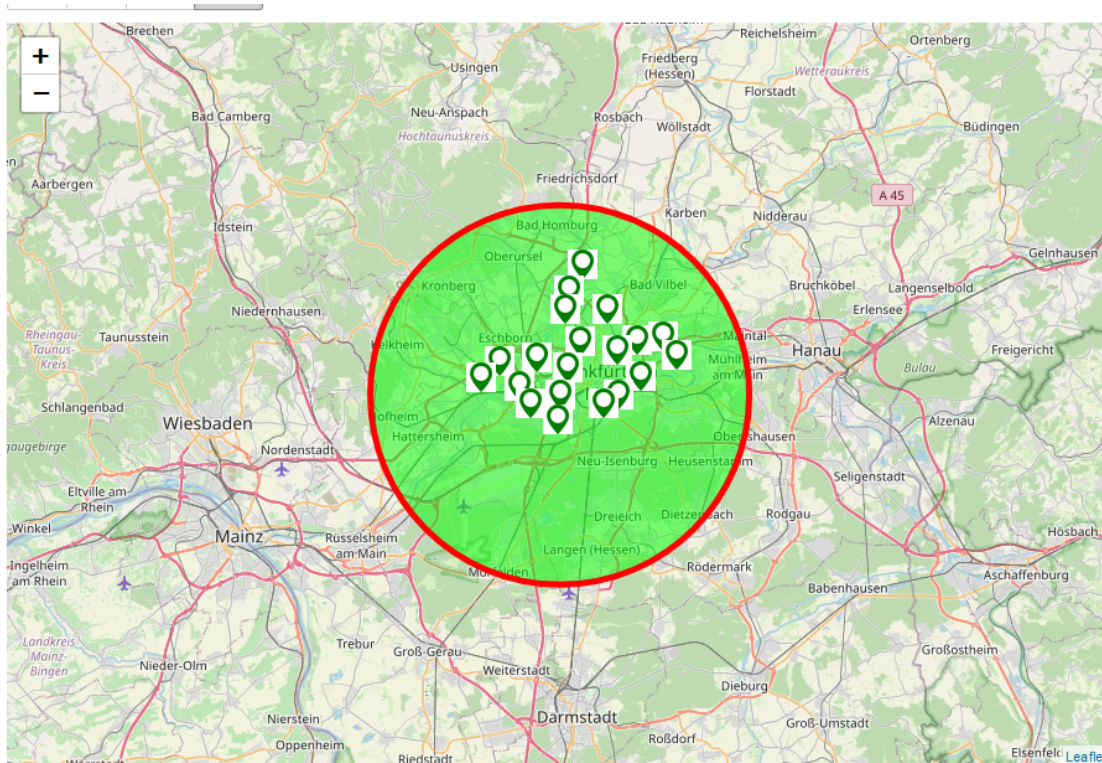
ALLE MARKER EINBLENDEN 3 MARKER EINBLENDEN 1 VON 3 MARKERN EINBLENDEN MARKER AUSBLENDEN



Wie kann ein Circle-Layer in GeoMap erstellt werden?

Eine Circle-Layer kann mit die folgende Optionen erstellt werden.

```
// circle options
dashlet.addOption("leaflet.layer.circle.show", "true");
dashlet.addOption("leaflet.layer.circle.radius", "15000"); // in meters
dashlet.addOption("leaflet.layer.circle.color", "red");
dashlet.addOption("leaflet.layer.circle.fillcolor", "#0f0");
dashlet.addOption("leaflet.layer.circle.opacity", "0.5");
dashlet.addOption("leaflet.layer.circle.weight", "5");
```



Wie kann Buttons für Umkreis-Suche mit Circle-Layer in GeoMap erstellt werden?

Die Buttons für Umkreis-Suche kann mit die folgende Optionen erstellt werden:

Option 1: Buttons automatic erstellen

```
//Create Buttons automatically
dashlet.addOption("magic.add.buttonbar", "true");
dashlet.addOption("leaflet.distancebutton.values", "2;8;10;15"); // in km
(semi-colon separated list)

// circle options
dashlet.addOption("leaflet.layer.circle.show", "true");
dashlet.addOption("leaflet.layer.circle.radius", "15000"); // in meters
dashlet.addOption("leaflet.layer.circle.color", "red");
dashlet.addOption("leaflet.layer.circle.fillcolor", "#0f0");
dashlet.addOption("leaflet.layer.circle.opacity", "0.5");
dashlet.addOption("leaflet.layer.circle.weight", "5");
```

Option 2: Buttons Manual erstellen

Diese Variant ist hilfreich wenn custom-events sollte an Button-Click definiert werden (zb. Andere Charts sollte geändert werden nach der Umkreis-Such Funktion)


```
// Create Buttons manually
DashletButtonBar buttonbar = (DashletButtonBar)
dashlet.createDashletBar(DashletBarType.BUTTON_BAR, DashletButtonBar.TOP_BAR);
BootstrapButton btn = buttonbar.addButton("2 km", false);
btn.addOption("button.onclick", "leaflet1_computeMarkerDistance(2);
updateBarChart(); ");
btn = buttonbar.addButton("5 km", false);
btn.addOption("button.onclick", "leaflet1_computeMarkerDistance(5);
updateBarChart(); ");
btn = buttonbar.addButton("10 km", false);
btn.addOption("button.onclick", "leaflet1_computeMarkerDistance(10);
updateBarChart(); ");
btn = buttonbar.addButton("20 km", true);
btn.addOption("button.onclick", "leaflet1_computeMarkerDistance(20);
updateBarChart(); ");

// circle options
dashlet.addOption("leaflet.layer.circle.show", "true");
dashlet.addOption("leaflet.layer.circle.radius", "15000"); // in meters
dashlet.addOption("leaflet.layer.circle.color", "red");
dashlet.addOption("leaflet.layer.circle.fillcolor", "#0f0");
dashlet.addOption("leaflet.layer.circle.opacity", "0.5");
dashlet.addOption("leaflet.layer.circle.weight", "5");
```

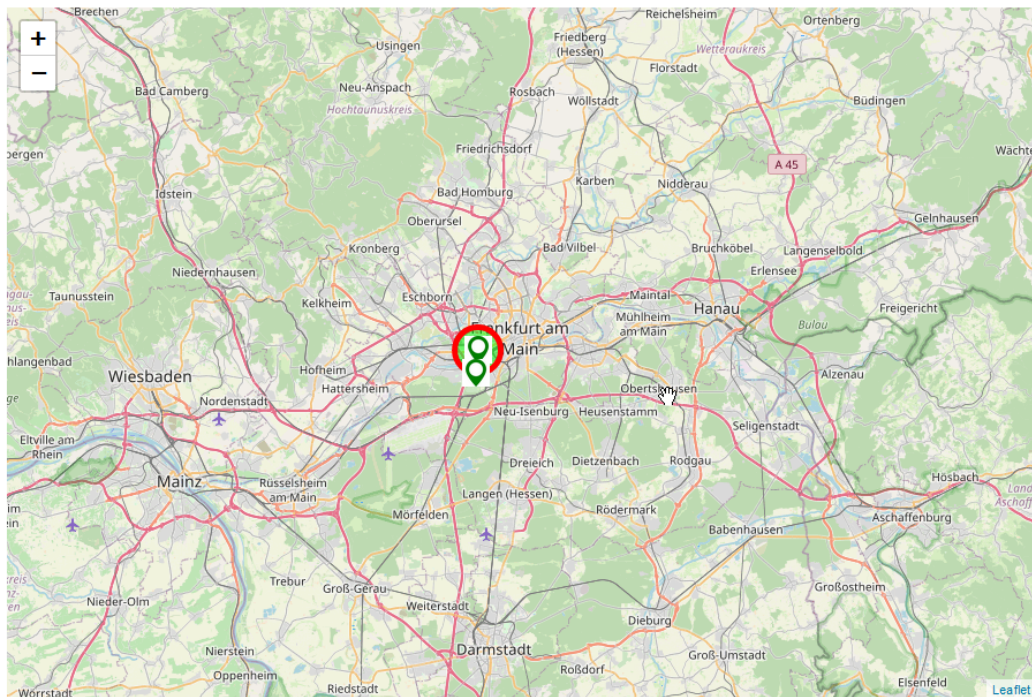
Umkreis-Suche - 2 Kilometer Radius

Umkreis-Suche - 10 Kilometer

Leaflet Beginner Feature Demo



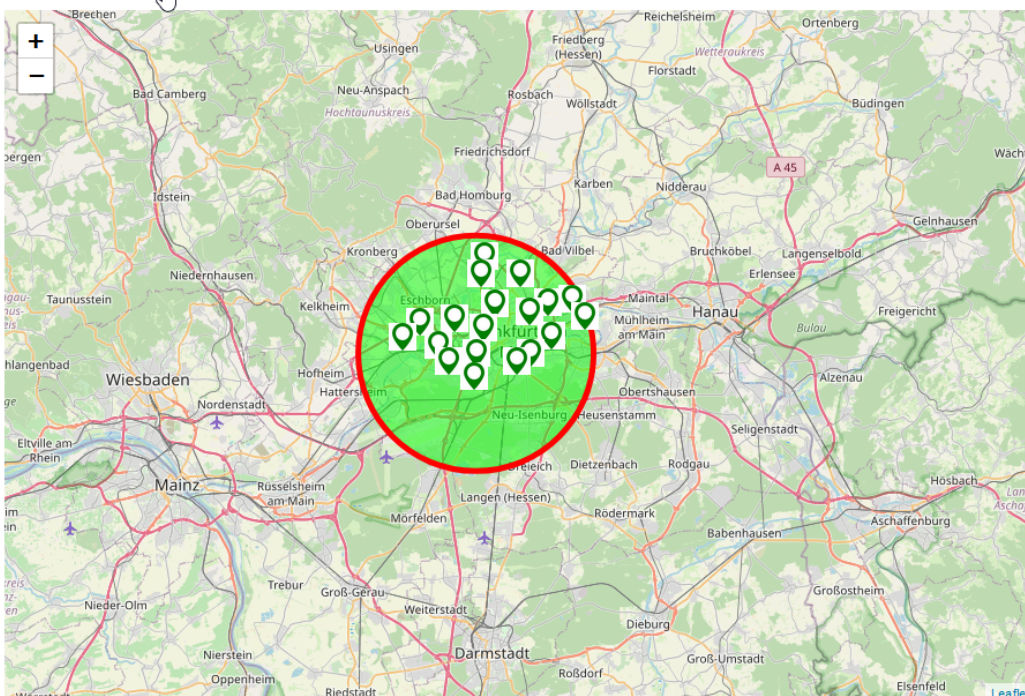
2 km 8 km 10 km 15 km



Leaflet Beginner Feature Demo



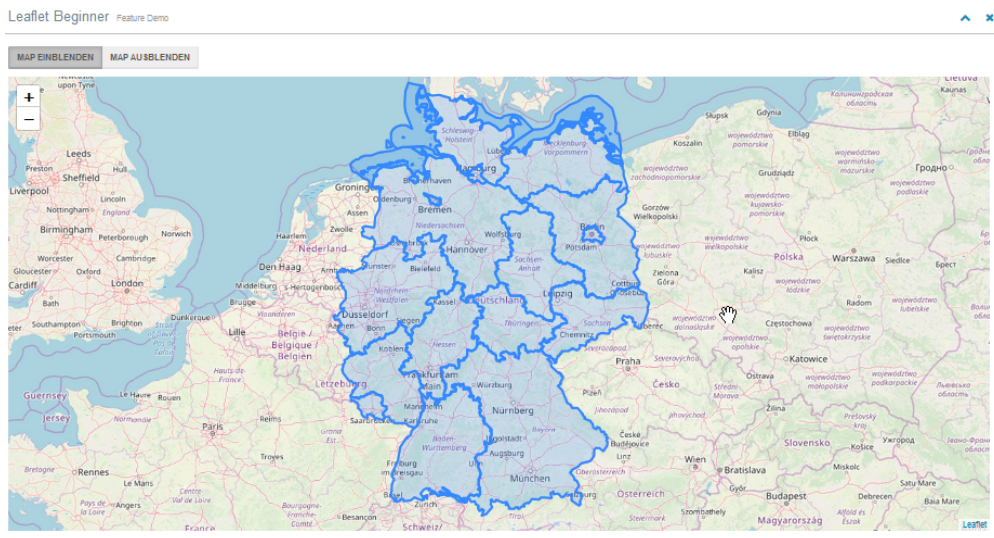
2 km 8 km 10 km 15 km

**Wie kann ein Map (Hintergrund) in GeoMap erstellt werden?**

Ein Map kann als Hintergrund der GeoMap mit die folgende Optionen gesetzt werden.

Hinweis: Die Pfad zu der Hintergrund Map (in JSON Format) sollte gesetzt werden.

```
dashlet.addOption("leaflet.map.filename", "/dashlets-showcase/getjson/maps/
bundeslaender.json"); //die Pfad desw JSON files
dashlet.addOption("leaflet.map.autoload", "true"); // Laden die Map bei starten des
Dashbaords
```



Wie kann ein Map (Hintergrund) in GeoMap mit Buttons ein und ausblenden werden?

Einblenden und Ausblenden ein Map(Hintergrund der GeoMap) kann mit die folgende Optionen gesetzt werden.

Hinweis :

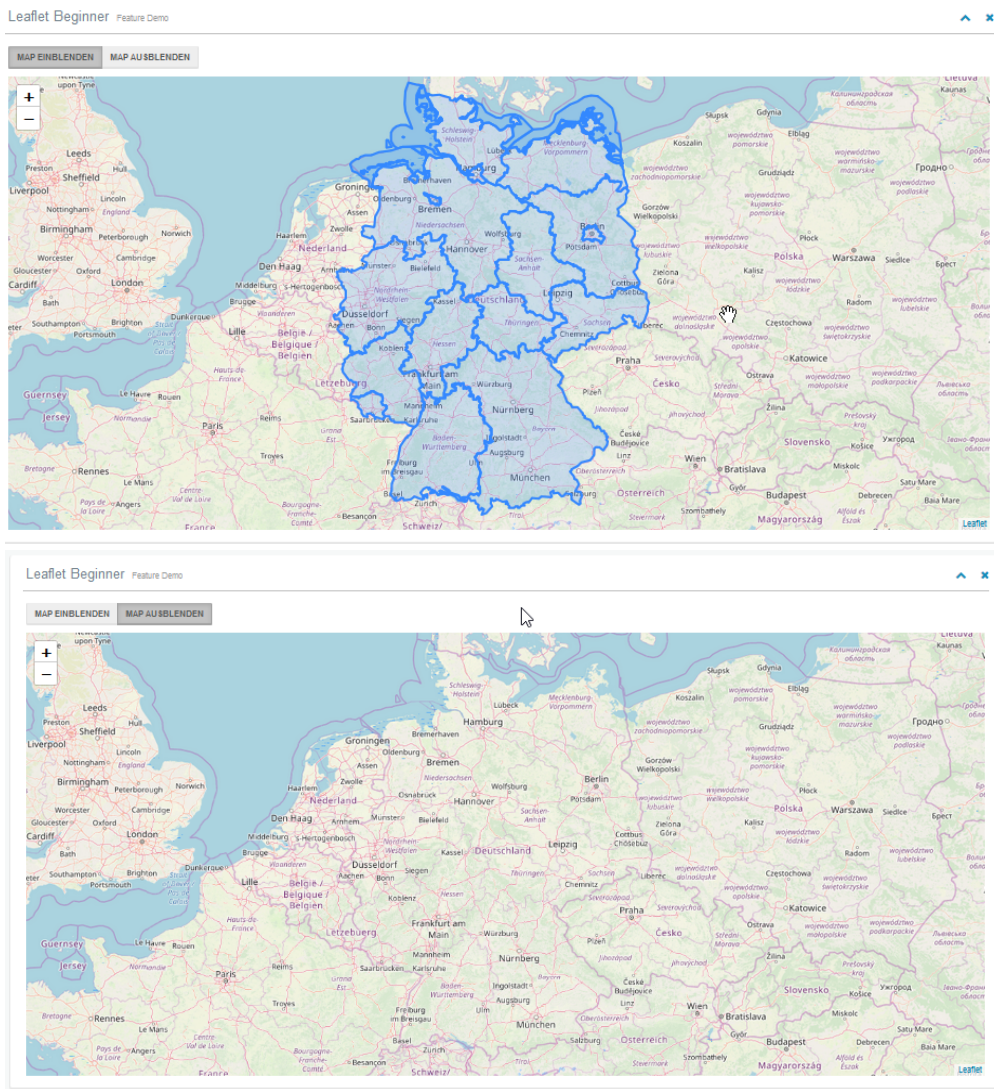
- Die Dashlet hat die ID '*leaflet1*', und deswegen die JSON array heißt '*leaflet1_geojson*'.

```
// Create buttons
buttonbar = dashlet.getDashlet().createDashletBar("DashletButtonBar", "TOP");

// Map einzublenden
btn = buttonbar.addButton("Map einblenden", false);
btn.addOption("button.onclick", "leaflet1_geojson.addTo(map_leaflet1);
console.log('Map ein');");

// Map auszublenden
btn = buttonbar.addButton("Map ausblenden", true);
btn.addOption("button.onclick", "leaflet1_geojson.remove(); console.log('Map aus');");
;
```

Die erste Bild zeigt die Ergebnis wenn Mann auf der Button "*Map einblenden*" gedrückt haben und die zweite Bild zeigt die Ergebnis für den Button "*Map ausblenden*".



Leaflet Markers

Einführung

Es gibt die Möglichkeit die Darstellung von Leaflet Markers einflüssen.

In der Memory-Buffer kann ein Spalte definiert werden mit die Icon-Farbe für jede Punkt.

Dashlets erzeugen

Die Farbe-Spalte-Namen sollte angepasst werden.

```
mbr.getColumnDefinition("Map_Icon").setDisplayNames("map-icon"); //Map-Icons Color
```

Für die MarkerManager Option zu aktivieren , muss die folgende Option definiert werden.

```
dashlet.addOption("leaflet.feature.markermanager","true");
```

Abhängig wie die Farbe-Werte in die Spalte definiert ist, muss der Icon-Objekt angepasst werden.

Beispiel:

Falls der Werte in die Spalte steht als "icon-red", "icon-blue", usw., dann sollten die ICON-Objekt als folgendes definiert werden:

Variant 1 : Als Optionen definieren

```
dashlet.addOption("leaflet.marker.icon.1.iconId","icon-red");
dashlet.addOption("leaflet.marker.icon.1.iconUrl", "data:image/
png;base64,iVBORw0KGgoAAAANSUHEUgAAACAAAAAwCAIAAAD/zU84AAAAABnRSTlMA/wD/
AP83WBt9AAAAACXBIWMAAA7EAAA0xAGVKw4bAAADi0LEQVRYw2P8//8/Ay0BC0EVvy9c/r5lx5+bd/
4+efr3wSMGBgZmBTlmGkWZUUOD2c2C1P82hlx+eDfm7dfZsz7tnTlnxu38DlQRYkrIpg704VZQpWEC76v2fg
+s+jfm7dEhg0TAD9/fzt3QhRhC/
5/+fouMev7mo1kBDDeHj4fw8rmMPNy4Lfjz57VnyM89B8i0UnYbC5Hta5HtYEI2/WlkMiWmMzAw/
Dxy4m1oPMOfP1gs+NTSQ17IoIEf0/Z8KK1FD6K/L16+UNT//+MHLRI/i8T1UyWqSoh88KmyEb/
pnCH+7PbWkFT/68TpX0dPfluxFqfqP38+VjYKr14I9cHfJ8+eK+ohBxxaShecPYndwQZN/NeJ0+8Ss/
HkEonb51hUlJgYGBh+HjmOy3QmAX7R/VswTWdgYGCzMBU7vJ1ZRgpnh085AI3knweP4lik0HsSHi0YRISF5k/
DachBo1ALfh05jsv5nCH+BBK+iwMkMjHBrzPnoRb8//IVqwo2G0tikgyrgS6uqEbNaNh8QIwFeMKQgAW/
r1wjxoLFz84TsIBZQQ67ZguXCRao/798/XXhMlyPrGf+qAXszva49L9PzcNvwYfCSlxRy0HhArUAwsIKvm/
Y+rmjH5fslmzvs5ZhEuWw90ZURY9l1T/++IlLqVsFqb83c3sFiYMLCyQ5PHrzPmPpbU/
j5zAk0Akn99k50CAWvBlwvQPhZX4Q40Rg4NVR50BgeH3lesEi0W++gq+hgqED/
7/+PFCUR+PJ0gCTAL8EvcvMcEjGeI6nuIcBioB7owkeB5CVJn/
Pnx8oaj378NHCK1n50CQuH8R3shgQvYXT34mNZyfiNyEqan0/334+FxWC1e6JgqwsEg+vopsARNa5HCnxFhk/
JgtwBYYeruIosoZqSrGwdgxS4hzZySS53yuiGDMugFL0/
HPnXsvNM1wVaJ4gPj5w5h1A5bimkVFiTsmjFTTQ08sdY82Bu/ZHgCq/
MZcFU4LCPknD7uxJv07mCDq+JkwlNaEW8BHsU4LWA100UM8CbS+VgbTgQsIN4T+JXhs4AYT+B3PgELiPEEQQU
ELMDvCYLQJ2wBfjcSE0mELcDLCKcT5QFuFxKZBojygmJMTxDpFAYGBob/xIFf5y89ZuCHox/7DX0pkVgL/
v///yYgCmL6Kwdv4nWRYMHv23efyWg+FVH6df4S8boYaT2cAwCXC2Tqk9n/OwAAAABJRUErkJggg==");
dashlet.addOption("leaflet.marker.icon.1.iconSize","[20, 30]");
dashlet.addOption("leaflet.marker.icon.1.iconAnchor","[10, 30]");
dashlet.addOption("leaflet.marker.icon.1.popupAnchor","[-3, -76]");
dashlet.addOption("leaflet.marker.icon.1.shadowUrl","data:image/
png;base64,iVBORw0KGgoAAAANSUHEUgAAACAAAApCAQAAAAACach9AAACMULEQVR4Ae3ShY7jQBAE0Aoz/
f9/HTMzhg1zrdKurJbdx+Kd2nD8VNudfsL/Th//dyQN2TH6f3y/
BGpC379rV+S+qqetB0xImNQXL8JCAr2V4iMQXHGNJxeCfZXhSRBcQMfvk0WUdtfzLLgAENmZDcmo2TVmt80SM
2eXxP3DjHSMFutqS7SbmemzBiR+xpKCNUIRkdkkYxhAkyGoBvyQFEJEefwSmmvBfJuJ6aKqKwNAkvGZ0aZXT
UgFqYULWNSHUCKZuR1HIIimUExutRwxz0LR0IG4vKmCKQt364mIhSyZaf1m9lHZHJZrLAOMMztRRiKimp/
rpdJDC9Awry5xTZCte7FHTuS8wJgeYGreX28xNTd086Dik7vUMscQ0a8y4DoGtCCSkAKlNwpgNtpHjrC6MIHU
kR6YWxxs6Sc5xqn222mmCRFzIt8lEdKx+ikCtg91qS2WpwVfBelJCiQJwvzixfi9cxZQWgiSJeLKnwBELKYtD
Ob2MFbhmUigbReQBv0Cg4+qMXSxXSyGun4UbF8l+7qdSGnTC0XLCmahIgUHLhLOhpVCTw4CzYXvLQWQBjNmxo
Cs0KAXSgBJno75avolkRw8iIAFcscd02e9iyCd8tHwmeSSoKTowIgvscSGZU0A7PuCN5b2BX9mQM7S0wYhMNU
74zgsPBj3HU7wguAfnxxjFQGBE6pwN+GjME9zHY7zGp8wVxMShYX9NXvEWD3HbwJf4gi04CFIQXXSch1/
TM+04kkBiAAAAAELFTkSuQmCC");
dashlet.addOption("leaflet.marker.icon.1.shadowSize","[30, 40]");
dashlet.addOption("leaflet.marker.icon.1.shadowAnchor","[8, 38]");

dashlet.addOption("leaflet.marker.icon.2.iconId","icon-green");
```

```

dashlet.addOption("leaflet.marker.icon.2.iconUrl", "data:image/
png;base64,iVBORw0KGgoAAAANSUHEUgAAACAAAAAwCAIAAAD/zu84AAAAABnRSTLMA/wD/
AP83WBt9AAAAACXBIWXMAAA7EAAAOxAGVKw4bAAADeLEQVRYw2P8//8/Ay0BC0EVe15f3Pj81IPvr558f/
vg2ysGBgYZTmEFLjEFTjF/STMHER0WRmY82hlx+eDJ97e9dzeueXbsyfe3ePSLsPHFyNrnK/
kocImRYMGKp0cyL8748PsrkeHAW8LRrZ2QoeB02IIvf34knp+85tkxMoLbR8JkuXExDwsHTgv+/P/
reb5z+uLZEepjBmdos6ZDuYkE2PPNNHiekMDAxH3l4PPd315/9fLBa03FxNXsiggr2vzpdeXYgeRC9+fLdc
lfbj32/qpH1G5uv0k1W4JRH5oPLaYvymh0hZ2YtoWwiqMTAwnHh/6+jb6yueHsGL+M//
v5XXlqw2LYX64Mn3t4q705EDDhmocEvONshyENFBEz/x/
lbIUUk3vjzFZc1t12kq3JJMDAwMR95dx2W6ACv3futmTNMZGBgsBNU027bJcArjLgIuQSP54JsruBTNNsjCY4
QIG998w1xcshBjoT7A5fwQKSv88ekiqq+JTEw5sNdqAVf/vzAlWuISTMG/
Iq4oholH2D1ATEW4AlDAhZc+fSIGAvoFLhDwAJcJe2Fj/ff/PqE3/
Qvf35c+HgfTwAwMTAw0Ivq4dKfemEafgsKr8zDFYUeYoZQCyAsrGDD85Mdt9fhkpl9f9uch7txybyqLGS�KIsk
diS9+fsCl1EJQrVs7wUJIDVI7/vn/
98yHu6VXFxx5ex1PAnnuMZ+DiRVqWYS7mwuvzMMfGhXMrDp88gwMDFc+PSRYLNarhzdoRCB880Pfb8VdaXg8Q
RIQYOW+7zoTEckQ1xWr+DNQCWQouMPzECIfpMi7EpmzCIZkvrIvlowmwMqdr+RDBecrekiwC2DPyQXKvmhtAl
IBCyNzuWoQzqJCgJU7Rd6VEgtiZ02RnY+LLCpXDeJgYiXb+dVqIQK0wl2gQxFD/
IsiJCxwawbsJSm2Yqe+NuzuECxsj/
20hQNqHBLxjsak2p6gKQ51poHe31QrRZCqifq1c0xim03QIVb0kfChHjTHUR0cFwCTKS6iFTFOC0w4FcMkDQn
0vlYG04ELCDeE/iV4b0AGE/
gdz4BC4jxBEEFBCzA7wmCzidsAX43EhNjHc3A5QlinE+UBbhcSmQaI8oCTE8Q6XxiLcB0L/
H5nFgLkD1BvPMZiBkMgYNu7XhIe3u6fgbxuhhpPZWDAIDoJt56NyOAAAAAEFTkSuQmCC");
dashlet.addOption("leaflet.marker.icon.2.iconSize", "[20, 30]");
dashlet.addOption("leaflet.marker.icon.2.iconAnchor", "[10, 30]");
dashlet.addOption("leaflet.marker.icon.2.popupAnchor", "[-3, -76]");
dashlet.addOption("leaflet.marker.icon.2.shadowUrl", "data:image/
png;base64,iVBORw0KGgoAAAANSUHEUgAAACAAAAApCAQAAAAACach9AAACMUlEQVR4Ae3ShY7jQBAE0Aoz/
f9/HTMzhg1zrdKurJbdx+Kd2nD8VNudfsL/Th//dyQN2TH6f3y/
BGpC379rV+S+qqetB0XImNQXL8JCAR2V4iMQXHGNJxeCfZXhSRBcQMfvk0WUdtfzLLgAENmZDcmo2TVmt80SM
2eXxBp3DjHSMFutqS7SbmemzBiR+xpKCNUiRkdkkYxhAkyGoBvyQFEJEefwSmmvBfJuJ6aKqKwnAkVgz0aZXT
UgFqYULWNSHUckZuR1HIiImUExutRxxwz0LR0IG4vKmCKQt364mIlhSyzAf1m9lHZHJZrLA0MMztRRiKimp/
rpdJDC9Awry5xTZCte7FHtuS8wJgeYGrex28xNTd086Dik7vUMscQ0a8y4DoGtCCSkAKlNwpgNtpbjrC6MIHU
kR6YWxxs6Sc5xqn222mmCRFzIt8lEdKx+ikCtg91qS2WpwVfBelJCiQJwvzixfi9cxZQWgiSjElKnwBELKYtD
Ob2MFbhmUigbReQBv0Cg4+qMXSxXSyGUN4UbF8l+7qdSGnTC0XLCmahIgUHLhLOhpVCTw4CzYXvLQWQbJNmXo
Cs0KAxSgBJno75avolkRw8iIAFcscd02e9iyCd8tHwmeSSoKTowIgvscSGZUOA7PuCN5b2BX9mQM7S0wYhMNU
74zgsPBj3HU7wguAfnxxjFQGBE6pwN+GjME9zHY7zGp8wVxMShYX9NXvEWD3HbwJf4gi04CFIQxXSCH1/
TM+04kkBiAAAAAEFTkSuQmCC");
dashlet.addOption("leaflet.marker.icon.2.shadowSize", "[30, 40]");
dashlet.addOption("leaflet.marker.icon.2.shadowAnchor", "[8, 38]");

dashlet.addOption("leaflet.marker.icon.3.iconId", "icon-blue");

```

```
dashlet.addOption("leaflet.marker.icon.3.iconUrl", "data:image/
png;base64,iVBORw0KGgoAAAANSUHEUgAAAB4AAAAwCAIAAACJ9F2zAAAAABnRSTLMA/wD/
AP83WBt9AAAAACXBIWXMAAA7EAAAOxAGVKw4bAAAF00LEQVRYw7WYw2wUVRjH/
+cys5d2aZfSdtnSlu1uV6ktbCil1JAiUJa2G8RQgomCmbQFgTbRmKAmRA2Jl0QU4iWaGF9MTAi8ICg+iK2GRE
0pIaEFW1trUKFQagtdddru7M3N8aLHtzuyLRc/bnHPmd77zXf7fZiGQAv/P4MmXw1H14sBoZ/
9I77WxkWAMgD1T8jptlW77ssVZVh0fC/rqU0jTbweOnfujbzCkkYltE/
cjAKhQ3PmWx1cXPbPBVZRrNSQqvU0iivbB17++caJn0AQIFZp6DzpzUAbC51vFK01LWgIemdMU6Buj480fnP/
q4t/QYhBao8SCio1LLN/
1lLlyDYnRF8fCW9+88eOgbtQxmcZMnNlsfXkyw8X5FimbjU9YjuOdHT8Fpw1F4Ay3vn73R1H00IRxQB96PiVs
92jUCNzzDU10nZ59PVjl+PRnf0jr0/3Q4sZv0YzmAlMBpPBTkDMeJsWe//MQEffyIzk03K6PxQjBnEjBJQ/
4DBvWuF8qCgLEN1X75w6f6lncByagrjsElo4Rt471fff81WTYey7Hlz+4tmxsBKPJlTi70BW7wuPeq0mRggBI
IQIRbR3v+w5dKi3pqj6V2wWfuHweo8jkwJo6xoa18DQ5MM7yw9uK8sw8wkuAEJIhpkd3FZ2eGc57k10N3wsngu
8uDU36+tyVYSP/
Sg2+vJaAJ1HYWgKeBl8umKRf+qF7CAAVQvzy522j6hDNDYuTJ0VznQt6dRNa77UxIUdHY9rI3Vh8KRNiLYTPL
Z0c7XNlZ8jQuUWMBKPhqEpVTcQUTScSxCSx5MIGwGpIMicTgjWNjKiikapGZU4tMos/
WYjguDJ0J0X5DN20BSnqV8ILDkX0aUypw67Jf5kiJjg7V23kqPbu4di4DdpdA672SwzCqC8KAUE6qKhfvRN/
/
BYNBF3eCz64Zl+CFWvhRVFWZPJt7ZigUGGaGrP9fCeJztDEdWo+2jPfXKhd3AcmqRpkHVL8ybRtWW5hTkyiE4
Z10iJn29ue037tq6bUUX9t1G0dd2se7X9+E83o0ruRFhhjlxbtmBKr1/6/
NLbJweMZY9KnmGTb3I7MgH0Dwb7bkQUFcZaxkwHNrve2lExhe79687KA+23Q4k6CwGlk9cSkjTnuKUR0s/
Cz7+zvnRh5pSoegvmbVLVAColiJmApkKNQoombJUAqNRUuzDBndEKdvsXS0TRZWH6g0hE2bPRbdBlqr05fp/
DUG7SGkzy+/JXltoN0AD21rsMkinNoa1760tmuGf6g9/nWOW1J/
Z44kGl6tJsv8+REM0Z2e0vmZ0fsWdjCWckIRpAU02B12EB5bMxmZfmmZtqFsVPxz3bLLy5zmVQmclmZrs2uGw
WngINYPuawnwbNRAsYy7Nt9Hta4oMLq0fcs63PLmmON1gUumJ2iLnFEtaAAC76oozJC214YRmyuJZv8v4UMPZ
JYvmbVm1KLXhVHpspfPBAatss0AD2Nbhkqh8aUyzWabq/
saEyZoQXe3NCVQ6khnOpEDlwpvzqzRAFoDHioSCRahQmLpTFzfzCPl0cmFCwm1S3NX1uRn0c0gNaAG5recA
JNaQ24U8Q4+bLfl1db1htf95SvXrKgfRnjvtCM0pbGEl1bEa0BD6PkvTAANlUtrHJnT6UK5ZWurE0rHKnrNOU
Ok8T2N0xzKyH7G91mmf0HaABNNc7ywkxQDsrLnBlba5xpqU56mzLMfF99CZgZzLy3viTTkpZypSv5T60tnor
KECeXlec5isk/Z8WExtJ2h8T/wD6c9lf8sGWSAAAAABJRUErkJggg==");
dashlet.addOption("leaflet.marker.icon.3.iconSize","[20, 30]");
dashlet.addOption("leaflet.marker.icon.3.iconAnchor","[10, 30]");
dashlet.addOption("leaflet.marker.icon.3.popupAnchor","[-3, -76]");
dashlet.addOption("leaflet.marker.icon.3.shadowUrl","data:image/
png;base64,iVBORw0KGgoAAAANSUHEUgAAACKAAAApCAQAAAAACach9AAACMUlEQVR4Ae3ShY7jQBAE0Aoz/
f9/HTMzhg1zrdKurJbdx+Kd2nD8VNudfsL/Th///dyQN2TH6f3y/
BGpC379rV+S+qqetBOxImNQXL8JCAR2V4iMQXHGNJxeCfZXhSRBcQMfvkOWUdtfzLLgAENmZDcmo2TVmt80SM
2eXxBp3DjHSMFutqS7SbmemzBiR+xpKCNUIRkdkkYxhAkyGoBvyQFEJEefwSmmvBfJuJ6aKqKWnAkVgz0aZXT
UgFqYULWNSHUckZuR1HIiimUExutRxwz0LR0IG4vKmCKQt364mIlhSyzAf1m9lHZHJZrLAOMMztRRiKimp/
rpdJDc9Awry5xTZCte7FhtuS8wJgeYGrex28xNTd086Dik7vUMscQ0a8y4DoGtCCSkAKlNwpgNtpjhjrc6MIHU
kR6YWxxs6Sc5xqn222mmCRFzIt8lEdKx+ikCtg91qS2WpwVfBelJCiQJwvzixfI9cxZQWgiSJeLKnwBELKYtD
Ob2MFbhmUigbReQBV0Cg4+qMXSxXSyGUN4UbF8l+7qdSGnTC0XLCmahIgUHLhLohpVctw4CzYXvLQWQbJNmXo
Cs0KAxSgBJno75avolkRw8iIAFcscd02e9iyCd8tHwmeSSoKTowIgvscSGZU0A7PuCN5b2BX9mQM7S0wYhMNU
74zgsPBj3HU7wguAfnxxjFQGBE6pwn+GjME9zHY7zGp8wVxMShYX9NXvEWD3HbwJf4gi04CFIQxXScH1/
TM+04kkBiAAAAAEFTkSuQmCC");
dashlet.addOption("leaflet.marker.icon.3.shadowSize","[30, 40]");
dashlet.addOption("leaflet.marker.icon.3.shadowAnchor","[8, 38]");
```

Variant 2 : Als JSON-Objekt in externe Datei rendern


```
// Lädt die Icon-Definitionen direk aus einem JavaScript-File. Damit die Einzelangabe
// per Optionen
dashlet.addOption("leaflet.marker.icon.iconDef",
readResourceFileAsString("dashlet_leaflet_icondef.js"));

// Aktuell noch notwendig, damit der Lade-Key erkannt wird / Nach Korrektur entfernen
dashlet.addOption("leaflet.marker.icon.1.iconId","red");
```



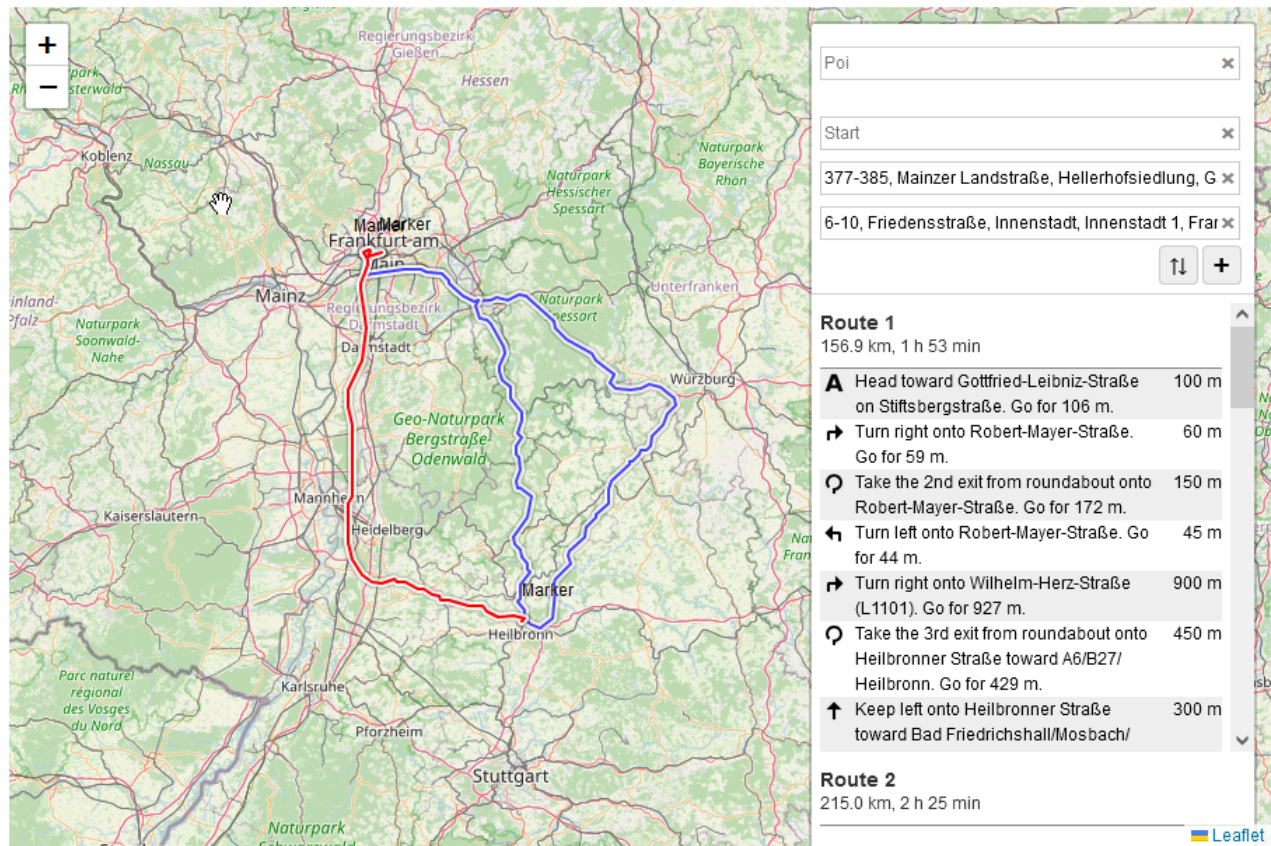
```
1  [
2  {
3    iconUrl: 'data:image/png;base64,iVBORw0KGgoAAAANSUhEUgAAACAAAAAwCAIAAAD/zu84AAAAABnRSTlMA/wI
4    iconId: 'red',
5    iconSize: [20, 30],
6    iconAnchor: [10, 30],
7    popupAnchor: [-3, -76],
8    shadowUrl: 'data:image/png;base64,iVBORw0KGgoAAAANSUhEUgAAACkAAAApCAQAAAAACach9AAACMUlEQVR4A
9    shadowSize: [30, 40],
10   shadowAnchor: [8, 38],
11  },
12  {
13   iconUrl: 'data:image/png;base64,iVBORw0KGgoAAAANSUhEUgAAACAAAAAwCAIAAAD/zu84AAAAABnRSTlMA/wI
14   iconId: 'green',
15   iconSize: [20, 30],
16   iconAnchor: [10, 30],
17   popupAnchor: [-3, -76],
18   shadowUrl: 'data:image/png;base64,iVBORw0KGgoAAAANSUhEUgAAACkAAAApCAQAAAAACach9AAACMUlEQVR4A
19   shadowSize: [30, 40],
20   shadowAnchor: [8, 38],
21  },
22  {
23   iconUrl: 'data:image/png;base64,iVBORw0KGgoAAAANSUhEUgAAAB4AAAAwCAIAAACJ9F2zAAAAABnRSTlMA/wI
24   iconId: 'blue',
25   iconSize: [20, 30],
26   iconAnchor: [10, 30],
27   popupAnchor: [-3, -76],
28   shadowUrl: 'data:image/png;base64,iVBORw0KGgoAAAANSUhEUgAAACkAAAApCAQAAAAACach9AAACMUlEQVR4A
29   shadowSize: [30, 40],
30   shadowAnchor: [8, 38],
31  },
32 ]
```

Leaflet Routing

Einführung

Es gibt die Möglichkeit ein Routing Option zu machen mit Leaflet.

Karte



Derzeit sind 3 Routing Modules implementiert

- HERE
- Mapbox
- Osmr

Dashlets erzeugen

Für die Routing Option zu aktivieren , muss die folgende Option definiert werden.

```
dashlet.addOption("leaflet.routing.feature", "true");
dashlet.addOption("leaflet.routing.apikey", "***YOUR*APIKEY***");
dashlet.addOption("leaflet.geocoder.apikey", "***YOUR*APIKEY***");
```

Durch das Hinzufügen von Optionen können Dashlets individuell angepasst werden.

EChart Dashlet erzeugen

```
dashlet.addOption("leaflet.geocoder.type", "nominatim");
dashlet.addOption("leaflet.routing.type", "here");
```

FAQ

Wie kann Sprache des Routing-Modules gesetzt werden?

Der Sprache kann mit die folgende Option gesetzt werden.

```
dashlet.addOption("leaflet.routing.language", "de"); //en
```

Wie kann der Units des Routing-Modules angepasst werden?

Der Unit des Routing-Modules kann mit der folgenden Option angepasst werden.

```
dashlet.addOption("leaflet.units", "imperial"); // /* Enum [imperial | metric] */
```

Wie kann Logging Option gesetzt werden?

Es gibt die Möglichkeit ein Logging auf der Console einzuschalten, und kann mit die folgende Option gesetzt werden.

```
// Logging von Routing Module  
dashlet.addOption("leaflet.routing.logging", "true");
```

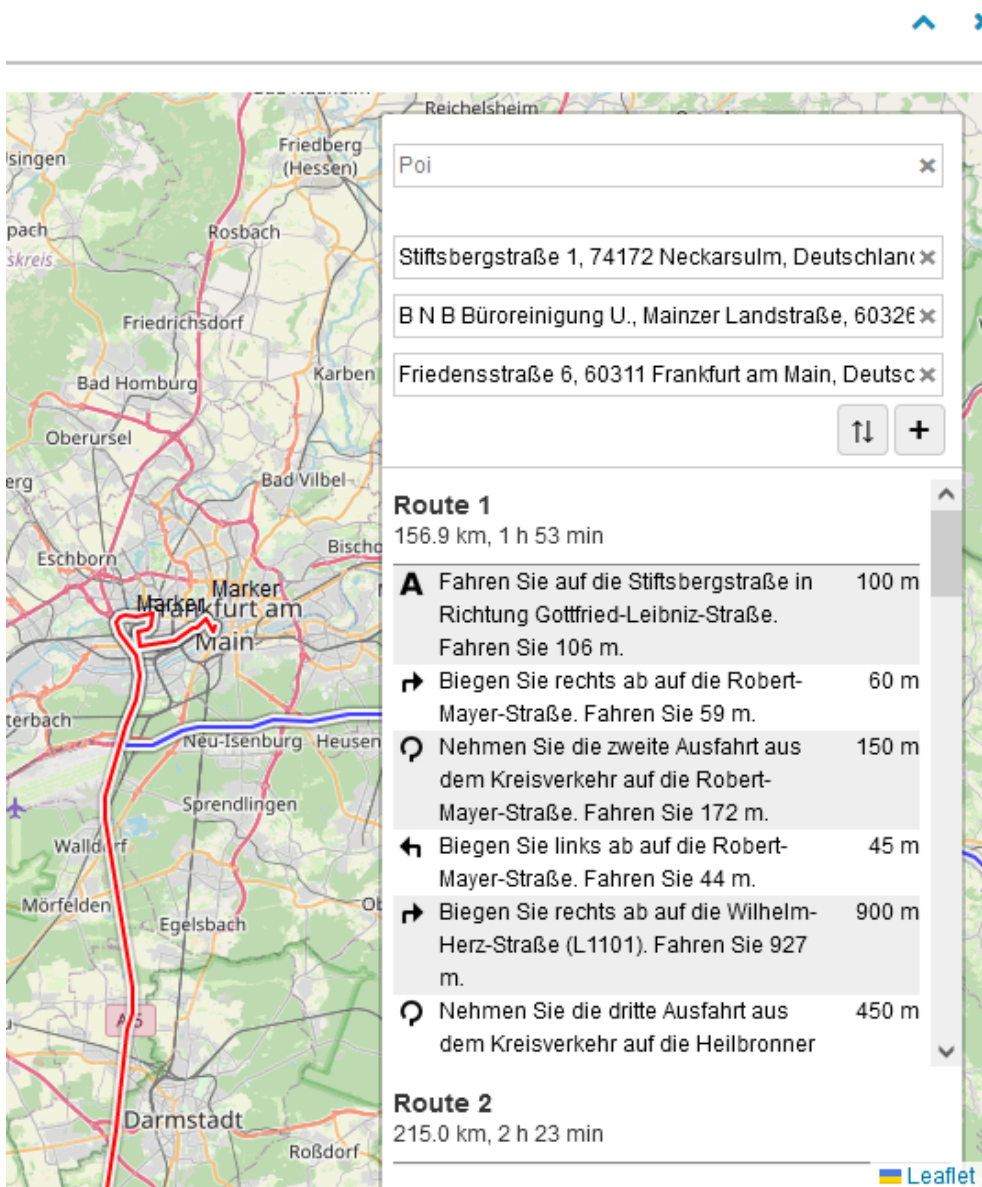
Wie kann der Routing-Control Box ein und ausgeschaltet werden?

Der Routing-Control-Box kann mit der Funktions- Aufruf ein-ausgeschaltet werden.

```
// Einschalten  
leaflet1_routing_control.show();  
  
// Ausschalten  
leaflet1_routing_control.hide();
```



'**leaflet1**' ist die ID des Leaflet-Dashlets.



Wie kann der transport-Mode für die Routing angepasst werden?

Der transport-Mode kann mit die folgende zwei Variante angepasst werden:

Variante 1 : Mit Option

```
dashlet.addOption("leaflet.routing.transportMode", "car");
```

Variante 2 : Javascript Call (zb. Falls die transport-mode sollte von Buttons-Onlick Event angepasst werden)

```
btn = buttonbar.addButton("Auto", true);  
btn.addOption("button.onclick",  
"leaflet1_routing_control.getRouter().options.transportMode = 'car';  
leaflet1_routing_control.route();");  
btn = buttonbar.addButton("Zu Fuß", false);  
btn.addOption("button.onclick",  
"leaflet1_routing_control.getRouter().options.transportMode = 'pedestrian';  
leaflet1_routing_control.route();");  
btn = buttonbar.addButton("Mit dem Fahrrad", false);  
btn.addOption("button.onclick",  
"leaflet1_routing_control.getRouter().options.transportMode = 'bicycle';  
leaflet1_routing_control.route();");
```



'**leaflet1**' ist die ID des Leaflet-Dashlets.

Wie kann Icons für Start, End und Way-points definiert werden?

Customized Icons für Start, End und Way-Points kann als folgendes definiert werden.

```

dashlet.addOption("leaflet.routing.iconWaypointStart","{" +
"iconId: 'icon-flag-green'," +
"iconUrl: 'data:image/
png;base64,iVBORw0KGgoAAAANSUHEUgAAAB0AAAApCAIAAABGGrXjAAAABnRSTlMA/wD/
AP83WBt9AAAAACXBIWXMAAA7EAAAOxAGVKw4bAAAB6LEQVRiX+3Vv2/
TQBQH80+dz3Fc15ErOyStCAiCyg8JUZWx6oaQ+B06s8AfwsjIQAf+DyQkBuJSSASqqpGKmiYFLGqapk79I7nz
Y2io1JCykGx+4z3po7t3994xIsIUQgDIQNsbc8oVQ3nPBFn/
XYU+GknLDEAW5gVwykXSo4+yxm7j0vJJJTxHF0AHiz/
+7pl1e3zYUPqy+8QmnzePd18+37YKsdBykk0A0AjAyIOX22ZrQ3rPlFe+G6WXYLtsH1AamjftiMDrfC1savnS
eV5fWl5wLax24DBa0Rf28lgS0stU8vv8oAfQUiECEjAGBIIX157J92N6M9BAABRMgABnAGBhAguIQa1oFUhoy
giDIihoGUSCXUxboTAILC6PpIn0kAH50bxEVYTCdyN3dzd/ycHNPjgg8z6s/c4QwaG3a5ykAYrL/
YJI0uQfAxLoHA2Q24j8oPlp2bNd0zhUmEExm14qARfts5Pdjt+X7SibUBNI7zUazIhvGwWF+rrIy6BNLA1+8/
W3HvWFrXsjNGKv2RdPajw4Mk00r3FGULfeaa6d21r9ZMj4H97UJj/PGVpX/XbkYz6la1blXzd5a7uZu7uZu7/
xkCOPtHORSBTcxLRPS5u9eMfmqMr7r3HN2amDuNOvwGtT3KCBbCfQ4AAAAASUVORK5CYII='," +
"iconSize: [23, 30]," +
"iconAnchor: [2, 29]," +
"popupAnchor: [-3, -76]," +
"shadowUrl: 'data:image/
png;base64,iVBORw0KGgoAAAANSUHEUgAAACkAAAApCAQAAAAACach9AAACMUlEQVR4Ae3ShY7jQBAE0Aoz/
f9/HTMzhg1zrdKUrJbdx+Kd2nD8VNudfsL/Th///dyQN2TH6f3y/
BGpC379rV+S+qqetB0xImNQXL8JCAR2V4iMQXHGNJxeCfZXhSRBcQMfvk0WUdtfzLLgAENmZDcmo2TVmt80SM
2eXxBp3DjHSMFutqS7SbmemzBiR+xpKCNUIRkdkkYxhAkyGoBvyQFEJEefwSmmvBfJuJ6aKqKwNAkvGZ0aZXT
UgFqYULWNSHUckZuR1HIIimUExutRxwz0LR0IG4vKmCKQt364mIlhSyzAf1m9lHZHJZrLA0MMztRRiKimp/
rpdJDC9Awry5xTZCte7FHTuS8wJgeYGreX28xNTd086Dik7vUMscQ0a8y4DoGtCCSkAKlNwpgNtpjhrc6MIHU
kR6YwXxs6Sc5xqn222mmCRFzIt8lEdKx+ikCtg91qS2WpwVfBelJCiQJwvzixfI9cxZQWgiSJelKwBELKYtD
Ob2MFbhmUigbReQBv0Cg4+qMXSxXSyGUN4UbF8l+7qdSGnTC0XLCmahIgUHLhL0hpVctw4CzYXvLQWQbJNmXo
Cs0KAxSgB3no75avolkRw8iIAFcscd02e9iyCd8tHwmeSSoKTowIgvscSGZU0A7PuCN5b2BX9mQM7S0wYhMNU
74zgsPBj3HU7wguAfnxxjFQGBE6pwN+GjME9zHY7zGp8wVxMShY9NXvEWD3HbwJf4gi04CFIQxXScH1/
TM+04kkBiAAAAAE1FTkSuQmCC'," +
"shadowSize: [30, 40]," +
"shadowAnchor: [8, 38]," +
"}");

dashlet.addOption("leaflet.routing.iconWaypointEnd","{" +
"iconId: 'icon-flag-red'," +
"iconUrl: 'data:image/
png;base64,iVBORw0KGgoAAAANSUHEUgAAAB0AAAAoCAIAAACNRmZGAAAABnRSTlMA/wD/
AP83WBt9AAAAACXBIWXMAAA7EAAAOxAGVKw4bAAAB40LEQVRiX+3Xu07jQBQG4P9MxhkU0SE0kCICHJCWW7Epu
JRUFNDwGjwRb5CKHGFvgGhZcQtCikApEsUEyDoxnzmzRZYQICDQ7lbrv/
SxvnPGL7FMzAygC33teF7CddELc1SrRfW6abUAJfXxJow4uRwphffDxpiHB5nNSgD1nZ3K1paYmf+vy89r10
u10sl/
+AguLzUd3fQGgCkLjMmMzY2ND2dmptLzc6qyUknL4PjsNb69jasVILj4+DwcGhl5dv2tgTQ2N3LZjM40mpfXb
nF4o/NTTo7ewRM/
yRaG99n39fn58HeHg0me4IQbK0EEgABEnAWFgBIACSE7SnMutUyr9BuBbAvjxAAawnQgAYAKICE+02+Srfwmf
D7pc8SX03sxm7svo38oGF3K6G+zcEC/
0HrSwCMGezaMEwCUTarpqaShYIcHgZz1GxG1WpYrUaNHmKmp078xBEGACuEmpgY4LIx3tra6MZGZnU1WShQIV
Fc0lr7fqdSCU5P5f56cdC4uwpbX8Pa0UqpcBH00tL3vp6enHxjWutUGq+VBq8RimdfN7J59PLy8/
NwpCtpWSyf4JB14HoS/eHlKL4+Y3d2I3d2P0/XGbufXf/pkv/YF5i5na5rGs1C7jF4otfrT/
ILw2oxeWmtq1ZAAAAAE1FTkSuQmCC'," +
"iconSize: [22, 30]," +
"iconAnchor: [2, 29]," +
"popupAnchor: [-3, -76]," +

```



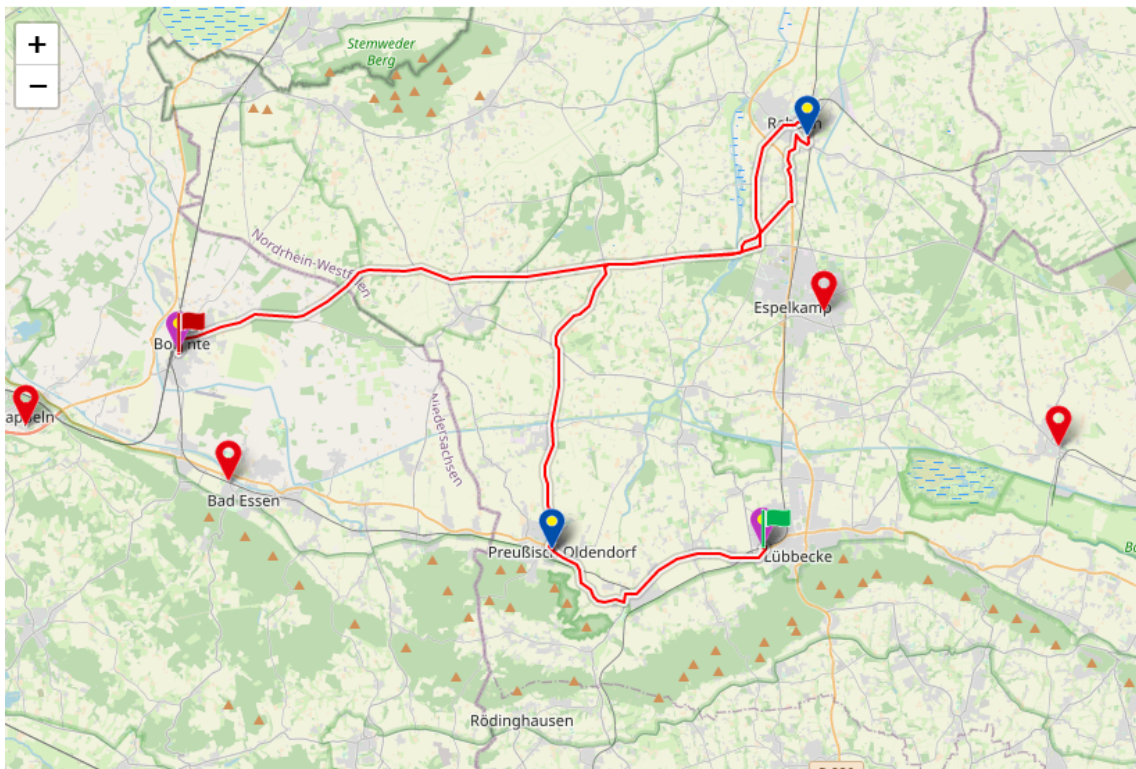
```
"shadowUrl: 'data:image/
png;base64,iVBORw0KGgoAAAANSUHEUgAAACkAAAApCAQAAAAACach9AAACMUlEQVR4Ae3ShY7jQBAE0Aoz/
f9/HTMzhglzrdKURJbdx+Kd2nD8VNudfsL/Th//dyQN2TH6f3y/
BGpC379rV+S+qqetB0xImNQXL8JCAR2V4iMQXHGNJxeCfZXhSRBcQMfVvK0WUdtfzLLgAENmZDcmo2TVmt80SM
2eXxBp3DjHSMFutqS7SbmemzBiR+xpKCNUIRkdkkYxhAkyGoBvyQFEJEefwSmmvBfJuJ6aKqKwNAkvGZ0aZXT
UgFqYULWNSHUckZuR1HIImUExutRwxz0LR0IG4vKmCKQt364mIlhSyZaf1m9lHZHJZrLAOMMztRRiKimp/
rpdJDC9Awry5xTZCte7FHTuS8wJgeYGreX28xNTd086Dik7vUMscQ0a8y4DoGtCCSkAKlNwpgNtpbjrC6MIHU
kR6YWxxs6Sc5xqn222mmCRFzIt8lEdKx+ikCtg91qS2WpwVfBelJCiQJwvzixfI9cxZQWgiSJeLKnwBELKYtD
0b2MFbhmUigbReQBv0Cg4+qMXSxXSyGUN4UbF8l+7qdSGnTC0XLCmahIgUHLhLOhpVCTw4CzYXvLQWQbJNmXo
Cs0KAXSgBJno75avolkRw8iIAFcscd02e9iyCd8tHwmeSSoKTowIgvscSGZUOA7PuCN5b2BX9mQM7S0wYhMNU
74zgsPBj3HU7wguAfnxxjFQGBE6pwN+GjME9zHY7zGp8wVxMShYX9NXvEWD3HbwJf4gi04CFIQxXScH1/
TM+04kkBiAAAAAEFTkSuQmCC'," +
"shadowSize: [30, 40]," +
"shadowAnchor: [8, 38]," +
"}");
```

```
dashlet.addOption("leaflet.routing.iconWaypointVia","{" +
"iconUrl: 'data:image/
png;base64,iVBORw0KGgoAAAANSUHEUgAAAB4AAAAwCAIAAACJ9F2zAAAAABnRSTlMA/wD/
AP83WBt9AAAAACXBIWXMAAA7EAAAOxAGVKw4bAAAF00LEQVRYw7WYw2wUVRjH/
+cys5d2aZfSdtnSlu1uV6ktbCil1JAiUJa2G8RQgomCmbQFgTbRmKAmRA2Jl0QU4iWaGF9MTAi8ICg+iK2GRE
0pIaEFW1trUKFQagtdddr7M3N8aLHtzuyLR/bnHPmd77zXf7fZIGQAv/P4MmXw1H14sBoZ/
9I77WxkWAMgD1T8jptlw77ssVZVh0fC/rqU0jTbweOnfujbzCkkYltE/
cjAKhQ3PmWx1cXPbPBVZRrNSQVvU0iivbB17++caJn0AQIFZp6DzpzUAbC51vFK01LWgIemdMU6Buj480fnP/
q4t/QYhBaa08SCio1LLN/
1LLlyDYnRF8fCW9+88eOgbtQxmcZMnNlsfXkyw8X5FimbjuY9YjuOdHT8Fpw1F4Ay3vn73R1H00IRxQB96PiVs
92jUCNzzDU10nZ59PVjL+PRnf0jR0/3Q4sZv0YZmAlMBpPBTKDMeJsWe//MQEffyIzk03K6PxQjBnEjBJQ/
4DBvWuF8qCgLEN1X75w6f61ncByagrjsElo4Rt471ffF81WTYey7Hlz+4tmxsBKPJlTi70BW7wuPeq0mRggBI
IQIRbR3v+w5dKI3pqj6V2wWfuHweo8jkWJo6xoi8DQ5MM7yw9uK8sw8wkuAEJIhpkd3FZ2eGc57k10N3wsGu
8uDU36+tyVYSP/
Sg2+vJaAJ1HYWgKeBl8umKRf+qF7CAAVQvzy522j6hDndYuTJ0VznQt6dRNa77UxIUdHY9rI3Vh8KRNiLYTPl
Z0c7XNlZ8jQuUWMBKPhqEpVtCQUTScSxCSx5MIGwGpiMicTgjWNjKiKapGZU4tMos/
WYjguDJ0J0X5DN20BsNqvE8ILDkX0aUypw67Jf5kiJjg7V23kqPbu4di4DdpdA672SwzCqC8KAUE6qKhfvRN/
/
BYNBF3eCz64Zl+CFWvhRVFWZPJt7ZigUGGaGrP9fCeJztDEdWo+2jPfXKhd3AcmqRpkHVL8ybRtWW5hTkyiE4
Z10iJn29ue037tq6bUUX9t1G0dd2se7X9+E83o0ruRFhhjlxbtmBKr1/6/
NLbJweMZY9KnMGtB3I7MgH0Dwb7bkQUFcZaxkwHNrve2lExhe79687KA+23Q4k6CwG1k9cSKjTnuKUR0s/
Cz7+zvnRh5pSoegvmbVLVAColiJmApkKNQ0ombJUAqNRUuzDBndEKdvsXS0TRZWH6g0he2bPRbdBlqr05fp/
DUG7SGkzy+/JXltoN0AD21rsMkinNoal760tmuGf6g9/nWOW1J/
Z44kG16tJsv8+REM0Z2e0vmZ0fsWdjCWckIRpAU02B12EB5bMxmZfmmZtqFsVPxz3bLLy5zmVQmclMZrs2uGw
WngINYPuawnwbNRAsYy7Nt9Hta4oMLq0fcs63PLmmON1gUumJ2iLnFEtaAAC76oozJC214YRmyuJZv8v4UMPZ
JYvmbVm1KLXhVHpspfPBAtss0AD2Nbhkqhp8aUyzWabq/
saEyZoQXe3NCVQ6khnOpEDlwmpvzqzRAFoDHioSCRahQm1pTFzfydCPl0cmFCwm1S3NX1uRn0c0gNaAG5recA
JNaQ24U8Q4+bLfl1db1htf95SvXrKgfrnjvtCM0pbGEl1bEa0BD6PkvTAANLutrHJnT6UK5ZWurE0rHKnrNOU
Ok8T2N0xzKyH7G91mmf0HaABNNc7ywKxQDsrLnBlba5xpqUs6mzLMfF99CZgZzLy3viTTkpZypSv5T60tnor
KECeXlec5isk/Z8WExtJ2h8T/wD6c9lf8sGwsAAAAABJRUErkJggg=='," +
"iconId: 'blue'," +
"iconSize: [20, 30]," +
"iconAnchor: [10, 30]," +
"popupAnchor: [-3, -76]," +
```

```
"shadowUrl: 'data:image/
png;base64,iVBORw0KGgoAAAANSUHEUgAAACkAAAApCAQAAAAACach9AAACMUlEQVR4Ae3ShY7jQBAE0Aoz/
f9/HTMzhglzrdKURJbdx+Kd2nD8VNudfsL/Th//dyQN2TH6f3y/
BGpC379rV+S+qqetB0xImNQXL8JCAR2V4iMQXHGNJxeCfZXhSRBcQMfVvK0WUdtfzLLgAENmZDcmo2TVmt80SM
2eXxBp3DjHSMFutqS7SbmemzBiR+xpKCNUIRkdkkYxhAkyGoBvyQFEJEefwSmmvBfJuJ6aKqKWNakvGZ0aZXT
UgFqYULWNSHUckZuR1HIIimUExutRwxz0LR0IG4vKmCKQt364mIhSyZaf1m9lHZHJZrlAOMMztRRiKimp/
rpdJDC9Awry5xTZCte7FHTuS8wJgeYGrex28xNTd086Dik7vUMscQ0a8y4DoGtCCSkAKlNwpgNtpjrc6MIHU
kR6YWxxs6Sc5xqn222mmCRFzIt8lEdKx+ikCtg91qS2WpwVfBelJCiQJwvzixfi9cxZQWgiSJeLKnwBELKYtD
0b2MFbhmUigbReQBv0Cg4+qMXSxXSyGUN4UbF8l+7qdSGnTC0XLCmahIgUHLhLOhpVCTw4CzYXvLQWQbJNmXo
Cs0KAxSgBJno75avolkRw8iIAFcscd02e9iyCd8tHwmeSSoKTowIgvscSGZUOA7PuCN5b2BX9mQM7S0wYhMNU
74zgsPBj3HU7wguAfnxxjFQGBE6pwN+GjME9zHY7zGp8wVxMShYX9NXvEWD3HbwJf4gi04CFIQxXSCh1/
TM+04kkBiAAAAAElFTkSuQmCC'," +
"shadowSize: [30, 40]," +
"shadowAnchor: [8, 38]," +
"}");

dashlet.addOption("leaflet.routing.iconWaypointPoi","{" +
"iconId: 'gray'," +
"iconUrl: 'data:image/
png;base64,iVBORw0KGgoAAAANSUHEUgAAAB4AAAAwCAIAAACJ9F2zAAAAABnRSTlMA/wD/
AP83WBt9AAAAACXBIWMAAA7EAAA0xAGVKw4bAAAEq0lEQVRYw7VYyy9zTxieMzP9WtGEEtWgro2m1RaVlghyq
tVKNO1ERpWBNpb+DXsLYS2xoAu0QiwtRCzEJeISIR1CVL0duZ8ixH69UxPyy+/
d9Uzc+aZZ573nWfmlBNFEfw/
geW7U6nU1dXVxcVFJBj5e3sDAJSWlp0uubm5vr6eqVS+Rvoh4eH7e3tnZ2dWCwGIQQA0PVxHAcAIIRotdqn
h63211ZWclE4KSCZDKZjY2NYDD48fBCCGEMEdCCCGEKpUqEAj4fD6McQH05+fnum5g4MDQRCKSQPhcQghi8
Uy0TlZXl6eF/r5+XlmZub6+jqTyfwsYxjX19dPT09rNJrvZWVnbHZ29he4VMPr6+vZ2dlkMsmAXllZOT4+/
gXuF/rJycny8nIu90XLZSgUEgQhX8YwxgghhBDGmBaMnARBCIfD5+fn/
xTf+vp6Jp0R5o3j0AiHvqu12+11dXUAGJubm729vfv7e0JIZvuiKAqCEAqFpqamPqGj0eje3p6UMsdxGONAI0
D3+5VKJaIoURRHR0dXVleDwaCUjSAI+/
v70Wi0uoroaAgC0jo5SqrST8stExMjiIEqlori0UaUjjoyMTExmFDMvE08mk4eHh59an5ycSIVDCNlsNp/
Ply9vPp/PZrMhhKRDFBCKonh7eyuLLIoiz/PyVchZPHPg3d2dKIownU5T38mRQqFQNDQ0yEM3NDRgjKWYvL+/
p9NpKAgCs+YwxiqVSh5aqVQqFAqmXxJC0EKH+PPnj7Q7kUi8vLzIQ7+
+viYSCeaUGG0IMS4rK8tZFFWQmd7s0Do6+no5W8yysjKMMQQA6PV6ZhmFw+F4PJ4PNx6Pb21tMutWr9d/Fp/
ZbJaaMiEkFostLCyK02mmmvPz85FIhDmwra3tczeaTKaKiorHx0fp1trd3X16ehobG2ttbaVmn8lkTk9Pl5aW
zs70mBu4oqLCZDJ9+/Xi4uLa2hrT9hBC1Ea0Wi11BWog+erK7/
ePj49/2xPP85ubm8yThVbn3d1dJBKhOch3pNHd8LXRPu1Rp9M5nc58bkkVpHPkw6Xr6+7u1ul0uUeB2+3mOE5
aKkUGHTg40Mg4ZVpaWiwwiwx+YAQ2mw2g8HAgAYAE1lemfXKByHE7Xb/
M1n2A532F8QRQgaDwWaz5YWGEFLFf8Ha7XbneHcuQYfDUVVV9SPit0qdTmdue85zSukJz/M/heZ5XurADIj+/
n61Wl2kLBzHqdXqvr4+xpTSJo1G09vbyzz0mAns7e3Nvo/
JQQMAXC6XQqEoSJzeJlWuF1soZmttbW1XV1dBxSGEDoejtrb2B9B0+0AIZYhTVxgaGso7cb40g8HQ0dEhQxxC
2NnZmb2zi4UGAPh8Ppn0yFE5gJUANpsNlutVmapIISsVqvZbP4lNCX0NCxCyPDwcIEky3e3t7cbjcYc4ggho
9GYY0Y/huY4Tqq4KIq0fv4TNACgq6ursbHxCwhC2NTUZLfbC3tLwTcwXl6vN/
t+7fV6mXe5H0MDAJx0Z01NDf0GrampcTgcRTliMS+pVCqPx0M/
kzweT8Eb70dyQXExMDDw9PREfxQ5hCv+T4vsz/9i4i9EbQm7gW5WYQAAAABJRUS5ErkJggg=='," +
"iconSize: [20, 30]," +
"iconAnchor: [10, 30]," +
"popupAnchor: [-3, -76]," +
```

```
"shadowUrl: 'data:image/
png;base64,iVBORw0KGgoAAAANSUHEUgAAACkAAAApCAQAAAAACach9AAACMUlEQVR4Ae3ShY7jQBAE0Aoz/
f9/HTMzhg1zrdKurJbdx+Kd2nD8VNudfsL/Th//dyQN2TH6f3y/
BGpC379rV+S+qqetB0XImNQXL8JCAr2V4iMQXHGNJxeCfZXhSRBcQMfvk0WUdtfzLLgAENmZDcmo2TVmt80SM
2eXxBp3DjHSMFutqS7SbmemzBiR+xpKCNUIRkdkkYxhAkyGoBvyQFEJEefwSmmvBfJuJ6aKqKWnAkVgZ0aZXT
UgFqYULWNSHUCKZuR1HIIimUExutRwxz0LR0IG4vKmCKQt364mIhSyZaf1m9lHZHJZrLA0MMztRRiKimp/
rpdJDC9Awry5xTZCte7FhtuS8wJgeYGrex28xNTd086Dik7vUMscQ0a8y4DoGtCCSkAKlNwpgNtpjrc6MIHU
kR6YWxxs6Sc5xqn222mmCRFzIt8lEdKx+ikCtg91qS2WpwVfBelJCiQJwvzixfI9cxZQWgiSJeLKnwBELKYtD
0b2MFbhmUigbReQBV0Cg4+qMXSxXSyGUn4UbF8l+7qdSGnTC0XLCmahIgUHLhLOhpVCTw4CzYXvLQWQbJNmXo
Cs0KAxSgBJno75avolkRw8iIAFcscd02e9iyCd8tHwmeSSoKTowIgvscSGZUOA7PuCN5b2BX9mQM7S0wYhMNU
74zgsPBj3HU7wguAfnxxjFQGBE6pwN+GjME9zHY7zGp8wVxMShYX9NXvEWD3HbwJf4gi04CFIQxXSCH1/
TM+04kkBiAAAAAEFTkSuQmCC'," +
"shadowSize: [30, 40]," +
"shadowAnchor: [8, 38]," +
"}");
```



4.5.19 Kerzendigramm (Candlestick)



Information

Dieser Charttyp ist derzeit noch nicht im Dashlet-Plugin verfügbar. Bitte kontaktieren Sie unseren [Support](#), wenn Sie an der Nutzung dieser Darstellungsform interessiert sind.

Einführung

Ein Candlestick-Chart (auch als japanisches Candlestick-Chart bezeichnet) ist ein Finanzdiagramm, mit dem Preisbewegungen beschrieben werden.

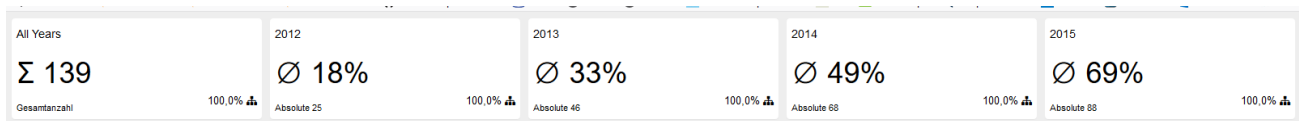


4.5.20 KPI-Box / -Leiste

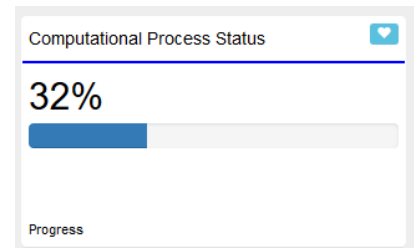
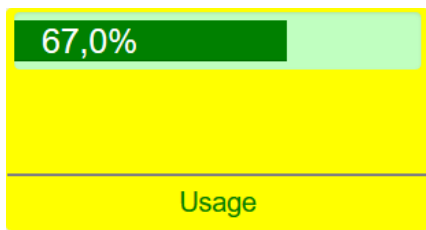
Einführung

Das KPI Dashlet zeigt die KPI (Key Performance Indicator) in verschiedene Boxen Format an.

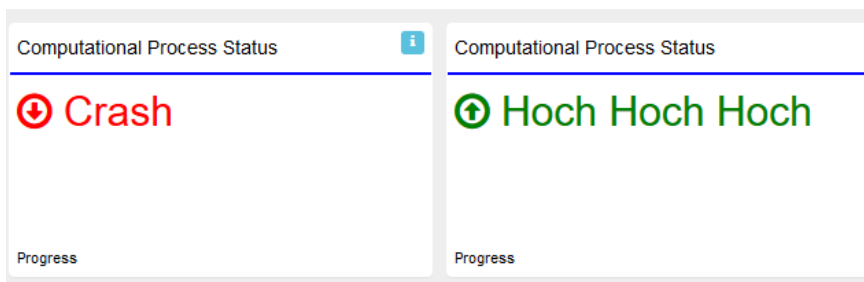
1. Trendbox type



2. Percentage-Box KPI

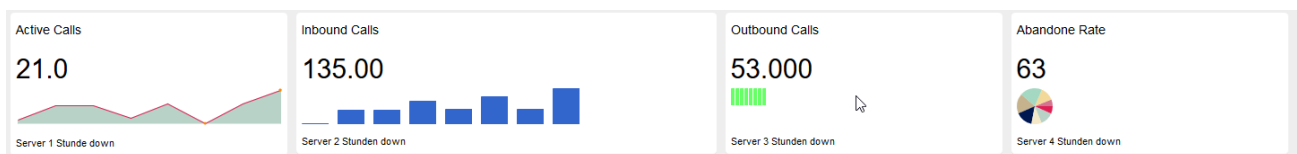


3. Updown Box



4. Charts Box

a. Mit einfache Charts

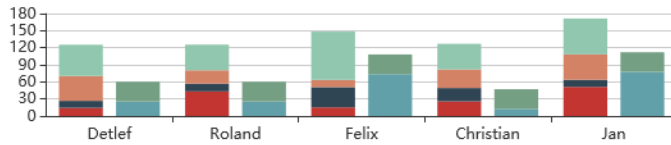


b. Mit ECharts

Es gibt auch die Variante wobei die ECharts innerhalb ein KPI-Box darstellen können.

Inbound Calls

135.00



Server 2 Stunden down

Dashlets erzeugen

Die KPI-Dashlet ist mit die folgende Code Block erstellt.

EChart Dashlet erzeugen

```
// DashletManager erzeugen
ExpertsDashlet.createDashletManager(reportContext, "DashletManager");

// Dashlet erzeugen
dashlet = DashletManager.addDashlet("DashletKpiBox", "KPI-Bar", "KPI-Bar", false,
null);
```

Dashlet Optionen

Durch das Hinzufügen von Optionen können Dashlets individuell angepasst werden.

1. Trendbox KPI erstellen

Dashlet Optionen

```
// trendbox KPI Optionen
dashlet = DashletManager_KPI.addDashlet("DashletKpiBox", "KPI-Bar", "KPI-Bar",
false, null);
dashlet.setId("KpiBox1");
dashlet.setResponsiveGrid("12-12-12");
dashlet.addOption("kpi.box.count", 5); // 5 Boxen erstellen
// KPI-Values von ein MemoryBuffer anzeigen
mbr.openBuffer();
var row = mbr.fetchBuffer();
var countBuf = 0;

// Create the summary of KPI for "Total"
while (row != null) {
    dashlet.addOption("kpi.box.i[" + countBuf + "].responsive","2-4-6-8");
    dashlet.addOption("kpi.box.i[" + countBuf + "].type"          , "trendbox");
    //PercentValue
    if (countBuf == 0) { // the values should precide with Sigma or EmptySet
        dashlet.addOption("kpi.box.i[" + countBuf + "].value"      , "&Sigma; " +
Formatter.format(mbr.getValueFromRow(row, "Value"),"#,##0"));
        dashlet.addOption("kpi.box.i[" + countBuf + "].footer"    ,
"Gesamtanzahl");
    } else {
        dashlet.addOption("kpi.box.i[" + countBuf + "].value"      , "&empty; " +
Formatter.format(mbr.getValueFromRow(row, "PercentValue"),"#,##0") + "%");
        dashlet.addOption("kpi.box.i[" + countBuf + "].footer"    , "Absolute
" + Formatter.format(mbr.getValueFromRow(row, "Value"),"#,##0"));
    }
    dashlet.addOption("kpi.box.i[" + countBuf + "].title"         ,
mbrKategorie07.getValueFromRow(row, "Series"));
    dashlet.addOption("kpi.box.i[" + countBuf + "].trendvalue"    , 1);
    dashlet.addOption("kpi.box.i[" + countBuf + "].trendvalueformat","#0.0%");
    dashlet.addOption("kpi.box.i[" + countBuf + "].icon"          , "fa fa-
sitemap"); //source-name: https://fontawesome.com/
    countBuf++;
    row = mbr.fetchBuffer();
}

mbr.closeBuffer();
mbr.clearFilterCondition();
```

2. PercentageBox KPI

```
// PercentageBox KPI Optionen
dashlet = DashletManager_KPI.addDashlet("DashletKpiBox", "KPI-Bar", "KPI-Bar", false,
null);
dashlet.setId("KpiBox2");
dashlet.addOption("kpi.box.i[0].type","percentbox");
dashlet.addOption("kpi.box.i[0].value","67");
dashlet.addOption("kpi.box.i[0].valueformat","#,##0.0");
dashlet.addOption("kpi.box.i[0].style","background-color: yellow;");
dashlet.addOption("kpi.box.i[0].progressbar.background.style","background-color:
#c0ffc0; height: 42px; margin-top: 10px; padding-top: 6px;");
```

3. Updown Box

```
// UpDownBoxKPI Optionen
dashlet = DashletManager_KPI.addDashlet("DashletKpiBox", "KPI-Bar", "KPI-Bar", false,
null);
dashlet.setId("KpiBox2");
dashlet.addOption("kpi.box.i[0].type","updownbox");
dashlet.addOption("kpi.box.i[0].value","-32");
dashlet.addOption("kpi.box.i[0].valueformat","#0");
dashlet.addOption("kpi.box.i[0].footer","Progress");
dashlet.addOption("kpi.box.i[0].title","Computational Process Status");
```

4. Charts Box

a. Mit einfache Charts

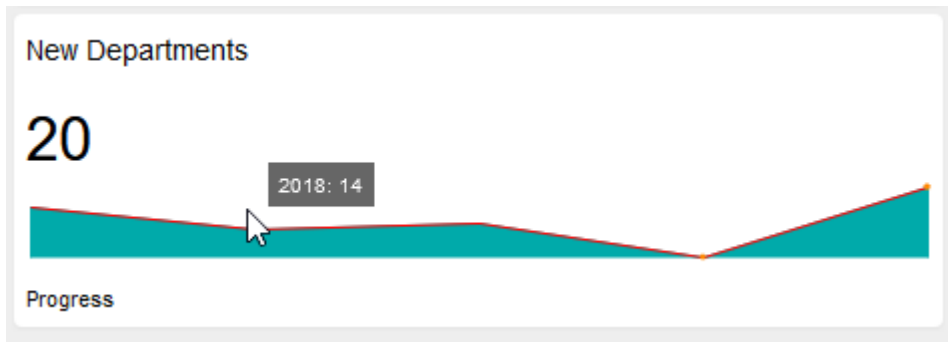
```
// ChartsKPI Optionen
dashlet = DashletManager_KPI.addDashlet("DashletKpiBox", "KPI-Bar", "KPI-Bar", false,
null);
dashlet.setId("KpiBox4");
dashlet.addOption("kpi.box.i[0].value" , "20"); // Value
dashlet.addOption("kpi.box.i[0].trendseries" , "20,14,16,7,25"); // Trend Values
dashlet.addOption("kpi.box.i[0].trendvalueformat" , "#0.0"); // Trend Value Format
dashlet.addOption("kpi.box.i[0].title.visible","true"); // Title Visibility
dashlet.addOption("kpi.box.i[0].title","New Departments"); // Title
dashlet.addOption("kpi.box.i[0].footer.visible","true"); // footer Visibility
dashlet.addOption("kpi.box.i[0].footer","Progress"); // footer

// Chart Typen
dashlet.addOption("kpi.box.i[0].type","trendlinebox"); // Other Types (trendpiebox,
tristate , trendpiebox)

// Optionen
dashlet.addOption("kpi.box.i[0].trend.chart.height","'40'");
dashlet.addOption("kpi.box.i[0].trend.chart.lineColor","'#FF0000'");
dashlet.addOption("kpi.box.i[0].trend.chart.fillColor","'#0AA'");
dashlet.addOption("kpi.box.trend.chart.disableHighlight", "true");

// Map the offset in the list of values to a name to use in the tooltip
//dashlet.addOption("kpi.box.trend.chart.tooltipFormat", "\"{{y}}\"");
dashlet.addOption("kpi.box.trend.chart.tooltipFormat", "\"{{offset:offset}}: {{y}}\"");
;
dashlet.addOption("kpi.box.trend.chart.tooltipValueLookups", "{\n"
+ "    'offset': {\n"
+ "        0: '2017',\n"
+ "        1: '2018',\n"
+ "        2: '2019',\n"
+ "        3: '2020',\n"
+ "        4: '2021',\n"
+ "    }\n"
+ "}");

// Bar Chart Optionen
dashlet.addOption("kpi.box.i[1].trend.chart.type","bar");
dashlet.addOption("kpi.box.i[1].trend.chart.barWidth","30");
dashlet.addOption("kpi.box.i[1].trend.chart.barSpacing","10");
dashlet.addOption("kpi.box.i[1].trend.chart.zeroAxis","true");
```



b. Mit ECharts

Es ist auch Möglich ein E-Chart innerhalb ein KPI-Box zu darstellen.

```
// -----EChart Variante-----
dashlet = DashletManager_KPI.addDashlet("DashletKpiBox", "KPI-Bar", "KPI-Bar", false,
null);
dashlet.addOption("kpi.box.i[0].echarts.inline.id","chart1");
dashlet.addOption("kpi.box.i[0].chart.height","200px");

// Echart erstellen und die beide Optionen definieren
var dashlet2 = DashletManager.addDashletEChart("bar", "", "", true, mbrKpi);
dashlet2.addOption("chart.inline.mode", "true");
dashlet2.setId("chart1");
```

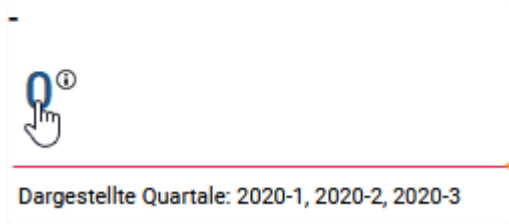
FAQ

Wie kann ein Link für die KPI-Wert gesetzt werden?

Ein Link für die KPI-Wert kann mit die folgende Option gesetzt werden.

```
var txtKpiValue = 0; // KPI-Wert
var txtValue += "<a href=\"#\" style=\"text-decoration:none;\"
onclick=\"table2_lazyLoadTable(103,'CUSTOMERNUMBER') ;
openDashletModalPanel(\"'table2'\"); \">\" + txtKpiValue + \"</a>\"; // Link für die KPI-
Wert
dashlet.addOption("kpi.box.i[0].value",txtValue); // KPI-Wert Option für dashlet
```

KPI Title



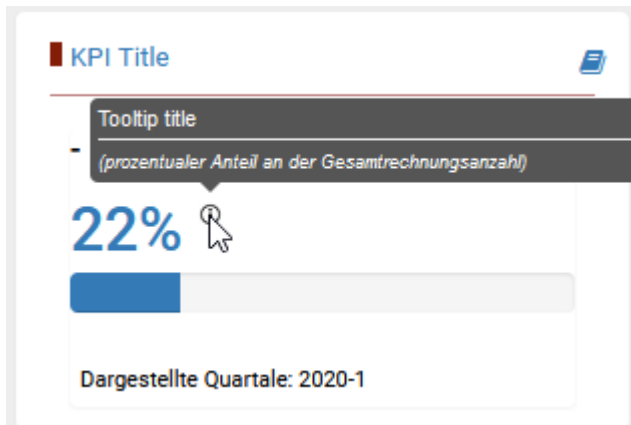
Wie kann ein Link für die KPI-Wert für Percentage-Box gesetzt werden?

Ein Link für die KPI-Wert kann mit die folgende Option gesetzt werden.

```
var txtKpiValue = 22; // KPI-Wert
var txtValue += "<a href='#\" style=\"text-decoration:none;\"
onclick=\"table2_lazyLoadTable(103,'CUSTOMERNUMBER') ;
openDashletModalPanel(\"'table2'\"); \">\" + txtKpiValue + \"</a>\"; // Link für die KPI-
Wert

// Information to be displayed after the percent-box (with tooltip)
txtValue_percentbox = \"\";
txtValue_percentbox += \"<span class=\"tt_kpinr\">
\";
txtValue_percentbox += \"<img class=\"tticon_kpinr\" src=\"\" + rawToolTipIcon + \"\"
alt=\"embedded folder icon\"/> \";
txtValue_percentbox += \"<span class=\"tttext_kpinr\">\" + txtKpiToolTipTitle + \"
\";
txtValue_percentbox += \"<hr style=\"color:white;margin-top:4px;margin-bottom:4px\" />
\";
txtValue_percentbox += \"<span style=\"font-size:7pt;font-style:italic\">\" +
txtKpiToolTipSubtitle + \"</span> \";
txtValue_percentbox += \"</span>
\";
txtValue_percentbox += \"</span> \";

dashlet.addOption(\"kpi.box.i[0].value\",txtValue); // KPI-Wert Option für dashlet
dashlet.addOption(\"kpi.box.i[0].value.style\",\"cursor: pointer;color: #337AB7;\"); ; //
set the style of the KPI Value for percentbox
dashlet.addOption(\"kpi.box.i[0].value.onclick\",\"table2_lazyLoadTable() ;
openDashletModalPanel(\"'table2'\");\"); // set the click functionality of the KPI Value
for percentbox
dashlet.addOption(\"kpi.box.i[0].value.html.after\",txtValue_percentbox); //set the
contents to display after the KPI Value for percentbox
```

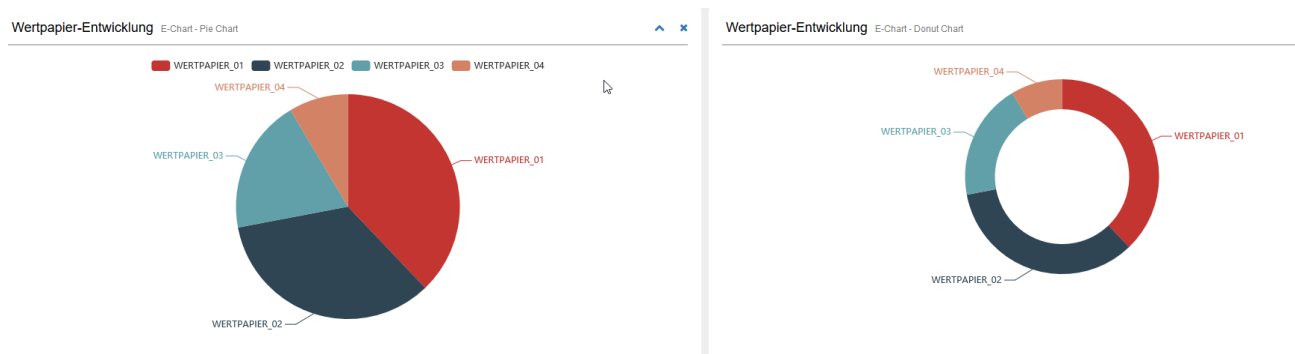



4.5.21 Kuchen- und Donutdiagramm (Pie Chart)

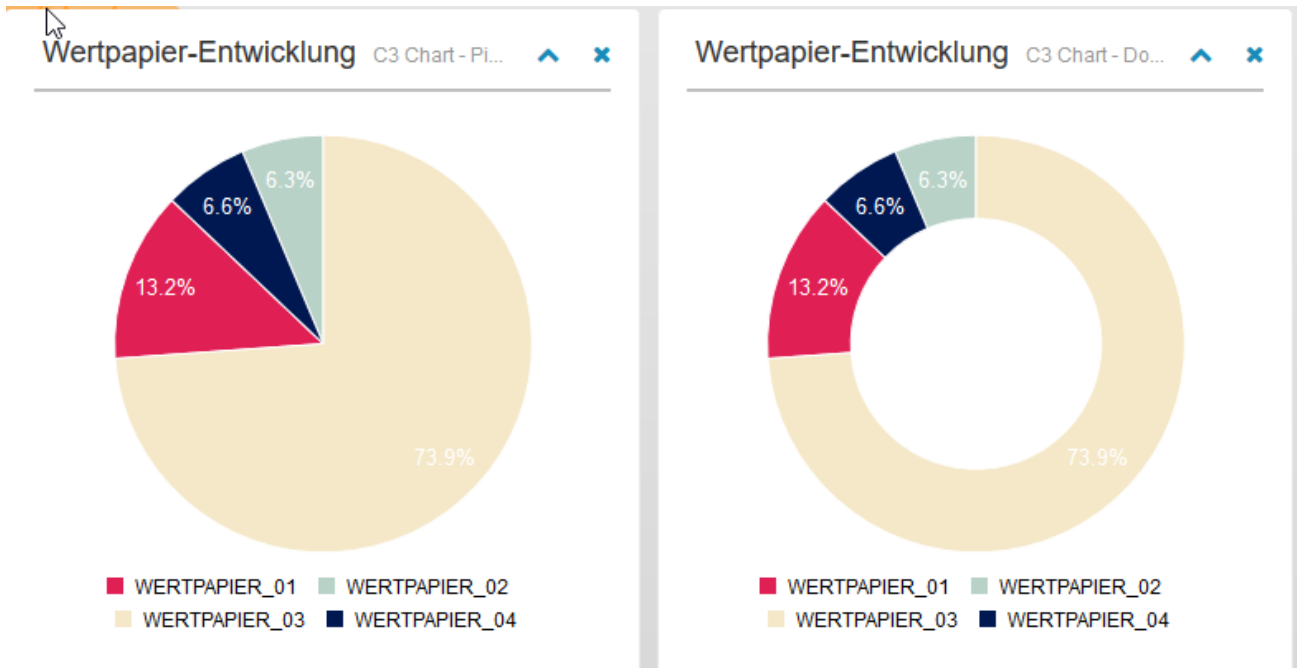
Einführung

Das Kuchen- und Donutdiagramm Dashlet könnte mit das C3Charts oder das eCharts platform entwickelt werden.

ECharts Beispiel



C3 Charts Beispiel



Dashlets erzeugen

Echart Chart

Hinweis: Die Spalten-Namen sollte angepasst werden.

```
mbr.getColumnDefinition("SERIES").setDisplayNames("Series");
mbr.getColumnDefinition("ISSUE_CONCAT").setDisplayNames("Pie-Name");
mbr.getColumnDefinition("COUNT_TYPE").setDisplayNames("Pie-Value");
```

Die Kuchen- und Donutdiagramm Dashlet ist mit die folgende Code Block erstellt.

EChart Dashlet erzeugen

```
// DashletManager erzeugen
ExpertsDashlet.createDashletManager(reportContext, "DashletManager");

// Pie Dashlet erzeugen
var dashlet = DashletManager.addDashletEChart("pie", "Wertpapier-Entwicklung", "C3
Chart-Pie Chart", true, mbr);
```

C3 Chart

Die Kuchen- und Donutdiagramm Dashlet ist mit die folgende Code Block erstellt.

C3 Dashlet erzeugen

```
// DashletManager erzeugen
ExpertsDashlet.createDashletManager(reportContext, "DashletManager");

// Pie Dashlet erzeugen
var dashlet = DashletManager.addDashletC3Chart("pie", "Wertpapier-Entwicklung", "C3
Chart-Pie Chart", true, mbr);

// Donut Dashlet erzeugen
var dashlet = DashletManager.addDashletC3Chart("donut", "Wertpapier-Entwicklung", "C3
Chart-Donut Chart", true, mbr);
```

Dashlet Optionen

Durch das Hinzufügen von Optionen können Dashlets individuell angepasst werden.

Dashlet Optionen

```
//Option für Donut chart in ECharts
dashlet.addOption("chart.series.radius", "['45%', '65%']");
// Option für position des PieCharts in ECharts
dashlet.addOption("chart.series.center", "['50%', '40%']");
```

Verscheide Anzeige Format sind möglich für diese Dashlet (siehe [ECharts \(4\) \[Options\]](#)).

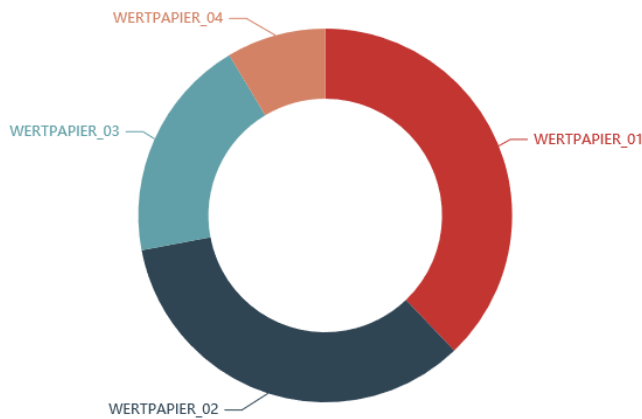
Für mehr Information für C3 Charts, siehe [C3-Charts](#).

FAQ

Wie kann eine Donut Chart erstellt werden?

Der Donut-Chart kann mit die folgende Option erstellt werden.

```
dashlet.addOption("chart.series.radius", "['50%', '80%']");
```



Wie kann der Center (Mitte-Punkt) des Pie-Charts wechselt werden?

Der Center eines Pie-Charts kann mit die folgende Option wechselt werden.

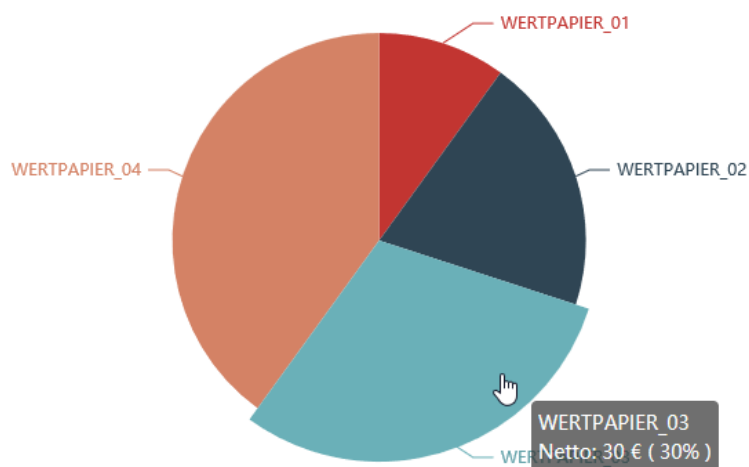
```
dashlet.addOption("chart.series.center", "['20%', '50%']");
```

Wie kann der Tooltip eines Pie-Charts formatiert werden?

Der Tooltip eines Pie-Charts kann mit die folgende Option formatiert werden.

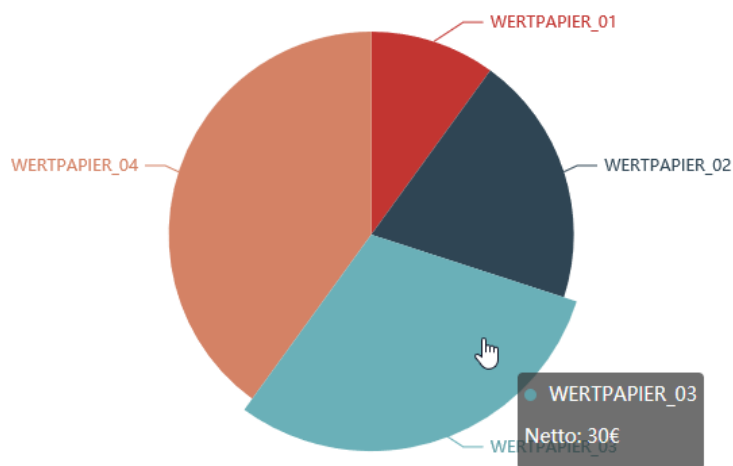
```
// tooltip formatter
dashlet.addOption("chart.tooltip.show" , "true");
dashlet.addOption("chart.tooltip.formatter", "{b} <br> Netto: {c} € ( {d}% )");

/*
Pie charts, gauge charts, funnel charts: {a} for series name, {b} for data item name,
{c} for data value, {d} for percentage.
*/
```



Oder andere Beispiel wobei die Farbe des Pie-Component ist auch angezeigt:

```
dashlet.addOption("chart.tooltip.formatter.object", "function (params) {\n" +
    "    //console.log(params);\n" +
    "    var colorSpan = color => '<span\n" +
style=\"display:inline-block;margin-right:5px;border-radius:10px;width:9px;height:\n" +
9px;background-color:' + color + '\"></span>';\n" +
    "    let rez = '<p>' + colorSpan(params.color) + ' '\n" +
+ params.name + '</p>';\n" +
    "    var xx = '<p> Netto: ' + Math.round(params.value\n" +
*100) /100 + '€' + '</p>'\n" +
    "    rez += xx;\n" +
    "    return rez;\n" +
    " } ");
```

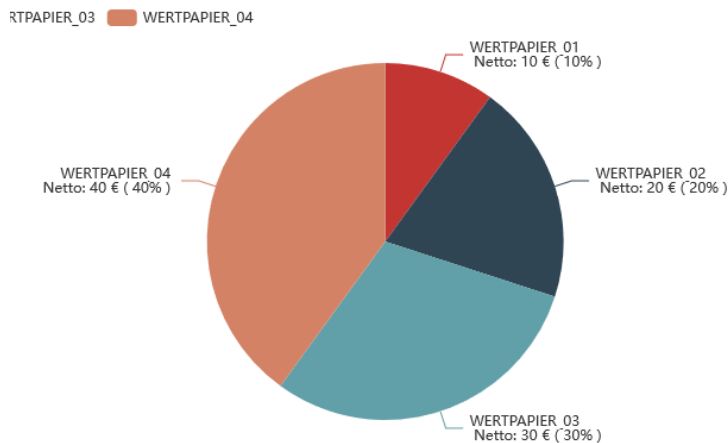


Wie kann Labels für ein PieChart dargestellt werden?

Der Pie-Chart Label kann mit die folgende Optionen dargestellt werden.

```
dashlet.addOption("chart.series.label.show", "true");
dashlet.addOption("chart.series.label.formatter", "{b} \\\n Netto: {c} € ( {d}% ) ");

// Falls der Label sollte nur für bestimmte PieChart dargestellt werden, dann mit der
// Series-Name benutzen (zb. Pie1)
dashlet.addOption("chart.series.n[Pie1].label.show", "true");
dashlet.addOption("chart.series.n[Pie1].label.formatter", "{b} \\\n Netto: {c} € ( {d}% ) ");
```

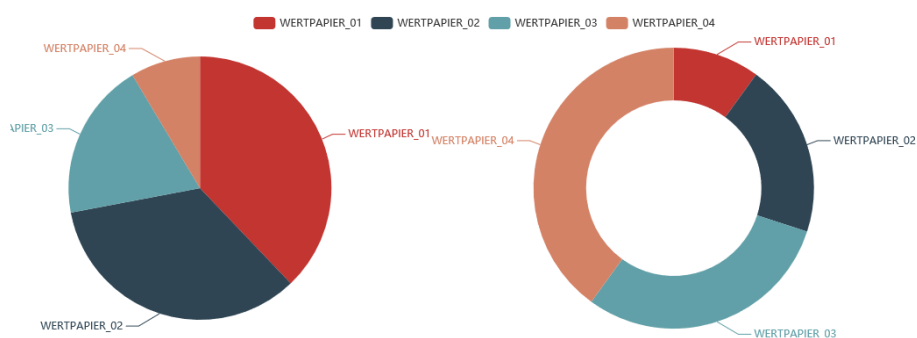


Wie kann zwei PieCharts in ein Dashlet erstellt werden?

Zwei PieCharts könnte in ein Dashlet erstellt werden wenn in der Spalte 'Series' zwei Werte enthalten.

KATEGORIE05_ID	SeriesFilter	Series	Pie-Name	Pie-Value
integer	string	string	string	string
8	Filter1	Pie1	WERTPAPIER_01	1823
2	Filter1	Pie1	WERTPAPIER_02	1644
4	Filter1	Pie1	WERTPAPIER_03	934
6	Filter1	Pie1	WERTPAPIER_04	413
9	Filter1	Pie2	WERTPAPIER_01	10
10	Filter1	Pie2	WERTPAPIER_02	20
11	Filter1	Pie2	WERTPAPIER_03	30
12	Filter1	Pie2	WERTPAPIER_04	40

Wertpapier-Entwicklung E-Chart - Pie Chart

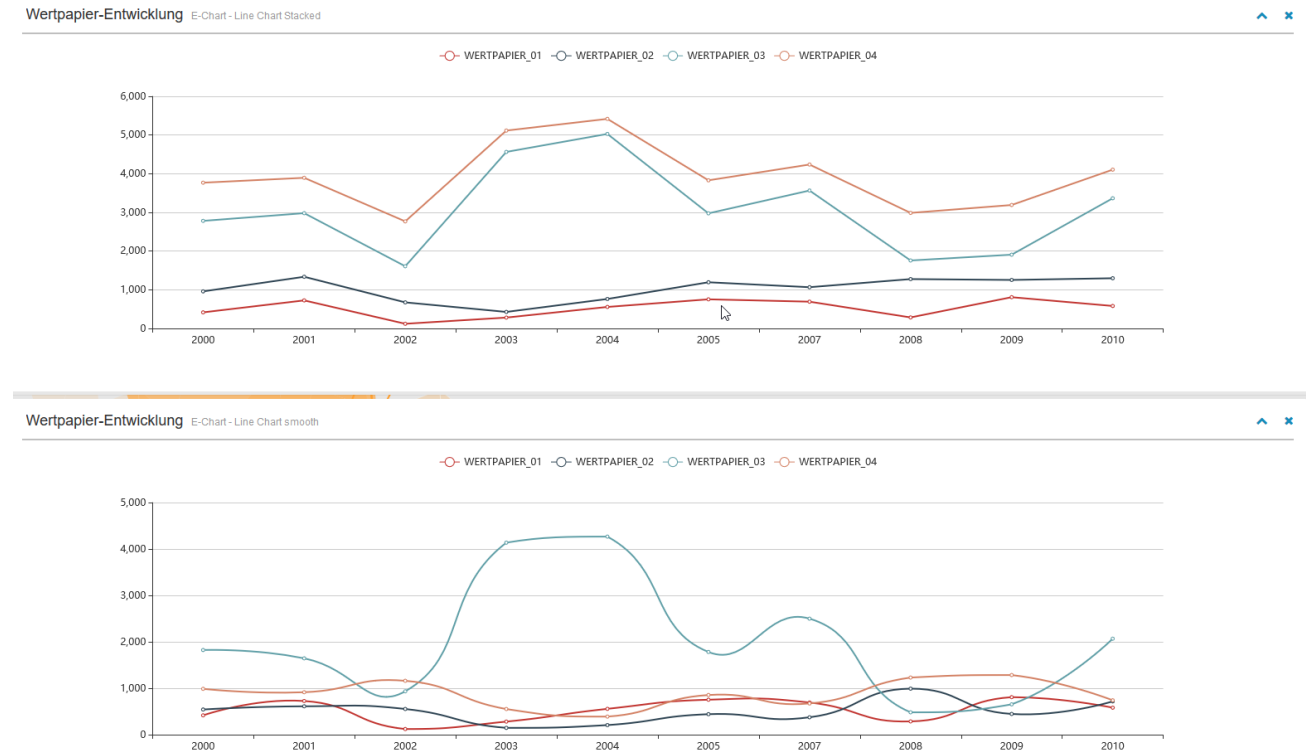


4.5.22 Linien-/Kurvendiagramm

Einführung

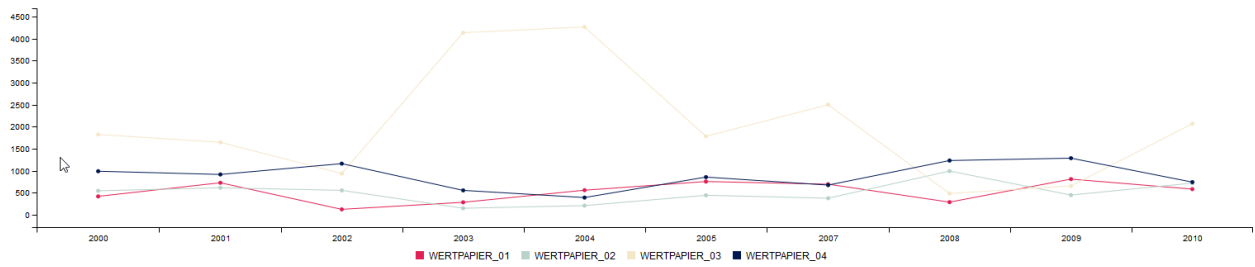
Das Linien-/Kurvendiagramm Dashlet könnte mit das C3Charts oder das eCharts platform entwickelt werden.

ECharts Beispiel

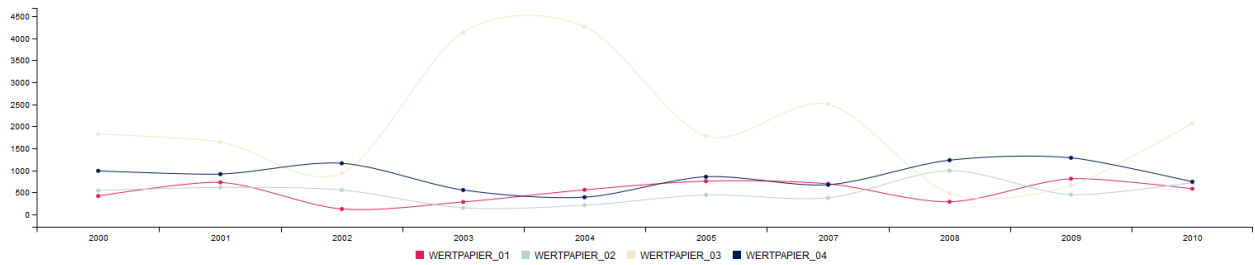


C3 Charts Beispiel

Wertpapier-Entwicklung C3 Chart - Line Chart



Wertpapier-Entwicklung C3 Chart - Spine Chart



Dashlets erzeugen

Echart Chart

Die Linien-/Kurvendiagramm Dashlet ist mit die folgende Code Block erstellt.

EChart Dashlet erzeugen

```
// DashletManager erzeugen
ExpertsDashlet.createDashletManager(reportContext, "DashletManager");

// Series Spalte definieren
mbr.getColumnDefinition("ORDERYEAR").setDisplayNames("Series");

// Dashlet erzeugen
var dashlet = DashletManager.addDashletEChart("line", "Wertpapier-Entwicklung", "C3
Chart-Line Chart", true, mbr);
```

C3 Chart

Die Linien-/Kurvendiagramm Dashlet ist mit die folgende Code Block erstellt.

EChart Dashlet erzeugen

```
// DashletManager erzeugen
ExpertsDashlet.createDashletManager(reportContext, "DashletManager");

// Series Spalte definieren
mbr.getColumnDefinition("ORDERYEAR").setDisplayNames("Series");

// Dashlet erzeugen
var dashlet = DashletManager.addDashletC3Chart("line", "Wertpapier-Entwicklung", "C3
Chart-Line Chart", true, mbr);
```

Dashlet Optionen

Durch das Hinzufügen von Optionen können Dashlets individuell angepasst werden.

Dashlet Optionen

```
//Option für Kurvendiagramm
dashlet.addOption("chart.series.smooth", "true");
// Ermöglicht es, alle Komponenten wie in einem "Stacked-chart" zu stapeln.
dashlet.addOption("chart.stacked", "all");
//Zeigt die Legende an
dashlet.addOption("chart.legend.show", "true");
```

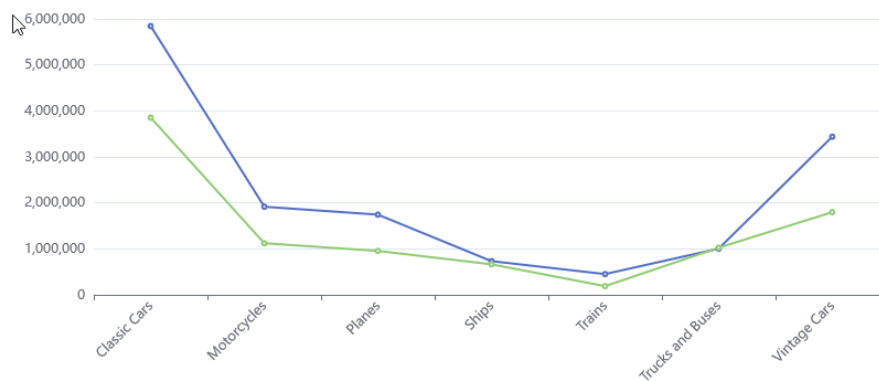
FAQ

Wie kann der X-Axis Label angepasst werden?

Der X-Axis Label kann mit die folgende Optionen angepasst werden.

```
dashlet.addOption("chart.xAxis.axisLabel.rotate", "45"); // x-Axis Label
anpassen
dashlet.addOption("chart.grid.height", "250px"); // Höhe des Grids anpassen
dashlet.addOption("chart.grid.bottom", "50px"); // Abstand zwischen des
Grids und der Box-unten
```

Umsatz-Entwicklung EChart - Line Chart



Verschiedene Anzeige Formate sind möglich für diese Dashlet (siehe [ECharts \(4\) \[Options\]](#)).

Für mehr Information für C3 Charts, siehe [C3-Charts](#).

4.5.23 Messzeiger (Gauge) Diagramm

Einführung

Das Gauge Diagramm Dashlet könnte mit das C3Charts oder das eCharts platform entwickelt werden.

Es ist eine perfekte Visualisierung von Leistungsindikatoren und ist genutzt für Management und andere Metriken.

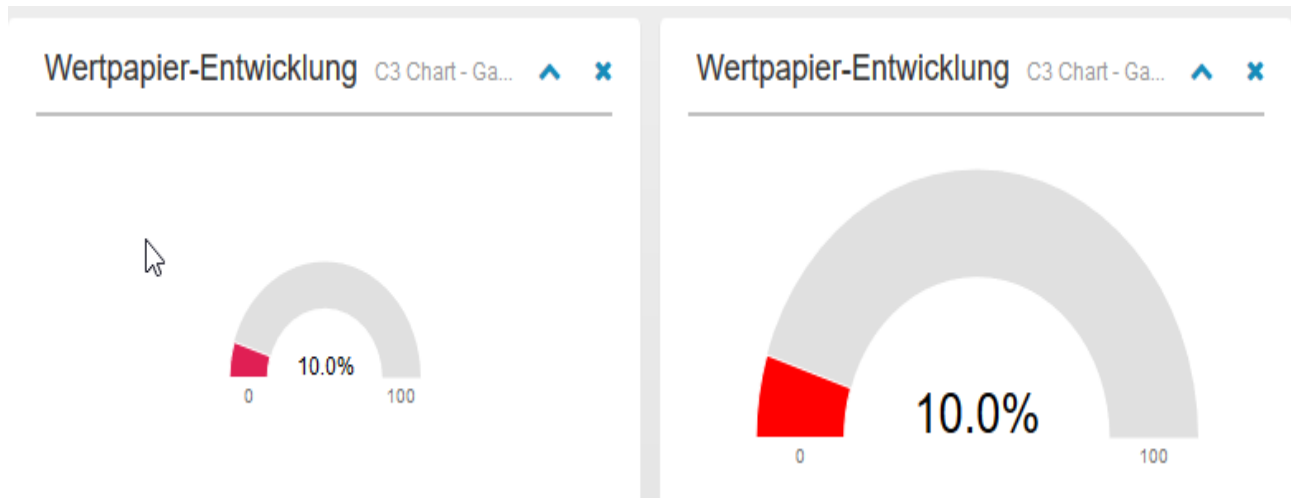
ECharts Beispiel

Wertpapier-Entwicklung E-Chart - Gauge Chart

2011 2012 2013 2014



C3 Charts Beispiel



Dashlets erzeugen

Echart Chart

Die Gauge Diagramm Dashlet ist mit die folgende Code Block erstellt.

Echart Dashlet erzeugen

```
// DashletManager erzeugen
ExpertsDashlet.createDashletManager(reportContext, "DashletManager");

// Dashlet erzeugen
var dashlet = DashletManager.addDashletEChart("gauge", "Wertpapier-Entwicklung", "C3
Chart-Gauge Chart", true, mbr);
```

C3 Chart

Die Gauge Diagramm Dashlet ist mit die folgende Code Block erstellt.

C3 Dashlet erzeugen

```
// DashletManager erzeugen
ExpertsDashlet.createDashletManager(reportContext, "DashletManager");

// Dashlet erzeugen
var dashlet = DashletManager.addDashletEChart("gauge", "Wertpapier-Entwicklung", "C3
Chart-Gauge Chart", true, mbr);
```

Dashlet Optionen

Durch das Hinzufügen von Optionen können Dashlets individuell angepasst werden.

Dashlet Option

```
//Option für Max und min Wert
```

```
dashlet.addOption("chart.gauge.min", "5");  
dashlet.addOption("chart.gauge.max", "205");
```

Verschiedene Anzeige Formate sind möglich für diese Dashlet (siehe [ECharts \(4\) \[Options\]](#)).

Für mehr Information für C3 Charts, siehe [C3-Charts](#).

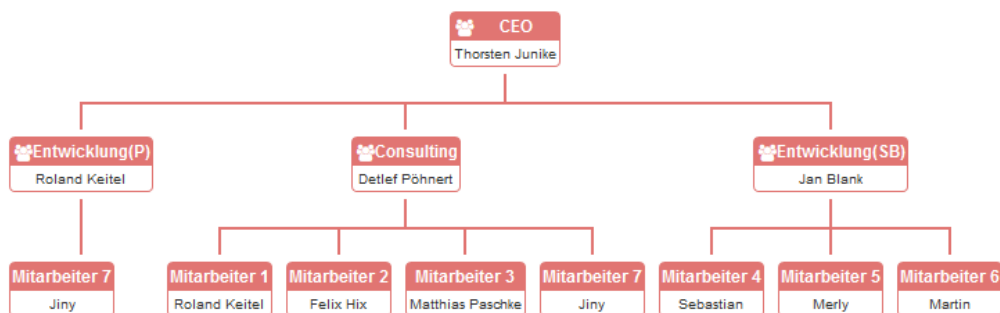
4.5.24 Organisational Chart

Einführung

Das OrgChart ermöglicht die Möglichkeit ein Organisational Chart darzustellen.

Mit der 'Export' button kann das OrgChart als Bild gespeichert werden.

Dashlet Org Chart

[Export](#)

Dashlets erzeugen

Die OrgChart Dashlet ist mit die folgende Code Block erstellt.

EChart Dashlet erzeugen

```
// DashletManager erzeugen
ExpertsDashlet.createDashletManager(reportContext, "DashletManager");

// memoryBuffer vorbereitern (Namen des Columns ändern)
mbr.getColumnDefinition("Id").setDisplayNames("org-id");
mbr.getColumnDefinition("Parent").setDisplayNames("org-parent");
mbr.getColumnDefinition("Name").setDisplayNames("org-name");
mbr.getColumnDefinition("Title").setDisplayNames("org-title")

// Dashlet erzeugen
var dashlet = DashletManager.addDashlet("DashletOrgChart", "Dashlet Org Chart", "",
true, mbr);
```

Dashlet Optionen

Durch das Hinzufügen von Optionen können Dashlets individuell angepasst werden.

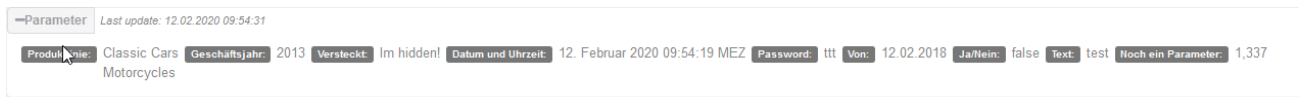
Dashlet Optionen

```
//OrgCharts Optionen
dashlet.addOption("orgchart.all.options", "//nodeContent: \"title\\\",\\n\" +
    \"    verticalLevel: 3,\\n\" +
    \"    visibleLevel: 3,\\n\" +
    \"    //visibleLevel: 999,\\n\" +
    \"    //exportButton: true,\\n\" +
    \"    //exportFilename: 'OrgChart',\\n\" +
    \"    // exportFileextension: 'png',\\n\" +
    \"    pan: true,\\n\" +
    \"    zoom: true,\\n\" +
    \"    zoominLimit:7,\\n\" +
    \"    zoomoutLimit:0.5,\\n\" +
    \"    //direction:'t2b',\\n\" +
    \"    toggleSiblingsResp:true,\\n\" +
    \"    //parentNodeSymbol:'fa-users',\");
```

4.5.25 Parameter Dashlet

Einführung

Das Parameter Dashlet zeigt die selektierten Parameter an, oben an dem Dashboard.



Dashlets erzeugen

Die Parameter Dashlet ist mit die folgende Code Block erstellt.

Create Dashlet

```
// DashletManager erzeugen
ExpertsDashlet.createDashletManager(reportContext, "DashletManager");

// mbrParam erzeugen
ExpertsDesign.createParameterBuffer(reportContext, true, false, false, false);
if (LogInfo.isRunTask()) {
    mbrParam.setColumnVisibility(true, 0);
    mbrParam.getColumnDefinition("Object").setVisible(false);
}

// Parameter Dashlet erzeugen
var paramDashlet = DashletManager.addDashlet("DashletParameter", "", "", false,
mbrParam);
```

Dashlet Optionen

Durch das Hinzufügen von Optionen können Dashlets individuell angepasst werden.

Dashlet Optionen

```
//Zeigt an, ob die Parameter immer angezeigt werden sollen oder über einen Knopf
zusammenklappbar sein sollen.
paramDashlet.addOption("parameter.toggleVisibility", "true");

//Steht der toggleVisibility-Parameter auf true, so kann der Zustand des
Knopfes zum Zusammenklappen durch diesen Parameter definiert werden.
paramDashlet.addOption("parameter.toggleVisibility.defaultState", "collapsed");

//Zeigt an, ob verborgene Parameter angezeigt werden sollen oder nicht. Zu
Debugzwecken verwendet.
paramDashlet.addOption("parameter.showHiddenParameters", "true");

//Spezifiziert den Separator der verwendet werden sollte, wenn der Parameter über
Mehrfachauswahl verfügt.
paramDashlet.addOption("parameter.multiValueSeparator", "linebreak");

//Spezifiziert, ob der 'HelpText', der für den Parameter definiert wurde, im
Parameter-Dashlet angezeigt werden soll oder nicht.
paramDashlet.addOption("parameter.helpTextAsTooltip", "true");

//Spezifiziert, ob die Parameter(s), die ein userproperty displayType als
"TextOutput" definiert ist, im Parameter-Dashlet angezeigt werden soll oder nicht.
paramDashlet.addOption("parameter.showTextOutputParameters", "true");
```

Verschiedene Anzeige-Formate sind möglich für diese Dashlet (siehe [Dashlet Engine Special Dashlet](#)).

Beispiel Report

Ein Beispiel für die Parameter Dashlet ist in die folgende Datei verfügbar.

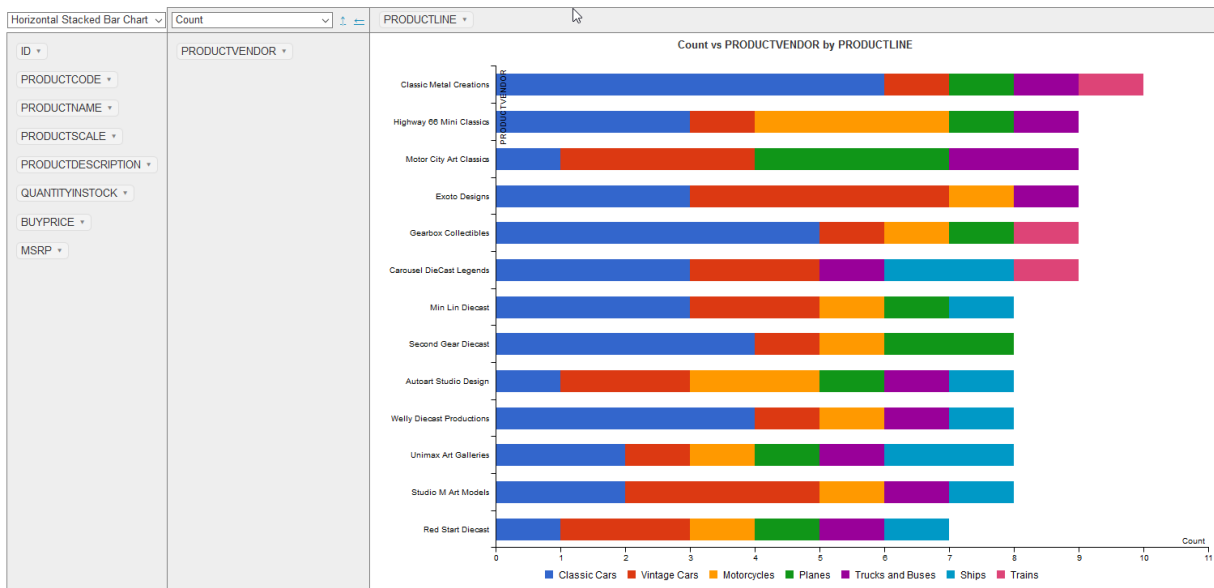
[parameterDashlet_Example.dashboard.rptdesign](#)

4.5.26 PivotTable

Einführung

Die PivotTabelle ist eine Open-Source-Implementierung der Javascript mit Drag & Drop-Funktion.

Es bietet die Möglichkeit, die x-Achse / y-Achse als Self-Service zu definieren. Es bietet auch die Funktionalität, Filter und Sortierreihenfolgen festzulegen sowie mehrere Dimensionen zu verwenden. Außerdem kann der Benutzer zwischen verschiedenen Visualisierungen wechseln. Die Hauptanforderungen sind, dass eine JSON-orientierte Datenquelle erforderlich ist.



Heatmap

Count

PRODUCTLINE

	PRODUCTLINE	Classic Cars	Vintage Cars	Motorcycles	Planes	Trucks and Buses	Ships	Trains	Totals
PRODUCTVENDOR									
Classic Metal Creations		6	1	0	1	1	0	1	10
Highway 66 Mini Classics		3	1	3	1	0	1	0	9
Motor City Art Classics		1	3	0	3	2	0	0	9
Exoto Designs		3	4	0	1	0	0	0	9
Gearbox Collectibles		5	1	1	1	0	0	1	9
Carousel DieCast Legends		3	2	0	0	1	2	1	9
Min Lin Diecast		3	2	1	1	0	1	0	8
Second Gear Diecast		4	1	1	2	0	0	0	8
Autoart Studio Design		1	2	2	1	1	1	0	8
Welly Diecast Productions		4	1	1	0	1	1	0	8
Unimax Art Galleries		2	1	1	1	0	1	2	8
Studio M Art Models		2	3	1	0	0	1	1	8
Red Start Diecast		1	2	1	1	0	1	0	7
Totals		38	24	13	12	11	9	3	110

Dashlets erzeugen

Die Pivot Dashlet ist mit die folgende Code Block erstellt.

Create Dashlet

```
// DashletManager erzeugen
ExpertsDashlet.createDashletManager(reportContext, "DashletManager");

// Pivot Dashlet erzeugen
var dashlet = DashletManager.addDashlet("DashletPivot", "", "", true, mbrPivotData);
```


Dashlet Optionen

Durch das Hinzufügen von Optionen können Dashlets individuell angepasst werden.

Dashlet Optionen

```
// Id setzen
dashlet.setId("pivot1");

// Tile und subtitle setzen
dashlet.addOption("title", "Pivot Title");
dashlet.addOption("subtitle", "Sub");

//Display typ setzen
dashlet.addOption("pivot.display.type", "pivotUI");

// Individuel Optionen setzen
dashlet.addOption("pivot.rows" , "[\"PRODUCTVENDOR\"]");
dashlet.addOption("pivot.cols" , "[\"PRODUCTLINE\"]");
dashlet.addOption("pivot.rendererName" , "\"Horizontal Stacked Bar Chart\"");
```

Verscheide Anzeige Format sind möglich für diese Dashlet (siehe [PivotTable General](#)).

FAQ

Wie kann den Anzeigetyp (Display-Typ) der Pivot-Tabelle eingestellt werden?

Der Anzeigetyp (display type) kann mit folgender Option eingestellt werden:

```
dashlet.addOption("pivot.display.type", "pivotUI");
```

Der Standardwert ist 'Pivot'.

Wie kann alle Pivot-Optionen als einen einzigen Optionsblock eingefügt werden?

Die einzelnen Optionen können auch mit der Dashlet-Option 'pivot.all.options' als einzelner Optionsblock gerendert werden, wie im folgenden Beispiel gezeigt.

```
// Pivot Optionen (wie renderName, rols, cols, rendereOptions) setzen
var options =
"  renderers: renderers,\n"
"  rows: [\"PRODUCTVENDOR\"], \n"
"  cols: [\"PRODUCTLINE\"], \n"
"  rendererName: \"Horizontal Stacked Bar Chart\", \n"
"  ";
dashlet.addOption("pivot.all.options", options);
```

Wie werden die Details der Pivot-Zelle in einer Tabelle angezeigt?

Es besteht die Möglichkeit, die Details der Pivot-Tabelle in einer externen Tabelle anzuzeigen. Die folgenden Schritte sollten dazu befolgt werden.

Schritt 1: Deklarieren eine Rückruffunktion im Pivot Dashlet

```
//-----Table click Option-----
dashlet.addOption("pivot.rendererOptions" , " {\n" +
    "         table: { \n" +
    "             clickCallback: function(e,
value, filters, pivotData){ \n" +
    "                 tmpData = [];
clickedPivotFilters = filters; \n" +
    "                 //console.log(filters); \n"
    +
    "                 pivotData.forEachMatchingRecord(filters, \n" +
    "                 function(record)
{ tmpData.push(record); }); \n" +
    "                 \n" +
    "                 console.log(tmpData); \n" +
    "                 table.clear(); \n"+
    "                 table.rows.add(tmpData); \n"+
    "
    table.columns.adjust().draw(); \n"+
    "             } \n" +
    "         } \n" +
    "     }");
```

Schritt 2: Erstellen ein HTML-Dashlet für die Tabelle.

Fügen die Rückruffunktionen "tableClickCallback ()" ein, um die Details in der Tabelle anzuzeigen.

Pivot Details in eine Tabelle anzeigen

```
// Ein HTML Tabelle erstellen
var dashletTable = DashletManager.addDashlet("DashletHTML", "", "", false, null);
dashletTable.setResponsiveGrid("12-12-12");
dashletTable.setId("Dash2");

// Table creation
var customHTML = "<table id=\"Dash2\" class=\"table table-striped table-bordered stripe compact row-borders\" width=\"100%\"></table>";
dashletTable.addOption("html", customHTML);

// Javascript for displaying the details
var customJS = "";
customJS =
    "$(document).ready(function(){ \n"+
    "Dash2_loadDataTable(); \n"+
    "}); \n"+
    "function Dash2_loadDataTable(){ \n"+
    " \n"+
    //d3.keys(pivot1_json['data'][0]); // the json is saved as key-value pairs,
    default is 'data'
    "var keys = d3.keys(pivot1_json[Object.keys(pivot1_json)[0]][0]); // Annahme:
    alle Objekte sind der sleben Natur/haben die selben Spalten \n"+
    "var columns = []; \n"+
    "//console.log(keys); \n"+
    "for(var i in keys) \n"+
    "{ \n"+
    "    var obj = new Object(); \n"+
    "    obj.mData = keys[i]; \n"+
    "    obj.title = keys[i]; \n"+
    "    \n"+
    "    \n"+
    "    columns.push(obj); \n"+
    "} \n"+
    " \n"+
    "//console.log(columns); \n"+
    " \n"+
    "//var data = $.map(mjson, function(el){return el}); \n"+
    " \n"+
    "var exportTitle = 'Testing_Dashboard_Data_' + $('#dashletbox-panel-pivot1 div h2
    small').text(); \n"+
    " \n"+
    "table = $('#Dash2').DataTable({responsive:true,dm:'Bfrrtip',buttons:
    [{extend:'csv', title:exportTitle},{extend:'excel',title:exportTitle, \n"+
    "customize: function ( xlsx ) { \n"+
    "    var sheet = xlsx.xl.worksheets['sheet1.xml']; \n"+
    "    \n"+
    "    / * Modify first row * / \n"+
    "    $('c[r=A1] t', sheet).text( 'Dimensionen:' +
    JSON.stringify(clickedPivotFilters) ); \n"+
```

```

"    $('row:first c', sheet).attr( 's', '22'); \n"+
"} \n"+
" \n"+
"}], 'aaData': null, 'aoColumns': columns \n"+
"}); \n"+
"}; \n";

/*Function to be declared in pivot, if the table clickcallback functionality is to be
used , ie: display the pivot details in a table*/
customJS += "function tableClickCallback() { \n "
              + "//alert('Reached
tableClickCallback!'); \n "
              + "try { \n "
              + "return(function(e, value, filters,
pivotData){ \n"+
              "                                tmpData = [];
clickedPivotFilters = filters; \n"+
              "                                //console.log(filters); \n"+
              "                                "
pivotData.forEachMatchingRecord(filters, \n"+
              "                                function(record)
{ tmpData.push(record); }); \n"+
              "                                \n"+
              "                                //console.log(tmpData); \n"+
              "                                \n"+
              "                                table.clear(); \n"+
              "                                table.rows.add(tmpData); \n"+
              "                                table.columns.adjust().draw();
\n"+
              "                                }) \n"
              + "}catch(err) { \n"
              + "console.log(err.message);"
              + " } \n"

              + "} \n";

dashletTable.addOption("javascript", customJS);

```

Wie kann eine Pivot-Tabelle mit Schaltflächen (Buttons) erstellt werden?

Die Schaltflächen (Buttons) können erstellt werden, indem das gewünschte Feld als "Serienfilter" festgelegt wird.

```

// Die Spalte Country wird als Series-Filer definiert
dashlet.addOption("datatable.column.seriesfilter", "PRODUCTLINE");

```

Die Schaltflächen (Buttons) können dann automatisch mit dieser Option erstellt werden:

```

// Fügt eine Buttonbar für den Seriesfilter hinzu
dashlet.addOption("pivot.add.buttonbar", "seriesfilter");

```

Um die Schaltflächen (Buttons) manuell zu erstellen, kann die folgende Option verwendet werden.

```

DashletButtonBar buttonbar = (DashletButtonBar)
dashlet.createDashletBar(DashletBarType.BUTTON_BAR, DashletButtonBar.TOP_BAR);
BootstrapButton btn = buttonbar.addButton("France", true);
btn.addOption("button.onclick", "pivot1_draw(pivot1_json['France']);");
btn = buttonbar.addButton("USA", false);
btn.addOption("button.onclick", "pivot1_draw(pivot1_json['USA']);");
btn = buttonbar.addButton("Germany", false);
btn.addOption("button.onclick", "pivot1_draw(pivot1_json['Germany']);");

```

Pivot My First Pivot

MOTORCYCLES CLASSIC CARS TRUCKS AND BUSES VINTAGE CARS **PLANES** SHIPS TRAINS

Table Count PRODUCTNAME

ID	PRODUCTVENDOR	PRODUCTNAME	1900s Vintage Bi-Plane	1900s Vintage Tri-Plane	1928 British Royal Navy Airplane	1980s Black Hawk Helicopter	ATA: B757-300	America West Airlines B757-
PRODUCTCODE		PRODUCTVENDOR						
PRODUCTSCALE		Autoart Studio Design	1					
PRODUCTDESCRIPTION		Classic Metal Creations			1			
QUANTITYINSTOCK		Gearbox Collectibles						
BUYPRICE		Highway 66 Mini Classics					1	
MSRP		Min Lin Diecast						
		Motor City Art Classics						
		Red Start Diecast				1		
		Second Gear Diecast						
		Unimax Art Galleries		1				
		Totals	1	1	1	1	1	

CSV EXCEL

Search:

ID	PRODUCTCODE	PRODUCTNAME	PRODUCTSCALE	PRODUCTVENDOR	PRODUCTDESCRIPTION	QUANTITYINSTOCK	BUYPRICE	MSRP
60	S24_1785	1928 British Royal Navy Airplane	1:24	Classic Metal Creations	Official logos and insignias	3627	66.74	109.42

Showing 1 to 1 of 1 entries

PREVIOUS 1 NEXT

Wie kann Pivot-Daten aus einer externen JSON-Datei erhalten werden?

Die folgende Option sollte festgelegt werden, um die Pivot-Daten aus einer externen JSON-Datei zu laden.

```

dashlet.addOption("dashlet.external.json", "true");

// set the path for the external JSON files (only for servBIRD)
glJobID = 0;
var appContext = reportContext.getAppContext(); // HashMap
if (appContext != null && appContext.containsKey("REPORT_JOBID"))
{
    glJobID = appContext.get("REPORT_JOBID");
}
dashlet.addOption("pivot.json.path", "/servBIRD/streamOutputFile?jobid=" + glJobID + "&filename=d/");

// For local testing
//dashlet.addOption("pivot.json.path", "http://localhost/");

```

Wie kann den Titel und den Untertitel der Pivot-Tabelle festgelegt werden?

Der Titel und der Untertitel können mit den folgenden Optionen festgelegt werden:

```
dashlet.addOption("title","Pivot Simple");
dashlet.addOption("subtitle", "Pivot Subtitle");
```

Wie kann Heatmaps mit benutzerdefinierten Sortimenten und Farben erstellt werden?

Stellen die Option wie folgt ein:

```
//-----Heatmap colorScale Modification Option-----  
dashlet.addOption("pivot.rendererOptions" , " {\n" +  
    "  
        heatmap: { \n" +  
            "  
                colorScaleGenerator:  
  
function(values) {\n"+  
    "  
        // Plotly happens to come  
with d3 on board\n"+  
    "  
        return d3.scale.linear()\n"+  
    "  
            .domain([0, 60000,  
120000])\n"+  
    "  
            .range([\\"#77F\\",  
\\"#FFF\\", \\"#F77\\"])\n"+  
    "  
        } \n" +  
    "  
    }");
```

Wie werden Zwischensummen (Subtotals) in PivotTable angezeigt?

Setzen die Option "pivot.showSubTotal" auf "true". Stellen auch im Optionsblock die 'Datenklasse' wie folgt ein:

```
// option to indicate if the subtotals are to be displayed or not
dashlet.addOption("pivot.dataClass" , "dataClass");
dashlet.addOption("pivot.showSubTotal" , "true");
```

Pivot My First Pivot

Table With Subtotal	Count	PRODUCTLINE
ID	PRODUCTVENDOR	
PRODUCTCODE	PRODUCTSCALE	
PRODUCTNAME		
PRODUCTDESCRIPTION		
QUANTITYINSTOCK		
BUYPRICE		
MSRP		

	PRODUCTLINE	Classic Cars	Motorcycles	Planes	Ships	Trains	Trucks and Buses	Vintage Cars	Totals
▲PRODUCTVENDOR	PRODUCTSCALE								
▲Autoart Studio Design		1	2	1	1		1	2	8
	1:12	1							1
	1:18							1	1
	1:24		1	1			1	1	4
	1:50		1						1
	1:700				1				1
▲Carousei DieCast Legends		3			2	1	1	2	9
	1:18	1				1		1	4
	1:24	2			1			1	4
	1:700				1				1
	Totals	4	2	1	3	1	2	4	17

Wie werden die Details der Zwischensummenberechnungen (Subtotals Computations) in einer externen Tabelle angezeigt?

Es besteht die Möglichkeit, die Details der Zwischensummenberechnung in einer externen Tabelle anzuzeigen.

Die folgenden Schritte sollten dazu befolgt werden.

Schritt 1: Deklarieren eine Rückruffunktion im Pivot Dashlet

```
//-----Table click Option-----
dashlet.addOption("pivot.rendererOptions" , " {\n" +
    "         table: { \n" +
    "             clickCallback: function(e,
value, filters, pivotData){ \n" +
    "                 tmpData = [];
clickedPivotFilters = filters; \n" +
    "                 //console.log(filters); \n"
    +
    "                 function(record)
pivotData.forEachMatchingRecord(filters, \n" +
    "                 { tmpData.push(record); }); \n" +
    "                 \n" +
    "                 console.log(tmpData); \n" +
    "                 table.clear(); \n"+
    "                 table.rows.add(tmpData); \n"+
    "                 }, \n" +
    "                 eventHandlers: { \n"+
    "                     click: function(e, value,
filters, pivotData){ \n"+
    "                         tmpData = [];
clickedPivotFilters = filters; \n"+
    "                         pivotData.forEachMatchingRecord(filters, \n"+
    "                         { tmpData.push(record); }); \n"+
    "                         \n" +
    "                         //console.log(tmpData); \n"+
    "                         \n"+
    "                         table.clear(); \n"+
    "                         table.rows.add(tmpData); \n"+
    "                         } \n"+
    "                     } \n"+
    "                 }, \n"+
    "             }");

// option to indicate if the subtotals are to be displayed or not
dashlet.addOption("pivot.dataClass" , "dataClass");
dashlet.addOption("pivot.showSubTotal" , "true");
```

Schritt 2: Erstellen ein HTML-Dashlet für die Tabelle.

Fügen die Rückruffunktionen "tableClickCallback ()" ein, um die Details in der Tabelle für Tabellendiagramme im Pivot Dashlet anzuzeigen, sowie "eventHandlersClick ()", um die Details in der Tabelle für Zwischensummen (subtotals) anzuzeigen.

Pivot Details in eine Tabelle anzeigen

```
// Ein HTML Tabelle erstellen
var dashletTable = DashletManager.addDashlet("DashletHTML", "", "", false, null);
dashletTable.setResponsiveGrid("12-12-12");
dashletTable.setId("Dash2");

// Table creation
var customHTML = "<table id=\"Dash2\" class=\"table table-striped table-bordered stripe compact row-borders\" width=\"100%\"></table>";
dashletTable.addOption("html", customHTML);

// Javascript for displaying the details
var customJS = "";
customJS =
    "$(document).ready(function(){ \n"+
    "Dash2_loadDataTable(); \n"+
    "}); \n"+
    "function Dash2_loadDataTable(){ \n"+
    " \n"+
    //d3.keys(pivot1_json['data'][0]); // the json is saved as key-value pairs,
    default is 'data'
    "var keys = d3.keys(pivot1_json[Object.keys(pivot1_json)[0]][0]); // Annahme:
    alle Objekte sind der sleben Natur/haben die selben Spalten \n"+
    "var columns = []; \n"+
    "//console.log(keys); \n"+
    "for(var i in keys) \n"+
    "{ \n"+
    "    var obj = new Object(); \n"+
    "    obj.mData = keys[i]; \n"+
    "    obj.title = keys[i]; \n"+
    "    \n"+
    "    \n"+
    "    columns.push(obj); \n"+
    "} \n"+
    " \n"+
    "//console.log(columns); \n"+
    " \n"+
    "//var data = $.map(mpson, function(el){return el}); \n"+
    " \n"+
    "var exportTitle = 'Testing_Dashboard_Data_' + $('#dashletbox-panel-pivot1 div h2
    small').text(); \n"+
    " \n"+
    "table = $('#Dash2').DataTable({responsive:true,dm:'Bfrrtip',buttons:
    [{extend:'csv', title:exportTitle},{extend:'excel',title:exportTitle, \n"+
    "customize: function ( xlsx ) { \n"+
    "    var sheet = xlsx.xl.worksheets['sheet1.xml']; \n"+
    "    \n"+
    "    / * Modify first row * / \n"+
    "    $('c[r=A1] t', sheet).text( 'Dimensionen:' +
    JSON.stringify(clickedPivotFilters) ); \n"+
```



```

"    $('row:first c', sheet).attr( 's', '22'); \n"+
"} \n"+
" \n"+
"}], 'aaData': null, 'aoColumns': columns \n"+
"}); \n"+
"}; \n";

/*Function to be declared in pivot, if the table clickcallBack functionality is to be
used , ie: display the pivot details in a table*/
customJS += "function tableClickCallback() { \n "
            + "//alert('Reached
tableClickCallback!'); \n "
            + "try { \n "
            + "return(function(e, value, filters,
pivotData){ \n"+
            "                tmpData = [];
clickedPivotFilters = filters; \n"+
            "                //console.log(filters); \n"+
            "
pivotData.forEachMatchingRecord(filters, \n"+
            "                function(record)
{ tmpData.push(record); }); \n"+
            "                \n"+
            "                //console.log(tmpData); \n"+
            "                \n"+
            "                table.clear(); \n"+
            "                table.rows.add(tmpData); \n"+
            "                table.columns.adjust().draw();
\n"+
            "                }) \n"
            + "catch(err) { \n"
            + "console.log(err.message);"
            + " } \n"
            + "} \n";

/*Function to be declared in pivot, if the subtotal clickcallBack functionality is to
be used , ie: display the pivot subtotal details in a table*/
customJS += "function eventHandlersClick() { \n "
            + "//alert('Reached
eventHandlersClick!'); \n "
            + "try { \n "
            + "return(function(e, value, filters,
pivotData){ \n"+
            "                tmpData = [];
clickedPivotFilters = filters; \n"+
            "                //console.log(filters); \n"+
            "
pivotData.forEachMatchingRecord(filters, \n"+
            "                function(record)
{ tmpData.push(record); }); \n"+
            "                \n"+
            "                //console.log(tmpData); \n"+
            "                \n"+
            "                table.clear(); \n"+
            "                table.rows.add(tmpData); \n"+

```

```

        "                                table.columns.adjust().draw();
\n"+
        "                                }) \n"
        + "}catch(err) { \n"
        + "console.log(err.message);"
        + " } \n"
    + "} \n";

dashletTable.addOption("javascript", customJS);

```

Wie kann Javascript-Inhalte für das Pivot-Dashlet gerendert werden ?

Javascript-Anweisungen können für das Pivot-Dashlet mit den folgenden Optionen gerendert werden.

```

// Script in HTML renderen
dashlet.addOption("pivot.custom.script.before", "var test_before = 0;");
dashlet.addOption("pivot.custom.script.after", "var test_after = 100;");

```

Wenn die Spaltennamen in der Detailtabelle Sonderzeichen (z. B. Punkt) enthalten, wie kann sichergestellt werden, dass die Details korrekt angezeigt werden?

Wenn die Spaltennamen Sonderzeichen enthalten und beim Anzeigen der Details in der Tabelle Fehler auftreten, sollten die folgenden Codezeilen ersetzt werden.

```

// Ersetzung 1 an der DataTable beim Columns erzeugen:

var columns = [];
for(var i in keys)
{
    var obj = new Object();
    var mD = keys[i].replace(/\. /g, "");
    obj.mData = mD;
    obj.title = keys[i];
    columns.push(obj);
}

// Ersetzung 2 beim Filtern der Pivot-Daten:

pivotData.forEachMatchingRecord(filters,function(record){
    for(var i in record)
    {
        var ni = i.replace(/\. /g, "");
        record[ni] = record[i];
    }
    tmpData.push(record);
});

```

4.5.27 Sankey Diagramm

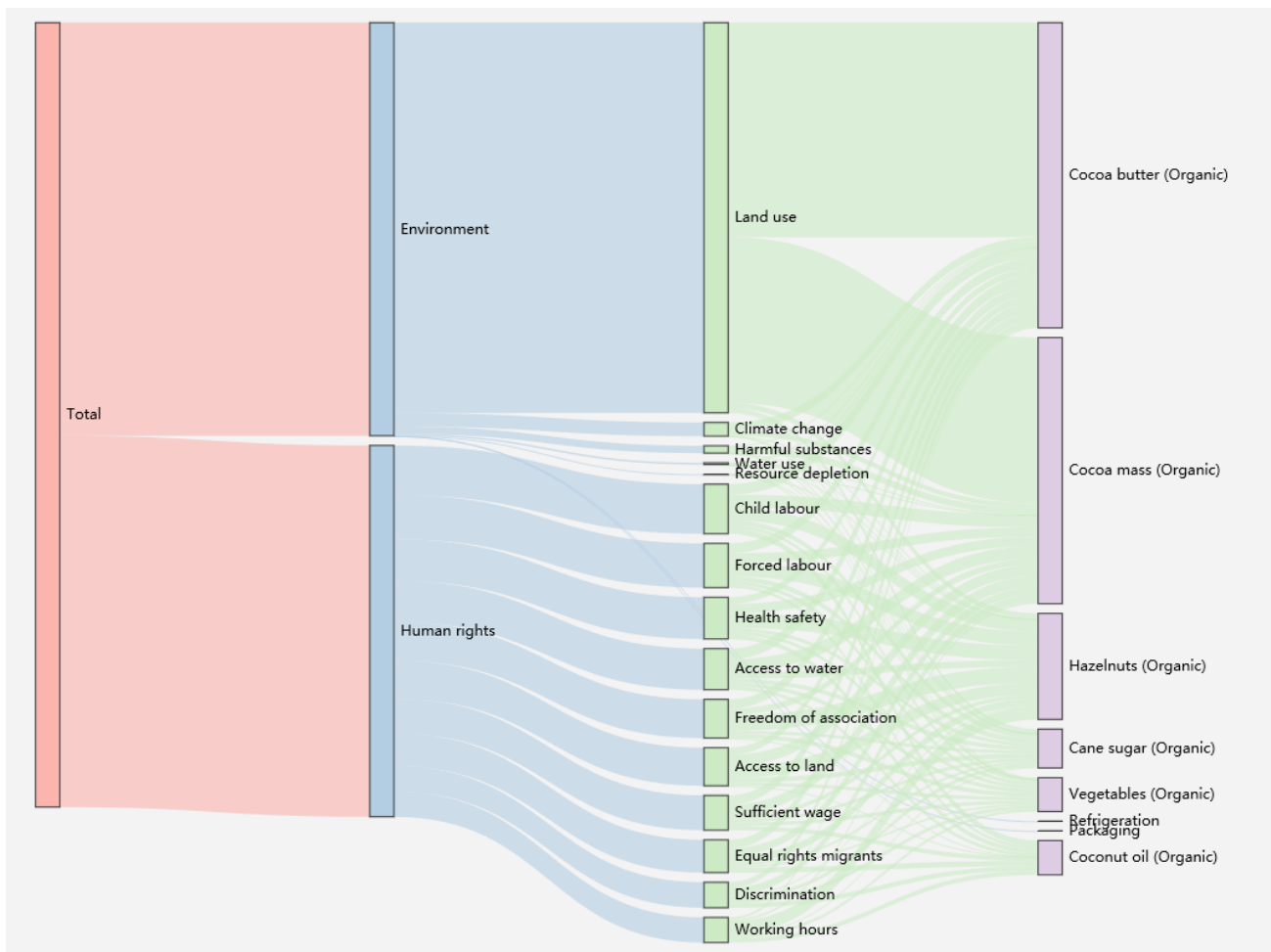


Information

Dieser Charttyp ist derzeit noch nicht im Dashlet-Plugin verfügbar. Bitte kontaktieren Sie unseren [Support](#), wenn Sie an der Nutzung dieser Darstellungsform interessiert sind.

Einführung

Sankey-Diagramm Das Sankey-Diagramm ist eine bestimmte Art von Streamgraph (kann auch als gerichteter azyklischer Graph angesehen werden), in dem die Breite jedes Zweigs proportional zur Durchflussmenge angezeigt wird. Diese Diagramme werden normalerweise verwendet, um Energie- oder Material- oder Kostentransfers zwischen Prozessen zu visualisieren.

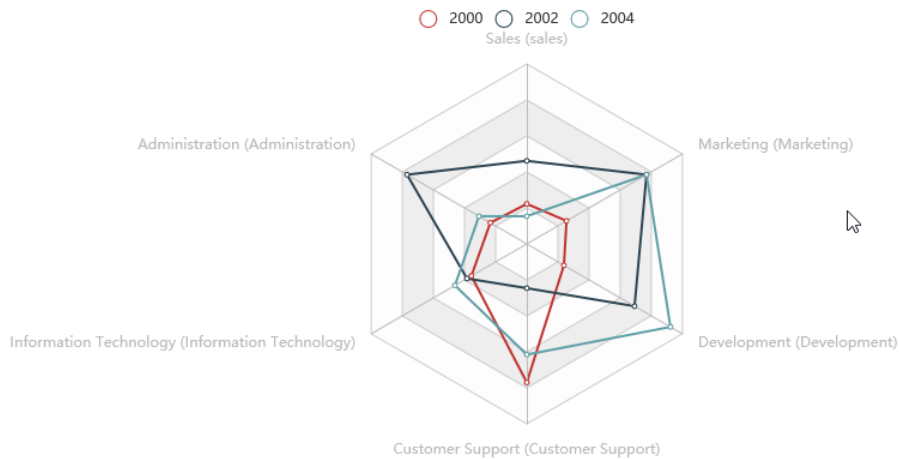


4.5.28 Spinnennetzdiagramm / Netzdiagramm (Radar chart)

Einführung

Das Spinnennetzdiagramm Dashlet könnte mit das eCharts platform entwickelt werden.

ECharts - Bar Standard



Dashlets erzeugen

Die Spinnennetzdiagramm Dashlet ist mit die folgende Code Block erstellt.

EChart Dashlet erzeugen

```
// DashletManager erzeugen
ExpertsDashlet.createDashletManager(reportContext, "DashletManager");

// memoryBuffer vorbereitern (Namen des Columns ändern)
mbr.getColumnDefinition("Departments").setDisplayNames("Indicator");
mbr.getColumnDefinition("Year").setDisplayNames("Series");
mbr.getColumnDefinition("Mandant").setDisplayNames("SeriesFilter");
mbr.getColumnDefinition("Count").setDisplayNames("Value");

// Dashlet erzeugen
var dashlet = DashletManager.addDashletEChart("radar", "Radar Chart", "", true, mbr);
```

Dashlet Optionen

Durch das Hinzufügen von Optionen können Dashlets individuell angepasst werden.

Dashlet Optionen

```
//indicator options
dashlet.addOption("chart.radar.indicator.object","\n" +
    "    { name: 'Sales (sales)', max: 65},\n" +
    "    { name: 'Administration (Administration)', max: 65}, \n" +
    "    { name: 'Information Technology (Information Technology)', max: 65},\n" +
    "    { name: 'Customer Support (Customer Support)', max: 65},\n" +
    "    { name: 'Development (Development)', max: 65},\n" +
    "    { name: 'Marketing (Marketing)', max: 65}\n" +
    "]);

// define the legend explicitly
dashlet.addOption("chart.legend.show", "true");
dashlet.addOption("chart.predefined.legend", "['2000', '2002', '2004']");
```

Verschiedene Anzeige-Formate sind möglich für diese Dashlet (siehe [ECharts \(4\) Radar](#)).

4.5.29 Streamgraph/Flussdiagramm

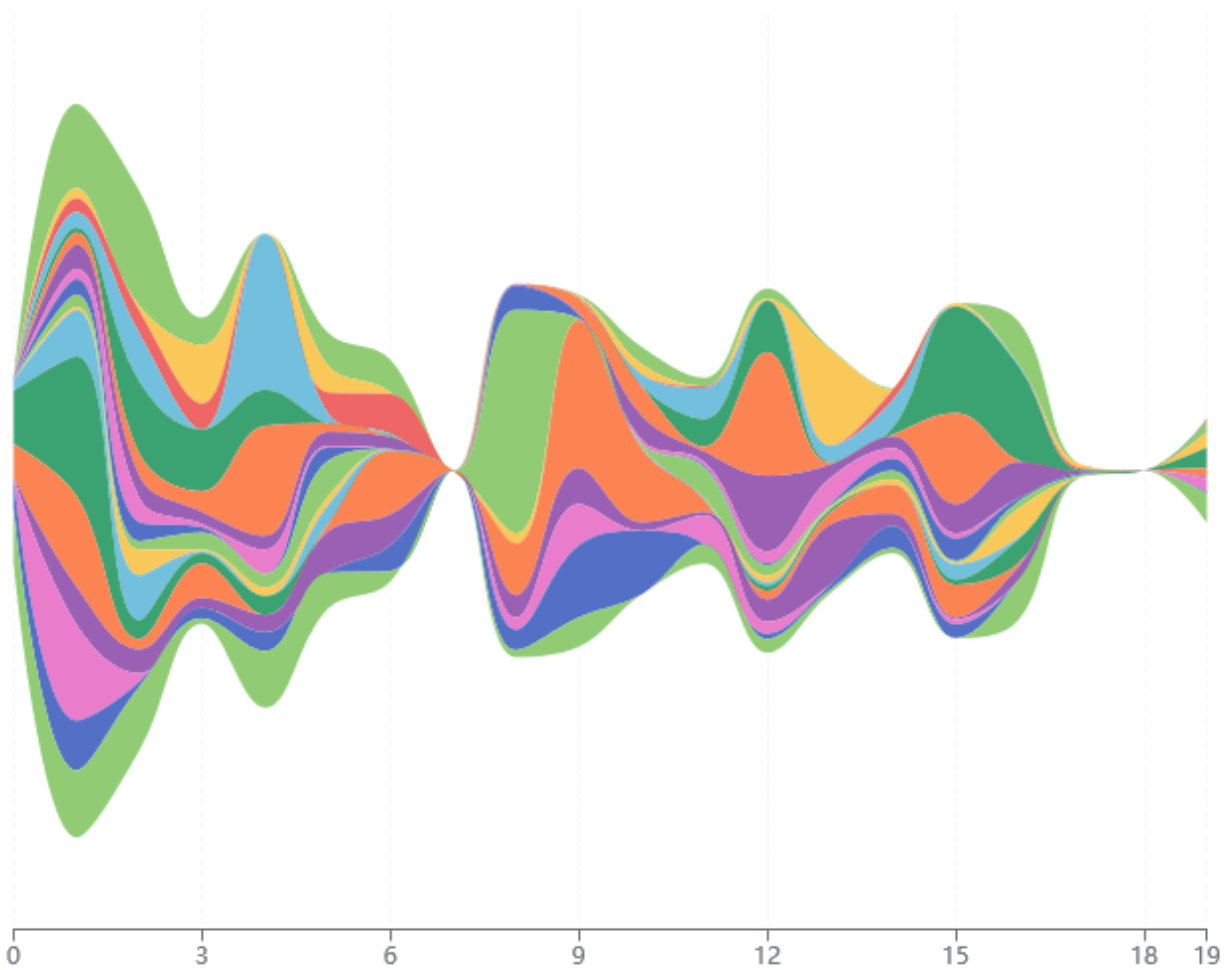


Information

Dieser Charttyp ist derzeit noch nicht im Dashlet-Plugin verfügbar. Bitte kontaktieren Sie unseren [Support](#), wenn Sie an der Nutzung dieser Darstellungsform interessiert sind.

Einführung

Es ist ein spezielles Flussdiagramm, das hauptsächlich verwendet wird, um die Änderungen eines Ereignisses oder Themas während eines Zeitraums darzustellen.



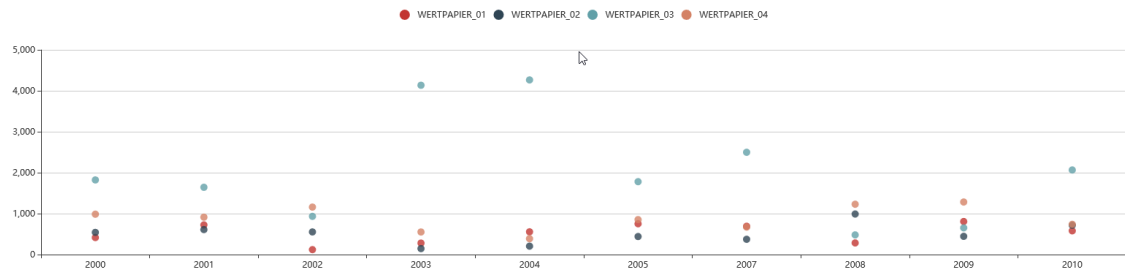
4.5.30 Streudiagramm (Scatter Chart)

Einführung

Das Streudiagramm (Bubble chart) Dashlet könnte mit das C3Charts oder mit Echarts entwickelt werden.

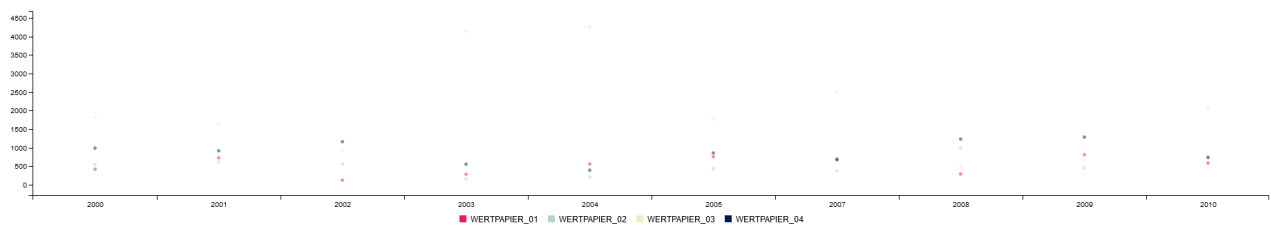
ECharts Beispiel

Wertpapier-Entwicklung E-Chart - Scatter Plot



C3-Charts Beispiel

Wertpapier-Entwicklung C3 Chart - Scatter Plot



Dashlets erzeugen

Die Streudiagramm (Bubble chart) Dashlet ist mit die folgende Code Block erstellt.

Echart Chart

Die Linien-/Kurvendiagramm Dashlet ist mit die folgende Code Block erstellt.

```
// DashletManager erzeugen
ExpertsDashlet.createDashletManager(reportContext, "DashletManager");

// Dashlet erzeugen
var dashlet = DashletManager.addDashletEChart("bar", "Wertpapier-Entwicklung", "E-Chart-Scatter Plot", true, mbr);

// Dashlet Option benutzen sodass der chart als scatter-plot angezeigt wird
dashlet.addOption("chart.series.type", "scatter");
```

C3 Chart

Die Linien-/Kurvendiagramm Dashlet ist mit die folgende Code Block erstellt.

EChart Dashlet erzeugen

```
// DashletManager erzeugen
ExpertsDashlet.createDashletManager(reportContext, "DashletManager");

// Dashlet erzeugen
var dashlet = DashletManager.addDashletC3Chart("scatter", "Wertpapier-Entwicklung",
"C3 Chart-Scatter Plot", true, mbr);
```

Für mehr Information für C3 Charts, siehe [C3-Charts](#).

Verschiedene Anzeige Formate sind möglich für diese Dashlet (siehe [ECharts \(4\) \[Options\]](#)).

FAQ

Wie kann die Größe des Scatter-Bubbles angepasst werden?

Die Größe des Bubbles kann mit die folgende Option angepasst werden.

```
// Set the symbolSize
dashlet.addOption("chart.series.symbolSize" , "20");
```

Wie kann die alle Scatter-Points in eine Farbe angezeigt werden?

Die alle Punkte kann in eine Farbe angezeigt werden mit die folgende Option

```
// Display all points in one color
dashlet.addOption("chart.series.itemStyle.color", "green");
```



```
// DashletManager erzeugen
ExpertsDashlet.createDashletManager(reportContext, "DashletManager");

// Rename the displayName
mbr.getColumnDefinition("dep_id").setDisplayNames("hierachy-id");
mbr.getColumnDefinition("parentId").setDisplayNames("parent-id");
mbr.getColumnDefinition("depName").setDisplayNames("NAME");

//Set the keyname attribute for the above columns
mbr.getColumnDefinition("depName").addAdditionalInfo("keyname","NAME");
mbr.getColumnDefinition("parentId").addAdditionalInfo("keyname","parent-id");
mbr.getColumnDefinition("dep_id").addAdditionalInfo("keyname","hierachy-id");

// Dashlet erzeugen
var dashlet = DashletManager.addDashlet("sunburst", "", "", true, mbr);
```

Dashlet Optionen

Durch das Hinzufügen von Optionen können Dashlets individuell angepasst werden.

Dashlet Optionen

```
dashlet.addOption("chart.data.hierachy.render", "all");
dashlet.addOption("chart.height", "680px");
dashlet.addOption("chart.series.fixed.name", "Gesamt");
dashlet.addOption("chart.series.highlightPolicy", "ancestor");
dashlet.addOption("chart.series.radius", "[0, '95%']");
dashlet.addOption("chart.series.startAngle", "180");
dashlet.addOption("chart.series.sort", "null");
dashlet.addOption("chart.series.levels.object", "[{label: {\n" +
"    rotate: 'radial'\n" +
"  }}, {\n" +
"    r0: '15%',\n" +
"    r: '45%',\n" +
"    itemStyle: {\n" +
"      borderWidth: 2\n" +
"    },\n" +
"    label: {\n" +
"      rotate: 'radial',\n" +
"    align: 'right',\n" +
"    padding: [0, 20, 0, 20],\n" +
"  }\n" +
"}, {\n" +
"    r0: '45%',\n" +
"    r: '90%',\n" +
"    label: {\n" +
"      align: 'right',\n" +
"      color: '#000'\n" +
"    },\n" +
"    itemStyle: {\n" +
"      borderWidth: 3,\n" +
"      opacity: 0.6\n" +
"    }\n" +
"}, {\n" +
"    r0: '70%',\n" +
"    r: '72%',\n" +
"    label: {\n" +
"      position: 'outside',\n" +
"      padding: 3,\n" +
"      silent: false\n" +
"    },\n" +
"    itemStyle: {\n" +
"      borderWidth: 3\n" +
"    }\n" +
"}]]");
```

FAQ

Wie kann ein onclick Funktion für Sunburst Chart gesetzt werden?

Das Onclick-Funktion kann mit die folgende Optionen gesetzt werden.

```
dashlet.addOption("chart.custom.function", "function click_sunburst_test(param)
{console.log(param);});
dashlet.addOption("chart.event.click", "click_sunburst_test");
```

Wie kann User-Spezifische Farbe für Sunburst Chart gesetzt werden?

Das User-Spezifische Farben kann mit die folgende Optionen gesetzt werden.

Ein Spalte ("Color") mit die User-Spezifische Farben sollte drinn sein in die MemoryBuffer

```
//Set the keyname attribute for the 'color' column
mbrSunburstData.getColumnDefinition("color").addAdditionalInfo("keyname","Color");

// Define the option for styling the Centre element of the sunburst chart
dashlet.addOption("chart.series.itemStyle.color", "#802080");
```

Wie kann das Sunburst Chart gefiltert werden?

Das SunBurst Chart kann mit die folgende Optionen gefiltert werden.

Hier ist ein Beispiel für die Filter-Funktion mit ein Button Component.

```
// button for filtering
buttonbar = dashlet.getDashlet().createDashletBar("DashletButtonBar", "TOP");
btn = buttonbar.addButton('All Data');
btn.addOption("button.onclick", "var opt = chart1.getOption(); var newdata =
chart1_predefined_series.data1[0].data; console.log(newdata); opt.series[0].data =
newdata; chart1.setOption(opt);");
btn = buttonbar.addButton("Data L1 ['Trains', 'Ships']", false);
btn.addOption("button.onclick", "var opt = chart1.getOption(); var newdata =
dashlet_searchHierachy(chart1_predefined_series.data1[0].data, 'NAME', ['Trains',
'Ships']); opt.series[0].data = newdata; chart1.setOption(opt);");
```

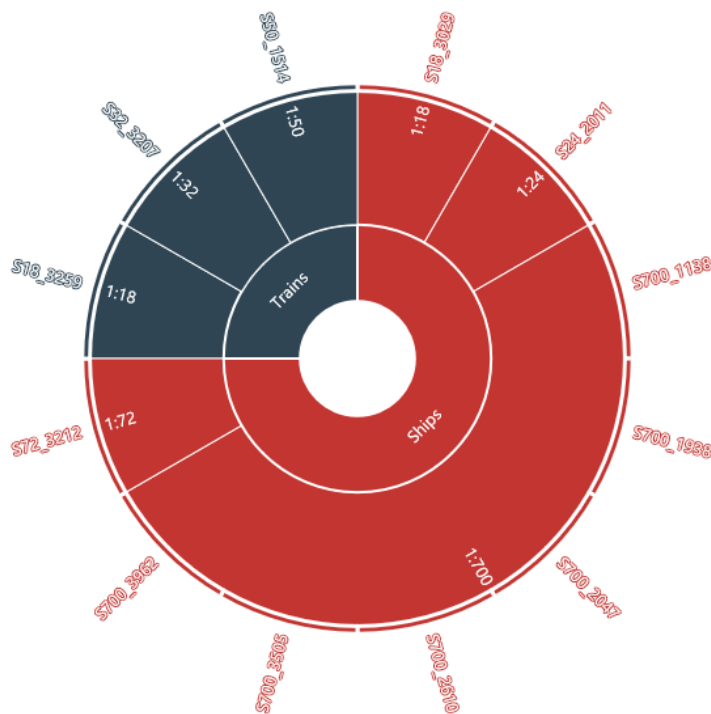
Hinweis:

- "chart1" ist die Name von der Sunburst Chart (Dashlet)
- "NAME" ist ein Feld in das SunburstChart

ClassicModels E-Chart - Sunburst Chart



All Data Data L1 ['Trains', 'Ships'] Data L2 1:32 Data L3 ['S700_1138', 'S700_1938']



4.5.32 SVG Visualisierungen/-Karten

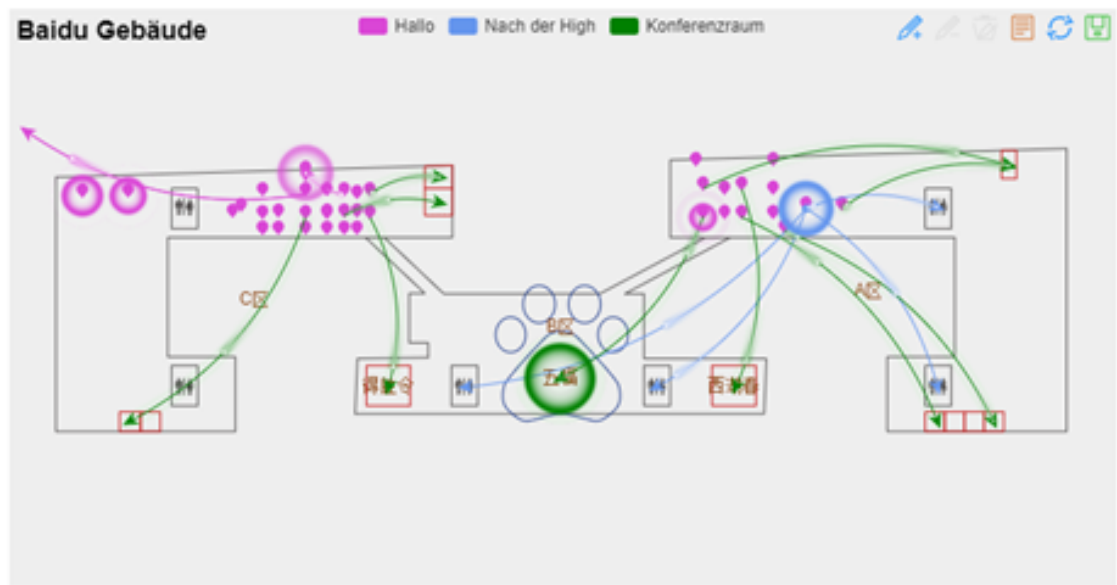


Information

Dieser Charttyp ist derzeit noch nicht im Dashlet-Plugin verfügbar. Bitte kontaktieren Sie unseren [Support](#), wenn Sie an der Nutzung dieser Darstellungsform interessiert sind.

Einführung

Es ermöglicht daten auf eigenen SVG-Grafiken anzeigen. Der Anwendungsfall: Fabrik- oder Gebäudeplan mit "Ampeln" (z.B. Lieferstatus oder Türzugang)



4.5.33 Tabellen

Einführung

Ein einfache Tabelle kann darstellt werden, wobei verschiedene Chart-Typen kann auch in jede Row angezeigt werden.

Einfache Tabelle Style gestreift ^ x

ID	Vorname	Nachname	E-Mail	Arbeitsleistung	Mini Pie
1	Detlef	Pöhnert	d.poehnert@tradui.de	<div></div>	<div></div>
2	Roland	Keitel	r.keitel@tradui.de	<div></div>	<div></div>
3	Felix	Busch	f.busch@tradui.de	<div></div>	<div></div>
4	Christian	Kaddatz	c.kaddatz@tradui.de	<div></div>	<div></div>
5	Jan	Blank	j.blank@tradui.de	<div></div>	<div></div>

Dashlets erzeugen

Die TabelleDashlet ist mit die folgende Optionen erstellt.

EChart Dashlet erzeugen

```
// DashletManager erzeugen
ExpertsDashlet.createDashletManager(reportContext, "DashletManager");

// Dashlet erzeugen
var dashlet = DashletManager.addDashletTable("Titel", "Subtitel", true, mbr);
```

Dashlet Optionen

Durch das Hinzufügen von Optionen können Dashlets individuell angepasst werden.

Dashlet Optionen

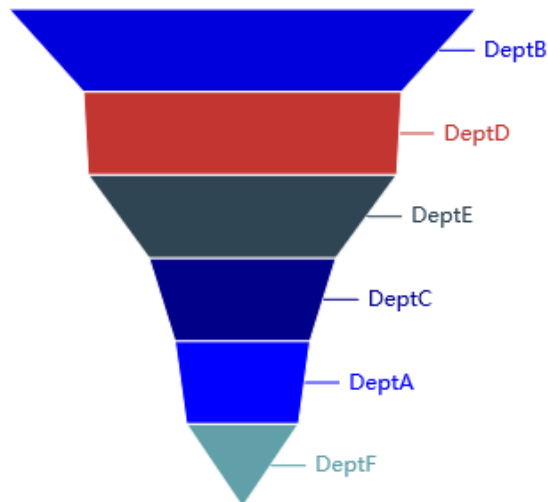
```
//styles setzen
dashlet.addOption("table.style", "table table-striped table-bordered stripe compact row-borders");

// bar Chart Optionen anpassen
tableDashlet.addOption("peity.chart.bar.height", "25");
tableDashlet.addOption("peity.chart.bar.width", "200");
```

4.5.34 Trichterdiagramm (Funnel Chart)

Einführung

Das Trichterdiagramm (Funnel Chart) kann mit der ECharts platform entwickelt werden.

ECharts Examples Funnel-Optionen**Funnel-Chart****Dashlets erzeugen**

Die Trichterdiagramm Dashlet ist mit die folgende Code Block erstellt.

EChart Dashlet erzeugen

```
// DashletManager erzeugen
ExpertsDashlet.createDashletManager(reportContext, "DashletManager");

// Dashlet erzeugen
var dashlet = DashletManager_PIE.addDashletEChart("pie", "Wertpapier-Entwicklung",
"E-Chart - Funnel Chart", true, mbrTmp);

// Setzt die series type als Funnel
dashlet.addOption("chart.series.type", "funnel");
```

Hinweis: Die Spalten-Namen sollte angepasst werden.


```
mbr.getColumnDefinition("SERIES").setDisplayNames("Series");  
mbr.getColumnDefinition("ISSUE_CONCAT").setDisplayNames("Pie-Name");  
mbr.getColumnDefinition("COUNT_TYPE").setDisplayNames("Pie-Value");
```

4.5.35 Venn Diagramm

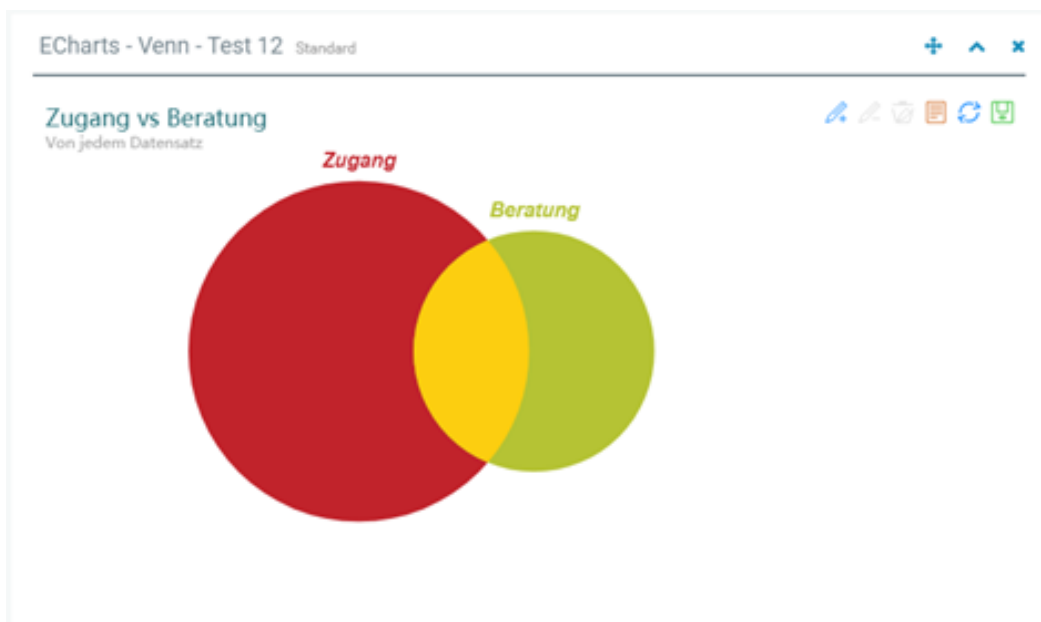


Information

Dieser Charttyp ist derzeit noch nicht im Dashlet-Plugin verfügbar. Bitte kontaktieren Sie unseren [Support](#), wenn Sie an der Nutzung dieser Darstellungsform interessiert sind.

Einführung

Venn Diagramm ist ein Logisches Diagramm und bezieht Beziehungen zwischen verschiedenen Sets proportional visualisieren



4.5.36 Wortwolke (Word-Cloud)



Information

Dieser Charttyp ist derzeit noch nicht im Dashlet-Plugin verfügbar. Bitte kontaktieren Sie unseren [Support](#), wenn Sie an der Nutzung dieser Darstellungsform interessiert sind.

Einführung

Möglichkeit zur Anzeige des gewichteten Vorkommens von Wörtern (z.B. bei Text Mining)



4.6 Optionen setzen

4.6.1 Anreicherung mit Dokumentation

Einführung

Es gibt mehrere Möglichkeiten, etwas Inline-html für die Dokumentation hinzuzufügen (z.B. um einen bestimmten KPI oder eine Metrik zu erklären).

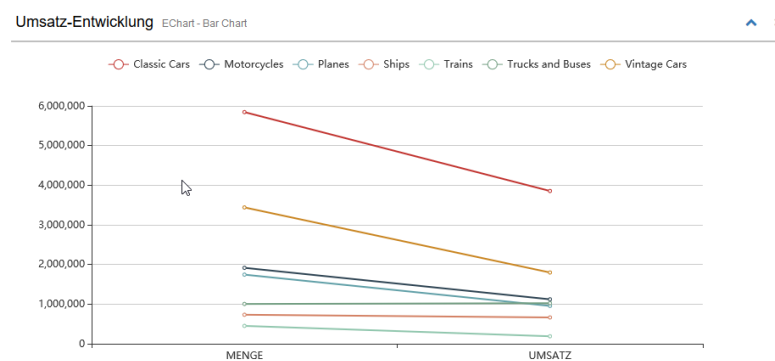
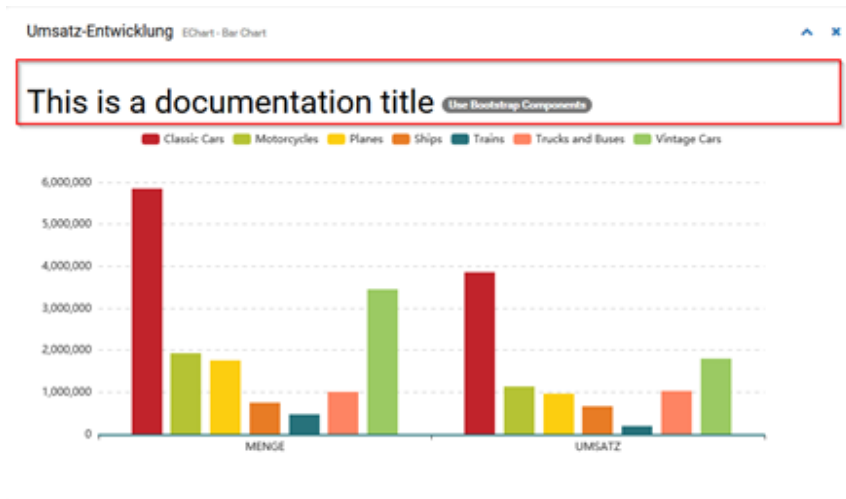
HTML Dokumentation in Chart-Bereich

Die HTML Content kann bevor oder nach ein Chart angezeigt werden, und ist mit die folgenden Optionen angezeigt.

Drilldown erstellen

```
//Display HTML Above Chart Content
dashlet.addOption("html.before.content", "<h1>This is a documentation title <span
class=\"badge badge-info\">Use Bootstrap Components</span></h1>");

//Display HTML After Chart Content
dashlet.addOption("html.after.content", "<h1>This is a documentation title</h1> <div
class=\"alert alert-success\">Lorem ipsum dolor si amet...</div>");
```



This is a documentation title

Lorem ipsum dolor si amet...

Dokumentation bei Box-Optionen

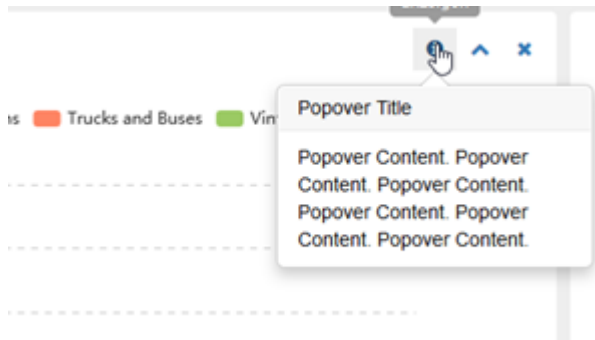
Die zweite Variante für Dokumentation ist die Möglichkeit zum Hinzufügen von Info-Icons mit Popover/Modal-Content beim Box-Optionen.

Popup Content erstellen

Das Popup Content kann mit der Box Optionen angezeigt werden, und ist mit die folgenden Optionen angezeigt.

Drilldown erstellen

```
//Display Popup Content
dashlet.addOption("box.info.popover.show", "show");
dashlet.addOption("box.info.popover.title", "Popover Title");
dashlet.addOption("box.info.popover.content", "Popover Content.");
```

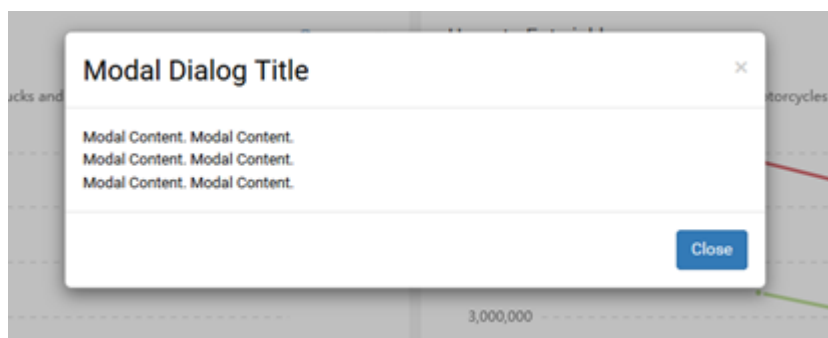


Modal-Dialog Content erstellen

Das Modal-Dialog Content kann mit der Box Optionen angezeigt werden, und ist mit die folgenden Optionen angezeigt.

Drilldown erstellen

```
//Display Modal-Dialog Content
dashlet.addOption("box.info.dialog.show", "show");
dashlet.addOption("box.info.dialog.title", "Modal Dialog Title");
dashlet.addOption("box.info.dialog.content", "Modal Content.");
```

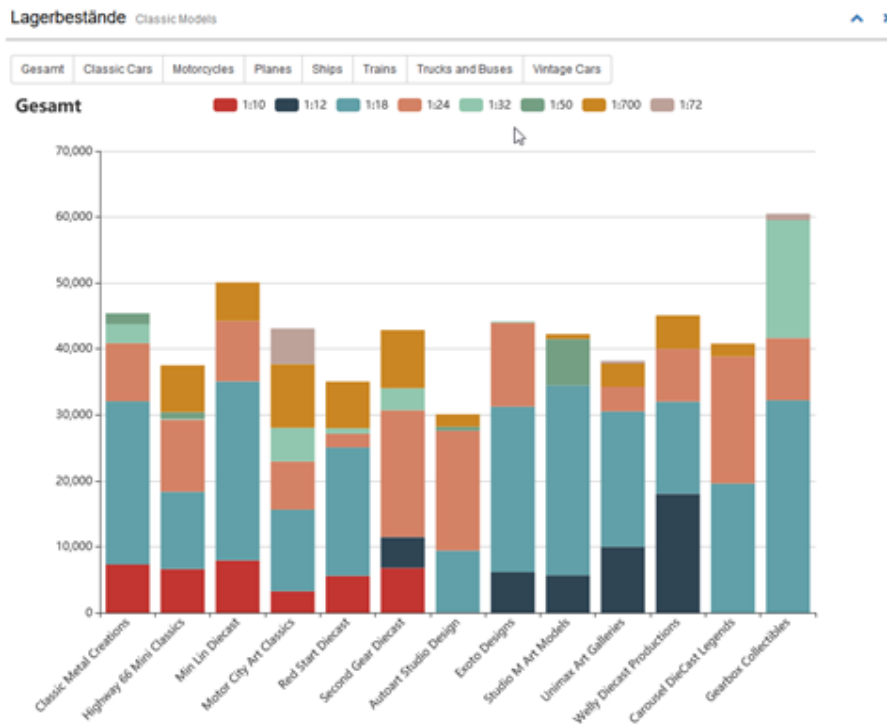


4.6.2 Komponenten

Buttons

Einführung

Es gibt Möglichkeiten Buttons hinzuzufügen.



Button erstellen

Die Buttons sind mit die folgenden Optionen erstellt.

Buttons erstellen

```
// z.B. Beim onFetch() in einem Dataset, die HashMap füllen
// store values in a HashMap
glProductlinesMap.put(row.PRODUCTLINE, row.PRODUCTLINE);

// Dashlet erstellen in onCreate() in einem Label
var dashlet = DashletManager.addDashletEChart("bar", "Lagerbestände", "Classic
Models", true, mbr);

// Button erstellen
buttonbar = dashlet.getDashlet().createDashletBar("DashletButtonBar", "TOP");
// create Buttons with the HashMap details
glProductlinesMapIterator = glProductlinesMap.keySet().iterator();
var dataid = 0;
while (glProductlinesMapIterator.hasNext())
{
    var prdline = glProductlinesMapIterator.next();
    dataid++;
    btn = buttonbar.addButton(prdline);
    btn.addOption("button.onclick", "chart1.clear();
        chart1_echartOption.series = chart1_predefined_series.data" + dataid + ";
chart1_echartOption.xAxis
    = chart1_predefined_xaxis.data" + dataid + ";
    chart1_echartOption.title.text = '" + prdline + "'";
    chart1.setOption(chart1_echartOption);");
}
```

Multiselect List

Einführung

Es gibt Möglichkeiten eine Multiselect List hinzuzufügen.

d.h. Abhängig von ein onclick-Funktion kann eine Liste angezeigt werden, die auch eine Möglichkeit bietet gelöscht zu machen.

Button 1 x Button 2 x 1:32 x Gesamt x Vintage Cars x

Multiselect Listen erstellen

Die Multiselect- List ist mit die folgenden Optionen erstellt.

Multiselect Box erstellen

```
// Create buttons
buttonbar = dashlet.getDashlet().createDashletBar("DashletBootstrapBar", "BOTTOM")
;
buttonbar.setId("bottombar1");
button = buttonbar.addBootstrapPiSelectButton();
button.setId("piselect1");

// Definition der Add und Delete Funktionen
dashlet.addOption("bar.n[bottombar1].component.n[piselect1].custom.function",
"function chart1_piselect_remove_callback(params)\n" +
"    {\n" +
"        alert('Callback Remove Button Piselect: ' + params.buttonvalue);
\n" +
"        console.log(params);\n" +
"    }\n" +
"    \n" +
"    function chart1_piselect_add_callback(params)\n" +
"    {\n" +
"        console.log(params);\n" +
"    }");

dashlet.addOption("bar.n[bottombar1].component.n[piselect1].initbuttons",["'Button 1', 'Button 2']"); // Initial Button-Liste anzuzeigen
dashlet.addOption("bar.n[bottombar1].component.n[piselect1].onRemoveButton","function(params) {chart1_piselect_remove_callback(params)}"); // Delete Funktions-Aufruf
dashlet.addOption("bar.n[bottombar1].component.n[piselect1].onAddButton","function(params) {chart1_piselect_add_callback(params)}"); //Add Funktions-Aufruf
dashlet.addOption("bar.n[bottombar1].component.n[piselect1].component.style","max-height: 50px; overflow-y: scroll; overflow-x: hidden;"); // Höhe der Multi-List Component

// Funktions-Aufruf in eine Dashlet, sodass die Multiselect-Liste erstellt wird, beim Click-Event (z.b.)
dashlet.addOption("chart.custom.function", "function click_test(param) {console.log(param.name);$('#piselect1').piselect('addButton', param.name);}");
dashlet.addOption("chart.event.click", "click_test");
```

Multiselect Boxen

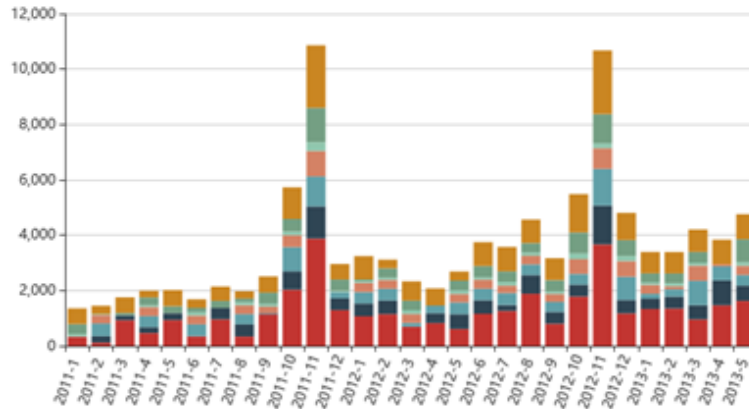
Einführung

Es gibt Möglichkeiten Multiselect Boxes hinzuzufügen. Derzeit kann bis 3 Multiselect Boxen angezeigt werden.

Bestellmengen

All selected (13) ▾

All selected (8) ▾

Classic Cars Motorcycles Planes Ships Trains Trucks and Buses Vintage Cars


Multiselect Boxen erstellen

Die Multiselect- Box ist mit die folgenden Optionen erstellt.

Der gibt zwei Teile für die Implementierung -erste Teil implementiert ein Data-Dashlet und die zweite Teil implementiert die akutelle Dashlet (z.b. Bar Chart Dashlet). Die Werte für die BarChart Dashlet ist von das Data Dashlet ermittelet.

Multiselect Box erstellen

```
// Part 1 - Dashlet Data erstellen.
// Dashlet 2 ist das unsichtbare Data-Dashlet data1
dashlet2 = DashletManager.addDashlet("DashletData","Bestellmengen data", "", false,
mbr1);
dashlet2.setId("data1");

// Das Script-Template für 2 Multiselect-Filter auf Dimension z0 und z1
dashlet2.addOption("data.script.template","js/echart/multiselect-dim-z2");

// Filter des Scriptes definieren! Diese verweisen auf die beiden später definierten
Multiselect-Buttons
// Abruf der selektierten Werte mit der Funktion (Name der
Komponente)_getSelectOptions()
dashlet2.addOption("data.script.dimZ0Filter","multiselect1_getSelectOptions()");
dashlet2.addOption("data.script.dimZ1Filter","multiselect2_getSelectOptions()");

// Setzen der Dimensionen am DataTable z sind die beiden Filter-Dimensionen
// Dimension x ist die x-Achse des Chart, Dimension y ist die Series-Abbildung

var dt = dashlet2.getDashlet().getData();
dt.getColumns().get(dt.getColumnIndex("PRODUCTVENDOR")).setKeyname("dim-z");
dt.getColumns().get(dt.getColumnIndex("PRODUCTSCALE")).setKeyname("dim-z");
dt.getColumns().get(dt.getColumnIndex("ORDERYEAR")).setKeyname("dim-x");
dt.getColumns().get(dt.getColumnIndex("PRODUCTLINE")).setKeyname("dim-y");

// Der Chart muss aus den Daten wissen, welches die Series-Spalte ist
mbr1.getColumnDefinition("PRODUCTLINE").setDisplayname("Series");

// Part 2 - Dashlet erstellen
// Das Bar-Chart-Dashlet erstellen
dashlet = DashletManager.addDashletEChart("bar","Bestellmengen", "", true, mbr1);
dashlet.setId("chart1");

// xAxis auf vordefinierte Dimension X des Data-Dashlets setzen
dashlet.addOption("chart.predefined.xAxis", "data1_dimX0");

// Legende auf vordefinierte Dimension Y des Data-Dashlets setzen
dashlet.addOption("chart.predefined.legend", "data1_dimY0");

// wegen der Matrix-Funktion für die Serie muss hier nochmal die Series-Spalte als
Erkennung definiert.
dashlet.addOption("chart.series.data.column","Series");

// Funktion mit der die Series-Daten aus der Matrix gelesen werden.
dashlet.addOption("chart.series.data.function",
"data1_getMatrixData('seriesidentifier')");

// Button-Bar topbar1 erzeugen
buttonbar = dashlet.getDashlet().createDashletBar("DashletBootstrapBar", "TOP");
```

```
buttonbar.setId("topbar1");

// MultiSelect-Button multiselect1 erzeugen
button = buttonbar.addBootstrapMultiSelectButton();
button.setId("multiselect1");

// MultiSelect-Button multiselect2 erzeugen
button = buttonbar.addBootstrapMultiSelectButton();
button.setId("multiselect2");

// Dimensions-Variable z0 des Data-Dashlets setzen
dashlet.addOption("bar.n[topbar1].component.n[multiselect1].options.var",
"data1_dimZ0");

// Dimensions-Variable z1 des Data-Dashlets setzen
dashlet.addOption("bar.n[topbar1].component.n[multiselect2].options.var",
"data1_dimZ1");
```

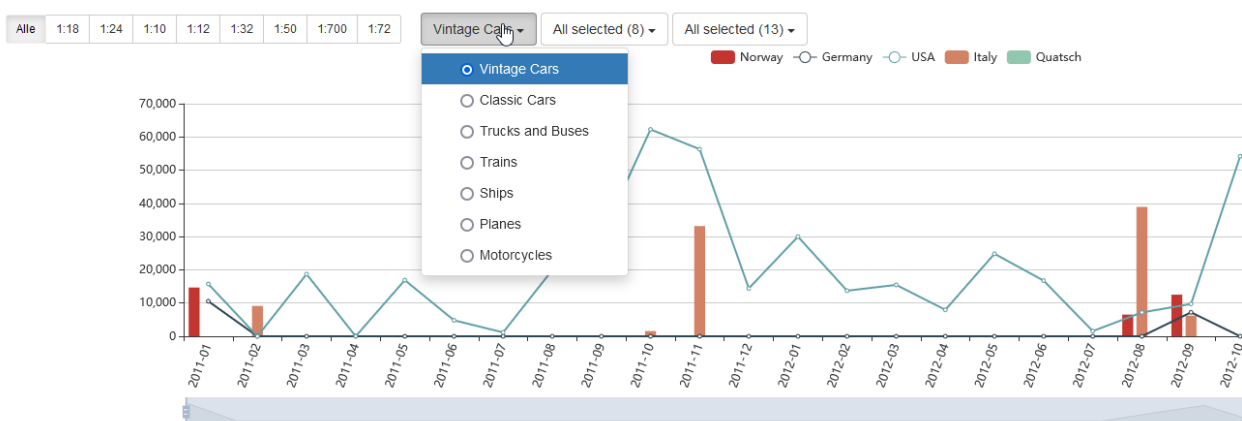
Hinweis:

- dim-x ist die X-Werte
- dim-z ist die MultiSelect-box Werte
- dim-y ist die Y-Werte (oder Legende). Wenn verschiedene Spalte als Legende verwenden soll, kann der Option 'value-dim-y' verwendet werden.

Single-select Boxen**Einführung**

Es gibt Möglichkeiten Single-select Boxes hinzuzufügen.

Umsatz nach Land

**Single-select Boxen erstellen**

Die Single select- Box ist mit die folgenden Optionen erstellt.

Multiselect Box erstellen

```
// MultiSelect nur als Single-Select verwenden  
dashlet3.addOption("bar.n[topbar1].component.n[multiselect1].select.type","single");
```

5 Dashboards designen

5.1 Best Practice

5.1.1 Einführung

Bei jede Dashlet entwickeln, ist es immer wichtig die folgende Punkte zu errinieren.

1. Vermeiden Sie die Verwendung "extremer" Farben; nur wenn Sie den Anwender auf etwas aufmerksam machen wollen (z.B. Hinweise/Warnungen/Tendenzen oder die Anzeige von Ausreißern oder Schwellwerten)
2. Bleiben Sie in Ihrem Farbmuster (gleiche Farbe für gleiche Serie, Wiedererkennung)
3. Versuchen Sie nicht, die „eierlegende Wollmilchsau" zu entwickeln (Aufwand/Nutzen)
4. Verwenden Sie keine ausgefallenen Schriftarten wie Comic Sans (selbst wenn es konfigurierbar ist).
5. Versuchen Sie, Ihre Stakeholder zu differenzieren:
6. Nicht jeder benötigt allumfassende Konfigurationsmöglichkeiten
7. Versuchen Sie, sich auf die relevanten Aspekte zu konzentrieren (80/20 Regel)
8. Beschreiben Sie, was der Benutzer vor sich sieht (Metrikdefinition) mit der Inline-Dokumentations-Funktion, beschriften Sie Achsen, aktivieren Sie Tooltips

5.2 Positionierung, Größe & Responsive Design

5.2.1 Positionierung

Einführung

Die Positionierung des Dashboards ist immer abhängig von der Reihenfolge der Instanziierung ,d.h. das Dashboard die als erstes gerendet ist, ist alles erstes angezeigt. Jedes einzelne Dashboard ist in ein Separates 'Box' gerendert.

Eine besondere Spezialität ist mehrere Dashboards in eine Box zu renderen (siehe [Kombination von meherere Dashlets in ein Box](#)).

5.2.2 Positionierung

Einführung

Als Default die Höhe ist 400px für ein Chart. Die Größe eines Dashboards kann mit das folgende Option gesetzt werden.

```
// create bar chart
var dashlet = DashletManager.addDashletEChart("bar", "", "", false, mbr);
// Chart Height
dashlet.addOption("chart.height", "50px");
```

5.2.3 Responsive Design

Einführung

Das Ziel eines responsive design ist es, Webseiten zu erstellen, die die Bildschirmgröße und Ausrichtung des Besuchers erkennen und das Layout entsprechend ändern.

- 12 Slots (Counter bedeutet Prozentsatz, 6 = 50%, 3 = 25%, ...)
- 3 Kategorien: Desktop, Tablet, Phone
- Unterschiedliche Breite für verschiedene Diagramme verwenden
- Vermeiden die Einstellung einer expliziten Breite



Die Responsive Grid könnte mit die folgende Optionen gesetzt werden.

```
// create bar chart
var dashlet = DashletManager.addDashletEChart("bar", "", "", false, mbr);
// ResponsiveGrid - Option 1
dashlet.setResponsiveGrid("6-12-12");
// ResponsiveGrid - Option 2
dashlet.addOption("responsive.grid","6-12-12");
```

Mehr Information zu ResponsiveGrid finden Sie unter [Tutorial - Grundlegendes zu Dashlets](#).

5.3 Mehrsprachigkeit/Lokalisierung

5.3.1 Einführung

Mit die folgende Option kann die Sprache gesetzt werden. Alle Sprachen sind nicht unterstützt in jede Dashlet-Typ

```
dashlet.addOption("locale","en_EN");
```

6 Interaktivität

6.1 Alert-Box Anzeigen

6.1.1 Einführung

Es gibt die Möglichkeit eine Alert Box in das Dashlet zeigen zu können.

6.1.2 Optionen

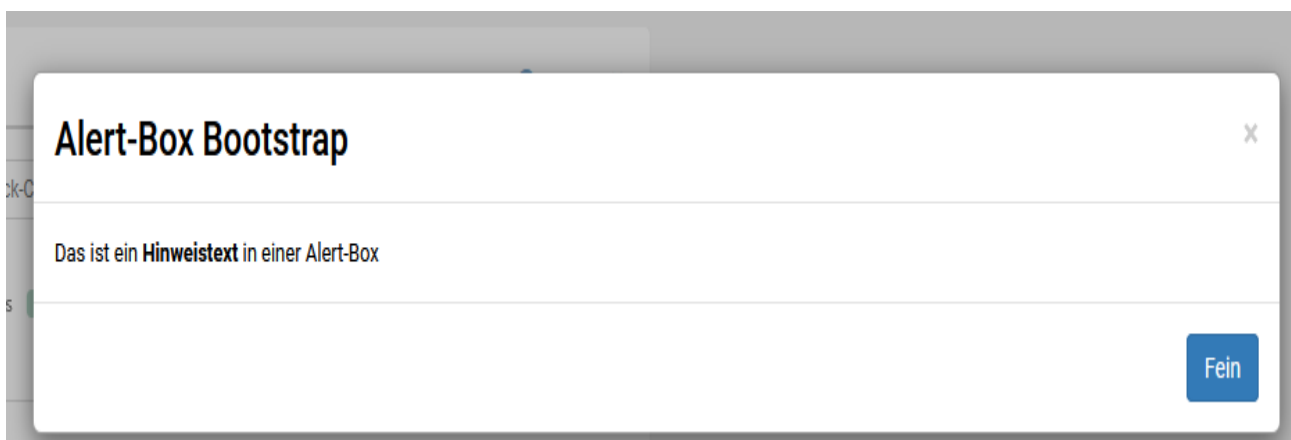
Die Alert-Box könnte mit die folgende Option erstellt werden.

Eigene Skripte erstellen

```
// Bar Chart Dashlet erstellen
var dashlet = DashletManager.addDashletEChart("bar", "Umsatz-Entwicklung", "EChart -
Bar Chart", true, mbrUmsatz);

// Info box as pop-up
dashlet.addOption("box.alert.dialog.show", "show");
dashlet.addOption("box.alert.dialog.title", "Alert");
dashlet.addOption("box.alert.dialog.type", "large");
dashlet.addOption("box.alert.dialog.content", "Empty");
dashlet.addOption("box.alert.dialog.footer.show", "show");
dashlet.addOption("box.alert.dialog.footer.button.close.title", "OK");

// Alert Box in ein Button darstellen
//Add buttons
buttonbar = dashlet.getDashlet().createDashletBar("DashletButtonBar", "TOP");
btn = buttonbar.addButton("Alert 2 - Bootstrap");
btn.addOption("button.onclick", "dashlet_Alert('chart4', 'Das ist ein
<b>Hinweistext</b> in einer Alert-Box', {title: 'Alert-Box Bootstrap',
closebuttontitle: 'Fein'}});");
```



6.2 Confirm-Box Anzeigen

6.2.1 Einführung

Es gibt die Möglichkeit eine Confirm Box in das Dashlet zeigen zu können.

6.2.2 Optionen

Einfache Confirm-Box

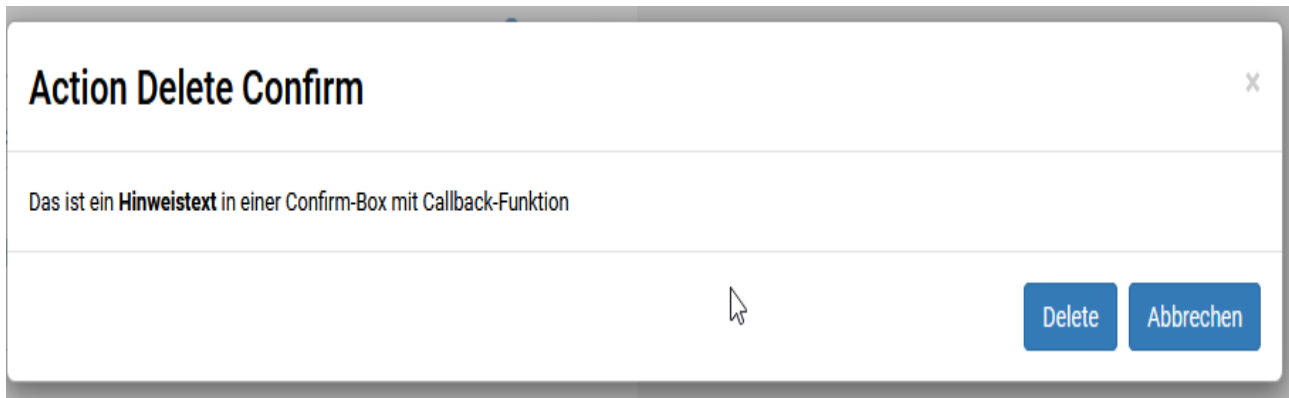
Die Confirm-Box könnte mit die folgende Option erstellt werden.

Eigene Skripte erstellen

```
// Bar Chart Dashlet erstellen
var dashlet = DashletManager.addDashletEChart("bar", "Umsatz-Entwicklung", "EChart -
Bar Chart", true, mbrUmsatz);

// Confirm box as pop-up
dashlet.addOption("box.confirm.dialog.show", "show");
dashlet.addOption("box.confirm.dialog.title", "Confirm");
dashlet.addOption("box.confirm.dialog.type", "large");
dashlet.addOption("box.confirm.dialog.content", "Empty");
dashlet.addOption("box.confirm.dialog.footer.show", "show");
dashlet.addOption("box.confirm.dialog.footer.button.close.title", "OK");
dashlet.addOption("box.confirm.dialog.footer.button.cancel.show", "show");
dashlet.addOption("box.confirm.dialog.footer.button.cancel.title", "Ups, doch nicht")
;

// Confirm Box mit Buttons darstellen
//Add buttons
buttonbar = dashlet.getDashlet().createDashletBar("DashletButtonBar", "TOP");
btn = buttonbar.addButton("Confirm - Bootstrap with Callback-Close");
btn.addOption("button.onclick", "dashlet_Confirm('chart4', 'Das ist ein
<b>Hinweistext</b> in einer Confirm-Box mit Callback-Funktion', {title: 'Action
Delete Confirm', closebuttontitle: 'Delete', closebuttonclick: function()
{alert('Callback Delete-Button')}, cancelbuttontitle: 'Abbrechen', cancelbuttonclick:
function(){alert('Callback Cancel-Button')}});");
btn = buttonbar.addButton("Input - Bootstrap with Callback-Close");
btn.addOption("button.onclick", "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 with input text:' + document.getElementById('inputfieldid').value)},
cancelbuttontitle: 'Abbrechen', cancelbuttonclick: function(){alert('Callback Cancel-
Button')}});");
```



Confirm-Box mit ein Input Feld

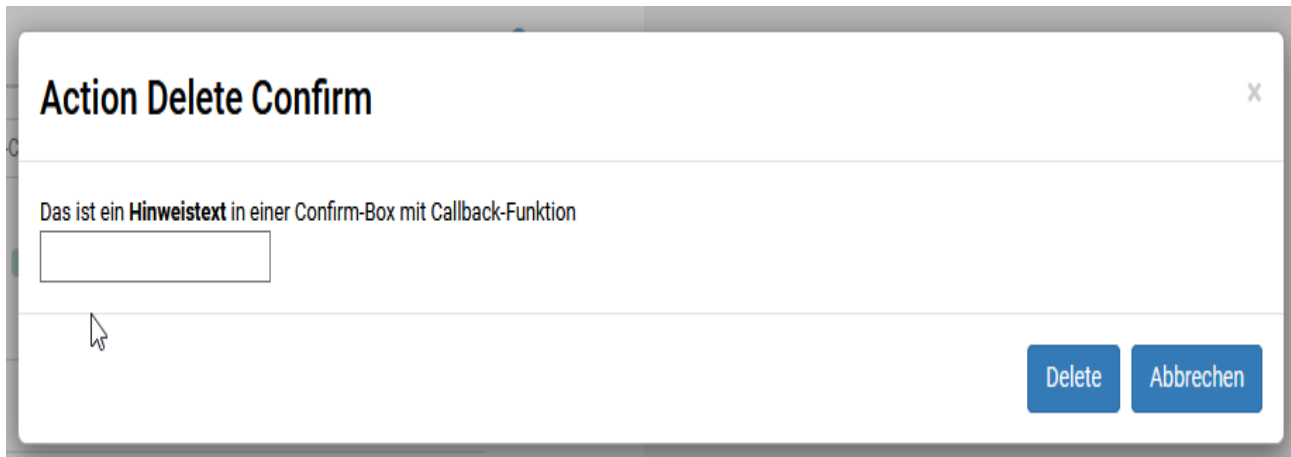
Es ist auch Möglich die Confirm Box mit ein Input Text anzuzeigen und es ist mit die folgende Option gemacht.

Eigene Skripte erstellen

```
// Bar Chart Dashlet erstellen
var dashlet = DashletManager.addDashletEChart("bar", "Umsatz-Entwicklung", "EChart - Bar Chart", true, mbrUmsatz);

// Confirm box as pop-up
dashlet.addOption("box.confirm.dialog.show", "show");
dashlet.addOption("box.confirm.dialog.title", "Confirm");
dashlet.addOption("box.confirm.dialog.type", "large");
dashlet.addOption("box.confirm.dialog.content", "Empty");
dashlet.addOption("box.confirm.dialog.footer.show", "show");
dashlet.addOption("box.confirm.dialog.footer.button.close.title", "OK");
dashlet.addOption("box.confirm.dialog.footer.button.cancel.show", "show");
dashlet.addOption("box.confirm.dialog.footer.button.cancel.title", "Ups, doch nicht");
;

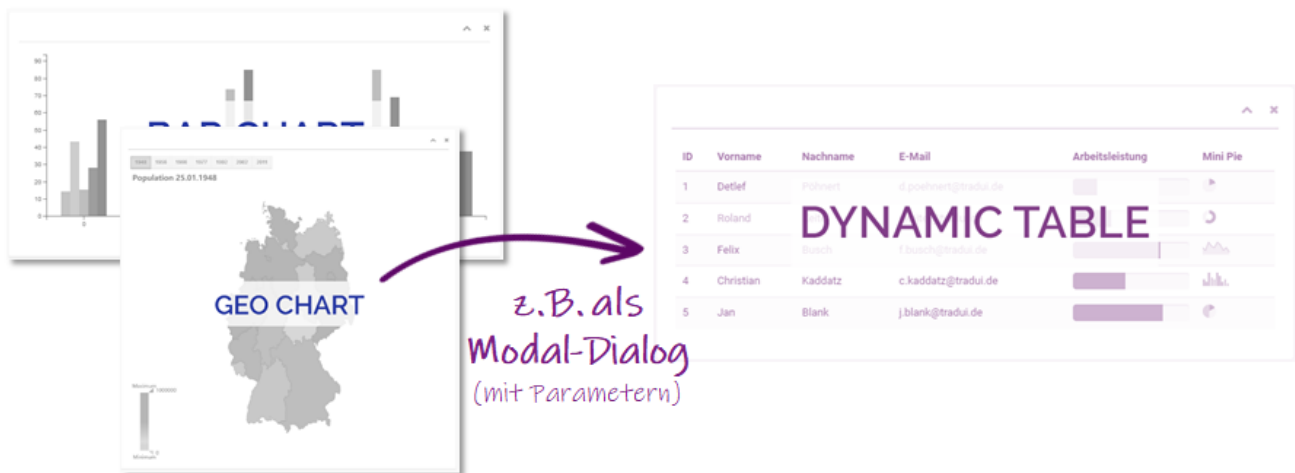
// Confirm Box mit Buttons darstellen
//Add buttons
buttonbar = dashlet.getDashlet().createDashletBar("DashletButtonBar", "TOP");
btn = buttonbar.addButton("Input - Bootstrap with Callback-Close");
btn.addOption("button.onclick", "dashlet_Confirm('chart4', 'Das ist ein
<b>Hinweistext</b> in einer Confirm-Box mit Callback-Funktion<br/><input type=text
id=inputfieldid name=inputfieldname>', {title: 'Action Delete Confirm',
closebuttontitle: 'Delete', closebuttonclick: function(){alert('Callback Delete-
Button with input text:' + document.getElementById('inputfieldid').value)},
cancelbuttontitle: 'Abbrechen', cancelbuttonclick: function(){alert('Callback Cancel-
Button')}});");
```



6.3 Drilldowns

6.3.1 Einführung

Die 'DrillDown' Funktionalität bietet die Möglichkeit für eine „Master-Detail“ Ansicht durch Klick auf Element an. Das Ziel ist von aggregierter Ansicht auf Detail-Datensätze gelangen zu können.



6.3.2 Drilldown erzeugen

Die Drilldown Funktionalität ist erstellt beispielsweise mit die folgende Code Block für ein click event in ein Bar Chart.

Drilldown erstellen

```
//drill down functionality
dashlet.addOption("chart.custom.function", "function table1_load(param)
{table1_lazyLoadTable(param.seriesName, 10, param.name , 11);
openDashletModalPanel('table1');}");
dashlet.addOption("chart.event.click", "table1_load");
```

Für der DataTable als Popup darzustellen muss die folgende Optionen definiert werden.

```
var tableDashlet1 = DashletManager.addDashletTableDynamic("table1" , "" , true ,
mbr);
tableDashlet1.setId("table1");
tableDashlet1.getDashlet().createModalBox();
```



Hinweis

Der DataTable muss als LazyLoad definiert sein.

6.4 Eigene CSS-Datei für jede Dashlet hinzufügen

6.4.1 Einführung

Es gibt die Möglichkeit eigene Styles (CSS-Datei) für jede Dashlet zu definieren.

6.4.2 Optionen

Die externe CSS-Datei könnte als folgendes zu jede Dashlet hinzugefügt werden.

Eigene Skripte erstellen

```
// Custom-Style-Sheets für Dashlet verwenden / Ressourcen-File --> dashlet_styles.css
dashlet.addOption("dashlet.custom.styles", readResourceFileAsString("dashlet_styles.css"));

// Funktion definition
function readResourceFileAsString(filename)
{
    try
    {
        return "" +
        ToolboxMasterObject.getResourceHelper().readResourceFileUTF8(reportContext,
        glResourceFolderReport + filename);
    } catch (er) {
        return "";
    }
}
```

6.5 Eigene Skripte erstellen

6.5.1 Einführung

Es gibt die Möglichkeit eigene Skripte in das HTML render zu können. Die Vorteil ist das die Interaktivität zwischen verschiedene Dashlets ist möglich zu machen.

6.5.2 Optionen

Die eigene Skripte könnte als folgendes erstellt werden.

Variant 1 (Echarts):

Eigene Code hinzufügen zb. für ein 'Click' event in ein Bar Chart.

Eigene Skripte erstellen

```
// Bar Chart Dashlet erstellen
dashlet = DashletManager.addDashletEChart("bar","Sales", "", true, mbr1);

// Option to render a function to the click event in a bar chart.
dashlet.addOption("chart.custom.function", "function table1_load(param)
{openDashletModalPanel('table1');table1_lazyLoadTable(param.name, 9,
param.seriesName, 7);}");
dashlet.addOption("chart.event.click", "table1_load");
```

Variant 2 (HTML Dashlet) :

Eigene Javascript Code hinzufügen in ein HTML Dashlet.

Eigene Skripte erstellen

```
// HTML Dashlet erstellen
var dashlet= DashletManager.addDashlet("DashletHTML", "", "", false, null);

// HTML Code
var customHTML = "";
customHTML += "<table id=\"Dash2\" class=\"table table-striped table-bordered stripe compact row-borders\" width=\"100%\"></table>";
// render the Code in HTML File
dashlet.addOption("html", customHTML);

// Javascript Code
var customJS = "";
// function to call the adhoc report from dashboard
customJS += "    function callAdhocBericht() {\n";
customJS += "        paramDateFrom = startDate ; \n";
customJS += "        paramDateTo = endDate; \n";
customJS += "        try{openJobCreatorDialog([\"";
customJS += "            {name:'--__report', value:'/kurzcheck/"
Kumulschadenfälle.rptdesign'},\"+
                                \"{name:'--__outputFormat',
value:'PDF,HTML'},\"+
                                \"{name:'--paramDateFrom',
value:paramDateFrom},\"+
                                \"{name:'--paramDateTo',
value:paramDateTo}]); \n"+
                                "    }catch(e){};\n" ;
customJS += "    } \n\n";
// render the Code in HTML File
dashlet.addOption("javascript", customJS + "\n\n");
```

Variant 3 (DataTable Dashlet) :

Eigene Javascript Code hinzufügen in ein DataTable Dashlet.

```
// Variante 1 :  
// Custom Skript aus einer externen js-Datei  
tableDashlet1.addOption("table.customScript", readResourceFileAsString("dashlet_table_planning_stock_custom_scripts.js"));  
  
// Variante 2 :  
// Custom Skript in der .rptdesign-Datei definieren  
dashlet.addOption("table.customScript",  
"$ (document).ready(function(){\n" +  
"    var table = $('#table1').DataTable();\n" +  
"    table.$('tr').css(\"cursor\", \"pointer\");\n" +  
"    $('#table1 tbody').on( 'click', 'tr', function () {\n" +  
"        alert('Click on table row: ' + table.rows(this).indexes()[0] + '\\ndata: [' +  
+ table.rows(this).data()[0] + ']);\n" +  
"    } );\n" +  
"});");
```

6.6 Expand Button anzeigen

6.6.1 Einführung

Es gibt die Möglichkeit, eigene Expand Buttons in Dashlets zu zeigen, sodass der Chart klein und groß gemacht werden kann.

6.6.2 Optionen

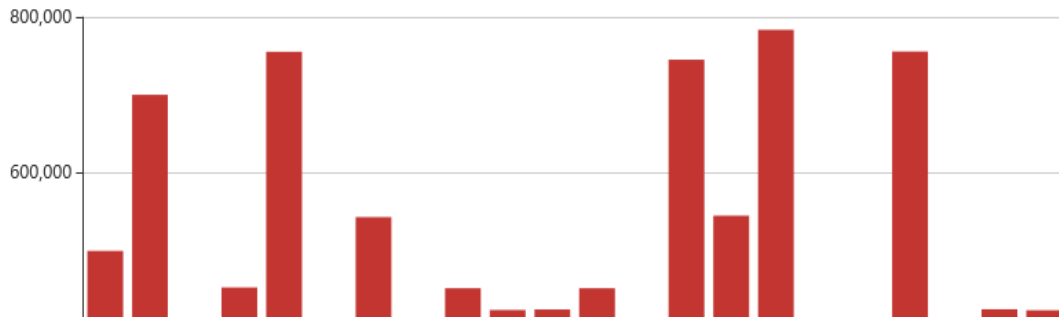
Die Expand Button könnte mit der folgenden Option erstellt werden.

Expand Button anzeigen

Eigene Skripte erstellen

```
// Bar Chart Dashlet erstellen  
var dashlet = DashletManager.addDashletEChart("bar", "Umsatz nach Land", "", true,  
mbrUmsatz);  
  
// show expand / compress icon  
dashlet.addOption("box.expand.show", "true");
```

Umsatz nach Land



6.6.3 FAQ

Wie kann andere Chart(s) ausgeblendet werden wenn ein Chart groß gemacht wird?

Ein andere Chart(s) kann mit die folgende Option ausgeblendet wird wenn ein Chart groß gemacht ist.

Die List von Chart-Ids sollte semi-colon seperated sein.

```
// list of dashlets that are to be hidden (semicolon seperated)
dashlet.addOption("box.expand.hide.ids", "leaflet1;table1");
```

6.7 Gruppieren verschiedene Dashlets zusammen

6.7.1 Einführung

Es gibt die Möglichkeit verschiedene Dashlets in eine Gruppe darstellen zu können. Diese ist besonders interessant für die KPI-Dashlets.

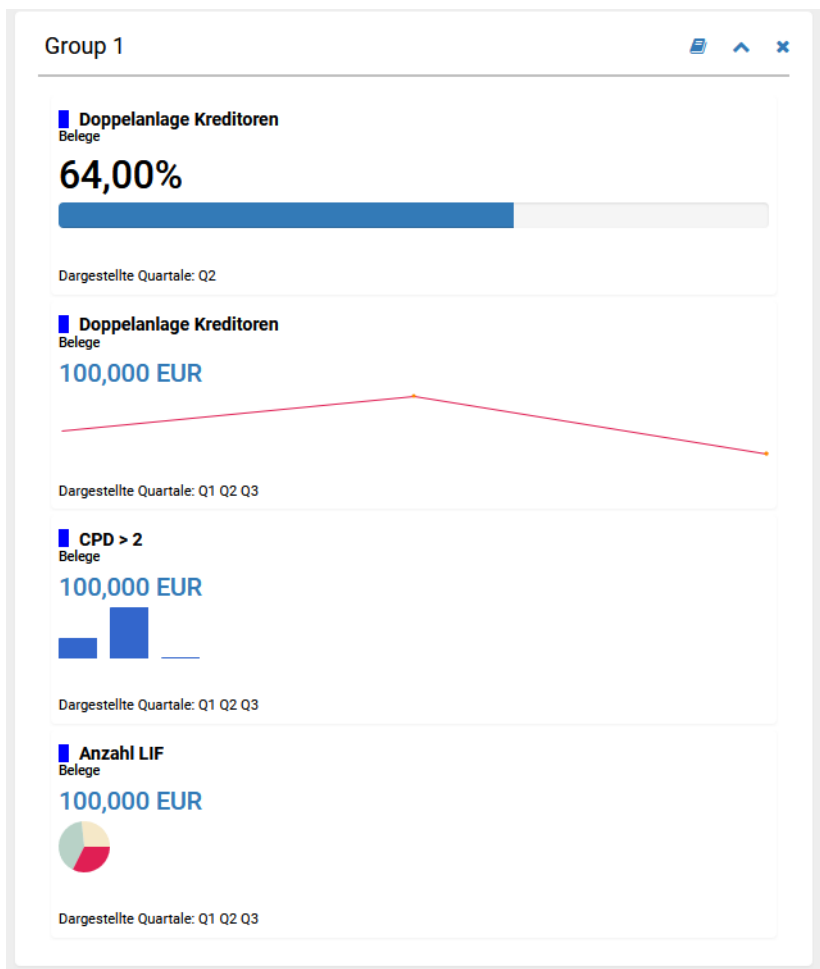
6.7.2 Optionen

Die Gruppierung könnte mit die folgende Option erstellt werden.


```
// HTML Group Dashlet
dashlet = DashletManager.addDashlet("DashletHtmlGroup", "Group 1", "", true, null);
dashlet.setId("kpi_group");
dashlet.setResponsiveGrid("4-4-6-12");
// Add children
dashlet.addOption("childid","kpi_boxes1");
dashlet.addOption("childid","kpi_boxes2");

// KPI Box (Percentage chart)
dashlet = DashletManager.addDashlet("DashletKpiBox", "", "", true, mbrKpi_2);
dashlet.setId("kpi_boxes1");
dashlet.addOption("kpi.box.responsive","12-12-12-12");

// KPI Box (Line, Bar and Pie chart)
dashlet = DashletManager.addDashlet("DashletKpiBox", "", "", true, mbrKpi_1);
dashlet.setId("kpi_boxes2");
dashlet.addOption("kpi.box.i[0].responsive","12-12-12-12");
dashlet.addOption("kpi.box.i[1].responsive","12-12-12-12");
dashlet.addOption("kpi.box.i[2].responsive","12-12-12-12");
```



6.7.3 FAQ

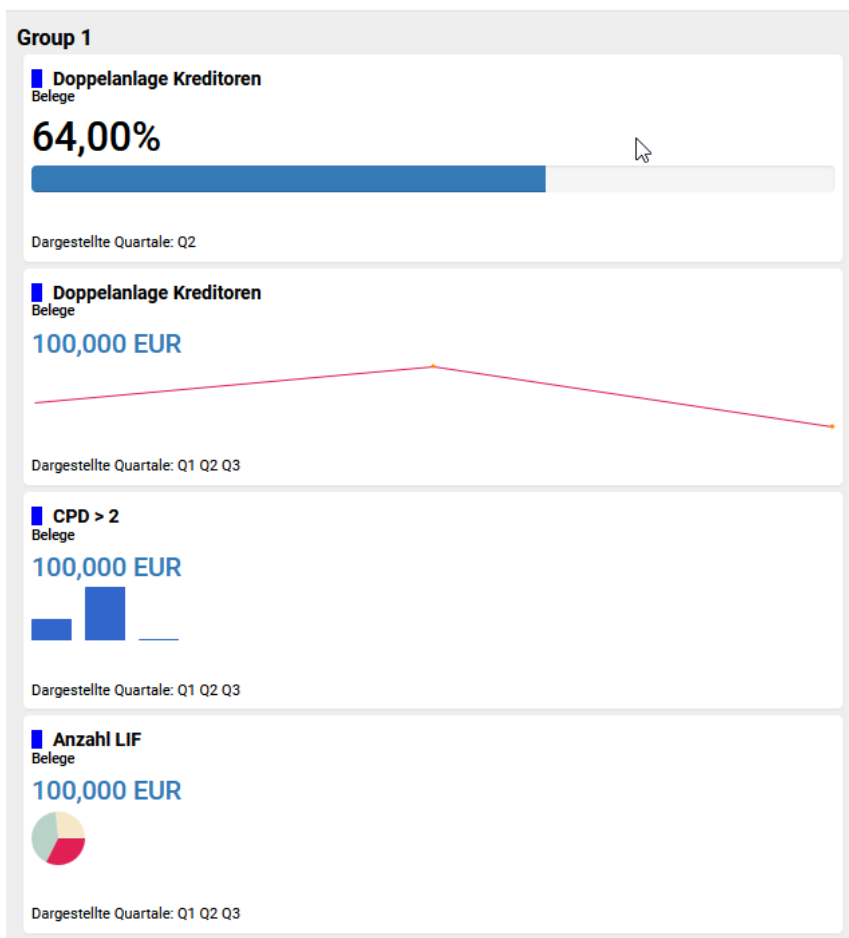
Wie kann die Box für die Main Group ausgeblendet werden und nur ein Gruppe-Titel anzeigen?

Diese Anforderungen kann mit die folgende Code-Block dargestellt werden.

```
// HTML Group Dashlet (BoxOption als FALSE setzen)
dashlet = DashletManager.addDashlet("DashletHtmlGroup", "Group 1", "", false, null);

dashlet.setId("kpi_group");
dashlet.setResponsiveGrid("4-4-6-12");
dashlet.addOption("childid", "kpi_boxes1");
dashlet.addOption("childid", "kpi_boxes2");

// Ein HTML Block rendern für die Title anzuzeigen
dashlet.addOption("html", "<span style = \"font-size:16px; font-weight:bold\">Group 1</span>");
```

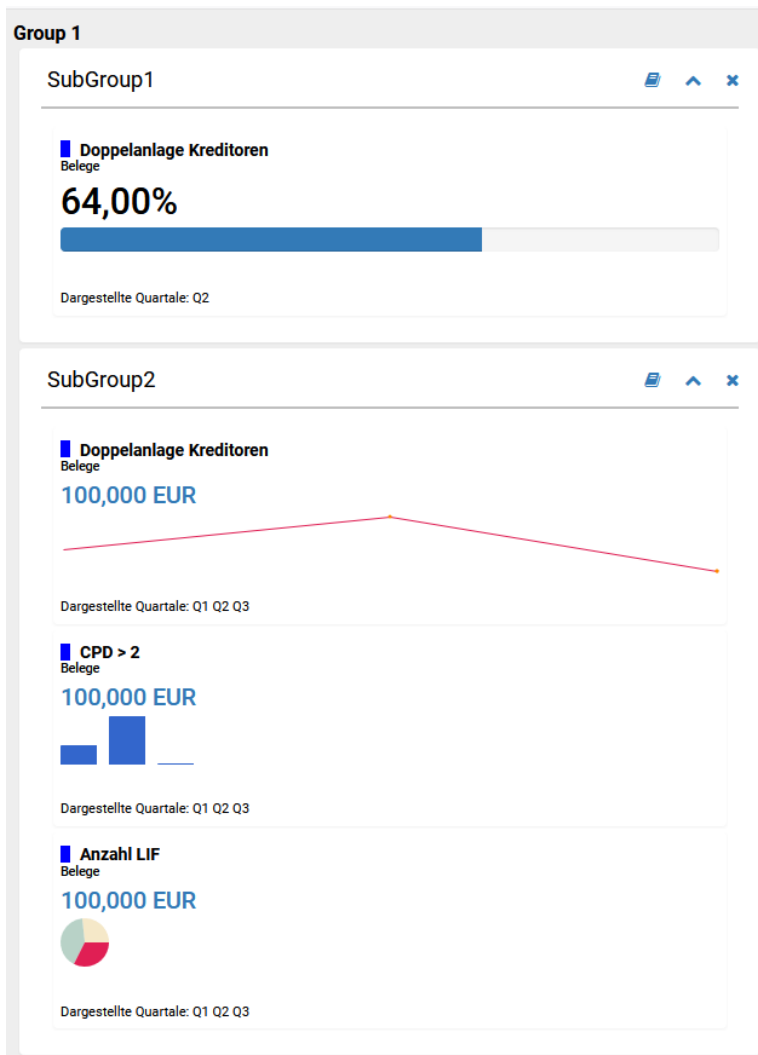


Wie kann eine Box für die Sub-Group Dashlet dargestellt werden?

Diese Anforderungen kann mit die folgende Code-Block dargestellt werden.

```
// KPI Box (Percentage chart)
dashlet = DashletManager.addDashlet("DashletKpiBox", "SubGroup1", "", true,
mbrKpi_2);
dashlet.setId("kpi_boxes1");
dashlet.addOption("child.box","true");
dashlet.addOption("box.info.icon","fa-book");
dashlet.addOption("box.info.dialog.show", "show");

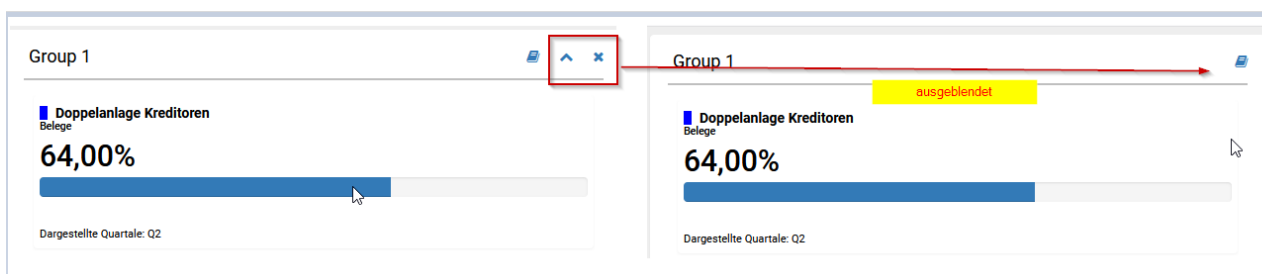
// KPI Box (Line, Bar and Pie chart)
dashlet = DashletManager.addDashlet("DashletKpiBox", "SubGroup2", "", true,
mbrKpi_1);
dashlet.setId("kpi_boxes2");
dashlet.addOption("child.box","true");
dashlet.addOption("box.info.icon","fa-book");
dashlet.addOption("box.info.dialog.show", "show");
```



Wie kann die Icon für Schließen und für Auf/Ein-Klappen ausgeblendet werden?

Diese Icons kann mit die folgende Optionen ausgeblendet werden.

```
dashlet.addOption("box.closelink.show", "hidden"); // hide close button
dashlet.addOption("box.collapselink.show", "hidden"); // hide collapse button
```



6.8 HTML Content Rendern

6.8.1 Einführung

Es gibt die Möglichkeit eigene HTML Content render zu können.

6.8.2 Optionen

Die eigene HTML Content könnte mit die folgende Option erstellt werden.

z.b. Eigene HTML Content bevor ein BarChart rendern.

Eigene Skripte erstellen

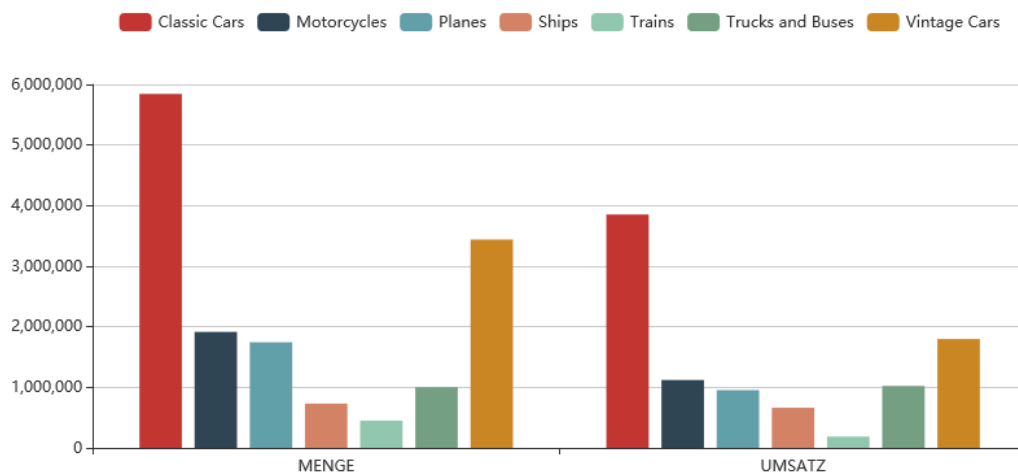
```
// Bar Chart Dashlet erstellen
var dashlet = DashletManager.addDashletEChart("bar", "Umsatz-Entwicklung", "EChart - Bar Chart", true, mbrUmsatz);

// HTML Content before the chart
dashlet.addOption("html.before.content", "<h1>This is a documentation title <span class=\"badge badge-info\">Use Bootstrap Components</span></h1>");
```

Umsatz-Entwicklung EChart - Bar Chart



This is a documentation title Use Bootstrap Components



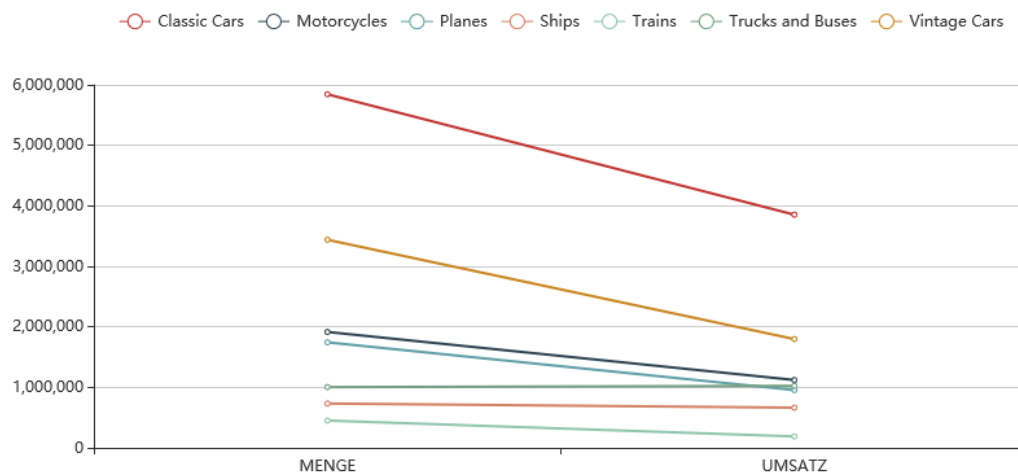
z.b. Eigene HTML Content nach ein Linie Chart rendern.

Eigene Skripte erstellen

```
// Dashlet erstellen
var dashlet = DashletManager.addDashletEChart("line", "Umsatz-Entwicklung", "EChart - Bar Chart", true, mbrUmsatz);

// HTML Content after the chart
dashlet.addOption("html.after.content", "<h1>This is a documentation title</h1> <div class=\"alert alert-success\">Lorem ipsum dolor si amet...</div>");
```

Umsatz-Entwicklung EChart - Bar Chart



This is a documentation title

Lorem ipsum dolor si amet...



6.9 Info-Box Anzeigen

6.9.1 Einführung

Es gibt die Möglichkeit eigene Info Box in das Dashlet zeigen zu können, z.b. ein Info über dashboard anzeigen.

6.9.2 Optionen

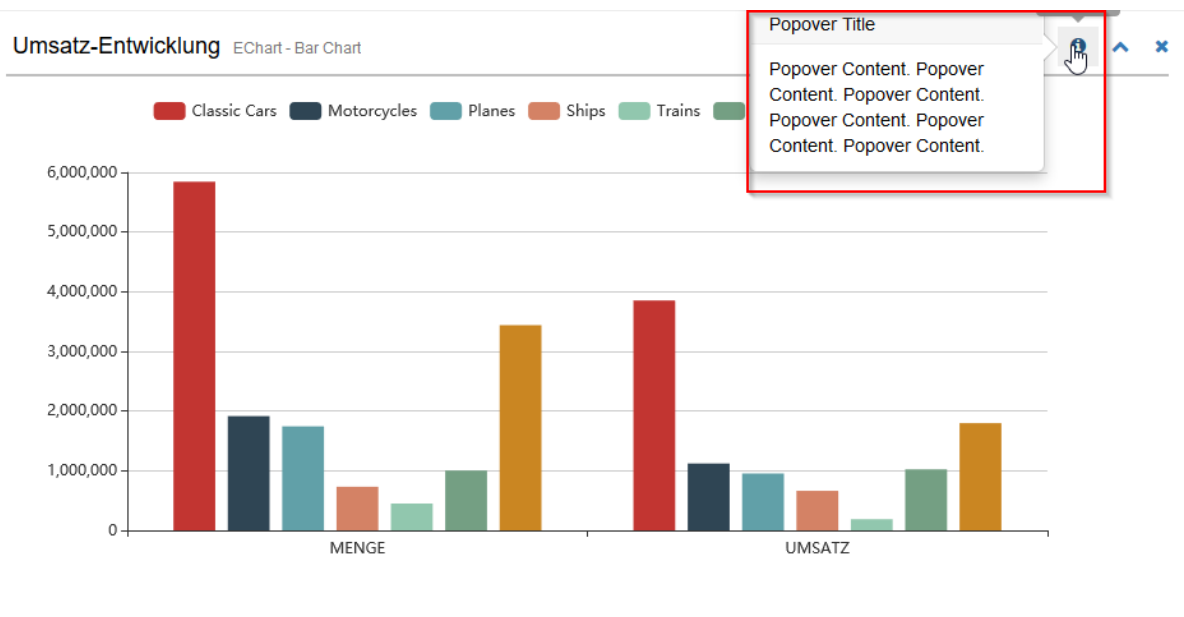
Die Info Box könnte mit die folgende Option erstellt werden.

Info Box als ein Pop-up anzeigen

Eigene Skripte erstellen

```
// Bar Chart Dashlet erstellen
var dashlet = DashletManager.addDashletEChart("bar", "Umsatz-Entwicklung", "EChart - Bar Chart", true, mbrUmsatz);

// Info box as pop-up
dashlet.addOption("box.info.icon", "fa-info-circle");
dashlet.addOption("box.info.tooltip.show", "show");
dashlet.addOption("box.info.tooltip.title", "Mehr Infos anzeigen");
dashlet.addOption("box.info.popover.show", "show");
dashlet.addOption("box.info.popover.placement", "left");
dashlet.addOption("box.info.popover.trigger", "focus");
dashlet.addOption("box.info.popover.offset", "50 50");
dashlet.addOption("box.info.popover.html", "true");
dashlet.addOption("box.info.popover.title", "Popover Title");
dashlet.addOption("box.info.popover.content", "<div style=\"width: 200px;\">Popover Content. Popover Content. Popover Content. Popover Content. Popover Content. Popover Content.</div>");
```

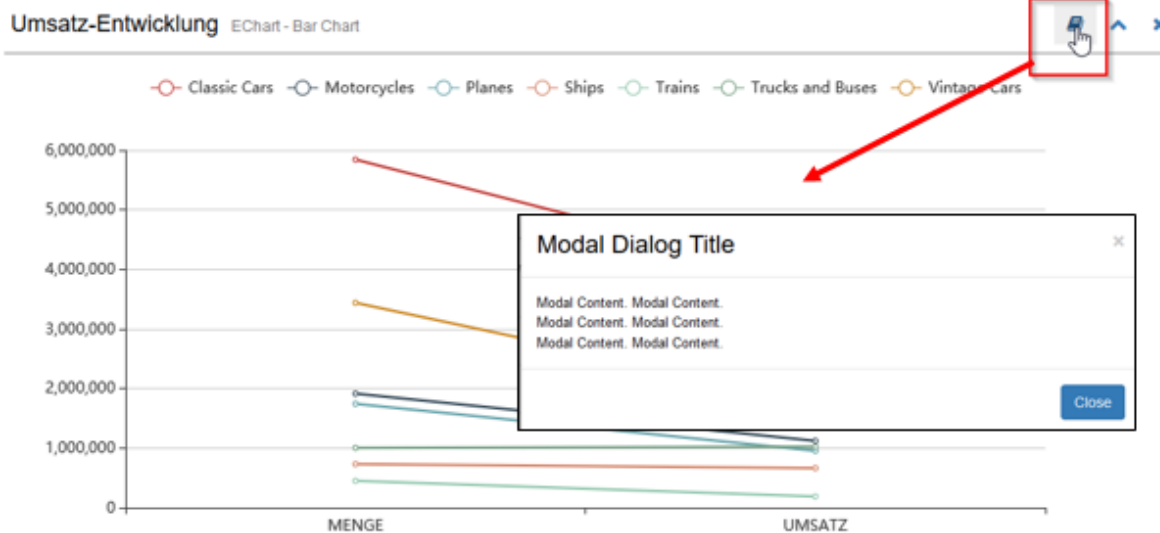


Info box als Modal dialog anzeigen

Eigene Skripte erstellen

```
// Dashlet erstellen
var dashlet = DashletManager.addDashletEChart("line", "Umsatz-Entwicklung", "EChart - Bar Chart", true, mbrUmsatz);

// Info box as Modal Dialog
dashlet.addOption("box.info.icon", "fa-book");
dashlet.addOption("box.info.dialog.show", "show");
dashlet.addOption("box.info.dialog.html", "true");
dashlet.addOption("box.info.dialog.title", "Modal Dialog Title");
dashlet.addOption("box.info.dialog.content", "<div style=\"width: 200px;\">Modal Content. Modal Content. Modal Content. Modal Content. Modal Content. Modal Content.</div>");
```



6.10 Kombination von mehreren Dashlets in ein Box

6.10.1 Einführung

Es gibt die Möglichkeit mehrere Dashboards in ein Box zu rendern.

6.10.2 Optionen

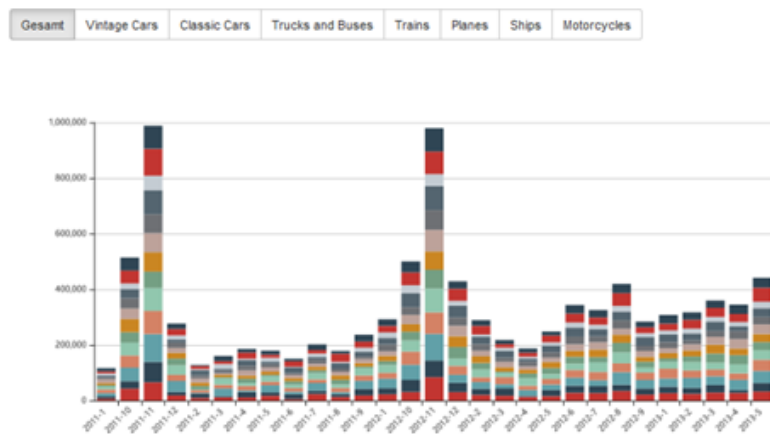
Diese Funktionalität ist mit die folgende Optionen gesetzt. In die Beispiel ist einzellene 'data Zoom' component benutzt für die beide Charts.


```
// chart1
var dashlet1 = DashletManager.addDashletEChart("bar","Sales - Productline","by
Vendor", true,mbr);
dashlet1.setId("chart1");

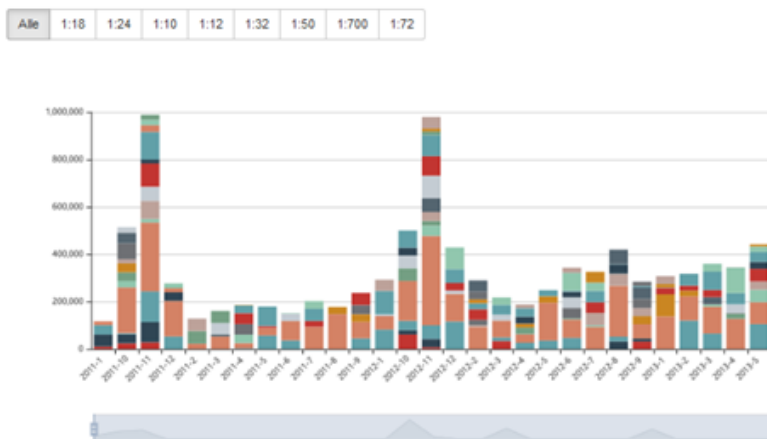
// chart2
var dashlet2 = DashletManager.addDashletEChart("bar","Sales - Productscale", "by
Country", true, mbr);
dashlet2.setId("chart2");

// Connect dashlets
dashlet1.addOption("childid","chart2");
dashlet2.addOption("chart.connect","chart1->chart2;chart2->chart1");
```

Sales - Productline by Vendor



Sales - Productscale by Country



6.10.3 FAQ

Wie kann ein DataZoom (Zeitscheine) dargestellt werden und bei alle Charts verbinden?

Wenn der verschiedene Charts mit connect Funktion verbunden sind, dann kann ein DataZoom (Zeitschiene) darstellt werden.

Schritt 1: Datazoom Option einblenden bei ein Chart1

```
dashlet1.addOption("chart.dataZoom.show", "true");
```

Schritt2 : DataZoom Option ausblenden bei ein Chart2

```
dashlet2.addOption("chart.dataZoom.show", "false");
```

Wie kann der Toolbox Optionen für verschiedene Charts benutzt werden?

Wenn der Toolbox Optionen mit Connected Charts verwenden wird, dann bitte die Custom-Toolbox Optionen nutzen.

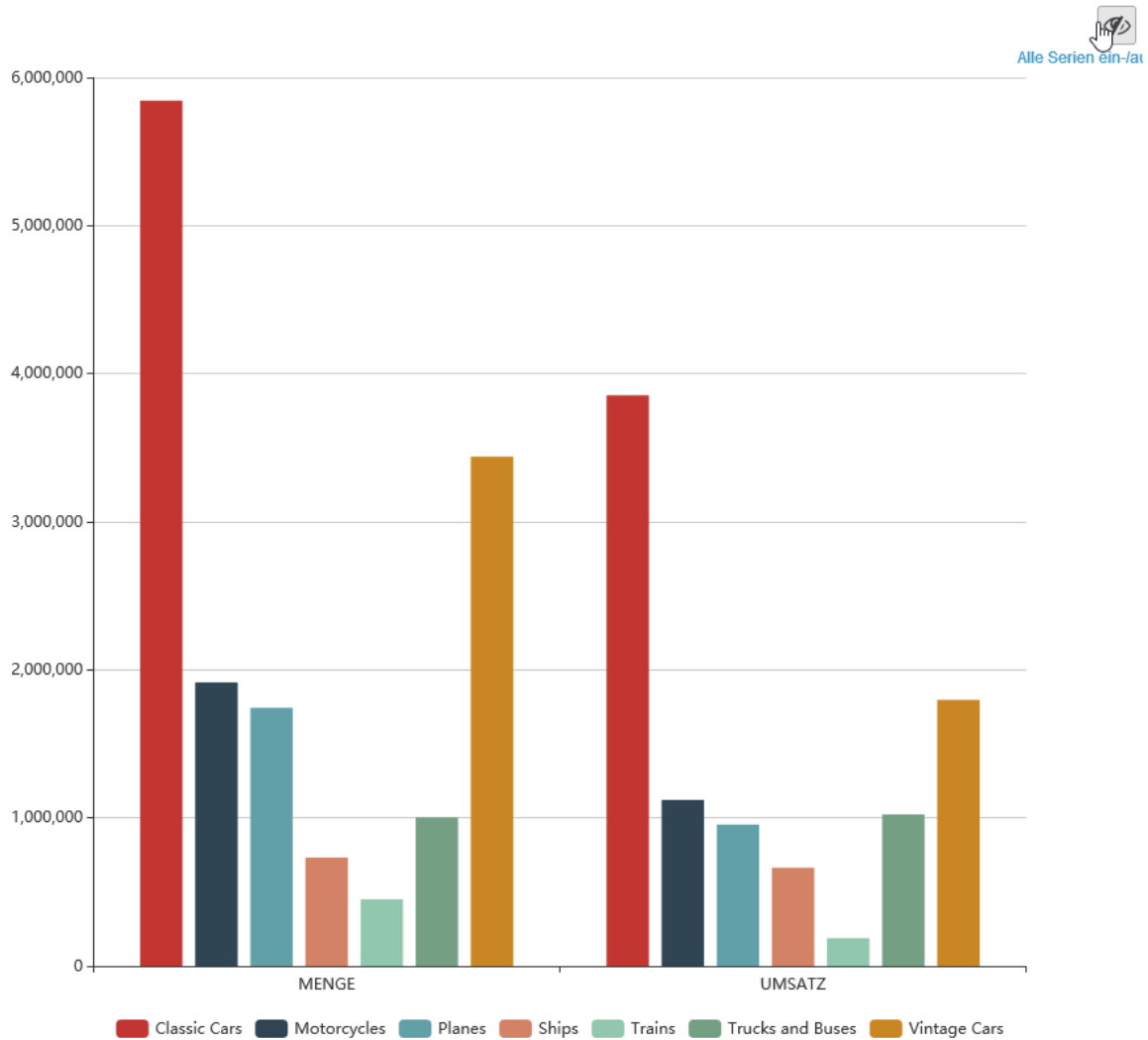
```
// Toolbox Options
dashlet.addOption("chart.toolbox.show", "true");
dashlet.addOption("chart.toolbox.feature.myToolRestore.show", "true");
dashlet.addOption("chart.toolbox.feature.myToolBarChart.show", "true");
dashlet.addOption("chart.toolbox.feature.myToolStackedChart.show", "true");
dashlet.addOption("chart.toolbox.feature.myToolLineChart.show", "true");
```

6.11 Legend - Toggle Visibility

6.11.1 Einführung

Es gibt die Möglichkeit, bei ECharts die Legend Wert ein und auszublenden mit ein Toolbox function.

Revenue-Trend EChart - Bar Chart (Toggle Visibility)



6.11.2 Toggle Visibility erzeugen

Die Legend Toggle Visibility Funktionalität ist erstellt mit die folgende Code Block.

Excel Download Link erstellen

```
//create Dashlet Manager
ExpertsDashlet.createDashletManager(reportContext, "DashletManager", "My First
Dashlet");

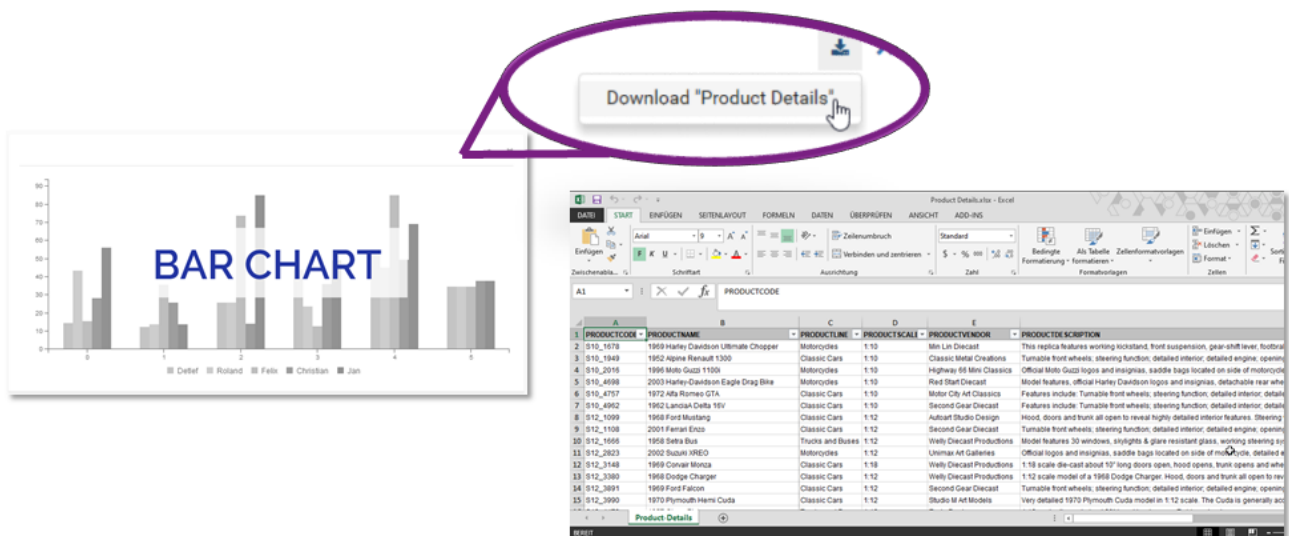
// create Dashlet
var dashlet = DashletManager.addDashletEChart("bar", "Revenue-Trend", "EChart - Bar
Chart (Toggle Visibility)", true, mbr);

// toolbox download button
dashlet.addOption("chart.toolbox.show", "true");
dashlet.addOption("chart.toolbox.feature.myToolToggleSeriesVisibility.show", "true");
```

6.12 Native Dokumente als Download anbieten

6.12.1 Einführung

Es gibt die Möglichkeit, weitere Dokumente (wie ein erzeugtes Excel Sheet) als Download-Link anzufügen.



6.12.2 Drilldown erzeugen

Die Excel Export Funktionalität ist erstellt mit die folgende Code Block.

Excel Download Link erstellen

```
//create Excel Output Manager
ExpertsOutput.createXlsxOutputManager(reportContext, "ExcelOutputManager", null,
false);

// excel download button
dashlet.addBoxOption("box.download.native.link",
generateDashletDownloadString(mbrCustomers));

// define functions
/*Script to generate the excel sheet link in the box*/
function generateDashletDownloadString(excelOutputManager)
{
    glJobID = 0;
    var appContext = reportContext.getAppContext();
    if(appContext != null && appContext.containsKey("REPORT_JOBID"))
    {
        glJobID = appContext.get("REPORT_JOBID");
    }

    var excel_filename = ExcelOutputManager.getFilename() +
ExcelOutputManager.getFilesuffix() + "." + ExcelOutputManager.getFileext();
    return "Download " + ExcelOutputManager.getFilename() + ";" + glJobID + ";" + excel_filename;
}
```

Die Excel Datei ist erstellt mit die folgende Code Block.

Excel Datei erstellen

```
/*Generate Excel File*/
area = excelOutputManager.addBufferArea("ClassicModels", mbrCustomers);
area.setAutoFilter(true);
area.setFreezePane(true);
area.setFreezePaneColumn(3);
excelOutputManager.renderPart();
os = area.getSheet();

excelOutputManager.writeFile(); // default the excel is created in the C:/Temp
folder
```

6.13 Unterberichte starten

6.13.1 Einführung

Es gibt die Möglichkeit aus einem Dashboard andere parametrisiert Berichte gestartet werden zu können.

Die Vorteil ist das die Clicks (Selections) in ein Dashlet ist gespeichert und weiterleitet zu die Bericht als parameter-Wert.

6.13.2 Unterberichte starten Skripte

Die Unterberichte starten Funktionalität ist erstellt mit die folgende Code Block.

Unterbericht starten Skript

```
// function to call the adhoc report from dashboard
customJS = "    function callAdhocBericht() {\n";
customJS += "        paramDateFrom = convertDate(startDate) ;\n";
customJS += "        paramDateTo = convertDate(endDate) ;\n";
customJS += "        try{openJobCreatorDialog([    {name:'--__report', value:'/
kurzcheck/AUDKurzcheckKategorie.rptdesign'},"+
                                "{name:'--__outputFormat',
value:'PDF,HTML'},"+
                                "{name:'--paramDateFrom',
value:paramDateFrom},"+
                                "{name:'--paramDateTo',
value:paramDateTo}]);"+
                                "}catch(e){};" ;
customJS += "    }    \n";

// render all the script in HTML File
jsDashlet.addOption("javascript", customJS + "\n\n");
```

Mehr Information zu die servBIRD-Aufruf für unter-Berichte zu starten kann hier gefunden werden: [Erweiterte Script-Funktionalitäten](#)

6.13.3

FAQ

Wie kann ein Datum Parameter weitergelitet werden?

Ein Date Parameter sollte in die Format yyyy-MM-dd Format weiterleitet werden.

Wie kann ein Datum-Zeit (DateTime) Parameter weiterleitet werden?

Ein Datetime Parameter sollte in die Format yyyy-MM-dd HH:mm:ss.SSS weiterleitet werden.

Wie kann ein Multiselect parameter weiterleitet werden?

Ein Multiselect parameter kann mit die folgende Skripte weiterleitet werden?

```
//Get the multiselect parameter seperated by pipe ('|') character
var countriesSelected = ""+params["paramCountry"].value.join("|")+"" ;

// function to call the adhoc report from dashboard
customJS += "      try{openJobCreatorDialog([" +
    "{name:'--__report', value:'/kurzcheck/" +
    AUDKurzcheckKategorie_PivotVersion.rptdesign'}, {name:'--__outputFormat',
    value:'PDF,HTML'}],"+
    "{name:'--paramCountry', value:"+countriesSelected+"});"+
    "}catch(e){};" +"}\n";

// render all the script in HTML File
jsDashlet.addOption("javascript", customJS + "\n\n");
```

Wie kann Bild von ein (oder mehr) Dashlets zu ein Bericht weiterleitet werden?

Das Bild von ein (oder mehr) Dashlets kann als folgendes in ein BIRT-Bericht weitergelitet werden.

Schritt 1 : In ein HTML-Dashlet kann ein Button erstellt werden.

- Der onclick-event für das Button würde ein Funktion "*exportDashlets*" anrufen. Diese Funktion ist automatisch in das Dashboard gerendert.
- Die Optionen für die Liste von Chart-Ids, die exportiert werden soll kann mit der Option "*exportChart.dashletId*" gegeben werden.
- Die servBIRD-Bericht Aufruf kann mit der Option "*exportChart.javascript*" definiert werden.

```
// HTML Dashlet
var jsDashlet1 = DashletManager.addDashlet("DashletHTML", "", "", false, null);

// HTML Script
var html = "<div class=\"btn btn-primary\" onclick=\"\" +
\"exportDashlets();\"+
\">Bericht</div>\";
jsDashlet1.addOption("html", html);

// Export options
jsDashlet1.addOption("exportChart.dashletId", "chart1;table1;leaflet1"); // Semi-
colon seperated list
jsDashlet1.addOption("exportChart.javascript",
"openJobCreatorDialog( [
"                                {name:'--__report',      \n" +
Export_Report.rptdesign'}      \n" +
"                                ,{name:'--__outputFormat', value: 'PDF,HTML'}
\n" +
"                                ,{name:'--paramImageMap',  value:
leaflet1_img.split(',')[1] }   \n" +
"                                ,{name:'--paramImageChart', value:
chart1_img.split(',')[1] }      \n" +
"                                ,{name:'--paramImageTable', value:
table1_img.split(',')[1] }      \n" +
"                                ]); ");
```

Schritt 2 : Der BIRT-Bericht sollte definiert werden mit diese Parameters.

- Image-Element sollte definiert werden in der Report, und der Expression ist als folgendes:

```
importPackage(Packages.org.apache.commons.codec.binary);  
this.data=Base64.decodeBase64(params["paramImageMap"].value);
```

Edit Image Item

Select Image from: ☐ URI
☐ Image file in shared resources
☐ Embedded image
☒ Dynamic image

Retrieve image from data set (databases)

Enter dynamic image expression:

`importPackage(Packages.org.apache.commons.codec.binary);this.data=Base64.decodeBase64(params["paramImageMap"].value);` Select Image Data...



Beim Leaflet-Dashlet, bitte bei der folgende Option aufpassen.

```
// Map-Starteinstellungen setzen  
dashlet.addOption("leaflet.map.tileLayer", "https://  
{s}.tile.openstreetmap.org/{z}/{x}/{y}.png");
```

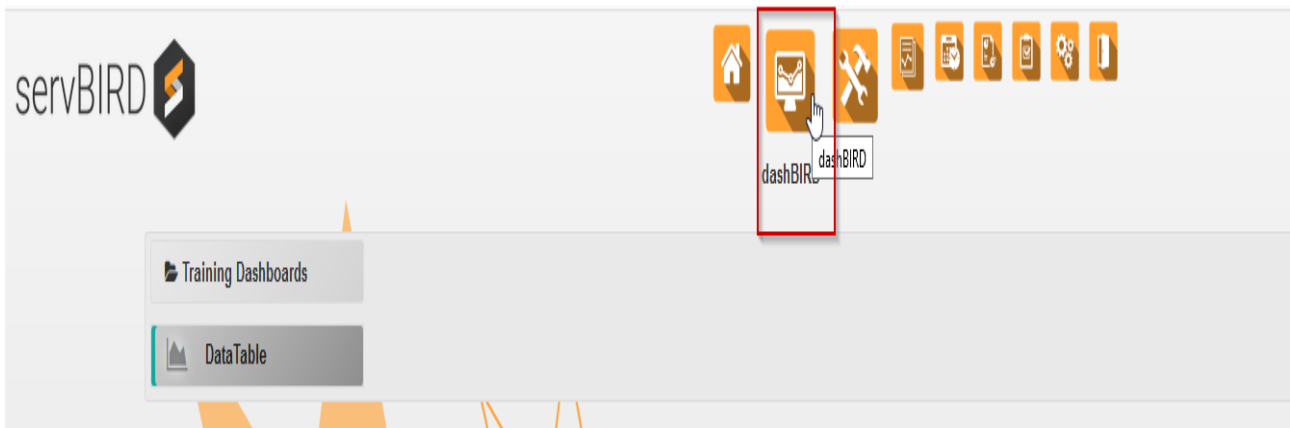

7 Bereitstellung und Weiterführendes

7.1 Grundlagen in der servBIRD Administration

7.1.1 Einführung

Das Dashboard kann in **servBIRD** hochgeladen werden und die Rechte zuweisen werden und auch noch die Pre-generation Schedules gesetzt werden.

In **servBIRD** es gibt ein Ansicht für Dashboards, wo die alle zugewiesene Dashboards angezeigt werden können.



Weitere Dokumentation

Weitere Information finden Sie unter <https://confluence.tradui.net/serv312/de/benutzerhandbuch/dashbird>

7.2 Bereitstellung/Deployment

7.2.1 Einführung

Das Dashboard können Sie von den Menüpunkt **Cockpits/Dashboards** unter **BIRD Interactives** in **Administrationsbereich** hochladen.

Das Dashboard sollte mit die Dateiname *.dashboard.rptdesign* enden, sodass es kann in servBIRD hochgeladen werden.

Die Vorausforderung sind das die folgende Properties sind für das Dashboard gesetzt:

1. Es sollte ein Title haben. Das Dashboard ist mit diese Namen in servBIRD angezeigt werden.
2. Es sollte ein User-Property 'SERV.INTERACTIVE.GROUP' enthalten. Das Dashboard ist in diese Gruppe in servBIRD angezeigt werden.
3. Es sollte ein User-Property 'SERV.INTERACTIVE.COCLOR' enthalten. Das Dashboard-Symbol ist mit diese Farbe angezeigt werden.

Als Optionen kann die folgende Schritte gemacht werden.

1. Das Icon-Image sollte die Namen als 'interactive.iconimage.jpg' oder 'interactive.iconimage.png' oder 'interactive.iconimage.gif' haben.

2. Das Vorschaubild sollte die Namen als 'interactive.thumbnail.jpg' oder 'interactive.thumbnail.png' oder 'interactive.thumbnail.gif' haben.

Das Dashboard und die Images sollte in ein zip-Archivs dargestellt.

Die andere Ressourcen sollte in das 'Resources' Verzeichnis hochgeladen werden.

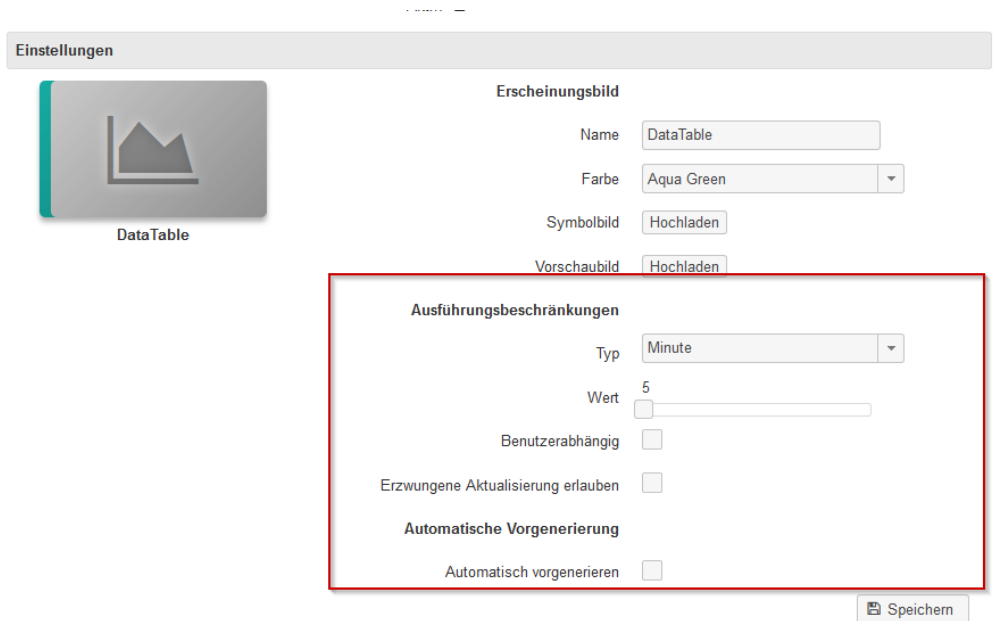
Weitere Dokumentation

1. <https://confluence.tradui.net/serv312/de/handbuch-fuer-berichtsentwickler/interaktive-berichte-bird-interactives/dashboards-entwickeln-fuer-dashbird>.
2. <https://confluence.tradui.net/serv312/de/administrationshandbuch/administrationsportal/bird-interactives/cockpits-dashboards>.

7.3 Caching und Vor-Generierung

7.3.1 Einführung

servBIRD bietet eine Möglichkeit das Caching Interval zu setzen und dazu noch eine Vor-Generierung für das Dashboard zu machen. Die Vor-Generierung ist besonders interessant im Falle wo das Dashboard lange Zeit benötigt beim Ausführen. Die beide Funktionalität kann der Nutzer in der Einstellungs- Seite konfigurieren.



Einstellungen

Erscheinungsbild

Name: DataTable

Farbe: Aqua Green

Symbolbild: Hochladen

Vorschaubild: Hochladen

Ausführungsbeschränkungen

Typ: Minute

Wert: 5

Benutzerabhängig: ☐

Erzwungene Aktualisierung erlauben: ☐

Automatische Vorgenerierung

Automatisch vorgenerieren: ☐

Speichern

Caching (Ausführungsbeschränkungen)


Mit dieser Konfiguration kann der Nutzer den Caching-Zeitraum für das Dashboard konfigurieren, d.h. das Dashboard ist aus dem Cache geladen, wenn der gleiche Parameter Set verwendet wird. Diese Funktionalität spart die Systemressourcen und die Ausführungszeit für das Dashboard.

Vor-Generierung

Durch Aktivieren des Kontrollkästchens 'Automatisch vorgenerieren', werden die Einstellungen angezeigt. Die Zeitraum, Interval und das Parameter Sets könnte erstellt würden.

Automatische Vorgenerierung


Automatisch vorgenerieren ☒

Zeitplan beginnen am * 

Typ * ▼

Wert *

Automatisch archivieren ☐


Vorgenerierung starten 

Weitere Dokumentation

Mehr Information finden Sie unter <https://confluence.tradui.net/serv312/de/administrationshandbuch/administrationsportal/bird-interactive/cockpits-dashboards>.

7.4 Rollen und Rechte

7.4.1 Einführung

Das Dashboard kann für ein Benutzer zugewiesen sein oder für ein Rolle. Die Zuweisung von Rechten kann im Administrationsbereich den Menüpunkt **Rollen** unter **Nutzer** erfolgen. Mit den Dashboard Button , kann die ausgewählte Dashboards zu ein Nutzer oder Rolle zugewiesen werden.

<

7.4.2 Weiter Dokumentation

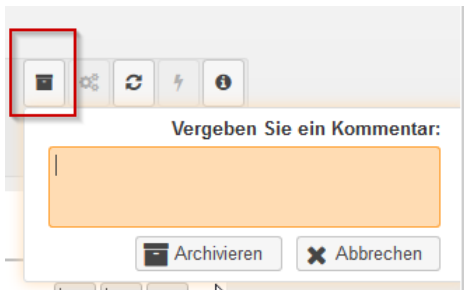
Weitere Dokumentation finden Sie unter <https://confluence.tradui.net/serv312/de/administrationshandbuch/administrationsportal/benutzerverwaltung/rollen-und-rechte>

7.5 Snapshots/Archive

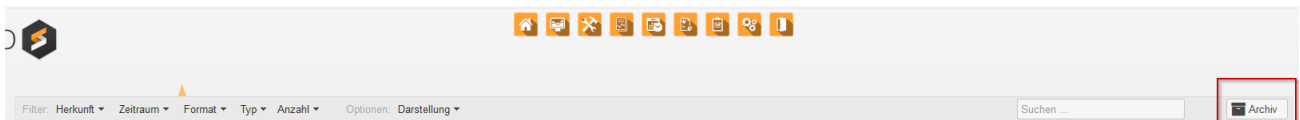
7.5.1 Einführung

Nach die Ausführung eines Dashboards, **servBIRD** bietet auch die Möglichkeit, das zuletzt generierte Dashboard zu archivieren. Hier sind das Dashboard und dazu gehörende Libraries in ein Zip Format gespeichert.

Wenn das Dashboard ausgeführt ist, dann ist der 'Archiv' Button (oben Rechts) aktiviert (abhängig ob der Archive-Funktionalität für das Dashboard aktiviert ist oder nicht). Wenn dieser Button gedrückt wird, öffnet sich ein Panel in dem Sie ein Kommentar schreiben können, und mit drücken auf des Buttons Archivieren ist ein Archive erstellt.



Die alle Archivierte-Dateien kann der Nutzer unter 'Fertige Dokumente' Ansicht wechseln, und dann die Button Archiv (rechts am Toolbar) wählen.



In die Liste von alle Archivierte-Dateien, ist die Funktionen wie die Information Anzeigen, herunterladen die archivierten-Dateien u.s.w. verfügbar.

Weitere Dokumentation

Weitere Informationen zum Archiv finden Sie unter <https://confluence.tradui.net/serv312/de/benutzerhandbuch/dashbird>.