

A L P A N A



[code**192**]



Alpana

3.1.x — Last update: 2019/08/30

Code 192

Table of Contents

1. About Alpana	10
1.1. Components of the application	11
1.2. License Portal and Installers Download	12
1.3. Release Notes	13
1.3.1. Upgrading Alpana from v3.0 to v3.1	14
1.3.2. v3.1.0.....	16
1.3.3. v3.1.2.....	18
2. Alpana Designer	20
2.1. Installation.....	21
2.1.1. Pre-Requisites	22
2.1.2. Installing Alpana Designer	23
2.1.3. Anti-virus and Firewall set up	28
2.1.4. Upgrading Alpana Designer	29
2.1.5. Uninstalling Alpana Designer	30
2.2. Troubleshooting	31
2.2.1. Finding error logs.....	32
2.3. Connecting to Data	35
2.3.1. Alpana data model	36
2.3.1.1. Note on data storage	38
2.3.1.2. Redundant Connections.....	39
2.3.2. The data Buffer	45
2.3.2.1. Bufferized Connection types : What gets bufferized ?	46
2.3.2.2. Data Flow : How bufferization happens ?	47
2.3.2.3. Buffer database setup.....	49
2.3.2.4. Buffer Refresh settings	50
2.3.2.5. refreshing file-based connections	52
2.3.3. Creating and Managing Connections	53
2.3.3.1. Create a Connection	54
2.3.3.2. Edit a Connection	55
2.3.3.3. Delete a Connection	56
2.3.4. Connection types	57
2.3.4.1. SQL Server.....	58
2.3.4.2. Excel	62
2.3.4.3. CSV.....	66
2.3.4.4. Wonderware Historian	68
2.3.4.5. Web.....	69
2.3.4.6. SQLite	71
2.4. Wonderware Historian	73
2.4.1. Connecting to Wonderware Historian	74

2.4.1.1. Version compatibility	75
2.4.1.2. Important pre-requisite.....	76
2.4.1.3. Selecting Connection Type	77
2.4.1.4. Login to a Wonderware Historian server	78
2.4.1.5. Test connection	80
2.4.1.6. Data refresh and Buffer	82
2.4.2. Creating Queries to Wonderware Historian	83
2.4.2.1. Picking Tags	84
2.4.2.1.1. Browsing for Tags	85
2.4.2.1.2. Selecting Tags.....	87
2.4.2.2. Configuring the Historian Query	89
2.4.2.2.1. Going to Query Configuration	90
2.4.2.2.2. Query Type	91
2.4.2.2.3. Time Settings	92
2.4.2.2.3.1. Static Time Settings	93
2.4.2.2.3.2. Parameter Based Settings.....	96
2.4.2.2.4. History Values : query parameters.....	98
2.4.2.2.5. History Values: query Format	102
2.4.2.2.6. State Statistics : query parameters	103
2.4.2.3. Renaming the Query Table	105
2.4.3. Wonderware Historian query result	106
2.4.3.1. Connection Schema.....	107
2.4.3.2. Important : data limits	109
2.5. Transforming Data	110
2.5.1. Introduction.....	111
2.5.1.1. Learning Pre-Requisites	112
2.5.1.2. What are Data Sources.....	113
2.5.1.3. Content : Data Preparation Activities	114
2.5.2. Creating and Managing Data Sources.....	115
2.5.2.1. Create a Data Source	116
2.5.2.2. Browsing Data Sources to Edit	117
2.5.2.3. Rename a Data Source.....	118
2.5.2.4. Delete a Data Source	119
2.5.2.5. Duplicate a Data Source	120
2.5.3. Export / Import Data Sources	121
2.5.3.1. What is a Data Source Export.....	122
2.5.3.2. Export a Data Source to a file	123
2.5.3.3. Import a Data Source from a file	124
2.5.3.4. Publish a Data Source to Alpana Server	126
2.5.3.5. Import a Data Source from Alpana Server.....	128
2.5.4. Fetching Data	129
2.5.4.1. using the Data Preview	130

2.5.4.2. getting and browsing the Connection schema	132
2.5.4.3. fetching data from a data Table	135
2.5.4.4. fetching data from a data View.....	136
2.5.4.5. fetching data from a Stored Procedure.....	137
2.5.4.6. fetching data from a Table-Valued Function	142
2.5.4.7. Removing a table from the Data Source.....	146
2.5.5. Custom Query Tables	147
2.5.5.1. Language	148
2.5.5.2. Creating a Custom Query Table.....	149
2.5.5.3. Updating a Custom Query Table	151
2.5.5.4. Using Parameters	152
2.5.6. Merging Data	154
2.5.6.1. Creating Joins.....	155
2.5.6.2. Configuring Joins	157
2.5.6.3. Updating Joins	162
2.5.7. Filtering Data	163
2.5.7.1. Opening the Filter Editor	164
2.5.7.2. Managing Filter Conditions	165
2.5.7.3. Configuring Filter Conditions.....	166
2.5.7.3.1. Filtering string fields	167
2.5.7.3.2. Filtering number fields	169
2.5.7.3.3. Filtering date fields	172
2.5.7.4. Limiting dataset size	176
2.5.8. Calculating Expressions.....	177
2.5.8.1. Opening Expression Editor	178
2.5.8.2. Managing Expressions.....	179
2.5.8.3. Expression properties	181
2.5.8.4. Adding a constant.....	182
2.5.8.5. Adding a field.....	184
2.5.8.6. Adding a function	187
2.5.8.7. Removing an existing item	191
2.5.8.8. Copying and Pasting an existing item	192
2.5.8.9. Aggregated Expressions	193
2.5.9. Expressions Troubleshooting	196
2.5.9.1. Pitfall : Incorrect result in Expression with division	197
2.5.9.2. Pitfall : Calculating ratios, averages,	199
2.5.9.3. Error when mixing aggregate fields and non-aggregate fields	202
2.5.9.4. Error when wrapping aggregate fields inside aggregate functions	203
2.5.10. Value-based Settings	204
2.5.10.1. Value-based Colors	205
2.5.11. Formatting Columns.....	208
2.5.11.1. Renaming Columns	209

2.5.11.2. Changing Column Type	210
2.5.11.3. Hiding Column	211
2.6. Configuring a Widget.....	213
2.6.1. Creating and Managing Widgets	214
2.6.1.1. Adding a Widget	215
2.6.1.2. Editing a Widget	220
2.6.1.3. Renaming a Widget	222
2.6.1.4. Deleting a Widget	223
2.6.2. Export / Import Widgets.....	225
2.6.2.1. What is a Widget Export	226
2.6.2.2. Export a Widget to a file.....	227
2.6.2.3. Import a Widget from a file.....	228
2.6.2.4. Publish a Widget to Alpana Server.....	230
2.6.2.5. Import a Widget from Alpana Server	232
2.6.3. Widget Data Binding	234
2.6.3.1. using the Widget Preview	236
2.6.3.2. Troubleshooting Widget data	237
2.6.3.3. Editing Widget data binding	239
2.6.3.4. Selecting a Data Source	241
2.6.3.5. Configuring the aggregation of number fields.....	242
2.6.3.6. Configure the grouping of category fields.....	245
2.6.3.7. Binding a data field	248
2.6.3.8. Un-binding a data field.....	251
2.6.3.9. Re-Ordering data fields.....	252
2.6.4. Configure a sorting order	254
2.6.5. Widget-level Filtering	255
2.6.5.1. Opening the Filter Editor	256
2.6.5.2. Managing Filter Conditions	257
2.6.5.3. Filtering aggregated values.....	258
2.6.5.4. Configuring Filter Conditions.....	261
2.6.5.4.1. Filtering string fields	262
2.6.5.4.2. Filtering number fields	264
2.6.5.4.3. Filtering date fields	267
2.6.5.5. Limiting dataset size	271
2.6.6. General Widget Properties	272
2.6.6.1. Editing Widget Properties	273
2.6.6.2. Name, Header and Description	275
2.6.6.3. Container Appearance	278
2.6.6.4. Widget Menu items	280
2.6.6.5. Measure Formatting and Date Formatting	282
2.6.6.6. Multi-items Viewport	287
2.6.6.7. Animation	292

2.6.7. Interactions	293
2.6.7.1. Drill-Down	294
2.6.7.2. Link To URL	295
2.6.7.3. Master Filter	299
2.7. Widget Properties Reference	300
2.7.1. Cartesian Chart	301
2.7.1.1. Binding Data	302
2.7.1.2. Chart Types	318
2.7.1.3. Switching Chart Types	320
2.7.1.4. Combining Chart Types	321
2.7.1.5. Series Settings	323
2.7.1.5.1. Series Name	324
2.7.1.5.2. Axis binding	325
2.7.1.5.3. Data Point information	326
2.7.1.5.4. Style and Appearance	328
2.7.1.6. Axes	332
2.7.1.6.1. X Axis	333
2.7.1.6.1.1. Strip Line	341
2.7.1.6.1.2. Paging / Scrolling	346
2.7.1.6.2. Y Axes	347
2.7.1.6.2.1. Strip Line	352
2.7.1.7. Legend	356
2.7.1.8. Managing Series Color	359
2.7.1.8.1. Chart value color binding	363
2.7.1.9. Trend Lines	367
2.7.2. Polar Chart	369
2.7.2.1. Binding Data	370
2.7.2.2. Chart Types / Draw Types	373
2.7.2.3. Switching Chart Types	375
2.7.2.4. Series Settings	376
2.7.2.4.1. Series Name	377
2.7.2.4.2. Style and Appearance	378
2.7.2.5. X Axis	380
2.7.2.6. Legend	381
2.7.3. Map	383
2.7.3.1. Maps format : GeoJSON	384
2.7.3.2. Choosing a map	386
2.7.3.3. Binding Data	389
2.7.3.4. Managing Colors	393
2.7.3.5. Data Point information	396
2.7.3.6. Runtime navigation	397
2.7.4. Grid	398

2.7.4.1. Binding Data	399
2.7.4.2. Layout and appearance	405
2.7.4.3. User interaction	408
2.7.4.4. Column names.....	414
2.7.4.5. Value display	416
2.7.4.6. Conditional Formatting.....	417
2.7.4.7. Grid KPI.....	423
2.7.4.8. HTML in the Grid	427
2.7.5. TreeMap	429
2.7.5.1. Binding Data	431
2.7.5.2. Managing Colors.....	440
2.7.5.3. Data Point information	442
2.7.6. Gauge.....	443
2.7.6.1. Binding Data	444
2.7.6.2. Value Display.....	450
2.7.6.3. Style and Appearance.....	454
2.7.7. Card.....	459
2.7.7.1. Binding Data	460
2.7.7.2. Value Display.....	464
2.7.7.3. Style and appearance	470
2.7.8. Proportion Charts.....	474
2.7.8.1. Binding Data	475
2.7.8.2. Switching Chart type	490
2.7.8.3. Data Point information	492
2.7.8.4. Legend	495
2.7.8.5. Managing Series Color	498
2.7.8.6. Interaction	502
2.7.8.7. Style and Appearance.....	503
2.7.9. Filter Widgets.....	505
2.7.9.1. ComboBox	506
2.7.9.1.1. Binding Data.....	507
2.7.9.1.2. Configuring behavior	508
2.7.9.1.3. Style and Appearance	510
2.7.9.2. CheckBox	511
2.7.9.2.1. Binding Data.....	512
2.7.9.2.2. Configuring behaviour	513
2.7.9.2.3. Style and Appearance	514
2.7.9.3. RadioButton.....	515
2.7.9.3.1. Binding Data.....	516
2.7.9.3.2. Configuring behaviour	517
2.7.9.3.3. Style and Appearance	518
2.7.9.4. RangeSlider	519

2.7.9.4.1. Binding Data.....	520
2.7.9.4.2. Configuring behaviour	522
2.7.9.4.3. Configuring style.....	523
2.7.9.5. DatePicker.....	525
2.7.9.5.1. Binding Data.....	526
2.7.9.5.2. Configuring behaviour	527
2.7.9.5.3. Configuring style.....	530
2.7.9.6. RangeNavigator.....	531
2.7.9.6.1. Binding Data.....	532
2.7.9.6.2. Configuring behaviour	534
2.7.9.6.3. Configuring style.....	535
2.7.10. Static Widgets	536
2.7.10.1. Image	537
2.7.10.1.1. bind image.....	538
2.7.10.1.2. Image Modes.....	539
2.7.10.2. Label	545
2.7.10.2.1. Configuration	546
2.7.10.3. Embed	547
2.7.10.3.1. Configuration	548
2.7.10.3.2. Note : Cross-origin resource sharing	549
2.8. Using Dashboard Parameters	550
2.8.1. What is a Parameter ?	551
2.8.2. Managing Parameters.....	552
2.8.2.1. Opening the Parameters Editor.....	553
2.8.2.2. Visualizing Parameter binding.....	556
2.8.2.3. Editing Parameter bindings.....	558
2.8.2.3.1. Necessary connections.....	559
2.8.2.3.2. Removing a connection	560
2.8.2.3.3. Adding a connection	563
2.8.2.4. Adding/Removing Parameter	568
2.8.2.5. Edit Parameter properties	569
2.8.2.5.1. Parameter Name	570
2.8.2.5.2. Value Type	571
2.8.2.5.3. Parameter Default Value	572
2.8.2.6. example : Multi-Data Source filtering	574
2.8.3. Widget as Master Filter	575
2.8.3.1. Configuring a Widget as Master	576
2.8.3.2. Setting a Widget Default Selection.....	578
2.8.3.3. Using filters at runtime	581
2.8.4. URL Parameters to receive values	584
2.8.4.1. Example	585
2.8.4.2. Reserved words.....	586

2.8.5. Parameter usage	587
2.8.5.1. Dynamic calls to Stored Procedures or Table-Valued Functions	588
2.8.5.2. Dynamic calls to Custom Queries	589
2.8.5.3. Dynamic Expression Values	590
2.8.5.4. Dynamic Data Source filters conditions	591
2.8.5.5. Dynamic Widget filters conditions	592
2.8.5.6. Dynamic Historian Queries	593
2.8.6. Widget-specific binding	594
2.8.6.1. Grid conditional formatting	595
2.9. Dashboard Settings and Publication	596
2.9.1. Dashboard Preview	597
2.9.2. Create and Export / Import Dashboards	598
2.9.2.1. Creating a New Dashboard	599
2.9.2.2. What is a Dashboard file	600
2.9.2.3. Save a Dashboard as file	601
2.9.2.4. Open a Dashboard file	602
2.9.2.5. Publishing a Dashboard to Alpana Server	604
2.9.2.6. Importing a Dashboard from Alpana Server	606
2.9.3. Dashboard Layout	608
2.9.3.1. Layout size and Responsiveness	609
2.9.3.2. Moving Widgets	614
2.9.3.3. Re-sizing Widgets	615
2.9.4. Dashboard-level Settings	616
2.9.4.1. Dashboard Title	617
2.9.4.2. Dashboard Theme and Background	618
2.9.4.3. Widget Auto-refresh	622
2.9.4.4. Disabling Comments	624
2.9.5. Designer-level Settings	625
2.9.5.1. Login to Alpana Server	626
3. (documentation in progress) Alpana Server	628
3.1. About Alpana server	629
3.1.1. Architecture	630
3.1.2. Multitenancy	633
3.2. Installation	635
3.2.1. Pre-Requisites	636
3.2.1.1. IIS Configuration pre-requisite	638
3.2.2. Installing	639
3.2.3. Note : Upgrading from v3 to v3	648
3.2.4. First login	649
3.2.5. Un-Installing	652
3.3. License Management	653


3.3.1. License Activation	654
3.3.2. Host-level Features.....	658
3.3.3. License Deactivation.....	659
3.4. Troubleshooting	661
3.4.1. Installation Troubleshooting	662
3.4.2. Log Files.....	664
3.4.3. Website Troubleshooting	665
3.5. Maintenance and Migration	667
3.5.1. Backup.....	668
3.5.2. Migration.....	670
3.6. Tenant Management	672
3.6.1. Create Tenant by Host Admin	673
3.6.2. Create Tenant using Email Address	675
3.6.3. Tenant Features.....	677
3.7. Categories Management	678
3.7.1. Creating Categories	679
3.8. Settings.....	681
3.8.1. Settings-Tenant Management	682
3.8.2. Settings-User Management.....	683
3.8.3. Settings- Security	684
3.8.4. Setting-Email (SMTP)	686
3.8.5. Settings-Other Settings.....	687
3.8.6. Settings-Appearence (Use Custom CSS).....	688
3.8.7. API Origins Whitelist	692
3.9. Manage Users and Permission.....	694
3.9.1. Create User	695
3.9.2. Edit User.....	698
3.9.3. Delete User.....	703
3.9.4. Excel Operations.....	706
3.9.4.1. Export users details	707
3.9.4.2. Import User Details	709
3.10. Manage Roles	711
3.10.1. Create Role.....	712
3.10.2. Edit Roles	716
3.10.3. Delete Role	720
3.10.4. Assign Role to User	722
3.11. API.....	724
3.11.1. General API help and troubleshooting.....	725
3.11.2. Authentication to the API.....	727
3.11.3. Remote Actions.....	729
3.11.3.1. Refreshing / Updating Filters.....	730

4. Alpana Mobile	732
4.1. Installation.....	733
4.1.1. How to get Alpana Mobile	734
4.1.2. Setup	735
4.2. Usage	736
4.2.1. Login.....	737
4.2.2. Tenancy.....	738
4.2.3. Dashboard Page	739
5. Alpana Update File Dependency.....	740
5.1. Installation.....	741
5.1.1. Pre-Requisistes	742
5.1.2. Installing	743
5.2. Running the application.....	746
5.3. Settings.....	747
5.4. Connecting To Alpana	749
5.5. Jobs	751
5.5.1. Creating a Job	752
5.5.2. Configuring a Job.....	753
5.5.2.1. Target Dependency	754
5.5.2.2. Source File	756
5.5.2.3. Update Mode	757
5.5.3. Running jobs.....	759
5.5.4. Deleting a Job.....	760
5.6. Troubleshooting	761

1. About Alpana

Let your Data turn into Insights and put the power into your hands to take Data driven decisions !

Alpana Platform provides real-time access to your data and lets users analyze complex information through graphical representations that are easy to understand.

Alpana is published by [\[code192\]](#) 

Last modified: 2019/06/13

1.1. Components of the application

Alpana Platform is composed of :

- [Alpana Designer](#) : the development tool for creating Data Sources, Widgets, Dashboards
- [Alpana Server](#) : the website deployed for sharing content
- [Alpana Mobile](#) : the mobile app used for viewing on mobile devices
- [Alpana Update File Dependency](#) : external tool for live update of file-based connections.

Last modified: 2019/08/12

1.2. License Portal and Installers Download

Web License Portal

The web license portal is located at :

<https://secure.softwarekey.com/solo/customers/Default.aspx?AuthorID=5159868>

This is where you activate and manage your license with [code192] Alpana software editor.

Your login and password was sent to you on license delivery.

Downloading Installers

Installers are available in the License Portal.

In order to download them, please follow the below instructions.

Login to the Web License Portal with your Customer ID.

In the home page, click “Downloads” :



Below your license ID, click *Download* :



This downloads .zip file containing all installers and documentation allowed by your license.

Last modified: 2019/06/17

1.3. Release Notes

Last modified: 2019/06/17

1.3.1. Upgrading Alpana from v3.0 to v3.1



Alpana v3.1 requires .NET Framework 4.8, which was incompatible with v3.0 Alpana Server installers.



You can install .NET Framework 4.8 while Alpana v3.0 is running.

To upgrade existing v3.0 installations, you can use the following procedure :

1 – Backup

As with any maintenance operation, you may wish to do a [backup](#) before the upgrade.

At any rate, you must make sure you have all information to install again :

Folder path to the website



It must be the same during the upgrade, otherwise you will not be able to keep your configuration and dashboards on upgrade !

Network configuration

DNS or IP addresses and ports of Frontend and Backend

SSL information



It must be the same during the upgrade, otherwise your users won't be able to reach the website on the same address again.

SQL database connection



It must be pointing to the same database during the upgrade, otherwise you will not be able to keep your configuration and dashboards on upgrade !

Other IIS configuration

If you tinkered with the IIS configuration manually, you must also back it up.

2 – Install new .NET Framework



Installing this component requires to restart the machine.

.NET Framework 4.8 can be found [here](#) then download the “Runtime”. Alternately, an offline installer exists on the same page .

3 – Un-install

Use Windows Control Panel to un-install previous versions of Alpana Server and Designer.

4 – Install

Run the new installers and perform the installation procedures again for [Alpana Server](#) and [Alpana Designer](#).

Last modified: 2019/08/29

1.3.2. v3.1.0

Release notes of items completed between Alpana and v3.0.2.2 and v3.1.0.7

Upgrading

New Pre-Requisites

Alpana Server and Designer v3.1.0.7 now require .NET Framework 4.8

Alpana Server install upgrade from v3

To upgrade Alpana Server from v3.0.2.2, see [Upgrading Alpana from v3.0 to v3.1.0](#)

Main Features

Redundant Connections

A [Redundancy](#) mechanism has been implemented on SQL, Historian and Web connectors.

Alpana Update File Dependency

The tool [Alpana Update File Dependency](#) has been added and needs to be installed separately for live update of file-based connections.

Tasks

No other new features have been implemented in this version, as it is mostly a technology upgrade and bugfix release.

Issues

Bugs (44) that were fixed between Alpana v3.0.2.2 and v3.1.0.7 :

- ALP-I1694 – Export PDF : issue with Gauges
- ALP-I1693 – Can't export widgets on this dashboard
- ALP-I1678 – Mobile : DatePicker pop-up doesn't open
- ALP-I1671 – Server PDF exports: charts overflow at page right
- ALP-I1713 – Chart : random JS error when resizing browser

- ALP-I1717 – Views > Reset Parameter doesn't clear filter UI
- ALP-I1759 – traduction Française Paramètres SMTP
- ALP-I1763 – all license features assigned to new tenant
- ALP-I1773 – Excel Export not filtered when using custom query and relative dates
- ALP-I1787 – Chart X Axis labels overflow container in some cases
- ALP-I1722 – Designer UI : more explicit message when no permission to publish
- ALP-I1712 – IQ_UnableToRetrieveDataTable happened randomly on this dashboard
- ALP-I1702 – UI Parameter Editor issue when too many Parameters
- ALP-I1706 – Designer allow mark as Public even if impossible
- ALP-I1705 – Designer : copy-paste widget on wrong location
- ALP-I1704 – Grid : clicking header after link to url opens link
- ALP-I1701 – Issue when Expression name contains dots
- ALP-I1700 – Configuring Drill-Down and Master on same widget gives errors
- ALP-I1677 – RangeSlider with datetime : labels shifted
- ALP-I1669 – Cartesian chart javascript error
- ALP-I1659 – Chart : Y Axis auto-scale wrongly applied in mixed cases
- ALP-I1762 – error when selecting in this multi-selection Combobox
- ALP-I1768 – Parameter not set when no Listener present
- ALP-I1802 – Designer crash on copy-paste this Proportion Chart
- ALP-I1788 – Filtering based on WHERE condition on 2 datetime fields
- ALP-I1770 – can't replace custom Map
- ALP-I1774 – Designer crash when deleting Parameter after selecting condition
- ALP-I1775 – custom Choropleth Map empty
- ALP-I1716 – Reload Dashboard resets filter UI but doesn't clear filter
- ALP-I1777 – TreeMap tooltips style inconsistent and error-prone
- ALP-I1804 – NullReference while opening this dashboard
- ALP-I1801 – Erreurs sur requête Historian Wide Query avec SQL Français
- ALP-I1791 – Alpana Server setup fails when listing available IP
- ALP-I1731 – Designer: Auto-creation of parameters behaviour
- ALP-I1764 – Excel dependency not updated from file path
- ALP-I1827 – Designer : widgets toolbox : widgets names
- ALP-I1642 – Bubble Map : this custom map doesn't display shapes
- ALP-I1691 – Bubble Map error : e.map is not a function
- ALP-I1719 – Cartesian chart => Enable scrolling => scrollbar should be hidden
- ALP-I1750 – Tenants not renamed properly
- ALP-I1816 – DatePicker pre-selection marked incorrect when using Highlight Available
- ALP-I1723 – DatePicker : dates unavailable when using Highlight Available and Relative
- ALP-I1769 – Designer : Tab Headers style
- ALP-I1673 – Widgets won't refresh when resizing application

Last modified: 2019/08/29

1.3.3. v3.1.2

Release notes of items completed between Alpana v3.1.0.7 and v3.1.2.1

Alpana Update File Dependency has been upgraded from v1.0.0.3 to v1.0.0.4

Main Features

Date Time Picker

The Date Picker widget now supports [selecting time intervals](#).

Custom date formats

[Custom date formats](#) are now supported.

Solved Issues

The following are the main bugs that were fixed between Alpana v3.1.0.7 and v3.1.2.1 :

- ALP-I1861 – Server: dashboard background image : Cannot insert null value to the cache
- ALP-I1867 – Card error when null value present
- ALP-I1865 – JSON Connector error : The method or operation is not implemented
- ALP-I1838 – Historian redundancy not triggered
- ALP-I1868 – Choropleth Map with Calculation broken
- ALP-I1724 – RangeNavigator : wrong selection with Relative Date in this configuration
- ALP-I1834 – X Axis StripLine behaviour incorrect when duration too short
- ALP-I1830 – Bubble Map doesn't support geojson Point
- ALP-I1871 – Date Picker with range : selection maintained after clearing filter
- ALP-I1729 – Server : notification icon doesn't display when notifications are present
- ALP-I1841 – Update Dependency : error when trying to update Image
- ALP-I1843 – disable X Axis Stripline with X Axis type Category
- ALP-I1828 – Chart tooltip text overflows background
- ALP-I1829 – Bubble Map tooltip has no background

Known issues

The following are the main bugs that are known and still present at the time of release of v3.1.2.1 :

- ALP-I1879 – Null reference exception when copy-paste Widget in Designer

- ALP-I1877 – Random IOException on Map : can't access file
- ALP-I1840 – “no such column” when re-opening certain v3.0 file-based dashboards in v3.1
- ALP-I1866 – Custom date format not applied in Chrome
- ALP-I1882 – Date Picker with time : display wrong when filter empty
- ALP-I1881 – Date Picker with time : can't select time when using relative date
- ALP-I1842 – Update Dependency : error when changing credentials
- ALP-I1736 – Server : pinned Chat disappears

Last modified: 2019/08/30

2. Alpana Designer

Drag and drop widgets into the interface and manipulate your data right away. The desktop application allows you full autonomy. Once the Dashboard ready, share with your colleagues through Alpana Server and start to collaborate.

Read below the documentation of Alpana Designer to learn more.

Last modified: 2019/05/29

2.1. Installation

Installation of Alpana Designer is light on resource and pretty straightforward.
Proceed to the next chapters to complete the installation.

Last modified: 2019/04/16

2.1.1. Pre-Requisites

This section explains the system requirements to run Alpana Designer.

Hardware Requirements

The following minimum hardware requirements are necessary to run Alpana Designer, but more is better for bigger projects :

- Processor : Dual Core 32-bit CPU (x86)
- Hard disk : 1 GB, preferably SSD
- RAM : 2 GB, preferably more
- Accelerated graphics



Note : Graphic acceleration

Alpana Designer is a WPF application that requires accelerated graphics for best performance.

On a physical host this is enabled, but on virtual machines this is a dedicated setting (in VmWare, under “Settings>Display>Accelerate 3D graphics”)

Software Requirements

The following minimum software requirements are necessary to run Alpana Designer:

- Operating System – Windows 7+, Windows Server 2012+
- .NET Framework 4.8 can be found [here](#) then download the “Runtime”. Alternately, an offline installer exists on the same page . **Installing this component requires to restart the machine.**
- [IIS Express](#)
- Browser : recent versions of Google Chrome, Firefox, Edge

Last modified: 2019/07/24

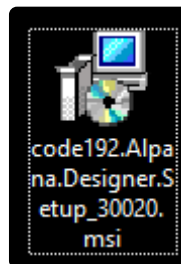
2.1.2. Installing Alpana Designer

Download installer

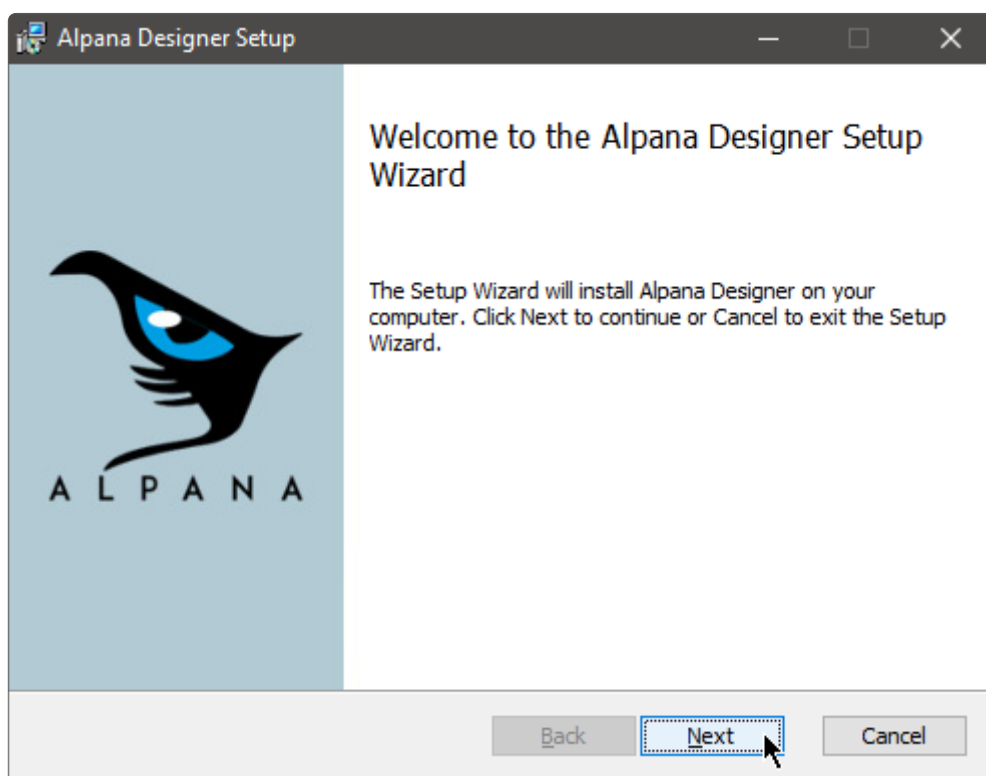
[See here to download the installer](#)

Run installer

Double-click the installer to start installation :

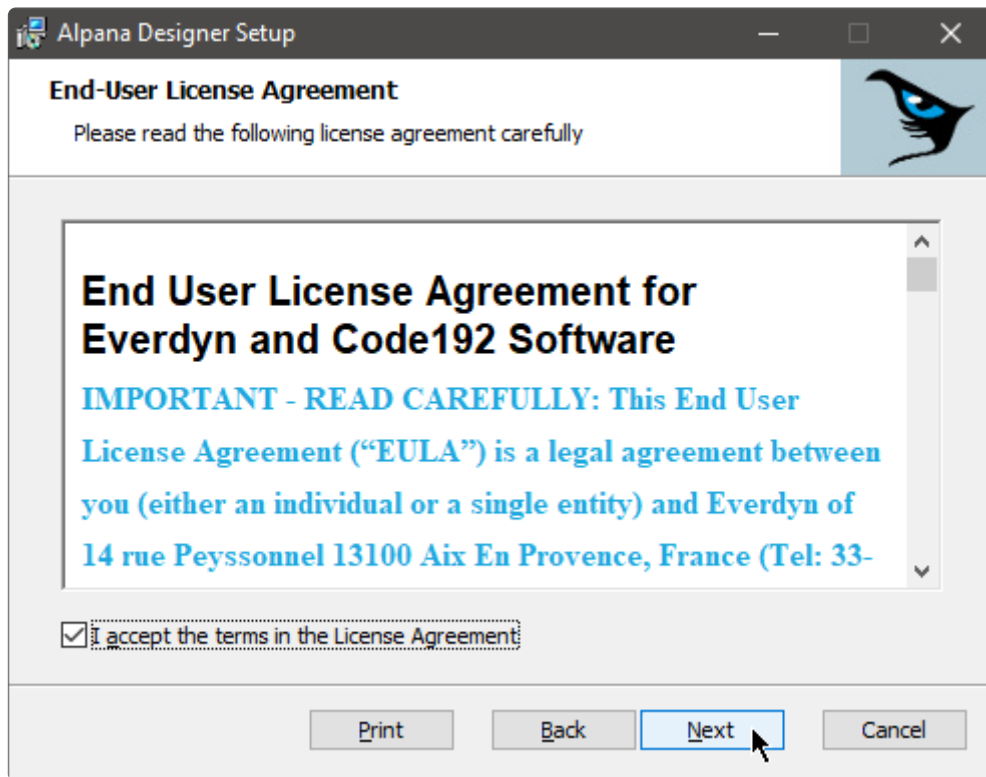


On the welcome screen, click *Next* to continue :



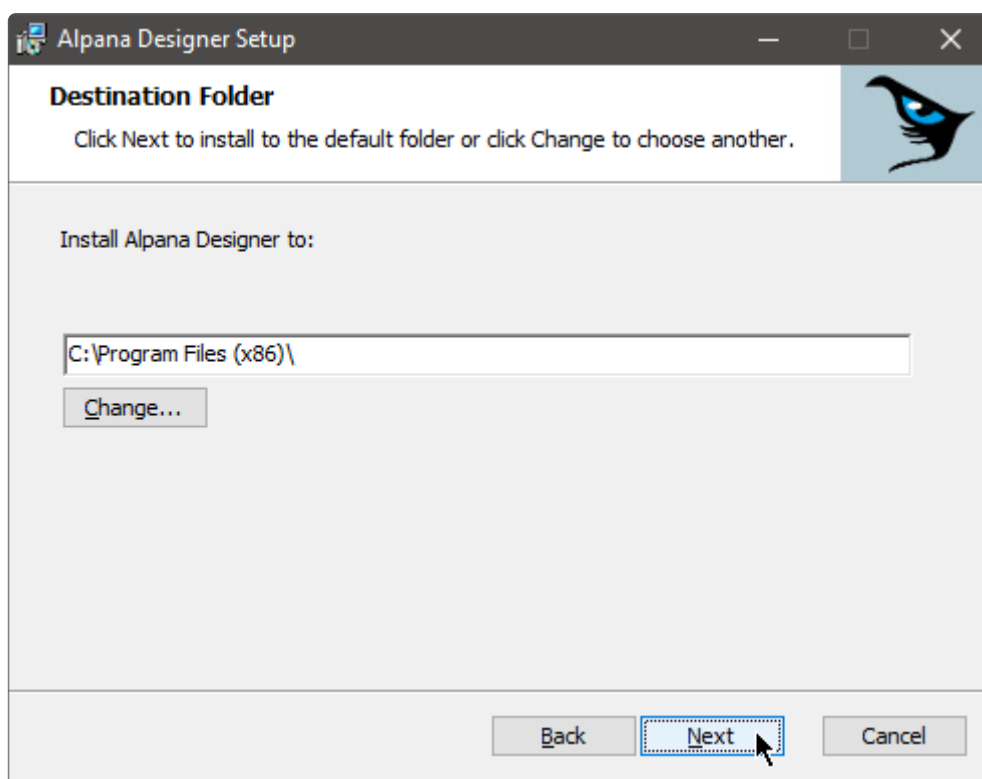
Accept EULA

Read and accept the license agreement, and click *Next* to continue :



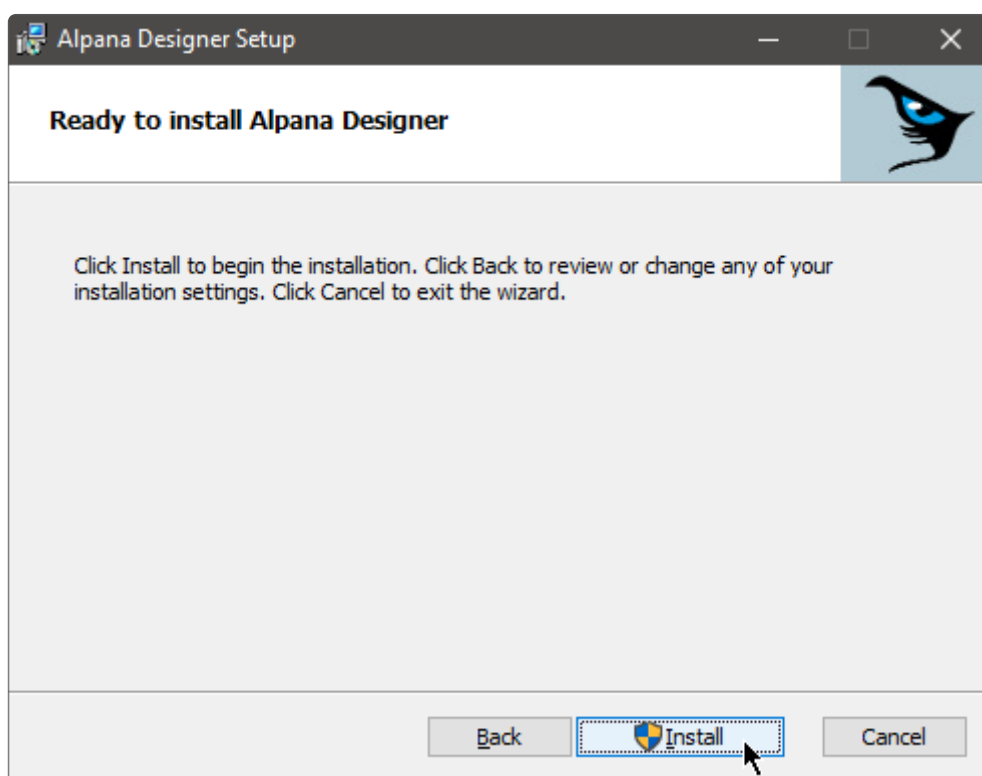
Install location

Validate the install location and click *Next* to continue :



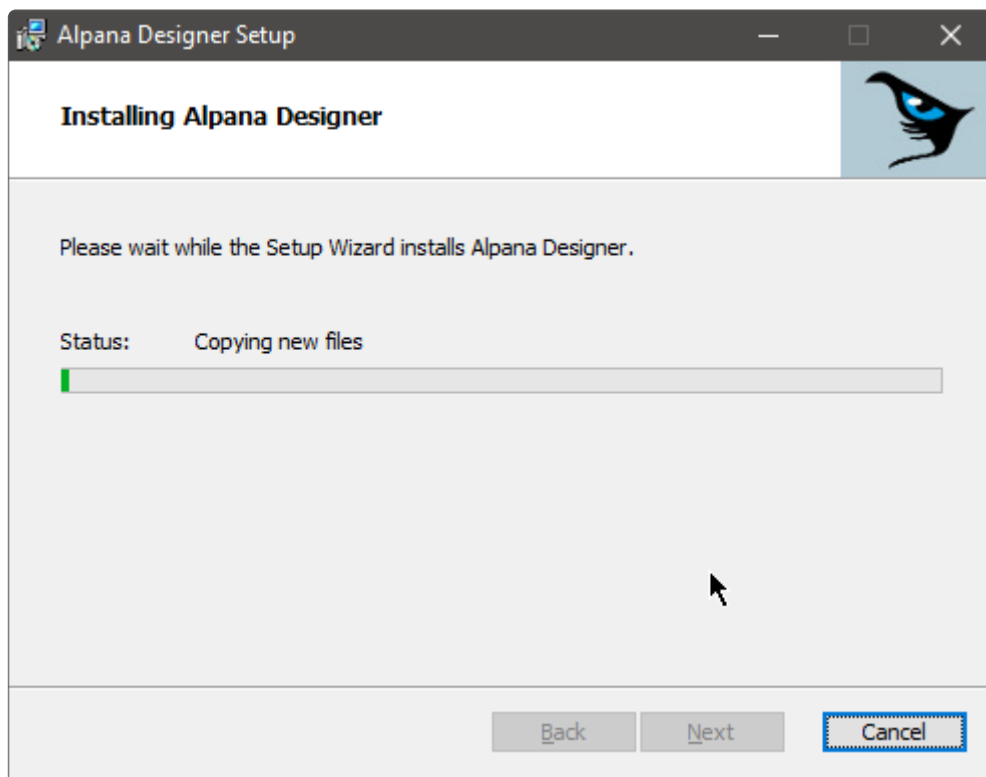
Start installation

Confirm the installation by clicking *Install*. You may be required to enter administrator credentials :



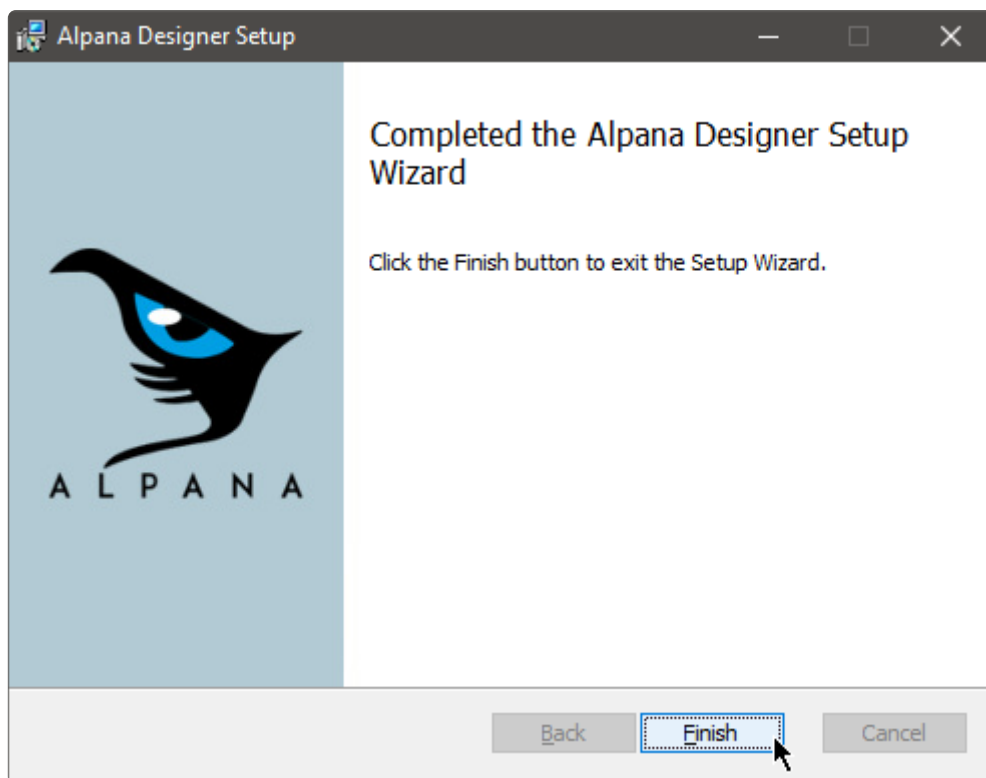
Wait

Wait for the installation to complete :



Complete

When the installation has completed, you can click *Finish* to exit the installer :



Last modified: 2019/06/17

2.1.3. Anti-virus and Firewall set up

Alpana Designer may make queries to external IP addresses (for fetching data from data sources, for example on port 1433 for SQL).

This may be detected as suspicious behavior by some security software.

For normal operation, you should let your security software allow this access to Alpana Designer.

Two processes may need to access external IP addresses :

`code192.Alpana.Designer.WPF.exe` : for the Data Source preview

`code192.Alpana.Designer.Preview.exe` : for the Widget previews

Last modified: 2019/05/28

2.1.4. Upgrading Alpana Designer

From version < v3.0

Un-install any previous version of Alpana.

From same major version numbers

If the version numbers have the same first 3 digits, then please uninstall the previous version manually first.

Example : version “a.b.c.d” and version “a.b.c.e”

Otherwise, you can run the new installer on top of the previous installation of Alpana Designer.

Last modified: 2019/07/26

2.1.5. Uninstalling Alpana Designer

To uninstall Alpana Designer, use *Windows Control Panel > Uninstall a Program* and select *Uninstall*.

Last modified: 2019/05/28

2.2. Troubleshooting

Troubleshooting Alpana Designer involves reading error messages from error logs.
To learn how to find those logs, see next.

Last modified: 2019/04/16

2.2.1. Finding error logs

What are error logs

Alpana Designer can encounter errors while configuring Dashboards. For example if the desired query cannot be run at that moment.

For similar reasons, during a Dashboard Preview, a “Data Retrieval Error” can be displayed.

Browsing error logs easily

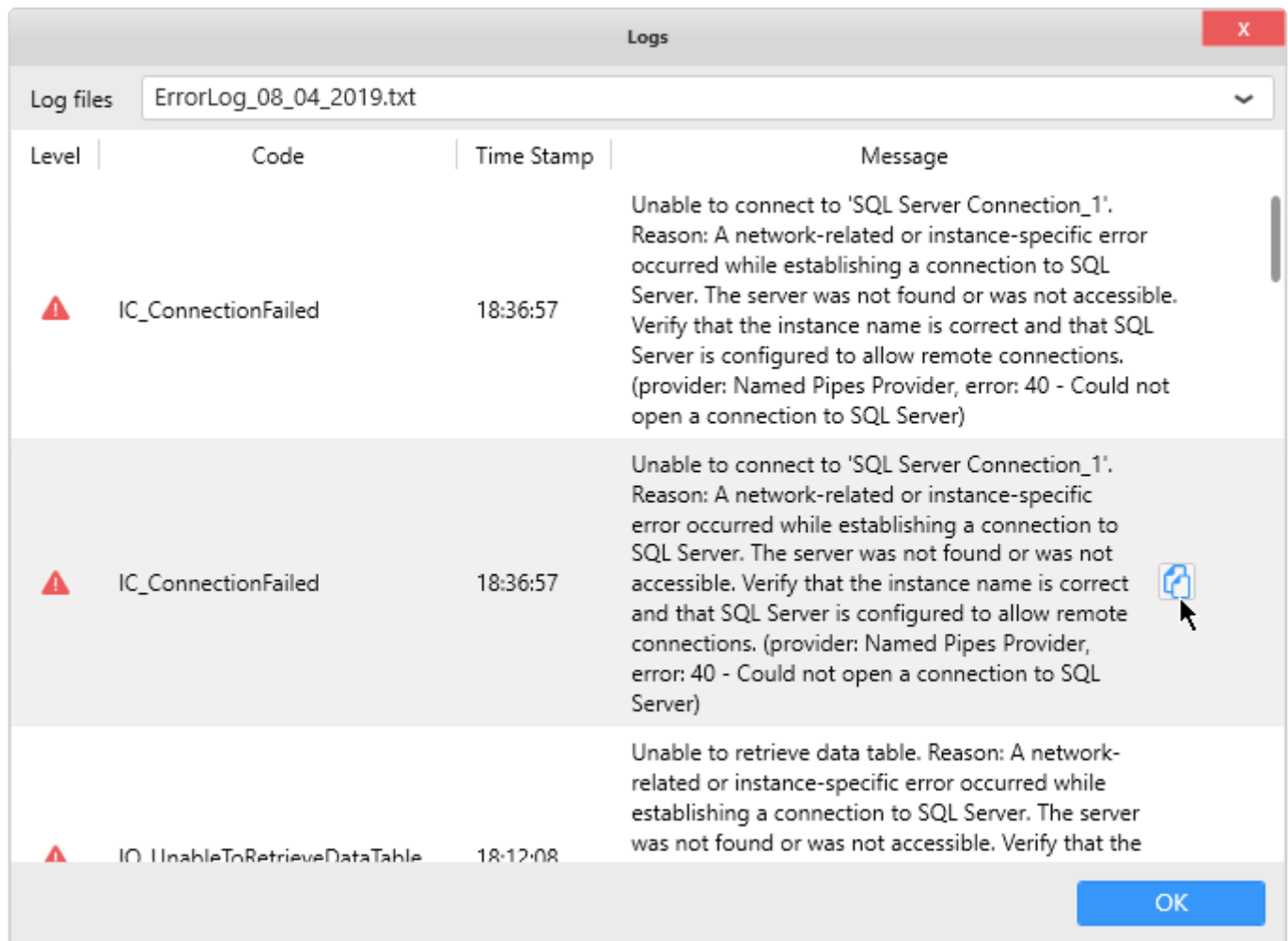
Since Alpana Designer v3.0, you can use a convenient browser for your error logs.

Click on the following icon at the top of the Designer window :



Select an error log file to open (by default, the latest is open), and browse the list of errors for that day.

You can also copy an error log to paste it into a ticket :



Where are error logs

In order to investigate and solve these errors, you can find the error log files at the following location on the machine where Alpana Designer is installed :

```
%localappdata%\code192\Alpana Platform\Alpana Designer\ (version number) \ErrorLogs\ErrorLog_*.txt
```



Tip :

To access the Local App Data folder, you can write the following in the location bar of Windows Explorer :

```
%localappdata%
```

Press enter and you will browse the desired location.

Usually, this folder is located at an address like C:\Users\ (your user name) \AppData\Local\

There is one ErrorLog_*.txt per day.

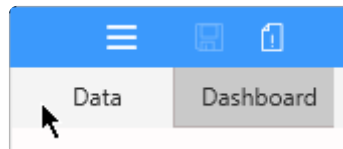
Last modified: 2019/06/17

2.3. Connecting to Data

Alpana Designer allows to connect to your data using *Connections*.

The current chapter describes this process.

Since Alpana Designer v3.0, all data preparation is made inside the *Data* activity tab.



Last modified: 2019/06/17

2.3.1. Alpana data model

Alpana Designer allows to access your data by providing a variety of **Connection** types. Once the appropriate credentials are validated, you will be able to navigate the data structure and retrieve the raw data.

Now, in order to feed your *Dashboard* with data, you will need to choose what data table from your *Connection* will become a **Data Source**.

In a basic workflow, this is enough. However, *Alpana Designer* lets you compose more complex preparation on your data when needed. For example, you can join tables together, including from multiple *Connections*, filter data, create Expressions, ...

Connections

What is a Connection

The **Connection** represents how you access the data :

- the Connection *type*, e.g. : Wonderware Historian, SQL Server, Excel, ...
- for server-based *Connections* : the *server address*
- for file-based *Connections* : the *file address*
- the necessary *credentials* : authentication mode, user name, password.

How Connections are managed in Alpana Designer

In *Alpana Designer*, you can :

- [Create a Connection](#)
- [Edit a Connection](#) to modify one of its parameters
- [Delete a Connection](#) to remove it from the current Dashboard

Data Sources

What is a Data Source

The **Data Source** is a representation of your data that you prepared from your created *Connection(s)*. It's what your *Widgets* (and your *Dashboard*) see.

A *Data Source* can be created by adding one or more element from one or more *Connection(s)* and joining them together :

- a data Table
- a data View
- a Stored Procedure
- a Custom Query

On top of that, further data preparation can be made :

- [hiding or renaming columns](#) to simplify the Data Source usage
- [filtering](#) the *Data Source* to limit the data
- [creating Expressions](#) in order to make more complex calculations

How Data Sources are managed in Alpana

In *Alpana Designer*, you can fully [manage](#) and [export](#) Data sources.

Last modified: 2019/04/30

2.3.1.1. Note on data storage

Alpana generally doesn't replicate data.

Alpana Dashboards, Data Sources and Connections generally don't contain the data itself. They only contain information on how to query the data sources and prepare the data on the fly.

Each Dashboard contains the definition of the Data Sources and Connections that were created within.

There are some exceptions listed below :

Exceptions

Buffer

A slight exception is the Buffer : for bufferized Connections, the buffer temporarily holds the data from the latest query, but it is flushed on the next refresh.

For details, see the chapter on [The data buffer](#)

File-based Connections

For file-based Connections (Excel, CSV, ...), Alpana stores a copy of the source file. This is to allow to continue displaying data even when the original file is not accessible.

When saving a Dashboard.alpd file, the file will contain a copy of all files from the dependent file-based Connections.

So if a dashboard is based only on file data, it can be re-opened in a different environment even without the original files.

When getting data for a file-based Connection, the workflow is :

1. if file exists at location defined in the Connection, query the file
2. else open file embarked withing the Dashboard.alpd file

Last modified: 2019/03/14

2.3.1.2. Redundant Connections

What are redundant Connections

Since v3.1.0, Alpana implements a redundancy mechanism on Connections that allows to store multiple connection configurations inside a single Connection.

Each connection configuration is defined by a set of parameters (database name, user name, etc...), called a *Parameter Set*.

This can be used in basically 3 types of scenarios :

1. Using different databases for design time and run time
2. Implementing a fail-over mechanism to keep Dashboards working even when the primary database is down
3. a combination of scenario 1 and 2

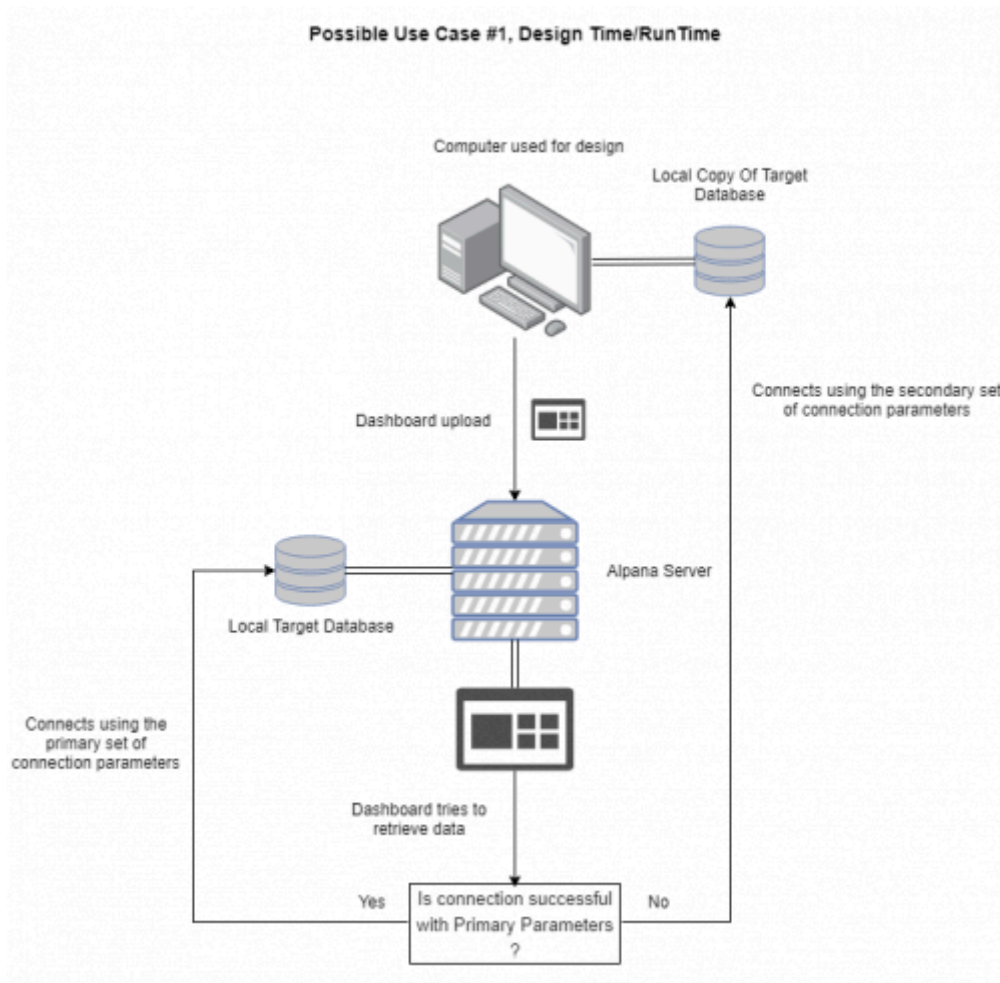
Scenario 1 : Design Time / Run Time

You can select a different Parameter Set for Alpana Designer and Alpana Server.

This is useful for example if the developer doesn't have access to the production database from the development machine, maybe because they are on a different network.

In this case, you can configure 2 different Parameter Sets :

1. one to work in the Designer by pointing on the local development database
2. one for the published Dashboards by pointing to the production database



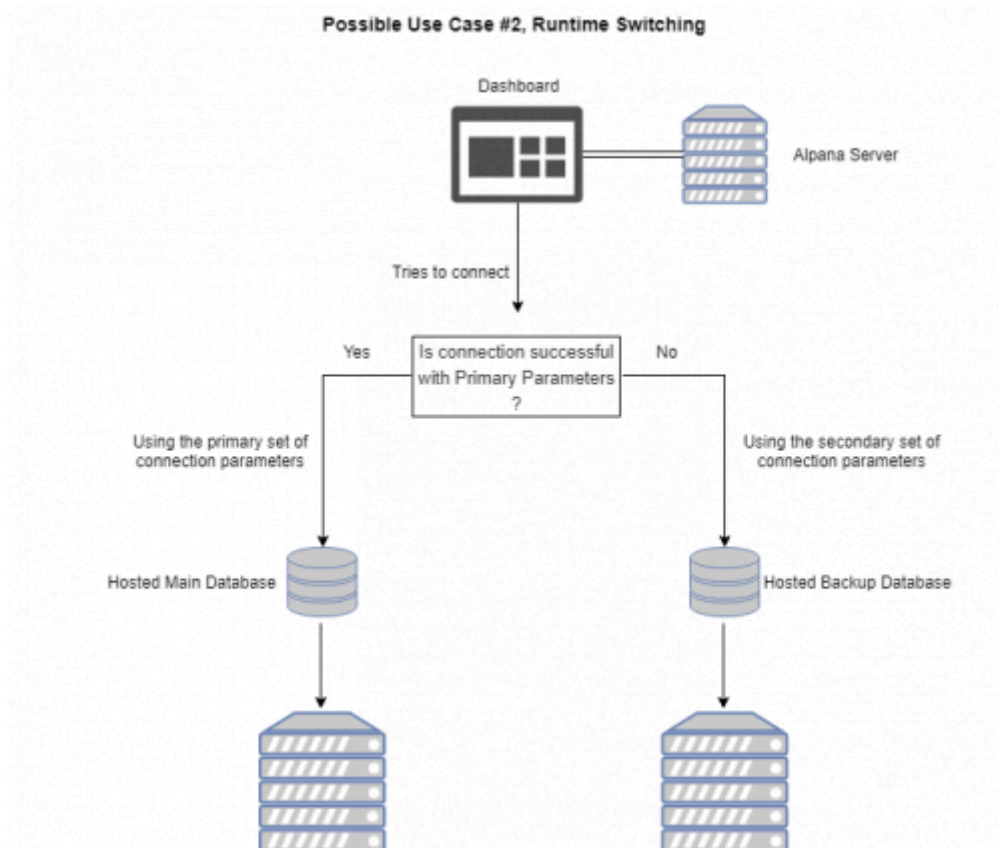
Scenario 2 : Database1 / Database2

The **Primary Parameter Set** is where Alpana will try to connect first.
If the connection fails, Alpana will use the **Secondary Parameter Set**.

This is useful for example to ensure that Dashboards can always get data, even when the main database is unavailable.

In this case, you can configure 2 different Parameter Sets :

1. one pointing to the main database
2. one pointing to the backup database



Scenario 3 : combine

You can create 3 parameter sets for handling both scenarios above :

1. Parameter Set 1 : primary development
2. Parameter Set 2 : primary production
3. Parameter Set 3 : backup production

Fallback Logic

When a primary connection fails, Alpana will check the next one in order until it succeeds, or until all fail.



Sometimes the connection failure is slow to detect, for example on some network timeouts.

If the primary connection fails with a 30s timeout, it means the fallback mechanism won't kick in until then.

This delay will be repeated each time the dashboard is loaded, and until the primary connection is back online.

As a mitigation, all widgets that use the same connection will inherit the fallback.

This means that during a single page load, the timeout will happen only once per Connection.



Note : there is a caching mechanism, so you may not notice the redundancy immediately.
When redundancy happens, you may need to refresh the page.
When the connection becomes good again, you may need to wait a few minutes for the dashboards to apply the primary connection again.

Alpana Designer

In Alpana Designer, the order is :

1. try to connect to the *Primary parameters at design time*
2. if it fails, try to connect to the *Primary parameters after publication*
3. if it fails, try to connect to other parameters

Alpana Server

In Alpana Server, the order is :

1. try to connect to the *Primary parameters after publication*
2. if it fails, try to connect to the *Primary parameters at design time*
3. if it fails, try to connect to other parameters

Connection Types

Redundant connections are supported by the following Connection types :

- SQL
- Historian
- Web

Managing *Parameter Sets*

A *Parameter Set* is the set of all the parameters required to connect to the data.

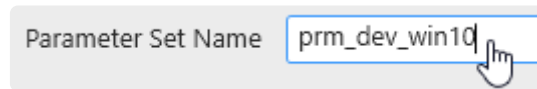
Default behavior

By default, only one Parameter Set is created for the current connection, and is used for both Designer

and Server.

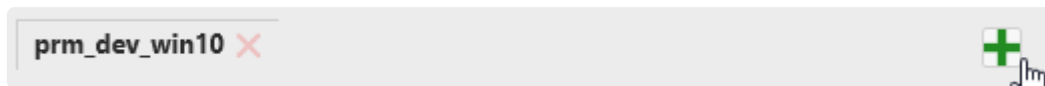
Renaming *Parameter Sets*

You can rename each Parameter Set :



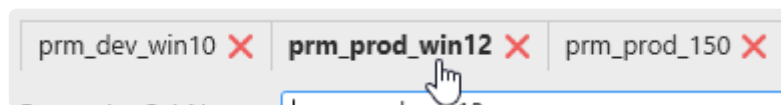
Creating *Parameter Sets*

Click the + icon to create a new Parameter Set :



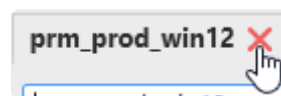
Editing other *Parameter Sets*

To edit a Parameter Set, click on the corresponding tab :



Deleting *Parameter Sets*

Click on the red cross to delete the corresponding Parameter Set :



Designer ↔ Server switching

To choose which Parameter Set to use for Designer or Server, select the desired Parameter Set in each list :

1. *Primary parameters at design time* : primary connection to use while in Alpana Designer
2. *Primary parameters after publication* : primary connection to use while published in Alpana Server

Primary parameters at design time :	<input type="text" value="prm_dev_win10"/>
Primary parameters after publication :	<input type="text" value="prm_prod_win12"/>

Last modified: 2019/08/23

2.3.2. The data Buffer

Introduction : What is the Buffer ?

The primary source for data in Alpana is a **database** (SQL Server, SQLite).

When a *Dashboard* requires data from a *Data Source* that is based only on **database Connections**, the query is composed by *Alpana* and executed directly on the target database.

✿ except in multi-source : when the database Connection is used together with another Connection, it also becomes **bufferized** as multi-source (see [next page](#))

On the other hand, Data that comes from other types of *Connections*, need to be **bufferized** : data is first fetched from the source and copied inside a *Buffer database* internal to *Alpana*, then *Dashboards* will query this *Buffer*.

Last modified: 2019/03/12

2.3.2.1. Bufferized Connection types : What gets bufferized ?

All Connection types that are not **databases** are bufferized :

- RESTful API (web)
- file-based Connections : Excel, CSV

Multi-Source case

Multi-Source is a special case : in a Multi-Source context, even database-based Connections are bufferized.

Example :

SQL Connection1 points to your “MESDB” database, and **SQL Connection2** points to your “OTHER_DB” database.

You create DataSource1 by making a join between Table1 from Connection1 and Table2 from Connection2.

⇒ Your DataSource1 is a **Multi-Source** Data Source. It is bufferized.

Last modified: 2019/03/12

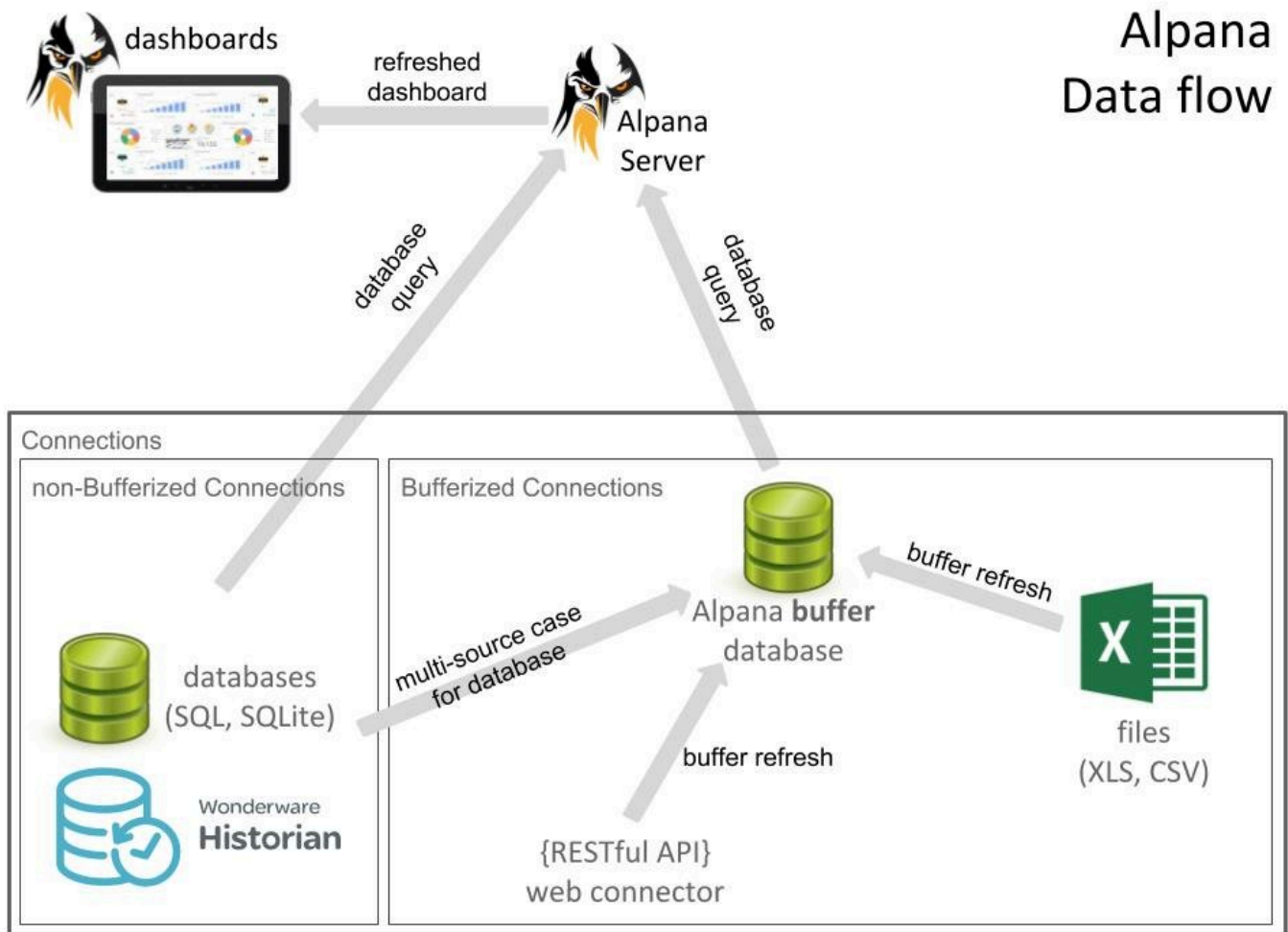
2.3.2.2. Data Flow : How bufferization happens ?

Data flow

When data is requested from a bufferized *Connection*, the following happens :

- If **Buffer Refresh** is not enabled : data is retrieved from the buffer.
- If **Buffer Refresh** is enabled :
 - If the buffer is valid \Rightarrow data is retrieved from the buffer.
 - If the buffer is out of date \Rightarrow the buffer is refreshed by data from the Connection (the Web connection, the CSV file, etc...).

Alpana Data flow



Content of the Alpana buffer database

Alpana Server creates, deletes and updates tables in the buffer database.

This is a purely internal mechanism and should not be relied on by external tools.



You shouldn't use the buffer database for other purposes, otherwise you might lose your data.

Last modified: 2019/03/12

2.3.2.3. Buffer database setup

In Alpana v3.0.0, the Buffer database is setup automatically :

- for the Alpana Server : this setting applies to published Dashboards. A Microsoft SQL database is created by Alpana Server to host the buffer tables.
- for each Alpana Designer installation : this setting applies when editing Dashboards on that host. A SQLite database is created in a local folder.

Last modified: 2019/05/28

2.3.2.4. Buffer Refresh settings

What is Buffer Refresh Settings

This setting represents : How long do you want to query the old data in the buffer instead of getting new data from the source ?

For example

You connect to an Excel file.

The data is first copied in the buffer.

If you don't set up Buffer Refresh Settings, it will never be updated.

If you set up "Refresh Every 1 Week", it will not be update for 1 Week.

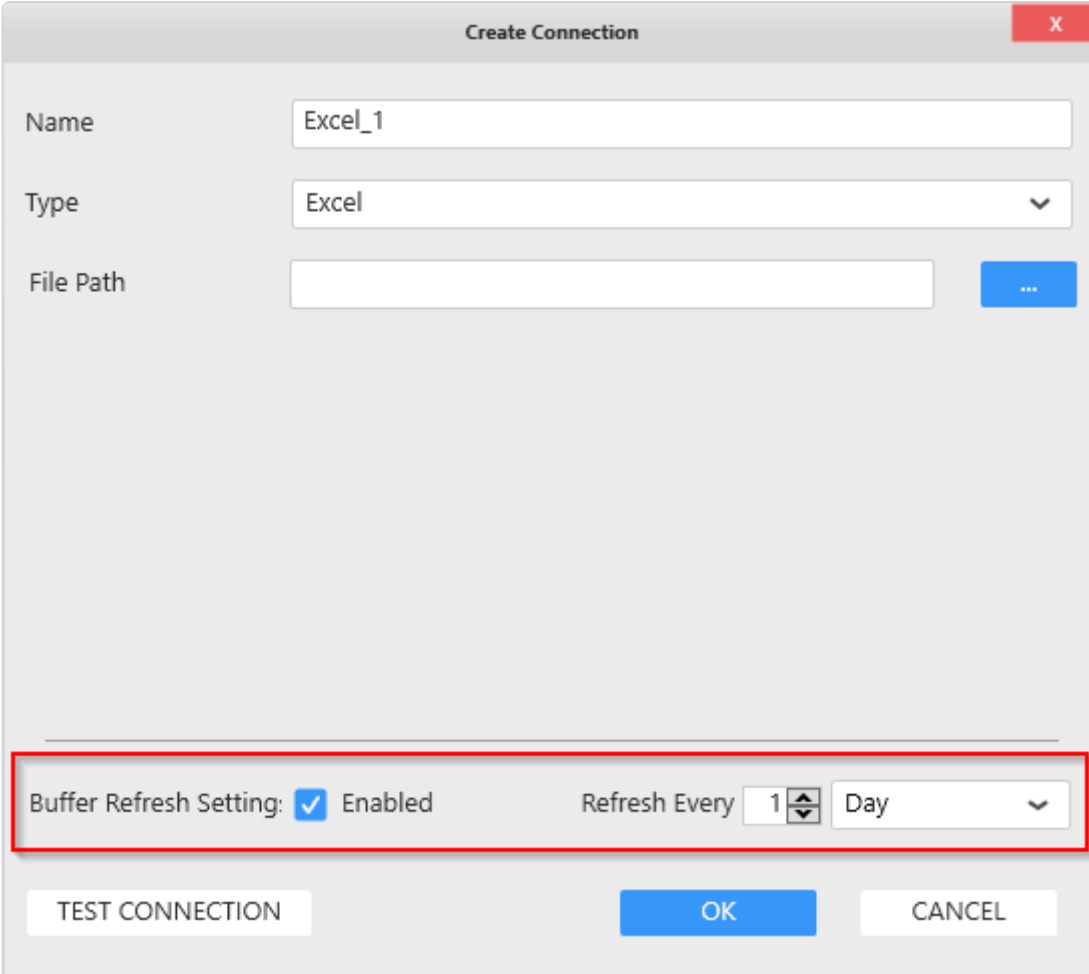
Then, the next time someone makes a query to this Connection, new data is copied in the Buffer.

Where to set up

Each *Connection* has its own refresh settings.

You can change the refresh duration by default at the Connection creation (or edition).

Just edit the Connection, and set up "Buffer Refresh Settings" as Enabled, and a duration to "Refresh Every".



The image shows a 'Create Connection' dialog box with a title bar containing a close button (X). The dialog has three main input fields: 'Name' with the value 'Excel_1', 'Type' with a dropdown menu showing 'Excel', and 'File Path' with an empty text box and a blue button with three dots. At the bottom, there is a section for 'Buffer Refresh Setting' which is highlighted with a red rectangle. This section includes a checked checkbox, the word 'Enabled', a 'Refresh Every' label, a spinner box with the value '1', and a dropdown menu with 'Day' selected. Below this section are three buttons: 'TEST CONNECTION', 'OK', and 'CANCEL'.

Create Connection

Name: Excel_1

Type: Excel

File Path: [Empty field] [...]

Buffer Refresh Setting: ☒ Enabled Refresh Every 1 Day

TEST CONNECTION OK CANCEL

* Note : don't forget to set Buffer Refresh settings also for a SQL Connection if you intend to use it in a multi-Connection Data Source !

Last modified: 2019/03/12

2.3.2.5. refreshing file-based connections

The file-based connections don't refresh based on the file location.

If you want to refresh data for a dashboard based on a file (CSV, Excel, JSON), use the external tool

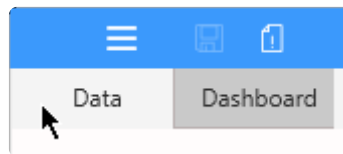
[Alpana Update File Dependency](#)

Last modified: 2019/08/12

2.3.3. Creating and Managing Connections

Alpana Designer allows to create as many Connections as required.

Work with Connections is all done inside the *Data* activity tab of Alpana Designer :

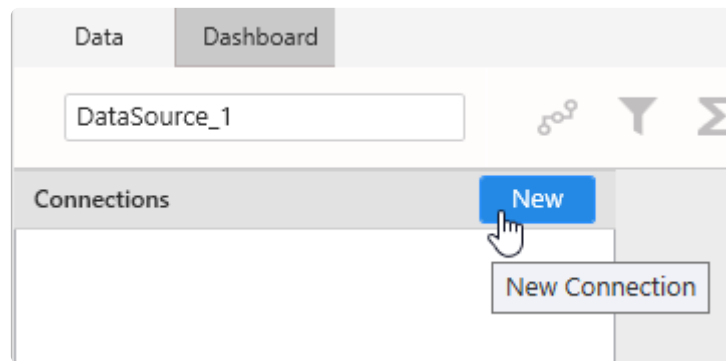


To learn how to create and manage Connections, see the current chapter.

Last modified: 2019/04/19

2.3.3.1. Create a Connection

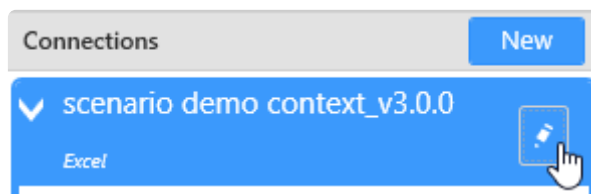
To create a new Connection, click *New* :



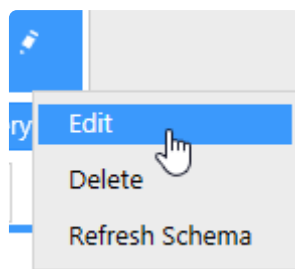
Last modified: 2019/04/16

2.3.3.2. Edit a Connection

To open a Connection to start to Edit it, click the pen icon next to the Connection name :



From the pop-up, select *Edit* :

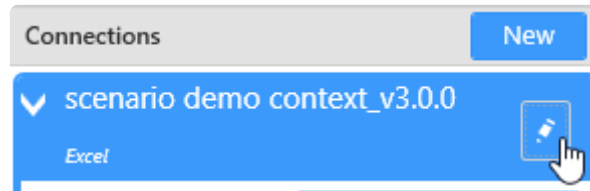


Then the corresponding “Edit Connection” dialog opens for edition (see below for each [Connection type](#)).

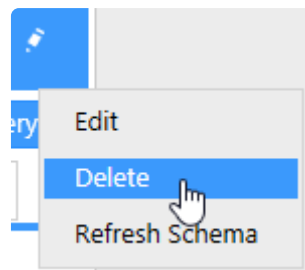
Last modified: 2019/04/16

2.3.3.3. Delete a Connection

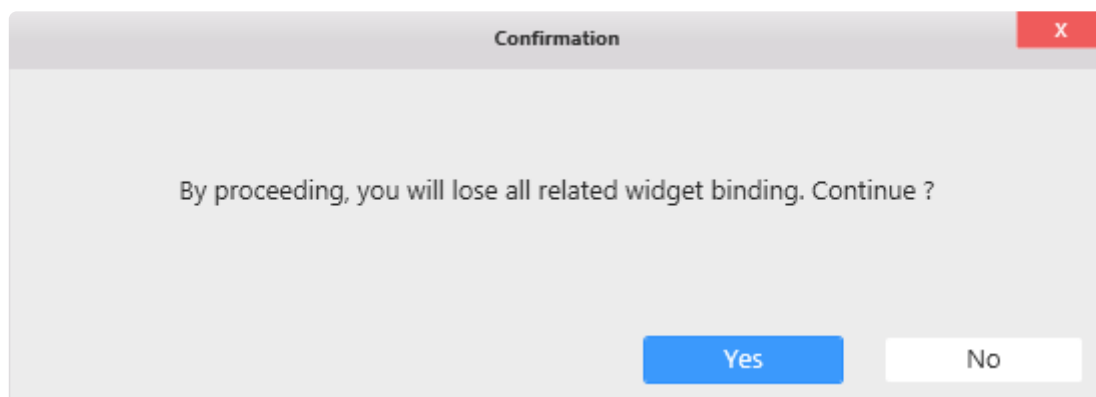
To Delete an existing Connection, click the pen icon next to the Connection name :



From the pop-up, select *Delete* :



If the Connection is used by any Widget or any Data Source, the following message will appear :



If you choose “Yes”, the Connection is deleted and all dependent Data Sources and Widgets lose their data binding.

Last modified: 2019/04/16

2.3.4. Connection types

All connection types are listed in the next chapters, with all their configuration properties.

Last modified: 2019/05/15

2.3.4.1. SQL Server

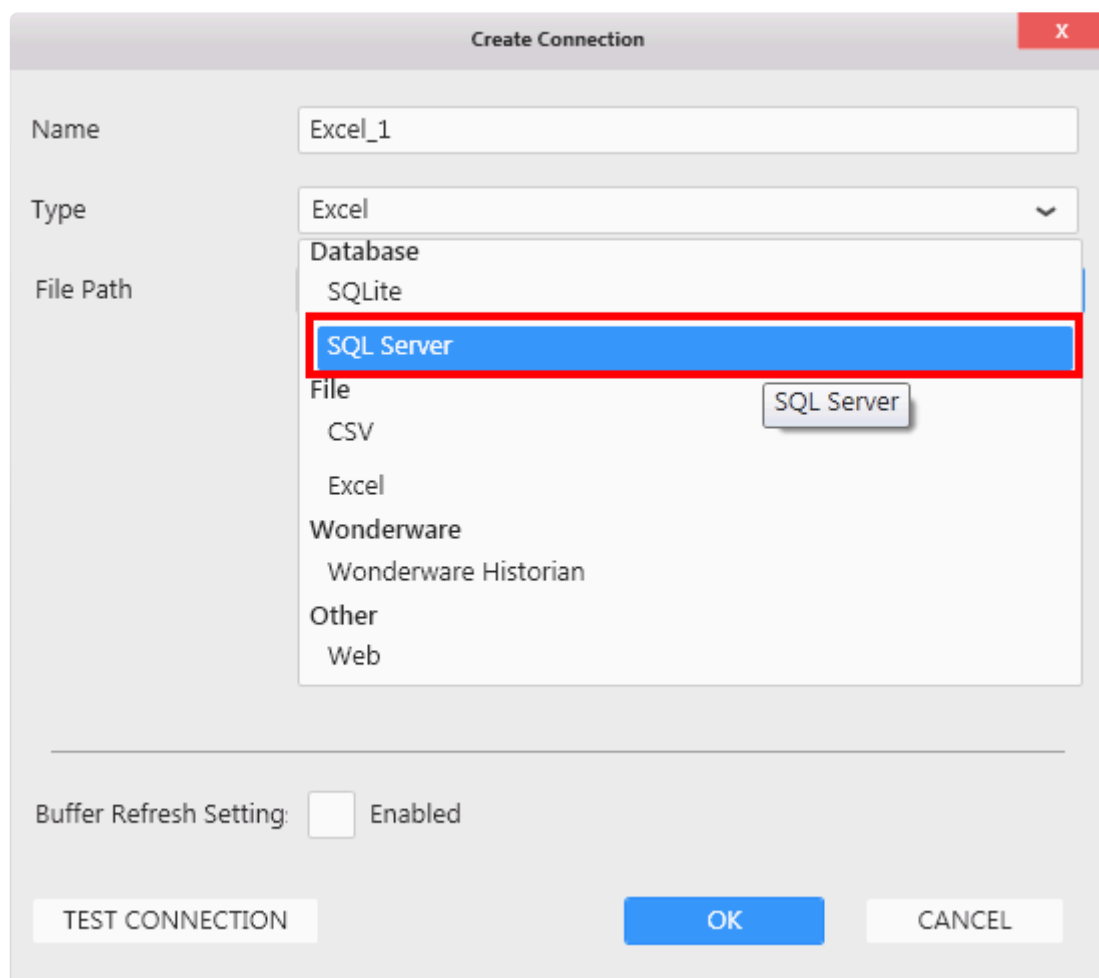
Versions

This connector can be used to get data from local or remote Microsoft SQL Server installations with the following versions :

- Microsoft SQL Server 2012 and later
- Microsoft SQL Server Express 2012 and later

Selecting Connection Type

In the “Create Connection” dialog change the Connection “Type” by selecting “SQL Server” :



Connection Parameters

The screenshot shows a configuration window with the following fields and options:

- Server:** A text box containing "localhost".
- Authentication Mode:** A dropdown menu set to "Windows".
- User Name:** An empty text box.
- Password:** An empty text box.
- Database:** A section containing two options:
 - Manual:** An unselected radio button next to a text box containing "Northwind".
 - Detection:** A selected radio button (highlighted with a blue circle) next to a dropdown menu set to "Northwind". A refresh icon is located to the right of this dropdown.

Server

This is the address of the SQL Server instance you need to access. Possible values are :

- (local) or localhost : local server
 - <NetBIOSName> : server named with a NetBIOS name
 - <IPv4Address> : server accessible through a IP v4 address
 - <IPv6Address> : server accessible through a IP v6 address
 - <server address>\<instance name> : connect to a specific instance. For SQL Express, it may be necessary to use <server address>\SQLEXPRESS
 - <server address>,1433 : 1433 is the default port for SQL Server, but can be changed like this
- Be aware that when searching for a server instance, Alpana waits for a response. If the server instance doesn't respond, there is a rather long timeout.

Authentication Mode

"SQL Server" : will use SQL Server authentication with "User name" and "Password".

"Windows" : will use Windows Authentication from the Windows user that generates the query.



Important :

Windows Authentication will use a **different Windows user** when in Designer and when published in Server. That Windows user may not have the required access to SQL Server.

When published in Server, the Windows user will be that configured in the IIS site IIS_A_PPOOL\Alpana3App.

For example, this user will need db_owner permissions to run stored procedures or

functions.

Database

The name of the database to connect to.

Detection

This lets you select the database name from the list of databases.

✿ Tip : After changing credentials, you can fetch again a new list of database names, by clicking the refresh icon.

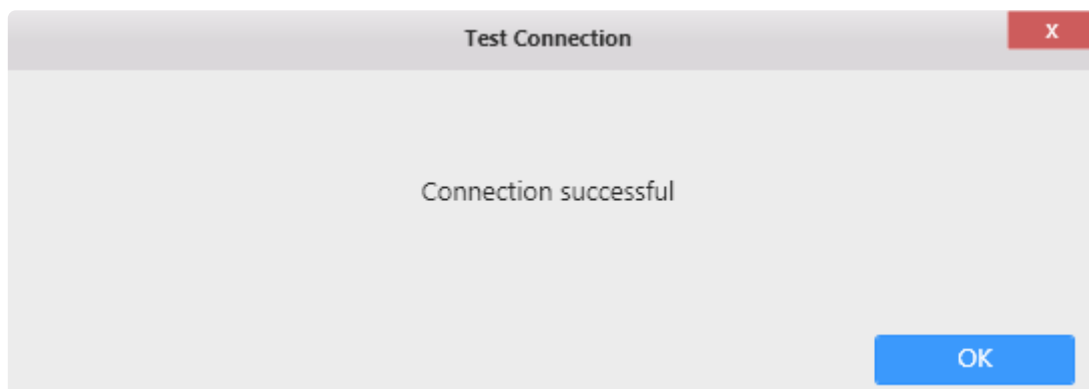
Manual

This lets you write the database name as free text.

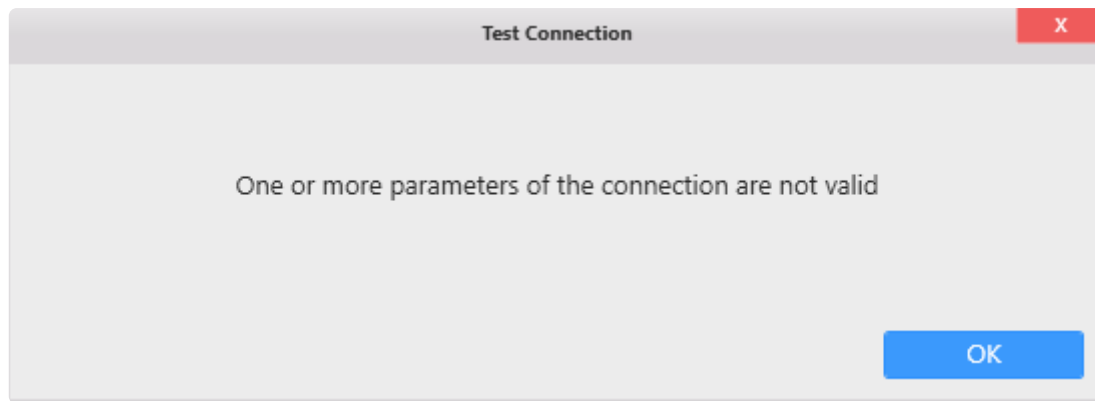
This is useful for creating a connection to a database which you cannot access while in the Designer. For example when the database name is different in the production environment.

Test Connection

To test the connection for validity, click “Test Connection” button. The following confirmation message will confirm its validity.



If the connection is invalid, the following message appears :



Then you will need to fix the Connection parameters (credentials, server name, ...) or check network connectivity (is the SQL Server accessible from here ?)

Connection Schema

The SQL Server Connector fetches the following object types in the database schema :

- Tables
- Views
- Stored Procedures
- Table-Valued Functions

Data refresh and Buffer

The SQL Connector is not bufferized by default.

However, when data is joined with another Alpana Connection (including another SQL Connection), data is bufferized.

If you need to get new data, see Chapter [The data Buffer](#)

Last modified: 2019/07/26

2.3.4.2. Excel

Formats

You can connect to Microsoft Excel workbooks in format `*.xlsx` or `*.xls` containing “database-like” tabular data.

The Excel workbook should have column names at first row of sheets followed by data rows.

The data type of all values in a column should all be the same. Trying to parse an Excel file where a column has a number as first value and string as second value will result in a parsing error in SQL buffer.

Other complex sheet designs may not be supported.

Formulas are executed every time the file is opened.

Selecting Connection Type

In the “Create Connection” dialog change the Connection “Type” by selecting “*Excel*” (should be selected by default) :

Create Connection

Name: Excel_1

Type: Excel

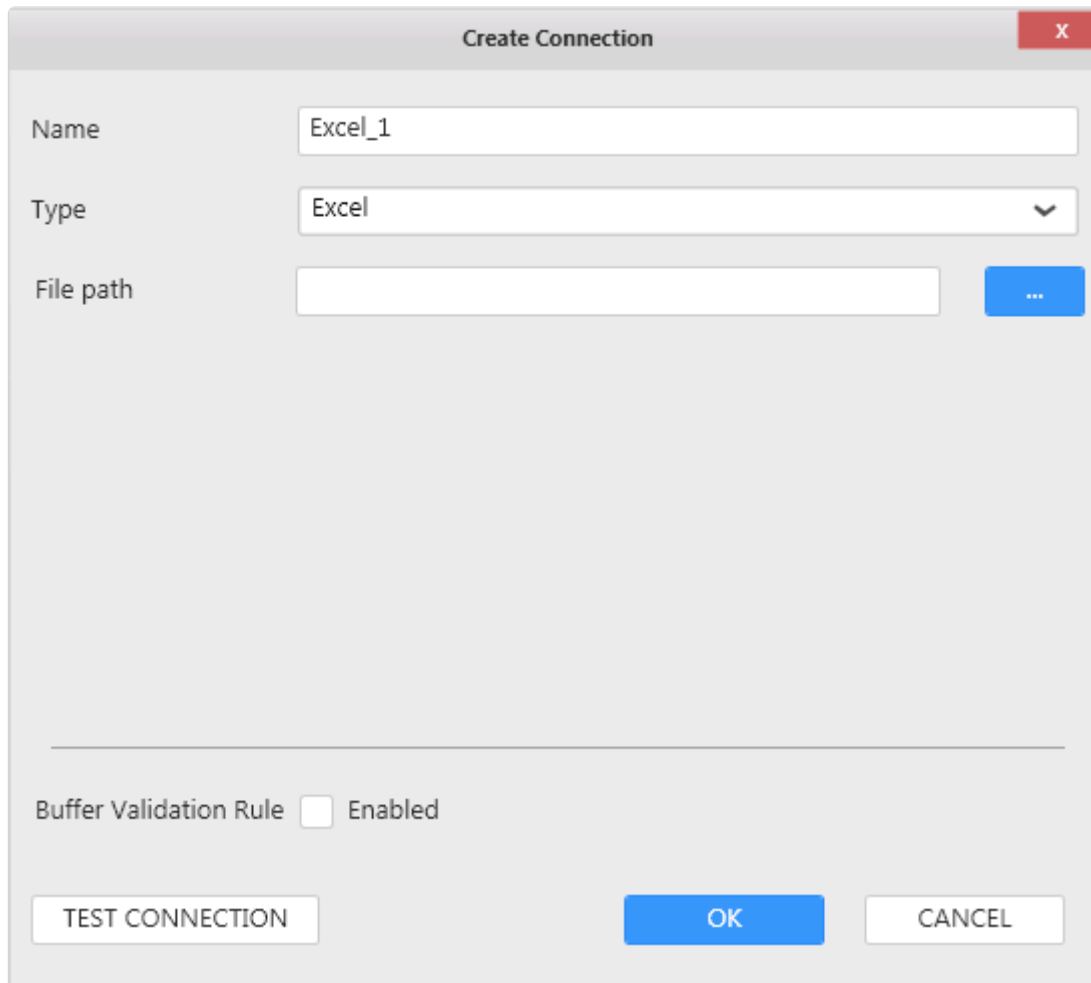
File Path:

- Database
 - SQLite
 - SQL Server
- File
 - CSV
 - Excel**
- wonderware
 - Wonderware Historian
- Other
 - Web

Buffer Refresh Setting: ☐ Enabled

TEST CONNECTION OK CANCEL

Connection Parameters



The image shows a 'Create Connection' dialog box with a title bar containing a close button (X). The dialog has three main input fields: 'Name' with the text 'Excel_1', 'Type' with a dropdown menu showing 'Excel', and 'File path' with an empty text box and a blue button with three dots (browse). Below these fields is a section for 'Buffer Validation Rule' with an unchecked checkbox and the text 'Enabled'. At the bottom are three buttons: 'TEST CONNECTION', 'OK', and 'CANCEL'.

File Path

Click to browse for an Excel file.



Important :

The path configured here may not be accessible when published in Alpana Server for different reasons (maybe it's a different machine and the Excel file is not there at the same path, maybe the Windows user doesn't have access to the path). When published in Server, the Windows user will be that configured in the IIS site, by default "LocalSystem", which can correspond to "NT AUTHORITY\SYSTEM". When the file is not found at the configured path, Alpana uses the copy of the Excel file that is embarked within the Dashboard. If so, then you will not see your data update.

Connection Schema

The Excel Connection presents Excel Sheets as tables to be used in a Data Source.

Data refresh and Buffer

The Excel Connector is bufferized.

If you need to get new data, see Chapter [The data Buffer](#)

Last modified: 2019/03/12

2.3.4.3. CSV

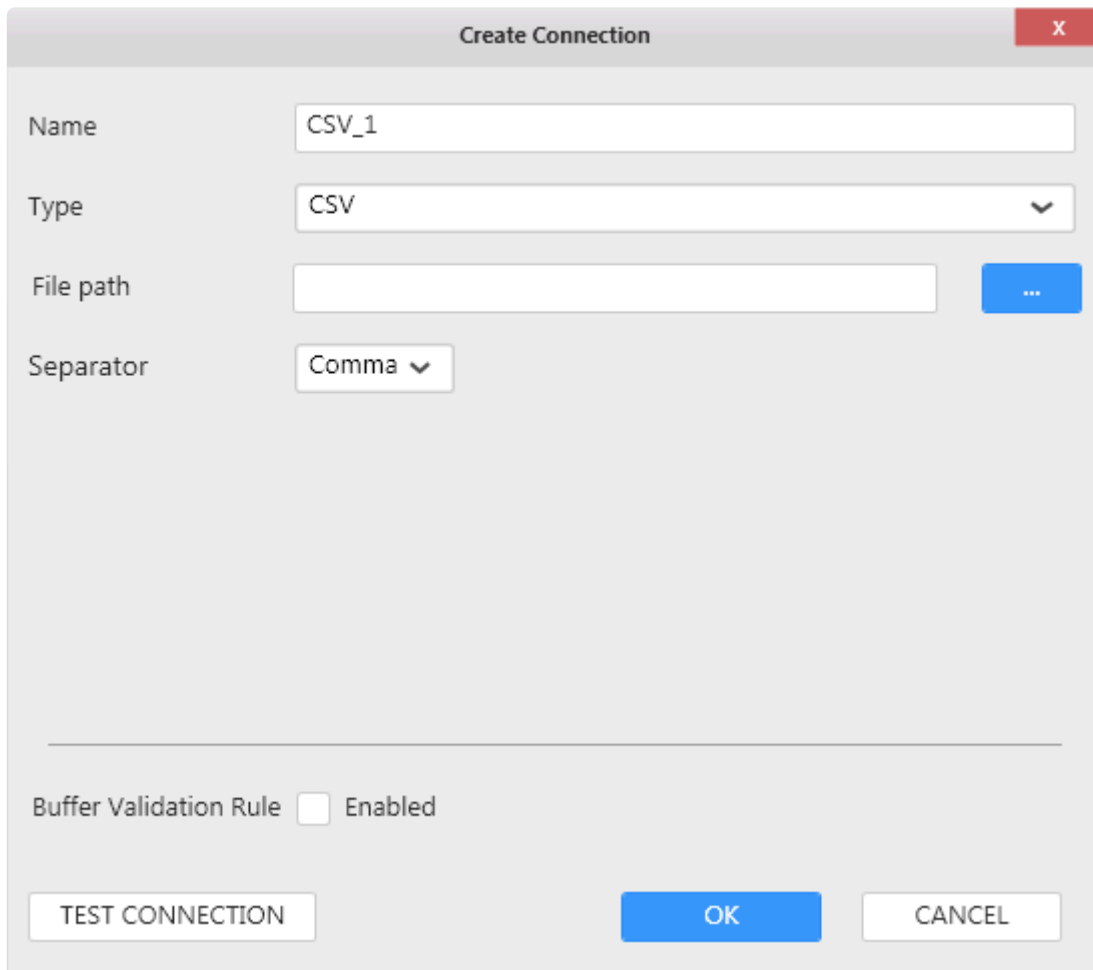
Formats

Although “CSV” means “Comma-Separated Values”, Alpana supports other separators than comma :

- Comma : “ , ”
- Semicolon : “ ; ”
- Space : “ ”
- Tab : “ ”

Selecting Connection Type

In the “Create Connection” dialog change the Connection “Type” by selecting “CSV” :



Create Connection [X]

Name: CSV_1

Type: CSV

File path: [Empty] [...]

Separator: Comma

Buffer Validation Rule ☐ Enabled

[TEST CONNECTION] [OK] [CANCEL]

Connection Parameters

File Path

Click to browse for a CSV file.



Important :

The path configured here may not be accessible when published in Alpana Server for different reasons (maybe it's a different machine and the CSV file is not there at the same path, maybe the Windows user doesn't have access to the path).

When published in Server, the Windows user will be that configured in the IIS site, by default "LocalSystem", which can correspond to "NT AUTHORITY\SYSTEM".

When the file is not found at the configured path, Alpana uses the copy of the CSV file that is embarked within the Dashboard. If so, then you will not see your data update.

Connection Schema

The CSV Connection presents a single object in the schema : a table with the same name as the file.

Data fields type

Since CSV doesn't allow to create explicit data types, Alpana has to detect the data types of fields.

If this automatic detection is incorrect, you can always change the data type in the Data Source :

[Changing Column Type](#)



Integers have a size limit. If this causes an issue, you can switch to the "Number" data type.

Data refresh and Buffer

The CSV Connector is bufferized.

If you need to get new data, see Chapter [The data Buffer](#)

Last modified: 2019/03/21

2.3.4.4. Wonderware Historian

Specific documentation to connect to Wonderware Historian is available here : [Wonderware Historian](#)

Last modified: 2019/03/13

2.3.4.5. Web

Formats

The Web Connector allows to get data from RESTful Web Services that serve JSON content.

The response content is expected with the following format only :

```
{
  "TableA": [
    { // data record 1 for TableA
      "FieldA1": "ValueA1_1",
      "FieldA2": "ValueA2_1",
      ...
    },
    { // data record 2 for TableA
      "FieldA1": "ValueA1_2",
      "FieldA2": "ValueA2_2",
      ...
    },
    { // data record 3 for TableA
      "FieldA1": "ValueA1_3",
      "FieldA2": "ValueA2_3",
      ...
    },
    ...
  ],
  "TableB": [
    { // data record 1 for TableB
      "FieldB1": "ValueB1_1",
      "FieldB2": "ValueB2_1",
      ...
    },
    ...
  ],
  ...
}
```

Selecting Connection Type

In the “Create Connection” dialog change the Connection “Type” by selecting “Web” :

Url	<input type="text" value="http://yoursite.com/yourendpoint"/>
Authentication Mode	<input type="text" value="No Authentication"/>
User Name	<input type="text"/>
Password	<input type="password"/>

Connection Parameters

URL

Enter the API URL which must be a valid REST API as seen above.

Authentication Type

Select the authentication to use :

- “No Authentication”
- “Basic HTTP Authentication” : with User name and Password

Connection Schema

The Web Connection presents first level object lists as tables to be used in a Data Source.

In the example above, it will list “Table1”, “TableB”, ...

Data refresh and Buffer

The Web Connector is bufferized.

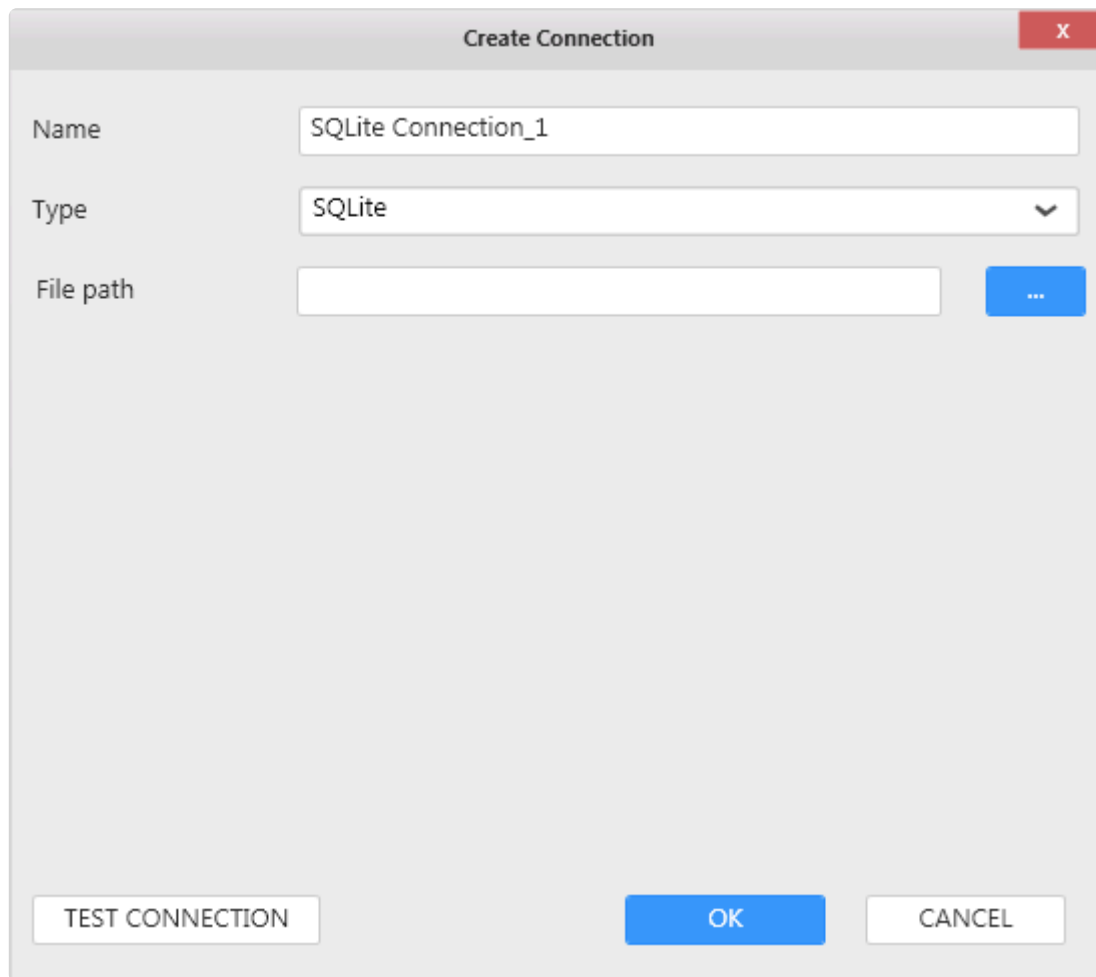
If you need to get new data, see Chapter [The data Buffer](#)

Last modified: 2019/07/26

2.3.4.6. SQLite

Selecting Connection Type

In the “Create Connection” dialog change the Connection “Type” by selecting “SQLite” :



The screenshot shows a "Create Connection" dialog box with the following fields and controls:

- Name:** A text input field containing "SQLite Connection_1".
- Type:** A dropdown menu currently showing "SQLite".
- File path:** An empty text input field with a blue button containing three dots (browse) to its right.
- Buttons:** At the bottom, there are three buttons: "TEST CONNECTION", "OK", and "CANCEL".

Connection Parameters

File Path

Click to browse for a SQLite database file.

Data refresh and Buffer

The SQLite Connector is not bufferized by default.

However, when data is joined with another Alpana Connection (including another SQLite Connection), data is bufferized.

If you need to get new data, see Chapter [The data Buffer](#)

Last modified: 2019/03/13

2.4. Wonderware Historian

The Alpana connector for Wonderware Historian allows to get data from the most common query types of Wonderware Historian in a familiar interface.

Getting data is done in two steps :

1. creating the *Connection* : logging into the Wonderware Historian database
2. using the Connection to create *Tables* : create queries to Historian and fetch data into tables

Last modified: 2019/05/15

2.4.1. Connecting to Wonderware Historian

As with other *Connection* types, connecting to Wonderware Historian means defining how Alpana will find the data : what is the server address and what is the login.

See the following chapters for detailed information.

Last modified: 2019/05/15

2.4.1.1. Version compatibility

Alpana allows to connect natively to the following Wonderware Historian versions :

- 2012 (10.0)
- 2012 R2 (11.0)
- 2014 (11.5)
- 2014 R2 (11.6)
- 2014 R2 SP1 (11.6)
- 2017 Update 3 (17.3)

Last modified: 2019/07/22

2.4.1.2. Important pre-requisite



Important

If you connect to a fresh install of *Wonderware Historian* Server, tags may not be available for browsing.

Starting Historian clients once

In order to make tags available for browsing, you will have to **startup Wonderware Historian Client Query or Wonderware Historian Client Trend**.

This needs to be done only once. Then the tags will be made available for Alpana.

Enabling SQL login to Historian

Alpana requires SQL authentication to login to Historian.

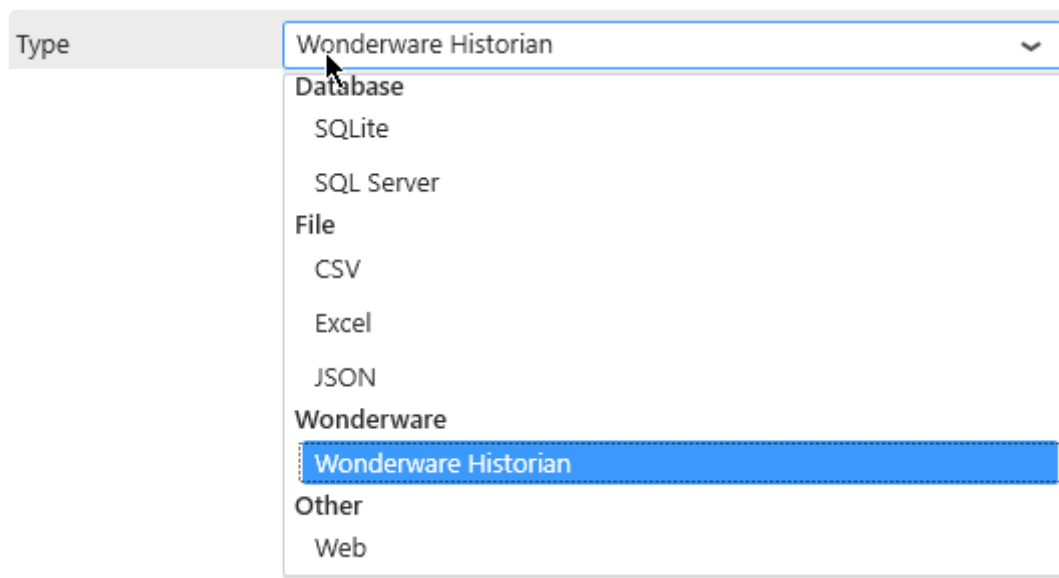
In some versions, those are disabled by default and need to be enabled or created.

See your Wonderware Historian documentation for more details

Last modified: 2019/07/19

2.4.1.3. Selecting Connection Type

In the “Create Connection” dialog change the [Connection Type](#) by selecting “*Wonderware Historian*” :



Last modified: 2019/05/15

2.4.1.4. Login to a Wonderware Historian server

Connection Parameters

To login to a Wonderware Historian server, fill-in the following fields :

Host	<input type="text" value="localhost"/>
Username	<input type="text" value="dashboards"/>
Password	<input type="password" value="••••••••••"/>

If in the current session you have already connected to a Historian server, you can select it in the combobox below to auto-fill the Host Input.



Important :

Because the Historian Connector can connect to any server regardless if it is a local server or not, you need to specify a **SQL Server account** type to connect to the Runtime Database.

Host

This is the address of the SQL Server instance you need to access for the Wonderware Historian “Runtime” database. Possible values are :

- (local) or localhost : local server
- <NetBIOSName> : server named with a NetBIOS name
- <IPv4Address> : server accessible through a IP v4 address
- <IPv6Address> : server accessible through a IP v6 address
- <server address>\<instance name> : connect to a specific instance. For SQL Express, it may be necessary to use <server address>\SQLEXPRESS
- <server address>,1433 : 1433 is the default port for SQL Server, but can be changed like this



Be aware that when searching for a server instance, Alpana waits for a response. If the

server instance doesn't respond, there is a rather long timeout.

Username and Password

The Username and Password must be valid SQL credentials configured with Wonderware Historian.

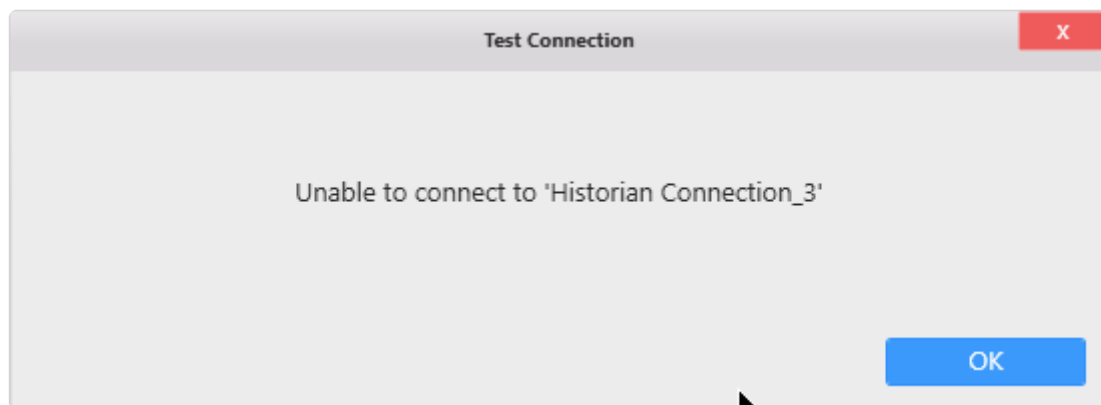
At least, the SQL user must have read permissions to the Historian “Runtime” database.

Connecting

When all fields are filled, click OK to create the Connection.

Connection errors

If a connection error happens, a pop-up window appears.



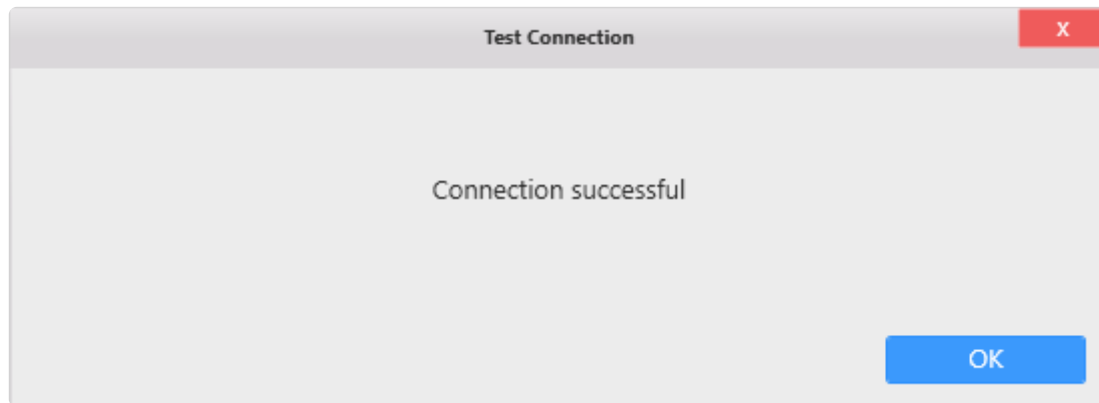
Editing Connection

As with other Connection types, the Historian Connection can be modified later, see [the corresponding chapter](#).

Last modified: 2019/07/26

2.4.1.5. Test connection

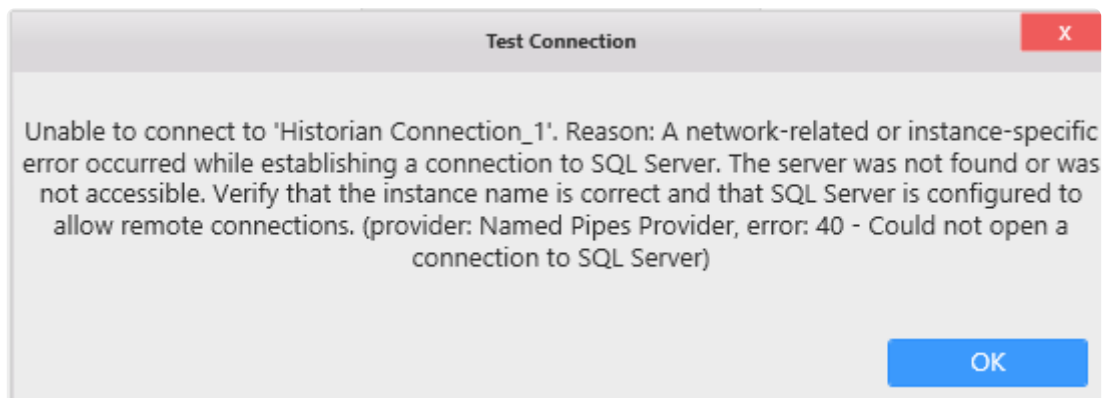
To test the connection for validity, click *Test Connection* button. The following message will confirm its validity.



If the connection is invalid, an appropriate error message will be displayed instead.

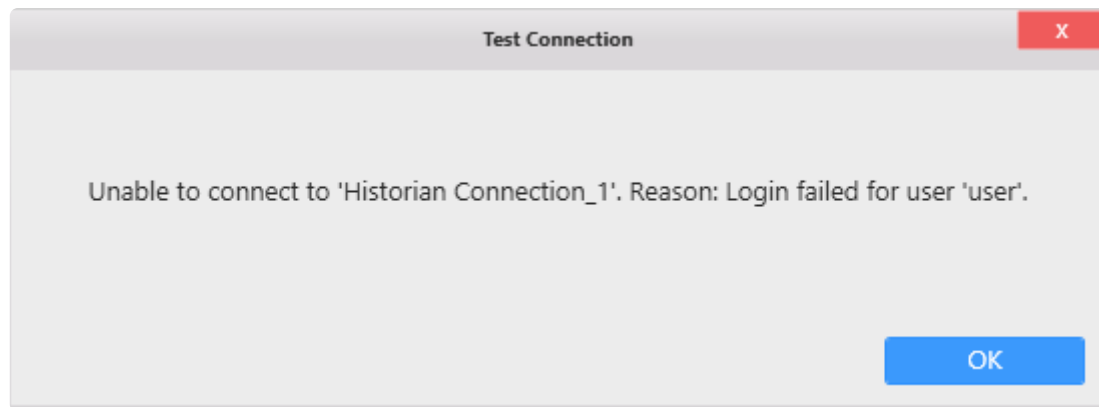
Then you will need to fix the Connection parameters (credentials, server name, ...) or check network connectivity (is the SQL Server accessible from here ?)

The server was not found



Usually after a long timeout, if the error message says "Unable to connect [...] The server was not found or was not accessible", check that the server is accessible from this machine on that port.

Login failed



If the message says “login failed for user”, then Alpana successfully connected to the server, but the credentials were invalid.

Check if that SQL user exist and the password is correct.

Last modified: 2019/05/15

2.4.1.6. Data refresh and Buffer

Since Alpana v3.0, the Wonderware Historian Connector is **not bufferized**.

Of course, like other data Connectors, it may become bufferized when used in Multi-Connection Data Sources.

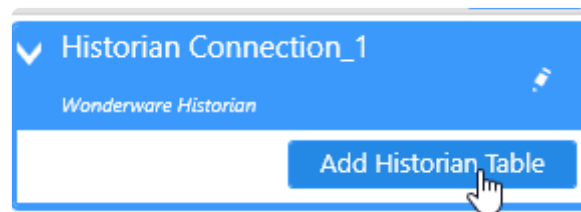
See [the corresponding chapter](#).

Last modified: 2019/06/17

2.4.2. Creating Queries to Wonderware Historian

Once the Connection to Wonderware Historian is created, you can re-use it to make several different queries.

Select the desired Historian Connection, and click on *Add Historian Table* :



This will pop-up the *configure Historian Query* dialog.

When the query configuration is complete, a query Table is added to the current Data Source.

Last modified: 2019/05/15

2.4.2.1. Picking Tags

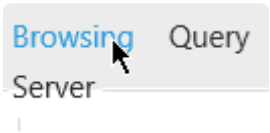
The first part of the configuration is selecting Historian tags.
This is documented in the next chapters.

Last modified: 2019/05/15

2.4.2.1.1. Browsing for Tags

Start browsing for Tags

Click on the “Browsing” tab (open by default) to start browsing for tags to pick :

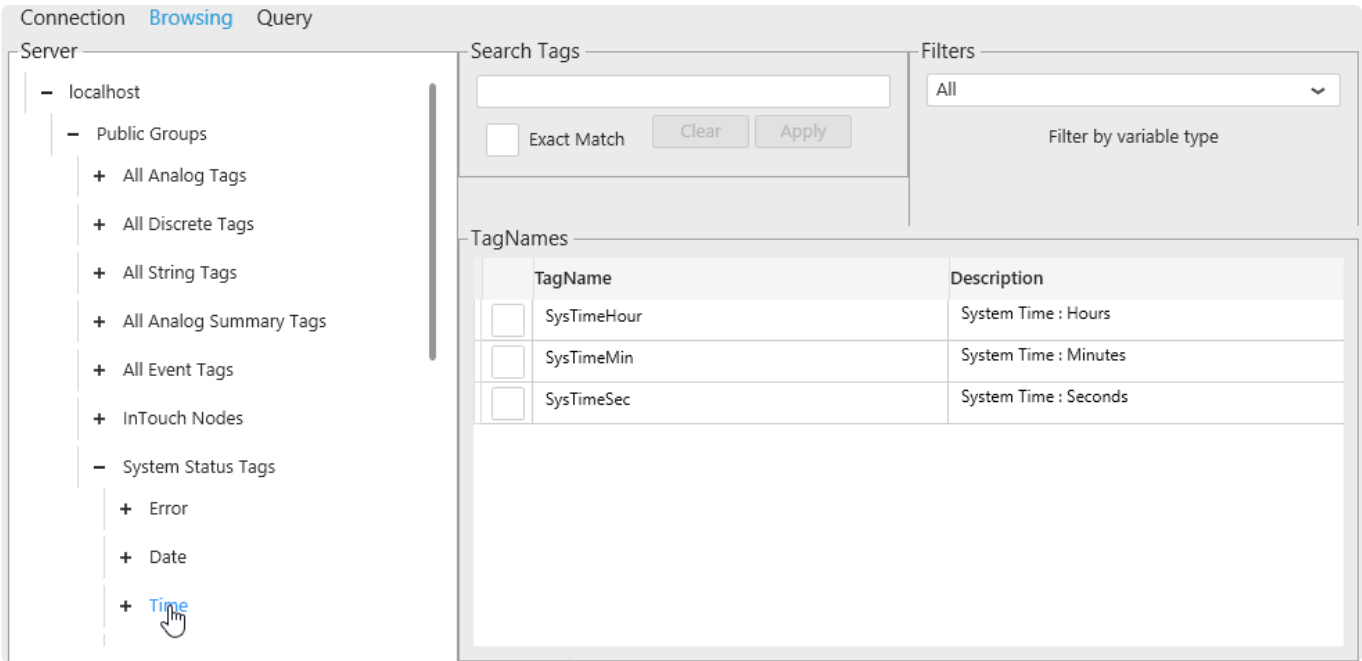


Navigating the Tag hierarchy

The Historian Tag hierarchy is presented in a tree on the left, with Tag Groups as nodes.

Click on a +/- in the tree to unfold/fold a Tag Group.

Click on Group name to view the list of Tags under that Group.



The list of Tags in the currently selected Group is presented in a list on the right.

Filtering the Tag list

* Search and Filter only apply to the currently selected Tag Group. If no Tag Group is selected, then the result is always empty.

Filter by Tag name and Description

In “Search tags”, enter part of a Tag name or Tag description and click “Apply” to filter Tags whose name contain that text.

You can also check for an “Exact Match” only.

Filter by data type

In “Filters”, you can filter the tag list by data type :

Last modified: 2019/05/15

2.4.2.1.2. Selecting Tags



Important

To make a correct Wonderware Historian Query you have to select one tag at least

Select Tags

To select a Tag, you can click on its checkbox to select it. Once a tag is selected, it appears in the Selected Tags panel like below:

	TagName	Description
<input type="checkbox"/>	FCV002.PV	modulating Control Valve
<input checked="" type="checkbox"/>	FCV101.PV	modulating Control Valve
<input type="checkbox"/>	FCV102.PV	modulating Control Valve
<input type="checkbox"/>	FCV201.PV	modulating Control Valve
<input type="checkbox"/>	FCV202.PV	modulating Control Valve
<input type="checkbox"/>	FCV301.PV	modulating Control Valve
<input type="checkbox"/>	FCV302.PV	modulating Control Valve
<input type="checkbox"/>	FCV401.PV	modulating Control Valve
<input type="checkbox"/>	FCV402.PV	modulating Control Valve

Selected Tags

Unselect Tags

You can unselect a Tag by clicking again on its Selected check box or you can click on the cross button in the Selected Tags panel like below:

TagNames

	TagName	Description
<input type="checkbox"/>	FCV002.PV	modulating Control Valve
<input checked="" type="checkbox"/>	FCV101.PV	modulating Control Valve
<input type="checkbox"/>	FCV102.PV	modulating Control Valve
<input type="checkbox"/>	FCV201.PV	modulating Control Valve
<input type="checkbox"/>	FCV202.PV	modulating Control Valve
<input type="checkbox"/>	FCV301.PV	modulating Control Valve
<input type="checkbox"/>	FCV302.PV	modulating Control Valve
<input type="checkbox"/>	FCV401.PV	modulating Control Valve
<input type="checkbox"/>	FCV402.PV	modulating Control Valve
<input type="checkbox"/>	FCV403.PV	modulating Control Valve

Selected Tags

×

FCV101.PV

Last modified: 2019/05/15

2.4.2.2. Configuring the Historian Query

Wonderware Historian allows many ways to query the data.

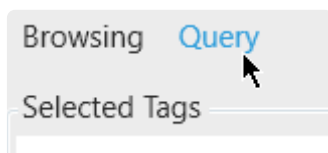
To make users feel at home, Alpana replicates these configuration options in the query configuration.

See the next chapters for details.

Last modified: 2019/05/15

2.4.2.2.1. Going to Query Configuration

To set all the main Wonderware Historian query parameters, click on the Query Tab shown below



Note that the list of currently selected tags appear at the top of the panel.

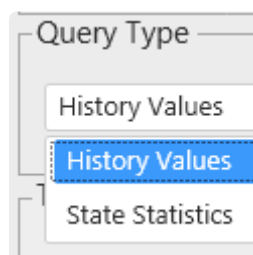
Last modified: 2019/05/15

2.4.2.2.2. Query Type

In Alpana, two Query Types are possible for Wonderware Historian :

- History Values
- State Statistics

You can select the Query Type from the listbox at the top of the Query panel :



History Values

This is the classic Historian Query for showing trends.

A history of Tag values is returned.

State Statistics

This query (sometimes called “State Summary Values”) allows to calculate how much time a Tag had value.

This allows to calculate for example, how long an equipment was in each of its states, and calculate performance indicators.

Last modified: 2019/05/15

2.4.2.2.3. Time Settings

In the panel on the left, you can configure the Time settings for the query

The screenshot shows a 'Time' settings panel. At the top, there's a 'Time Settings' dropdown menu currently set to 'Static Time Settings'. Below this, there are two sections. The first section has a checkbox labeled 'Link to start date' which is unchecked, followed by a date picker showing '15'. Below the date picker is a time selector with fields for 'Time', '00', 'H', '00', 'M', '00', and 'S'. The second section has a dropdown menu labeled 'Interval' set to '5 Minute'. Below this is another checkbox labeled 'Link to end date' which is unchecked, followed by a date picker showing '15'. Below the date picker is another time selector with fields for 'Time', '00', 'H', '00', 'M', '00', and 'S'.

The time interval of the query can be :

- Static : User interaction at runtime cannot change the time interval
 - a [frozen time range](#)
 - a [moving time window](#) always ending **now**
- [Parameter-based](#) : User interaction at runtime can dynamically change the time interval of the query

The screenshot shows the 'Time' settings panel with the 'Time Settings' dropdown menu open. The dropdown menu has three options: 'Static Time Settings' (highlighted in blue), 'Parameter Based Settings', and 'Static Time'.

Last modified: 2019/06/18

2.4.2.2.3.1. Static Time Settings

What are Static Time settings

When using *Static Time Settings*, the start date and end date of the Historian query cannot be changed by user interaction at runtime.

Note that you can still create a moving time frame (see [below](#)).

This allows to make a pre-defined query to show users precisely the data you want to show.

You can still let users filter this data by time, but they cannot explore outside the pre-defined query.

* If you want to let users explore any time frame at runtime, use [Parameter-Based Time Settings](#) instead.

How to configure

Use the panel on the left to configure the Time settings for the query :

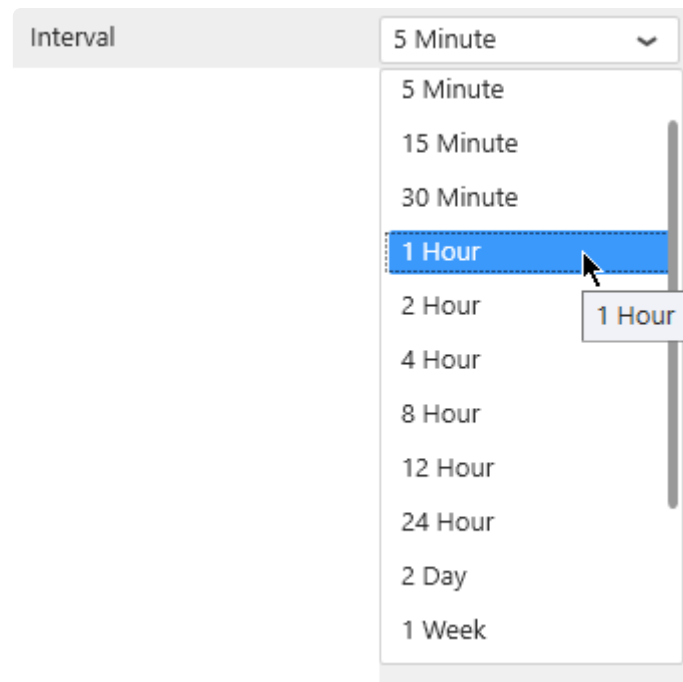
The screenshot shows a configuration panel titled "Time". It contains the following elements:

- A dropdown menu labeled "Time Settings" with "Static Time Settings" selected.
- A checkbox labeled "Link to start date" which is unchecked.
- A date selection field labeled "Select a date" with a calendar icon and the number "15" visible.
- A time selection field labeled "Time" with inputs for "00" hours, "00" minutes, and "00" seconds, followed by an "S" label.
- A horizontal separator line.
- A dropdown menu labeled "Interval" with "5 Minute" selected.
- Another horizontal separator line.
- A checkbox labeled "Link to end date" which is unchecked.
- A date selection field labeled "Select a date" with a calendar icon and the number "15" visible.
- A time selection field labeled "Time" with inputs for "00" hours, "00" minutes, and "00" seconds, followed by an "S" label.

Moving time window

By default, the time range is a moving time window that ends **now** and starts some time ago depending on the *Interval*.

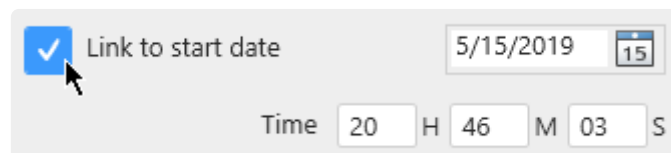
You can set the time Interval by using the corresponding listbox :



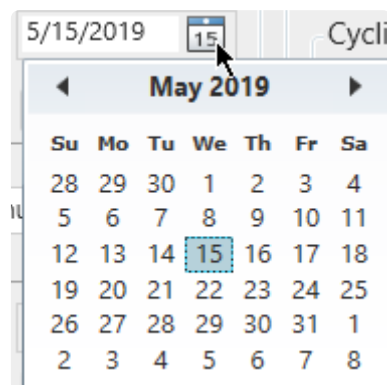
Frozen time range

Instead, you can choose to freeze a time range : the data returned will always be between a pre-defined *start date* and *end date*.

You have the possibility to freeze the *start date* or the *end date* here by checking it like below the desired checkbox :



Then you can configure the Datetime from the text box or by clicking on the calendar :



You can set the time by typing it in the following input :

Time	<input type="text" value="20"/>	H	<input type="text" value="46"/>	M	<input type="text" value="03"/>	S
------	---------------------------------	---	---------------------------------	---	---------------------------------	---



Note :

If you select both, the interval option will be disabled

Last modified: 2019/06/18

2.4.2.2.3.2. Parameter Based Settings

What are Parameter-Based Time Settings

With *Parameter-Based Time Settings*, the start date and end date of the Historian query are contained in a [Alpana Parameter](#).

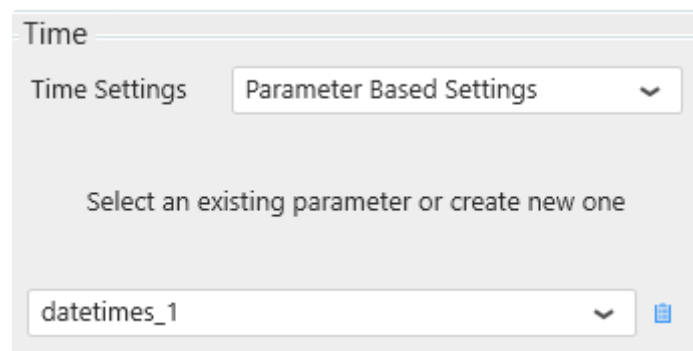
This means that the values can be changed at runtime using User interaction (filters, URL parameters).

✿ This is useful for letting users explore any time range in the Historian data.

How to configure

! To configure a parameter-based Historian query, the [Alpana Parameter](#) containing the values must already exist and have multiple values (exactly 2 values) : the start date and end date.

Use the panel on the left to select an existing Parameter from the dropdown list, or create a new Parameter using the shortcut button :



Time

Time Settings Parameter Based Settings

Select an existing parameter or create new one

datetimes_1

Hint

A recommended way to proceed is to :

1. have a separate [DataSource](#) with field containing a list of possible dates/times for the user to select from (this is easy to do in Excel, CSV, SQL)
2. create a [Master Widget](#) for the user to select the date/time, and configure it with the above DataSource
3. make a [default selection](#) in the Master Widget, possibly using [relative time default selection](#) (if the

Widget type allows it), so that the Parameter always has a value

4. configure the Historian query with Parameter-Based Time Settings, using the Parameter that was automatically generated by the Master Widget

Last modified: 2019/06/18

2.4.2.2.4. History Values : query parameters

Introduction

The History Values query has many parameters to configure.

Below is a brief description on how to configure them.

For more information, please see the corresponding help topics in the Wonderware Historian documentation.

In the *Retrieval* part of the window you can set all retrieval parameters :

Retrieval

Main Options Other

Retrieval Mode: Cyclic

Query Row Limit: First 0 rows

Cyclic Attributes

☒ Values over equal time intervals 100

☐ Values spaced every 0 s

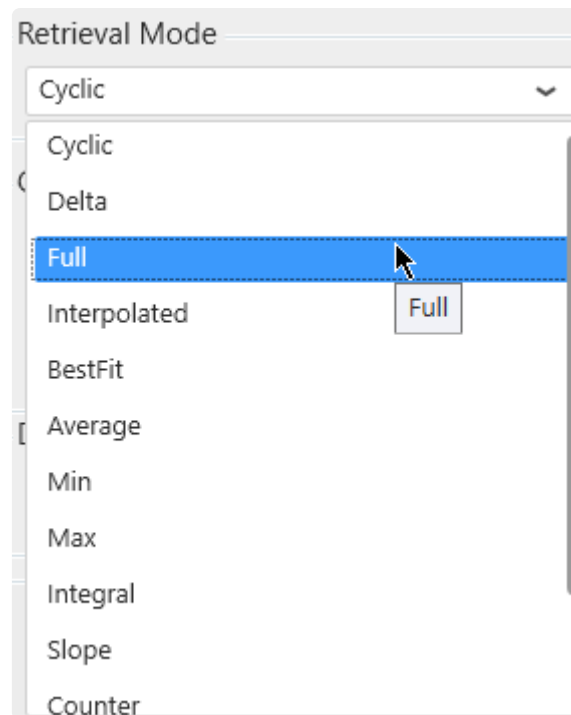
Interpolation Type Tag Setting

Delta Retrieval Deadband

Time 0 ms Value 0.00 %

Retrieval mode

You can select a *Retrieval Mode* (Cyclic is the default) :




All Wonderware Historian retrieval modes are available :

- Cyclic
- Delta
- Full
- Interpolated
- BestFit
- Average
- Min
- Max
- Integral
- Slope
- Counter
- ValueState
- RoundTrip

Query row limit

A *Query Row Limit* can be set :

Query Row Limit


First  rows


This allows to limit the number of data rows returned.

Cyclic Attributes


The *Cyclic Attributes* can be configured :

Cyclic Attributes

☒ Values over equal time intervals 

☐ Values spaced every  s

Interpolation Type

Tag Setting 





Note
The Cyclic panel control is accessible only when one of the following Retrieval Modes is selected: Cyclic, BestFit, Average, Min, Max, Integral, slope, Counter, ValueState, Roundtrip

Delta Attributes

The *Delta Retrieval Deadband* can be configured :

Delta Retrieval Deadband

Time  ms Value  %



Note
The Delta panel control is accessible only when *Delta* Retrieval Mode is selected.

Other Attributes

To display the other retrieval options click on the *Other* button :

Retrieval

Main Options **Other**

History Version

Latest

Rules

Time Stamp Server default

Values to include in Calculations Good and uncertain quality

State retrieval

State Calculation

☐ Specify State

Transformation

No transformation

Here, you can configure the History Version, the Rules Settings, the State Retrieval (available depending the retrieval mode chosen) and the Transformation options.

Limitation



Note

The Phantom Cycle : Do not include boundary values feature is currently not supported

Last modified: 2019/05/15

2.4.2.2.5. History Values: query Format

As with the Wonderware Historian Query client, the Alpana query to Historian is by default in *Narrow* format.

<input checked="" type="radio"/> Narrow query format			<input type="radio"/> Wide query format		
TagName	DateTime	vValue	DateTime	SysCPU0	SysCPU1
SysCPU 0	1/20/2005	2	1/20/2005	0	2
SysCPU 1	1/20/2005	3	1/20/2005	0	5
SysCPU 2	1/20/2005	0	1/20/2005	0	5

Alternately, you can select the **Wide** Query Format by checking the checkbox under *Format* :

Format
 ☒ Use Wide Query

* Note : This is only available when using the *History Values* Query type.

Instead of the default narrow query, it will act as a pivot table to compare your selected tags into different columns over time.

* This is *very useful* to achieve some widget configurations.
For example, it will allow you to display 2 different tags on 2 different Y Axis on the same Chart.

Last modified: 2019/05/15

2.4.2.2.6. State Statistics : query parameters

Parameters

Introduction

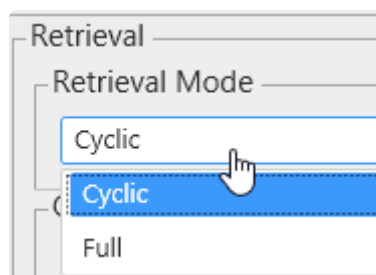
The State Statistics query has a few parameters to configure.

Below is a brief description on how to configure them.

For more information, please see the corresponding help topics in the Wonderware Historian documentation.

Retrieval Mode

The Retrieval Mode can be changed with the corresponding listbox :

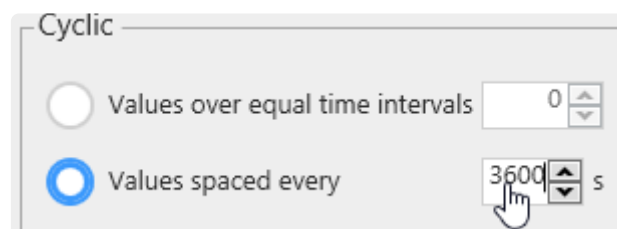


The following Retrieval Modes are available :

- Cyclic
- Full

Cyclic parameters

For Cyclic retrieval, the value spacing can be selected from fixed number or fixed duration :



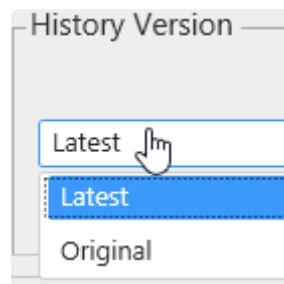
Full parameters

For Full retrieval, a row limit can be set :



History Version

The History Version listbox allows to select “Latest” or “Original” :



Common usage example

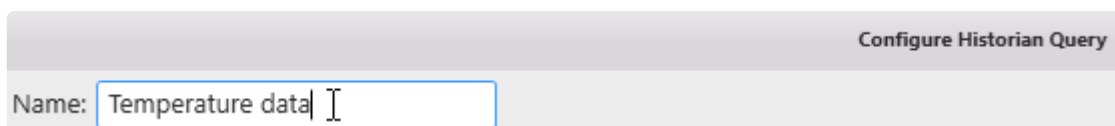
A common usage is to set Cyclic retrieval with values spaced over 1h or 1 day, and a start time at a full hour like midnight.

This will return state statistics with hourly (or daily) information of how much time (total, %, ...) was spent in each state.

Last modified: 2019/05/15

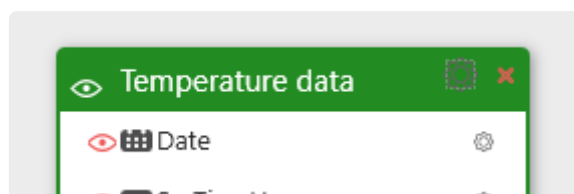
2.4.2.3. Renaming the Query Table

The Query Table can be renamed so that it has a meaningful name inside the Data Source configuration :



Configure Historian Query

Name: Temperature data |



Last modified: 2019/05/15

2.4.3. Wonderware Historian query result

After configuring the query, Alpana is able to send it to the Wonderware Historian server and get the result.

This result will be modeled as a Table inside the current Data Source.

For details about the schema and data of this query result, see the next chapters.

Last modified: 2019/05/15

2.4.3.1. Connection Schema

The Wonderware Historian Connection presents its Result in a table which can be [renamed at the query level](#).

Data Fields

The fields of the Historian table depend on the query Type.

For a detailed description of these fields and their meaning, see the corresponding documentation from Wonderware.

History Values

For a History Values query in [narrow Format](#) (the default), the following fields are returned :

- TagName
- Date
- Value
- vValue
- OPCQuality
- QualityDetail
- QualityDescription

For a History Values query in [Wide Format](#), the following fields are returned :

- Date
- (first tag)
- (second tag)
- (...)
- wwResolution

State Statistics

For a State Statistics query, the following fields are returned :

- TagName
- StartDateTime
- EndDateTime
- Value

- vValue
- OPCQuality
- StateCount
- StateTimeMin
- StateTimeMax
- StateTimeAvg
- StateTimeTotal
- StateTimePercent

Last modified: 2019/05/15

2.4.3.2. Important : data limits



Important

In order to limit performance issues for huge Historian queries, some limitations have been set.

The Wonderware Historian Connector has the following intentional limitations :

- 10000 max data rows per query
- 30s timeout

When a Historian query returns more than 10000 results, it returns the first 10000.

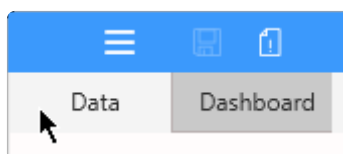
Last modified: 2019/05/15

2.5. Transforming Data

This Chapter is a reference on how to create and configure a Data Source to prepare data for your Widgets and Dashboard.

To understand the [Alpana Data Model](#), see the corresponding chapter in this documentation.

Since Alpana Designer v3.0, all data preparation is made inside the *Data* tab.



Last modified: 2019/06/17

2.5.1. Introduction

Data Sources in the Dashboard design workflow

You don't need to be able to create a Data Source in order to create a Dashboard : someone may do it for you !

The workflow to design Dashboards is built around the idea that Dashboards should be easy to create and edit. As such, the data preparation steps can be made by another person and shared to you.

Example :

- User1 needs to make a dashboard but doesn't have the skills or credentials to prepare data
- User2 is skilled in IT/SQL/data/... and has the credentials to access the data, but has no process knowledge and won't know how to make the Dashboard. She creates and shares a DataSource.syds file to User1

In this Course, you will learn all about Data Sources and how to share them with your "User1" if needed.

Last modified: 2019/04/16

2.5.1.1. Learning Pre-Requisites

To create Data Sources, you will need to connect to your data first. Please see the corresponding chapter : [Connecting to Data](#).

More generally, it is also important to understand the [Alpana data model](#).

Last modified: 2019/04/16

2.5.1.2. What are Data Sources

In Alpana, data preparation happens in **Data Sources**.

Each Data Source prepares a query to the original data to be presented inside Widgets.

Data is fetched through the Alpana Connections, see Course [Connecting to Data](#).

Each Widget displays data from a single Data Source (but a Data Source can be re-used in any number of Widgets).

Alpana Designer provides a graphical editor that allows to make this data preparation using simple mouse interaction.

Last modified: 2019/04/16

2.5.1.3. Content : Data Preparation Activities

The main data preparation activities in Alpana Designer include :

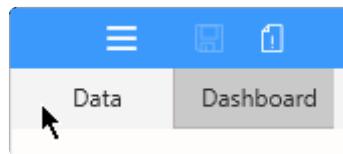
- [Fetch](#) : select an object (table, view, stored procedure, ...) from which to get data
- [Merge](#) : join tables from one or more Connections
- [Filter](#) : remove data based on multiple criteria
- [Calculate](#) : create Expressions to make calculations
- [Format](#) : format, rename and hide fields to make it easier for another user to re-use the Data Source.
- [Share](#) : export/import Data Sources

Last modified: 2019/04/16

2.5.2. Creating and Managing Data Sources

This chapter describes how to create and manage Data Sources.

Since Alpana Designer v3.0, all data preparation is made inside the *Data* activity tab.



Last modified: 2019/06/17

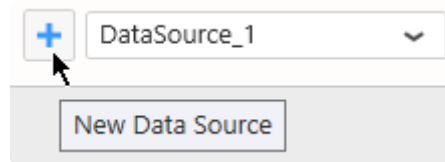
2.5.2.1. Create a Data Source

New Dashboard : first Data Source

When creating a new Dashboard, a new Data Source called “DataSource_1” is created and opened in Alpana Designer.

Creating a Data Source

After creating a Connection, you can create a Data Source by clicking the **+** button next to the Data Sources list in the top bar :



Then, the new Data Source is automatically open for edition.

Data Source storage

A Data Source contains only the information on how to prepare the data. It does not contain the data itself.

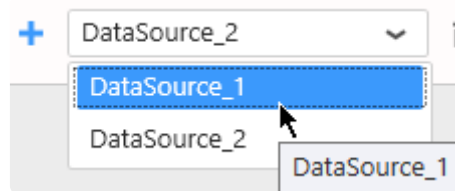
When a Dashboard is saved, all the Data Sources are saved inside.

You can create as many Data Sources as needed.

Last modified: 2019/04/17

2.5.2.2. Browsing Data Sources to Edit

To browse Data Sources, open the Data Sources list in the top bar, and select by name :



All the actions to actually edit the Data Source are explained in the following chapters of this Course.

Last modified: 2019/04/17

2.5.2.3. Rename a Data Source

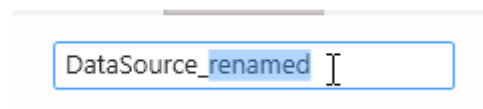
Why rename a Data Source?

The Data Source name will appear in widget data binding.

Giving good names can help the user who will make the Dashboard based on this Data Source.

How

Use the text box on the left of the top bar to rename the currently open Data Source :



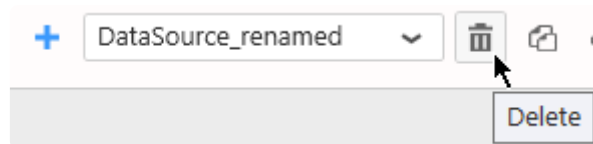
The new name is immediately validated as you type it.

Last modified: 2019/04/17

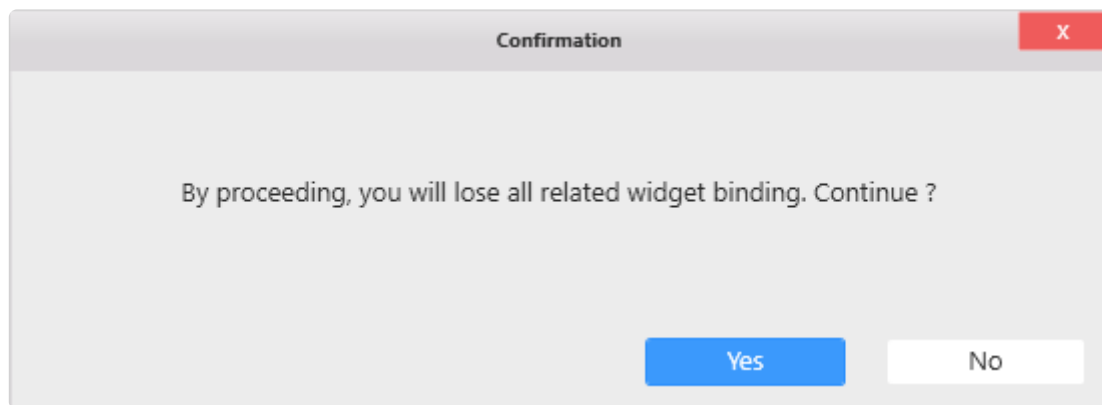
2.5.2.4. Delete a Data Source

How to Delete a Data Source

To Delete an existing Data Source, left click then click on the trash icon in the top bar :



If the Data Source is used by any Widget, the following message will appear :



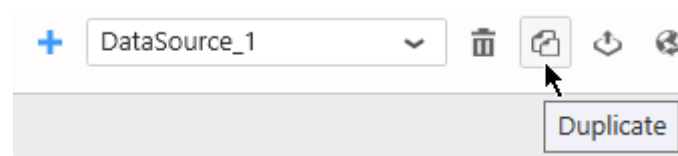
If you choose “Yes”, the Data Source is deleted and all dependent Widgets lose their data binding.

Last modified: 2019/04/17

2.5.2.5. Duplicate a Data Source

You can Duplicate an existing Data Source to start working on a copy.

In the top bar, click on the Duplicate_ icon :



A copy of your Data Source is created, and you can start working on it independently.

Last modified: 2019/04/17

2.5.3. Export / Import Data Sources

Sharing Data Sources allows to collaborate between users with different skill sets : an IT engineer creates a Data Source, exports it, then a process engineer only has to import it and start creating visualizations.

This chapter describes how to proceed.

Last modified: 2019/04/25

2.5.3.1. What is a Data Source Export

A Data Source can be exported to a file with format `*.alps`.

This DataSource.syds file contains the definition of the Data Source and all the dependent Connections, including their credentials.

This is useful for splitting roles in Dashboard creation :

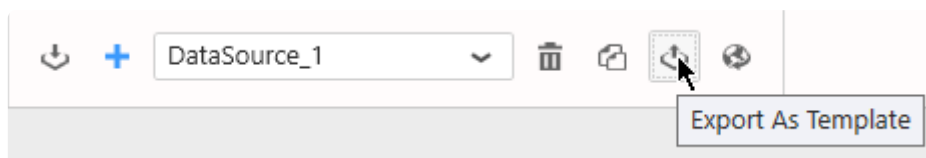
- User1 needs to make a dashboard but doesn't have the skill or credentials to prepare data
- User2 is skilled in IT/SQL/data/... and has the credentials to access the data, but has no process knowledge and won't know how to make the Dashboard. She creates and shares a DataSource.alps file to User1

Last modified: 2019/03/14

2.5.3.2. Export a Data Source to a file

How to Export to a file

In the top bar, click the *Export as Template* icon :



Select a file name and save locally.

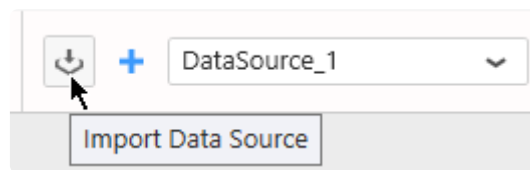
Last modified: 2019/04/17

2.5.3.3. Import a Data Source from a file

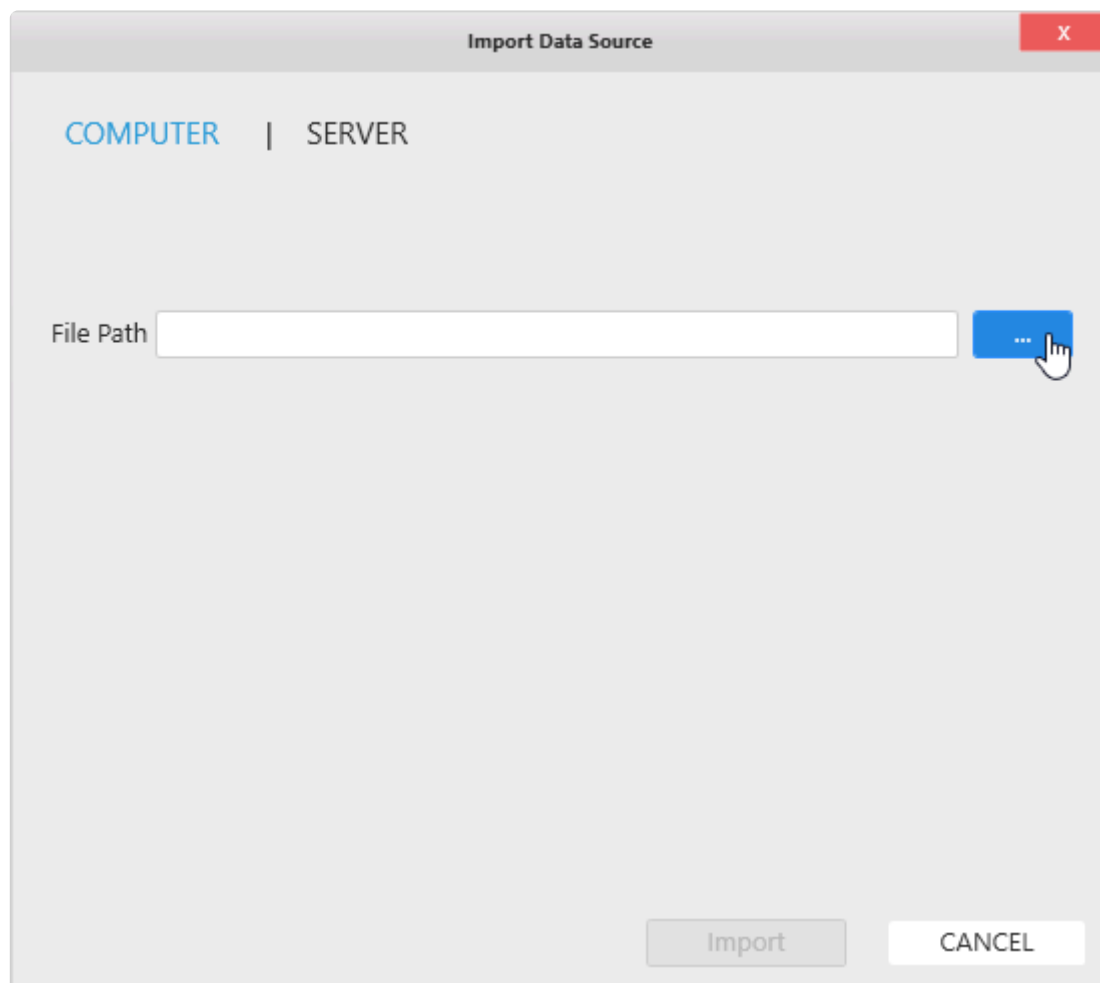
A Data Source can be imported from a file in a few ways :

Import in an existing Dashboard

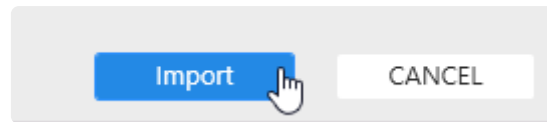
Click on the “Import” button in the Data Sources list :



In the “Import Data Source” dialog, under “Computer”, click “...” to browse for a DataSource.alps file :



Then click “Import” :



The Data Source is now imported in your current Dashboard.

Create a new Dashboard based on a DataSource file

In Windows Explorer, you can double click or open the DataSource.alps file with Alpana Designer.

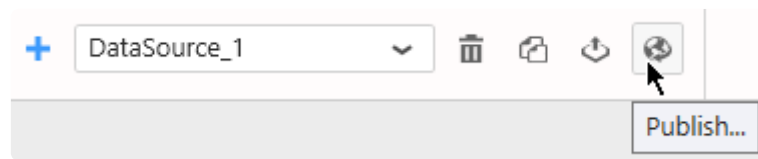
This creates a new Dashboard with the corresponding Data Source imported.

Last modified: 2019/04/17

2.5.3.4. Publish a Data Source to Alpana Server

* In order to Publish a Data Source to Alpana Server, you need to be **logged in** from Alpana Designer to that Alpana Server, and have the necessary **permissions** to create Data Sources.

In the top bar, click *Publish* :



In the “Publish” dialog, fill in details and click “Publish” :

A screenshot of the 'Publish' dialog box. The dialog has a title bar with 'Publish' and a close button (X). It contains three input fields: 'Datasource name' with 'DataSource_1' entered, 'Description' (empty), and 'Version Comment' (empty). At the bottom, there are two buttons: 'Publish' (blue) and 'CANCEL' (white). A mouse cursor is clicking on the 'Publish' button.

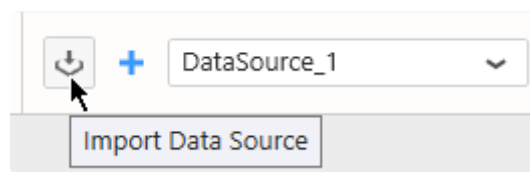
The Data Source is now published to Alpana Server and available to other users depending on permissions.

Last modified: 2019/04/26

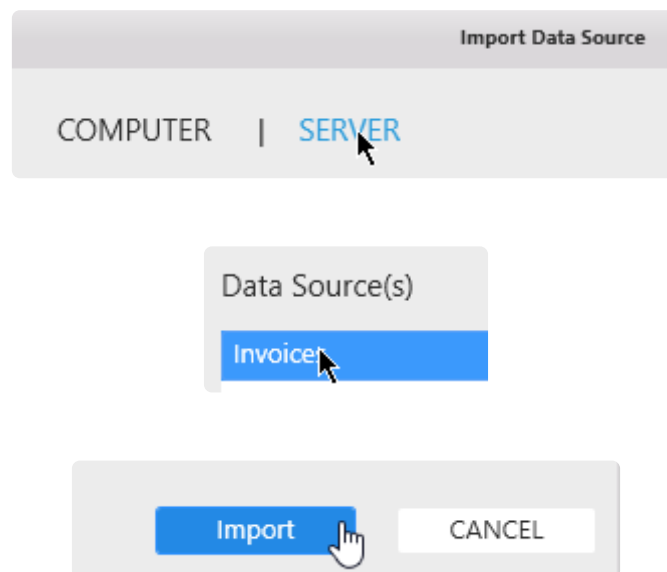
2.5.3.5. Import a Data Source from Alpana Server

* In order to Import a published Data Source from Alpana Server, you need to be logged in from Alpana Designer to that Alpana Server, and have the necessary permissions to access the Data Source.

Click on the “Import” button in the Data Sources list :



In the “Import Data Source” dialog, click “Server”, select a published Data Source, and click Import :



The Data Source is now imported in your current Dashboard.

Last modified: 2019/04/25

2.5.4. Fetching Data

Introduction

A Data Source can fetch data from Connections by selecting the corresponding object in the Connection schema :

- a data Table
- a data View
- a Stored Procedure
- a Table-Valued Function
- an Alpana Custom Query Table (see next Chapter [Custom Query Tables](#))

For details on how these elements exist or not based on the Connection type, please see the dedicated Course [Connecting to Data](#).

Pre-requisite

In this Chapter, we will edit a Data Source.

To learn how to create or edit a Data Source, see the previous Chapter [Creating and Managing Data Sources](#).

In this Chapter, it is assumed that you have created a Data Source and that it is open for edition.

Last modified: 2019/04/17

2.5.4.1. using the Data Preview

Previewing data

When the Data Source definition is updated, the bottom main area shows a data preview :

Data Preview		Interactions in this grid will only be reflected in the preview and will not affect the underlying data				Rows Count 100	Load More
continent	Country	FactoryName	FactorySize	ProfitObjective	RatingO		
Europe	Denmark	Viborg	8	6100000	2.4200		
Europe	Denmark	Viborg	8	6100000	2.4200		
Europe	Denmark	Viborg	8	6100000	2.4200		
Europe	Denmark	Viborg	8	6100000	2.4200		

Loading more data rows

In order to improve performance while editing the Data Source, only a few data rows are fetched.

By default, only the first 100 data rows are fetched.

If your data set contains more than 100 rows, you can click the “Load More” button on the top right to get up to 1000 rows.

Then if your data set exceeds 1000 rows, the Data Preview will still limit to 1000 rows :

Data Preview		Interactions in this grid will only be reflected in the preview and will not affect the underlying data				Rows Count 1000
FactoryName	FactorySize	ProfitObjective	RatingObjective	Enabled	Da	
Beijing	50	17600000	8.19999980926514	True	1/1/20	
Beijing	50	17600000	8.19999980926514	True	1/2/20	
Beijing	50	17600000	8.19999980926514	True	1/3/20	
Beijing	50	17600000	8.19999980926514	True	1/4/20	

Refreshing the Data Preview

You can force to get new data by clicking the Refresh button on the top toolbar :



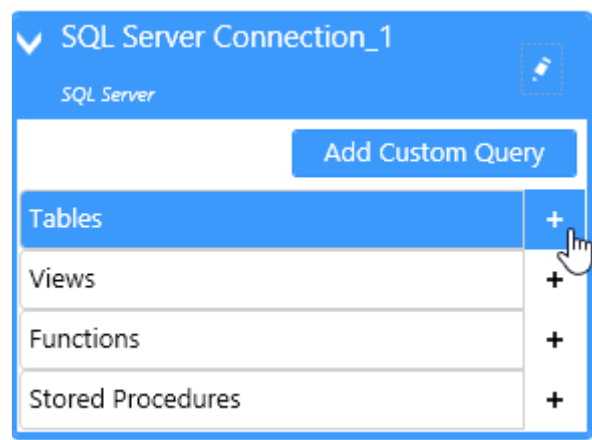
Last modified: 2019/03/13

2.5.4.2. getting and browsing the Connection schema

Browsing the Connection schema

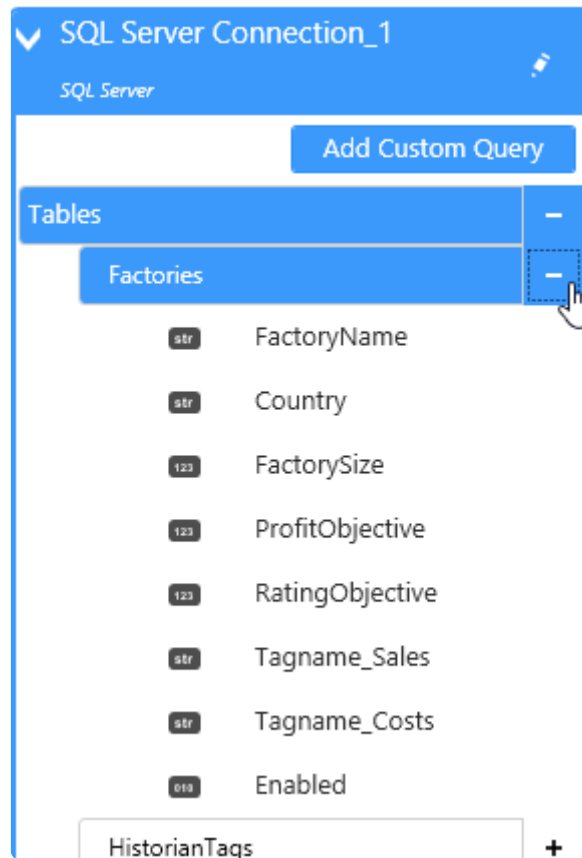
In order to get data from an object in the Connection, you will need to browse the Connection schema.

To do so, click on the Connection on the left, and use the + icons to navigate the schema tree :







Showing the fields of an object

Clicking one last time the + icon will show the list of fields in that object :



Field Type icon

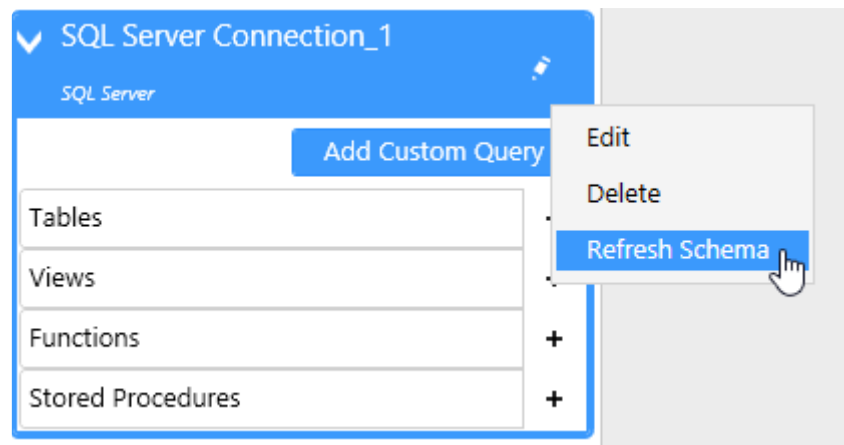
Next to each field is an icon showing the data type of that field :

-  : string
-  : number
-  : date
-  : boolean

Refreshing the Connection schema

If the Connection schema has changed since you started browsing (maybe a table was added or removed, etc...), you will need to refresh it manually.

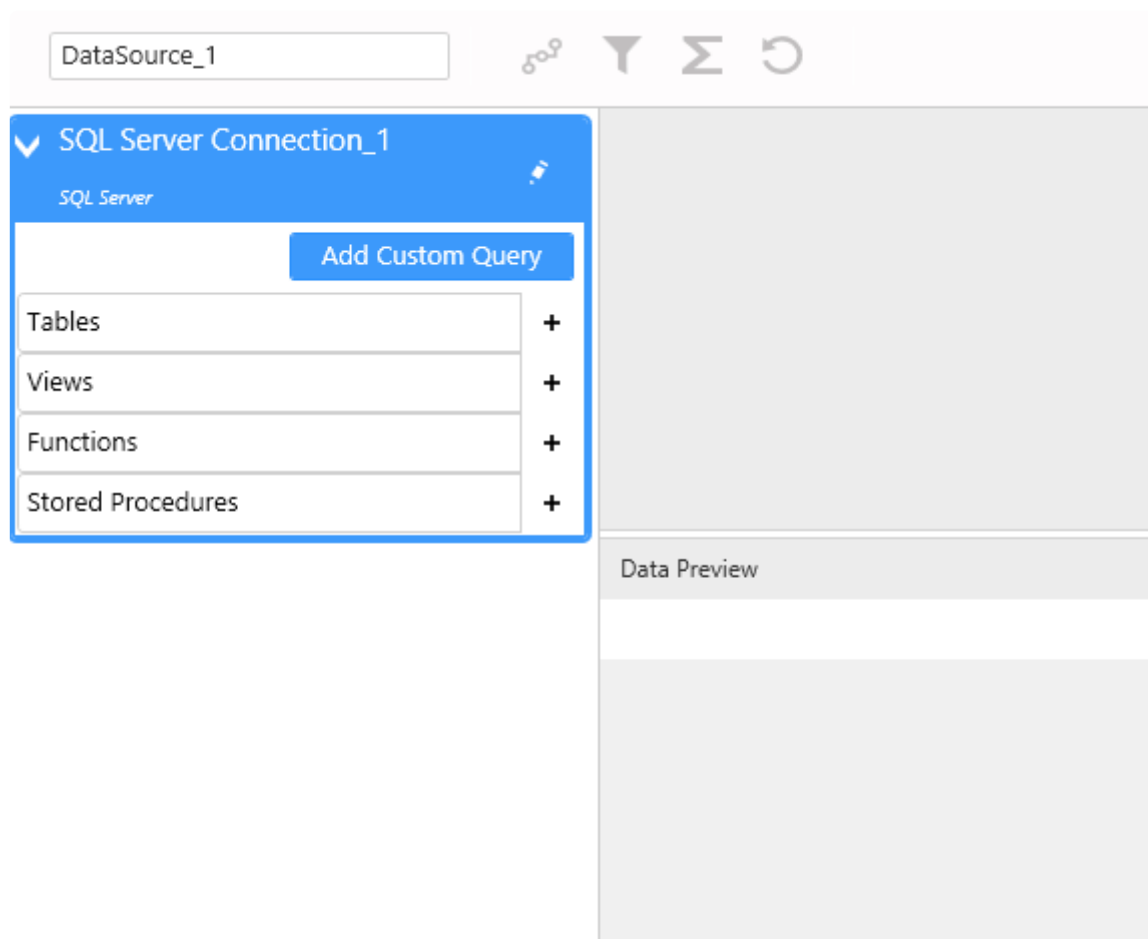
To do so, click on the pen icon on the Connection on the left, and select “Refresh Schema” :



Last modified: 2019/03/13

2.5.4.3. fetching data from a data Table

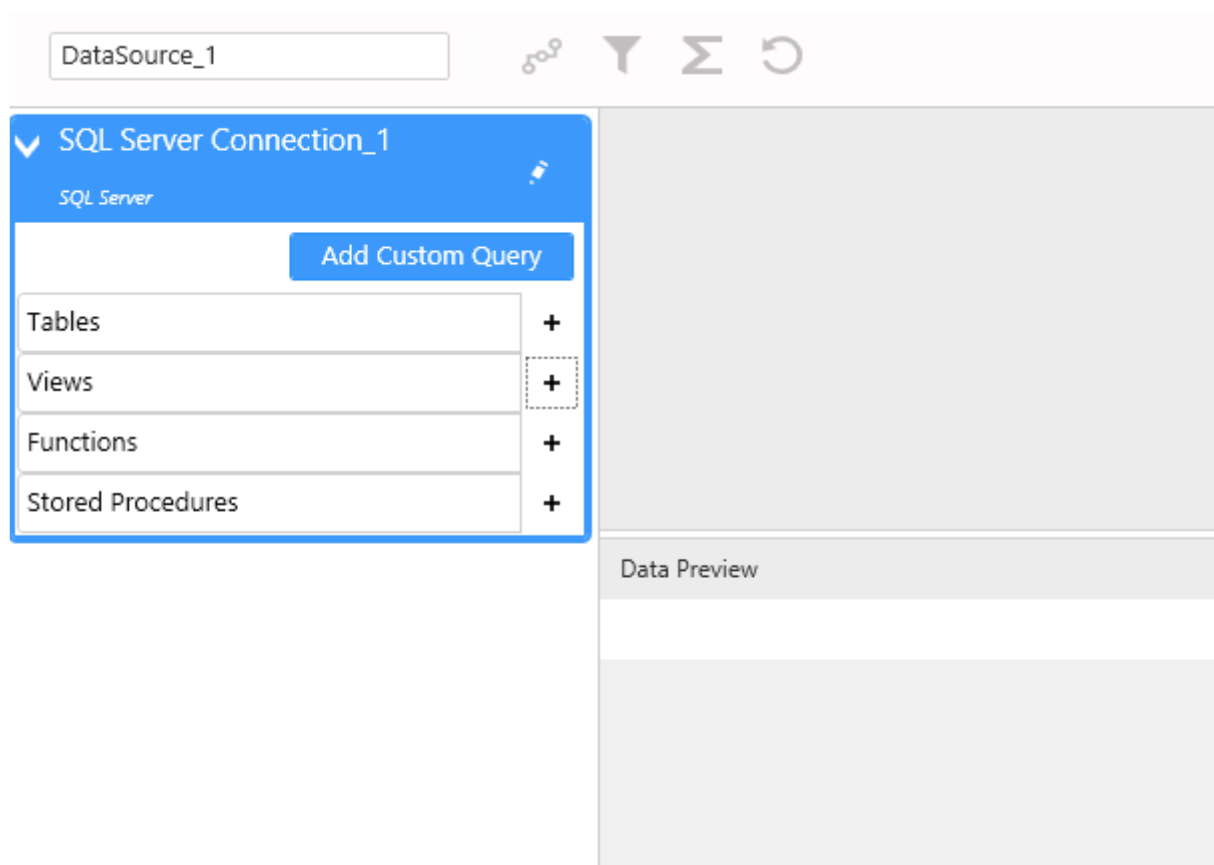
To get data from a data Table, browse to the desired Table in the schema and drag it to the top main area :



Last modified: 2019/03/13

2.5.4.4. fetching data from a data View

To get data from a data View, browse to the desired View in the schema and drag it to the top main area :



Last modified: 2019/03/13

2.5.4.5. fetching data from a Stored Procedure

Important notice

Stored Procedure can perform destructive actions when called : deleting tables, inserting data, etc...

In Alpana, the Stored Procedure will be called every time data is fetched : every time anyone loads a dashboard or refreshes a filter pointing to such a Data Source, the Stored Procedure will be executed.



Before executing a Stored Procedure, please make sure that it doesn't cause unwanted side effects.

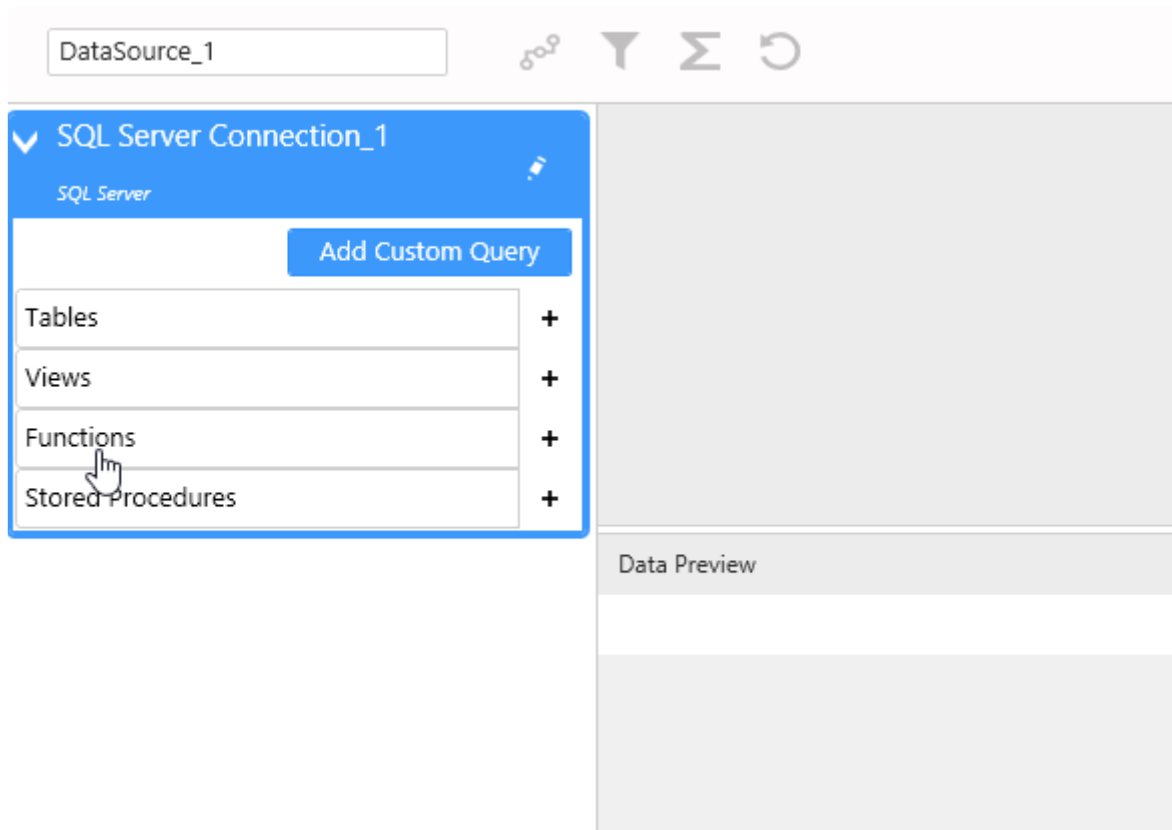
If necessary, instead of calling existing Stored Procedures, you can :

- create Table-Valued Functions or Stored Procedures that only SELECT data and don't make changes to the data.
- create dedicated Stored Procedures for your dashboarding and reporting tools, or Alpana specifically, and make sure that they are safe

Ask your database administrator for advice.

Stored Procedure without parameters

To get data from a Stored Procedure, browse to the desired Stored Procedure in the schema and drag it to the top main area :



If the Stored procedure takes no input parameters, then that's all.

If the Stored procedure does take input parameters, then please proceed below :

Stored Procedure with parameters

If the Stored procedure takes input parameters, a dialog appears to let you set the parameters. It lists the input parameters and their data type :

Parameter	Type	Value		
@year	int	<input type="text"/>	<input type="checkbox"/>	Use Parameter
@factory	varchar(50)	<input type="text"/>	<input type="checkbox"/>	Use Parameter

OK CANCEL

Stored Procedure with static parameters

To fill in static values for the Stored Procedure parameters, simply fill the text boxes and click “OK” :

Parameter	Type	Value		
@year	int	2018	<input type="checkbox"/>	Use Parameter
@factory	varchar(50)	Beijing	<input type="checkbox"/>	Use Parameter

OK CANCEL

With a static value, the Stored Procedure will always be called with the value you entered.

Stored Procedure with dynamic parameters

Instead, you can bind a Stored Procedure parameter to an Alpana Dashboard Parameter (see the dedicated Course Chapter on [Parameters](#)).

This will allow you for example to change the Stored Procedure call depending on Master Widget clicks on the dashboard.

To do so, in the “Stored Procedure Parameters Editor” dialog, check “Use Parameter” and select an existing Alpana Dashboard Parameter :

Parameter	Type	Value	Use Parameter
@year	int	2018	<input type="checkbox"/>
@factory	varchar(50)	param_Factory	<input checked="" type="checkbox"/>

You can also create a Parameter directly form this dialog by clicking the parameter list button :



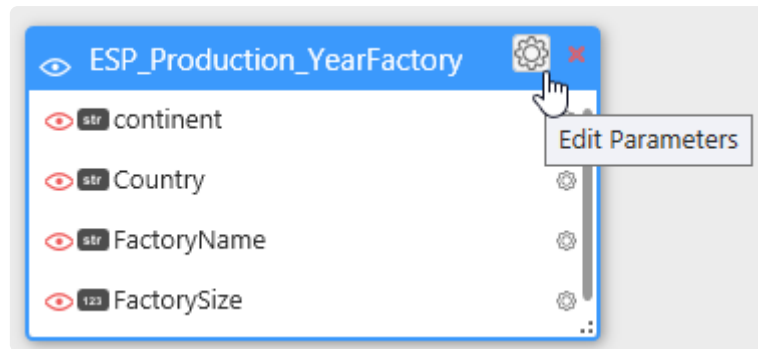
Note :

Be careful that some Stored Procedures require non empty parameters. In order to complete this step, you may have to set a [default value to the Alpana Parameter](#) :

Name	param_Factory	Type	String
Default Value	Paris		<input type="checkbox"/> Return Multiple Values

Editing existing Stored Procedure parameter binding

Once the Stored Procedure parameters are bound, you can change them at any time by clicking the “Edit Parameters” cog icon on the corresponding table :



Stored procedure limitations

SQL Stored Procedures can do a lot of things, and some of them don't even return data.

Alpana can only deal with certain types of stored procedures :

Single data set

Alpana supports stored procedures that return a single data set.

Stored procedures that return several data sets are not supported. For example :

```
SELECT 123; -- first data set
SELECT 465; -- second data set
```

... is not supported.

No output parameters

Stored procedures that use output parameters are not supported. For example :

```
CREATE PROCEDURE output_param
@outparam INT OUTPUT
AS
BEGIN
SELECT @outparam = 123;
END
```

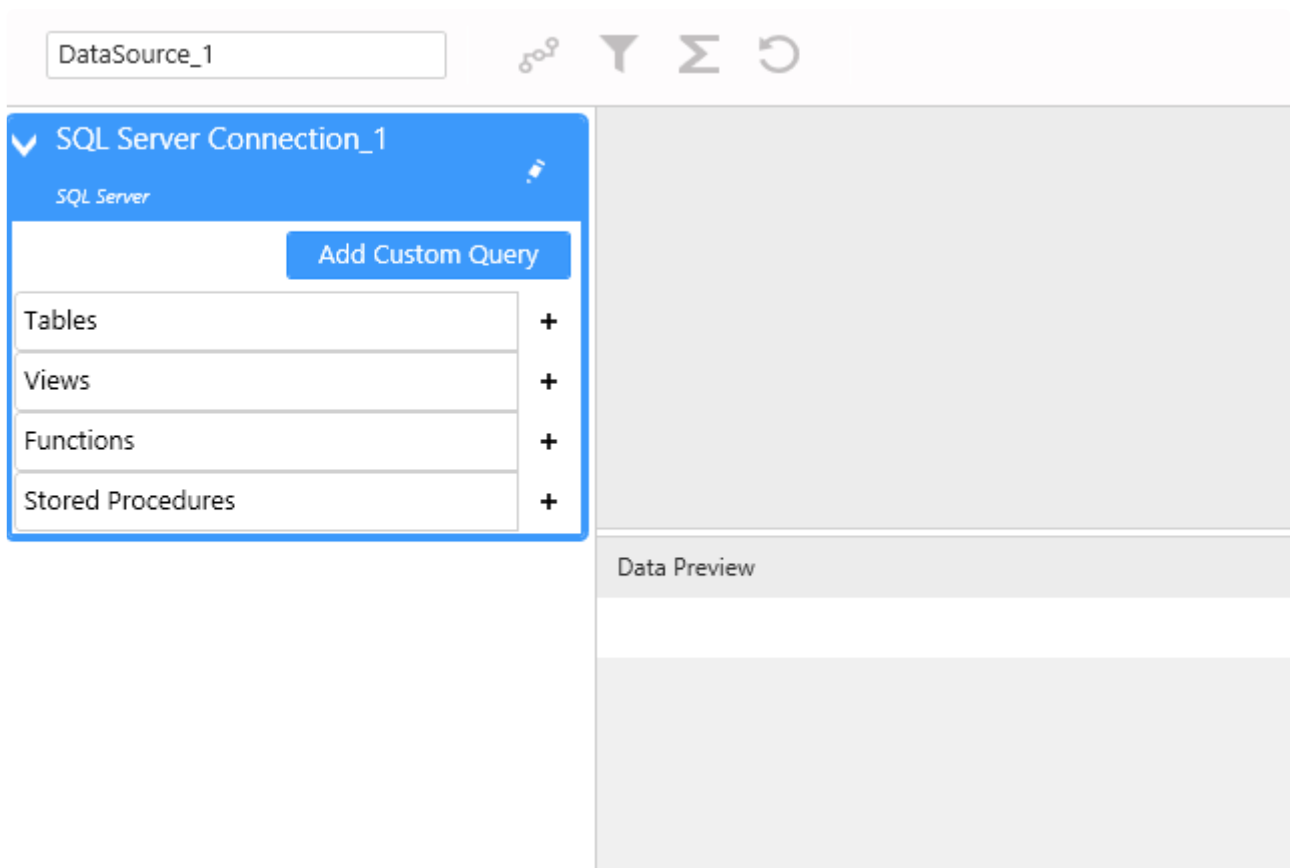
... is not supported.

Last modified: 2019/03/13

2.5.4.6. fetching data from a Table-Valued Function

Table-Valued Functions without parameters

To get data from a Table-Valued Function, browse to the desired Table-Valued Function in the schema and drag it to the top main area :

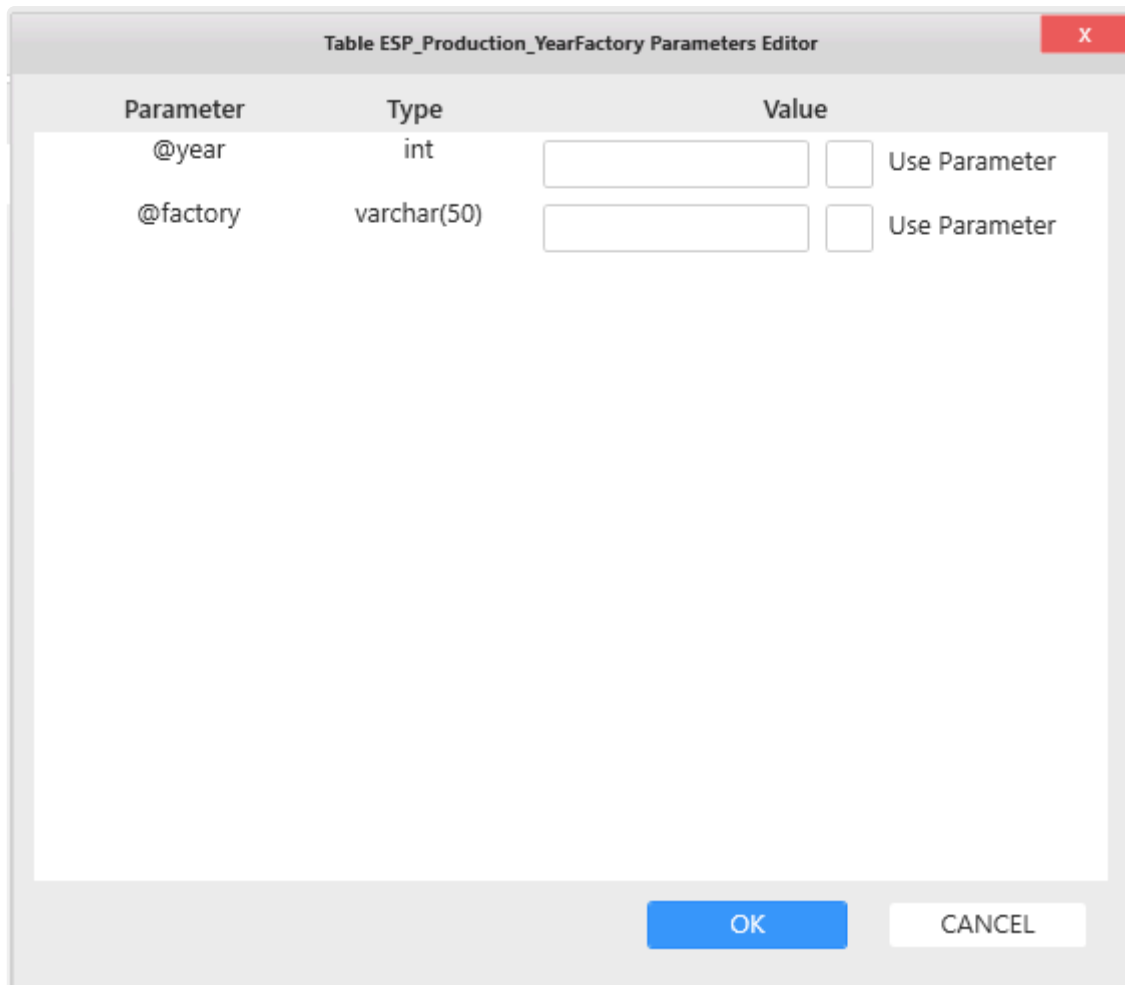


If the Table-Valued Function takes no input parameters, then that's all.

If the Table-Valued Function does take input parameters, then please proceed below :

Table-Valued Function with parameters

If the Table-Valued Function takes input parameters, a dialog appears to let you set the parameters. It lists the input parameters and their data type :

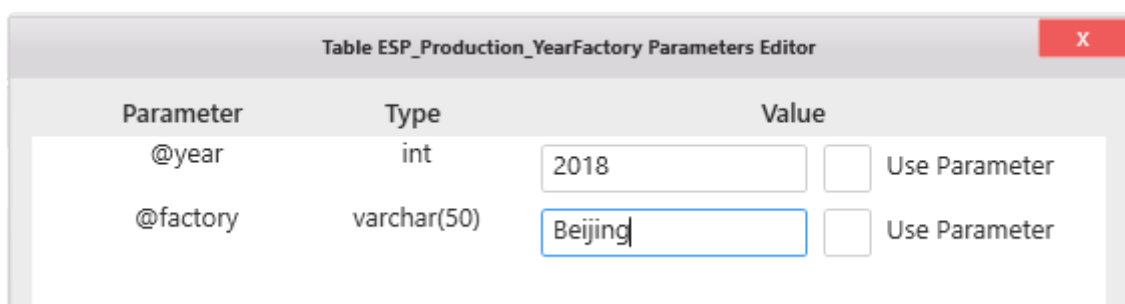


Parameter	Type	Value		
@year	int	<input type="text"/>	<input type="checkbox"/>	Use Parameter
@factory	varchar(50)	<input type="text"/>	<input type="checkbox"/>	Use Parameter

OK CANCEL

Table-Valued Function with static parameters

To fill in static values for the Table-Valued Function parameters, simply fill the text boxes and click “OK” :



Parameter	Type	Value		
@year	int	2018	<input type="checkbox"/>	Use Parameter
@factory	varchar(50)	Beijing	<input type="checkbox"/>	Use Parameter

OK CANCEL

With a static value, the Table-Valued Function will always be called with the value you entered.

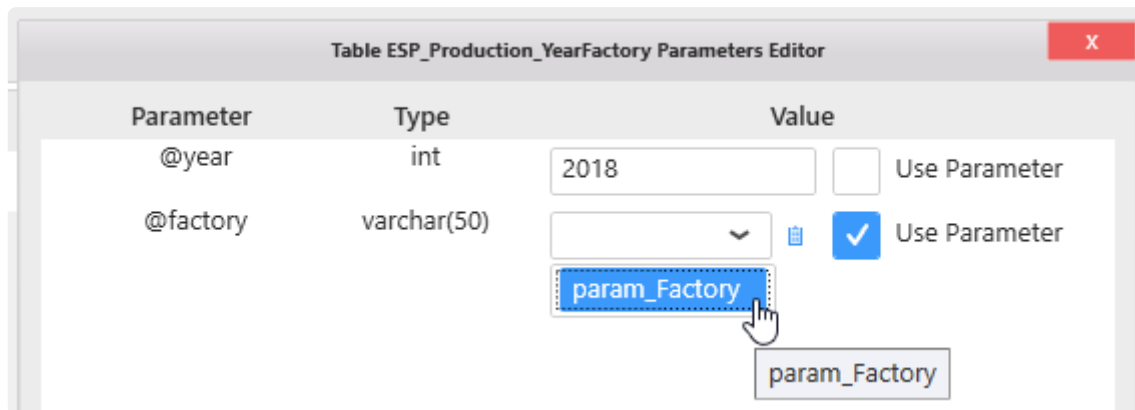
Table-Valued Function with dynamic parameters

Instead, you can bind a Table-Valued Function parameter to an Alpana Dashboard Parameter (see the

dedicated Course Chapter on [Parameters](#)).

This will allow you for example to change the Table-Valued Function call depending on Master Widget clicks on the dashboard.

To do so, in the “Table-Valued Function Parameters Editor” dialog, check “Use Parameter” and select an existing Alpana Dashboard Parameter :



You can also create a Parameter directly form this dialog by clicking the parameter list button :



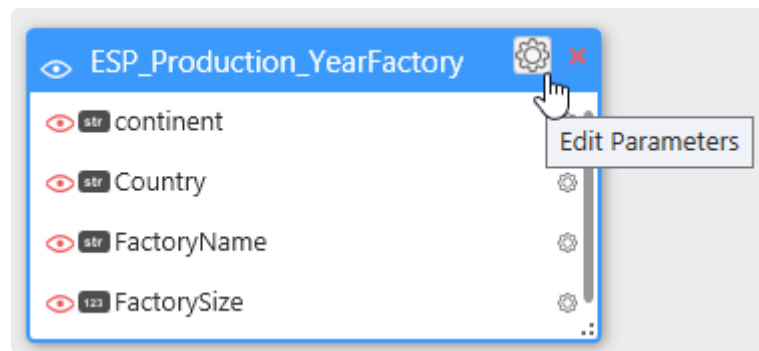
Note :

Be careful that some Table-Valued Function require non empty parameters. In order to complete this step, you may have to set a default value to the Alpana Parameter :

Name	param_Factory	Type	String
Default Value	Paris		<input type="checkbox"/> Return Multiple Values

Editing existing Table-Valued Function parameter binding

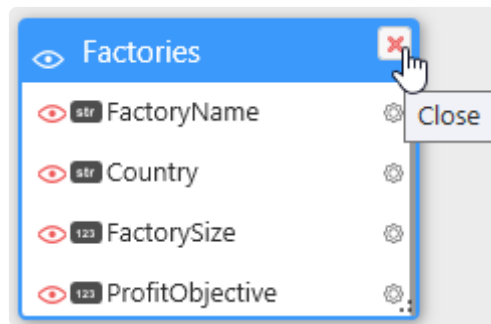
Once the Table-Valued Function parameters are bound, you can change them at any time by clicking the “Edit Parameters” cog icon on the corresponding table :



Last modified: 2019/03/13

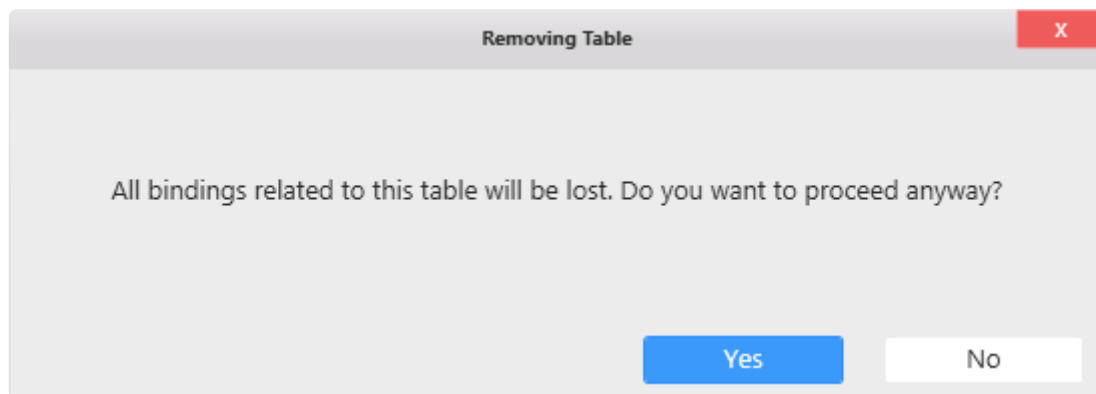
2.5.4.7. Removing a table from the Data Source

You can remove an object (Table, View, etc, ...) from the Data Source by clicking the cross on the top right of the table :



If the table is in a relationship with other tables and removing it breaks Join conditions, a confirmation dialog appears :

All bindings related to this table will be lost. Do you want to proceed anyway ?



If you press OK, the Join Editor dialog appears and lets you fix the missing relationships (see below Chapter “Merging Data”).

Last modified: 2019/03/13

2.5.5. Custom Query Tables

What is a Custom Query Table ?

When composing a Data Source, you are not limited to using Tables or Views from your Connection.

You can also add a **Custom Query Table** : similarly to a SQL View, you write a SQL query and a Table will be created in the currently opened Data Source.

However, this can be more than a simple way to create a custom View, as we will see further in the Using Parameters chapter.

Pre-requisite

In this Chapter, we will edit a Data Source.

To learn how to create or edit a Data Source, see the previous Chapter “Creating and Managing Data Sources”.

In this Chapter, it is assumed that you have created a Data Source and that it is open for edition.

Last modified: 2019/03/07

2.5.5.1. Language

When connecting directly to a database, the language of the Custom Query must be compatible with the database.

When connecting to a bufferized Connection, the Custom Query must be written in the language used by the **Buffer** database.



Important

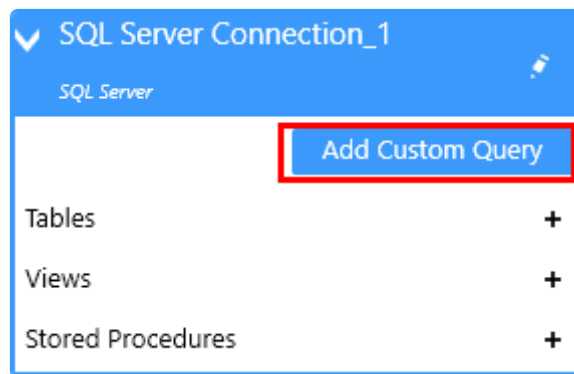
When changing the underlying buffer database technology (for example when switching from SQL Server to SQLite), the Custom Query Table will probably need to be updated accordingly, since there are some differences in their SQL syntaxes.

Last modified: 2019/03/07

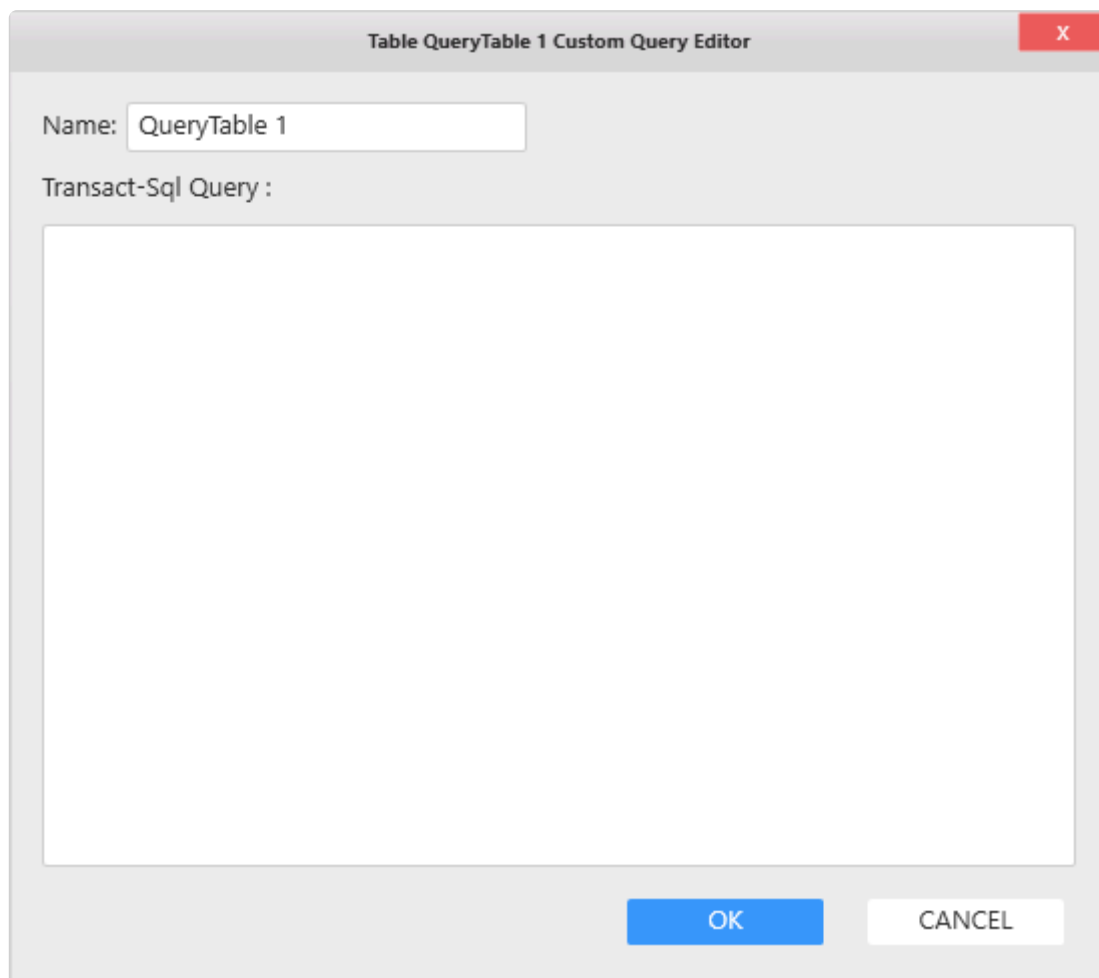
2.5.5.2. Creating a Custom Query Table

To do so, open the Data Source where you need to add a **Custom Query Table**.

From the Connection schema below, click on Add Custom Query.

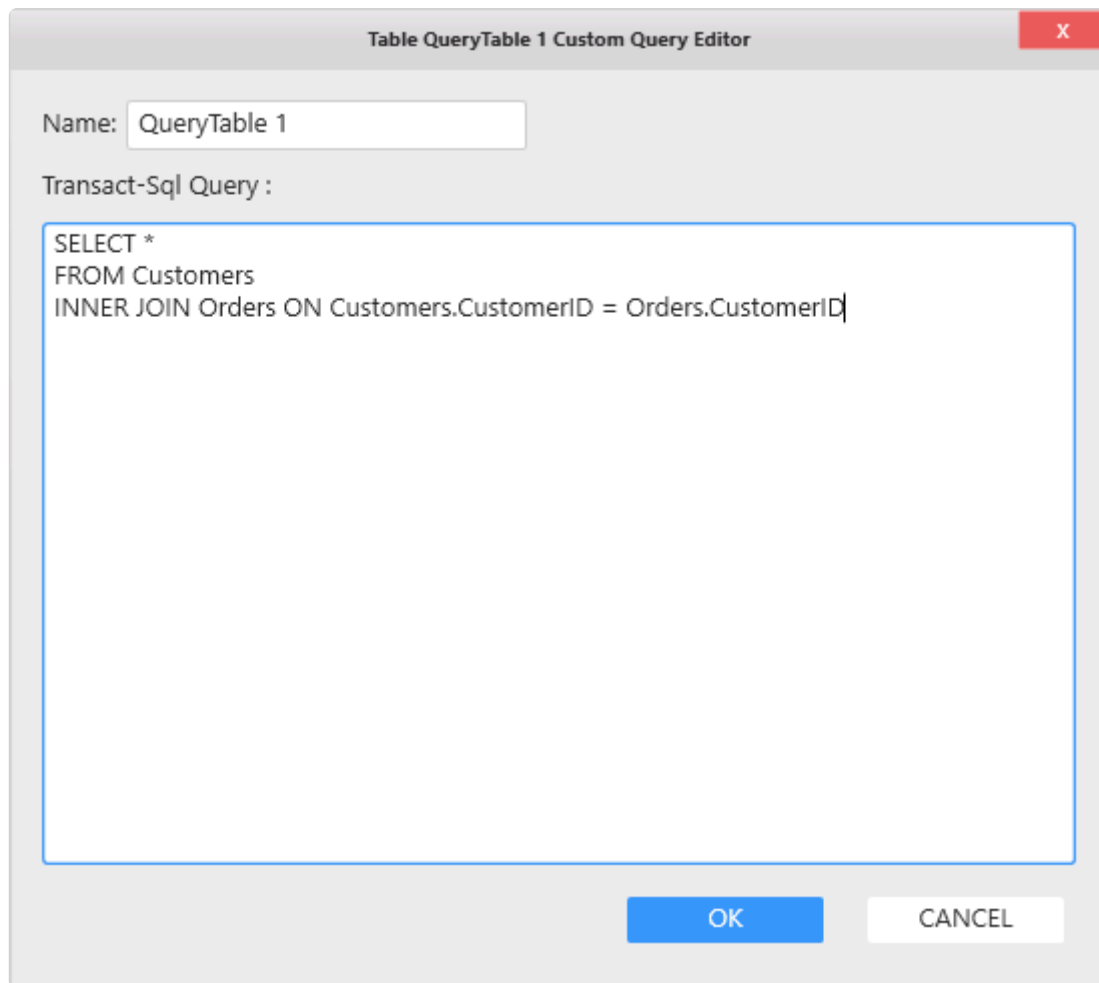


This will bring up the Custom Query Editor frame.



Write the query in the correct SQL language (see the previous chapter “Language”).

For bufferized Connections, you can use the table names as listed in the schema.



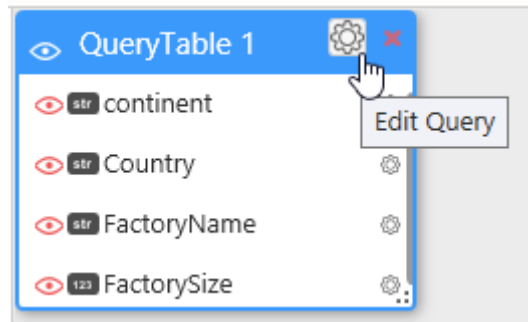
Click OK to create the Custom Query Table. Fix any SQL error if necessary.

Now the Custom Query Table is created in the Data Source currently opened. It can be used like a normal table from the same Connection and, for example, can be used in a Join.

Last modified: 2019/03/13

2.5.5.3. Updating a Custom Query Table

To do so, click on the cog icon to view and update the Custom Query.



Important

Be careful that any changes in the schema (column names and types) could impact the items that use this Custom Query Table, such as Widgets or Expressions. Changing the schema of the query could require to re-do some or all of the related data binding.

Click OK to update the Custom Query Table.

Fix any SQL error if necessary.

Last modified: 2019/03/13

2.5.5.4. Using Parameters

Parameters can be used to parametrize a Custom Query : the Custom Query changes dynamically.

Syntax

`${parametername}` will be replaced in-place by the values of “*parametername*”.

Example :

```
SELECT *
FROM MyTable
WHERE MyColumn = ${parametername}
```

will get replaced at runtime by :

```
SELECT *
FROM MyTable
WHERE MyColumn = parametervalue
```

Multiple values

Parameters that return multiple values will return them separated with commas.

So that you can write :

```
SELECT *
FROM MyTable
WHERE MyColumn IN( ${multiparameter} )
```

and it will get replaced at runtime by :

```
SELECT *
FROM MyTable
WHERE MyColumn IN( Value1,Value2,Value3 )
```



NOTE

In Alpana Designer, the Custom Query will be executed with the default parameter values. This may cause unwanted syntax errors if the default value is empty for example. Setting a default value to the parameter fixes that issue.

Last modified: 2019/03/07

2.5.6. Merging Data

Last modified: 2019/02/21

2.5.6.1. Creating Joins

Why Join ?

When an object (a Table, a View, etc...) is already present in the Data Source and you want to add another object, you will have to specify how the two objects will be merged together to create a single data query.

In order to do this, a database JOIN operation is used.



What is a join ?

In simple terms, a join between a data Table1 and a Table2 expresses how an entry in Table1 describes the same information as another entry in Table2.

“Joining” comes from SQL terminology. If you know SQL, this editor should expose familiar concepts.

This operation can be very useful for adding context to your data :

- your Table1 contains physical measurements
 - your Table2 contains context information like where was data measured, what product you were producing at that time, etc...
- ⇒ This is all multiple aspects of the same information. By joining, you will be able to present all this data consistently in your visualizations.

What can I join ?

Joins can be created across all Connections, all Connection types (SQL, Excel, Historian, ...), all object types (Tables, Views, Stored Procedures, ...).

Any Number of join conditions can be created.



Note : Multi-Connection

When a Data Source joins tables from at least 2 different Alpana Connections, the Connections become bufferized.

How to Join tables

When an object (a Table, a View, etc...) is already present in the Data Source and you add another

object, the Join Editor dialog automatically opens to let you configure the join relationship.

If you cancel this dialog, the new object is not added.

Last modified: 2019/04/17

2.5.6.2. Configuring Joins

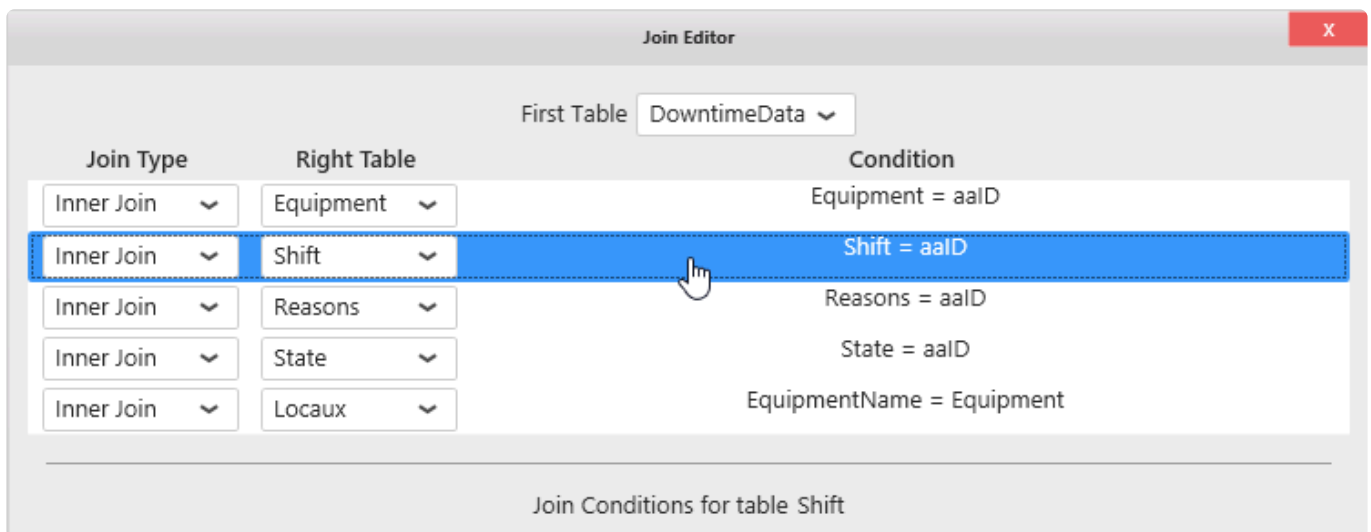
Configuring a relationship

The Join Editor presents a list of relationships similar to a SQL join :

Data is retrieved from a first table joined with X other tables.

This relationship expresses how an entry in Table1 will be on the same row as another entry in Table2.

Each relationship can be selected by clicking in the list :



Join Editor

First Table: DowntimeData

Join Type	Right Table	Condition
Inner Join	Equipment	Equipment = aaID
Inner Join	Shift	Shift = aaID
Inner Join	Reasons	Reasons = aaID
Inner Join	State	State = aaID
Inner Join	Locaux	EquipmentName = Equipment

Join Conditions for table Shift

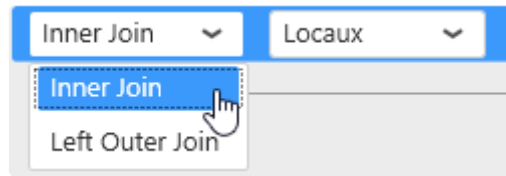


There is 1 relationship for each table added beyond the first. If this is your first join, there is only one relationship.

Selecting the Join Type

A list allows to select the Join Type :

- *Inner Join* : only retrieve data where the join condition is true
- *Left Outer Join* : retrieve all data from the first table ("left table") even if the join condition is not true. If the join condition is false and no data exists in the second table ("right table"), then *NULL* data is retrieved for the second table.



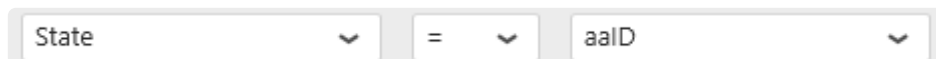
Join Conditions

In order to match data between tables, a condition must be expressed.

For example, if Table1 contains Customer orders and Table2 contains Persons profiles, you may want to join on “CustomerName = PersonsName”.

This means that the resulting dataset will show Persons information on the same row as Customer order information when the Name is the same.

By default a join condition is proposed in the editor.



Configuring a Join Condition

A condition is composed of 3 elements from right to left :

- a right member : a field from the currently selected Right Table. For your first join, this is the table you just added to the Data Source
- an operator : will express the relationship
- a left member : a field from any other table

To configure the Condition :

Select a right member

Select a field from the current table which will have to match the left field :

Join Editor

First Table DowntimeData ▾

Join Type	Right Table	Condition
Inner Join ▾	Equipment ▾	Equipment = aaID
Inner Join ▾	Shift ▾	Shift = aaID
Inner Join ▾	Reasons ▾	Reasons = aaID
Inner Join ▾	State ▾	State = aaID
Inner Join ▾	Locaux ▾	EquipmentName = Equipment

Join Conditions for table State

State ▾

= ▾

aaID ▾

aaID

StateCode

State

aaTimeStamp

aaTimeStamp_UTC

aaTimeStamp_Offset

Add

Select a left member

Select a field from any other table which will have to match the selected right field.

In the list, the fields are grouped by table names :

Join Editor

First Table **DowntimeData** ▼

Join Type	Right Table	Condition
Inner Join ▼	Equipment ▼	Equipment = aaID
Inner Join ▼	Shift ▼	Shift = aaID
Inner Join ▼	Reasons ▼	Reasons = aaID
Inner Join ▼	State ▼	State = aaID
Inner Join ▼	Locaux ▼	EquipmentName = Equ

Join Conditions for table State

State ▼ = ▼ aaID

^ DowntimeData

aaID

DowntimeDuration

DowntimeCount

aaValueTimeStamp

Date

Add

Select an operator

Select an operator from the list to express the desired relationship between the two selected fields :

=

=

!=

>

>=

<

<=

Multiple join conditions

Sometimes a single condition is insufficient to express the desired relationship.

Adding a condition

It's possible to add as many conditions as desired.

Conditions can be added by clicking the “Add” button :

Join Conditions for table State

×	State	▼	=	▼	aalD	▼
×	aalD	▼	=	▼	aalD	▼
×	aalD	▼	=	▼	aalD	▼
×	aalD	▼	=	▼	aalD	▼

Add

Removing a condition

A minimum of 1 condition is required.

Each condition can be removed from the relationship by clicking on the corresponding red cross :

Join Conditions for table State

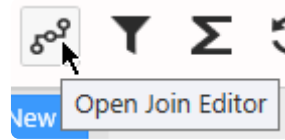
×	State	▼	=	▼	aalD	▼
×	aalD	▼	=	▼	aalD	▼
×	aalD	▼	=	▼	aalD	▼
×	aalD	▼	=	▼	aalD	▼

Add

Last modified: 2019/03/14

2.5.6.3. Updating Joins

You can edit existing Join relationships by clicking on the “Open Join Editor” button in the top toolbar :



In the Join Editor, select a relationship, change it, and press OK.

Last modified: 2019/05/28

2.5.7. Filtering Data

Data Source Filters

You can apply filters at the Data Source level.

All Widgets that will use this Data Source will only see the data after the filter is applied.

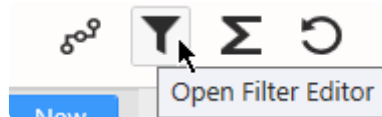
Note : If you wish to apply different filters to each Widget separately, see [the corresponding chapter](#).

Last modified: 2019/04/30

2.5.7.1. Opening the Filter Editor

Add or Edit Filters

To add Filters to a Data Source or edit existing filters, press the “Open Filter Editor” funnel icon in the top toolbar :



See the presence of Filters

When a filter is applied on a DataSource, the Filter button is colored :



Each filtered table field appears with a yellow background and the word (Filtered) next to its name :

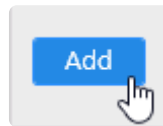


Last modified: 2019/04/30

2.5.7.2. Managing Filter Conditions

Adding a condition

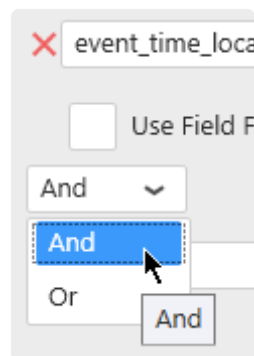
In the Filter Editor, click Add to add a filter condition :



Combining conditions

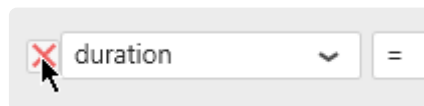
If filter conditions already exist, they will be chained using a logical operator.

By default, the operator is “And”, but it can be changed to “Or” :



Removing Conditions

Click the red cross on the left to remove the corresponding condition :



Last modified: 2019/03/15

2.5.7.3. Configuring Filter Conditions

Filter conditions are expressed with :

- a field name from the Data Source. This can also be an [Expression](#).
- a boolean operator
- a value. This can also be a [Parameter](#).

Depending on the Data Source field type, the settings are slightly different, please see for each type in the next pages.

Last modified: 2019/03/15

2.5.7.3.1. Filtering string fields

This is the most basic case.

The filter condition will be :

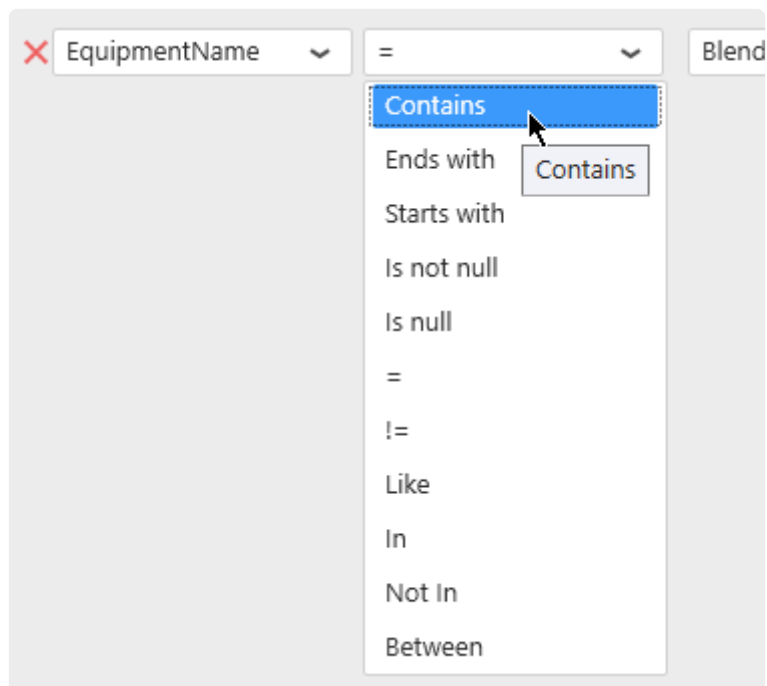
<FieldName> <Operator> <Value>

Example :

EquipmentName StartsWith Blender

Operators

Operators appropriate to Strings can be selected.



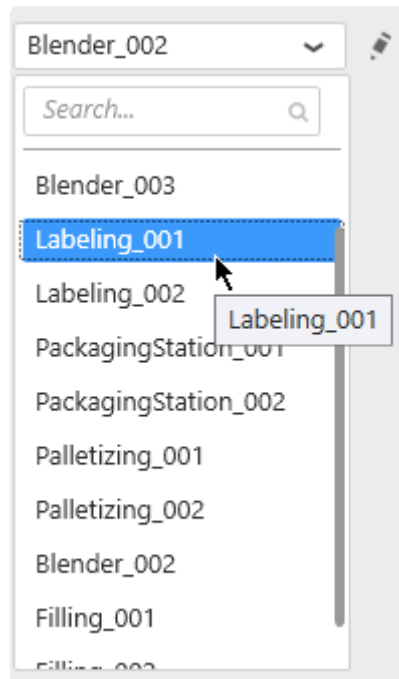
Values

Static values

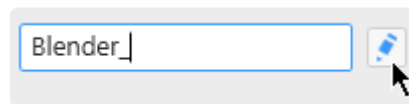
The “Edit values” button can be check/unchecked manually :



When the “Edit values” button is un-checked, a value can be selected from current values in the database :



When the “Edit values” button is checked, a value can also be entered as text :



* Depending on the operator selected, some of these options may not be available.

Dynamic filter values using Parameters

Instead of comparing to a static value, it is possible to compare to a dynamic Parameter that will depend on actions inside the Dashboard.

For this, select “Use Parameter” and see the Chapter about [Parameters](#).



Last modified: 2019/03/15

2.5.7.3.2. Filtering number fields

In a basic configuration, the condition is expressed as :

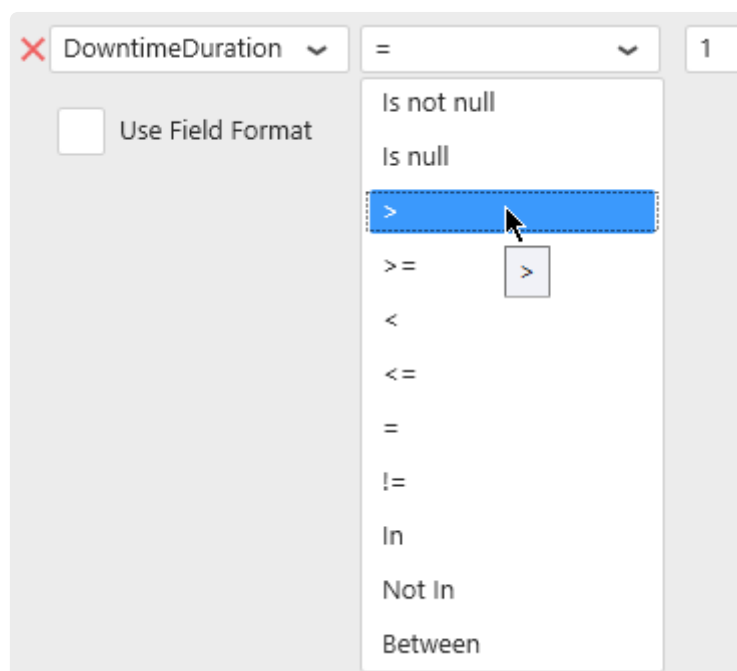
```
<FieldName> <Operator> <Value>
```

For example :

```
DownTimeDuration > 10
```

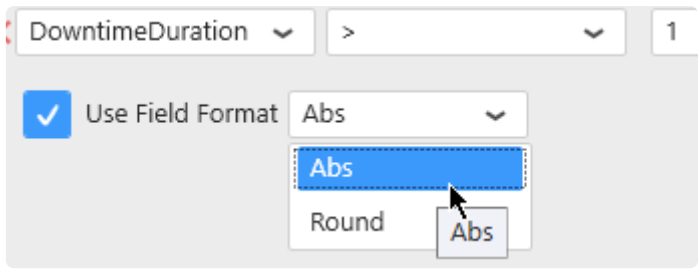
Operators

Operators appropriate to Numbers can be selected :



Field formatting

In addition to the basic configuration, a formatting function can be applied to the numerical field while the comparison is made :



This way, the filter condition will be :

```
<Function>(<FieldName>) <Operator> <Value>
```

For example :

```
Abs (DownTimeDuration) > 10
```

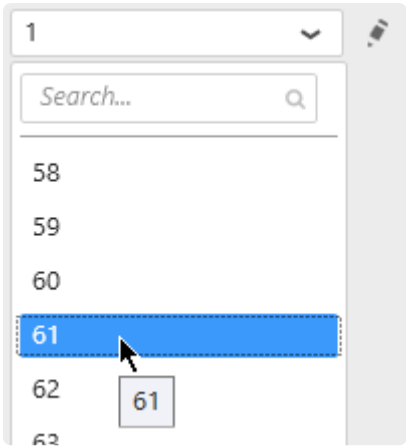
Values

Static values

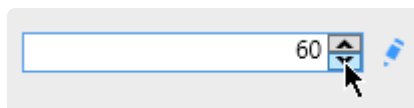
The “Edit values” button can be check/unchecked manually :



When the “Edit values” button is un-checked, a value can be selected from current values in the database :



When the “Edit values” button is checked, a value can also be entered as text, or using the up/down arrows to increase/decrease :



✿ Depending on the operator selected, some of these options may not be available.

Dynamic filter values using Parameters

Instead of comparing to a static value, it is possible to compare to a dynamic Parameter that will depend on actions inside the Dashboard.

For this, select “Use Parameter” and see the Chapter about [Parameters](#).



Last modified: 2019/03/15

2.5.7.3.3. Filtering date fields

In a basic configuration, the condition is expressed as :

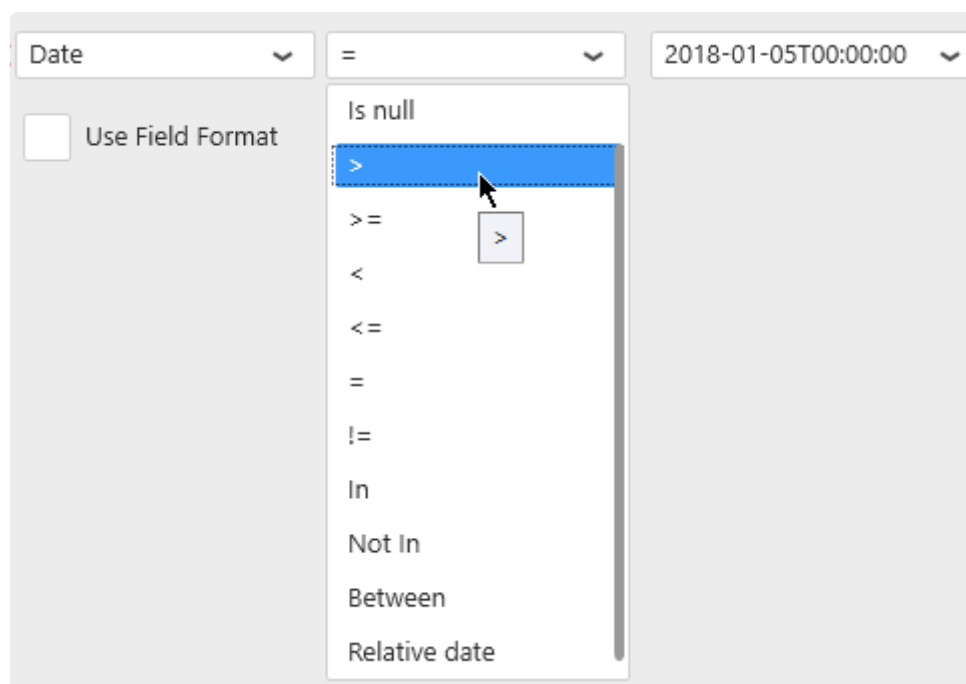
```
<FieldName> <Operator> <Value>
```

For example :

```
aaValueStartTime > 2018-01-05T00:00:00
```

Operators

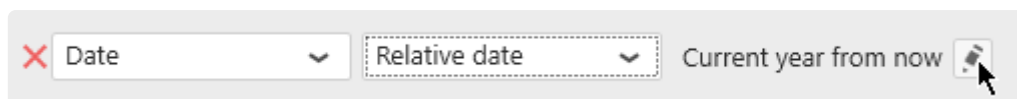
Operators appropriate to Dates and Times can be selected :



Relative date filtering

The special operator “Relative date” allows to select a “date window” / “time frame” that is moving over time relative to the current moment.

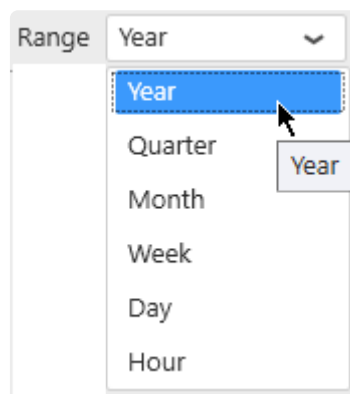
When selecting this operator, click the edit icon to define the relative date window :



✖ Date
Relative date
Current year from now

Then for each time range :

Year, Quarter, Month, Week, Day, Hour

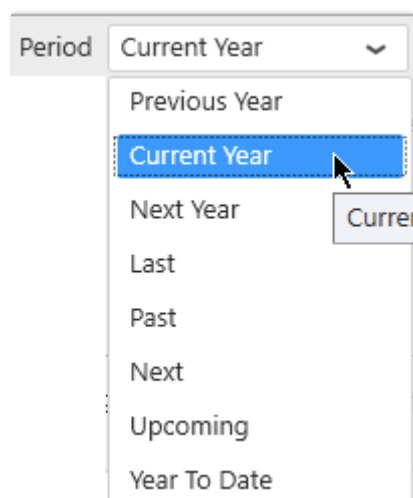


Range

- Year
- Quarter
- Month
- Week
- Day
- Hour

... a Period can be selected :

Previous 1, Current, Next 1, Last N, Past N, Next N, Upcoming N, To Date



Period

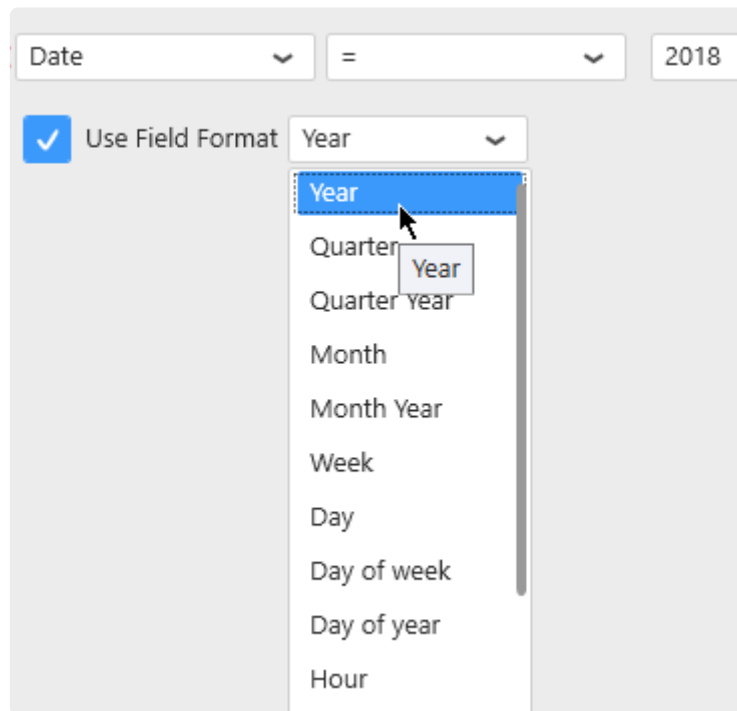
- Current Year
- Previous Year
- Next Year
- Last
- Past
- Next
- Upcoming
- Year To Date

A preview of the selected date range for the current time appears at the bottom of the window to help you configure :

1/1/2019 12:00:00 AM to 12/31/2019 11:59:59 PM

Field formatting

Similarly to the number fields, a formatting function can be applied to the date field :



This way, the filter condition will be :

```
<Function>(<FieldName>) <Operator> <Value>
```

For example :

```
Year(Date) BETWEEN 2015 AND 2016
```

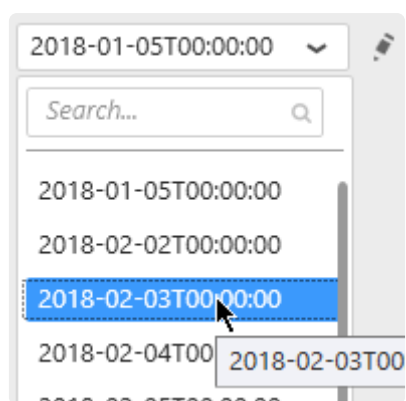
Values

Static values

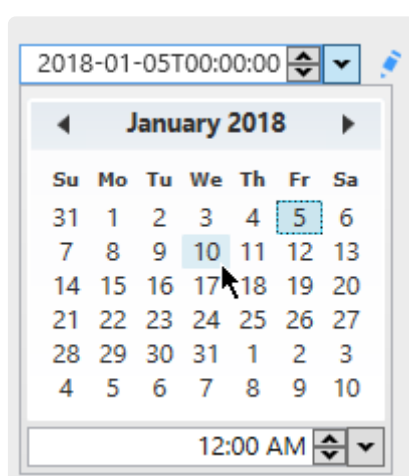
The “Edit values” button can be check/unchecked manually :



When the “Edit values” button is un-checked, a value can be selected from current values in the database :



When the “Edit values” button is checked, a value can also be entered as text :



* Depending on the operator and formatting function selected, some of these options may not be available.

Dynamic filter values using Parameters

Instead of comparing to a static value, it is possible to compare to a dynamic Parameter that will depend on actions inside the Dashboard.

For this, select “Use Parameter” and see the Chapter about [Parameters](#).

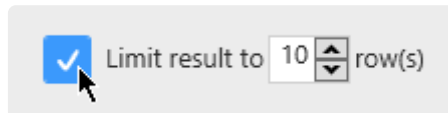


Last modified: 2019/05/03

2.5.7.4. Limiting dataset size

How to limit

Check “Limit result to” and select the number of rows you want to keep in the dataset :



Why limit ?

Checking this option with “Limit result to 10 rows” for example will apply a “TOP 10” to the query. When widgets are sorting the data, this can be a useful to display the 10 Items with the worst performance, or the 10 equipments with the highest consumption for example.

Last modified: 2019/03/15

2.5.8. Calculating Expressions

Alpana can add any number of calculated fields to a Data Source.

These are called *Expressions*.

Expressions are stored inside the Data Source and exported along with it.

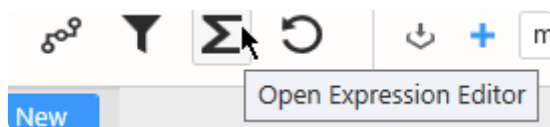
Expressions are evaluated when the Widget query is made, and will show accurate result depending on the Widget aggregation.

Because of this, using Expressions can be mandatory for certain types of calculations.

Last modified: 2019/04/17

2.5.8.1. Opening Expression Editor

Click on the Sigma Σ icon “Open Expression Editor” button :



The Expression Editor dialog opens to let you create and manage Expressions.

Last modified: 2019/04/17

2.5.8.2. Managing Expressions

Adding Expressions

Click the **+** button at the bottom left to add an Expression :



Deleting Expressions

Click the trash icon button at the bottom left to remove the selected Expression :



Hiding Expressions

Click on the eye icon next to an Expression name to hide it :



This can be useful for example when you create an intermediate Expression that will be re-used in other Expressions but not displayed to the end user.

Using Expressions

Expressions can be used in any other place where a data field can be found :

- Widget data binding
- Filter configuration
- ...

In the Data Preview of the Data Source, you can also view a preview of the Expression as if it were another field in the data.



Please note that this is only a Preview. Since aggregation Expressions are calculated differently depending on aggregation, they will show the expected result in Widgets, which might be different.

Last modified: 2019/04/17

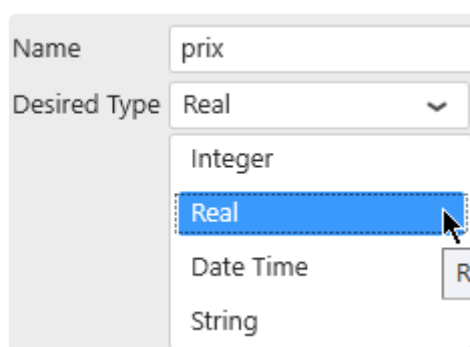
2.5.8.3. Expression properties

Expression Name

You can chose a name for the Expression.

Expression Type

Select a desired type for the Expression :



The screenshot shows a dialog box for configuring an expression. It has two main sections: 'Name' and 'Desired Type'. The 'Name' field contains the text 'prix'. The 'Desired Type' section features a dropdown menu that is currently open, displaying a list of data types: 'Real', 'Integer', 'Date Time', and 'String'. The 'Real' option is highlighted with a blue background and a mouse cursor is pointing at it. A small 'Real' button is also visible to the right of the dropdown list.

This will determine how the Expression can be used. For example, what kind of aggregation can be made when it is bound to a Widget.

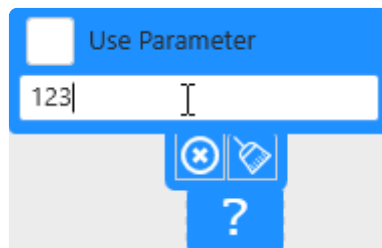
Last modified: 2019/04/17

2.5.8.4. Adding a constant

Constant value

By clicking on a ? question mark, you can input free form text :

- number constant
- string constant


 A screenshot of a blue-bordered input field. Inside the field, the text '123' is entered, followed by a cursor. Below the field, there is a small blue box containing a question mark icon. To the left of the question mark icon, there are two smaller icons: a circle with a cross and a document icon.

Note on dates

- * To input a constant date, it is strongly advised to use the [ISO](#) date format :
 YYYY-MM-DD for dates, for example 2019-12-25
 YYYY-MM-DDThh:mm:ss.sss for datetimes, for example 2019-12-25T23:59:59.999

When inputting dates, you are actually writing a string that will be converted to date on the target database where the query is made.

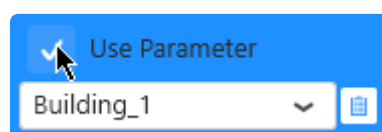
Note : in case of a [Bufferized connection](#), it is the Buffer database.

This means that your date string may be interpreted wrong depending on the language of the target database.

A date format that is not ambiguous is advised, for example ISO date format.

Dashboard Parameter

To use an Alpana Parameter instead of a constant value, check *Use Parameter* and select a Parameter (it is also possible to create a new Parameter from here by clicking the list icon) :


 A screenshot of a blue-bordered input field. At the top left, there is a checkbox labeled 'Use Parameter' which is checked. Below this, there is a dropdown menu showing 'Building_1' with a downward arrow. To the right of the dropdown, there is a small icon of a document with a plus sign.

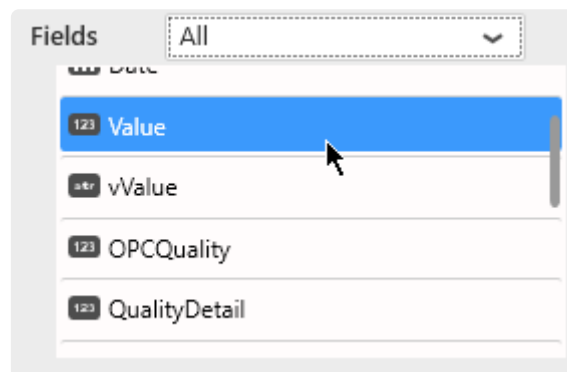
Last modified: 2019/07/02

2.5.8.5. Adding a field

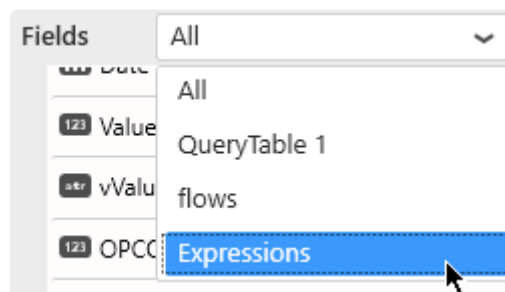
Any field from the Data Source can be used as function argument, including other Expressions

Browsing for a field

Fields can be browsed from the *Fields* container at the bottom :

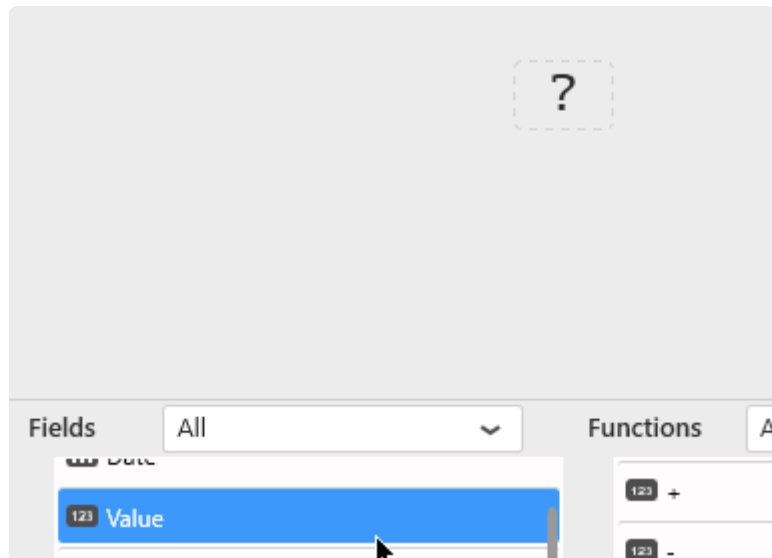


This list can be filtered by table name, or by selecting Expressions to list expressions only :



Adding a field to the Expression

To add a field to the Expression, simply drag it from the list into the Expression :

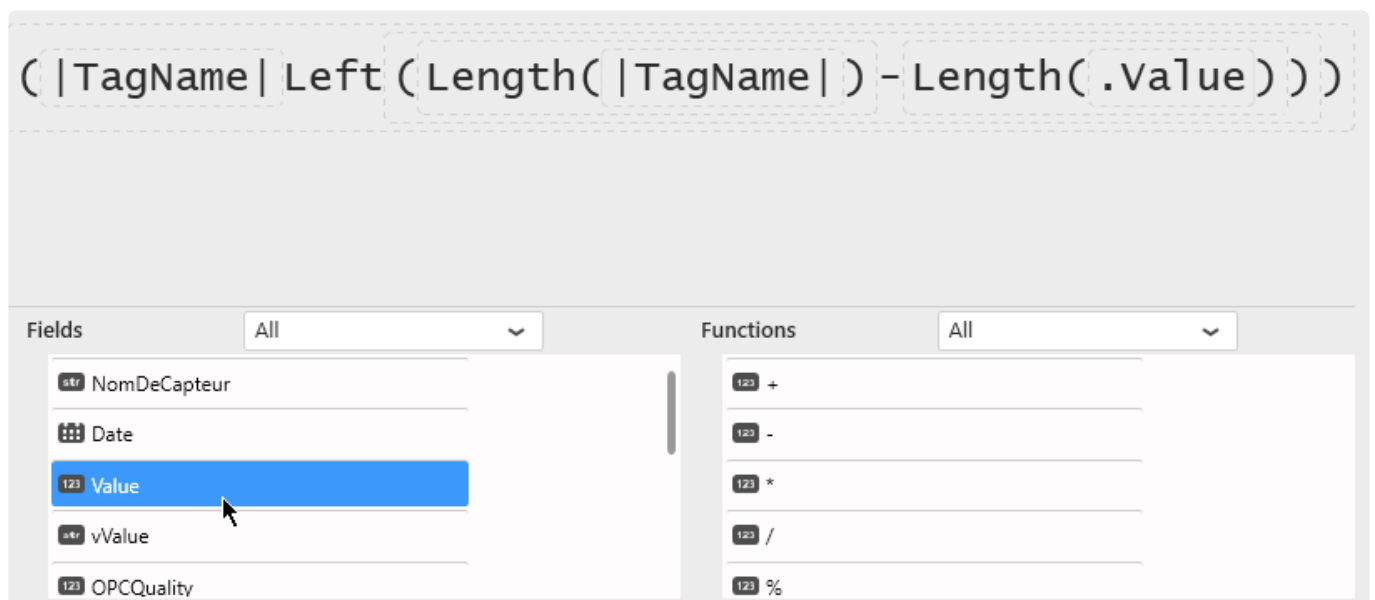


When adding a field on the ? question mark, the field will take this place.

Replacing an existing field

Using drag and drop

To replace part of an Expression by a field, it's possible to drag the field from the list and drop it onto the expression part to be replaced :



Browsing on the field

To replace a field in an Expression, it's possible to click on the field to be replaced, and select a

replacement field :



Last modified: 2019/04/17

2.5.8.6. Adding a function

Functions take part in the calculation and take other elements as argument.

There are two main types of functions :

- Calculation functions (like Arithmetic functions $*$, $/$, $-$, $+$, Abs, ... or String functions Concat, Left, ...)
- Aggregation functions (SUM, Count, AVG, ...)

Most functions require arguments. A function argument itself can be anything :

- Function
- Field
- Value

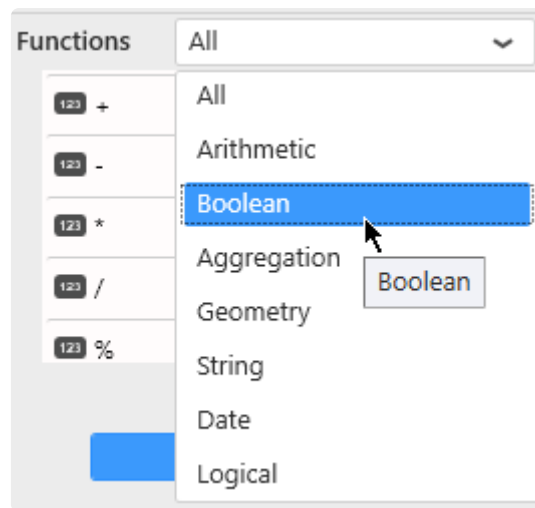
Any missing item will be displayed as a ? question mark.

Browsing for a function

Functions can be browsed from the *Functions* container at the bottom :

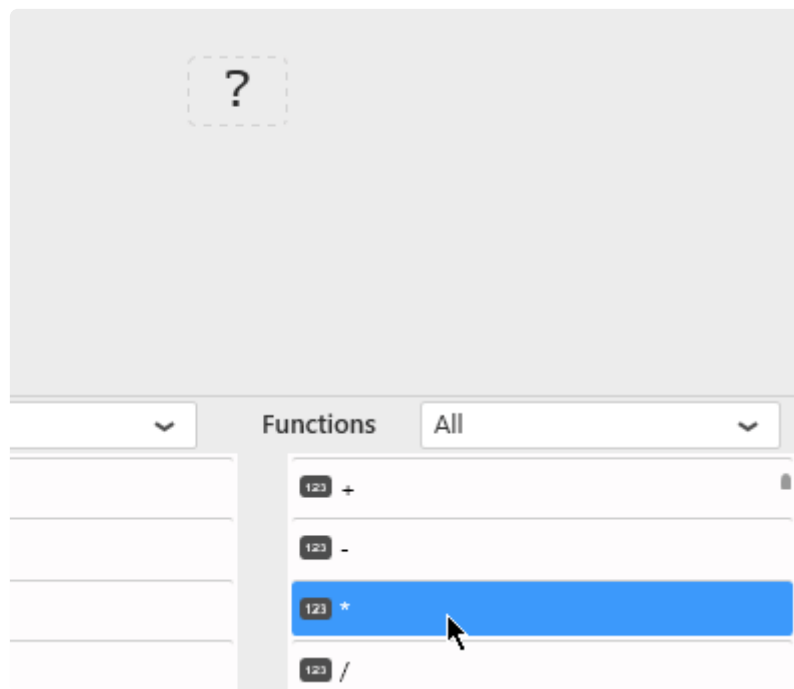


This list can be filtered by function type, or by selecting All functions :



Adding a function to the Expression

To add a function to the Expression, simply drag it from the list into the Expression :



When adding a function on the ? question mark, the function will take this place.

Replacing an existing function

Browsing on the function

To replace a function in an Expression, it's possible to click on the function to be replaced, and select a

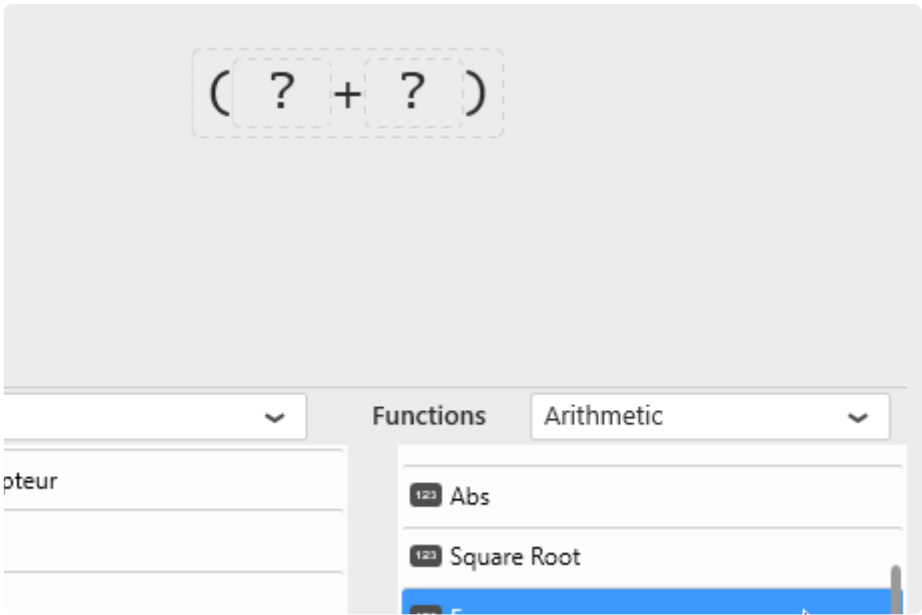
replacement function :



Wrapping an Expression into a function

Maybe you have started creating an Expression, but now you want to make it an argument to a new function.

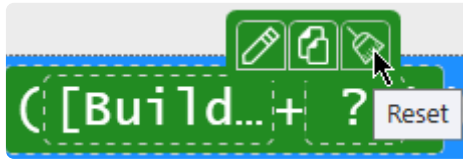
To wrap part of an Expression into a function, it's possible to drag the function from the list and drop it onto the Expression part to be wrapped :



Last modified: 2019/04/17

2.5.8.7. Removing an existing item

To remove an item and all the dependent items, click the item and click on the “Reset” button :



The item itself and all items underneath will be removed.

Last modified: 2019/04/17

2.5.8.8. Copying and Pasting an existing item

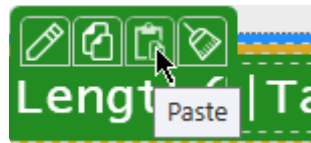
Copy

Click to select any item (function, field, value), then click on the *Copy* icon :



Paste

Click on the target item, then click on the *Paste* icon :



You can paste onto any item :

- to replace a ? question mark as a function argument
- replace a single field
- replace a function and all its dependent items
- into a different Expression

Last modified: 2019/04/17

2.5.8.9. Aggregated Expressions

What are Aggregated Expressions

There is a special type of Expressions : *aggregated Expressions*.

An aggregated Expression is simply an Expression that uses a function from the “*Aggregation*” category of functions.

! These types of Expressions are **essential** to solve some common issues.
For example, see [Pitfall : Calculating ratios, averages,](#).

Note : Aggregation Functions

Since Alpana v3.0, supported Aggregation Functions are :

- Avg
- Count
- Distinct Count
- Max
- Min
- Sum
- STDev
- STDevP
- Var
- VarP

They are listed in the “*Aggregation*” category of functions in the Expression Editor.

Rules

Note : these rules come from the underlying SQL engine. You will find the same rules in SQL queries.

- Non-aggregated fields cannot be used in the same Expression as aggregated fields. See [this chapter](#).
- Aggregation functions cannot wrap other aggregation functions. [See this chapter](#).

In particular, when binding an aggregated Expression as Measure to a Widget, only the “aggregation” type “*Value of*” can be used.

Usage

Aggregated Expressions will always be calculated correctly, independently of how data is grouped inside the Widget.

Example

Imagine a simple dataset :

TagName	Building	Product	Price
FT101.Value	Building A	Product X	10
FT102.Value	Building A	Product Y	15
FT201.Value	Building B	Product X	20
FT202.Value	Building B	Product Y	25
FT301.Value	Building C	Product X	30
FT302.Value	Building C	Product Y	35

You create the following Expression :

```
ExpressionTotal = SUM(Price)
```

If you display this full dataset in Grid, you get :

TagName	Building	Product	Price	ExpressionTotal
FT101.Value	Building A	Product X	10	10
FT102.Value	Building A	Product Y	15	15
FT201.Value	Building B	Product X	20	20
FT202.Value	Building B	Product Y	25	25
FT301.Value	Building C	Product X	30	30
FT302.Value	Building C	Product Y	35	35

⇒ It seems like the `SUM()` part of the Expression didn't do anything.

Now if you display this same Expression in a Pie Chart grouped by "Product", you get :

Product	ExpressionTotal
Product X	60
Product Y	75

⇒ because **60** is the SUM $10 + 20 + 30$ and **75** is the SUM $15 + 25 + 35$

Now if you display this same Expression in a Card Widget, grouped across all the dataset, you get :

ExpressionTotal
135

⇒ because **135** is the total SUM $10+15+20+25+30+35$

Last modified: 2019/06/17

2.5.9. Expressions Troubleshooting

Expressions are defined as objects in Alpana v2.0 and translated to SQL expressions when the widget queries are composed.

Some errors may happen from operations that are undefined in SQL, or when the calculation is not well understood.

This Chapter will help you in these cases.

Last modified: 2019/03/08

2.5.9.1. Pitfall : Incorrect result in Expression with division

Your issue

In some cases, Expressions with a division will give an unexpected result : 0 everywhere, or imprecise value for example.

Example Expression :

```
Expression1 = 100 * (SUM([isActive]) / COUNT([ID]))
```

In this case I want the % of Active items.

[isActive] is 0 or 1 and ID is some ID for the items.

Why am I getting weird results for this Expression1 ?

(0 everywhere, or imprecise value for example)

Explanation

Expressions are executed as part of the final SQL query.

For buffer-based Connections (Excel, CSV, Historian, Web, ...), the final queries are also **database queries**.

SQL division behaves differently if the two sides are Integer.

Test the following SQL query :

```
SELECT 123/456;
```

The result is 0

Why ?

We are performing an Integer Division (see https://en.wikipedia.org/wiki/Euclidean_division)

The quotient is 0 and the remainder is 123.

More info

To explore further, you can test the following SQL query (paste in SQL Management Studio for example) :

```
SELECT
    29/10 AS quotient,    -- 2
    29%10 AS remainder,   -- 9
    (29/10) * 10 + 29%10 AS result,    -- 29 = 2*10 + 9
    1.0 * (29/10) AS wrong,    -- 2.0
    (1.0 * 29)/10 AS real_division;    -- 2.9 because 1.0 * 29 = 29.0
```

In my example Expression1 above, SUM, and the result of COUNT is also an Integer.

So I am not going to get the % of Active items I expect

Solution

To fix it, I need to **force the conversion to a Real number** for at least one side of the division.

Example solution :



Multiply one member by 1.0 to force the conversion to a Real number.

```
Expression2 = 100 * (SUM([isActive] * 1.0) /COUNT([ID]))
```

Last modified: 2019/03/08

2.5.9.2. Pitfall : Calculating ratios, averages,

...

Introduction

An Expression may seem to perform the correct calculations, but it is important to double-check results during the development process, and question the calculations themselves.

One crucial point to remember is that fields must be aggregated in order to be displayed inside a widget. This performs a second layer of calculation and needs to be thought out carefully.

The easy case : SUM

Sum is easy. Sum of sums gives the expected total.

If you configure an Expression like :

```
ExpressionSUM = [DurationLoading] + [DurationPacking]
```

You can easily use it in a widget with default aggregation SUM, and you should get the total durations as expected result.

The only things to look out for is :

- unit consistence (adding minutes with minutes)
- NULL handling (what happens when [DurationPacking] is NULL)

The general case

Consider the following dataset :

Day	GoodProduction	TotalProduction	ratio
1	8	80	0.1
2	2	2	1
3	3	15	0.2
4	3	3	1
(total)	16	100	should be 16/100 = 0.16



Assignment :

I need to make an Expression for the ratio `GoodProduction / TotalProduction`.

ratio : First Attempt (*wrong*)

I create :

```
Expression1 = [GoodProduction] / [TotalProduction]
```

It **seems** to give good results in the Data Preview grid in Alpana Designer. But **in the Data Preview, rows are not aggregated !**

When I add the expression to a widget, what aggregation do I choose ?

`SUM([Expression])` will give 2.3 which is **incorrect** ! The reason is that you calculate the sum of ratios $(0.1 + 1 + 0.2 + 1)$ which has no meaning.

`AVG([Expression])` will give 0.575 which is **incorrect** too ! The reason is that you calculate the un-weighted average of ratios $(0.1 + 1 + 0.2 + 1)/4$ which has no meaning also.

ratio : Solution

The Expression must express what you really want to calculate, including aggregation :

```
Expression2 = SUM([GoodProduction]) / SUM([TotalProduction])
```

This will be used as a calculated column in all queries, including widgets and will always make the correct calculation.

For example, in the Total of the data above, you will correctly get : $(8+2+3+3) / (80+2+15+3) = 0.16$

There is only one fix needed :

Since we are making a ratio of integers, we should force to cast as float to make float division :

```
Expression3 = (SUM([GoodProduction]) * 1.0) / SUM([TotalProduction])
```

General case : Averages and other

In many cases, this phenomenon can be hidden.

If your database field is called `[AverageDuration]`, it is probably wrong to make a SUM or AVG of it, for

the same reasons as above.

In some cases, you will need to *expose to the dashboard Data Source the fields necessary to re-aggregate the calculation properly*.

Example on adapting your data

Your Data Source points to an SQL View with a calculated column “OEE”.

You made the calculation before reaching the dashboard, and you defined it as : $[OEE] = \frac{SUM([Uptime])}{(SUM([Uptime]) + SUM([Downtime]))}$

Now in your Dashboard, how will you aggregate this OEE ? You will run into the issues described above as you will implicitly make wrong calculations.



The solution here can be to expose the fields [Uptime] and [Downtime] inside your SQL View. Like that, in your Data Source, you will be able to recreate this calculation as an Expression, and get correct results in all cases.

see also

[Aggregated Expressions](#)

Last modified: 2019/04/29

2.5.9.3. Error when mixing aggregate fields and non-aggregate fields

The Expression engine (SQL) doesn't allow to mix aggregate fields with non-aggregate fields, as it results in undefined behaviour.

Example :

```
Expression = SUM(ValueField_A) + ValueField_B
```

ValueField_A is inside an aggregation expression (SUM) and ValueField_B is not.

This results in an error :

"The following expression(s) contain(s) non aggregate field references and aggregate functions at the same time which is not allowed"

To solve this, think about the calculation you really want to do, and what will happen when it will be included in a GROUP BY operation.

see also

[Aggregated Expressions](#)

Last modified: 2019/04/29

2.5.9.4. Error when wrapping aggregate fields inside aggregate functions

The Expression engine (SQL) doesn't allow to apply an aggregation function on top of another aggregation function.

Example :

```
Expression = AVG(SUM(ValueField_A))
```

This results in an error :

“Cannot perform an aggregate function on an expression containing an aggregate or a subquery.”

To solve this, you should leave only 1 level of aggregation functions (think about what you really want to calculate and test on a small dataset).

see also

[Aggregated Expressions](#)

Last modified: 2019/04/29

2.5.10. Value-based Settings

In addition to data preparation itself, Data Sources also contain *information that is bound to the data*.

These “*value-based settings*” have a different value based on the value inside a Data Source field.

Since v2.0, the only value-based setting is Value-based Colors.



Note :

This is only available for text type fields.

Last modified: 2019/06/17

2.5.10.1. Value-based Colors

Goal

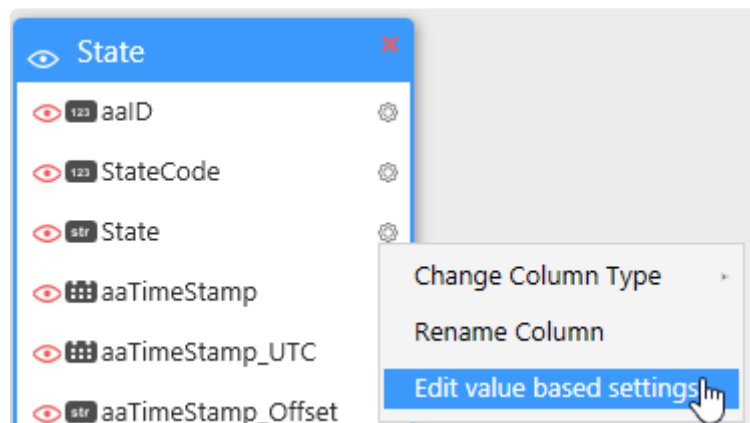
The “Value-based color” mechanism allows to bind a central unique value for a color based on a Data Source field value.

For example, you may want to associate colors to values in the “Machine State” field :

- value “Downtime” is always **red**
- value “Running” is always **green**

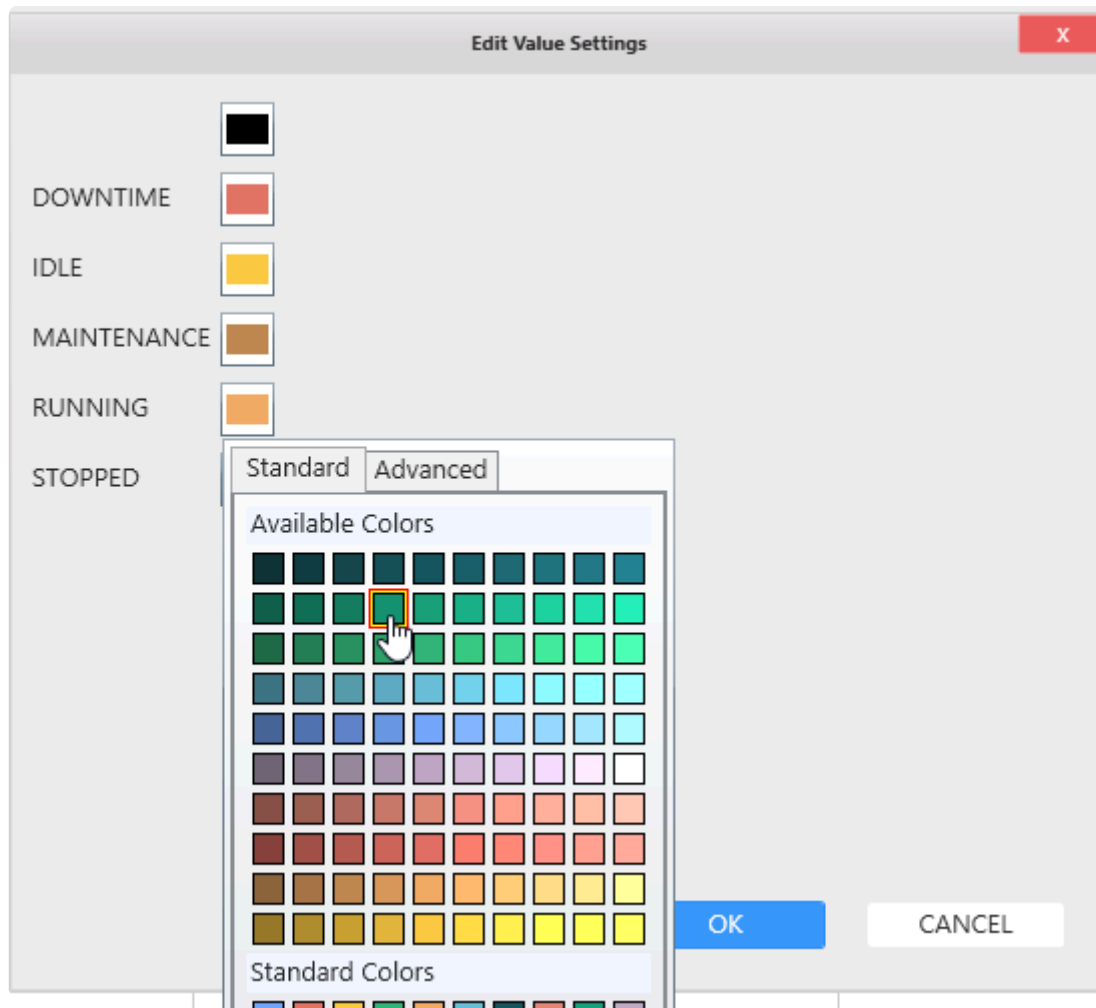
Edit value-based colors on a data field

To edit value-based colors on a data field, find the relevant field and click on the cog icon directly to its right, then select “Edit value based settings” :



Each value currently in the field is listed.

To assign a color, just click on the current color :



Change to your desired color and click OK to validate.

Edit value-based colors on an Expression

To edit value-based colors on an Expression, find the relevant field and click on the cog icon directly to its right, then select “Edit value based settings” :



Then proceed as above to assign colors.

Last modified: 2019/03/14

2.5.11. Formatting Columns

Last modified: 2019/02/21

2.5.11.1. Renaming Columns

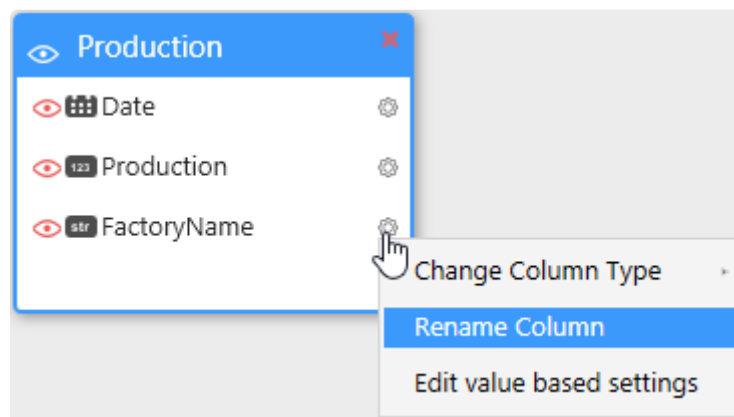
Why rename columns ?

The column names will appear in widget data binding.

Giving good names can help the user who will make the Dashboard based on this Data Source.

How

To rename a column, click on the cog icon and select “Rename Column” to enable the edit mode :

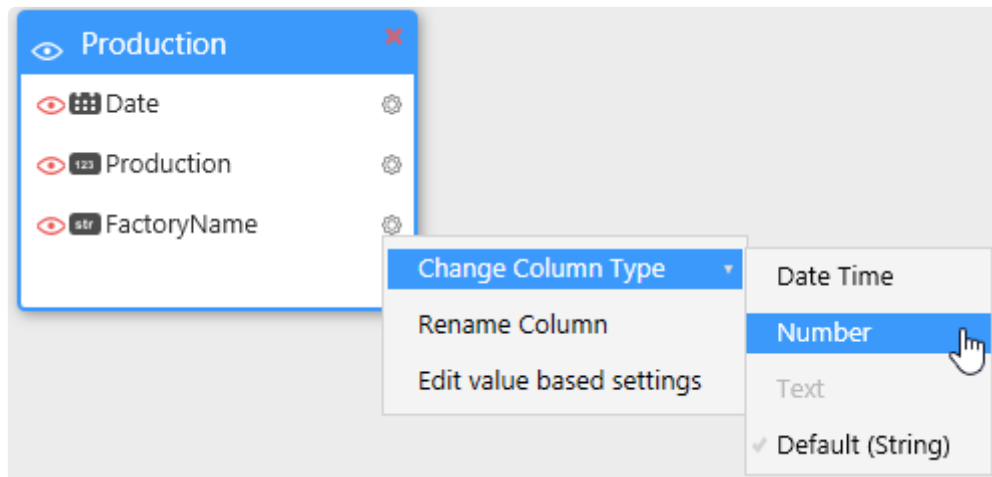


Press Enter to validate the new name.

Last modified: 2019/03/14

2.5.11.2. Changing Column Type

To change a column data type, click on the cog icon and select “Change Column Type”, then choose the new type of the column :



* On Connections Types that don't explicit types (like CSV), the default selected type may be incorrect. This allows you to select a better data type.

Last modified: 2019/03/21

2.5.11.3. Hiding Column

Why hide columns ?

The visible columns will appear in widget data binding.

hiding un-necessary or confusing columns can help the user who will make the Dashboard based on this Data Source.

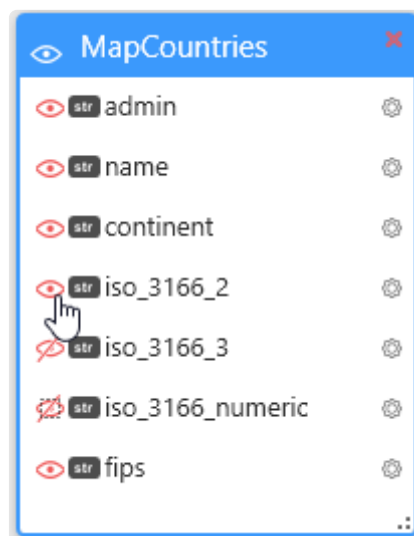


NOTE

If you hide a column, it won't be accessible for Widgets data binding. However, it will still be accessible for making Joins, Expressions and Data Source Filters.

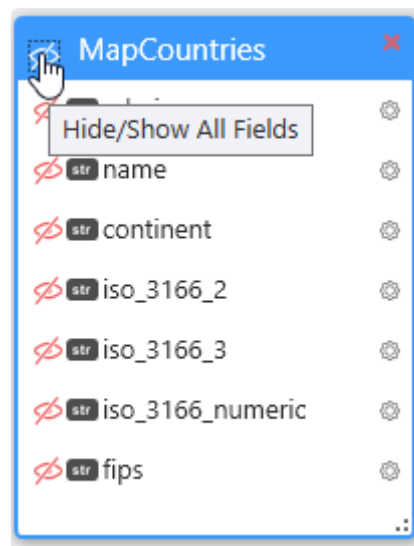
How

To Hide or Unhide a column, click on the red eye, just to the left of the desired column :



Hide all

To Hide or Show all Columns of a table, click on the eye icon just to the left of the table name :



Last modified: 2019/03/14

2.6. Configuring a Widget

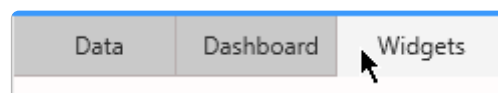
The main Widget configuration activities in Alpana Designer include :

- binding data : what to display
- graphical properties : how to display it
- interactions : how visualizations interact with each other



To create Widgets, you will need to connect and prepare your data first. Please see the relevant Courses : [Connecting to Data](#) and [Transforming Data](#).

Since Alpana Designer v3.0, all Widget configuration activities are made inside the *Widgets* activity tab.

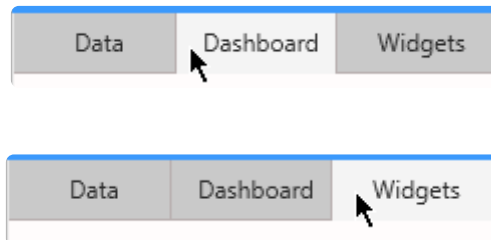


Last modified: 2019/06/17

2.6.1. Creating and Managing Widgets

This chapter describes how to create and manage Widgets.

Since Alpana Designer v3.0, Widgets are spawned in the *Dashboard* activity tab, and then configured inside the *Widgets* activity tab :

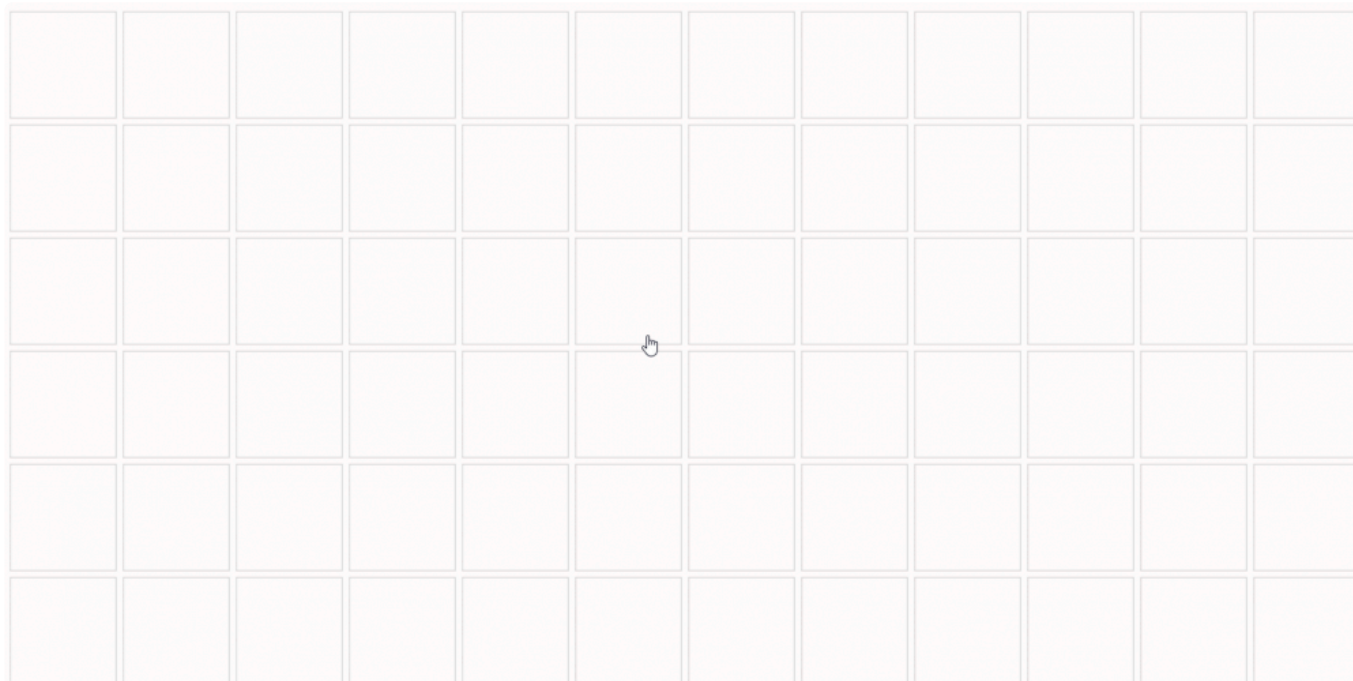


Last modified: 2019/06/17

2.6.1.1. Adding a Widget

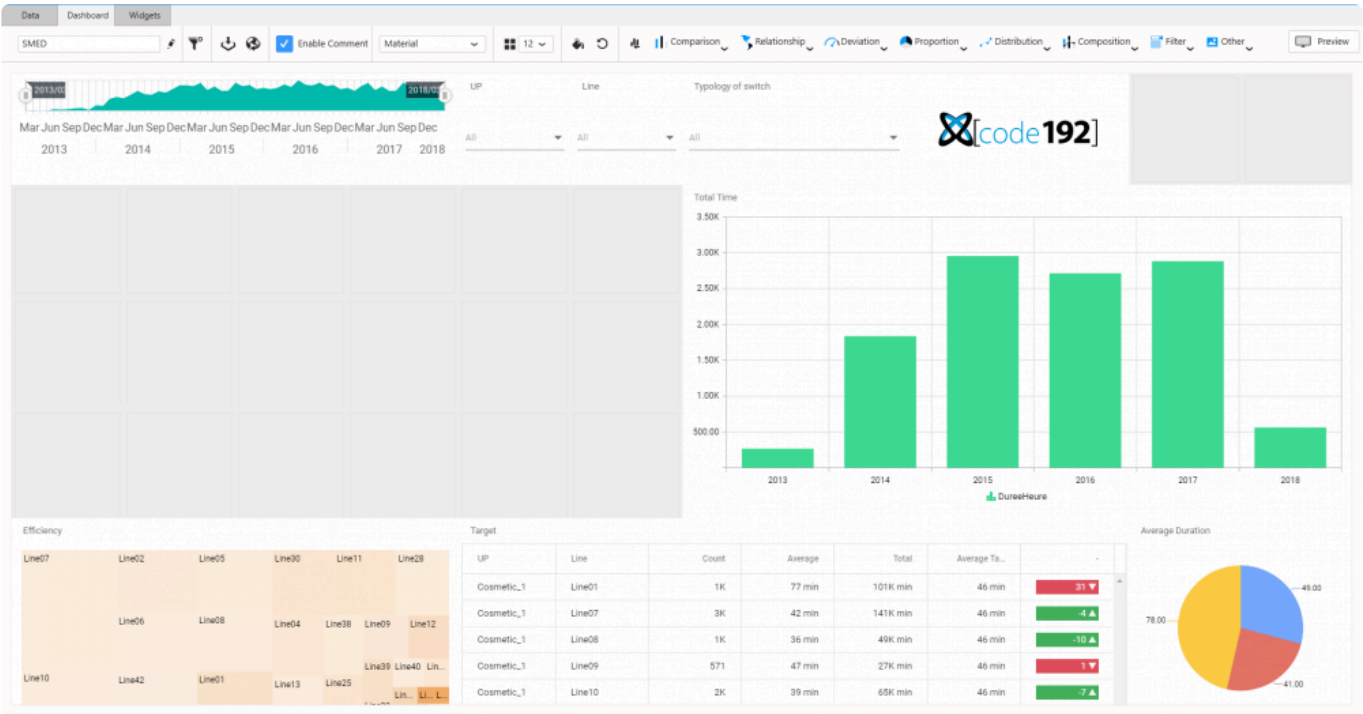
Dashboard Layout

In the *Dashboard* activity tab, the main area shows the Dashboard layout canvas where some square tiles are displayed to guide you.



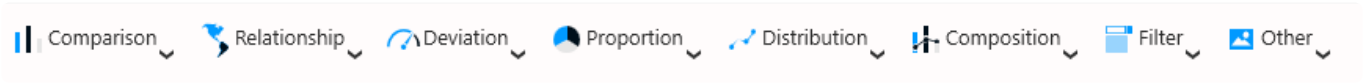
Created Widgets will appear on this layout canvas and can be re-arranged later.

For example, this is what a work in progress Dashboard will look like in this view :

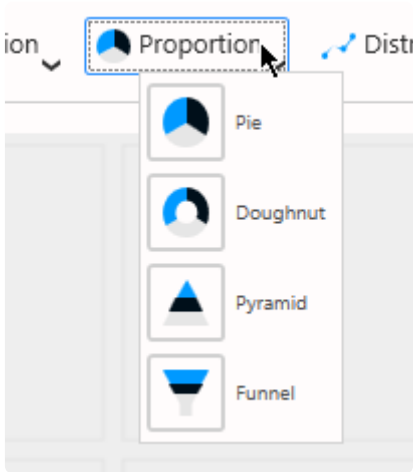


Browsing for available Widget types

In the *Dashboard* activity tab, the top bar displays drawers that contain the Widget library :

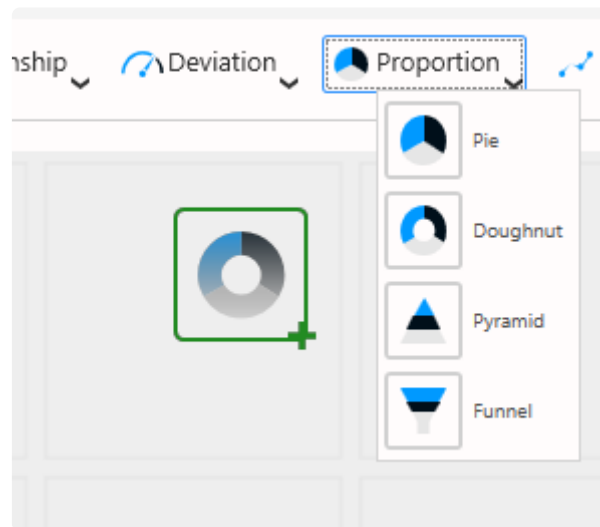


Click on a drawer to open it and list all widget types in this category :

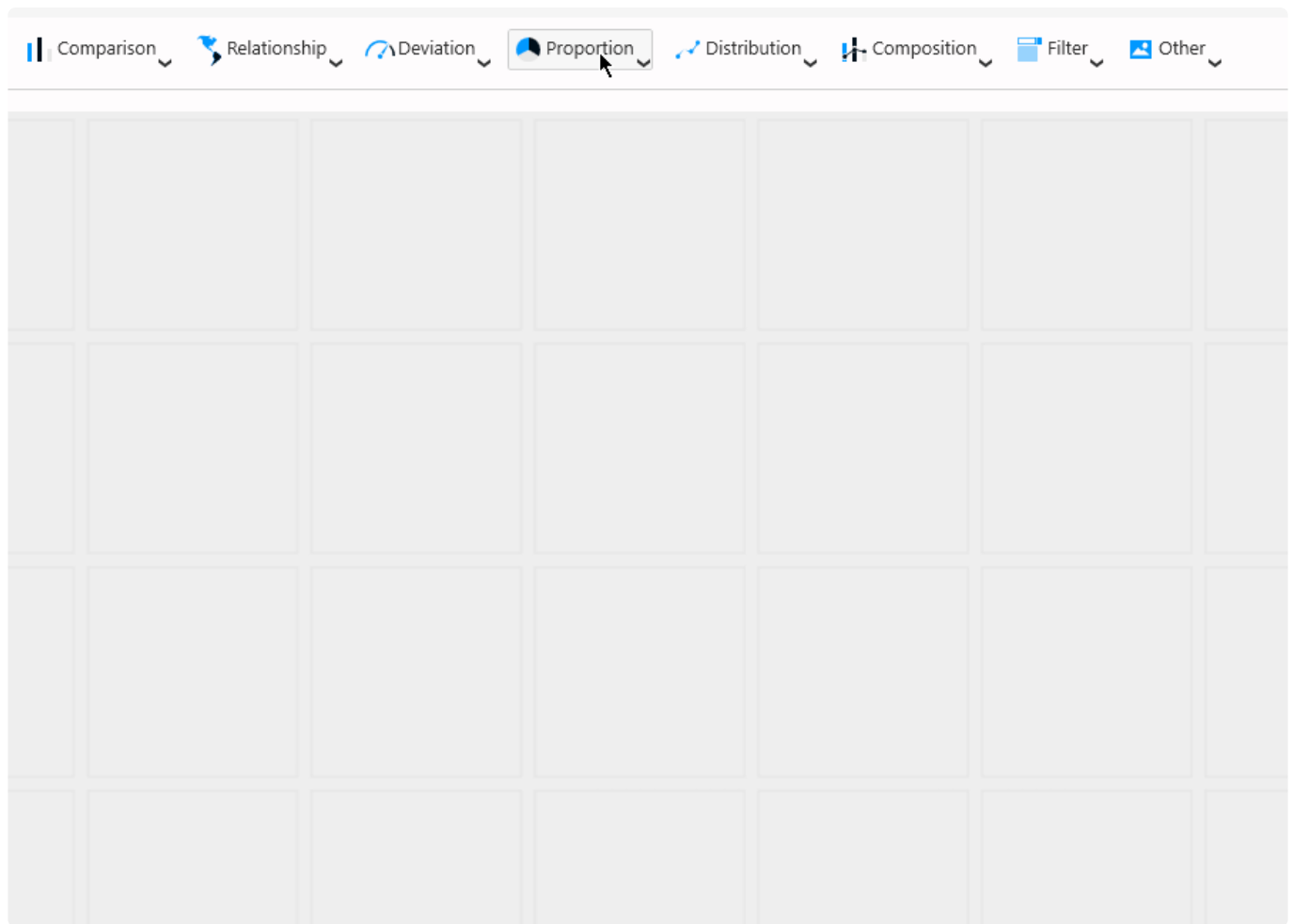


Adding a Widget

To add a Widget to the Dashboard, simply drag it from the drawer into any place in the layout canvas.



See the below animation for the full workflow :



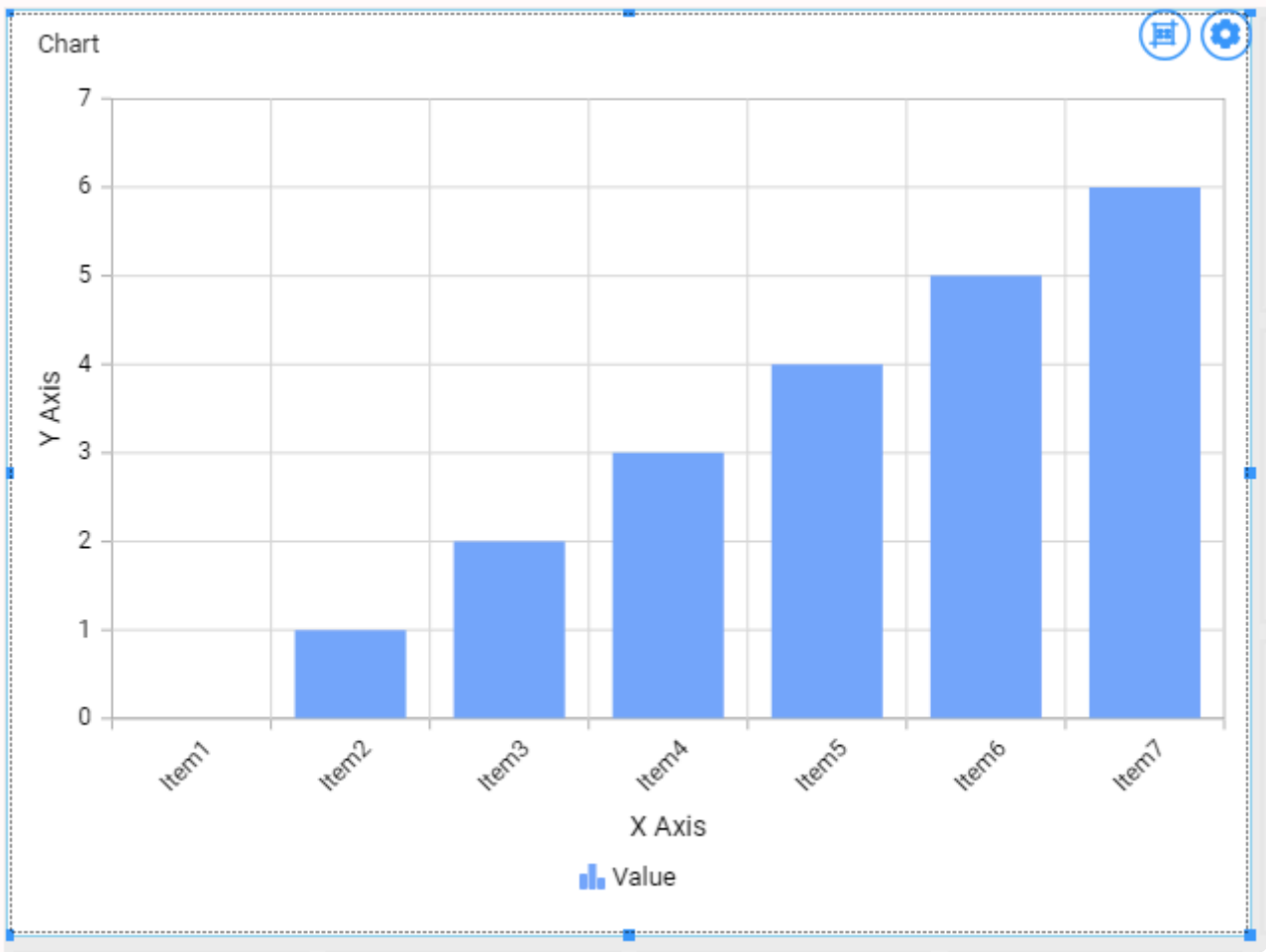
* Don't worry, you will be able to move, resize, delete, ... this Widget at any time later.

Note : Sample values

When a Widget is created, it is not configured.

However, in order to help you choose a Widget Type and have an idea of what the Dashboard will look like, some sample data are displayed, like "Item1, Item2, ...".

This data is a placeholder only and is not related to any real data.



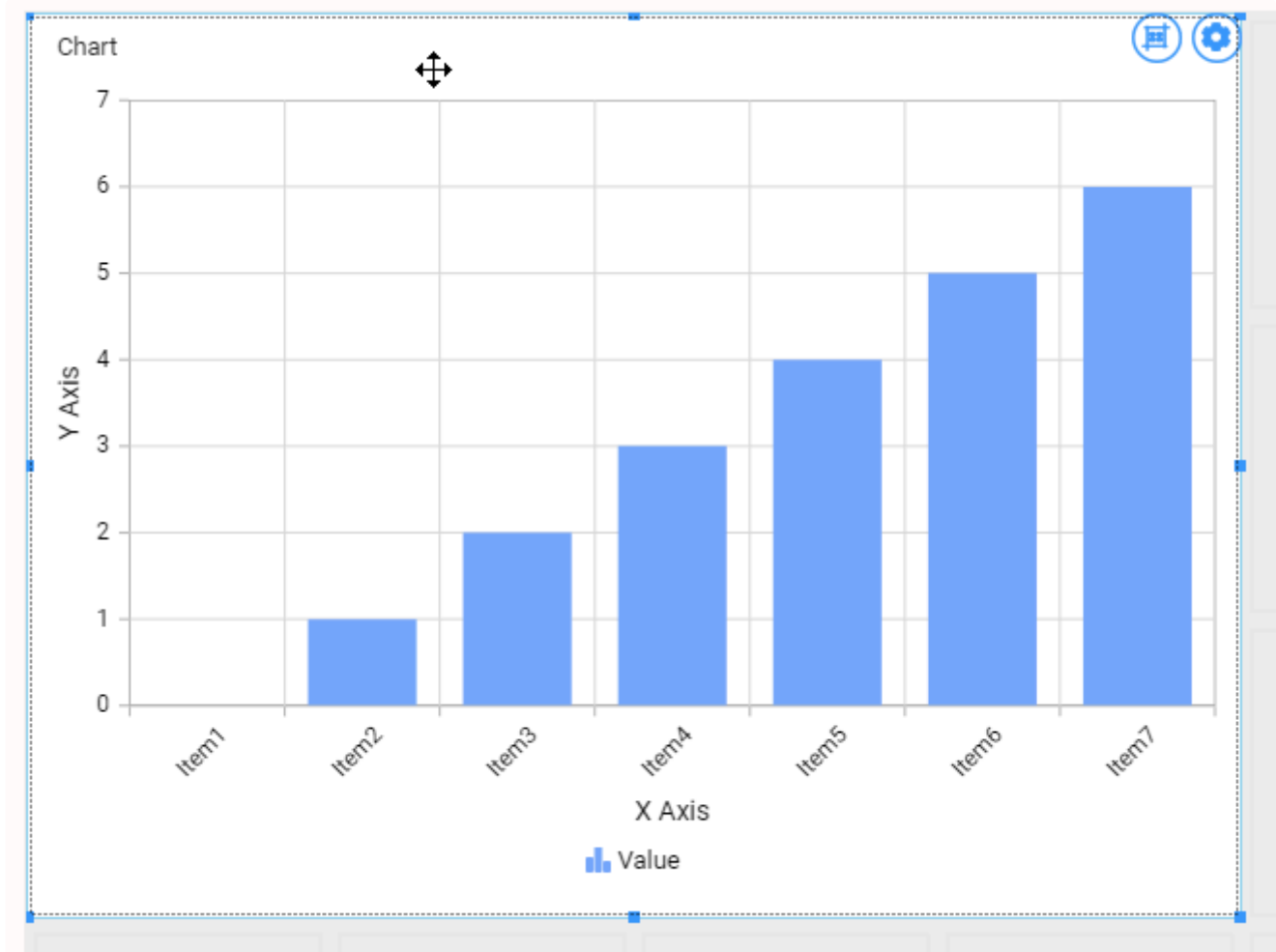
Last modified: 2019/04/19

2.6.1.2. Editting a Widget

How to Edit a Widget

From the Dashboard activity tab

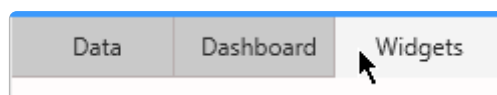
To open an existing Widget to start to Edit it, first left click on the Widget on *Dashboard* activity tab, so that it appears selected :



Then on the cog icon on the top right of the Widget itself :

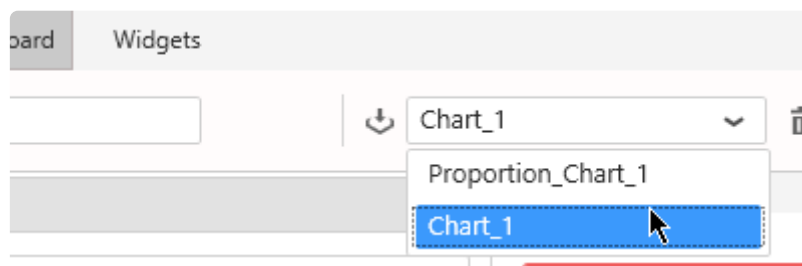


This will take you to the *Widgets* activity tab focused on this Widget



From the Widgets activity tab

When already inside the *Widgets* activity tab, you can select a Widget to edit from the list in the top bar :

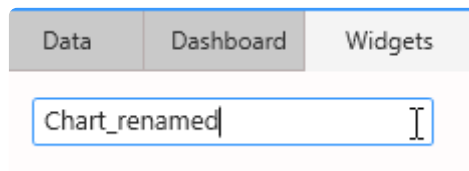


✿ All the actions to actually edit the Widget are explained in the following chapters.

Last modified: 2019/04/26

2.6.1.3. Renaming a Widget

When the Widget is already open to Edit, you can use the top bar on the left to change its name :



The screenshot shows a top navigation bar with three tabs: 'Data', 'Dashboard', and 'Widgets'. The 'Widgets' tab is currently selected. Below the tabs is a text input field with the text 'Chart_renamed' and a cursor at the end of the text.

No need to validate, the name is updated live.



Note :

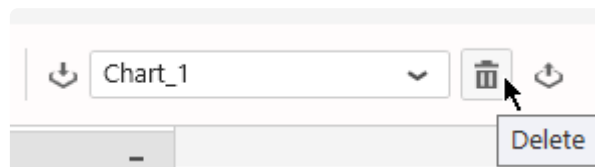
This Widget name is used internally in the widgets list for example. However, this is not the Widget header displayed at runtime. To edit the header displayed at runtime, see the corresponding chapter in [Header and Description](#).

Last modified: 2019/06/17

2.6.1.4. Deleting a Widget

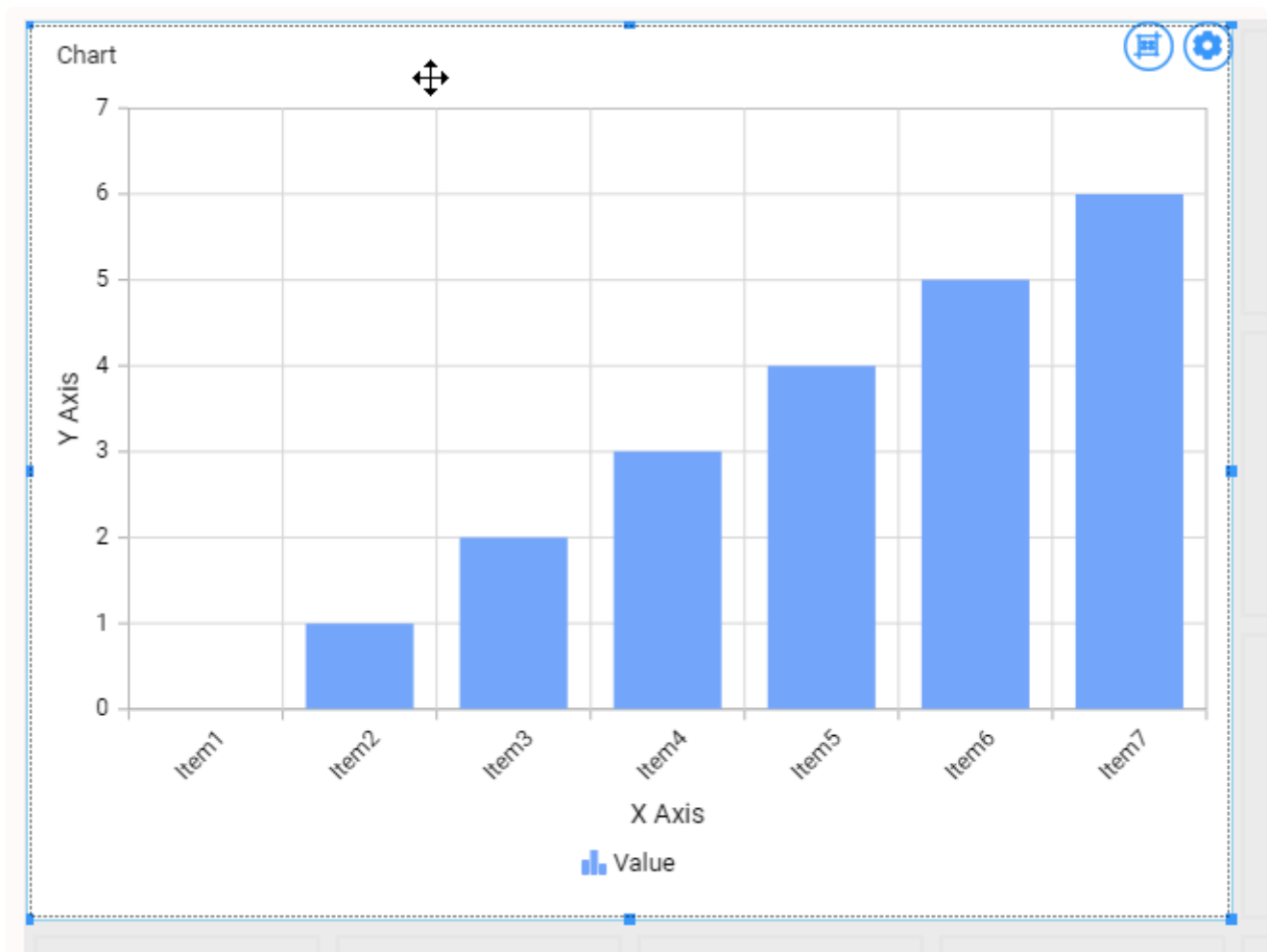
From the Widgets activity tab

To delete a Widget, select it in the *Widgets* activity tab top bar, then click on button “Delete” :



From the Dashboard activity tab

To delete a Widget when you are in the main *Dashboard* activity tab, first left click on the Widget, so that it appears selected :



Then press the “Delete” key on your keyboard.

Last modified: 2019/04/19

2.6.2. Export / Import Widgets

Sharing Widgets allows to re-use your work or collaborate between users.

This chapter describes how to proceed.

Last modified: 2019/04/26

2.6.2.1. What is a Widget Export

A Widget can be exported to a file with format `*.alpw`

This Widget.sydw file contains :

- the configuration of the Widget itself
- its position on the Dashboard canvas
- the dependent Data Source
- all the dependent Connections, including their credentials
- all related Parameter binding (for filters)

Data Sources and Connections use an internal unique identifier.

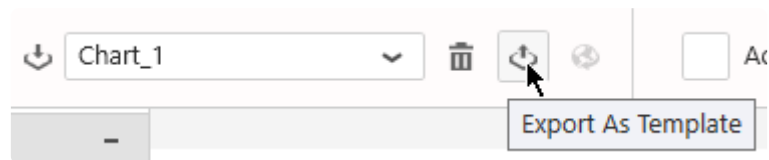
When importing several Widgets that were exported from the same Data Sources or Connections, the common elements are re-used instead of duplicated.

Last modified: 2019/03/14

2.6.2.2. Export a Widget to a file

How to Export to a file

In the top bar, click the *Export as Template* icon :



Select a file name and save locally.

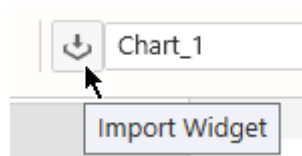
Last modified: 2019/04/19

2.6.2.3. Import a Widget from a file

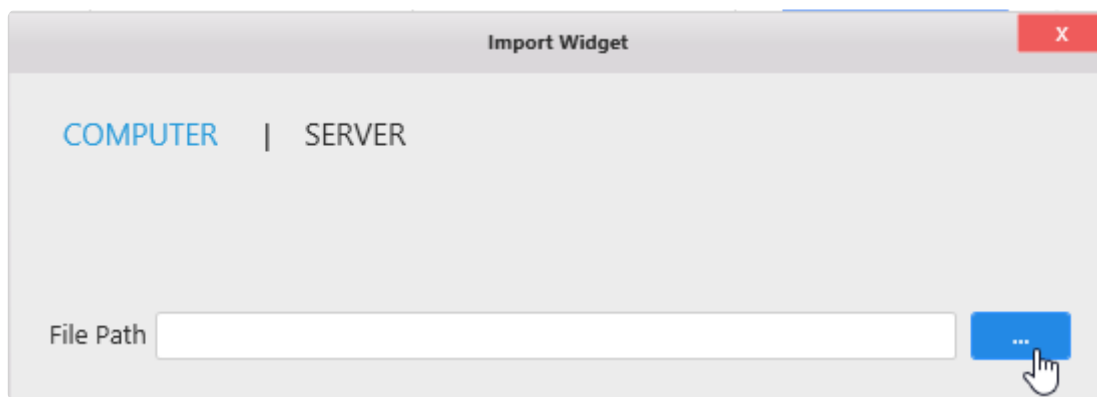
A Widget can be imported from a file in a few ways :

Import in an existing Dashboard

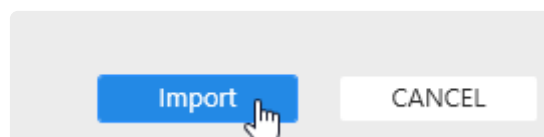
Click on the “Import” button in the Widgets list :



In the “Import Widget” dialog, under “Computer”, click “...” to browse for a Widget.alpw file :



Then click “Import” :



The Widget is now imported in your current Dashboard.

Create a new Dashboard based on a Widget file

In Windows Explorer, you can double click or open the Widget.alpw file with Alpana Designer.

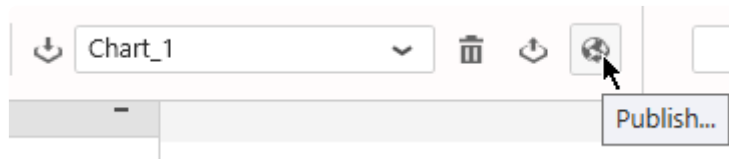
This creates a new Dashboard with the corresponding Widget imported.

Last modified: 2019/04/19

2.6.2.4. Publish a Widget to Alpana Server

* In order to Publish a Widget to Alpana Server, you need to be **logged in** from Alpana Designer to that Alpana Server, and have the necessary **permissions** to create Widgets.

In the top bar, click *Publish* :



In the “Publish” dialog, fill in details and click “Publish” :

A screenshot of the 'Publish' dialog box. The dialog has a title bar with 'Publish' and a close button. It contains three text input fields: 'Widget name' (filled with 'Chart_1'), 'Description', and 'Version Comment'. Below these fields is a checkbox labeled 'Mark As Public'. At the bottom right, there are two buttons: 'Publish' (highlighted with a mouse cursor) and 'CANCEL'.

The Widget is now published to Alpana Server and available to other users depending on permissions.

Last modified: 2019/04/26

2.6.2.5. Import a Widget from Alpana Server

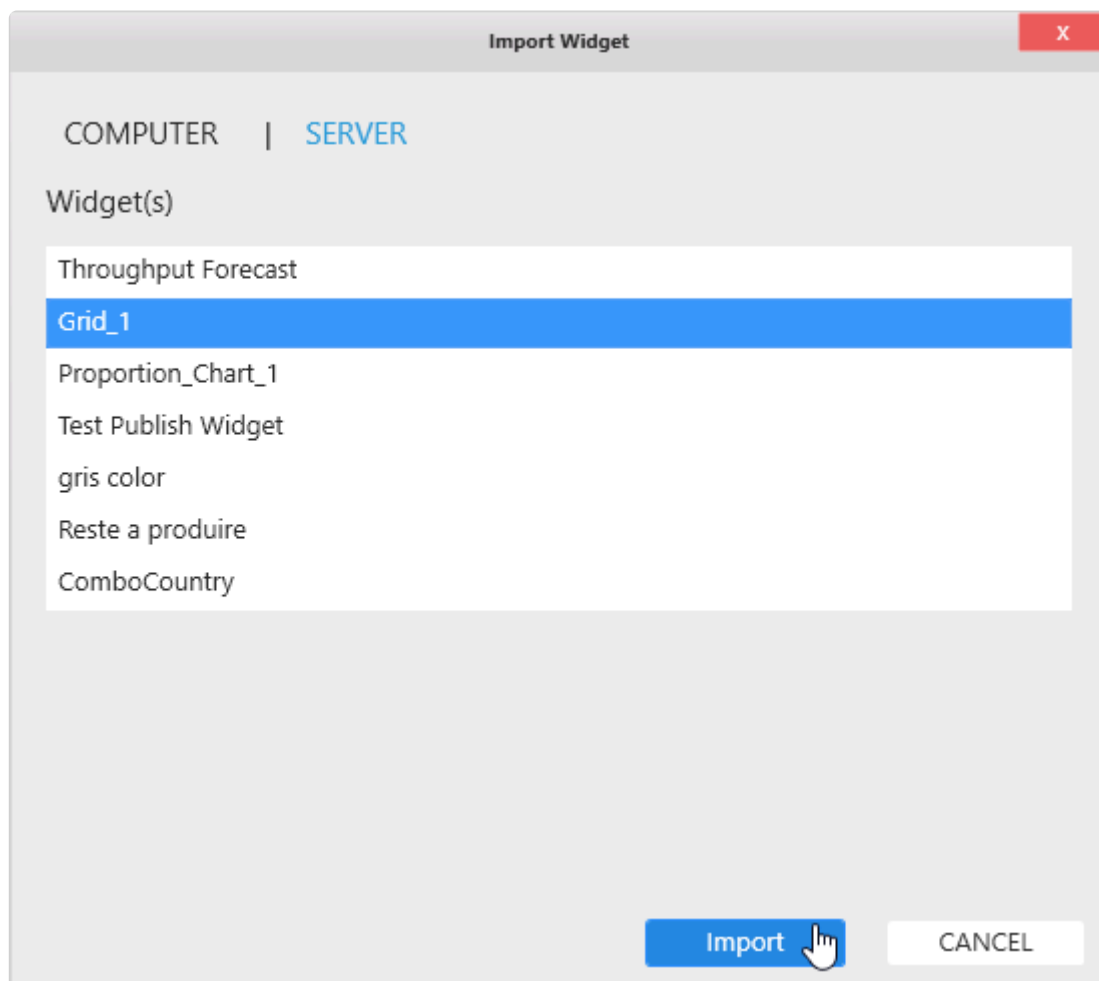
* In order to Import a published Widget from Alpana Server, you need to be **logged in** from Alpana Designer to that Alpana Server, and have the necessary **permissions** to access the Widget.

From the Dashboard activity

Click on the “Import Widget” button in the top bar :



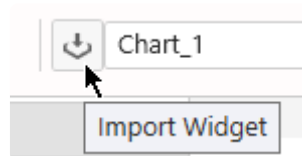
In the “Import Widget” dialog, click “Server”, select a published Widget, and click Import :



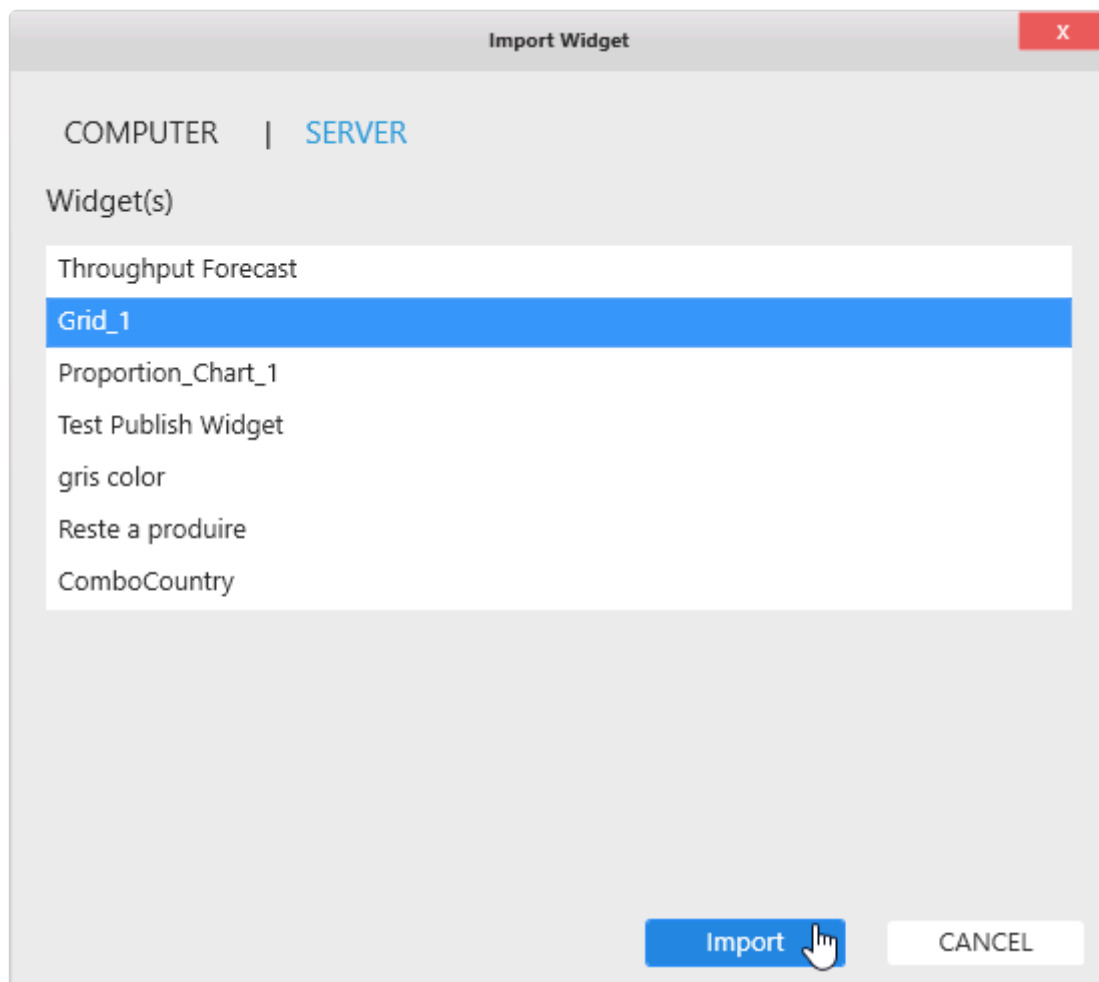
The Widget is now imported in your current Dashboard.

From the Widgets activity

Click on the “Import” button in the top bar :



In the “Import Widget” dialog, click “Server”, select a published Widget, and click Import :



The Widget is now imported in your current Dashboard.

Last modified: 2019/04/26

2.6.3. Widget Data Binding

Introduction

Each Widget displays data from a single Data Source.

Some Widget graphical properties are automatically generated from the values in the data. As a result, it is preferable to start Widget configuration by binding data.



In addition, data binding in Alpana Designer is very interactive and is a nice way to start exploring your data.

What is data binding ?

When binding data, you are in fact configuring a query to the Data Source.

Internally to Alpana, the result of this query is a table of values which will be used to dynamically change some graphical properties of the Widget.

For example, when you want to see the average temperature by Equipment on a Column Chart, the Widget will draw as many rectangles (columns) as there are Equipments, and the height of the rectangles (Y position) will be a number of pixels proportionate to the average temperature.

Why aggregate Quantities ?

In Dashboard visualization, we want to turn many data points into few simple indicators that allow to take a decision efficiently.

Several data rows will turn into a single shape (this is called “*grouping*”), and a single data value will be associated to this shape (this is called “*aggregating*”).

Even if you “know” that for each group your data contains only a single data row right now, your KPI definition needs to account for the fact that there could be many.

Example

You are trending the temperature from a sensor. Right now you know you only have 1 data point at each instant.

So why would you need to choose an aggregation for the Temperature ?

Well, tomorrow you will need to integrate the temperature measurements from 3 sensors. Then, at each instant you will have 3 data points, but your chart is configured to show only 1 trend.

To solve this, you will need to create an aggregation expression.
Maybe a simple `AVG(Temperature)`, or maybe something more complex involving Alpana Expressions.

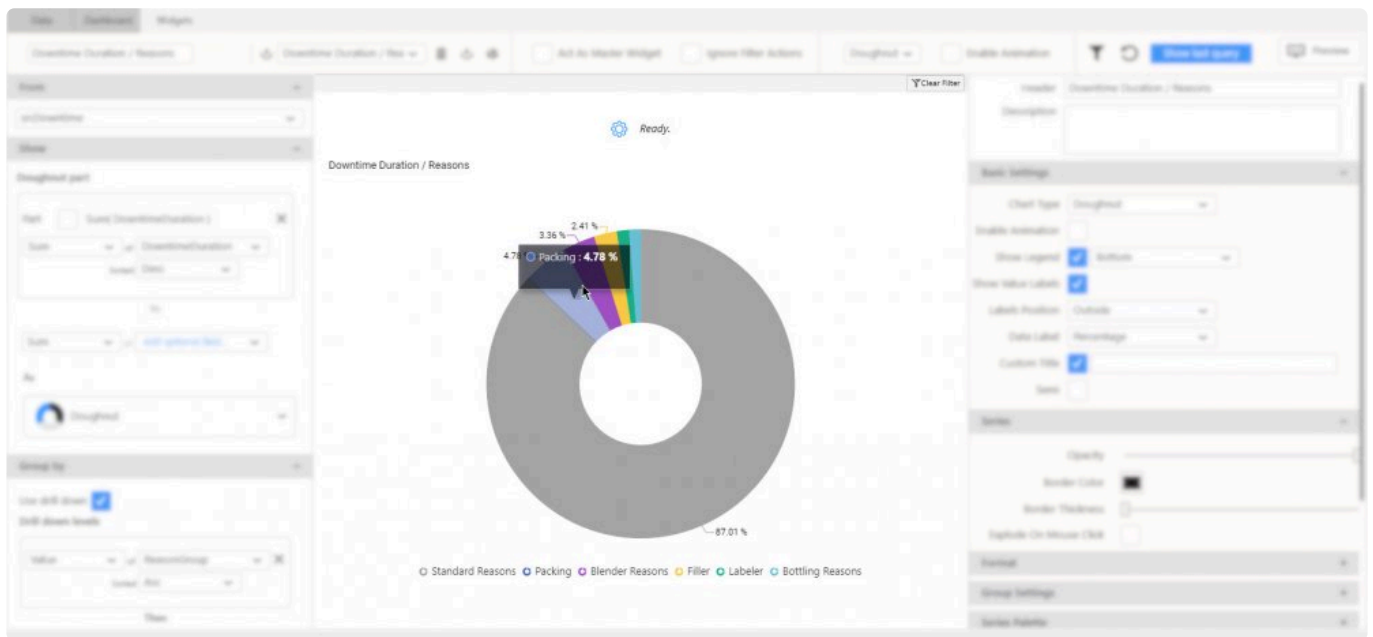
Last modified: 2019/04/30

2.6.3.1. using the Widget Preview

Widget Preview

When editing a Widget, the middle part of the window contains a live Widget preview.

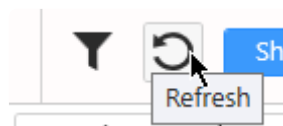
Every time you make a change to the Widget configuration, the Widget Preview is refreshed to reflect that change.



This preview is interactive and can be used to show tooltips, click to drill-down, or scroll to zoom on maps.

Refreshing the Widget Preview

This Widget Preview can be manually refreshed using the refresh button in the top bar :



This makes a new query to the data, and refreshes the preview.

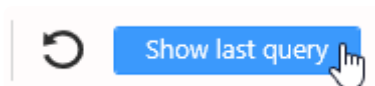
Last modified: 2019/04/29

2.6.3.2. Troubleshooting Widget data

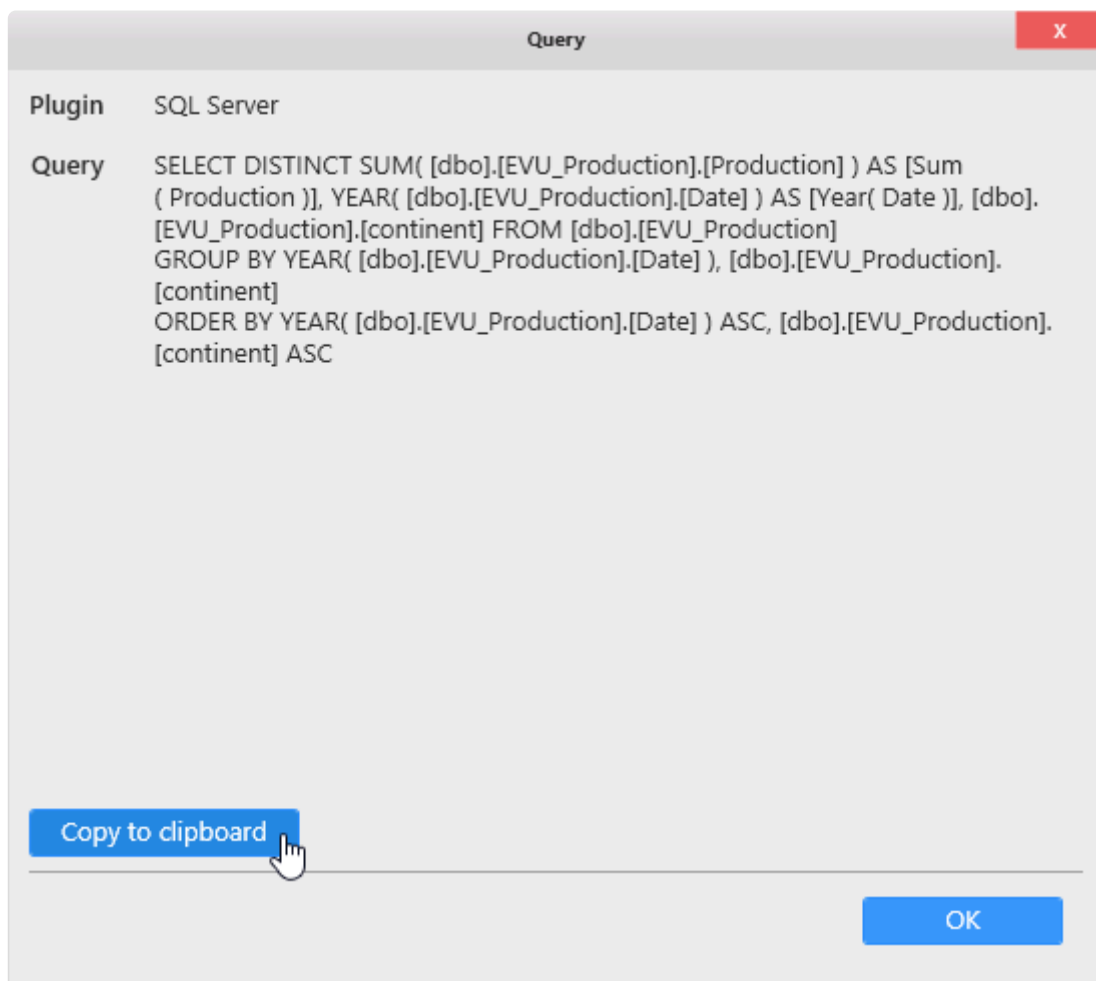
Show last query

In order to troubleshoot widget data, it is possible to get the last data query that was successfully composed for this Widget.

Select the desired Widget, and click the “Show last query” button on the top :



A dialog pops up and lets you see the last successful query for this Widget. You can also press “Copy to clipboard” in order to paste this query in your source data tool and investigate the query’s results manually :

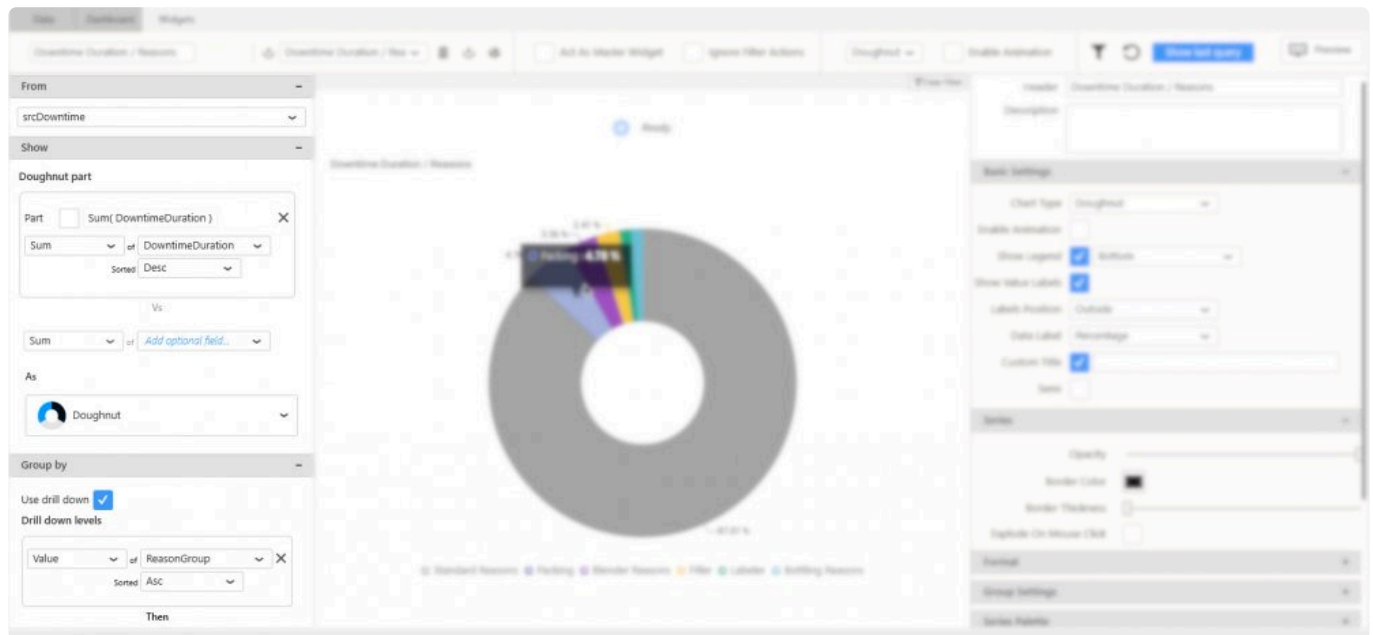


Last modified: 2019/04/29

2.6.3.3. Editing Widget data binding

Accessing the data binding

Starting with Alpana v3.0.0, all Widget data binding happens in the data panel on the left of the Widgets activity :



Folding / Unfolding sections

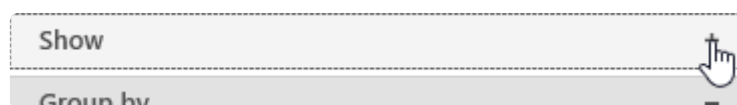
Different parts of the data binding interface are grouped into sections.

For better readability, each section can be :

... folded ...



... and un-folded :



Data binding workflow

The data binding should read like a sentence :

> From *My Data Source*, Show *My Aggregated Measure Field* Group By *My Dimension Field*.

From : Data Source selector

This is where you will select the Data Source to bind to this Widget.

(see chapter [Selecting a Data Source](#))

Show : Quantities

This is where you will configure the KPI : what **quantity** you want to visualize.

This will be an **aggregated Measure**.

(see chapter [Configuring the aggregation of number fields](#) and chapter [Binding a data field](#))

Group : Context

This is where you will define what **context** you will use to group data together.

This will be an un-aggregated **Dimension** with unique separate values.

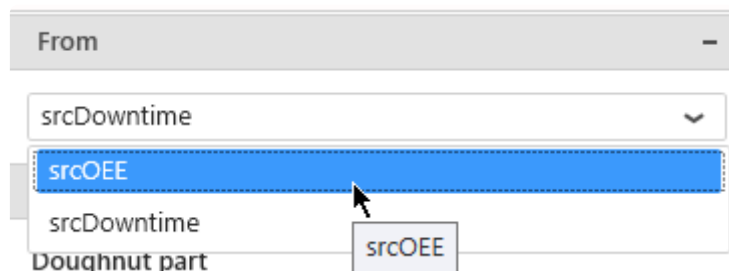
(see chapter [Configure the grouping of category fields](#) and chapter [Binding a data field](#))

Last modified: 2019/04/30

2.6.3.4. Selecting a Data Source

* Each Widget binds data from a single Data Source.

In order to select a Data Source for this Widget, open the Data Source listbox on the top of the Data panel, under the **From** title :



Warning

If the Widget had any data binding from another Data Source, it will be lost !

Last modified: 2019/04/26

2.6.3.5. Configuring the aggregation of number fields

Introduction

Most Widgets have configuration fields that require to generate a single number for each corresponding shape in the Widget.

Examples :

- Chart : the chart line Y position of each point
- TreeMap : the rectangle relative size for each rectangle
- ...

Since data is grouped in a Widget query (“for each point”, “for each rectangle”, ...), numerical fields must be aggregated in order to give a single number for each Widget “shape”.

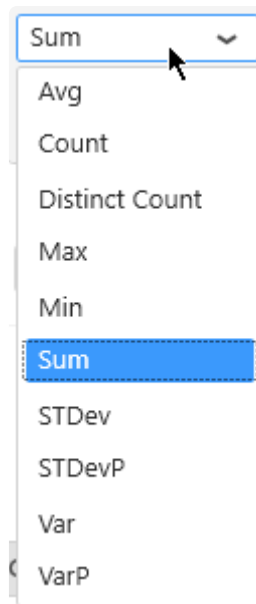
This operation is done with an aggregation function.



An Aggregation Function is a function that can take any number of values and turn them into a single number.

For Measure fields

For Measure fields (numbers), the following aggregation functions are available : Avg, Count, Distinct Count, Max, Min, Sum, STDev, STDevP, Var, VarP.



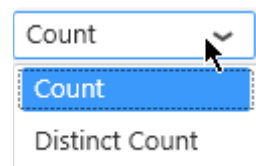
These functions absolutely require a number (except `Count` and `Distinct Count`, see below).

They translate to the corresponding functions in your source data engine (SQL, SQLite, ... depending on your Data Source). Please refer to the corresponding documentation if you need to understand better these functions.

For Dimension fields

Dimension fields (strings, dates, booleans) can also be used to generate numbers for Widget queries.

The following aggregation functions are available : `Count`, `Distinct Count`.



These functions accept all data types.

They translate to the corresponding functions in your source data engine (SQL, SQLite, ... depending on your Data Source). Please refer to the corresponding documentation if you need to understand better these functions.

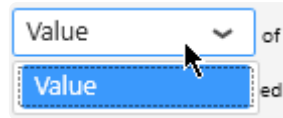
For Aggregated Expressions

(see [corresponding chapter](#))

Aggregated Expressions already guarantee they provide a single number, and no aggregation function

can be used on top of them.

In order to use an Aggregated Expression as a number field, select “*Value of*” instead of an aggregation function :



Last modified: 2019/04/29

2.6.3.6. Configure the grouping of category fields

Introduction

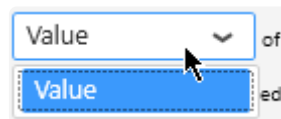
Most Widgets have configuration fields that are used to generate separate shapes. These fields are used to group numerical fields together and generate a single shape for each context combination.

Examples :

- Chart : the chart column category (x position) of each point ; the creation of each trend series
- TreeMap : the creation of each rectangle
- ...

Grouping function

There is no particular grouping function.
Only “*Value of*” is available.



It creates a group for each distinct value of the selected field.

For string, number and boolean fields

There is no particular configuration for the following types of fields :

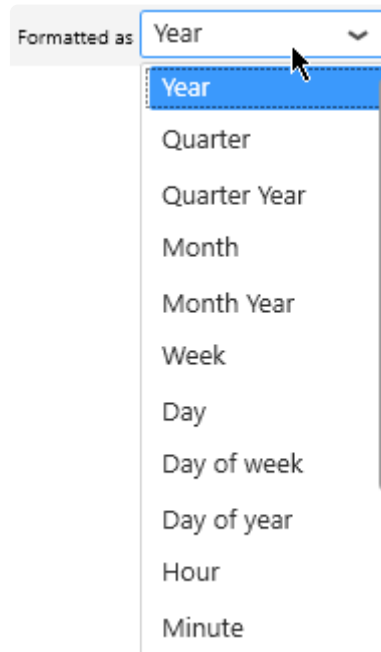
- string
- number
- boolean

When they are added to a category configuration, they group the query without any formatting : each separate value creates a separate group.

* Note : [Aggregated Expressions](#) cannot be used as category fields because they only generate a single value, which is not desirable.

For date fields

Date fields can be formatted to create different groups :



- Year (default when adding a field)
- Quarter
- Quarter Year
- Month
- Month Year
- Week
- Day Of Week
- Day Of Year
- Day
- Hours
- Minutes
- Seconds
- Date
- Date Time : the full datetime detail (no formatting happens, each different value is a different category)
- Time


Last modified: 2019/04/29

2.6.3.7. Binding a data field

Mandatory fields, optional fields

In order to complete the Widget configuration, some Widget properties **must** be configured : the Widget will not work without this field bound.

The corresponding field selector is marked with “*Select required field...*”



Some fields are optional : you may need to configure them in order to get what you want, but the Widget will still work without this field bound.

The corresponding field selector is marked with “*Select optional field...*”



When you want to configure a field, just use those field selectors to select the desired field.

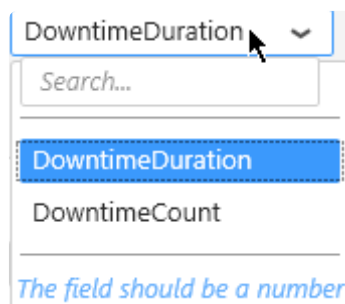
Selecting a field to bind

Depending on the aggregation function selected and the property you are trying to configure, not all field types may be available.

For example, if the aggregation function `Sum` is selected, then only the numerical fields and numerical Expressions are listed.

Number fields

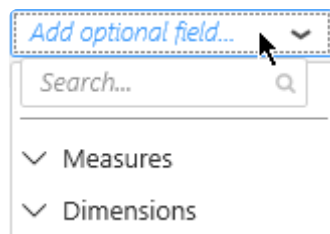
If the configuration only accepts numbers, all number fields are listed in a single flat selector :



“Any” fields

If the configuration accepts any type of fields, they are organized in 2 categories :

- *Measures* : the numerical fields from your data.
- *Dimensions* : the non-numerical fields from your data.



Just click a category to unfold it and list all fields of that type :



Searching for a field

Instead of browsing for a field, you can use the search box to start typing part of the field name :



This will incrementally list all fields partially matching the search pattern.

To clear the search, simply click the cross button next to the search box :

Add optional field... ▾

rea

Reason

Last modified: 2019/04/30

2.6.3.8. Un-binding a data field

To remove a field binding from a Widget, click the cross inside the field container :



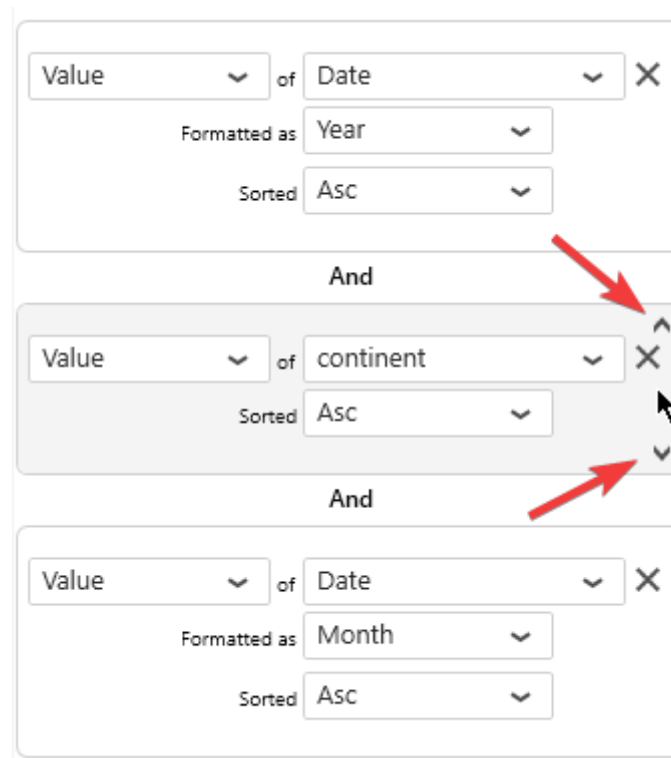
! All related configuration will be lost.

Last modified: 2019/04/30

2.6.3.9. Re-Ordering data fields

Some Widget configurations rely on the order of the list of bound fields.

In order to change the order of a field, hover it, and see small *up* and *down* arrows appear :



Click the *down* arrow to move the field down :



As a result, the field is swapped with the below field :

Value

▼

of

Date

▼

×

Formatted as

Year

▼

Sorted

Asc

▼

And

Value

▼

of

Date

▼

×

Formatted as

Month

▼

Sorted

Asc

▼

And

Value

▼

of

continent

▼

×

Sorted

Asc

▼

Click the *up* arrow to move the field up :



Last modified: 2019/05/07

2.6.4. Configure a sorting order

Introduction

Most Widgets will present data differently depending on how it is sorted in the query.

Example : a Chart is by default sorted by ascending X Axis values. It can instead be sorted by descending Y Axis values.

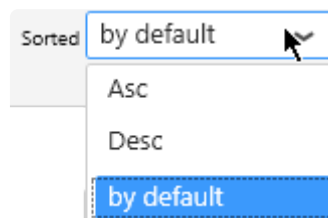
Sorting Priority

As a rule, the fields at the top of the Data binding panel will have sorting priority over the fields below.

Example : if you set a Chart Y Value field sorted descending, it will override the X Axis sorting because it is located above in the Data binding panel.

Configure sorting

On each bound data field, you can configure sorting :



- *by default* : do not take part in query sorting
- *Asc* : sorted in Ascending order
- *Desc* : sorted in Descending order

Last modified: 2019/04/30

2.6.5. Widget-level Filtering

Filtering data at the Widget level

You can apply filters at the Widget level.

The current Widget will only see the data after the filter is applied.

Those **Widget-level filters** are applied on top of any [Data Source filters](#) (if any) and apply only to the current Widget.

Last modified: 2019/04/30

2.6.5.1. Opening the Filter Editor

Add or Edit Filters

To add Filters to a Widget or edit existing filters, press the “Open Widget Filter Editor” funnel icon in the top toolbar :



See the presence of Filters

When a filter is applied on a Widget, the Filter button is colored :

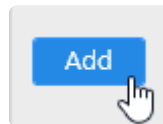


Last modified: 2019/04/30

2.6.5.2. Managing Filter Conditions

Adding a condition

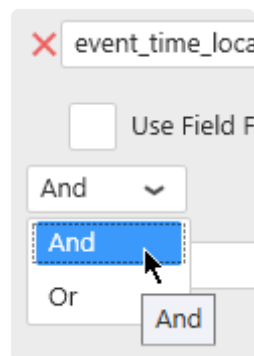
In the Filter Editor, click Add to add a filter condition :



Combining conditions

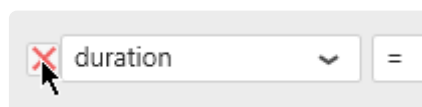
If filter conditions already exist, they will be chained using a logical operator.

By default, the operator is “And”, but it can be changed to “Or” :



Removing Conditions

Click the red cross on the left to remove the corresponding condition :



Last modified: 2019/04/30

2.6.5.3. Filtering aggregated values

Introduction

There are 2 ways to apply a widget filter :

- on the **raw** values : each row in the raw dataset is evaluated against the filter condition
- on the **aggregated** values : the raw dataset is first aggregated by the Widget, and only *then* the filter condition is applied

Example

The Data Source

Imagine a simple dataset :

TagName	Building	Product	Price
FT101.Value	Building A	Product X	10
FT102.Value	Building A	Product Y	15
FT201.Value	Building B	Product X	20
FT202.Value	Building B	Product Y	25
FT301.Value	Building C	Product X	30
FT302.Value	Building C	Product Y	35

The Widget

You are displaying this data in a Pie Chart with `Sum(Price)` grouped by “Product”, you get :

Product	Sum(Price)
Product X	60
Product Y	75

Filtering raw values

You apply a filter on the **raw** value to see only data where :

`Price > 20`

The raw dataset will be filtered like this :

TagName	Building	Product	Price
FT202.Value	Building B	Product Y	25
FT301.Value	Building C	Product X	30
FT302.Value	Building C	Product Y	35

And the Widget will display the following :

Product	Sum(Price)
Product X	30
Product Y	60

(because $60 = 25 + 35$)

Filtering aggregated values

You apply a filter on the **aggregated** value to see only data where :

`Sum(Price) > 70`

It is not the raw dataset that will be filtered.

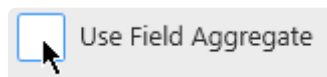
The Widget will display the following :

Product	Sum(Price)
Product Y	75

(Product X was filtered because its Sum(Price) was $60 \leq 70$)

How to filter by aggregated values

For each filter condition, you can check *Use Field Aggregate* :



Then select the desired aggregation function :

A screenshot of a software interface showing a dropdown menu for field aggregation. The menu is titled 'Use Field Aggregate' with a blue checkmark icon. The dropdown list contains the following options: Avg, Count, Distinct Count, Max, Min, Sum, STDev, STDevP, Var, and VarP. The 'Avg' option is currently selected and highlighted in blue.

Note on non-numerical fields

On fields where the data type is not number (string, date, boolean), the aggregation functions will return a number, so the condition becomes a number condition.

For those fields, the list of available aggregation functions is also different :

A screenshot of a software interface showing a dropdown menu for field aggregation. The menu is titled 'Use Field Aggregate' with a blue checkmark icon. The dropdown list contains the following options: Count and Distinct Count. The 'Count' option is currently selected and highlighted in blue.

For example, the resulting filter condition below reads as :

Distinct Count of EquipmentName < 3

A screenshot of a filter condition configuration interface. It shows a red 'X' icon followed by a dropdown menu containing 'EquipmentName'. To the right of this is a comparison operator dropdown menu showing '<'. Further right is a text input field containing the number '3'. Below this row, there is a blue checkmark icon followed by the text 'Use Field Aggregate' and a dropdown menu showing 'Distinct Count'.

Last modified: 2019/04/30

2.6.5.4. Configuring Filter Conditions

Filter conditions are expressed with :

- a field name from the Data Source. This can also be an [Expression](#).
- a boolean operator
- a value. This can also be a [Parameter](#).

Depending on the Data Source field type, the settings are slightly different, please see for each type in the next pages.

Last modified: 2019/04/30

2.6.5.4.1. Filtering string fields

This is the most basic case.

The filter condition will be :

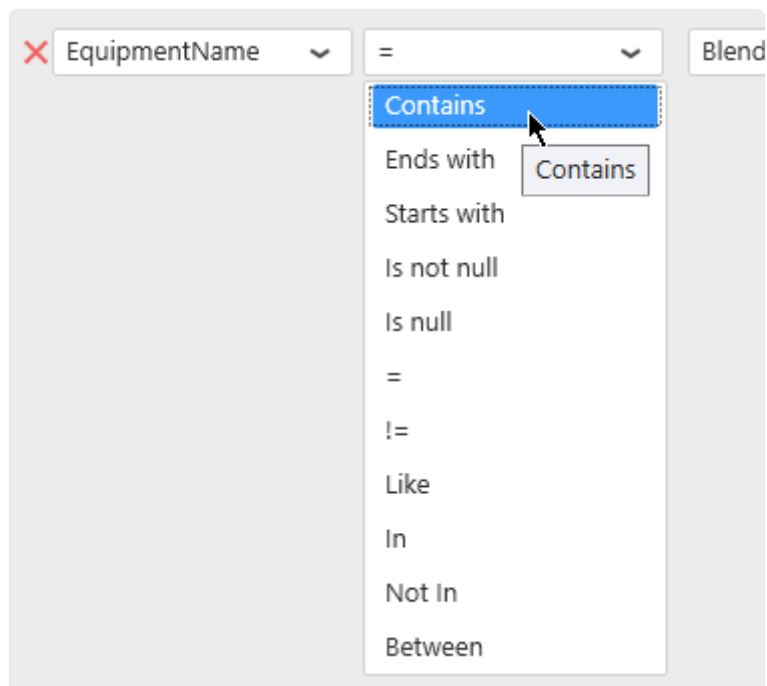
<FieldName> <Operator> <Value>

Example :

EquipmentName StartsWith Blender

Operators

Operators appropriate to Strings can be selected.



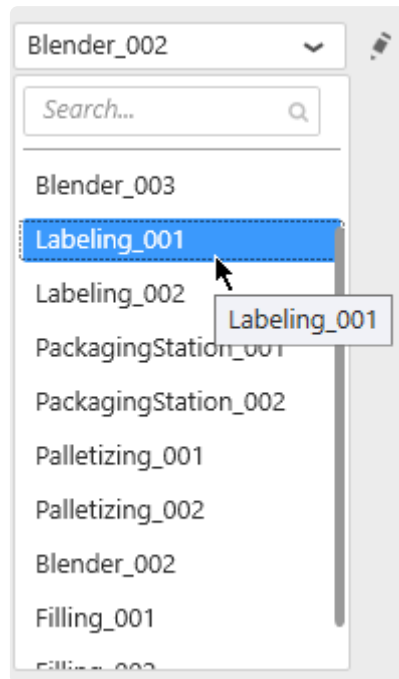
Values

Static values

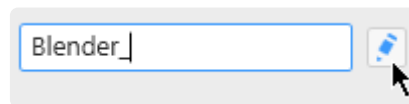
The “Edit values” button can be check/unchecked manually :



When the “Edit values” button is un-checked, a value can be selected from current values in the database :



When the “Edit values” button is checked, a value can also be entered as text :



* Depending on the operator selected, some of these options may not be available.

Dynamic filter values using Parameters

Instead of comparing to a static value, it is possible to compare to a dynamic Parameter that will depend on actions inside the Dashboard.

For this, select “Use Parameter” and see the Chapter about [Parameters](#).



Last modified: 2019/04/30

2.6.5.4.2. Filtering number fields

In a basic configuration, the condition is expressed as :

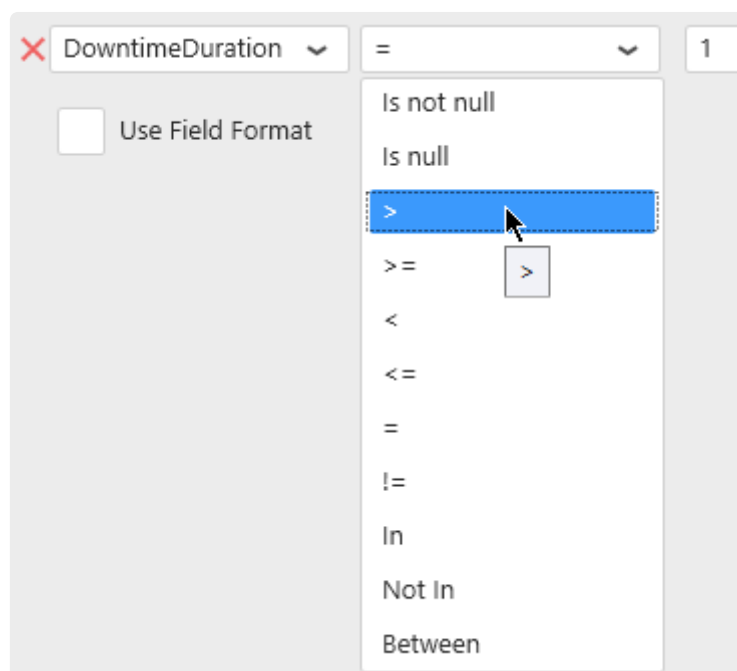
```
<FieldName> <Operator> <Value>
```

For example :

```
DownTimeDuration > 10
```

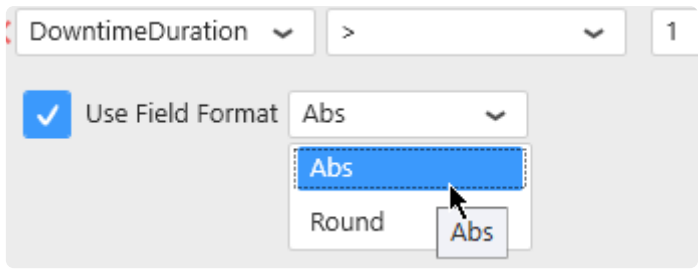
Operators

Operators appropriate to Numbers can be selected :



Field formatting

In addition to the basic configuration, a formatting function can be applied to the numerical field while the comparison is made :



This way, the filter condition will be :

```
<Function>(<FieldName>) <Operator> <Value>
```

For example :

```
Abs (DownTimeDuration) > 10
```

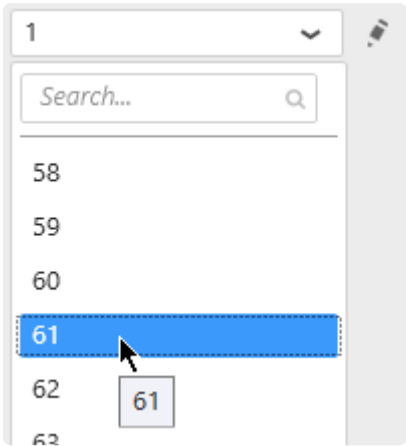
Values

Static values

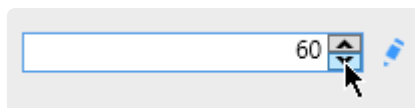
The “Edit values” button can be check/unchecked manually :



When the “Edit values” button is un-checked, a value can be selected from current values in the database :



When the “Edit values” button is checked, a value can also be entered as text, or using the up/down arrows to increase/decrease :



✿ Depending on the operator selected, some of these options may not be available.

Dynamic filter values using Parameters

Instead of comparing to a static value, it is possible to compare to a dynamic Parameter that will depend on actions inside the Dashboard.

For this, select “Use Parameter” and see the Chapter about [Parameters](#).



Last modified: 2019/04/30

2.6.5.4.3. Filtering date fields

In a basic configuration, the condition is expressed as :

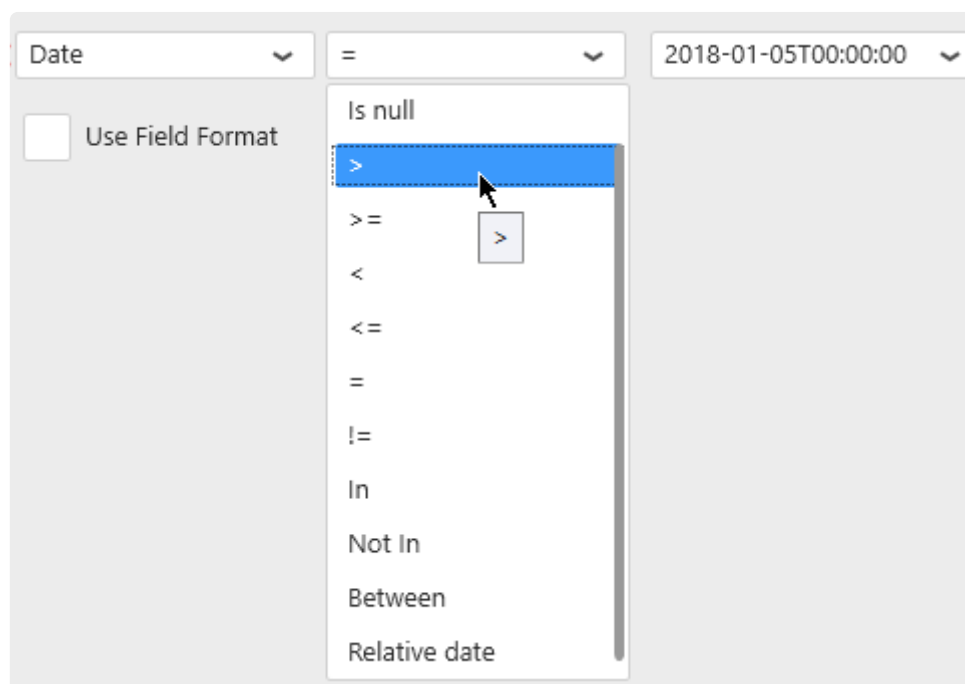
```
<FieldName> <Operator> <Value>
```

For example :

```
aaValueStartTime > 2018-01-05T00:00:00
```

Operators

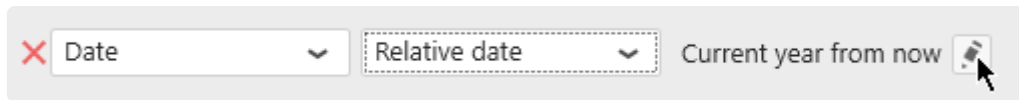
Operators appropriate to Dates and Times can be selected :



Relative date filtering

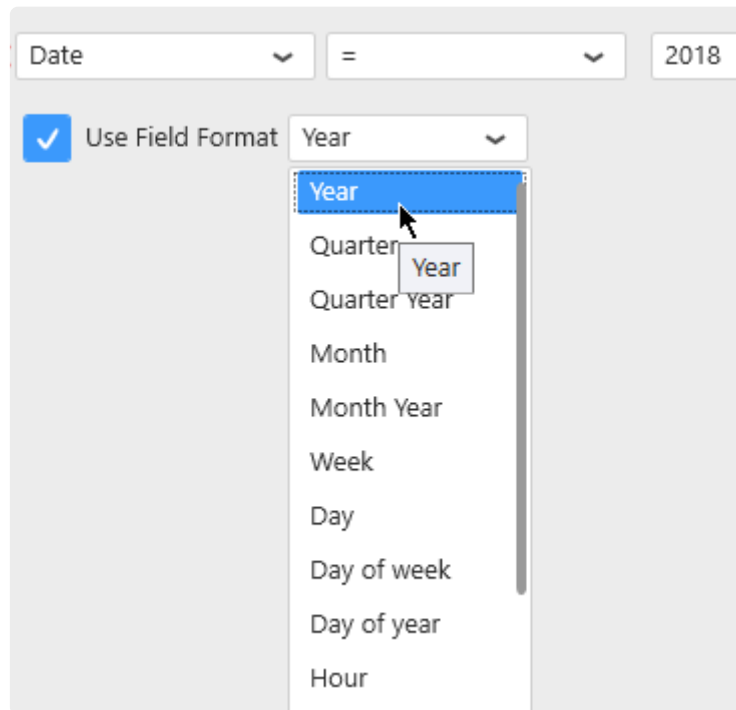
The special operator “Relative date” allows to select a “date window” / “time frame” that is moving over time relative to the current moment.

When selecting this operator, click the edit icon to define the relative date window :



Field formatting

Similarly to the number fields, a formatting function can be applied to the date field :



This way, the filter condition will be :

```
<Function>(<FieldName>) <Operator> <Value>
```

For example :

```
Year(Date) BETWEEN 2015 AND 2016
```

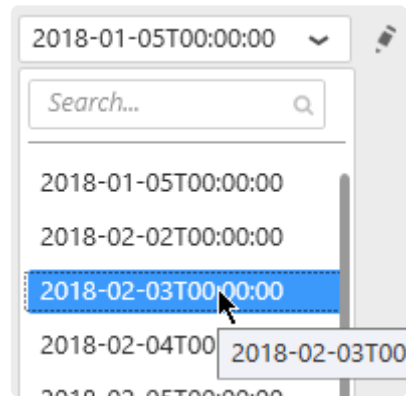
Values

Static values

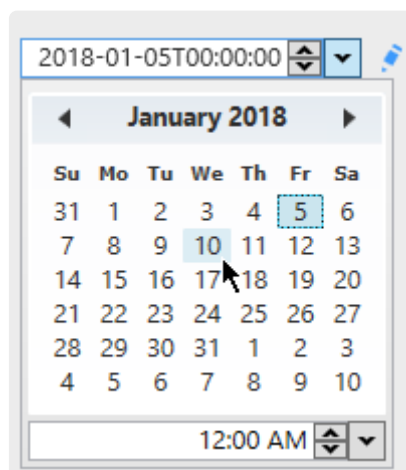
The “Edit values” button can be check/unchecked manually :



When the “Edit values” button is un-checked, a value can be selected from current values in the database :



When the “Edit values” button is checked, a value can also be entered as text :



Depending on the operator and formatting function selected, some of these options may not be available.

Dynamic filter values using Parameters

Instead of comparing to a static value, it is possible to compare to a dynamic Parameter that will depend on actions inside the Dashboard.

For this, select “Use Parameter” and see the Chapter about [Parameters](#).

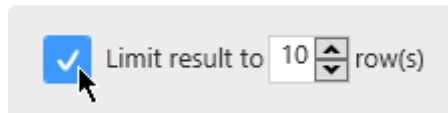


Last modified: 2019/04/30

2.6.5.5. Limiting dataset size

How to limit

Check “Limit result to” and select the number of rows you want to keep in the dataset :



Why limit ?

Checking this option with “Limit result to 10 rows” for example will apply a “TOP 10” to the query. When widgets are sorting the data, this can be a useful to display the 10 Items with the worst performance, or the 10 equipments with the highest consumption for example.

Last modified: 2019/04/30

2.6.6. General Widget Properties

All Widget types have specific Properties that are documented in [the corresponding chapter](#).

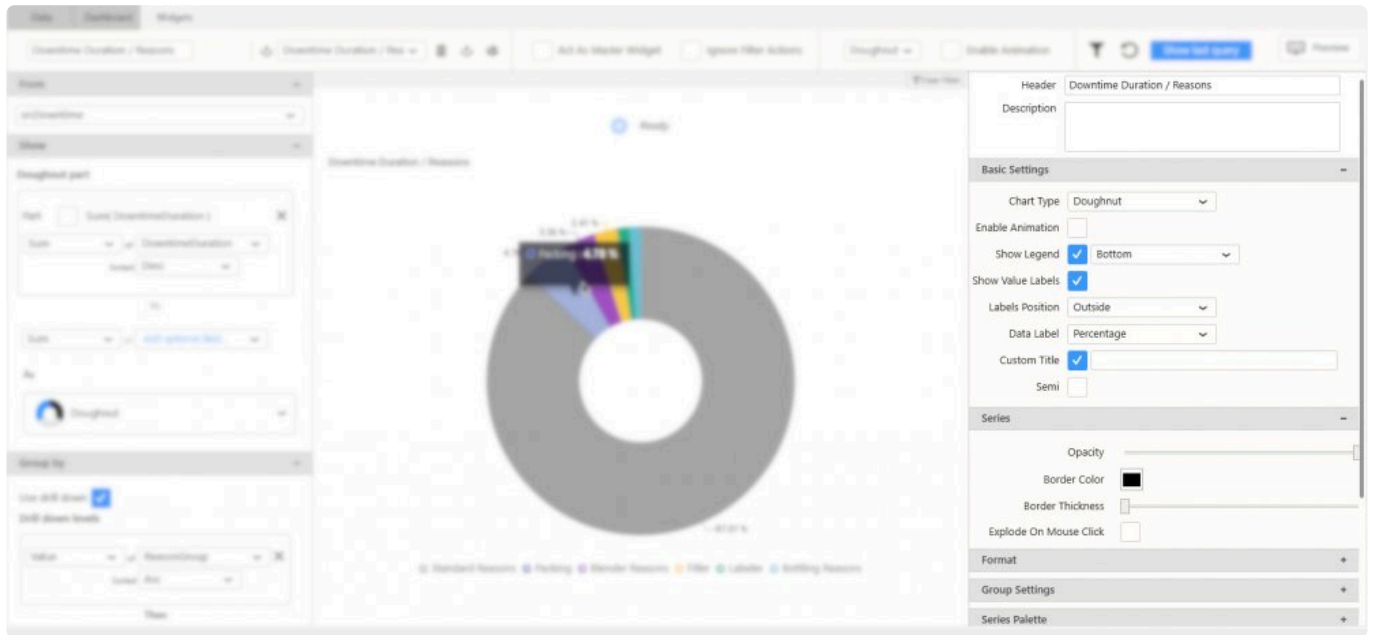
However, some Properties are common to several Widget types and listed in the current Chapter.

Last modified: 2019/04/30

2.6.6.1. Editing Widget Properties

Accessing the properties

Starting with Alpana v3.0.0, almost all Widget properties configuration happens in the properties panel on the right of the Widgets activity :



However, some properties are also located in the top bar for quick access.

Folding / Unfolding categories

Different parts of the properties configuration interface are grouped into categories.

For better readability, each category can be :

... folded ...



... and un-folded :



Last modified: 2019/04/30

2.6.6.2. Name, Header and Description

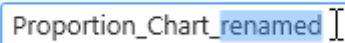
Widget Name

Each Widget has an internal Name used only at design-time, for example in the Widgets list.

This Name is not visible to the end user when the Widget is published.

It allows to have a clear useful name for the developer while the end user will want to see a different custom header, or even no header.

The Widget Name can be configured from the left of the top bar :

A text input field with a blue border containing the text "Proportion_Chart_renamed" and a cursor at the end.

Header

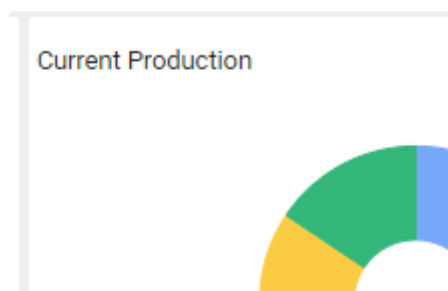
Widgets can also have a Header visible to the end user.

This can be used to display a title that can be a short description of the Widget's purpose.

The Header can be configured from the top of the properties panel :

A text input field with a blue border containing the text "Current Production" and a cursor at the end. To the left of the field is the label "Header" and to the right is a green circular refresh icon.

At runtime, this will be displayed at the top of the Widget container :



Header options

Some Header options are available in the *Container appearance* section :

Container Appearance

Show Border ☐

Override Border Color ☐

Border Thickness

Rounded Corners ☐

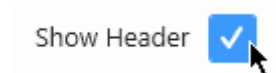
Show Header ☒

Header Alignment

Override Header Color ☐

Show / Hide Header

The *Show Header* option (check by default) allows to display the Header at runtime :

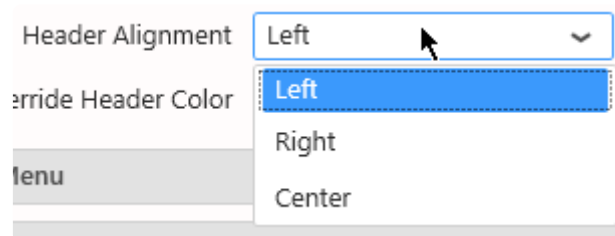


When the option is un-checked, the Header is invisible at runtime.

Header Alignment

Allows to align the Header :

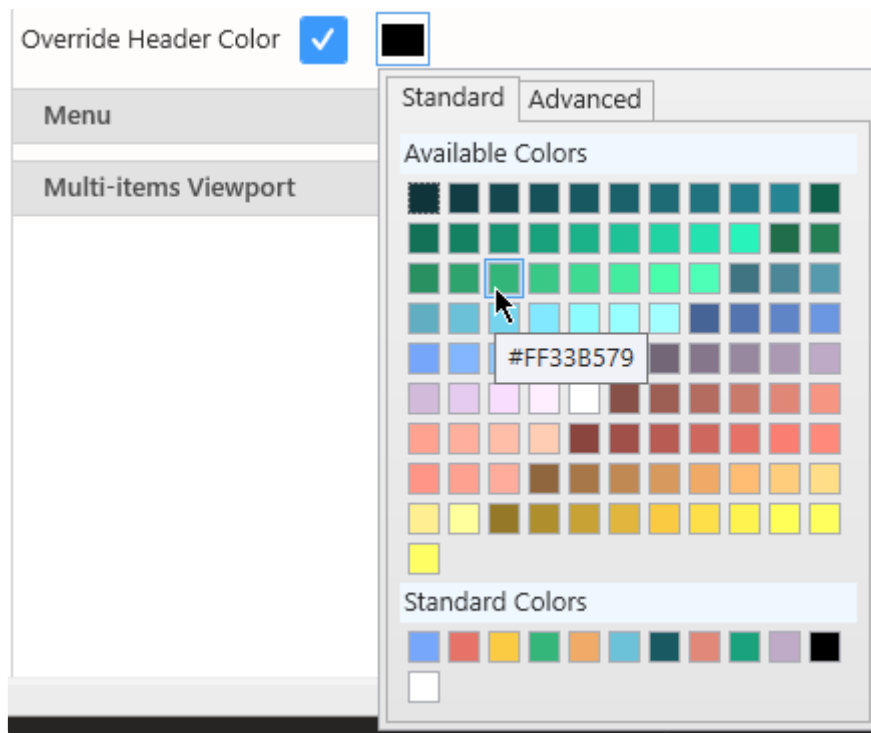
- Left
- Right
- Center



Header Color

By default, Header Color is managed by the currently applied Theme.

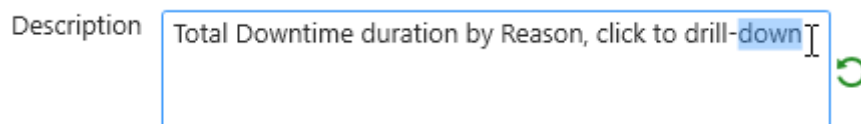
However, you can override the Theme color for the Header by checking *Override Header Color* and selecting the desired color :



Description

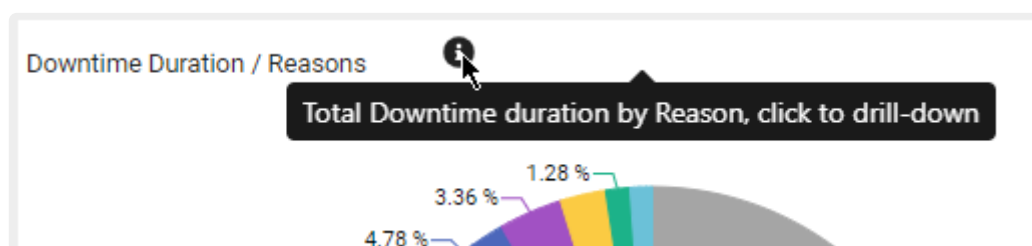
The Description can be used to display a longer text describing the Widget's purpose, its data, how to use it, ...

The Description can be configured from the top of the properties panel :



At runtime, the Description is hidden and a **i** icon appears near the Widget title to indicate the presence of a Description :

Hovering the **i** icon shows the description :



Last modified: 2019/04/30

2.6.6.3. Container Appearance

Container Appearance

These properties rule the appearance of the *container* that contains the Widget within the Dashboard.

Example :

Container Appearance

Show Border

☒

Override Border Color

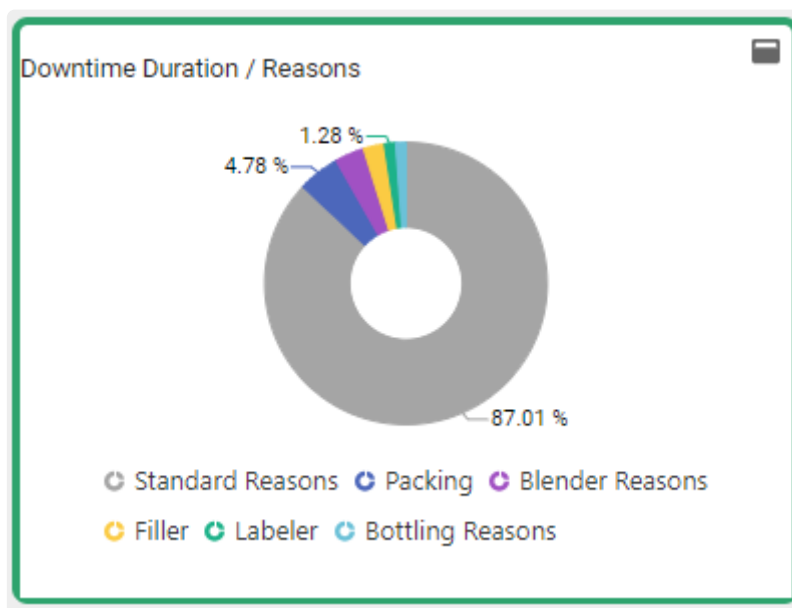
☒

Border Thickness

Rounded Corners

☒

results in :



Border

Show border

Show a border line around the Widget.

Border Color

By default, Border Color is managed by the currently applied Theme.

However, you can override the Theme color for the Border Color by checking *Override Border Color* and selecting the desired color.

Border Thickness

Define the thickness of the line defining the border color, in pixels.

Rounded Corners

Round the corners of the Widget container, to turn it into a rounded rectangle.

Header

Header settings : see [previous chapter about Header and Description](#).

Last modified: 2019/04/30

2.6.6.4. Widget Menu items

Menu

At runtime, the top of the Widget Container will include buttons to run some actions on the Widget.



The *Menu* section allows to configure those buttons.

Menu

Enable Comment	<input type="checkbox"/>
Show Maximize	<input checked="" type="checkbox"/>
CSV Export	<input checked="" type="checkbox"/>
Excel Export	<input checked="" type="checkbox"/>
Image Export	<input checked="" type="checkbox"/>
PDF Export	<input checked="" type="checkbox"/>
Show Pin Widget	<input checked="" type="checkbox"/>

* Note : Some of those properties are only visible when the Dashboard is published.

Enable Comment

Shows a control at runtime that allows to create comments at the Widget level.

Show Maximize

Shows a control at runtime that lets the user maximize the Widget full page.

Exports

Each options will allow at runtime the user to export the Widget in the selected format(s).

If none is selected, the *Export* button disappears at runtime.

- CSV
- Excel
- image
- PDF

Pin Widget

Shows a control at runtime that lets the user pin the Widget to a Homepage.

Last modified: 2019/04/30

2.6.6.5. Measure Formatting and Date Formatting

Introduction

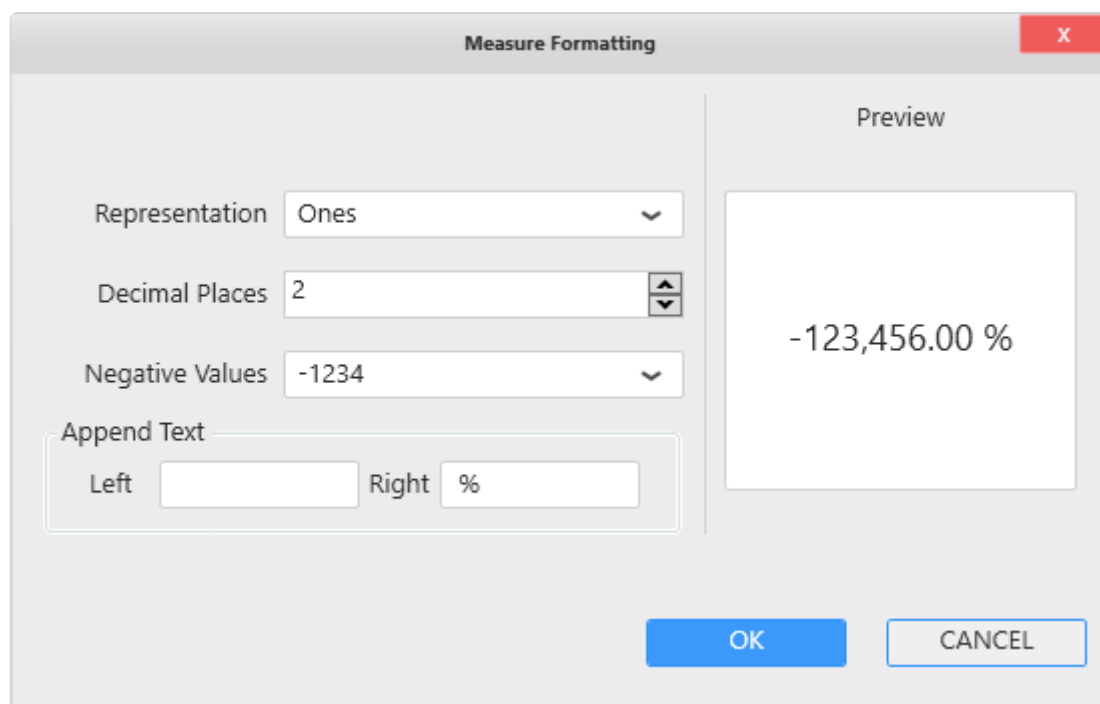
Most Widgets show data labels for values and dates, which may depend on user culture and personal choices.

The properties panel has a “*Format*” section which allows to configure these labels. All number and date fields are listed there, and labels can be configured by clicking on the cog icon :



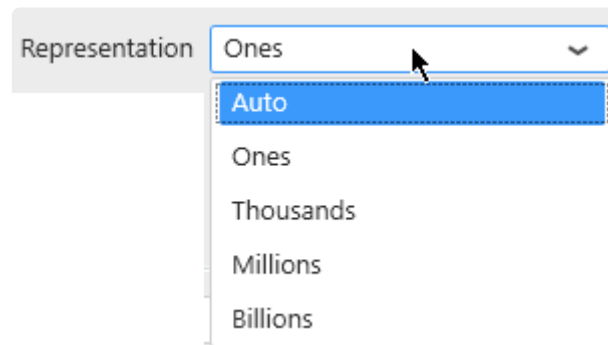
Measure Formatting

The Measure Formatting dialog allows to configure the label formatting for number fields :



Number Representation

Numbers can be shortened in several ways :



- Auto (default) : depending on how large the number is, a multiplier symbol can be added : “K” for thousand, “M” for million, “B” for billion
- Ones : the full number is displayed without multiplier symbol
- Thousands
- Millions
- Billions

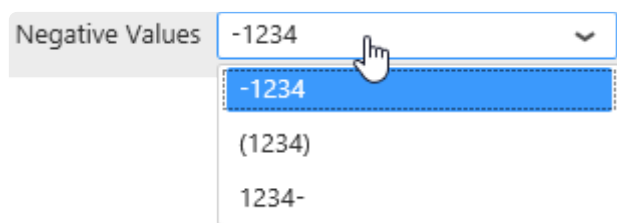
Decimal Places

The Decimal Places selector allows to force how many decimal places should be displayed for numbers :



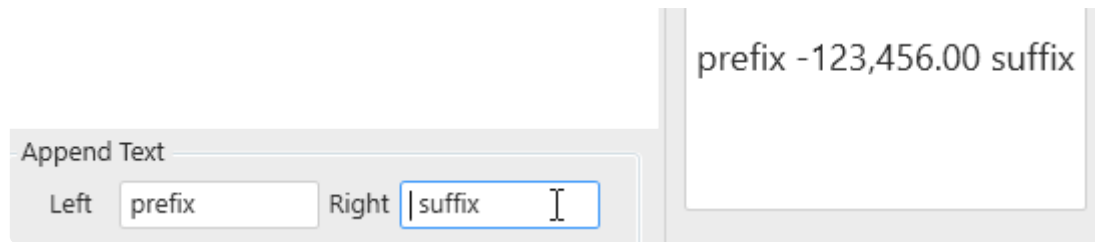
Negative Values

Alternate symbols can be used to represent negative values :



Prefix / Suffix

Prefix (before the number) and/or Suffix (after the number) text can be used :



Append Text

Left Right

prefix -123,456.00 suffix

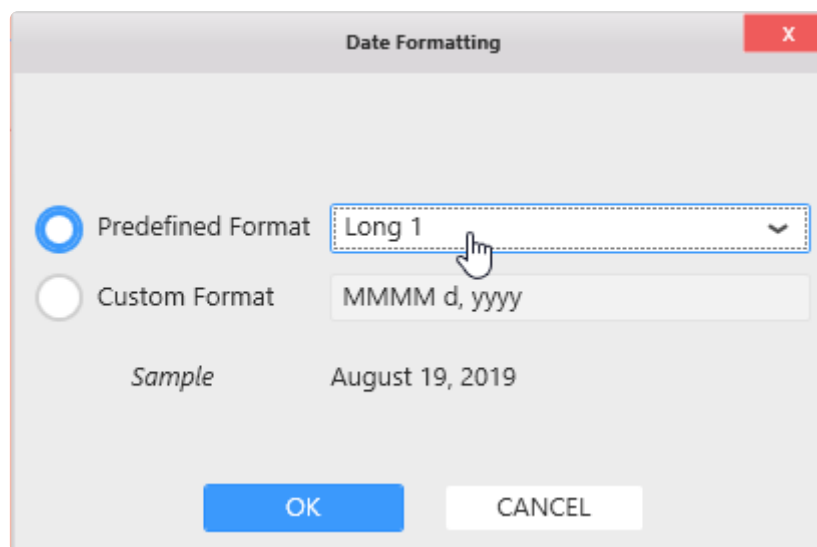
* Hint : This is useful for example for showing units of measurement.

No space is added by default between the number and the prefix/suffix.

Date Formatting

The Date Formatting dialog allows to configure the label formatting for `Date` or `DateTime` fields :

Pre-defined formats



Date Formatting

☒ Predefined Format

☐ Custom Format

Sample August 19, 2019

OK CANCEL

A selection of date format types is available, and a preview is shown at the bottom of the window as "Sample".

Date format localization

Only the *type* of date format can be configured (short, long, etc...).

At runtime, the date localization will depend on localization choices of the user.

In order of decreasing priority :

1. The Language selected by the user on **Alpana Server** contains culture information that will be

used to localize the dates. Note : this is not available during Dashboard Preview from Alpana Designer.

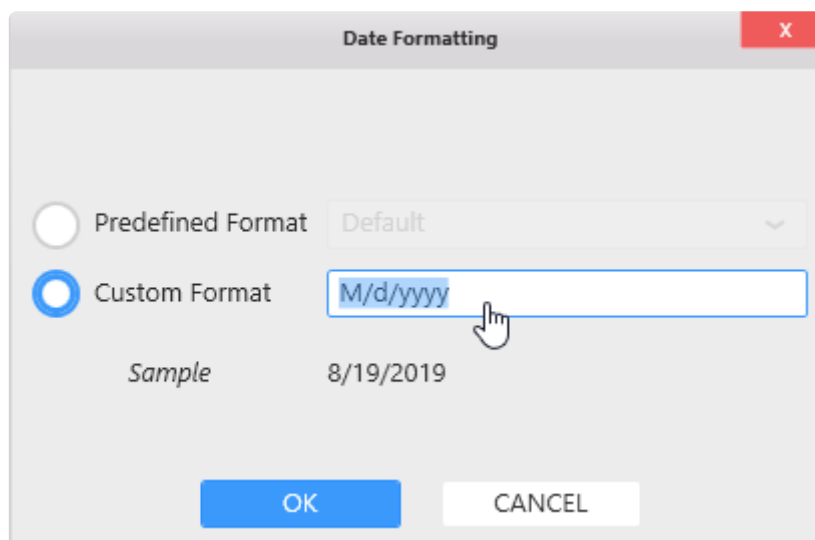
2. The culture selected by the user as preferred language in the **web browser** will be used.
3. The culture of the display language of the **web browser** will be used.



In the Widget Preview, no locale is applied, and only the default EN-US English locale is applied.

Custom formats

Custom formats can also be configured, and a preview is shown at the bottom of the window as “Sample”. :



This allows to use text placeholders to define the custom date string.

There are 2 ways to configure a date format string :

- from a standard date format
- compose a custom string

Standard date formats

If the custom string is a single character, then you can use with one of the placeholders defined [here](#) as part of the .NET framework, and you will end up with one of the standard date formats that will adapt to your culture :

- d : Short date pattern

- D : Long date pattern
- f : Full date/time pattern (short time)
- F : Full date/time pattern (long time)
- g : General date/time pattern (short time)
- G : General date/time pattern (long time)
- M, m : Month/day pattern
- O, o : Round-trip date/time pattern
- R, r : RFC1123 pattern
- s : Sortable date/time pattern.
- t : Short time pattern
- T : Long time pattern
- u : Universal sortable date/time pattern
- U : Universal full date/time pattern
- Y, y : Year month pattern

Custom date strings

This allows to configure any date part in any order and use that as a label.

Use any text and include special placeholders to compose the date format.

The placeholders are defined [here](#) as part of the .NET framework, and a summary is below :

- d, dd : the day of the month
- ddd, dddd : the day of the week
- f, ff, fff, ffff, fffff, fffffff, ffffffff, ffffffff : decimals of the second
- h, hh : the hour, using a 12-hour clock
- H, HH : the hour, using a 24-hour clock
- m, mm : the minute
- M, MM, MMM, MMMM : the month
- s, ss : the second
- t, tt : AM/PM
- y, yy, yyyy : the year
- : : the time separator
- / : the date separator
- \ : the escape character allows to write a character without interpreting as a placeholder
- "my text", 'my text' : the escape quotes allow to write a text without interpreting as a placeholder

Last modified: 2019/08/30

2.6.6.6. Multi-items Viewport

Introduction

Some Widgets can be turned into a container for many Widgets.

These Widgets are displayed in a grid showing a certain number of Widgets horizontally (Columns) and vertically (Rows).

These many Widgets are generated automatically based on a data field, and their number may not be known in advance.

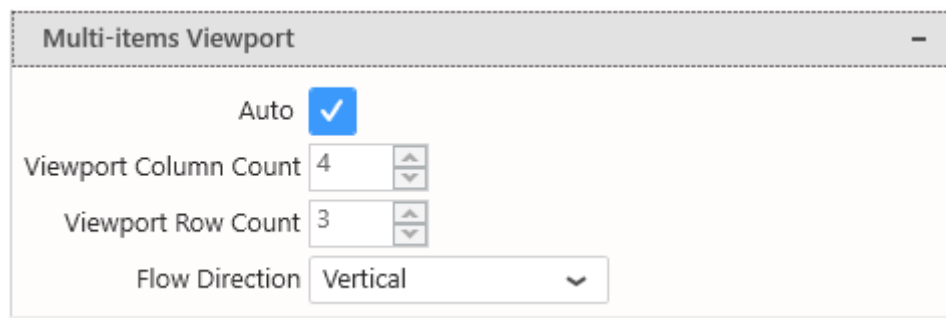
If there are too many Widgets contained, Alpana allows to view only some of them, and then use a scrollbar to see the others.

The properties sections “*Multi-items Viewport*” allows to configure how many Widgets can be displayed at the same time inside the Widget container.

Setting Multi-items Viewport

From the Widget properties

In the Widget properties, the section *Multi-items Viewport* allows to configure this setting :

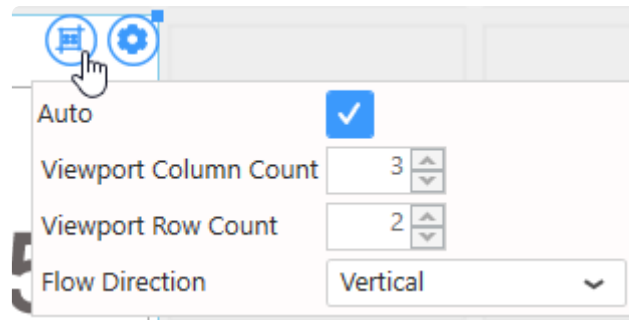


The screenshot shows a settings panel titled "Multi-items Viewport". It contains the following controls:

- An "Auto" checkbox that is checked with a blue checkmark.
- A "Viewport Column Count" spinner box set to the value 4.
- A "Viewport Row Count" spinner box set to the value 3.
- A "Flow Direction" dropdown menu currently set to "Vertical".

From the Dashboard layout

In the *Dashboard* activity, each Widget has a shortcut to this setting :



Auto Mode

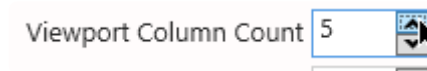
By default, *Auto* is checked and the Viewport is calculated automatically.

Manual Mode

By un-checking *Auto*, you can configure manually the Viewport.

Columns


Viewport Column Count allows to configure how many Widgets are displayed horizontally (the number of columns in the grid) :



Blender_001 5,125	Blender_003 1,521	Filling_002 4,776	Labeling_002 2,256	PackagingStation_002 7,296
Blender_002 2,725	Filling_001 6,360	Labeling_001 3,285	PackagingStation_001 7,244	Palletizing_001 4,872

Rows

Viewport Row Count allows to configure how many Widgets are displayed vertically (the number of rows in the grid) :

Viewport Row Count 

Blender_001 5,125	Labeling_001 3,285
Blender_002 2,725	Labeling_002 2,256
Blender_003 1,521	PackagingStation_001 7,244
Filling_001 6,360	PackagingStation_002 7,296
Filling_002 4,776	Palletizing_001 4,872

Flow direction

Flow direction allows to configure how Widgets are placed in the grid :

- *Vertical* : Widgets are placed from top to bottom, and then left to right. If there are too many Widgets to display, the next pages are on the right.
- *Horizontal* : Widgets are placed from left to right, then top to bottom (english reading direction). If there are too many Widgets to display, the next pages are at the bottom.

Flow Direction Horizontal

Blender_001 5,125	Blender_002 2,725
Blender_003 1,521	Filling_001 6,360
Filling_002 4,776	Labeling_001 3,285
Labeling_002 2,256	PackagingStation_001 7,244
PackagingStation_002 7,296	Palletizing_001 4,872

Last modified: 2019/04/30

2.6.7. Interactions

At runtime, users can click on widgets in order to perform an action depending on what data item is clicked.

In Alpana, three types of actions can be defined on widgets :

- Drill-Down (for the Widget types that support it)
- Link To URL
- Filter

When several of these actions are configured, only one is performed.

The order of precedence is the order in the list above.

Last modified: 2019/04/30

2.6.7.1. Drill-Down

Some Widgets define a Drill-Down action : when the user clicks on a data item, the Widget shows the underlying data as if the user were digging into it :

- all types of Cartesian Charts
- all types of Proportion Charts
- TreeMap

To learn to configure Drill-Down, see for each Widget in the [Widget Reference](#).

Last modified: 2019/04/30

2.6.7.2. Link To URL

Introduction

Most Widgets can be configured as a link to a URL : when the user clicks on the Widget, the URL is opened.

It is possible to configure the URL to be generated dynamically : when the user clicks on different parts of the Widget, different URLs are open.

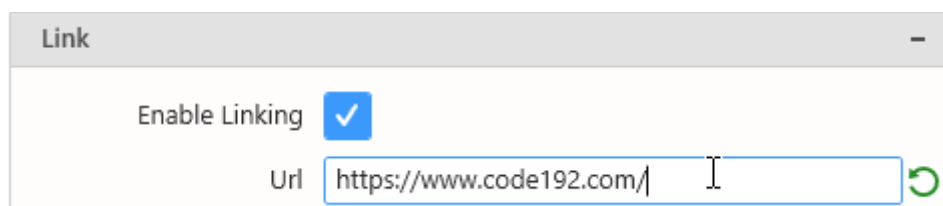
Use cases

This can be used to :

- open another Dashboard with different context (different URL Parameters values) depending on where the user clicked.
- open a different Dashboard depending on where the user clicked
- open a different website depending on where the user clicked
- open another website (for example a reporting solution) with different URL Parameters depending on where the user clicked

Configuring a Link To URL

The Properties panel has a “Link” section. Check the “Enable Linking” checkbox and write the desired destination URL :



With this configuration, when clicking the Widget, the same URL will always be opened.

Syntax for dynamically generated URLs

In the URL string, the special syntaxes below will be replaced by a value dynamically depending on the context on the widget at runtime.

Un-formatted field value

```
{{!MyFieldName}}
```

will be replaced by the corresponding field value depending on where the user clicked on the widget at runtime.

If the field is formatted (for example, a *MyDate* field formatted as *Month Year(MyDate)*), then the replaced value will be the raw value (un-formatted).

For example : 201702



This is the value that can be used as input to a [URL Parameter](#).

Example

If the Widget category field is *Month Year(MyDate)*, then you can send its value to another Dashboard with :

```
https://aveva.alpana.io/viewer/dashboard/MyDashboard?MyTargetParameter={{!MyDate}}
```

At runtime, when the user clicks on the part of the Widget where the value corresponds to *February 2017*, then the link opened will be :

```
https://aveva.alpana.io/viewer/dashboard/MyDashboard?MyTargetParameter=201702
```

Formatted field value

```
{{:MyFieldName}}
```

will be replaced by the corresponding field value depending on where the user clicked on the widget at runtime.

If the field is formatted (for example, a *MyDate* field formatted as *Month Year(MyDate)*), then the replaced value will contain the format.

For example : February 2017

This is a value that can be used for display.

Example

If the Widget category field is *Month (MyDate)*, then you can send its value to Wikipedia with :

```
https://en.wikipedia.org/wiki/{{:MyDate}}
```

At runtime, when the user clicks on the part of the Widget where the value corresponds to *February*, then the link opened will be :

```
https://en.wikipedia.org/wiki/February
```

Parameter value

```
${MyParameterName}
```

will be replaced by the current value of the [Parameter](#).

If the Parameter contains multiple values, they are separated by commas ,

Example

If your dashboard contains a Parameter called `MySourceParameter`, then you can send its value to another Dashboard with :

```
https://aveva.alpana.io/viewer/dashboard/MyDashboard?MyTargetParameter=${MySourceParameter}
```

At runtime, when the user clicks the Widget, if the value of `MySourceParameter` is currently `Buildin g_A`, then the link opened will be :

```
https://aveva.alpana.io/viewer/dashboard/MyDashboard?MyTargetParameter=Buildin g_A
```

Examples and tips

Dynamic e-mail creation

In this example, you have a Grid displaying current maintenance issues. You want to let users click an item in the Grid to open an e-mail to support.

The solution is to use `mailto:` links, with the following syntax :

```
mailto:someone@yoursite.com?cc=someoneelse@theirsite.com, another@thatsite.com, me@mysite.com&bcc=lastperson@theirsite.com&subject=Need%20help%20with%20...&body=Hello...
```


Except the recipient, all parts are optional.

Of course, with Alpana you can now generate a link like this dynamically from fields in your data !

In the Grid Widget, those fields can even be hidden.

Example :

```
mailto:{{:support_email}}?subject=Issue%20with%20{{:equipment}}&body={{:Downtime_cause}}%20is%20happening
```

Opening PDF at the desired page

In this example, you have a Grid displaying current maintenance issues.

You want to let users click an item in the Grid to open the correct maintenance document opened at the correct page.

On hosted PDF documents, the syntax is :

```
http://your-site.com/your-document-location.pdf#page=123
```

Of course, with alpana you can now generate a link like this dynamically from fields in your data !

In the Grid Widget, those fields can even be hidden.

Example :

```
http://your-site.com/{{:manual_name}}.pdf#page={{:manual_page}}
```

Last modified: 2019/05/29

2.6.7.3. Master Filter

Widgets can be configured as Filter : when a data item in the widget is clicked, other Widgets only show the corresponding data.

This behaviour is closely related to the notion of Dashboard Parameters and is documented in [the corresponding Chapter](#).

Last modified: 2019/04/30

2.7. Widget Properties Reference

Each Widget type has specific properties and configuration options.

This chapter is a reference of all Widget properties.

A Widget is a visualization entity that uses data from a single Data Source and displays it in an interactive way.

Widgets are assembled to create a Dashboard.



To learn about Widgets and their generic settings, see the corresponding generic chapter : [Configuring a Widget](#).

Last modified: 2019/05/24

2.7.1. Cartesian Chart

Cartesian Charts allow to view data on a 2D plane using perpendicular X-Y axes.

They are split into several types, but any type of chart can be changed to any other type of chart, and almost all types can be combined together.

Last modified: 2019/05/24

2.7.1.1. Binding Data

Introduction

The following fields are available for configuring data binding on a Cartesian Chart :

Show

Y Value(s)

Value Sum(price)

Line Field

Sum

of

price

Sorted

by default

Vs

Sum

of

Add optional field...

As

Line

Group by

Use drill down

X Axis

Value

of

Date

Formatted as

Datetime

Sorted

ASC

Split

Value

of

Product

Sorted

ASC

Split into

many charts

Value

of

Building

Sorted

ASC

Split into

the same chart

Page 303 of 761

- *Y Value(s)* : the height of the chart line along the Y Axis
- *X Axis* : how we group values together on the X Axis to create points on the line
- *Split into the same chart* : how we split each series into different lines on the same chart
- *Split into many charts* : how we split series of different charts in the same container

Show : *Y Value(s)*

This represents the *quantity* we want to display on the Chart, and will bind for example to the height of the line (Line chart) or the height of the column (Column chart).



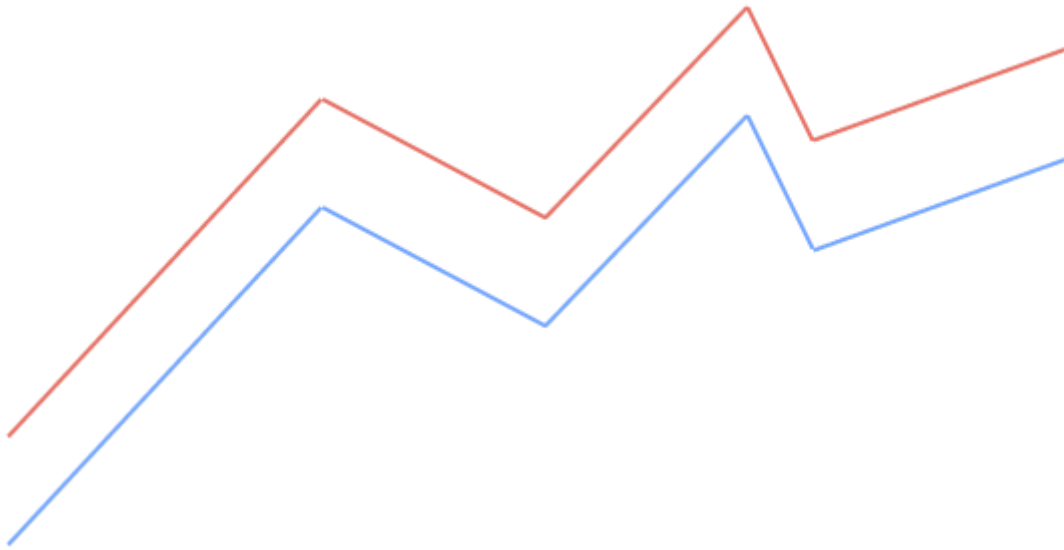
These fields **must be aggregated** : you must choose an aggregation function (SUM, AVG, ...)
At least one field must be bound.



Some Chart Types require more than one field to define a point.
For example the Range Area requires to define one field for each side of the area : Low and High.

Multiple fields : multiple Series

When binding multiple fields at the same time, each field will generate a separate Series.



This way, each Series can have its own graphical properties, its own Y axis, etc... and even different Chart Types.

To change the order of fields, see [the corresponding chapter](#).



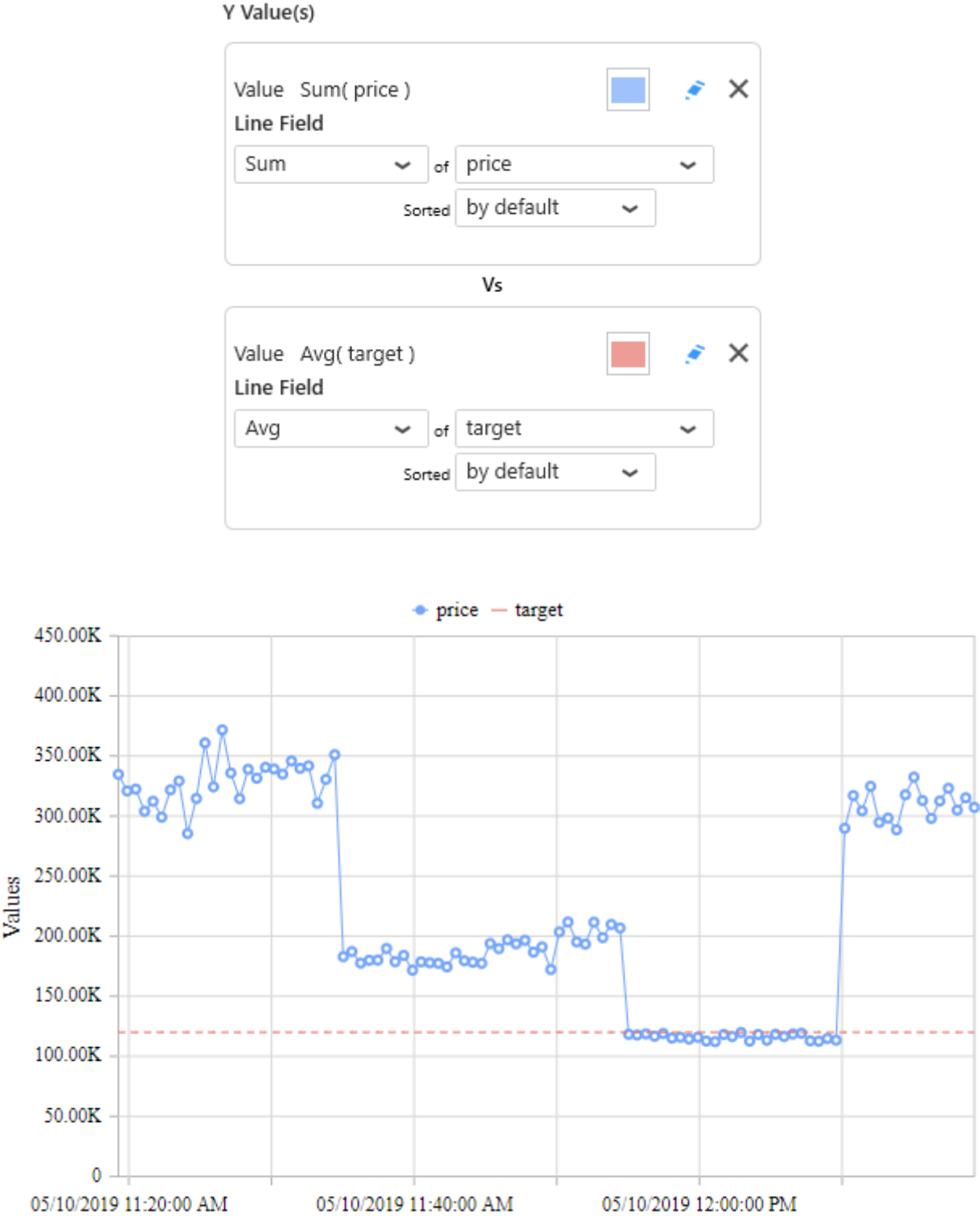
Note :

This only works if your series are already into separate data fields. For example, with a [Historian Wide Query](#).

If you want to create series based on different values/names from a single data field, see below the “*Split into the same chart*” configuration.

Example

With a field called “Price” and a field called “Target”, you will chart 2 separate series, one for each field.



Group by : X Axis

This represents the *category* we want to use to separate data points.

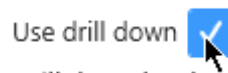


These fields **cannot be aggregated** : they must have several separate values in order to group the “Values” aggregation.

At least one field must be bound.

Multiple X Axis fields : Drill-Down

To enable Drill-Down, you can check the box *Use drill down* :



This allows to bind any number of fields to drill-down levels, in the same order as they are in the list. To change the order of fields, see [the corresponding chapter](#).

Example

With fields called “continent”, “country”, and “Date”, the following configuration will configure the corresponding drill-down :

Group by

Use drill down

Drill down levels

Value

of

continent

X

Sorted

Asc

Then

Value

of

Country

X

Sorted

Asc

Then

Value

of

Date

X

Formatted as

Year

Sorted

Asc

Then

Value

of

Date

X

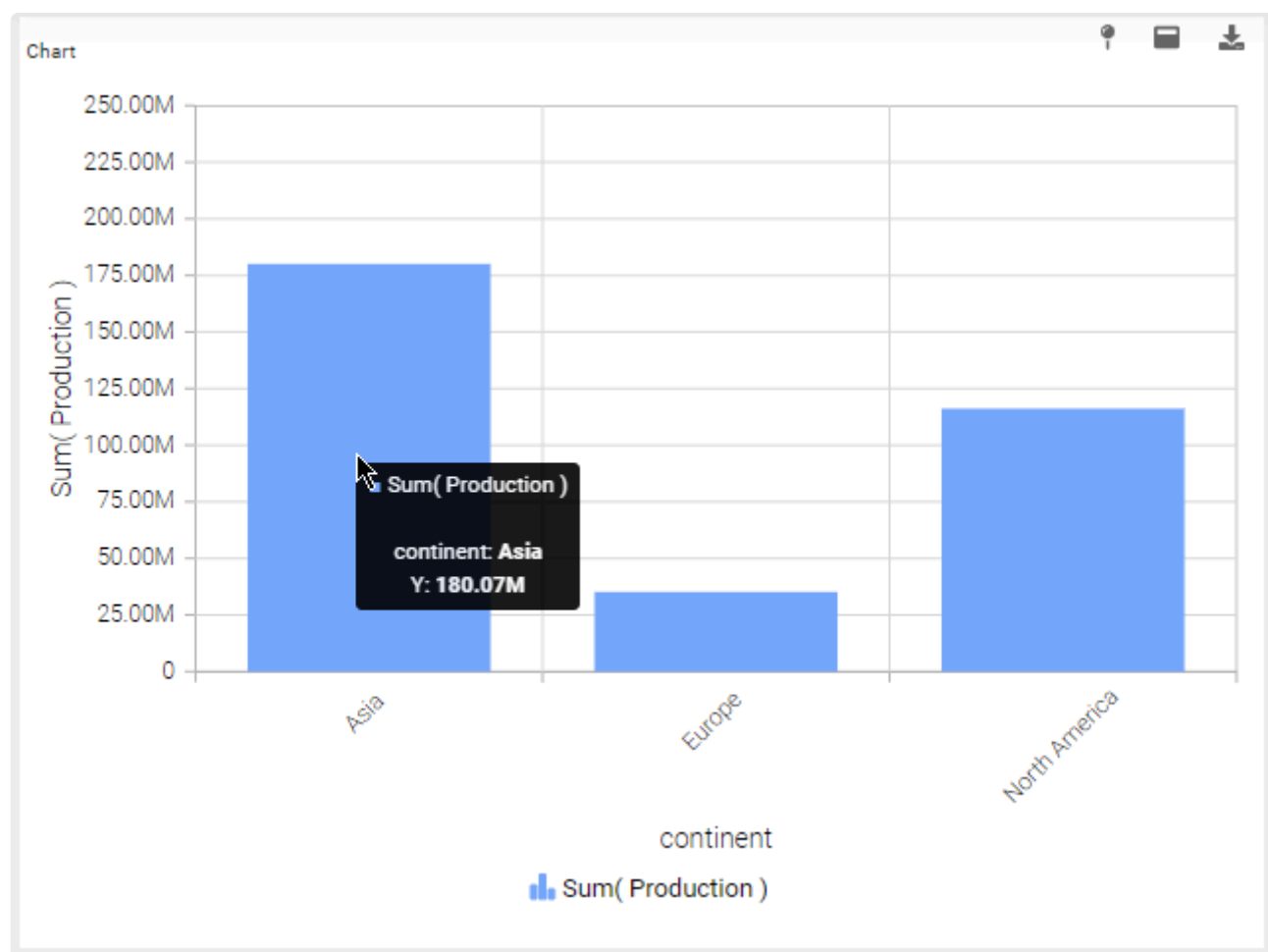
Formatted as

Month

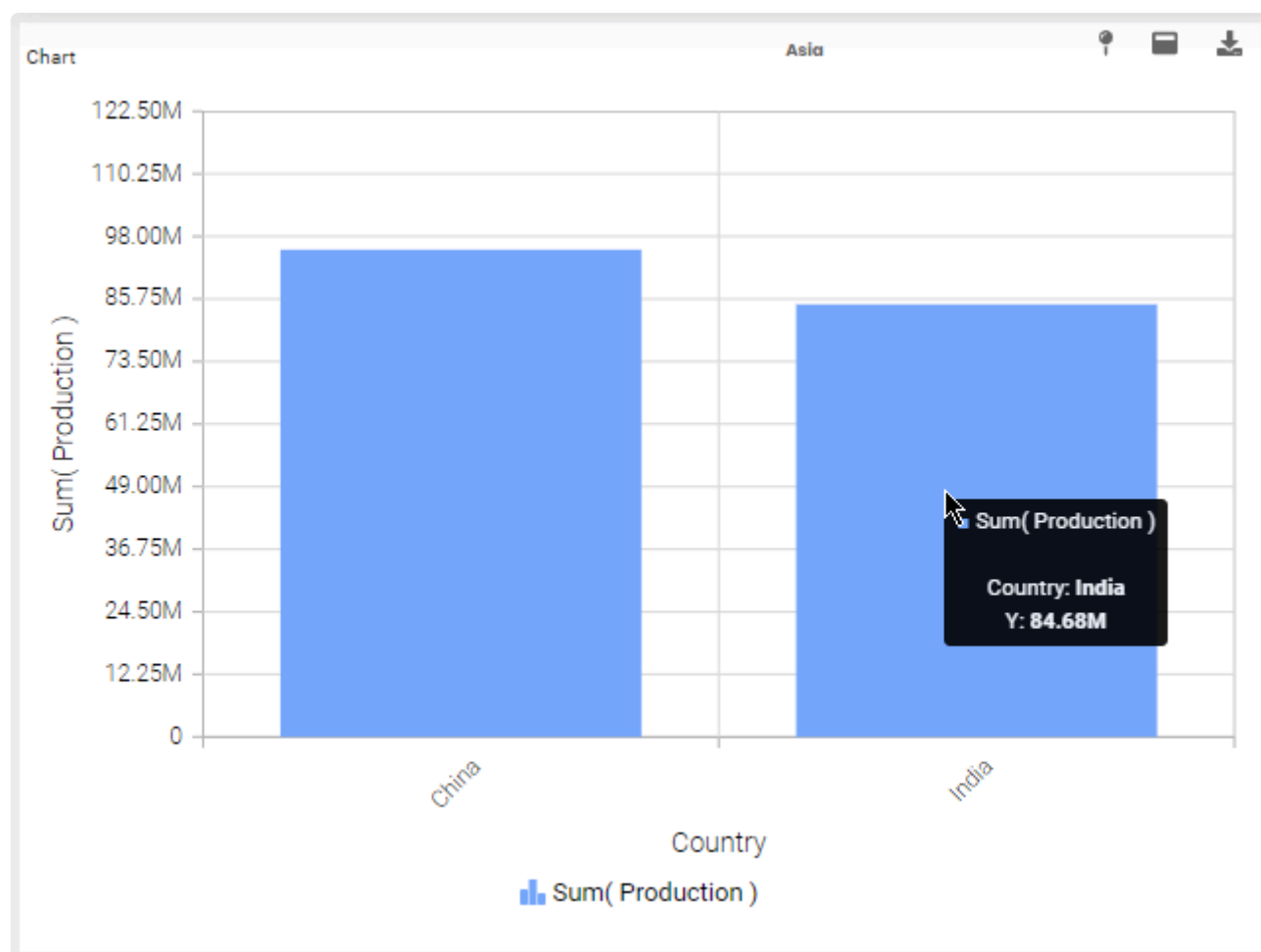
Sorted

Asc

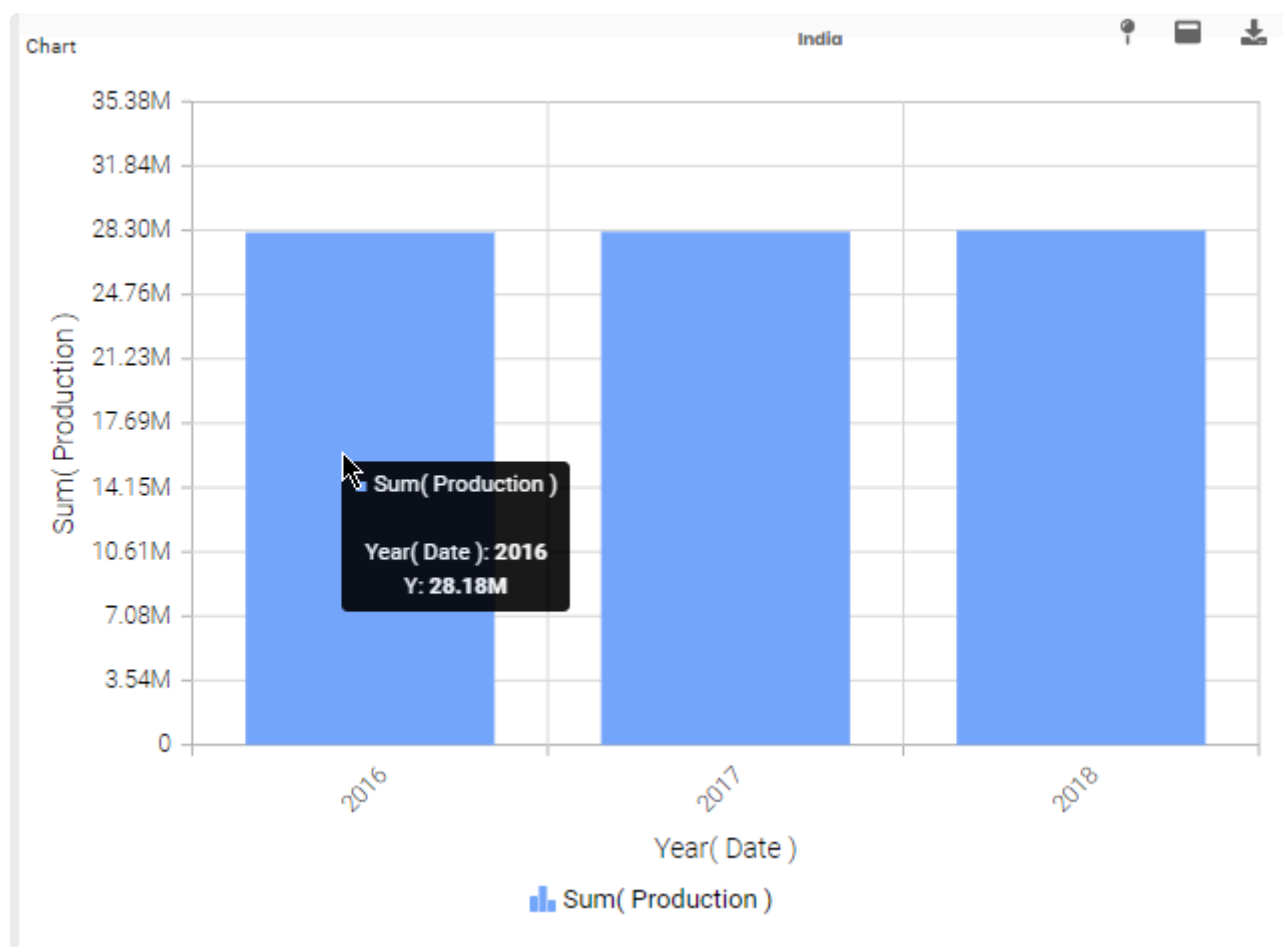
First, the data will be grouped by “continent” :



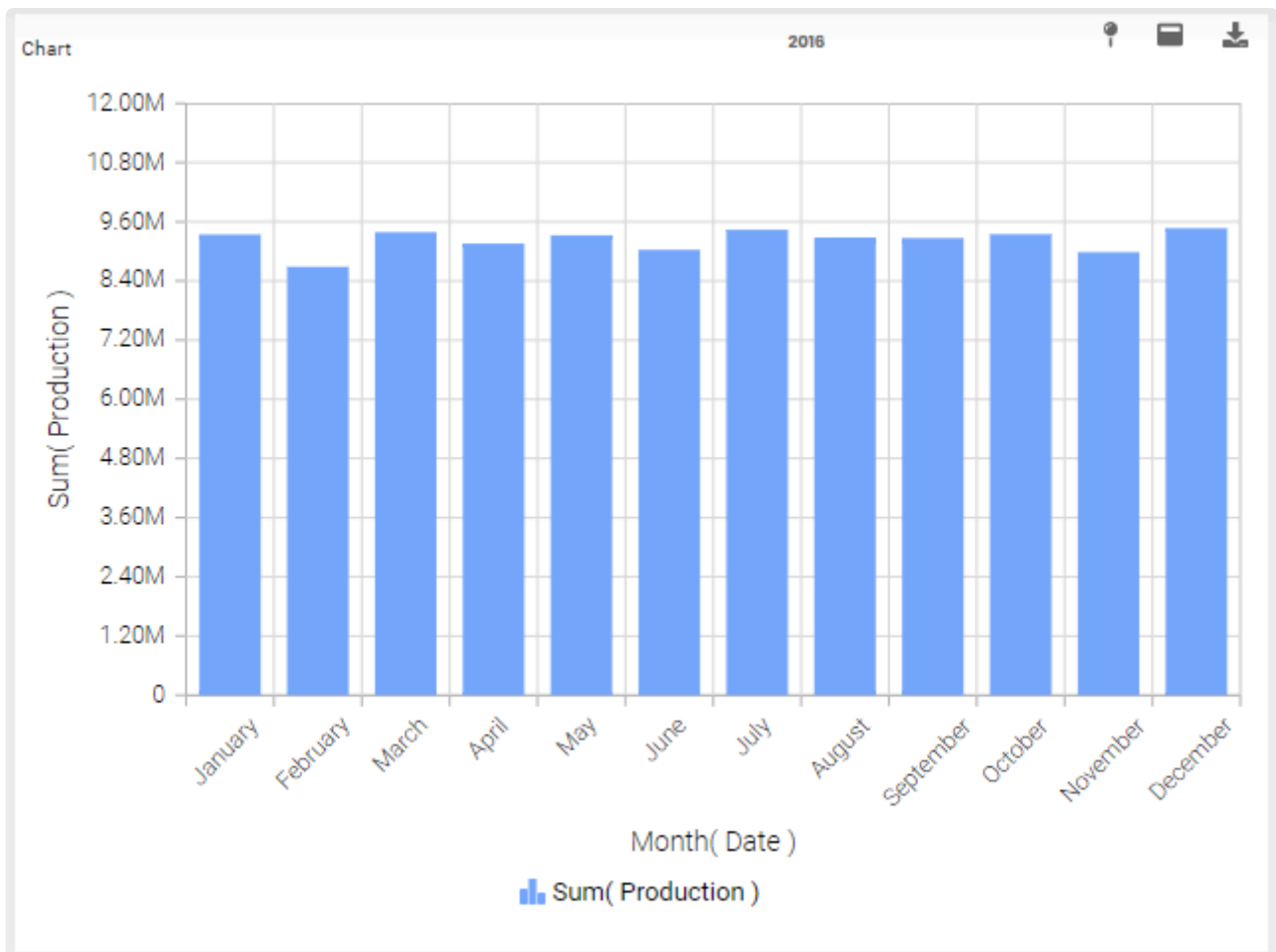
When the user clicks a continent (example : Asia), the Chart is drilled-down to the next level : the Countries of Asia :



Next, when the user clicks a Country (example : India), the Chart is drilled-down to the next level : the Years of India :



Next, when the user clicks a year (example : 2016), the Chart is drilled-down to the last level : the Months of 2016 of India :



Meanwhile, the Drill-Down navigation at the top shows the value of the current level. When clicking it, you can drill back up to any level, or directly to the top by selecting *Close* :



Group by : *Split into the same chart*

This represents the series field to generate a different chart series for each different value in this *Split into the same chart* field.



This field is optional.

It can be combined with *Split into many charts*.



Note :

This only works if your series are values into a single data field. For example, with a [Historian Narrow Query](#).

If you want to create series based on separate data fields, see above the “multiple Series” configuration.

Example

If you have Historian Narrow Query with a field called “TagName”, you may want to plot a separate Line chart for each value of TagName :

Show

Y Value(s)

Value Value

Line Field

Sum of Value

Sorted by default

Vs

Sum of Add optional field...

As

Line

Group by

Use drill down

X Axis

Value of Date

Formatted as Datetime

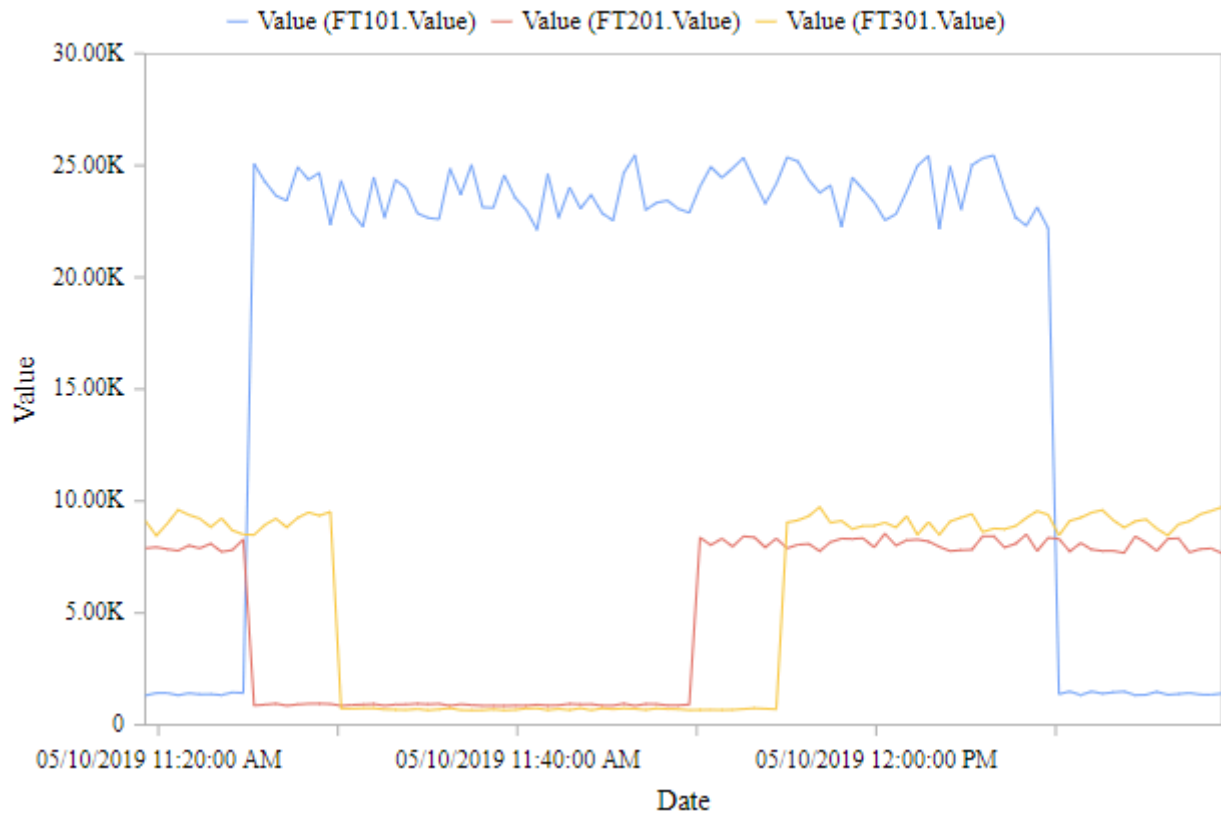
Sorted Asc

Split

Value of TagName

Sorted Asc

Split into the same chart



Group by : *Split into many charts*

This represents the series field to generate a different contained Chart for each different value in this *Split into many charts* field.

This will turn the widget into a container, and generate a complete different chart for each value in the series field, contained in the Widget.

The layout of the charts inside the container is configurable.

Please see [the corresponding chapter to configure the view of the Widget container](#).

This field is optional.

It can be combined with *Split into the same chart*.

✿ Note : This only works if your series are values into a single data field.

Example

If you have a field called “continent”, you may want to plot a completely separate chart for each value of continent :

Show

Y Value(s)

Value Value

Line Field

Sum of Value

Sorted by default

Vs

Sum of [Add optional field...](#)

As

Line

Group by

Use drill down

X Axis

Value of Date

Formatted as Datetime

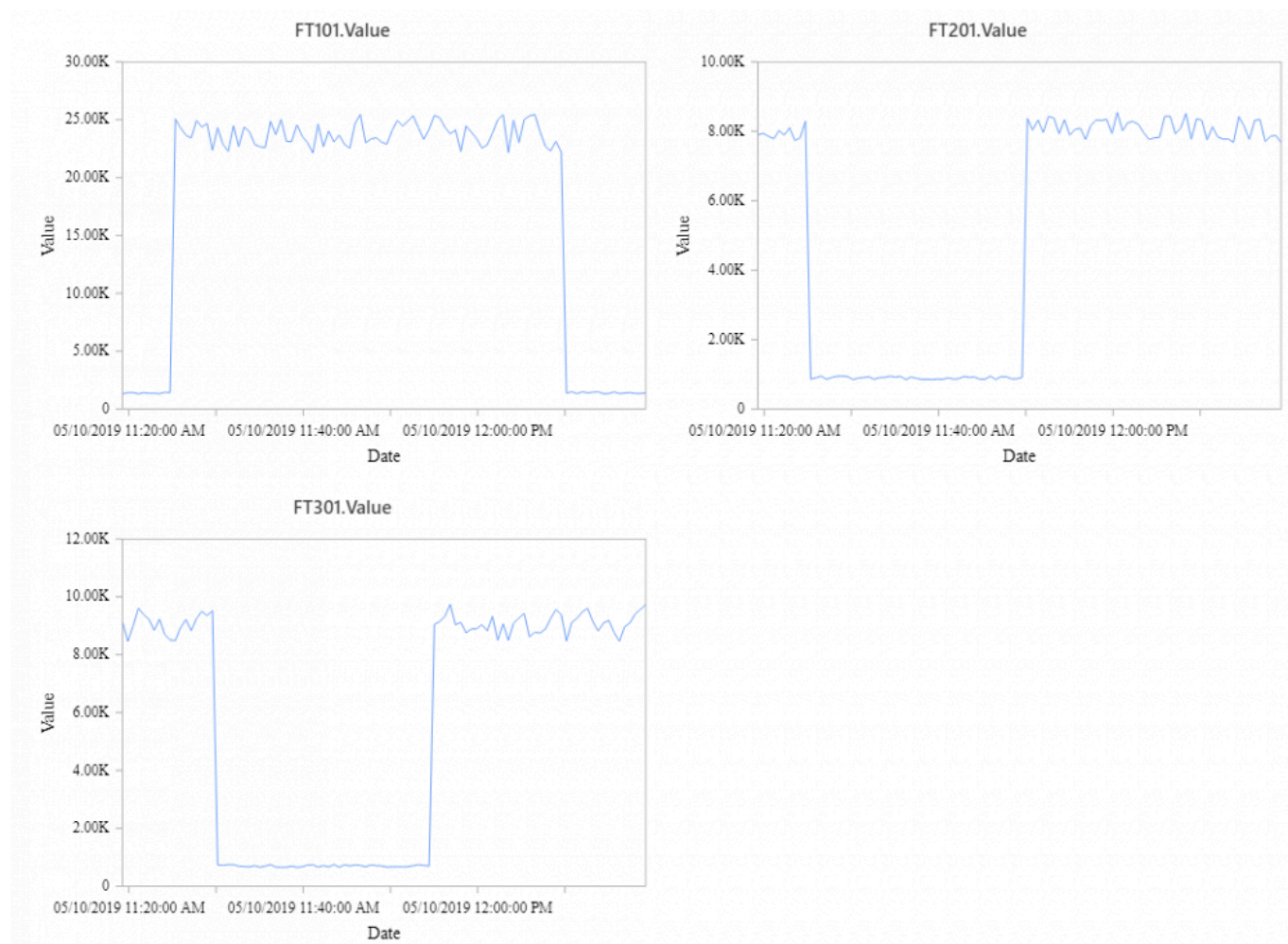
Sorted ASC

Split

Value of TagName

Sorted ASC

Split into many charts



Last modified: 2019/05/16

2.7.1.2. Chart Types

Categories

There are two main categories of Chart types :

- Horizontal X Axis : the grouping axis (X) is displayed from left to right
- Vertical X Axis : the grouping axis (Y) is displayed from bottom to top

Types

Cartesian Charts are split into several Widget types, listed below.










In addition, a special shortcut exists for combining different chart types :







Combo





Horizontal X Axis

1 Value Field



-  Column
-  Stacked Column
-  100% Stacked Column
-  Area
-  Stacked Area
-  100% Stacked Area
-  Line
-  Spline
-  Spline Area

-  Step Line
-  Step Area
-  Scatter
-  Waterfall

2 Value Fields


-  Bubble
-  Range Area
-  Range Column
-  Hi/Lo

4 Value Fields

-  Hi/Lo Open/Close
-  Candle

Vertical X Axis

1 Value Field

-  Bar
-  Stacked Bar
-  100% Stacked Bar

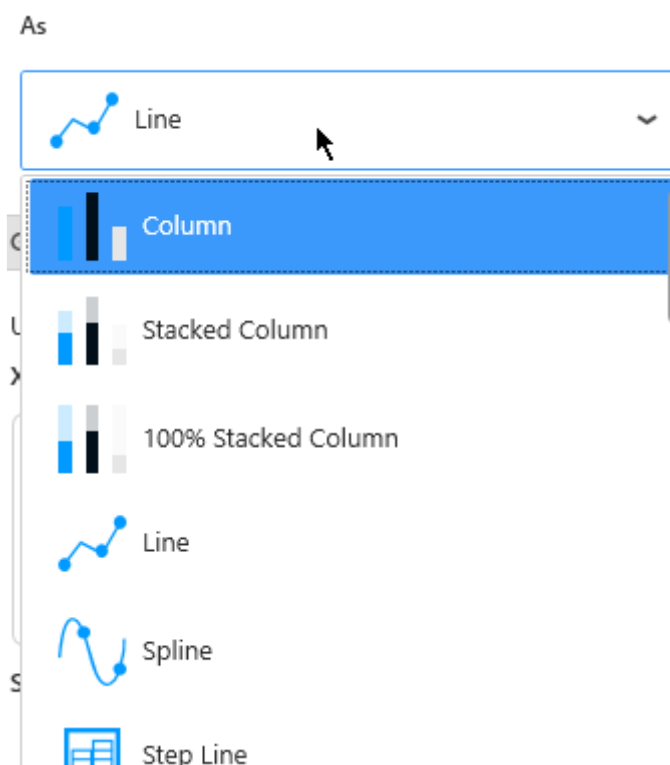
Last modified: 2019/05/16

2.7.1.3. Switching Chart Types

When adding a Widget, the Chart will have a single type globally : all series will have the same type.

Chart type can be changed on the fly while keeping the widget configuration.

To change the Chart type, use the selector under *As* in the data configuration :



Last modified: 2019/05/16

2.7.1.4. Combining Chart Types

Enabling Combo Chart

Any Chart can be turned into a Combo Chart.

This allows to combine multiple Chart Types into the same widget.


! *Horizontal X Axis* types of Charts cannot be combined with *Vertical X Axis* types.
Example : a Bar and a Column cannot be combined together since they don't have the same orientation of the X Axis.

Click *Enable Combo Chart* under *Series* :



The *Default Chart Type* can be selected under *Series* also.

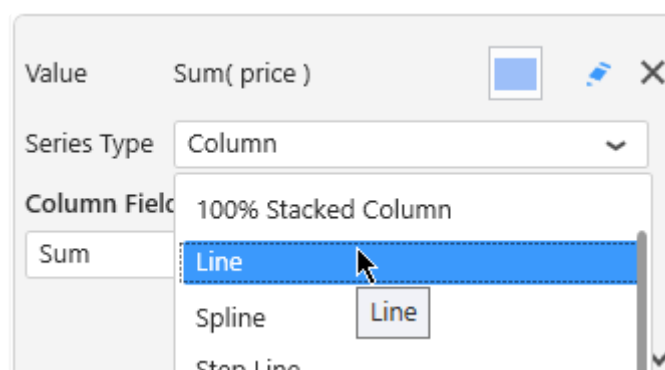
This will be the Chart type used by default when adding new *Y Value* Series to the data configuration.

In addition, a shortcut exists to create a Combo Chart immediately : the widget type  Combo.

Selecting Types

When Combo Chart is enabled, the Chart Type is no longer a global setting to the widget.

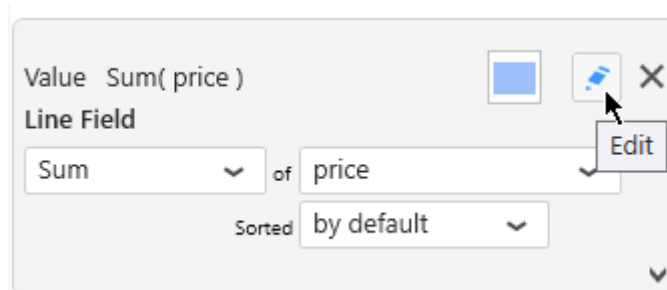
Instead, each *Y Value* Series has a new setting called *Series Type* :



Last modified: 2019/05/16

2.7.1.5. Series Settings

Each *Y Value* Series can be configured individually by clicking the pen icon next to its name :



This allows to fine-tune your chart by configuring exactly what you want on each series inside the widget.

* Keep in mind that some configuration options are not available for all chart types.

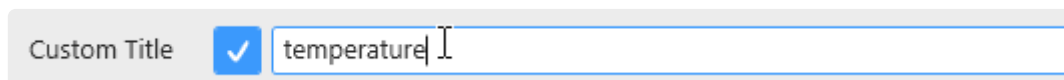
Last modified: 2019/05/24

2.7.1.5.1. Series Name

Each Series has a name displayed in various places.

By default, this name is automatically generated from the data field bound to the Series.

To configure this name, edit the Series Settings, in tab *Main*, check *Custom title* and write the desired name :



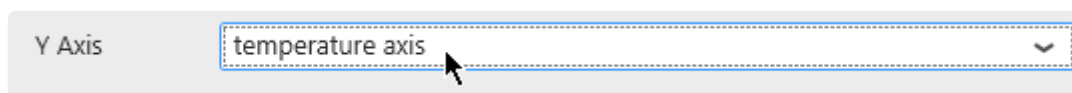
Custom Title ☒

Last modified: 2019/05/24

2.7.1.5.2. Axis binding

Each Series can be bound to a different Y Axis.

Edit the Series Settings, in tab *Main*, select the desired Y Axis:



Listed are all the available [Y axes](#).

Last modified: 2019/05/24

2.7.1.5.3. Data Point information

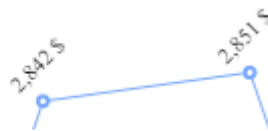
Value Labels

Text labels can be used to display the value next to each data point.

Edit the Series Settings, in tab *Advanced*, check *Value Labels* :

Value Labels	<input checked="" type="checkbox"/>
Value Labels Rotation	-45°

With the following result for example :



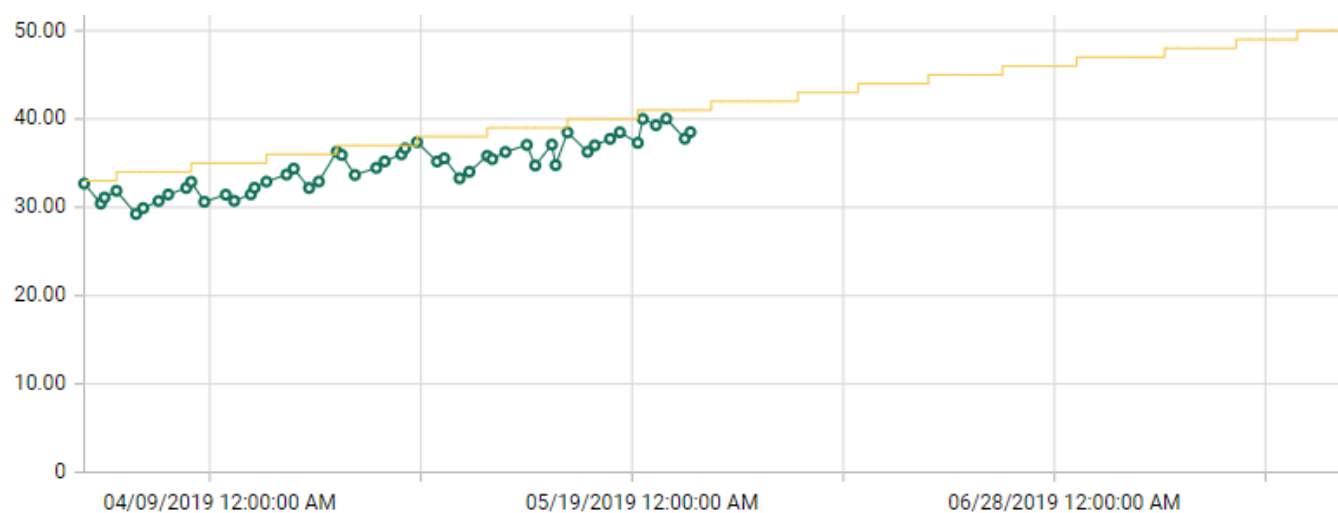
Null values

By default, when the *Y Value* is NULL, the chart doesn't display any shape. Instead, you can force to display a point with a predefined value for NULL values.

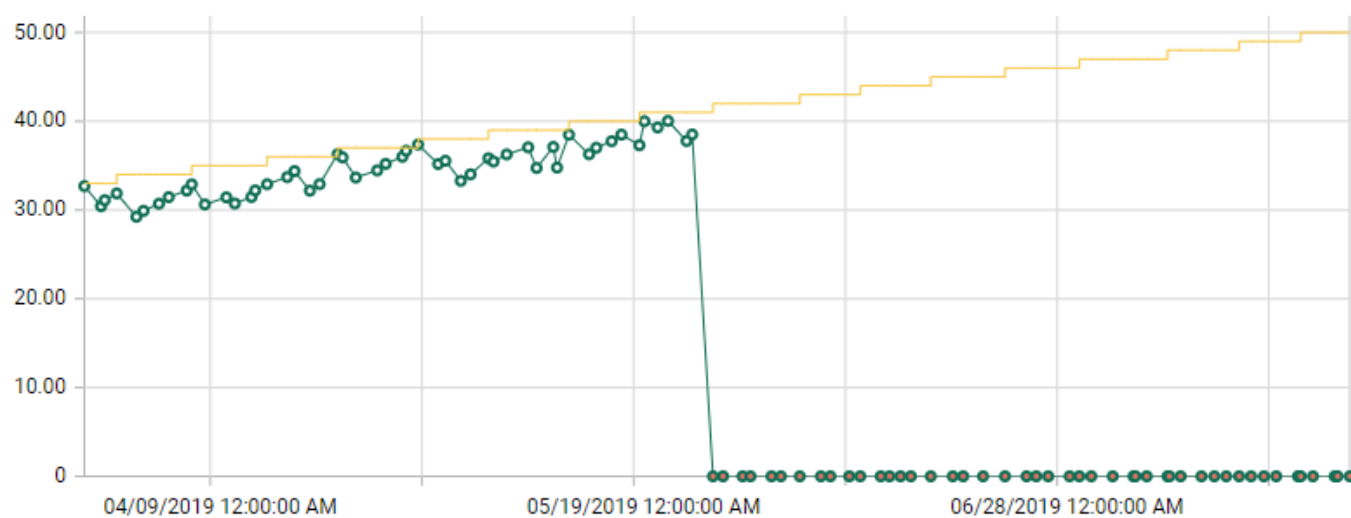
Edit the Series Settings, in tab *Advanced*, check *Plot Null Values* :

Plot Null Values	<input checked="" type="checkbox"/>
Fallback Value	0
Color	<input checked="" type="checkbox"/>

For example, in the following chart, there are NULL values after a certain date (data unknown) :



Instead of a broken line, you may choose to display those unknown values as 0 with a red point :



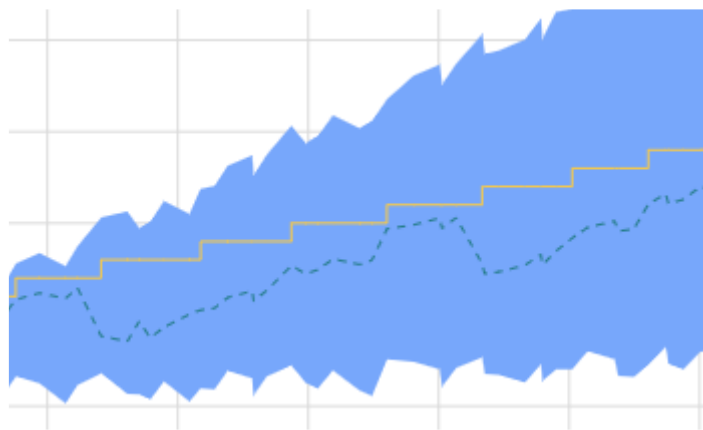
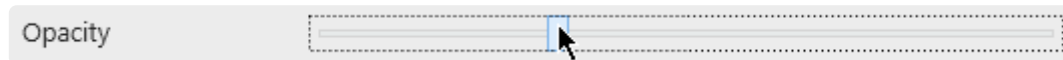
Last modified: 2019/05/24

2.7.1.5.4. Style and Appearance

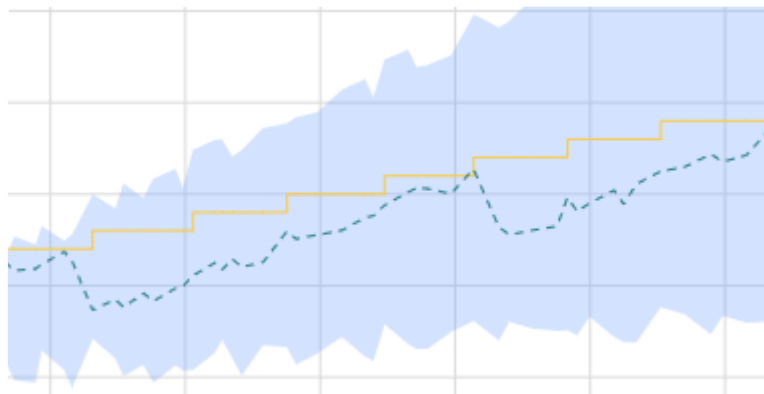
Opacity

Each chart series can be made semi-transparent to allow to see the other series behind it.

The *Opacity* of the Series can be configured in tab *Advanced* :



... will turn into :



Markers

Some Chart types can display markers on the series.

Edit the Series Settings, in tab *Advanced*, check *Show Markers* :

Show Markers



... will turn into :



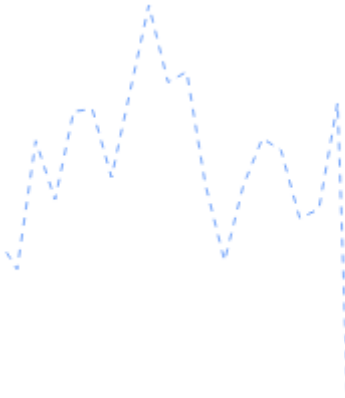
Dashed

Some Chart types can display a dashed line instead of a continuous line :

Dashed



... will turn into :



Border

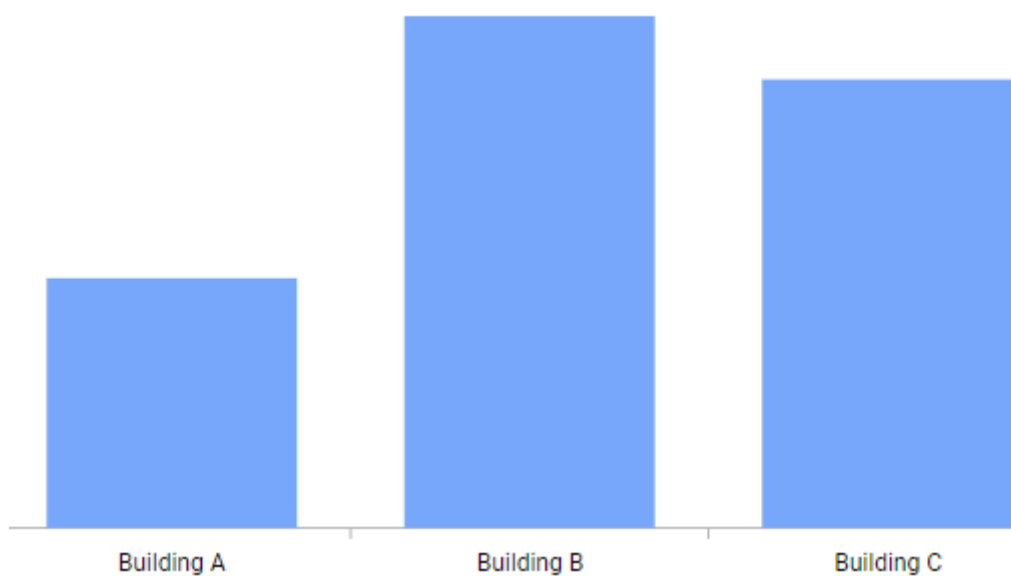
Some Chart types can have a border around the shapes :

Border Color	
Border Thickness	
Border Dashed	<input checked="" type="checkbox"/>

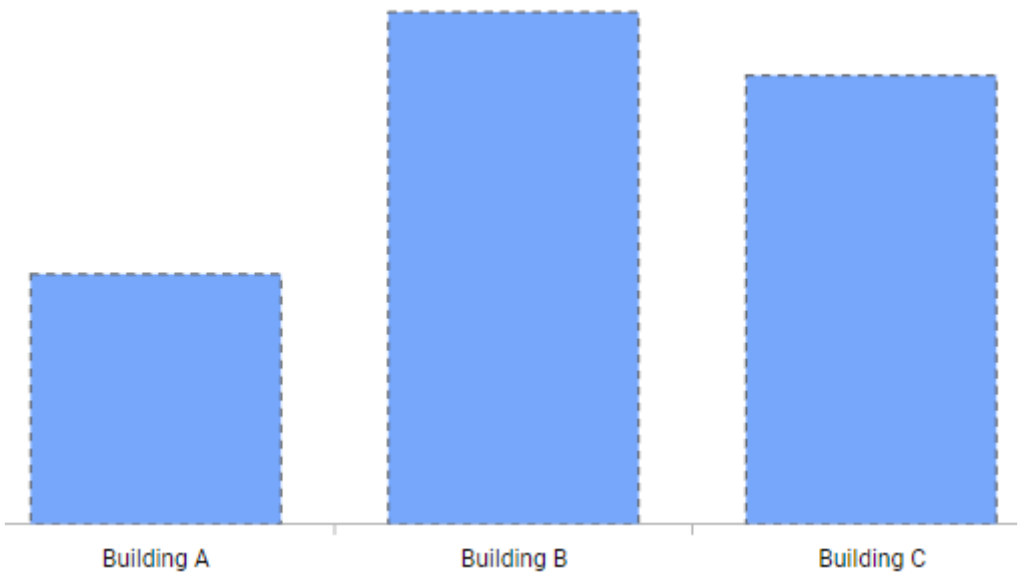
Border Color : select a color from the color picker

Border Thickness : give a value greater than zero to see the border, increase to have a thicker border

Border Dashed : if the border exists (Thickness > 0), you can display it as a dashed line instead of continuous



... will turn into :



Last modified: 2019/05/24

2.7.1.6. Axes

Since Alpana v3.0, Charts have 2 types of axes :

- one X Axis : for displaying values in the *X Axis* field. It's the grouping axis : grouping categories are displayed along this axis.
- one or more Y Axis : for displaying values in the *Y Values Series* fields. It's the value axis : aggregated values are displayed along this axis.

Each axis type has different configurations described next.

Last modified: 2019/06/17

2.7.1.6.1. X Axis

X Axis configuration is located in the right side in the properties, under section *X Axis* :

X Axis

Custom Title

☐

Date

Show Title

☒

Override Theme Color

☐

Show Grid Lines

☐

Labels Overflow

Hide

Labels Rotation

0°

Value Type

Date / Time

Position

Bottom

Field

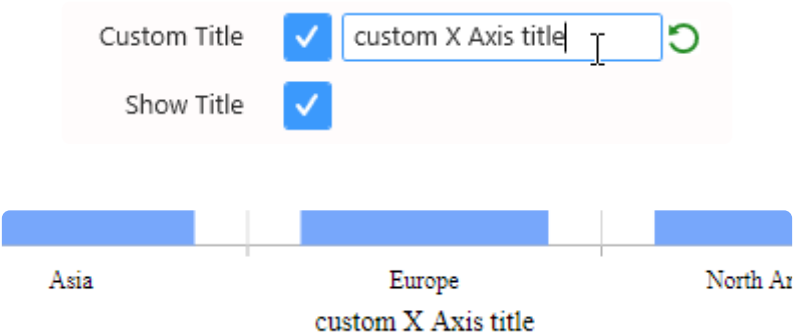
Date

Strip Line...

Title

X Axis has a default title generated from the name of the currently displayed data field.

If you wish to customize this title, check the box *Custom Title*, and write a new title :





When using Drill-down, the X Axis Title changes dynamically at each drill-down level to display the current X Axis field name.
If you set a *Custom Title*, it will be the same at all Drill-down levels.

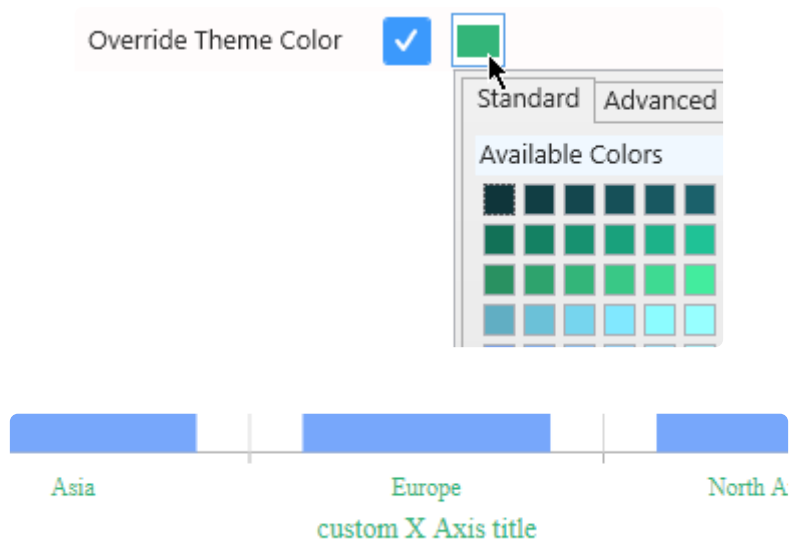
Hiding Axis Title

Use the *Show Title* check box to hide or show the title on the axis.

Color

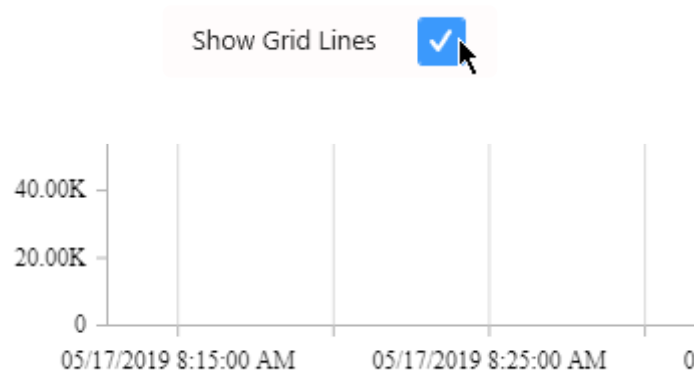
By default, the Labels and Title colors are based on the Theme.

If you want to set a specific different color, check *Override Theme Color* and set the desired color :



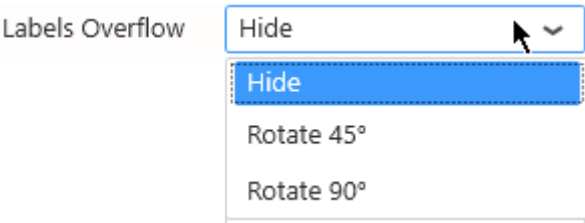
Grid Lines

Grid lines can be hidden or displayed with *Show Grid Lines* :



Labels Overflow

When Axis Labels are too long to be displayed, the following behaviors can be configured using *Labels Overflow* :



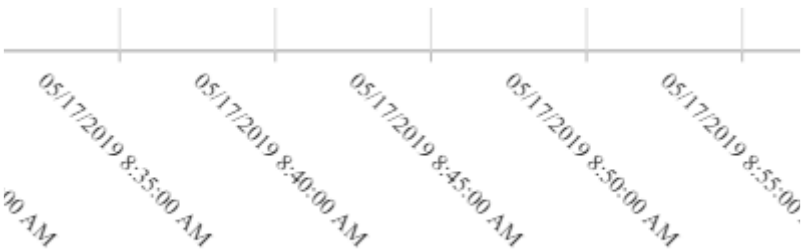
Hide

Labels are displayed entirely in their original orientation, but the labels that overlap are hidden :



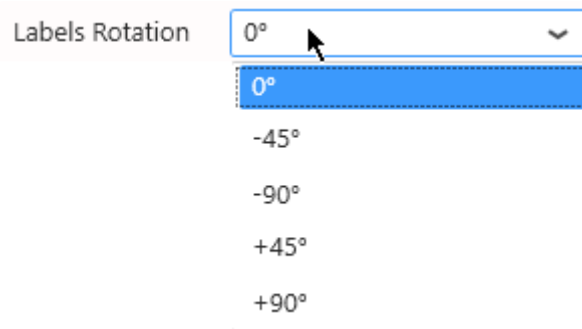
Rotate

Labels are displayed entirely, but if the labels overlap they are rotated to give more space :

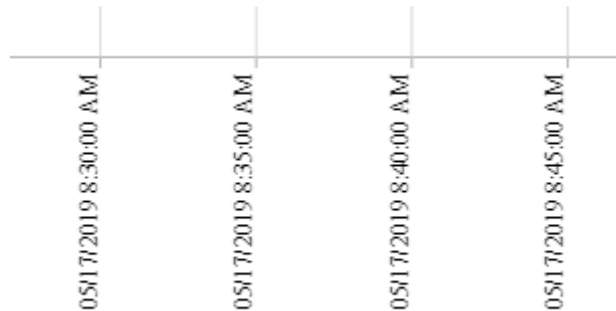


Labels Rotation

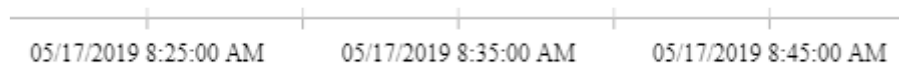
The Axis Labels orientation can also be rotated for better readability, by setting *Labels Rotation* :



It can allow for example to fit more labels when using the “Hide” Overflow Mode :



Or alternately to take less space vertically :

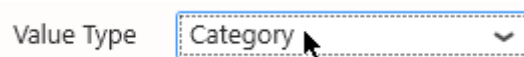


Value Type

Sometimes you want to compare groups of values side by side, and the distance on the X Axis doesn't matter.

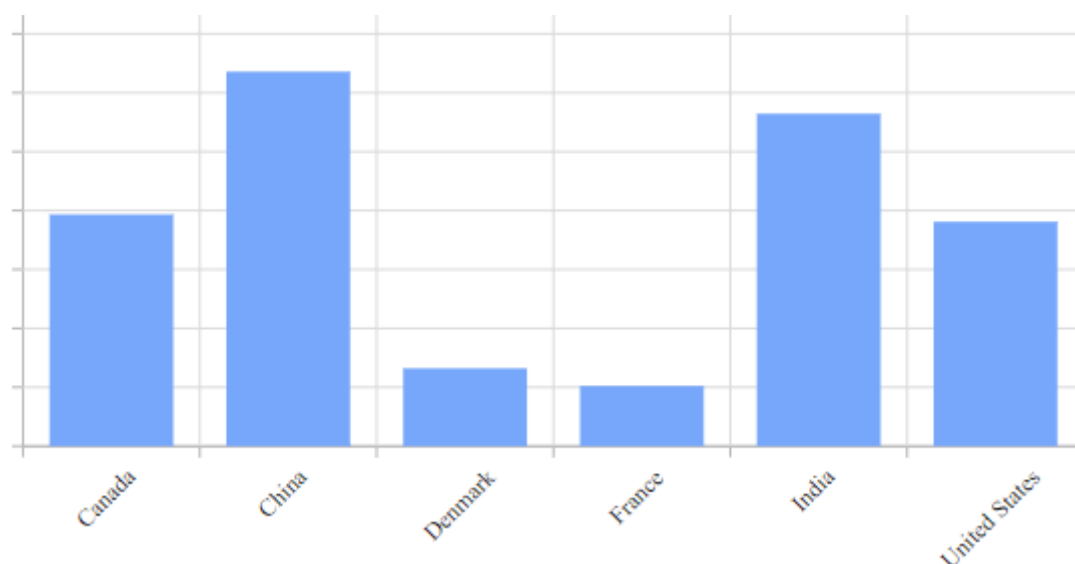
Other times, the distances on the X Axis are meaningful and you want to show how close or far the points are horizontally.

This is what the X Axis *Value Type* setting is for :



Category

This is the default, and the only option available for text type of fields. With *Category* X Axis type, all items on the X Axis are equally spaced.

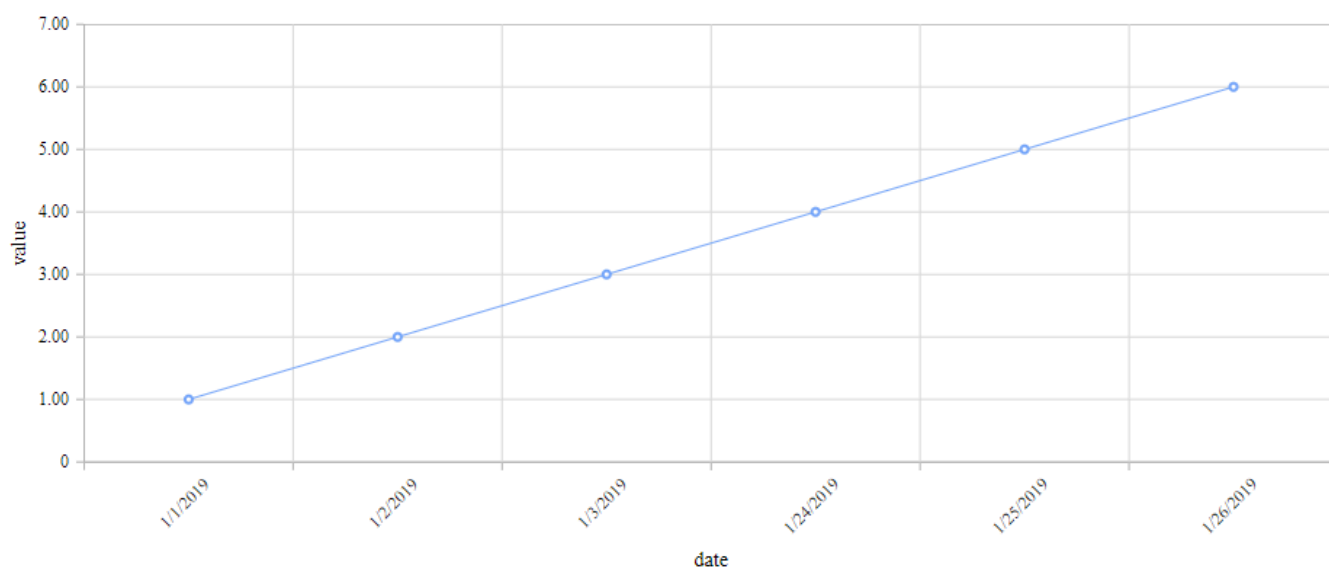


Date / Time

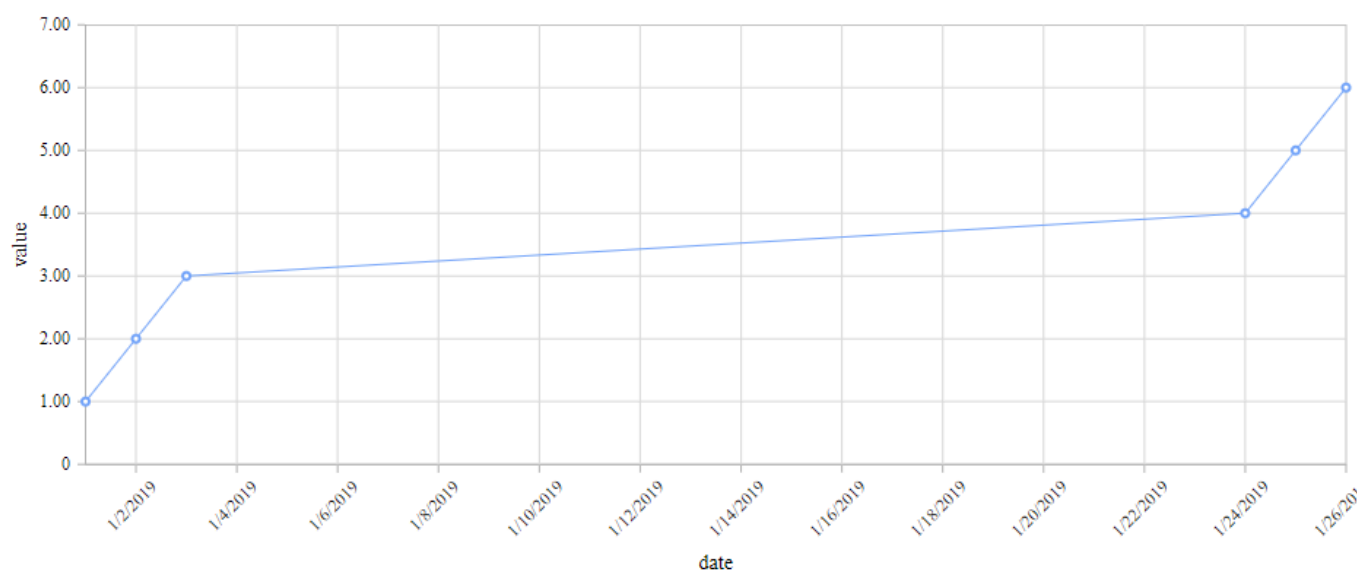
When the data field is of type Date, DateTime or Time, the *Date / Time* option is available.

The X position of points depends on the time difference in the data.

For example, the following data has missing points between day 3 and day 24 but this is not clearly visible and the progression looks linear :



With *Date / Time* type, the real trend is visible :

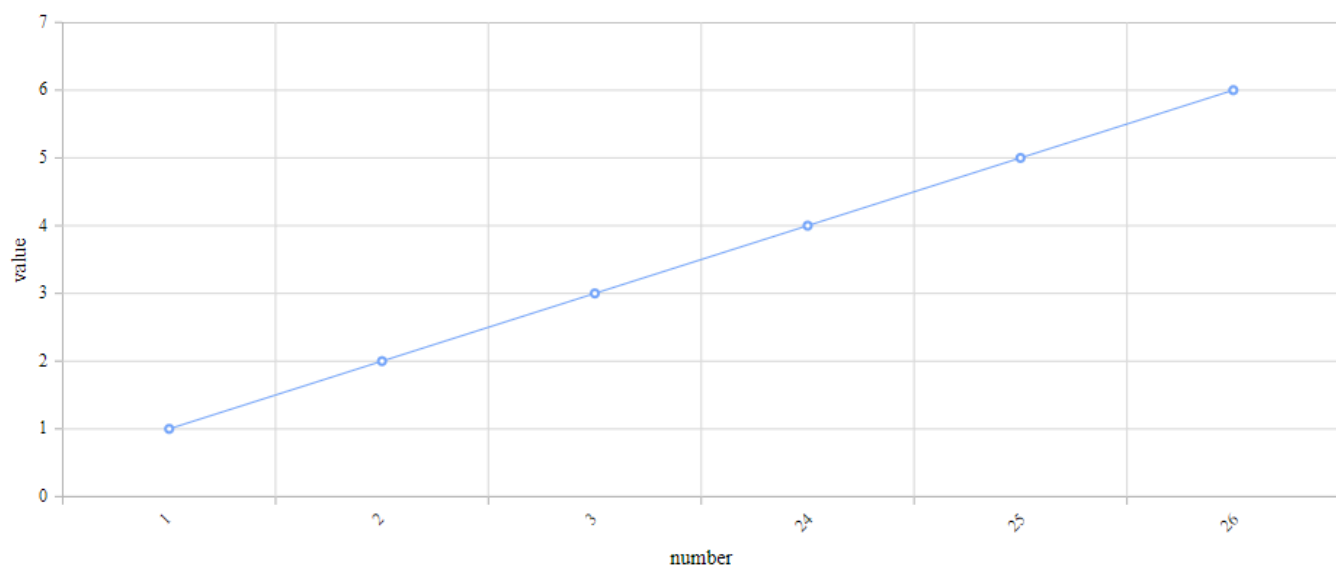


Numeric

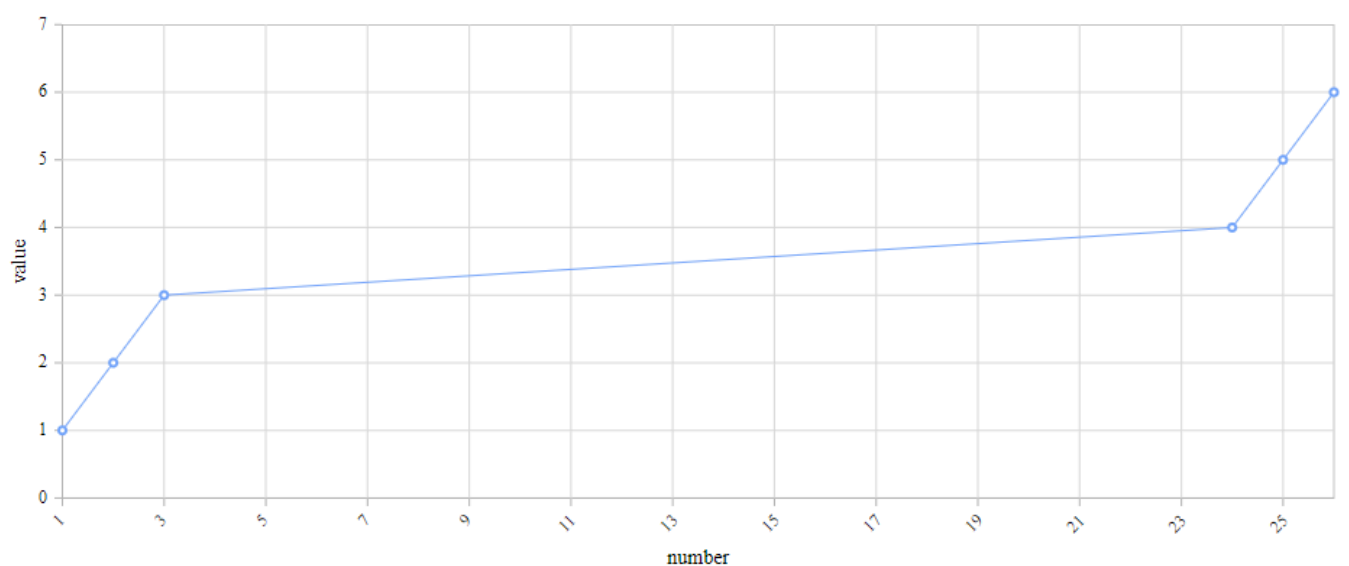
When the data field is of type Number, the *Numeric* option is available.

The X position of points depends on the number difference in the data.

For example, the following data has missing points between X value 3 and X value 24 but this is not clearly visible and the progression looks linear :



With *Numeric* type, the real trend is visible :

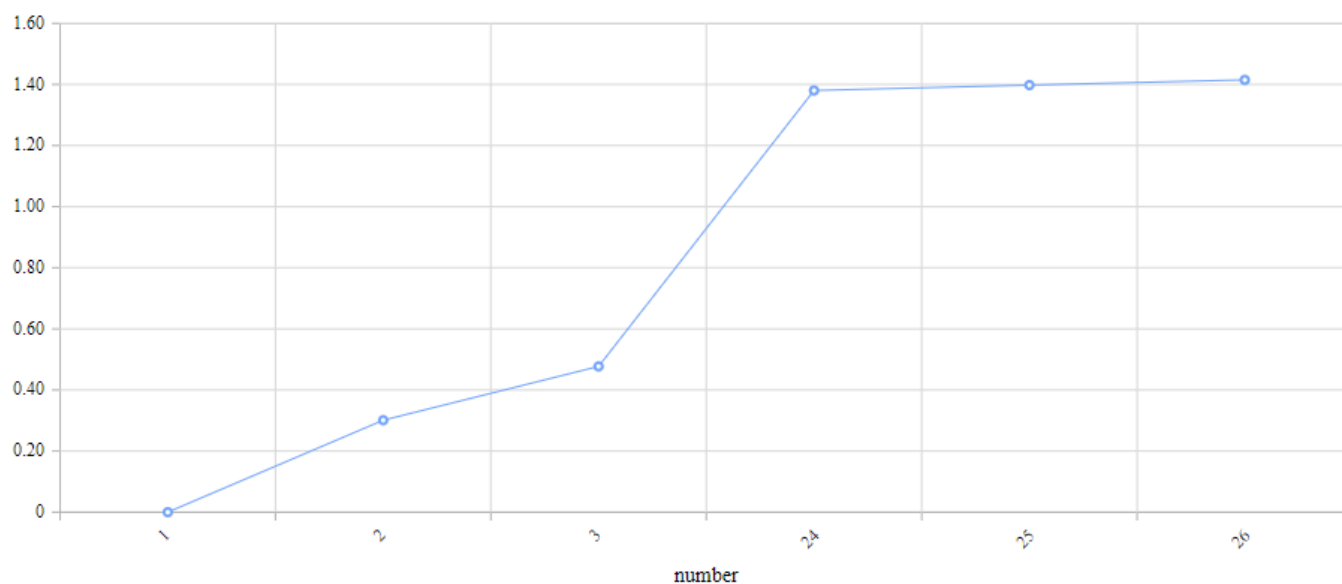


Logarithmic

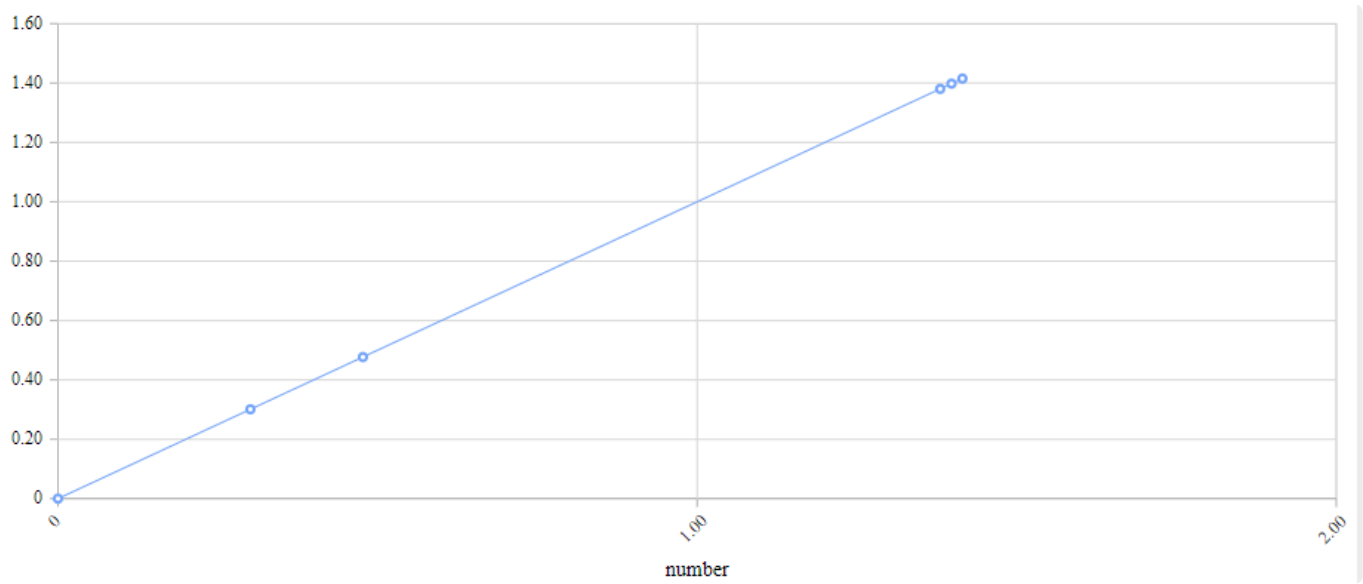
When the data field is of type Number, the *Logarithmic* option is available.

The X position of points depends on the number difference in the data.

For example, the following data has missing points between X value 3 and X value 24 but this is not clearly visible and the progression looks random :



With *Logarithmic* type, the real trend is visible :

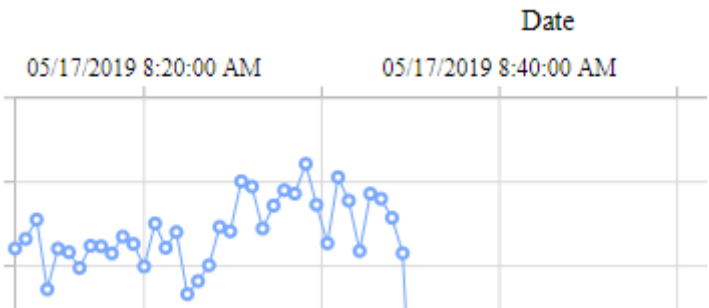


Position

The X Axis *Position* can be set to *Top* instead of the default *Bottom* :

Position

Top



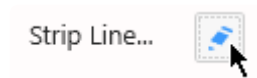
Last modified: 2019/05/24

2.7.1.6.1.1. Strip Line

Strip Lines allow to change the background color of the full chart between a minimum and maximum X value.

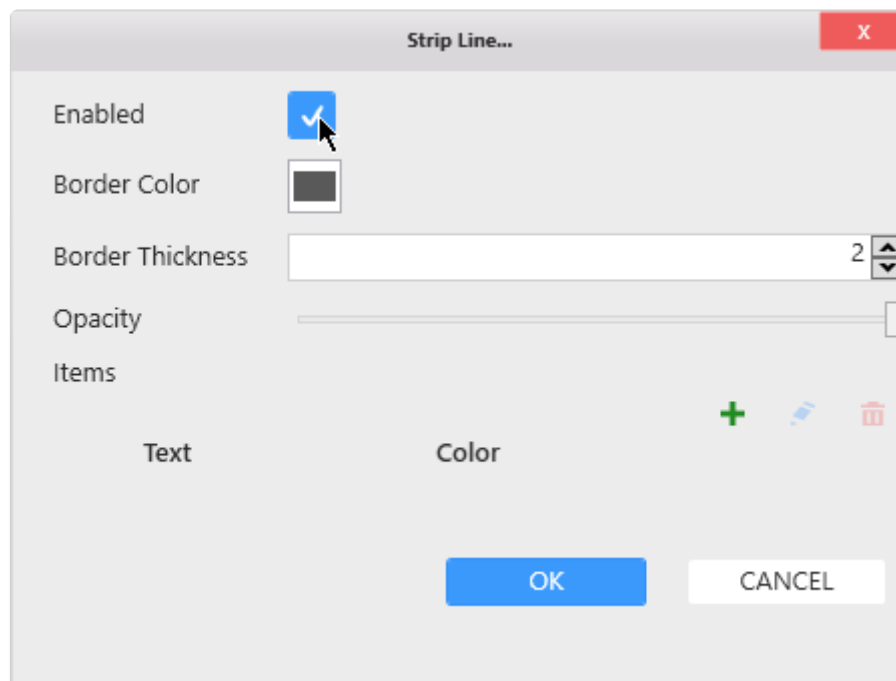
This can be useful for adding context information to a chart, for example the states of an equipment.

To open the Strip Line editor, click on *Strip Line...* :



Global Properties

In the *Strip Line...* pop up window, the top part lists some global properties of the Strip Line :



Enable

Check the *Enabled* box to enable to Strip Line and start configuring.

Border

The Strip Line can have a border to better show where is starts and ends.

Use *Border Color* to set the color.

Use *Border Thickness* to set the thickness. A thickness of 0 means no border.

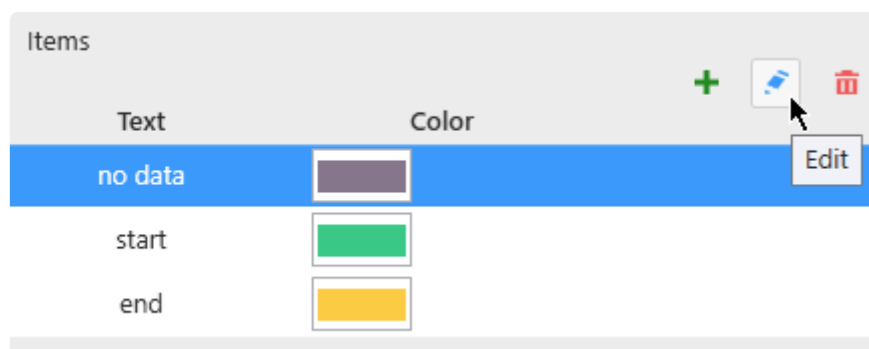
Opacity

The Strip Line can be made semi-transparent using the *Opacity* control.

This allows to integrate better with the background color of the theme and the rest of the chart.

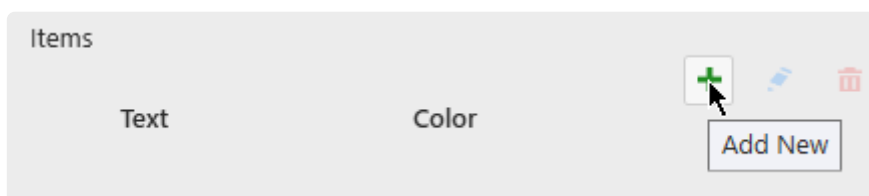
Items

There can be several X Axis Strip Lines, and they are all listed under *Items*.



Adding a Strip Line

To create a first Strip Line, click the + icon :



This opens the *Items* configuration pop-up window to let you configure this Strip Line. See below for *Configuring*.

Later, to add more, use the + button again.

Editing a Strip Line

When Strip Lines exist, you can edit one by selecting it in the list and clicking the pen icon.

This opens the *Items* configuration pop-up window to let you configure this Strip Line. See below for *Configuring*.

Removing a Strip Line

When Strip Lines exist, you can remove one by selecting it in the list and clicking the trash icon.

Configuring

Style and Appearance

The Style following options are available for each Strip Line :

Color	<input type="color" value="#4b4b8b"/>
Text	<input type="text" value="no data"/>
Foreground	<input type="color" value="#ffffff"/>
Text Alignment	<input type="text" value="Middle"/>

Color : Fill color of the Strip Line.

Text : Text label displayed on the Strip Line. Leave empty to hide text.

Foreground : Text color

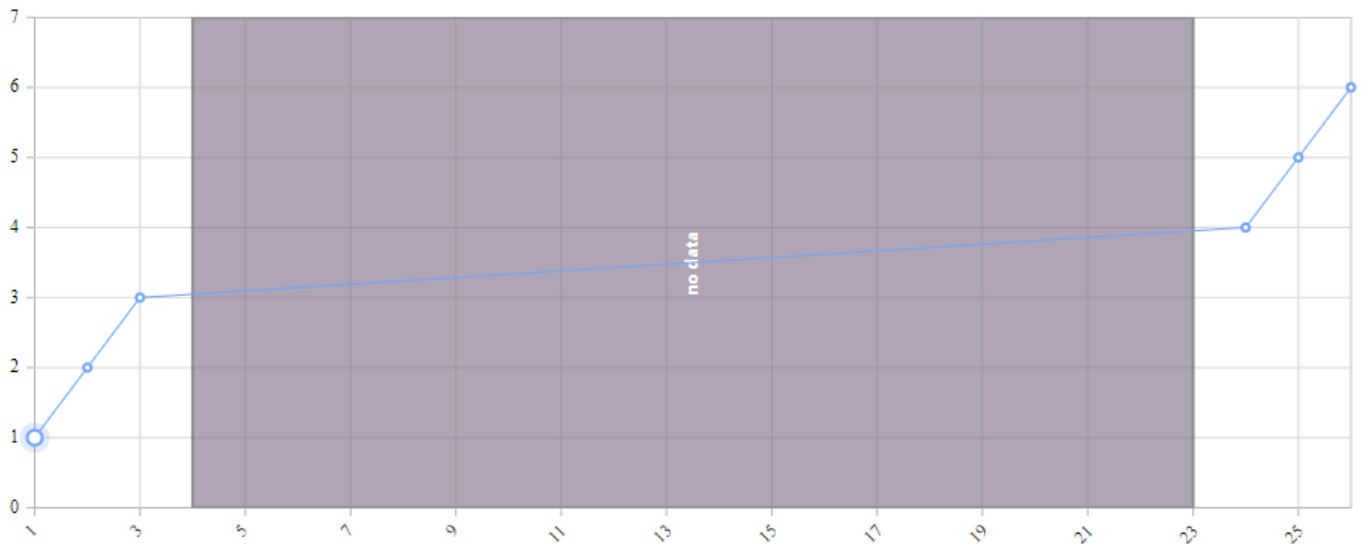
Text Alignment : Text position : at the start, end or middle of the Strip Line.

Manual mode

The Start and End value can be configured statically by using “Manual” *Mode* :

Mode	<input type="text" value="Manual"/>
Start	<input type="text" value="4"/>
End	<input type="text" value="23"/>

In this example, it means the Strip Line will start on X Value = 4 and end on X Value = 23 :



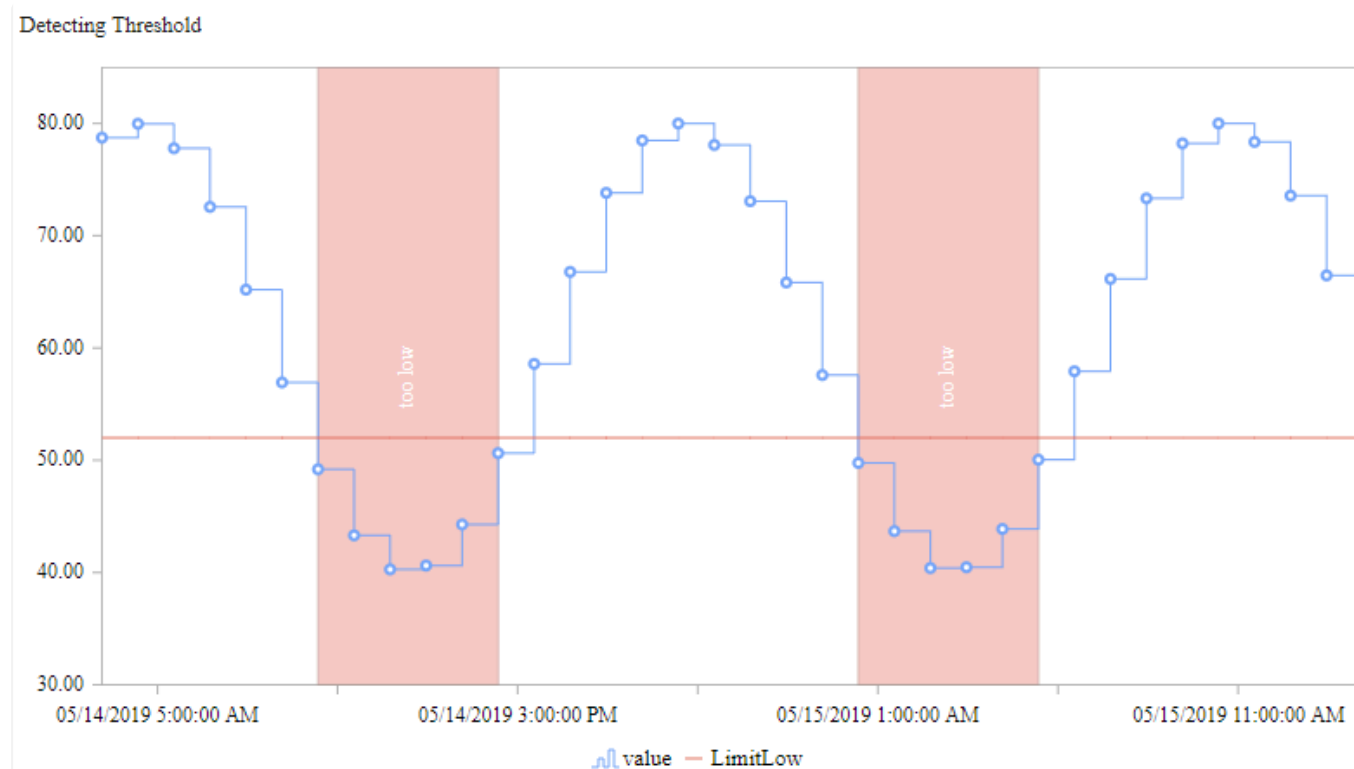
- * If the X Axis Value Type is “Category”, then the Strip Line *Start* and *End* values are the indexes of the points on the X Axis. The indexes start at 0.
Example : You have 6 X values : A, B, C, D, E, F in order
A Manual Strip Line starting at 1 and ending at 4 means that the first X value to start the Strip Line is B (zero-based index) and the End is at E.

Field Value Mode

! The *Field Value Mode* is only valid if the X Axis Value Type is not “Category”.

Instead of static values, the Strip Line can be displayed depending on a condition based on the data. With *Field Value Mode*, you configure a condition, so that when `Field = Value` then the strip line is displayed.

Mode	Field Value	▼
Field	isDanger	▼ + ×
Value	1	



Field

A list of fields is available to select from.

If your field is not available, click the + icon on the right to add a new field to the widget query :



Select the desired aggregation and field, and **click the check icon to validate** :

Field of ☒ ☐

Value

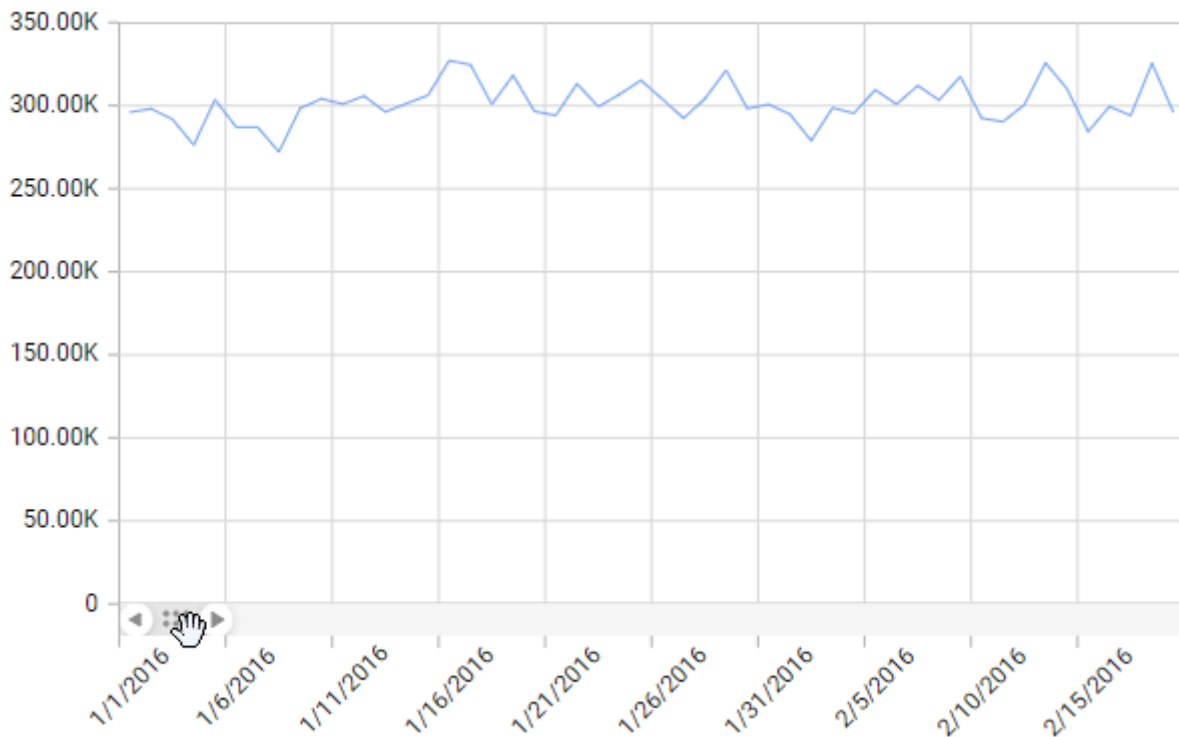
As a Value you can enter any number.

Last modified: 2019/07/23

2.7.1.6.1.2. Paging / Scrolling

When too many data points are displayed in a chart, performance may be affected.

That's why a paging mechanism has been set up to limit how many data points are loaded at a time. When this is active, a horizontal scrollbar is displayed on the X Axis to let you scroll to the rest of the data :



This default behavior can be configured under *Basic Settings* :

Enable Scrolling ☒

Page Size

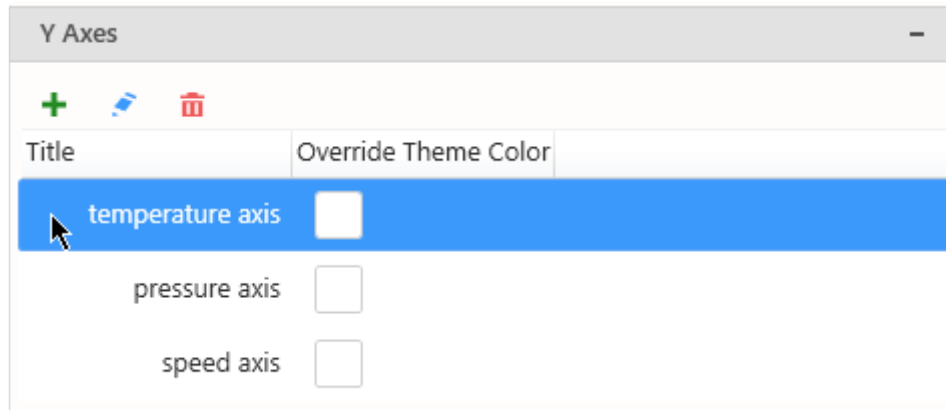
Enable Scrolling : enable / disable this behaviour. When disabled, all the data points are always loaded.

Page Size : the maximum number data points displayed per page

Last modified: 2019/05/24

2.7.1.6.2. Y Axes

All Y Axes are listed in the right side in the properties, under section Y Axes :



Managing the list of Axes

Adding

By default, only one Y Axis exists, named after the first *Y Value* Series field.

Click the + icon to add a new Y Axis.

This opens the *Edit Axis* configuration pop-up window to let you configure this new Y Axis.

See below for *Configuring*.

Editing

Click the pen icon to edit the currently selected Y Axis.

This opens the *Edit Axis* configuration pop-up window to let you configure this Y Axis.

See below for *Configuring*.

Removing

Click the trash icon to remove the currently selected Y Axis.

Binding a Series to a Y Axis

When a new Y Axis is created, no Series are bound to it.

To bind a *Y Value* Series to a specific Y Axis, see [Series Settings](#).

Y Axes Properties

Title

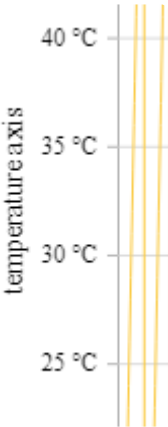
The Y Axis can be customized of hidden :

Custom Title

☒

Show Title

☒




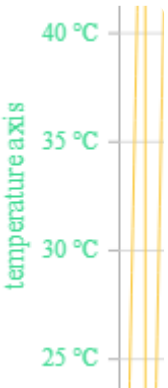
Color

By default, the Labels and Title colors are based on the Theme.
If you want to set a specific different color, check *Override Theme Color* and set the desired color :

Override Theme Color

☒



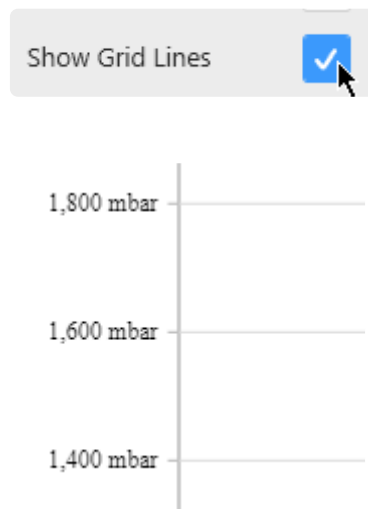


There is also a shortcut in the list of Y Axes :



Grid Lines

Grid lines can be hidden or displayed with *Show Grid Lines* :

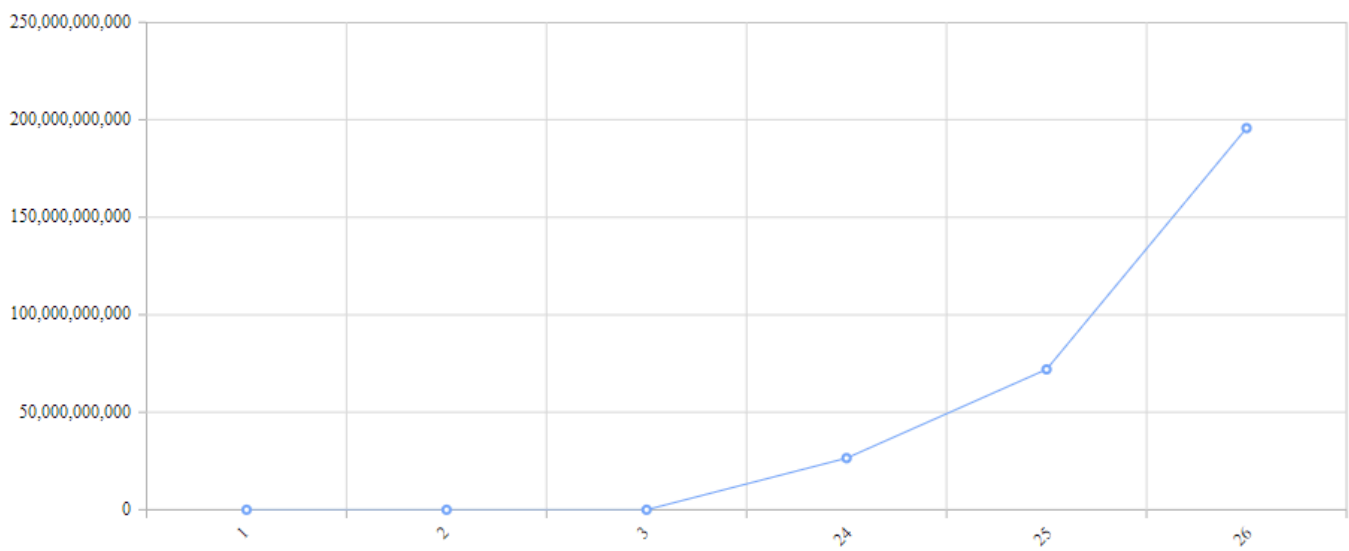


Value Type

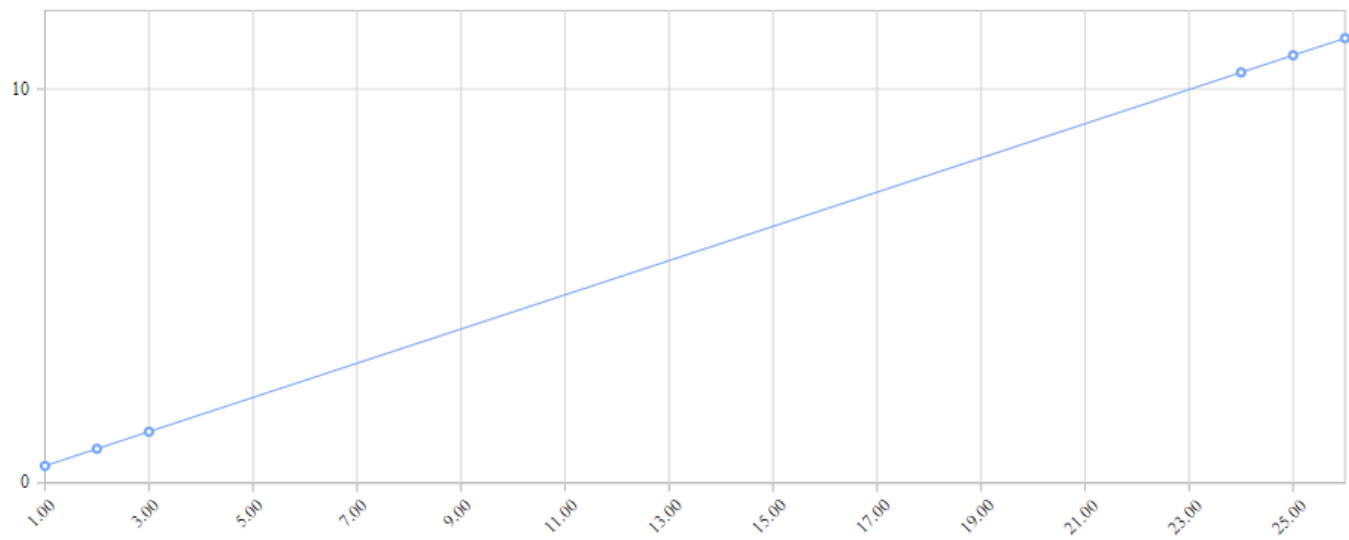
By default, the Y Axis *Type* is “Numeric” : this means that the Y position of points directly maps to the numeric value of the *Y Value* Series data field.

The *Type* “Logarithmic” can also be selected : this means that the Y position maps to the logarithm of the *Y Value* Series data field.

For example, here is a simple chart with Y Axis as “Numeric” (default) and X Axis as “Category” (default) :



The variations happen on many orders of magnitude, and it's difficult to track what is the real trend. With a “Logarithmic” *Value Type* of the Y Axis, and Numeric for the X Axis, we see that the trend is exactly exponential :



Custom Range

By default, the minimum and maximum value of the Y Axis is automatically calculated to scale to the data displayed.

Instead, you can force a *Custom Range* :

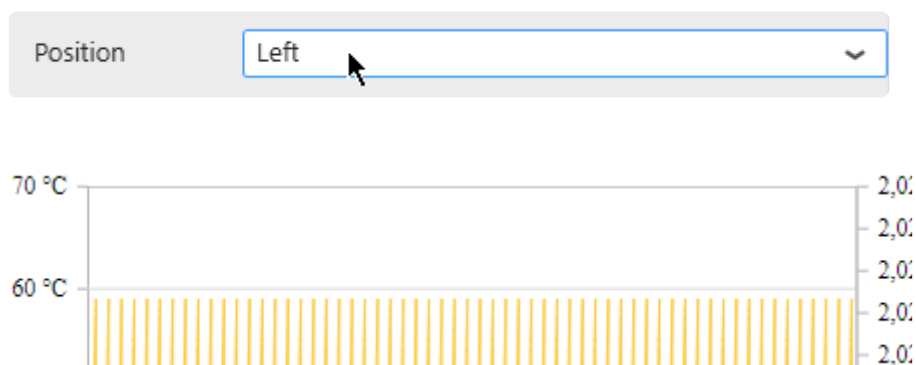
Custom Range ☒

Min

Max

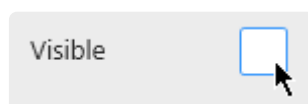
Position

Each Y Axis can be positionned to the *Left* or *Right* :



Hiding the Axis

All the Y Axes can be hidden by un-checking the *Visible* box.



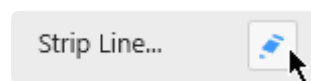
Last modified: 2019/05/24

2.7.1.6.2.1. Strip Line

Strip Lines allow to change the background color of the full chart between a minimum and maximum Y value.

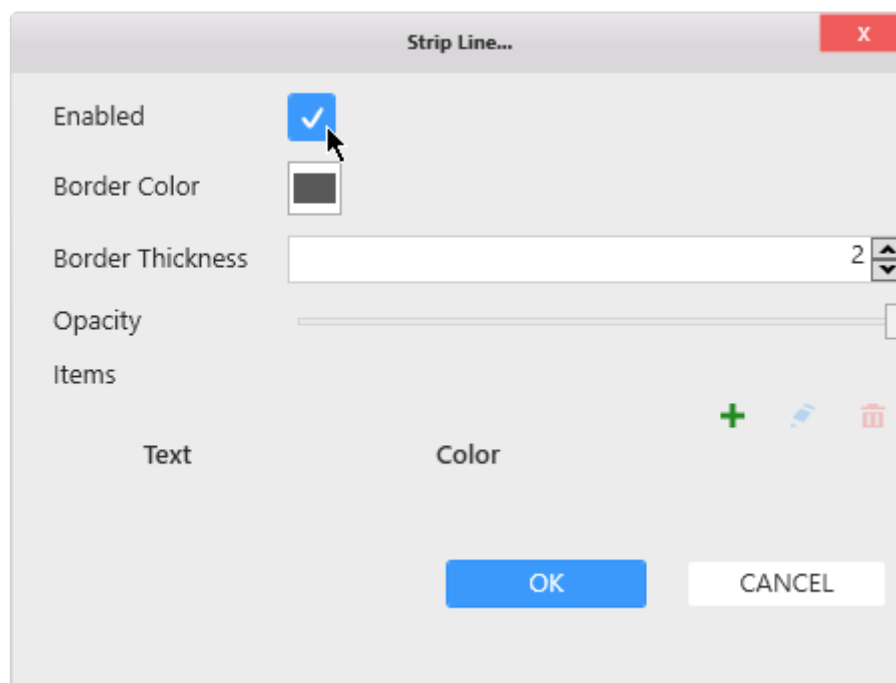
This can be useful for adding context information to a chart, for example the limit values of a measurement.

To open the Strip Line editor, click on *Strip Line...* :



Global Properties

In the *Strip Line...* pop up window, the top part lists some global properties of the Strip Line :



Enable

Check the *Enabled* box to enable to Strip Line and start configuring.

Border

The Strip Line can have a border to better show where it starts and ends.

Use *Border Color* to set the color.

Use *Border Thickness* to set the thickness. A thickness of 0 means no border.

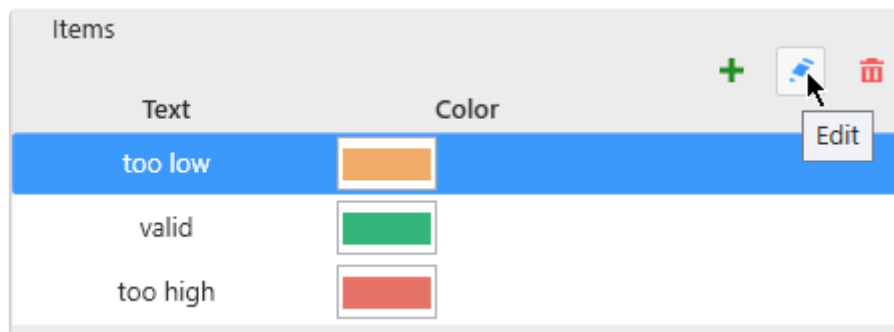
Opacity

The Strip Line can be made semi-transparent using the *Opacity* control.

This allows to integrate better with the background color of the theme and the rest of the chart.

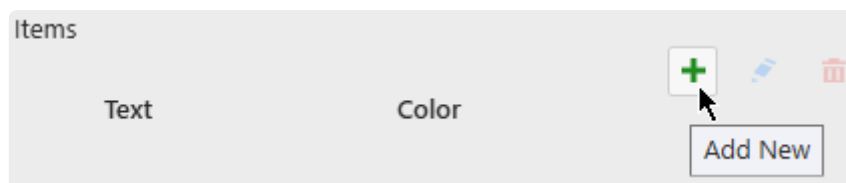
Items

There can be several Y Axis Strip Lines, and they are all listed under *Items*.



Adding a Strip Line

To create a first Strip Line, click the + icon :



This opens the *Items* configuration pop-up window to let you configure this Strip Line.

See below for *Configuring*.

Later, to add more, use the + button again.

Editing a Strip Line

When Strip Lines exist, you can edit one by selecting it in the list and clicking the pen icon.

This opens the *Items* configuration pop-up window to let you configure this Strip Line.
See below for *Configuring*.

Removing a Strip Line

When Strip Lines exist, you can remove one by selecting it in the list and clicking the trash icon.

Configuring

Style and Appearance

The Style following options are available for each Strip Line :

Color	<input type="color" value="#FFA500"/>
Text	<input type="text" value="too low"/>
Foreground	<input type="color" value="white"/>
Text Alignment	<input type="text" value="Middle"/>

Color : Fill color of the Strip Line.

Text : Text label displayed on the Strip Line. Leave empty to hide text.

Foreground : Text color

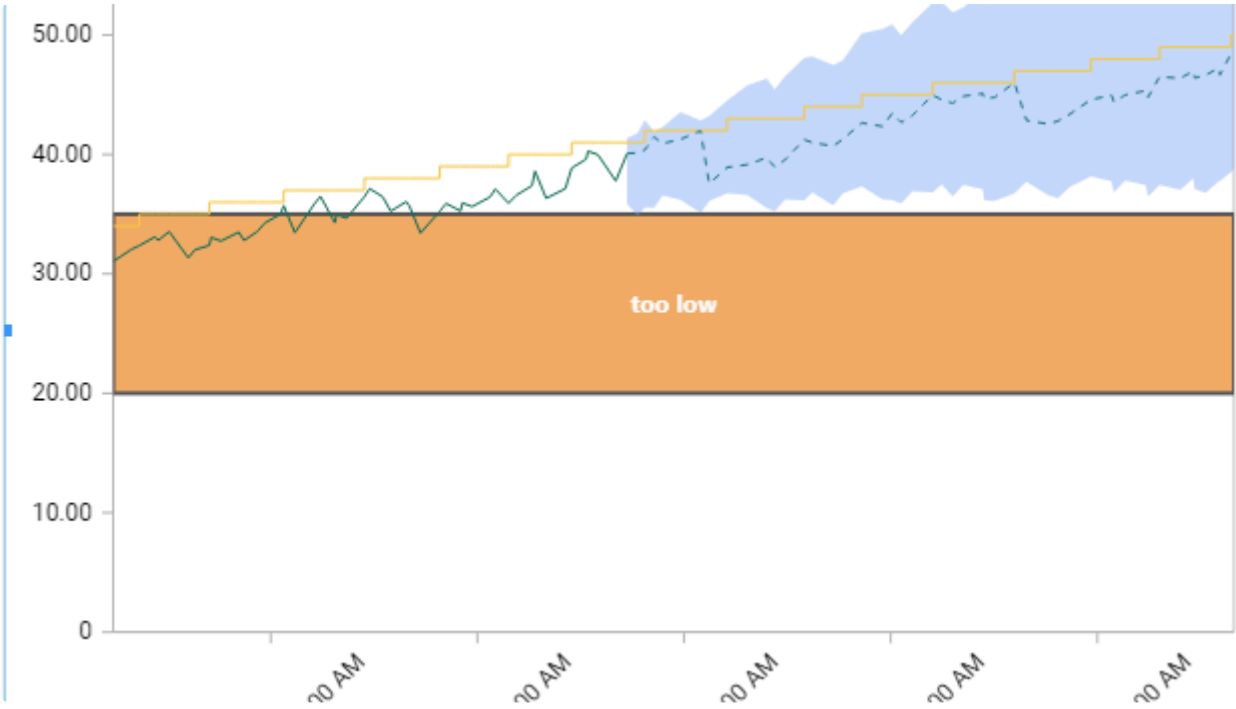
Text Alignment : Text position : at the start, end or middle of the Strip Line.

Start / End

The *Start* and *End* value can be configured statically :

Start	<input type="text" value="20"/>
End	<input type="text" value="35"/>

In this example, it means the Strip Line will start on Y Value = 20 and end on Y Value = 35 :



Last modified: 2019/05/24

2.7.1.7. Legend

Series Legend

The Legend shows the meaning of each series :

▲ predicted range — value 📊 set_point — predicted_value

Hide/Show Legend

The chart Legend can be hidden or displayed using *Show Legend* under *Basic Settings* :

Show Legend ☒ Bottom

Legend position

The chart Legend can be positioned on any side of the widget :

Show Legend ☒ Bottom

- Bottom
- Top
- Left
- Right

Customize Legend

Legend items are automatically generated from the field names and values, but you can customize how they are generated under *Group Settings* :

Group Settings


Split into many charts

Title Format
{{:Row}}

Split into the same chart

Title Format
{{:Value}} {{:Row}}

With *Split into the same chart*, you can use the placeholder `{{:Row}}` to be replaced by the field value :

Title Format 

 total Production of country Canada
 total Production of country China
 total Production of country India
 total Production of country United States

* The `{{:Value}}` placeholder gets replaced by the [Series Title](#).

Hide items from the Legend

Any series can be hidden from the Legend.

Edit the Series Settings, in tab *Advanced*, un-check *Show In Legend* :

Show In Legend ☒

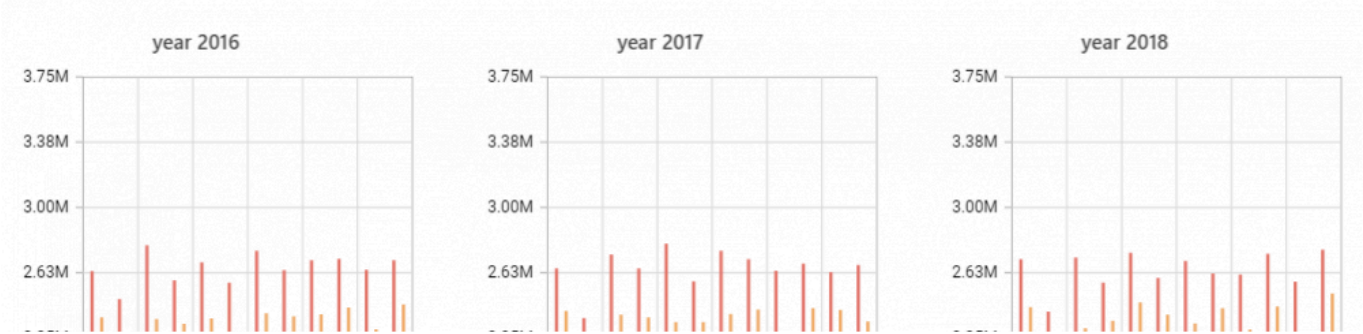
Customize Titles of contained charts

Group Settings also allows to configure the series titles when the Chart has a field bound to *Split into many charts*.

Depending on your data configuration, not all configurations may be available because they wouldn't have meaning.

With *Split into many charts*, you can use the placeholder `{{:Row}}` to be replaced by the field value :

Title Format 



Last modified: 2019/05/24

2.7.1.8. Managing Series Color

Indexed Colors

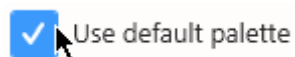
By default, Series are assigned a color based on their index in the ordered list in which they appear (usually alphabetical order).

Item1 is assigned Color1, Item2 is assigned Color2, etc...


The palette has 15 colors. When there are more series items than 15, the colors assigned loop back to Color 1.

Default Palette (default)

To enable the Default Palette (it is enabled by default), check *Use Default Palette* under *Series Palette* :

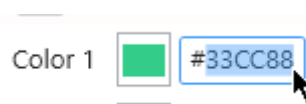


The list of colors in the default palette is below :

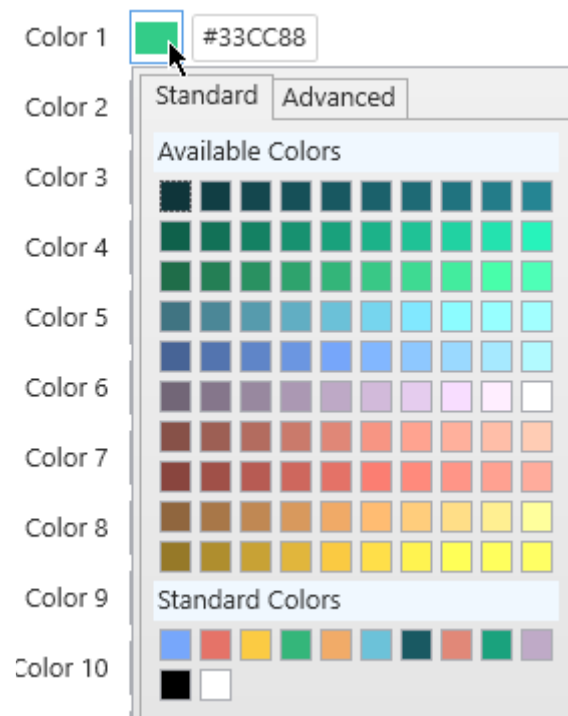
Color 1		#77A7FB
Color 2		#E57368
Color 3		#FBCB43
Color 4		#34B67A
Color 5		#F1AB68
Color 6		#6CC2D9
Color 7		#195962
Color 8		#E18878
Color 9		#1AA27D
Color 10		#BFAAC7
Color 11		#89695E
Color 12		#A5C77F
Color 13		#F48FB1
Color 14		#90CAF9
Color 15		#CC4452

Custom Palette

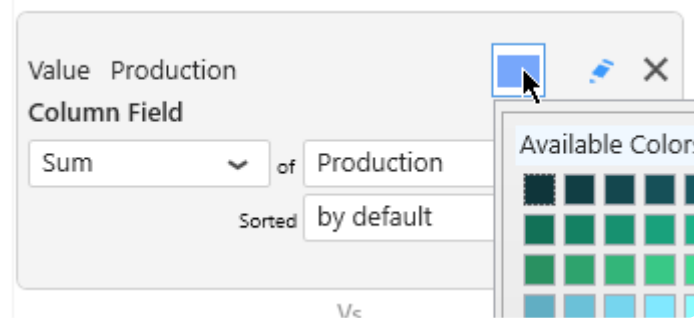
To start customizing the Default Palette, un-check *Use Default Palette* under *Series Palette*. Then you can customize colors either by entering their hexadecimal code (#RRGGBB) :



... or by clicking the color and using the graphical color picker :



When there is no data field bound to *Split into same chart*, there is also a shortcut to each serie's color next to the series settings :



✿ If *Use Default Palette* is checked, then this setting is disabled.

Value-based Colors

What are Value-based colors ?

But what if your series changes over time ?

For example, when binding a field as *Split into the same chart*, the items are generated dynamically from values in that field. Their index cannot be known in advance.

When items are renamed, added, or deleted from the data, the index of existing items will change, and

their color will change !

This could happen simply during filtering at runtime for example.

Instead of index-based, you may then choose **Value-based** colors : the same value in the Series field will always have the same color in the Chart.

For this, simply check *Use value-based colors* under *Series Palette* :



Customizing Value-based colors

Value-based colors are defined **at the Data Source level**.

This is in order to allow re-use of the colors everywhere the same data is used.

To configure value-based colors, see the corresponding chapter [Value-based Settings](#), under [Transforming Data](#).

Last modified: 2019/05/24

2.7.1.8.1. Chart value color binding

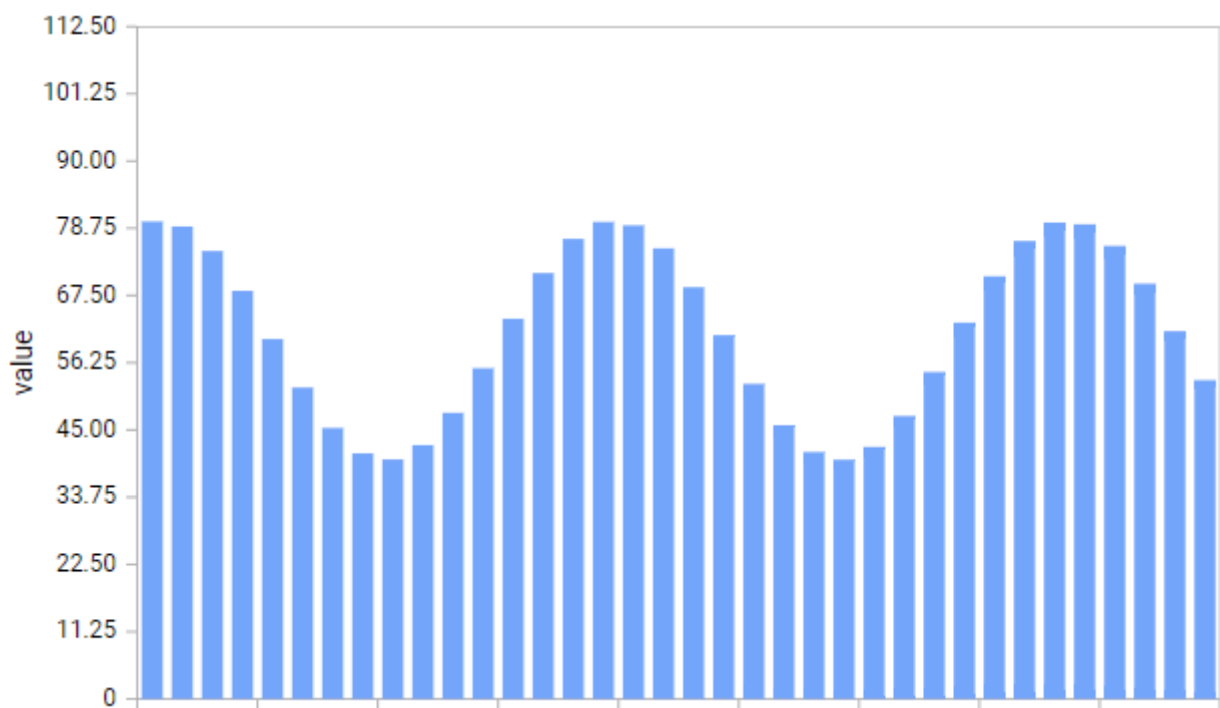
What is Color Binding

Instead of static colors, a series color can be calculated using a condition based on the data.

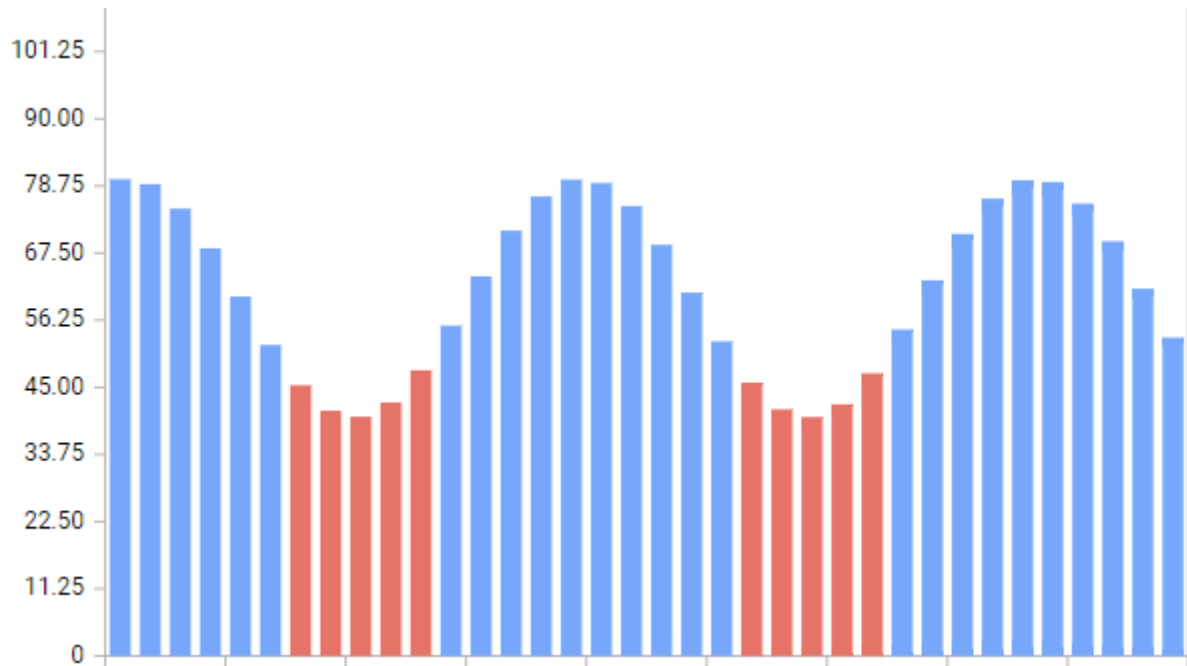
With *Color Bindings*, you configure a condition, so that when `Field = Value` then the color is applied.

Example

On a Column Chart with a blue series :



... you want the color to become red in some conditions :



Enabling Color Binding

Edit the Series Settings, in tab *Advanced*, check *Color Bindings* to enable the feature :

Colors Bindings ☒

Color Field value + ×

+ ×

Data Value Target Color

Selecting a field

Select a *Color Field* from the existing fields in the chart query, or click the + icon to add a new aggregated field to the query :

Color Field value + ×

Select the desired aggregation and field, and **click the check icon to validate** :

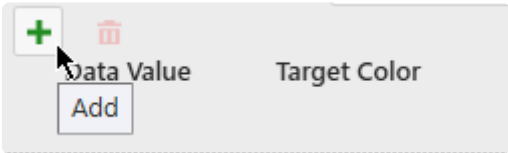
Color Field Sum of limit_type × ✓ ×

Managing conditions

Based on the selected field, several conditions can be configured : `Field = Value1`, `Field = Value 2`, ...

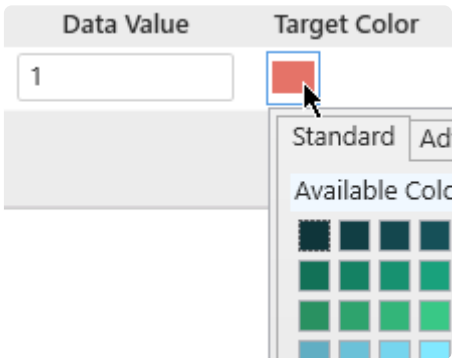
Adding a condition

Click the **+** icon on top of the list to add a condition :



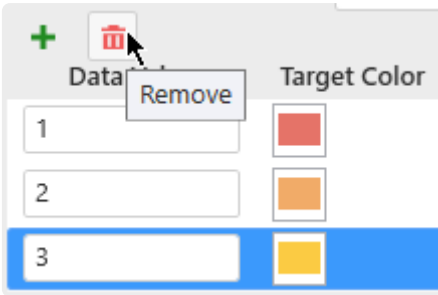
Configuring a condition

Configure a *Data Value* and a *Target Color* so that when `Field = DataValue`, then the color of the series turns into *Target Color* :



Removing a condition

If you wish to remove a condition from the list, select it by clicking it, then remove it using the trash icon :



Supported Chart Types

Some Chart Types like *Line* only show this value color when [Show Marker](#) is configured.

The following chart types don't support value color binding :

- Spline
- Step Line
- Area, Stacked Area, 100% Stacked Area
- Spline Area
- Step Area
- Bubble
- Scatter
- Range Area
- Hi Lo
- Hi Lo Open Close
- Candle
- Waterfall

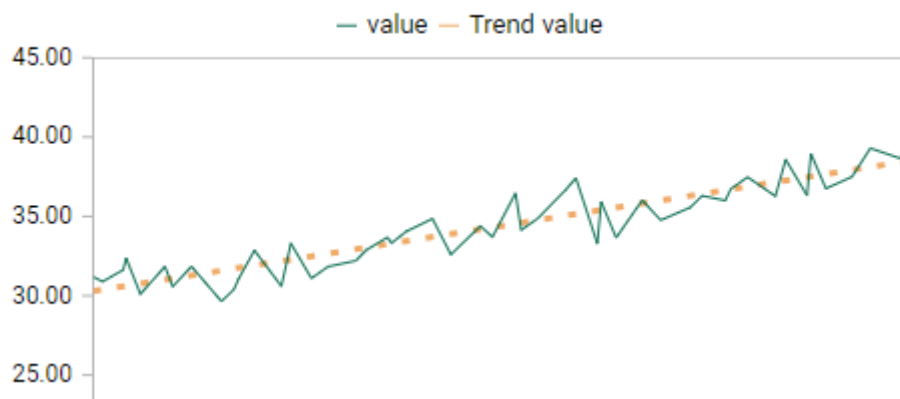
Last modified: 2019/05/24

2.7.1.9. Trend Lines

Introduction

In addition to displaying the data itself, each series can be configured to display a *Trend Line*. This can be useful for displaying a linear regression or an average on the data points.

Example



Enabling the Trend Line

To enable a Trend Line on a series, go to the Series Settings, in tab *Advanced*, check *Trend Line* :

Trend Line	<input checked="" type="checkbox"/>
Legend Text	<input type="text" value="Trend value"/>
Line Type	<input type="text" value="Linear"/>
Line Color	<input type="color" value="#FFA500"/>
Line Style	<input type="text" value="Dashed"/>

Configuring the Trend Line

Legend Text : label of the Trend Line withing the [Legend](#)

Line Type : type of regression used to match the data values

Line Color : the Trend Line can have a different color from the series color

Line Style : the Trend Line can be displayed as a continuous line or as a variety of dashed lines

Last modified: 2019/05/24

2.7.2. Polar Chart

Polar Charts allow to view data on a 2D plane using polar coordinates : a center at 0 and values displayed as distance to that center.

Last modified: 2019/05/24

2.7.2.1. Binding Data

Introduction

The following fields are available for configuring data binding on a Polar Chart :

Show

Y Value(s)

Value Avg(OEE)

Draw Type Column

Source Field

Avg of OEE

Sorted by default

Vs

Sum of Add optional field...

As

Polar Chart

Group by

X Axis

Value of Entity

Sorted ASC

- *Y Value(s)* : the distance between the data point and the center of the chart
- *X Axis* : how we group values together at different angles to create points on the line

Show : *Y Value(s)*

This represents the *quantity* we want to display on the Chart, and will bind for example to the distance between the data point and the center of the chart.

These fields **must be aggregated** : you must choose an aggregation function (SUM, AVG, ...)

At least one field must be bound.

Multiple fields : multiple Series

When binding multiple fields at the same time, each field will generate a separate Series.




This way, each Series can have its own graphical properties, its own Draw Type, etc...


To change the order of fields, see [the corresponding chapter](#).

Example



With a field called "Price" and a field called "Target", you will chart 2 separate series, one for each field.


Y Value(s)

Value Avg(OEE)   




Draw Type Column 


Source Field

Avg  of OEE 



Sorted by default 


Vs

Value Avg(Target)   

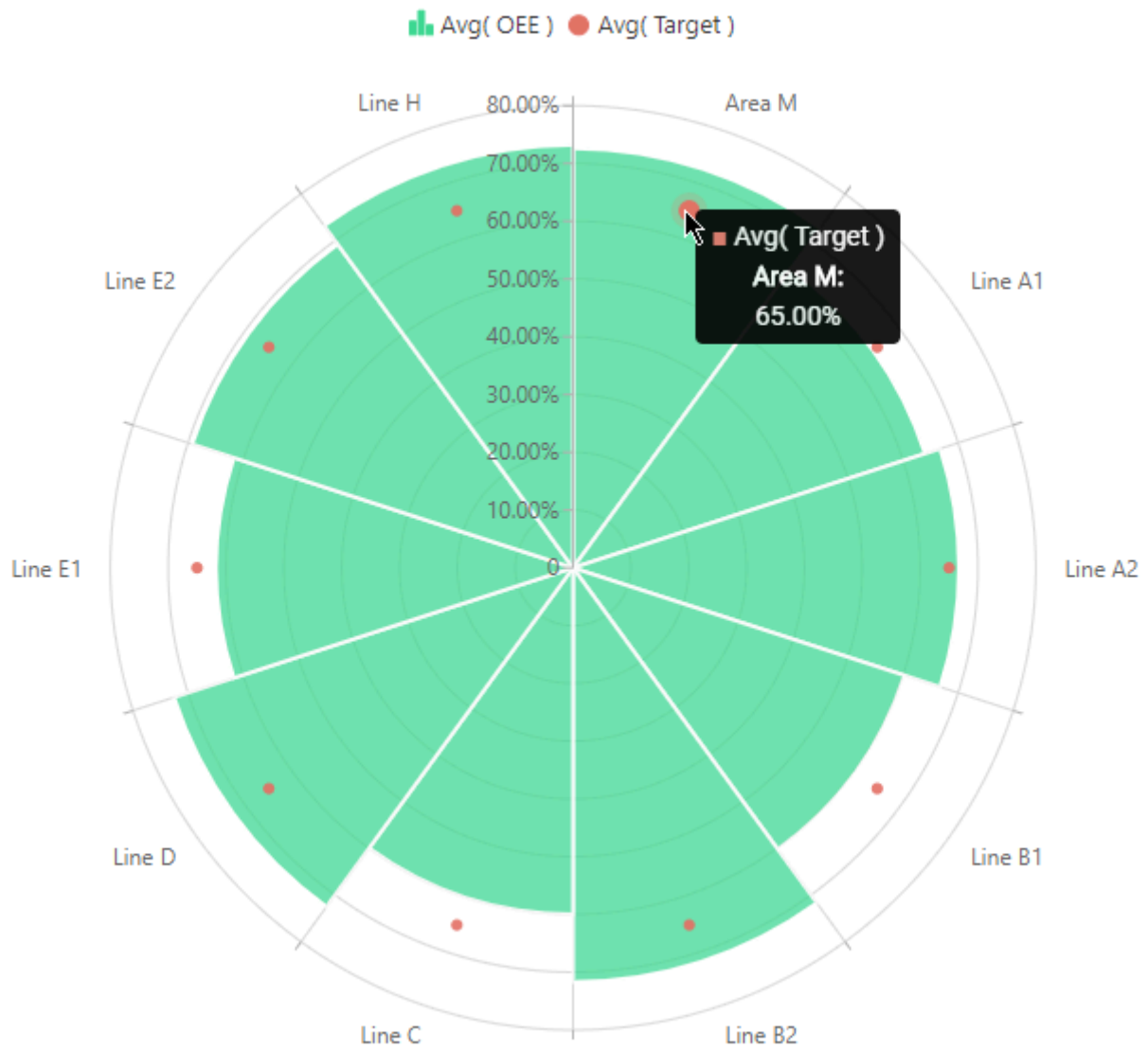
Draw Type Scatter 

Source Field

Avg  of Target 

Sorted by default 

..



Group by : *X Axis*

This represents the *category* we want to use to separate data points.

These fields **cannot be aggregated** : they must have several separate values in order to group the “Values” aggregation.

At least one field must be bound.

Last modified: 2019/05/24

2.7.2.2. Chart Types / Draw Types

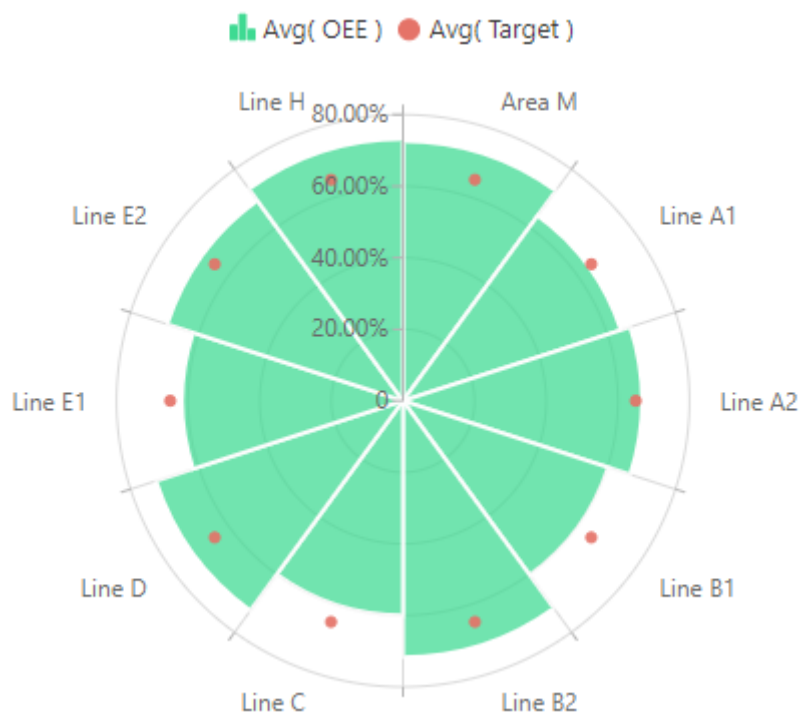
Chart Widget Types

There are 2 main types of polar charts :

Polar Chart

Constant values are circles.

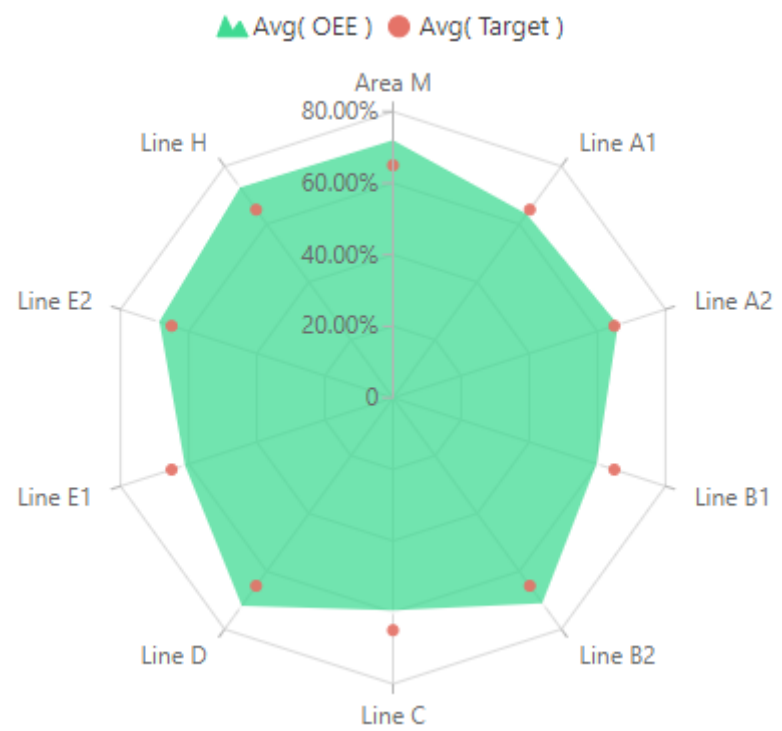
Each separate X Axis value is in the middle of a radial portion.



Radar Chart

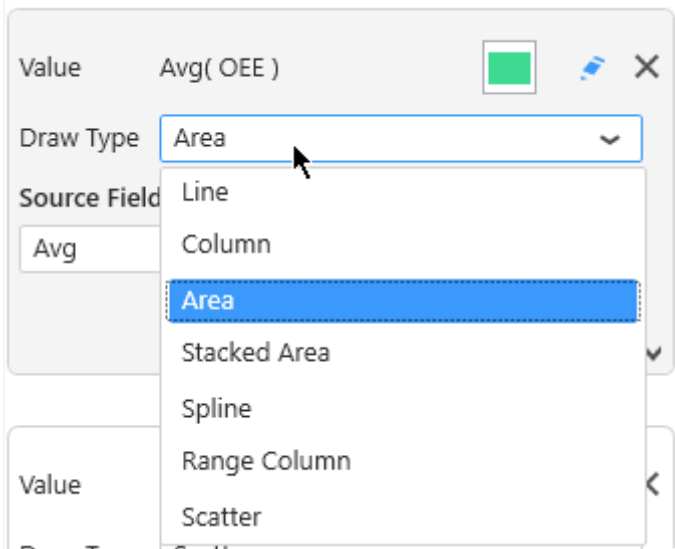
Constant values are straight lines.

Each separate X Axis value is on the edge of a radial portion.



Draw Types

For each Chart Type (Polar or Radar), each series can have a variety of *Draw Types* :



In the same chart, each separate series can have a different *Draw Type*.

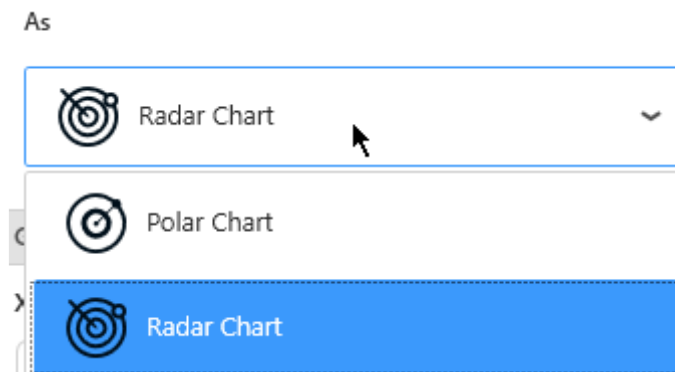
Last modified: 2019/05/24

2.7.2.3. Switching Chart Types

When adding a Widget, the Chart will have a single type globally.

The Chart type can be changed on the fly while keeping the widget configuration.

To change the Chart type, use the selector under *As* in the data configuration :



There is also a shortcut under *Basic Settings* :



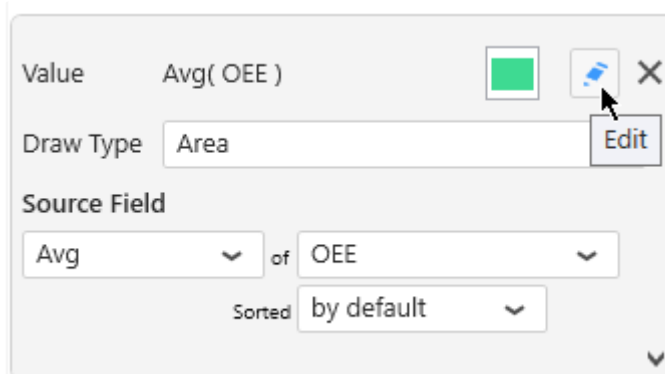
There is also a shortcut in the top bar :



Last modified: 2019/05/24

2.7.2.4. Series Settings

Each *Y Value* Series can be configured individually by clicking the pen icon next to its name :



The screenshot shows a configuration window for a chart series. At the top, the series name is 'Avg(OEE)' next to a green square icon. To the right of the icon are a pen icon (highlighted by a mouse cursor) and a close 'X' icon. Below the name is a 'Draw Type' dropdown menu set to 'Area', with an 'Edit' button to its right. Underneath is a 'Source Field' section with two dropdowns: the first is set to 'Avg' and the second to 'OEE', separated by the word 'of'. Below these is a 'Sorted' dropdown menu set to 'by default'. A small downward arrow is at the bottom right of the window.

This allows to fine-tune your chart by configuring exactly what you want on each series inside the widget.

✿ Keep in mind that some configuration options are not available for all chart types.

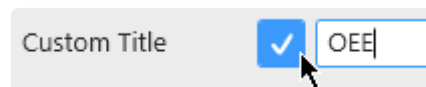
Last modified: 2019/05/24

2.7.2.4.1. Series Name

Each Series has a name displayed in various places.

By default, this name is automatically generated from the data field bound to the Series.

To configure this name, edit the Series Settings, check *Custom title* and write the desired name :



A screenshot of a user interface element for configuring a series name. It consists of a light gray rounded rectangle containing the text 'Custom Title' on the left. To the right of the text is a blue square checkbox with a white checkmark. Further right is a text input field with a blue border and a light gray background, containing the text 'OEE'. A black mouse cursor arrow is pointing at the checkbox.

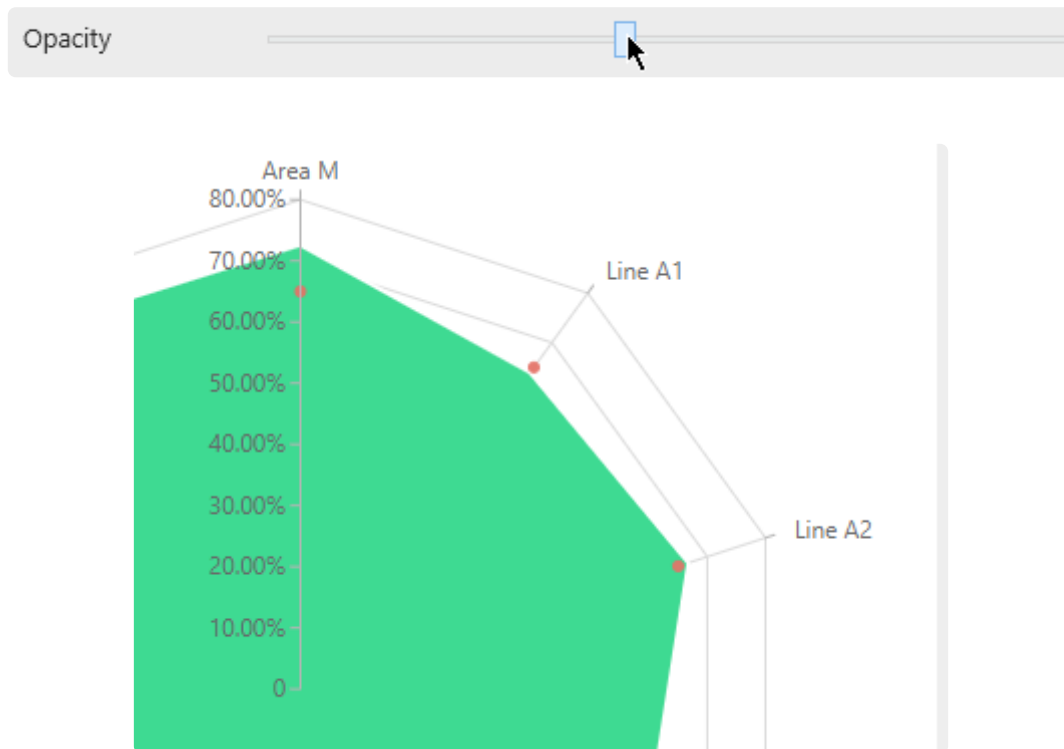
Last modified: 2019/05/24

2.7.2.4.2. Style and Appearance

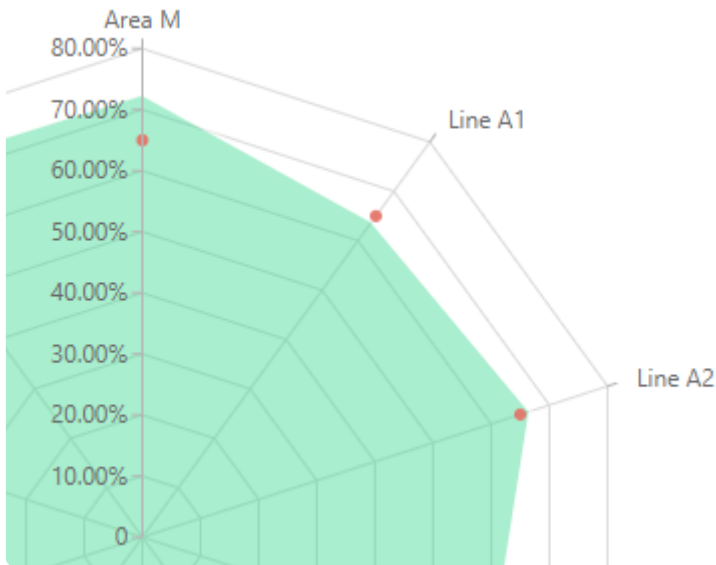
Opacity

Each chart series can be made semi-transparent to allow to see the other series behind it.

The *Opacity* of the Series can be configured :



... will turn into :



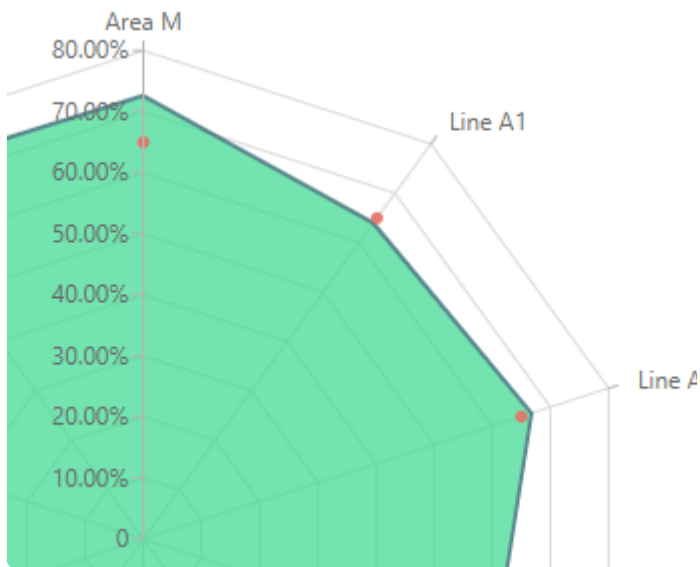
Border

Some Draw Types can have a border around the shapes :

Border Color

Border Thickness

- Border Color* : select a color from the color picker
- Border Thickness* : give a value greater than zero to see the border, increase to have a thicker border



Last modified: 2019/05/24

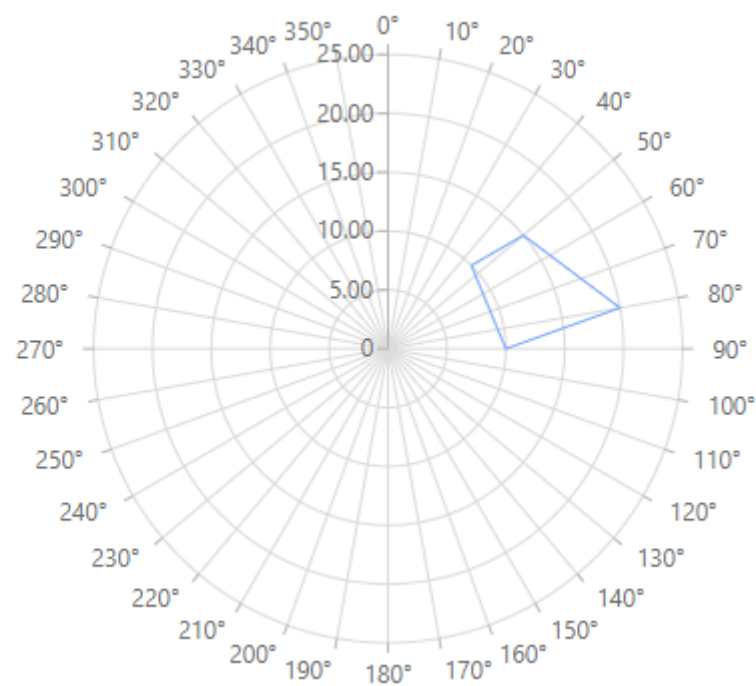
2.7.2.5. X Axis

By default, the angular position of X Axis items is calculated automatically to divide the plane evenly : $360^\circ / (\text{number of items})$

When the X Axis field is a Number, the minimum and maximum angles are calculated from the data.

Instead, if a Number type field is bound as X Axis, then you can force the X Axis scale to 0° - 360°

Force 0 - 360° ☒



Last modified: 2019/05/24

2.7.2.6. Legend

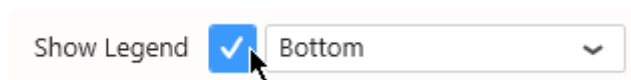
Series Legend

The Legend shows the meaning of each series :

 Avg(OEE)  Avg(Target)

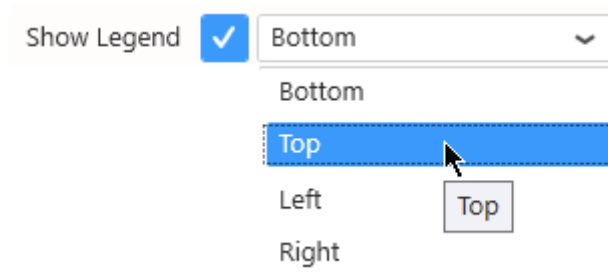
Hide/Show Legend

The chart Legend can be hidden or displayed using *Show Legend* under *Basic Settings* :



Legend position

The chart Legend can be positioned on any side of the widget :



Customize Legend


Legend items are generated from the [Series Title](#).

Chart Title

The chart has a Title.

By default, this is generated from the X Axis field name.

Instead, you can configure it under *Basic Settings* by checking *Use Custom Title* and writing the desired title :

Use Custom Title ☒ 





* You can hide this Title by writing an empty Custom Title.

Last modified: 2019/05/24

2.7.3. Map

Alpana allows to display geographical data using 2 types of maps :

-  Choropleth Maps show a color on the complete map areas
-  Bubble Maps show a bubble centered on the map areas

Last modified: 2019/05/14

2.7.3.1. Maps format : GeoJSON

File format

Alpana supports a map format called [GeoJSON](#)

This is a structured text format similar to Javascript Objects and can be edited with any text editor.

Geographic or X-Y

The shapes represented by Alpana maps can be :

- geographic : coordinates are longitudes/latitudes and a Mercator projection is applied
- normal : coordinates are X/Y and are considered as on a flat surface

Features

The GeoJSON format allows to define shapes as “Features” (mostly polygons) which Alpana can draw in Widgets as vector graphics very efficiently.

GeoJSON “*Features*” supported by Alpana are :

- Polygon : any set of points define a closed shape as a polygon
- MultiPolygon : any set of polygons can be grouped into a single shape (example : the British Isles may be considered a single shape made of several separate polygons)
- Point : a single point can also be represented (point of interest, city, etc)

Properties

Each Feature can possess a set of *Properties* which can contain any metadata.

These Properties allow map makers to give unique names to each shape, and will allow Alpana to bind dashboard data to each shape depending on its name (“to each Feature depending on its Property”).

These Properties are a dictionary of key-value pairs.

Finding or Creating new Maps

Editing

Some GeoJSON files can be easily edited by hand using a text editor (Notepad), especially for modifying Properties.

This can be made easier using a text editor with syntax highlighting for Javascript (or JSON) like Visual Studio or Notepad++.

Some easy-to-use graphical tools exist, including free web-based tools like <http://geojson.io/>

Creating

Creating maps from scratch is more difficult when using only a text editor.

One option is to use a graphical tool like <http://geojson.io/>

Another option is to generate the GeoJSON Features programatically (using Excel for example) by concatenating JSON strings and actual coordinates.

Finding GeoJSON Maps

It's easy to find existing GeoJSON maps with relatively open licenses on the web.

Usually just type the name of your region + "geojson", and you should find many results.

Last modified: 2019/03/08

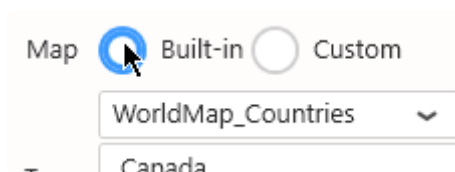
2.7.3.2. Choosing a map

Select a Map

Built-in Maps

Some geographical maps of the world are provided with Alpana.

They are available under Map Settings, by selecting Built-in :

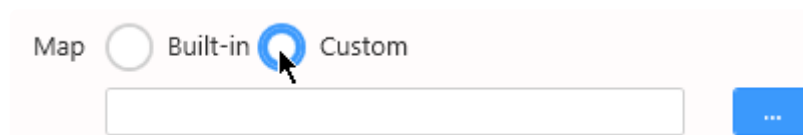


By default, the selected map shows the world's countries.

Custom Maps

You own custom GeoJSON files can be used instead.

Under Map Settings, select Custom :



Click the “...” to browse for a *.json file on your machine.



Note : Performance VS Details

This file will be embedded in the Dashboard itself.

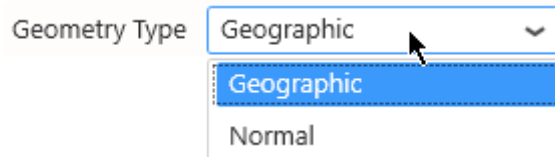
This may impact the file size of the web page generated by the dashboard.

Any time a user will load the dashboard in their web browser, they will download the page in their browser. If the map is huge with many details, this may impact dashboard performance.

Geometry type

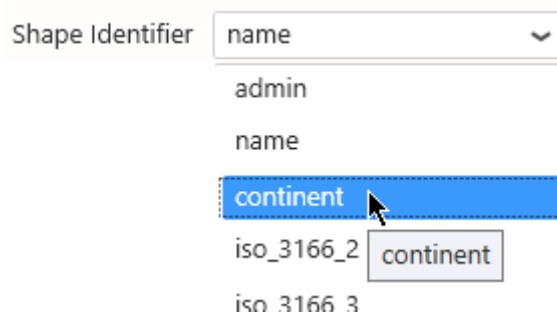
Select a Geometry Type :

- *Geographic* : coordinates are longitude/latitude with a corresponding Mercator projection. This is useful for geographical data like country shapes. Not suitable for viewing very small values of latitude and longitude (eg: 0.0001 degree latitude or longitude)
- *Normal* : coordinates are X/Y with no projection. This is useful for representing a plan, like an equipment shape or a plant floor disposition.



Shape Identifier

When a map file is selected, the GeoJSON *Properties* key names are listed in the “Shape Identifier” combobox in Alpana Designer :



For example, the default map has Properties named “admin”, “name”, “continent”, ...
And each Property has values like name=France, continent=Europe, ...

```
"features": [
{ "type": "Feature", "properties": { "admin": "Afghanistan", "name": "Afghanistan", "continent": "Asia", "iso_3166_2": "AF" },
{ "type": "Feature", "properties": { "admin": "Angola", "name": "Angola", "continent": "Africa", "iso_3166_2": "AO" },
{ "type": "Feature", "properties": { "admin": "Albania", "name": "Albania", "continent": "Europe", "iso_3166_2": "AL" },
{ "type": "Feature", "properties": { "admin": "United Arab Emirates", "name": "United Arab Emirates", "continent": "Asia", "iso_3166_2": "AE" },
{ "type": "Feature", "properties": { "admin": "Argentina", "name": "Argentina", "continent": "South America", "iso_3166_2": "AR" },
{ "type": "Feature", "properties": { "admin": "Armenia", "name": "Armenia", "continent": "Asia", "iso_3166_2": "AM" },
{ "type": "Feature", "properties": { "admin": "French Southern and Antarctic Lands", "name": "Fr. S. Antarctic L.", "continent": "Antarctica", "iso_3166_2": "TF" },
{ "type": "Feature", "properties": { "admin": "Australia", "name": "Australia", "continent": "Oceania", "iso_3166_2": "AU" },
{ "type": "Feature", "properties": { "admin": "Austria", "name": "Austria", "continent": "Europe", "iso_3166_2": "AT" },
{ "type": "Feature", "properties": { "admin": "Azerbaijan", "name": "Azerbaijan", "continent": "Asia", "iso_3166_2": "AZ" },
{ "type": "Feature", "properties": { "admin": "Bahrain", "name": "Bahrain", "continent": "Asia", "iso_3166_2": "BH" },
{ "type": "Feature", "properties": { "admin": "Bangladesh", "name": "Bangladesh", "continent": "Asia", "iso_3166_2": "BD" },
{ "type": "Feature", "properties": { "admin": "Barbados", "name": "Barbados", "continent": "North America", "iso_3166_2": "BB" },
{ "type": "Feature", "properties": { "admin": "Belarus", "name": "Belarus", "continent": "Europe", "iso_3166_2": "BY" },
{ "type": "Feature", "properties": { "admin": "Belgium", "name": "Belgium", "continent": "Europe", "iso_3166_2": "BE" },
{ "type": "Feature", "properties": { "admin": "Belize", "name": "Belize", "continent": "North America", "iso_3166_2": "BZ" },
{ "type": "Feature", "properties": { "admin": "Benin", "name": "Benin", "continent": "Africa", "iso_3166_2": "BJ" },
{ "type": "Feature", "properties": { "admin": "Bhutan", "name": "Bhutan", "continent": "Asia", "iso_3166_2": "BT" },
{ "type": "Feature", "properties": { "admin": "Bolivia", "name": "Bolivia", "continent": "South America", "iso_3166_2": "BO" },
{ "type": "Feature", "properties": { "admin": "Bosnia and Herzegovina", "name": "Bosnia and Herzegovina", "continent": "Europe", "iso_3166_2": "BA" },
{ "type": "Feature", "properties": { "admin": "Botswana", "name": "Botswana", "continent": "Africa", "iso_3166_2": "BW" },
{ "type": "Feature", "properties": { "admin": "Brazil", "name": "Brazil", "continent": "South America", "iso_3166_2": "BR" },
{ "type": "Feature", "properties": { "admin": "Bulgaria", "name": "Bulgaria", "continent": "Europe", "iso_3166_2": "BG" },
{ "type": "Feature", "properties": { "admin": "Burkina Faso", "name": "Burkina Faso", "continent": "Africa", "iso_3166_2": "BF" },
{ "type": "Feature", "properties": { "admin": "Burundi", "name": "Burundi", "continent": "Africa", "iso_3166_2": "BI" },
{ "type": "Feature", "properties": { "admin": "Cambodia", "name": "Cambodia", "continent": "Asia", "iso_3166_2": "KH" },
{ "type": "Feature", "properties": { "admin": "Cameroon", "name": "Cameroon", "continent": "Africa", "iso_3166_2": "CM" },
{ "type": "Feature", "properties": { "admin": "Canada", "name": "Canada", "continent": "North America", "iso_3166_2": "CA" },
{ "type": "Feature", "properties": { "admin": "Cape Verde", "name": "Cape Verde", "continent": "Africa", "iso_3166_2": "CV" },
{ "type": "Feature", "properties": { "admin": "Cayman Islands", "name": "Cayman Islands", "continent": "North America", "iso_3166_2": "KY" },
{ "type": "Feature", "properties": { "admin": "Central African Republic", "name": "Central African Republic", "continent": "Africa", "iso_3166_2": "CF" },
{ "type": "Feature", "properties": { "admin": "Chad", "name": "Chad", "continent": "Africa", "iso_3166_2": "TD" },
{ "type": "Feature", "properties": { "admin": "Chile", "name": "Chile", "continent": "South America", "iso_3166_2": "CL" },
{ "type": "Feature", "properties": { "admin": "China", "name": "China", "continent": "Asia", "iso_3166_2": "CN" },
{ "type": "Feature", "properties": { "admin": "Christmas Island", "name": "Christmas Island", "continent": "Oceania", "iso_3166_2": "CX" },
{ "type": "Feature", "properties": { "admin": "Cocos (Keeling) Islands", "name": "Cocos (Keeling) Islands", "continent": "Oceania", "iso_3166_2": "CC" },
{ "type": "Feature", "properties": { "admin": "Colombia", "name": "Colombia", "continent": "South America", "iso_3166_2": "CO" },
{ "type": "Feature", "properties": { "admin": "Comoros", "name": "Comoros", "continent": "Africa", "iso_3166_2": "KM" },
{ "type": "Feature", "properties": { "admin": "Congo", "name": "Congo", "continent": "Africa", "iso_3166_2": "CG" },
{ "type": "Feature", "properties": { "admin": "Congo (Kinshasa)", "name": "Congo (Kinshasa)", "continent": "Africa", "iso_3166_2": "CD" },
{ "type": "Feature", "properties": { "admin": "Costa Rica", "name": "Costa Rica", "continent": "North America", "iso_3166_2": "CR" },
{ "type": "Feature", "properties": { "admin": "Cote d'Ivoire", "name": "Cote d'Ivoire", "continent": "Africa", "iso_3166_2": "CI" },
{ "type": "Feature", "properties": { "admin": "Croatia", "name": "Croatia", "continent": "Europe", "iso_3166_2": "HR" },
{ "type": "Feature", "properties": { "admin": "Cuba", "name": "Cuba", "continent": "North America", "iso_3166_2": "CU" },
{ "type": "Feature", "properties": { "admin": "Curaçao", "name": "Curaçao", "continent": "North America", "iso_3166_2": "CW" },
{ "type": "Feature", "properties": { "admin": "Cyprus", "name": "Cyprus", "continent": "Europe", "iso_3166_2": "CY" },
{ "type": "Feature", "properties": { "admin": "Czechia", "name": "Czechia", "continent": "Europe", "iso_3166_2": "CZ" },
{ "type": "Feature", "properties": { "admin": "Democratic Republic of the Congo", "name": "Democratic Republic of the Congo", "continent": "Africa", "iso_3166_2": "CD" },
{ "type": "Feature", "properties": { "admin": "Denmark", "name": "Denmark", "continent": "Europe", "iso_3166_2": "DK" },
{ "type": "Feature", "properties": { "admin": "Djibouti", "name": "Djibouti", "continent": "Africa", "iso_3166_2": "DJ" },
{ "type": "Feature", "properties": { "admin": "Dominica", "name": "Dominica", "continent": "North America", "iso_3166_2": "DM" },
{ "type": "Feature", "properties": { "admin": "Dominican Republic", "name": "Dominican Republic", "continent": "North America", "iso_3166_2": "DO" },
{ "type": "Feature", "properties": { "admin": "East Timor", "name": "East Timor", "continent": "Asia", "iso_3166_2": "TL" },
{ "type": "Feature", "properties": { "admin": "Ecuador", "name": "Ecuador", "continent": "South America", "iso_3166_2": "EC" },
{ "type": "Feature", "properties": { "admin": "Egypt", "name": "Egypt", "continent": "Africa", "iso_3166_2": "EG" },
{ "type": "Feature", "properties": { "admin": "El Salvador", "name": "El Salvador", "continent": "North America", "iso_3166_2": "SV" },
{ "type": "Feature", "properties": { "admin": "Equatorial Guinea", "name": "Equatorial Guinea", "continent": "Africa", "iso_3166_2": "GQ" },
{ "type": "Feature", "properties": { "admin": "Eritrea", "name": "Eritrea", "continent": "Africa", "iso_3166_2": "ER" },
{ "type": "Feature", "properties": { "admin": "Estonia", "name": "Estonia", "continent": "Europe", "iso_3166_2": "EE" },
{ "type": "Feature", "properties": { "admin": "Ethiopia", "name": "Ethiopia", "continent": "Africa", "iso_3166_2": "ET" },
{ "type": "Feature", "properties": { "admin": "Fiji", "name": "Fiji", "continent": "Oceania", "iso_3166_2": "FJ" },
{ "type": "Feature", "properties": { "admin": "Finland", "name": "Finland", "continent": "Europe", "iso_3166_2": "FI" },
{ "type": "Feature", "properties": { "admin": "France", "name": "France", "continent": "Europe", "iso_3166_2": "FR" },
{ "type": "Feature", "properties": { "admin": "French Polynesia", "name": "French Polynesia", "continent": "Oceania", "iso_3166_2": "PF" },
{ "type": "Feature", "properties": { "admin": "Gabon", "name": "Gabon", "continent": "Africa", "iso_3166_2": "GA" },
{ "type": "Feature", "properties": { "admin": "Gambia", "name": "Gambia", "continent": "Africa", "iso_3166_2": "GM" },
{ "type": "Feature", "properties": { "admin": "Georgia", "name": "Georgia", "continent": "Asia", "iso_3166_2": "GE" },
{ "type": "Feature", "properties": { "admin": "Germany", "name": "Germany", "continent": "Europe", "iso_3166_2": "DE" },
{ "type": "Feature", "properties": { "admin": "Ghana", "name": "Ghana", "continent": "Africa", "iso_3166_2": "GH" },
{ "type": "Feature", "properties": { "admin": "Greece", "name": "Greece", "continent": "Europe", "iso_3166_2": "GR" },
{ "type": "Feature", "properties": { "admin": "Greenland", "name": "Greenland", "continent": "North America", "iso_3166_2": "GL" },
{ "type": "Feature", "properties": { "admin": "Guatemala", "name": "Guatemala", "continent": "North America", "iso_3166_2": "GT" },
{ "type": "Feature", "properties": { "admin": "Guinea", "name": "Guinea", "continent": "Africa", "iso_3166_2": "GN" },
{ "type": "Feature", "properties": { "admin": "Guinea-Bissau", "name": "Guinea-Bissau", "continent": "Africa", "iso_3166_2": "GW" },
{ "type": "Feature", "properties": { "admin": "Guyana", "name": "Guyana", "continent": "South America", "iso_3166_2": "GY" },
{ "type": "Feature", "properties": { "admin": "Haiti", "name": "Haiti", "continent": "North America", "iso_3166_2": "HT" },
{ "type": "Feature", "properties": { "admin": "Honduras", "name": "Honduras", "continent": "North America", "iso_3166_2": "HN" },
{ "type": "Feature", "properties": { "admin": "Hungary", "name": "Hungary", "continent": "Europe", "iso_3166_2": "HU" },
{ "type": "Feature", "properties": { "admin": "Iceland", "name": "Iceland", "continent": "Europe", "iso_3166_2": "IS" },
{ "type": "Feature", "properties": { "admin": "India", "name": "India", "continent": "Asia", "iso_3166_2": "IN" },
{ "type": "Feature", "properties": { "admin": "Indonesia", "name": "Indonesia", "continent": "Asia", "iso_3166_2": "ID" },
{ "type": "Feature", "properties": { "admin": "Iran", "name": "Iran", "continent": "Asia", "iso_3166_2": "IR" },
{ "type": "Feature", "properties": { "admin": "Iraq", "name": "Iraq", "continent": "Asia", "iso_3166_2": "IQ" },
{ "type": "Feature", "properties": { "admin": "Ireland", "name": "Ireland", "continent": "Europe", "iso_3166_2": "IE" },
{ "type": "Feature", "properties": { "admin": "Israel", "name": "Israel", "continent": "Asia", "iso_3166_2": "IL" },
{ "type": "Feature", "properties": { "admin": "Italy", "name": "Italy", "continent": "Europe", "iso_3166_2": "IT" },
{ "type": "Feature", "properties": { "admin": "Jamaica", "name": "Jamaica", "continent": "North America", "iso_3166_2": "JM" },
{ "type": "Feature", "properties": { "admin": "Japan", "name": "Japan", "continent": "Asia", "iso_3166_2": "JP" },
{ "type": "Feature", "properties": { "admin": "Jordan", "name": "Jordan", "continent": "Asia", "iso_3166_2": "JO" },
{ "type": "Feature", "properties": { "admin": "Kazakhstan", "name": "Kazakhstan", "continent": "Asia", "iso_3166_2": "KZ" },
{ "type": "Feature", "properties": { "admin": "Kenya", "name": "Kenya", "continent": "Africa", "iso_3166_2": "KE" },
{ "type": "Feature", "properties": { "admin": "Kiribati", "name": "Kiribati", "continent": "Oceania", "iso_3166_2": "KI" },
{ "type": "Feature", "properties": { "admin": "Korea", "name": "Korea", "continent": "Asia", "iso_3166_2": "KR" },
{ "type": "Feature", "properties": { "admin": "Kuwait", "name": "Kuwait", "continent": "Asia", "iso_3166_2": "KW" },
{ "type": "Feature", "properties": { "admin": "Kyrgyzstan", "name": "Kyrgyzstan", "continent": "Asia", "iso_3166_2": "KG" },
{ "type": "Feature", "properties": { "admin": "Laos", "name": "Laos", "continent": "Asia", "iso_3166_2": "LA" },
{ "type": "Feature", "properties": { "admin": "Latvia", "name": "Latvia", "continent": "Europe", "iso_3166_2": "LV" },
{ "type": "Feature", "properties": { "admin": "Lebanon", "name": "Lebanon", "continent": "Asia", "iso_3166_2": "LB" },
{ "type": "Feature", "properties": { "admin": "Lesotho", "name": "Lesotho", "continent": "Africa", "iso_3166_2": "LS" },
{ "type": "Feature", "properties": { "admin": "Liberia", "name": "Liberia", "continent": "Africa", "iso_3166_2": "LR" },
{ "type": "Feature", "properties": { "admin": "Liechtenstein", "name": "Liechtenstein", "continent": "Europe", "iso_3166_2": "LI" },
{ "type": "Feature", "properties": { "admin": "Lithuania", "name": "Lithuania", "continent": "Europe", "iso_3166_2": "LT" },
{ "type": "Feature", "properties": { "admin": "Luxembourg", "name": "Luxembourg", "continent": "Europe", "iso_3166_2": "LU" },
{ "type": "Feature", "properties": { "admin": "Madagascar", "name": "Madagascar", "continent": "Africa", "iso_3166_2": "MG" },
{ "type": "Feature", "properties": { "admin": "Malawi", "name": "Malawi", "continent": "Africa", "iso_3166_2": "MW" },
{ "type": "Feature", "properties": { "admin": "Malaysia", "name": "Malaysia", "continent": "Asia", "iso_3166_2": "MY" },
{ "type": "Feature", "properties": { "admin": "Maldives", "name": "Maldives", "continent": "Asia", "iso_3166_2": "MV" },
{ "type": "Feature", "properties": { "admin": "Mali", "name": "Mali", "continent": "Africa", "iso_3166_2": "ML" },
{ "type": "Feature", "properties": { "admin": "Malta", "name": "Malta", "continent": "Europe", "iso_3166_2": "MT" },
{ "type": "Feature", "properties": { "admin": "Marshall Islands", "name": "Marshall Islands", "continent": "Oceania", "iso_3166_2": "MH" },
{ "type": "Feature", "properties": { "admin": "Mauritania", "name": "Mauritania", "continent": "Africa", "iso_3166_2": "MR" },
{ "type": "Feature", "properties": { "admin": "Mauritius", "name": "Mauritius", "continent": "Africa", "iso_3166_2": "MU" },
{ "type": "Feature", "properties": { "admin": "Mexico", "name": "Mexico", "continent": "North America", "iso_3166_2": "MX" },
{ "type": "Feature", "properties": { "admin": "Micronesia", "name": "Micronesia", "continent": "Oceania", "iso_3166_2": "FM" },
{ "type": "Feature", "properties": { "admin": "Moldova", "name": "Moldova", "continent": "Europe", "iso_3166_2": "MD" },
{ "type": "Feature", "properties": { "admin": "Monaco", "name": "Monaco", "continent": "Europe", "iso_3166_2": "MC" },
{ "type": "Feature", "properties": { "admin": "Mongolia", "name": "Mongolia", "continent": "Asia", "iso_3166_2": "MN" },
{ "type": "Feature", "properties": { "admin": "Montenegro", "name": "Montenegro", "continent": "Europe", "iso_3166_2": "ME" },
{ "type": "Feature", "properties": { "admin": "Morocco", "name": "Morocco", "continent": "Africa", "iso_3166_2": "MA" },
{ "type": "Feature", "properties": { "admin": "Mozambique", "name": "Mozambique", "continent": "Africa", "iso_3166_2": "MZ" },
{ "type": "Feature", "properties": { "admin": "Myanmar", "name": "Myanmar", "continent": "Asia", "iso_3166_2": "MM" },
{ "type": "Feature", "properties": { "admin": "Namibia", "name": "Namibia", "continent": "Africa", "iso_3166_2": "NA" },
{ "type": "Feature", "properties": { "admin": "Nauru", "name": "Nauru", "continent": "Oceania", "iso_3166_2": "NR" },
{ "type": "Feature", "properties": { "admin": "Nepal", "name": "Nepal", "continent": "Asia", "iso_3166_2": "NP" },
{ "type": "Feature", "properties": { "admin": "Netherlands", "name": "Netherlands", "continent": "Europe", "iso_3166_2": "NL" },
{ "type": "Feature", "properties": { "admin": "New Zealand", "name": "New Zealand", "continent": "Oceania", "iso_3166_2": "NZ" },
{ "type": "Feature", "properties": { "admin": "Nicaragua", "name": "Nicaragua", "continent": "North America", "iso_3166_2": "NI" },
{ "type": "Feature", "properties": { "admin": "Niger", "name": "Niger", "continent": "Africa", "iso_3166_2": "NE" },
{ "type": "Feature", "properties": { "admin": "Nigeria", "name": "Nigeria", "continent": "Africa", "iso_3166_2": "NG" },
{ "type": "Feature", "properties": { "admin": "North Macedonia", "name": "North Macedonia", "continent": "Europe", "iso_3166_2": "MK" },
{ "type": "Feature", "properties": { "admin": "Norway", "name": "Norway", "continent": "Europe", "iso_3166_2": "NO" },
{ "type": "Feature", "properties": { "admin": "Oman", "name": "Oman", "continent": "Asia", "iso_3166_2": "OM" },
{ "type": "Feature", "properties": { "admin": "Pakistan", "name": "Pakistan", "continent": "Asia", "iso_3166_2": "PK" },
{ "type": "Feature", "properties": { "admin": "Palau", "name": "Palau", "continent": "Oceania", "iso_3166_2": "PW" },
{ "type": "Feature", "properties": { "admin": "Palestine", "name": "Palestine", "continent": "Asia", "iso_3166_2": "PS" },
{ "type": "Feature", "properties": { "admin": "Panama", "name": "Panama", "continent": "North America", "iso_3166_2": "PA" },
{ "type": "Feature", "properties": { "admin": "Papua New Guinea", "name": "Papua New Guinea", "continent": "Oceania", "iso_3166_2": "PG" },
{ "type": "Feature", "properties": { "admin": "Paraguay", "name": "Paraguay", "continent": "South America", "iso_3166_2": "PY" },
{ "type": "Feature", "properties": { "admin": "Peru", "name": "Peru", "continent": "South America", "iso_3166_2": "PE" },
{ "type": "Feature", "properties": { "admin": "Philippines", "name": "Philippines", "continent": "Asia", "iso_3166_2": "PH" },
{ "type": "Feature", "properties": { "admin": "Poland", "name": "Poland", "continent": "Europe", "iso_3166_2": "PL" },
{ "type": "Feature", "properties": { "admin": "Portugal", "name": "Portugal", "continent": "Europe", "iso_3166_2": "PT" },
{ "type": "Feature", "properties": { "admin": "Puerto Rico", "name": "Puerto Rico", "continent": "North America", "iso_3166_2": "PR" },
{ "type": "Feature", "properties": { "admin": "Qatar", "name": "Qatar", "continent": "Asia", "iso_3166_2": "QA" },
{ "type": "Feature", "properties": { "admin": "Romania", "name": "Romania", "continent": "Europe", "iso_3166_2": "RO" },
{ "type": "Feature", "properties": { "admin": "Russia", "name": "Russia", "continent": "Europe", "iso_3166_2": "RU" },
{ "type": "Feature", "properties": { "admin": "Rwanda", "name": "Rwanda", "continent": "Africa", "iso_3166_2": "RW" },
{ "type": "Feature", "properties": { "admin": "Saint Kitts and Nevis", "name": "Saint Kitts and Nevis", "continent": "North America", "iso_3166_2": "KN" },
{ "type": "Feature", "properties": { "admin": "Saint Lucia", "name": "Saint Lucia", "continent": "North America", "iso_3166_2": "LC" },
{ "type": "Feature", "properties": { "admin": "Saint Vincent and the Grenadines", "name": "Saint Vincent and the Grenadines", "continent": "North America", "iso_3166_2": "VG" },
{ "type": "Feature", "properties": { "admin": "Samoa", "name": "Samoa", "continent": "Oceania", "iso_3166_2": "WS" },
{ "type": "Feature", "properties": { "admin": "San Marino", "name": "San Marino", "continent": "Europe", "iso_3166_2": "SM" },
{ "type": "Feature", "properties": { "admin": "Saudi Arabia", "name": "Saudi Arabia", "continent": "Asia", "iso_3166_2": "SA" },
{ "type": "Feature", "properties": { "admin": "Senegal", "name": "Senegal", "continent": "Africa", "iso_3166_2": "SN" },
{ "type": "Feature", "properties": { "admin": "Serbia", "name": "Serbia", "continent": "Europe", "iso_3166_2": "RS" },
{ "type": "Feature", "properties": { "admin": "Sierra Leone", "name": "Sierra Leone", "continent": "Africa", "iso_3166_2": "SL" },
{ "type": "Feature", "properties": { "admin": "Singapore", "name": "Singapore", "continent": "Asia", "iso_3166_2": "SG" },
{ "type": "Feature", "properties": { "admin": "Slovakia", "name": "Slovakia", "continent": "Europe", "iso_3166_2": "SK" },
{ "type": "Feature", "properties": { "admin": "Slovenia", "name": "Slovenia", "continent": "Europe", "iso_3166_2": "SI" },
{ "type": "Feature", "properties": { "admin": "Solomon Islands", "name": "Solomon Islands", "continent": "Oceania", "iso_3166_2": "SB" },
{ "type": "Feature", "properties": { "admin": "Somalia", "name": "Somalia", "continent": "Africa", "iso_3166_2": "SO" },
{ "type": "Feature", "properties": { "admin": "South Africa", "name": "South Africa", "continent": "Africa", "iso_3166_2": "ZA" },
{ "type": "Feature", "properties": { "admin": "South Korea", "name": "South Korea", "continent": "Asia", "iso_3166_2": "KR" },
{ "type": "Feature", "properties": { "admin": "South Sudan", "name": "South Sudan", "continent": "Africa", "iso_3166_2": "SS" },
{ "type": "Feature", "properties": { "admin": "Spain", "name": "Spain", "continent": "Europe", "iso_3166_2": "ES" },
{ "type": "Feature", "properties": { "admin": "Sri Lanka", "name": "Sri Lanka", "continent": "Asia", "iso_3166_2": "LK" },
{ "type": "Feature", "properties": { "admin": "Sudan", "name": "Sudan", "continent": "Africa", "iso_3166_2": "SD" },
{ "type": "Feature", "properties": { "admin": "Suriname", "name": "Suriname", "continent": "South America", "iso_3166_2": "SR" },
{ "type": "Feature", "properties": { "admin": "Swaziland", "name": "Swaziland", "continent": "Africa", "iso_3166_2": "SZ" },
{ "type": "Feature", "properties": { "admin": "Sweden", "name": "Sweden", "continent": "Europe", "iso_3166_2": "SE" },
{ "type": "Feature", "properties": { "admin": "Switzerland", "name": "Switzerland", "continent": "Europe", "iso_3166_2": "CH" },
{ "type": "Feature", "properties": { "admin": "Taiwan", "name": "Taiwan", "continent": "Asia", "iso_3166_2": "TW" },
{ "type": "Feature", "properties": { "admin": "Tajikistan", "name": "Tajikistan", "continent": "Asia", "iso_3166_2": "TJ" },
{ "type": "Feature", "properties": { "admin": "Tanzania", "name": "Tanzania", "continent": "Africa", "iso_3166_2": "TZ" },
{ "type": "Feature", "properties": { "admin": "Thailand", "name": "Thailand", "continent": "Asia", "iso_3166_2": "TH" },
{ "type": "Feature", "properties": { "admin": "Timor-Leste", "name": "Timor-Leste", "continent": "Asia", "iso_3166_2": "TL" },
{ "type": "Feature", "properties": { "admin": "Togo", "name": "Togo", "continent": "Africa", "iso_3166_2": "TG" },
{ "type": "Feature", "properties": { "admin": "Tonga", "name": "Tonga", "continent": "Oceania", "iso_3166_2": "TO" },
{ "type": "Feature", "properties": { "admin": "Trinidad and Tobago", "name": "Trinidad and Tobago", "continent": "North America", "iso_3166_2": "TT" },
{ "type": "Feature", "properties": { "admin": "Tunisia", "name": "Tunisia", "continent": "Africa", "iso_3166_2": "TN" },
{ "type": "Feature", "properties": { "admin": "Turkey", "name": "Turkey", "continent": "Europe", "iso_3166_2": "TR" },
{ "type": "Feature", "properties": { "admin": "Turkmenistan", "name": "Turkmenistan", "continent": "Asia", "iso_3166_2": "TM" },
{ "type": "Feature", "properties": { "admin": "Tuvalu", "name": "Tuvalu", "continent": "Oceania", "iso_3166_2": "TV" },
{ "type": "Feature", "properties": { "admin": "Uganda", "name": "Uganda", "continent": "Africa", "iso_3166_2": "UG" },
{ "type": "Feature", "properties": { "admin": "Ukraine", "name": "Ukraine", "continent": "Europe", "iso_3166_2": "UA" },
{ "type": "Feature", "properties": { "admin": "United Kingdom", "name": "United Kingdom", "continent": "Europe", "iso_3166_2": "GB" },
{ "type": "Feature", "properties": { "admin": "United States of America", "name": "United States of America", "continent": "North America", "iso_3166_2": "US" },
{ "type": "Feature", "properties": { "admin": "Uruguay", "name": "Uruguay", "continent": "South America", "iso_3166_2": "UY" },
{ "type": "Feature", "properties": { "admin": "Uzbekistan", "name": "Uzbekistan", "continent": "Asia", "iso_3166_2": "UZ" },
{ "type": "Feature", "properties": { "admin": "Vanuatu", "name": "Vanuatu", "continent": "Oceania", "iso_3166_2": "VU" },
{ "type": "Feature", "properties": { "admin": "Vatican City", "name": "Vatican City", "continent": "Europe", "iso_3166_2": "VA" },
{ "type": "Feature", "properties": { "admin": "Venezuela", "name": "Venezuela", "continent": "South America", "iso_3166_2": "VE" },
{ "type": "Feature", "properties": { "admin": "Vietnam", "name": "Vietnam", "continent": "Asia", "iso_3166_2": "VN" },
{ "type": "Feature", "properties": { "admin": "Yemen", "name": "Yemen", "continent": "Asia", "iso_3166_2": "YE" },
{ "type": "Feature", "properties": { "admin": "Zambia", "name": "Zambia", "continent": "Africa", "iso_3166_2": "ZM" },
{ "type": "Feature", "properties": { "admin": "Zimbabwe", "name": "Zimbabwe", "continent": "Africa", "iso_3166_2": "ZW" }
]
```



Note :

This is specific to this map file and will be different in other files.



This setting is **very important** since it defines which GeoJSON Property will be bound to

your data.

The values in the selected *Shape Identifier* **must match** the values in your Data Source.

For this, you have basically 2 solutions :

- edit the GeoJSON file so that there is a Property that possesses values that match your data
- edit your Data Source so that the values match the values of a map Property

Last modified: 2019/05/14

2.7.3.3. Binding Data

Introduction

The following field containers are available for configuring data binding on a Map :

The 'Show' dialog box displays three configuration sections:

- Value:** Aggregation function 'Sum', field 'Production', sort order 'Asc'.
- Shape:** Aggregation function 'Value', field 'name', sort order 'Asc'.
- Calculations:** Aggregation function 'Value', field 'continent', sort order 'Asc'.

- *Value* : the color intensity of the shape (Choropleth Map), or the size of the bubble (Bubble Map)
- *Shape* : what Data Source field will be bound to the selected GeoJSON Property/Shape Identifier
- *Calculations* : (Choropleth Map only) assign a single color to a shape or group of shapes

Value

This represents the quantity we want to display on the Map, and will bind for example to the color intensity of the shape (Choropleth Map), or the size of the bubble (Bubble Map).

The Value field **must be aggregated** : you must choose an aggregation function (SUM, AVG, ...) This field is required.

Shape

This will link the values from the selected Data Source field to the values of the selected GeoJSON Property/Shape Identifier.



This is very important : this is how Alpana knows which map shape is associated to which data field.

The Shape field **cannot be aggregated** : it must have several separate values in order to group the “Value” aggregation.

This field is required.

Calculations (Choropleth Map)

When this field is configured, the Choropleth Map colors no longer show gradients of a single color depending on the intensity of the Value.

Instead, each separate value of the Calculations field generates a single separate color, and all corresponding shapes are painted with that color.

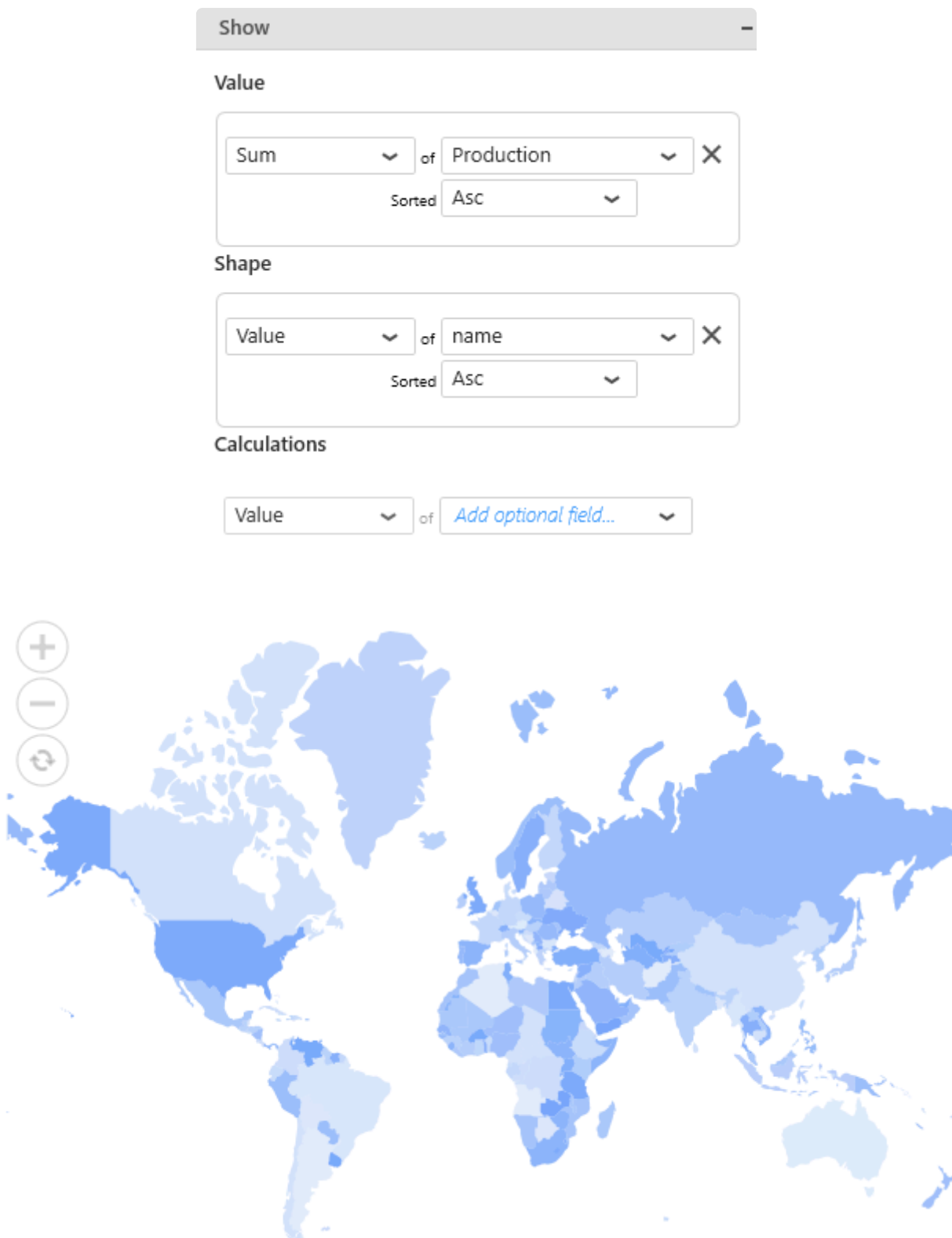
Then the “Value” field is only displayed as indication in the tooltip.

The *Calculations* field **cannot be aggregated** : it must have several separate values in order to group the “Value” aggregation.

This field is optional.

Example

Before applying a calculation field, all countries have a different shade of blue :



After using “continent” as Calculations field, each country have the color of the corresponding continent :

Value

Sum

▼

of

Production

▼

×

Sorted

ASC

▼

Shape

Value

▼

of

name

▼

×

Sorted

ASC

▼

Calculations

Value

▼

of

continent

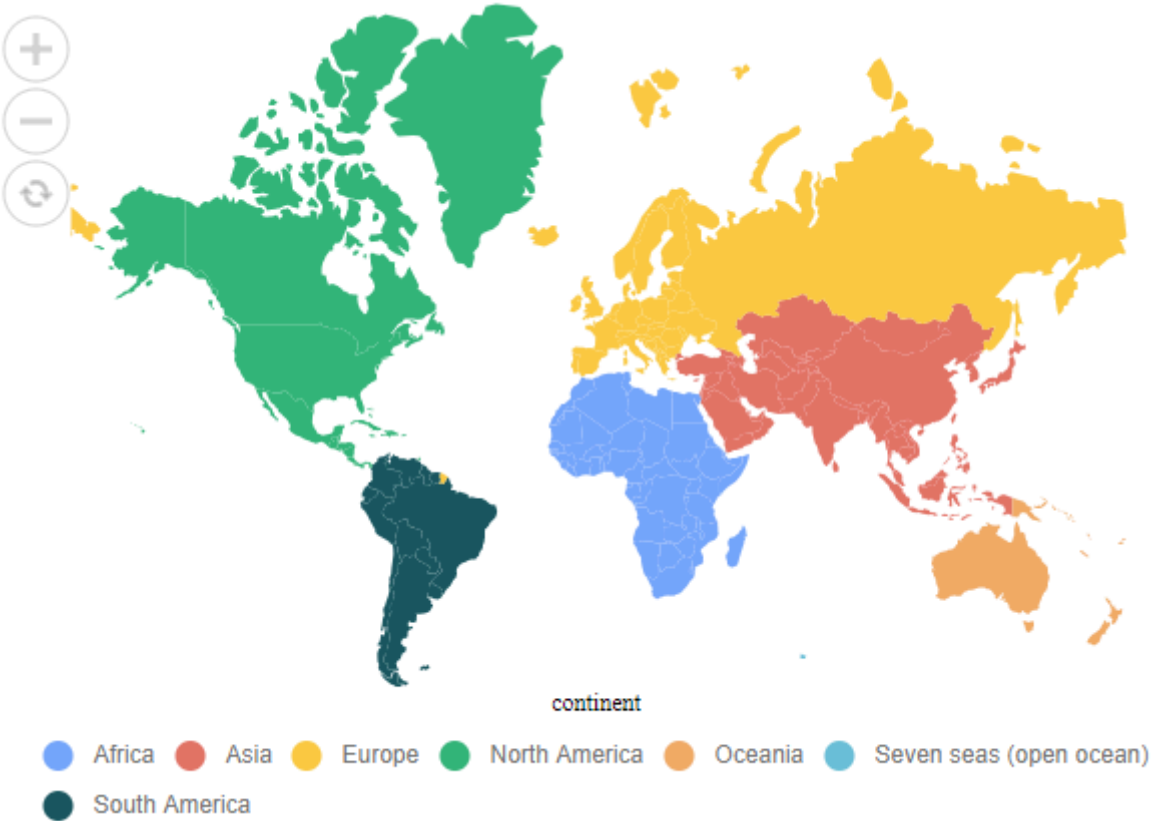
▼

×

Sorted

Asc

▼



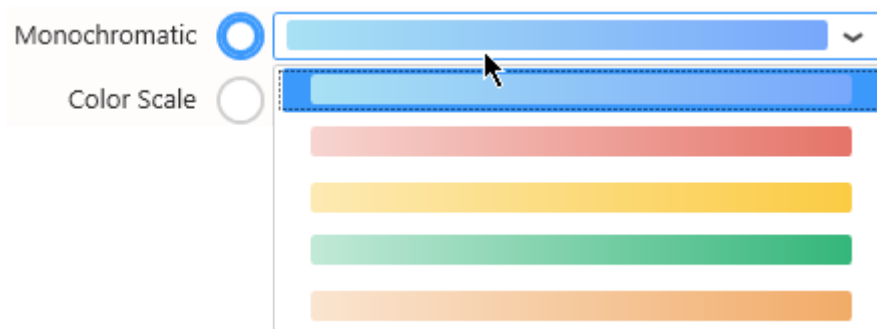
Last modified: 2019/05/14

2.7.3.4. Managing Colors

Monochromatic (Choropleth Map)

By default, Choropleth Map with **no Calculations** show value intensity as a gradient of a single color.

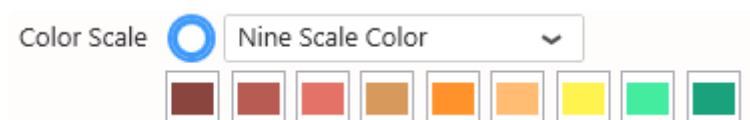
This *Monochromatic* Gradient can be set to a different color by using the dedicated combobox under *Map Settings* :



Color Scale (Choropleth Map)

Choropleth Map with no Calculations can be configured to show value intensity in ranges of single colors.

Select Color Scale and choose the number of colors and the colors of each interval :



The interval between the maximum of Value and the minimum of Value will be split in equal intervals and assigned a single color.




This can be useful for setting a color code like “low is red, medium is orange, high is green”.

Value-based Colors (Choropleth Map)

Choropleth Map **with Calculations** show a single different color for each different value of the

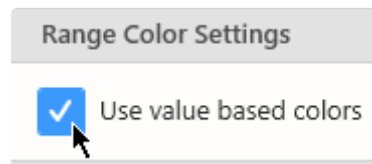
Calculations field.

By default, these colors are indexed from a default palette.

Color 1		#77A7FB
Color 2		#E57368
Color 3		#FBCB43
Color 4		#34B67A
Color 5		#F1AB68
Color 6		#6CC2D9
Color 7		#195962
Color 8		#E18878
Color 9		#1AA27D
Color 10		#BFAAC7
Color 11		#89695E
Color 12		#A5C77F
Color 13		#F48FB1
Color 14		#90CAF9
Color 15		#CC4452

Instead of index-based, you may then choose **Value-based** colors : the same value in the Calculations field will always have the same color in the Map.

For this, simply check *Use value-based colors* under *Range Color Settings* :



Customizing Value-based colors

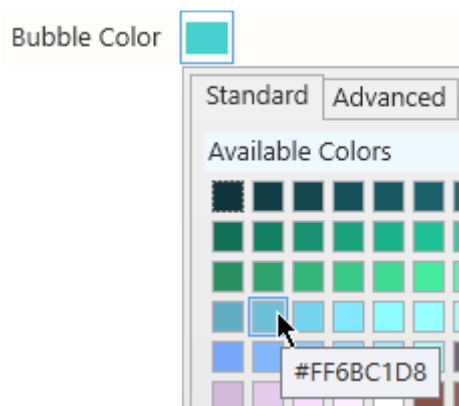
Value-based colors are defined **at the Data Source level**.

This is in order to allow re-use of the colors everywhere the same data is used.

To configure value-based colors, see the corresponding chapter [Value-based Settings](#) under [Transforming Data](#).

Bubble Color (Bubble Map)

Bubbles have a single color defined by *Bubble Color* under *Basic Settings* :



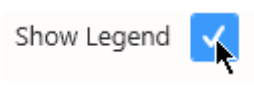
Last modified: 2019/05/14

2.7.3.5. Data Point information

Legend (Choropleth)

Depending on the color mode selected, the Choropleth Map can display a legend.

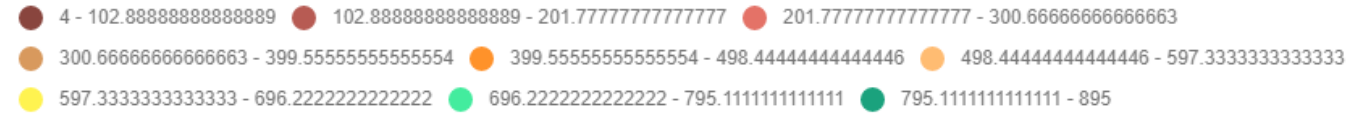
Click on *Show Legend* under *Basic Settings* :



In Monochromatic color, a gradient is displayed with the minimum and maximum values and an arrow for the current value :



In Color Scale mode, the scale is displayed with the minimum and maximum values and an arrow for the current value :



In Calculations mode, the legend displays the name and color of each item :



Last modified: 2019/05/14

2.7.3.6. Runtime navigation

At runtime, the maps can be navigated by the user :

- zoom : mouse scroll allows to zoom in/out
- move : click and drag to move

Navigator

In addition, a Navigator can be displayed to help users navigate using mouse clicks.

Under *Basic Settings*, select *Enable Navigator* :

Enable Navigator 

At runtime, 3 icons are displayed at the top left of the map :



-  : resets zoom level
-  : zoom in
-  : zoom out

Last modified: 2019/05/14

2.7.4. Grid



The Grid is a very powerful and simple widget.
It allows to display tabular data in many ways.

Last modified: 2019/05/07

2.7.4.1. Binding Data

Introduction

The following field containers are available for configuring data binding on a Grid :

Show

Column(s)

Value ▾ of Date ▾ X

Formatted as Quarter ▾

Sorted Asc ▾

And

Value ▾ of [Add optional field...](#) ▾

Hidden Columns(s)

Value ▾ of Date ▾ X

Formatted as Month ▾

Sorted Asc ▾

And

Value ▾ of [Add optional field...](#) ▾

- *Column(s)* : fields that will be Grid columns
- *Hidden Column(s)* : field that will be in the query but not visible to the user

Columns

Any number of fields of any type can be added as Grid Columns.

The order in the Columns container defines the order of Grid columns from left to right.

Value fields may or may not be aggregated.

Each field groups all the other fields on the right of it.
Changing the order of fields is documented [here](#).

Example

With fields Year, Continent :

Show

Column(s)

Value

of

Date

X

Formatted as

Year

Sorted

Asc

And

Value

of

continent

X

Sorted

Asc

The Grid shows the first Year (2016) and all its Continents,(Asia, Europe, North America), then the second Year (2017) and all its Continents, etc... :

Year(Date)	continent
2015	Asia
2015	Europe
2015	North America
2016	Asia
2016	Europe
2016	North America
2017	Asia
2017	Europe
2017	North America

With fields Continent, Year :

Show

Column(s)

Value
of
continent
X

Sorted
Asc

And

Value
of
Date
X

Formatted as
Year

Sorted
Asc

The Grid shows the first Continent (Asia) and all its Years (2015, 2016, 2017), then the second Continent (Europe) and all its Years, etc... :

continent	Year(Date)	
Asia	2015	▲
Asia	2016	
Asia	2017	
Europe	2015	
Europe	2016	
Europe	2017	
North America	2015	
North America	2016	
North America	2017	▼

Hidden Columns

Optionally, any number of fields of any type can be added as *Hidden Columns*.



Hints :

This can be useful for setting up conditional formatting based on calculations that the user doesn't need to see.

This can also be useful for grouping fields by a field where the values don't have to be shown to the user.

The Hidden columns group the visible columns.

Value fields may or may not be aggregated.

Example

When configuring a single column called Year :

Show

Column(s)

Value

of

Date

X

Formatted as

Year

Sorted

Asc

... the data is grouped, and only distinct values of Year (2015, 2016, 2017) are displayed, and only once :

Year(Date)
2015
2016
2017

When configuring additionally a hidden field Continent which has 3 distinct values :

Show

Column(s)

Value

of

Date

X

Formatted as

Year

Sorted

Asc

And

Value

Add optional field...

Hidden Columns(s)

Value

of

continent

X

Sorted

Asc

... each Year will appear 3 times : once for each value in Continent :

Year(Date)	
2015	▲
2015	
2015	
2016	
2016	
2016	
2017	
2017	
2017	▼

Last modified: 2019/05/07

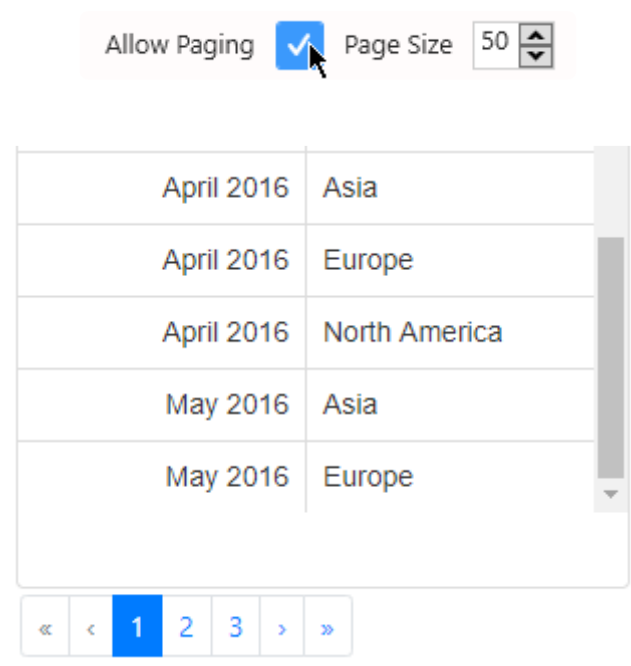
2.7.4.2. Layout and appearance

Paging

When loading a Grid Widget, all the rows of data are loaded.

The Grid can also be configured with *Paging* : it only loads pages of 50 rows (configurable), and a page navigator allows to see the rest of the data.



Check *Allow Paging* under *Basic Settings* and set a *Page Size* (default 50) :



The User can still scroll down each page (depending on the widget height), but not more than 50 rows at a time.

Then the navigator can be used to go to :



-  first page
-  previous page

- 1

2

3

 a specific page number
- >

 next page
- >>

 last page

Column width

Grid column width is by default constant, but can be configured to adapt to the content width.

Year	Country_Full
2015	Asia - China
2015	Asia - India
2015	Europe - Denmark
2015	Europe - France
2015	North America - C...
2015	North America - U...

Check *Fit To Content* under *Basic Settings* :

Fit To Content ☒

Year	Country_Full
2015	Asia - China
2015	Asia - India
2015	Europe - Denmark
2015	Europe - France
2015	North America - Canada
2015	North America - United States

Grid Lines

Grid lines (columns/rows separators) can be displayed or hidden.

Check the corresponding *Horizontal*__*Vertical* option :

Horizontal Grid Line

☐

Vertical Grid Line

☒

Month Year(Date)	continent
January 2015	Asia
January 2015	Europe
January 2015	North America

Horizontal Grid Line

☒

Vertical Grid Line

☐

Month Year(Date)	continent
January 2015	Asia
January 2015	Europe
January 2015	North America

Last modified: 2019/05/13

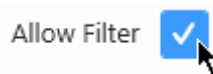
2.7.4.3. User interaction

Manual Filtering

Users can be allowed to filter column contents at runtime based on a text value.

Configuring Manual Filter

In the *Basic Settings* section, select *Allow Filter* :



Applying Manual Filter

At runtime, a text box appears under each column name, allowing users to filter by this column content :

ite)	continent	FactoryName	Sum(Pr
		san I ✕	
2015	North America	San Francisco	
2016	North America	San Francisco	
2017	North America	San Francisco	

Only values that **start with** the input text is filtered.

! If paging is enabled, only the current page is filtered. To see the result across all pages, navigate to the corresponding page.

Clearing Manual Filter

To clear the filter, the User can click on the cross of that text box :

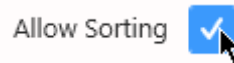


Manual Sorting

When loading a Grid widget, sorting order depends on data binding and sorting configuration.

However, when manual sorting is allowed, the User can click on column titles to manually sort the Grid on that column :

To enable this feature (enabled by default), un-check *Allow Sorting* under *Basic Settings* :



Single sorting

Clicking the column header will cycle between 3 states :

- not sorted
- sorted ascending
- sorted descending

Example


Here is the default sorting : Year ascending, continent ascending, (Production ascending) :

Year(Date)	continent	Sum(Production)
2015	Asia	59.86M
2015	Europe	11.75M
2015	North America	38.80M
2016	Asia	60.21M
2016	Europe	11.71M
2016	North America	38.72M
2017	Asia	59.99M
2017	Europe	11.70M
2017	North America	38.65M


When the User clicks on continent, the Grid is now sorted by continent in ascending alphabetical order :

Year(Date)	continent	↑	Sum(Production)
2015	Asia		59.86M
2016	Asia		60.21M
2017	Asia		59.99M
2015	Europe		11.75M
2016	Europe		11.71M
2017	Europe		11.70M
2015	North America		38.80M
2016	North America		38.72M
2017	North America		38.65M

When clicking a second time, the column is now sorted descending :

Year(Date)	continent 	↓	Sum(Production)	
2015	North America		38.80M	▲
2016	North America		38.72M	
2017	North America		38.65M	
2015	Europe		11.75M	
2016	Europe		11.71M	
2017	Europe		11.70M	
2015	Asia		59.86M	
2016	Asia		60.21M	
2017	Asia		59.99M	
				▼

Number fields can also be sorted manually :

Year(Date)	continent	↓ 	Sum(Production)
2016	Asia		60.21M
2017	Asia		59.99M
2015	Asia		59.86M
2015	North America		38.80M
2016	North America		38.72M
2017	North America		38.65M
2015	Europe		11.75M
2016	Europe		11.71M
2017	Europe		11.70M

Multiple sorting

Using the `Ctrl` keyboard key and clicking another column header (`Ctrl+click`) will combine the sorting clauses in the correct order.

A number icon indicates the order of the sorting clauses.

Example

1. Clicking on UP twice sorts it in descending order,
2. `Ctrl+click` on Count sorts by “UP descending” first, then “Count ascending”
3. `Ctrl+click` on Line twice sorts by “UP descending” first, then “Count ascending”, then “Line descending”

UP 1 ↓	Line 3 ↓	↑ 2 Count
Frarance 2	Line15	4

Last modified: 2019/08/22

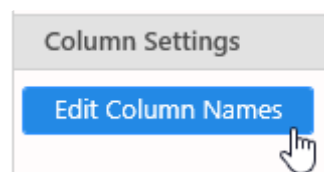
2.7.4.4. Column names

Edit Column Names

Columns are named automatically based on the corresponding field name and aggregation function.

These default names can be replaced by custom name for any column.

Under *Column Settings*, click *Edit Column Names* :

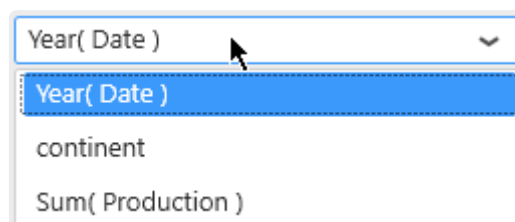


Rename column

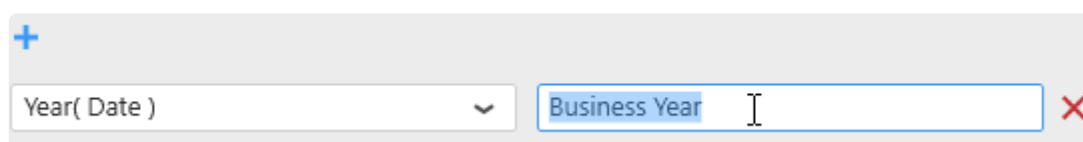
In the Edit Column Names dialog, click the **+** icon to add a column name customization :



From the new customization item, select a column name to rename :



Write a new name for the column :



Reset column name

To reset a column name to its original (auto-generated) name, click the red cross on the right of the corresponding item :



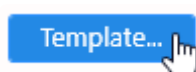
Last modified: 2019/05/07

2.7.4.5. Value display

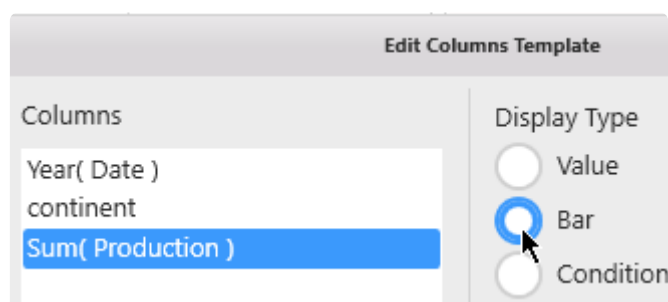
Bar graph

Value fields can be displayed as horizontal bar graphs.

For this, click *Template* under *Column Settings* :



In the Edit Columns Template dialog, select a **numerical** column from the left, and change its display type from Value (default) to Bar :



The value is now displayed as a green horizontal bar in each cell of the column :

continent	Total Production	Total Production ...
Asia	60M	<div></div>
Europe	12M	
North America	39M	<div></div>
Asia	60M	<div></div>

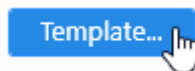
Last modified: 2019/05/07

2.7.4.6. Conditional Formatting

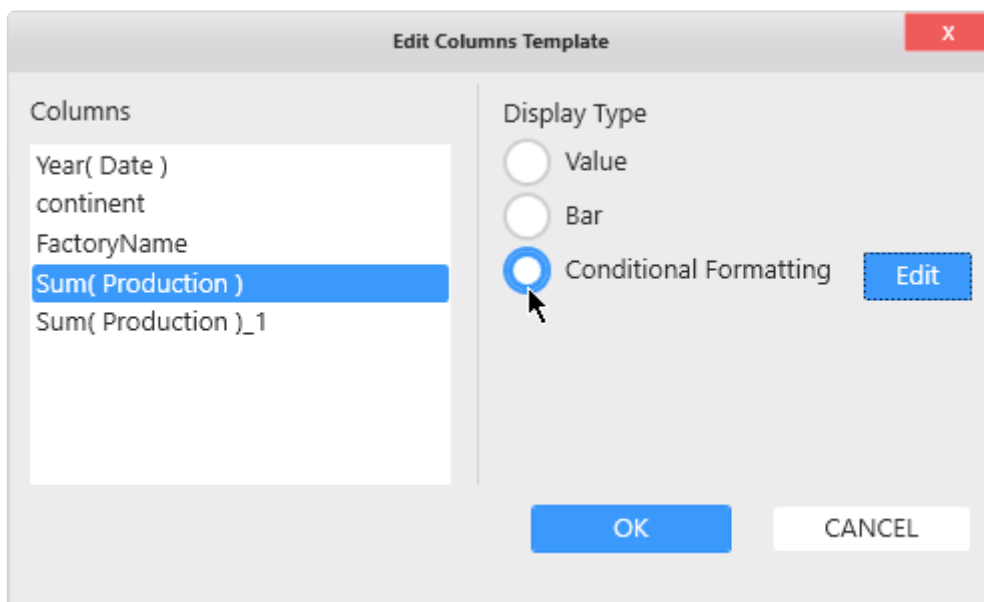
Grid cells can be formatted with color and text style based on data conditions.

Access Conditional Formatting

For this, click *Template* under *Column Settings* :

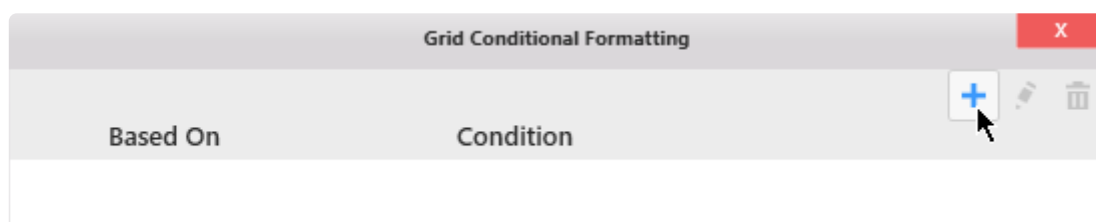


In the *Edit Columns Template* dialog, select on the left a column where you want to apply a formatting, then select *Conditional Formatting* and click *Edit* :



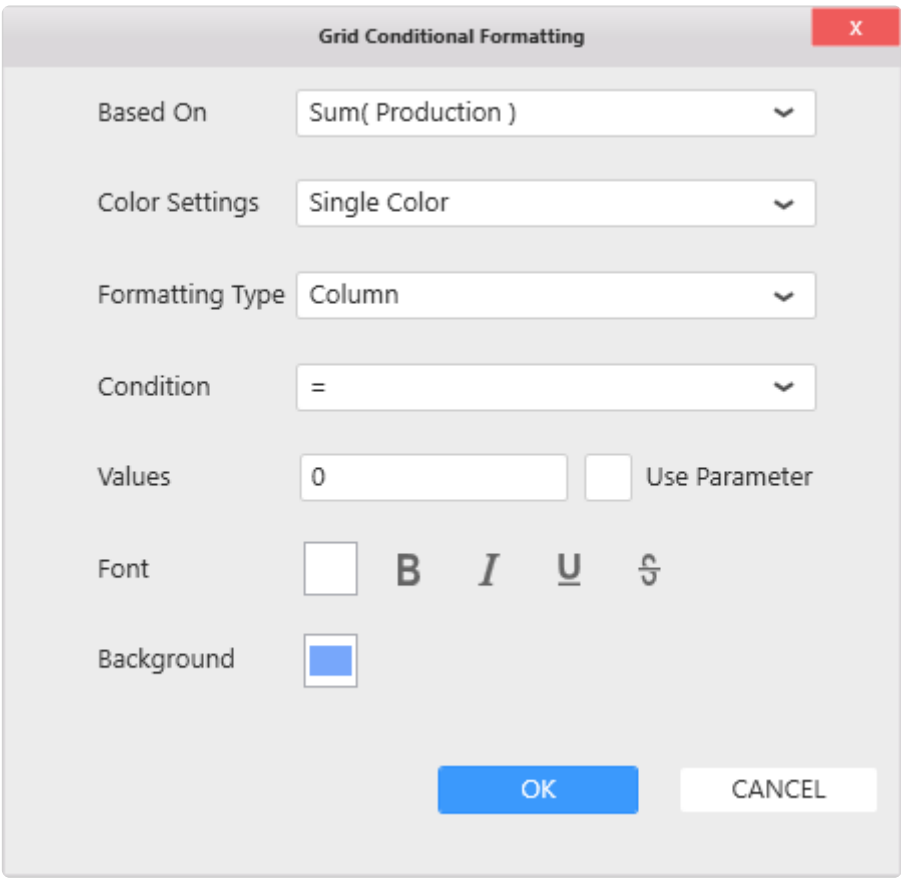
Add a Condition

In the *Grid Conditional Formatting* dialog, click **+** to add a first Condition :



Configure a Condition

A new *Grid Conditional Formatting* dialog appears for this new Condition :

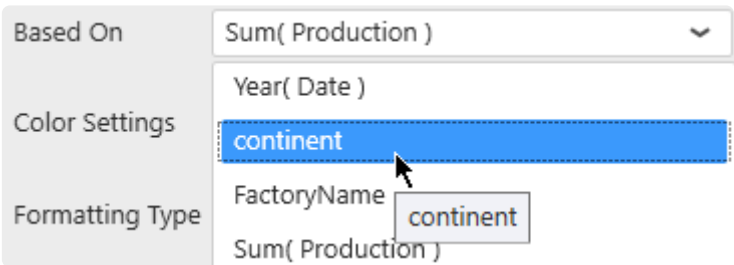


The dialog box is titled "Grid Conditional Formatting" and has a red close button in the top right corner. It contains several settings:

- Based On:** A dropdown menu showing "Sum(Production)".
- Color Settings:** A dropdown menu showing "Single Color".
- Formatting Type:** A dropdown menu showing "Column".
- Condition:** A dropdown menu showing "=".
- Values:** A text input field containing "0" and a checkbox labeled "Use Parameter" which is unchecked.
- Font:** A section with a text input field, and buttons for Bold (B), Italic (I), Underline (U), and Strikethrough (ABC).
- Background:** A color selection button showing a blue square.
- Buttons:** "OK" and "CANCEL" buttons at the bottom right.

Based On column

The Based On column selector allows to select which column will be used to calculate the condition :



This image shows the "Based On" dropdown menu from the dialog box. The menu is open, showing a list of options:

- Sum(Production)
- Year(Date)
- continent (highlighted in blue)
- FactoryName
- Sum(Production)

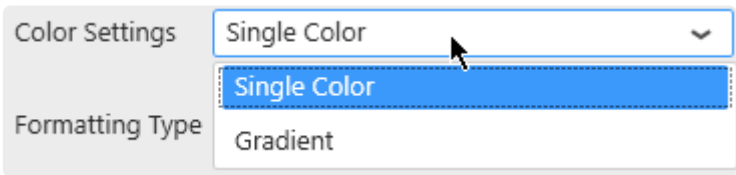
A mouse cursor is pointing at the "continent" option. A small tooltip box next to the cursor also displays the word "continent".



Hint :
A Condition can be based on a different column than the one the formatting is applied to. It can also be based on a Hidden Column. This allows to base the condition on a complex Expression, but not show it to the end user.

Color Settings

Two color modes are possible :

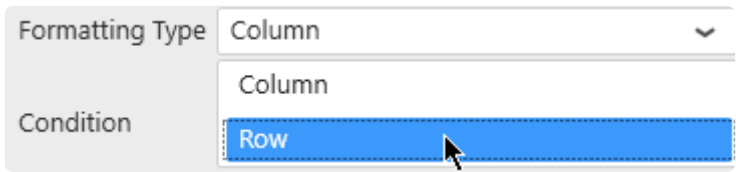


- *Single Color* : based on a boolean condition, a single color is wholly applied or not, depending if the row meets the condition
- *Gradient* : based on a numerical field, a color gradient is applied to all cells, with color intensity depending on the magnitude of the number

Formatting Type

The formatting can be applied either :

- *Column* : on the cells of the selected column only
- *Row* : on all the cells of the same row

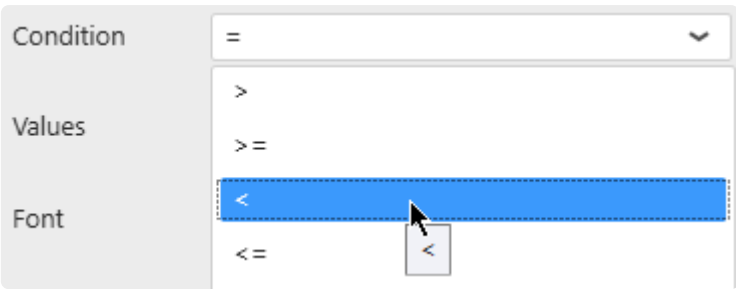


Single Color : Condition

For *Single Color* mode, the boolean condition is configured as : Column Operator Value

Example : AlarmLevel >= 100

The operator can be selected from the *Condition* combobox :



The value to compare to can be entered in the text field called *Values* or a [Dashboard Parameter](#) can be used instead :

Values

☐ Use Parameter

Single Color : Style

For *Single Color* mode, the style applied includes font color and style (**Bold**, *Italic*, Underlined, ~~Stricken~~) :

Font ☐ **B** *I* U ~~S~~

Background ☐

StandardAdvanced

Available Colors

Gradient : Style

For *Gradient* color mode, the style applied includes font color and gradient color :

Color Settings Gradient

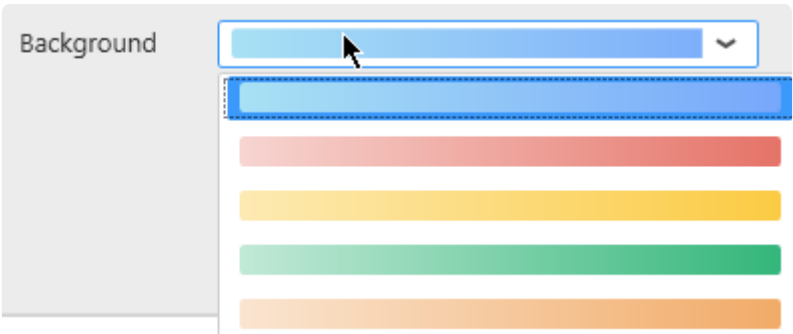
Formatting Type Column

Font ☐

Background ☐

StandardAdvanced

Available Colors



Condition Order

Conditions will be applied in the same order as the list of conditions.

Example

The following list of conditions will apply the widest condition last (>1M), which will override any formatting done in the previous narrower condition (>10M) :

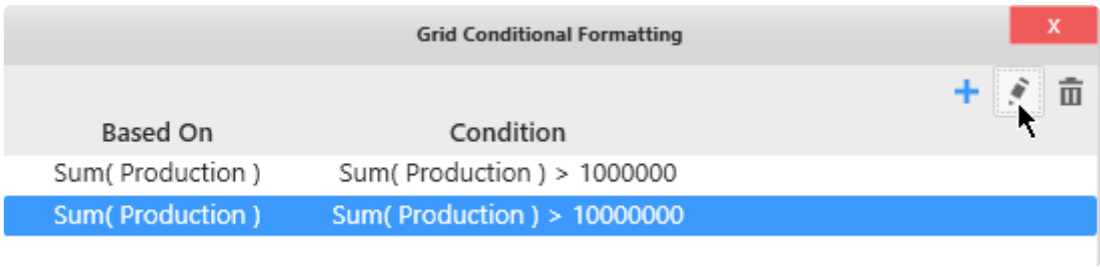
Based On	Condition
Sum(Production)	Sum(Production) > 10000000
Sum(Production)	Sum(Production) > 1000000

In order to apply a different color to the narrower condition (>10M), it needs to be listed last :

Based On	Condition
Sum(Production)	Sum(Production) > 1000000
Sum(Production)	Sum(Production) > 10000000

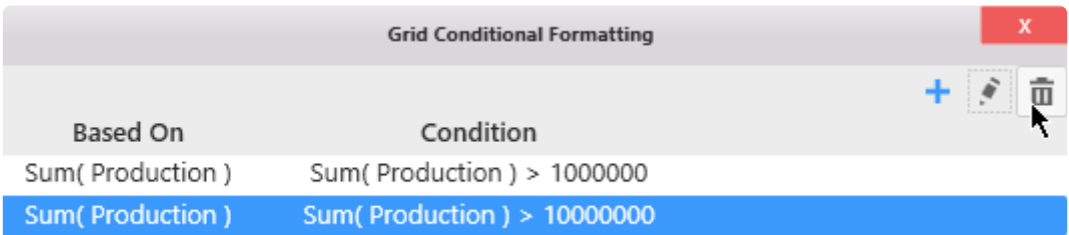
Editing a Condition

Click on a Condition and click on the pen icon to edit :



Deleting a Condition

Click on a Condition and click on the trash icon to delete :



Last modified: 2019/05/07

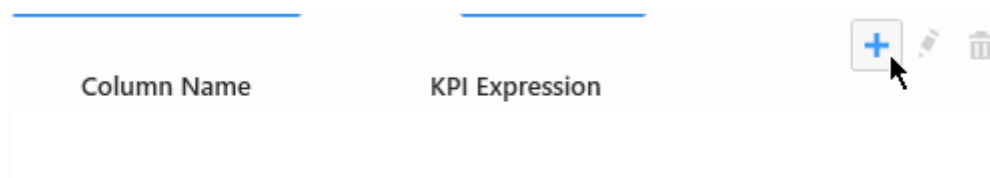
2.7.4.7. Grid KPI

Introduction

A **KPI Expression** (a new calculated column) can be generated to show how a value deviates from a target.

Creating a KPI

To create a KPI, navigate under section Column Settings near the list of KPI Expressions and click the **+** icon :



Configuring a KPI

In the *KPI Column Configuration* dialog that appears, you can configure the KPI :

Kpi Column Configuration

Actual Value

Sum(Production)

Target Value

Avg(ProfitObjective)

Value Type

Actual Variation

Result

Greater is good

Visual

☒ Value

☐ Bar

☐ Delta Background

OK

CANCEL

- *Actual Value* : the column that contains the value you want to track
- *Target Value* : the column to which you want to compare the Actual Value
- *Value Type* : what type of calculation is made to show the deviation
- *Result* : should we consider higher values as a good thing (green) or not (red)
- *Visual* : do we want to visualize the value as a number or a bar graph
- *Delta Background* : do we want to apply conditional color to the numbers or to the cell background

Value Types

Click to select a *Value Type* : what type of calculation is made to show the deviation :

Value Type

Actual Variation

Actual Variation

Result

Actual Value

Visual

Percentage of Variation

Percentage of Target

Value and Percentage

Actual Variation

Shows as a single value the difference between the Value and the Target.

```
Calculation = Value - Target
```

Actual Value

Shows as a single value the Actual Value itself

```
Calculation = Value
```

Percentage of Variation

Shows as a single value the % of variation to target.

```
Calculation = 100 * (Value - Target) / Target
```

Percentage of Target

Shows as a single value the % of value to target.

```
Calculation = 100 * Value / Target
```

Value and Percentage

Shows 2 values in the cell :

- *Actual Variation*
- *Percentage of Variation*

Result interpretation

The font and symbol of the KPI has a color :

- **green** for “good”
- **red** for “bad”

You can define high values as being “good” or “bad” using the Result configuration :

Result	Greater is good
Visual	Greater is good Less is good

Style : background

By default, the good/bad color is applied to the font, and the cell is white.

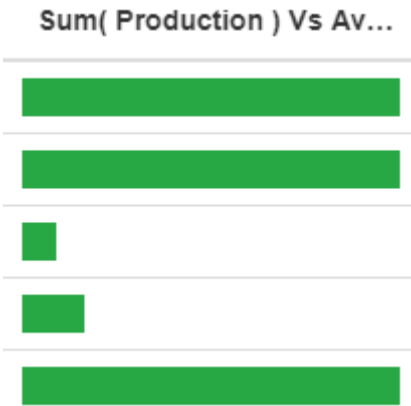
This can be configured as the opposite, for better visibility. Check *Delta Background* :

☒ Delta Background

Style : Bar chart

Instead of showing the value as a number, the Actual Value calculation can be displayed as a Bar chart by selecting the appropriate Visual :

Visual ☐ Value ☒ Bar



Last modified: 2019/05/07

2.7.4.8. HTML in the Grid

Introduction

The Grid Widget will display any text content as HTML (provided there is no conflict with the page structure, styles, scripts, etc).

There is no clear boundaries of what is supported and what might break a dashboard, and you should make tests for the HTML features you want to use.

General example

Your data contains a text field (called "html" here) with some HTML markup :

	A	B	C	D	E
1	html				
2	©[code192]				
3	html image :				

(this is from the same data as above)

This can be used for many customization cases, including displaying animated GIFs conditionally :

day




↺

1

▼

alarms

☰

html_img	name
	pH meter
	pressure meter
	temperature sensor

Last modified: 2019/05/07

2.7.5. TreeMap

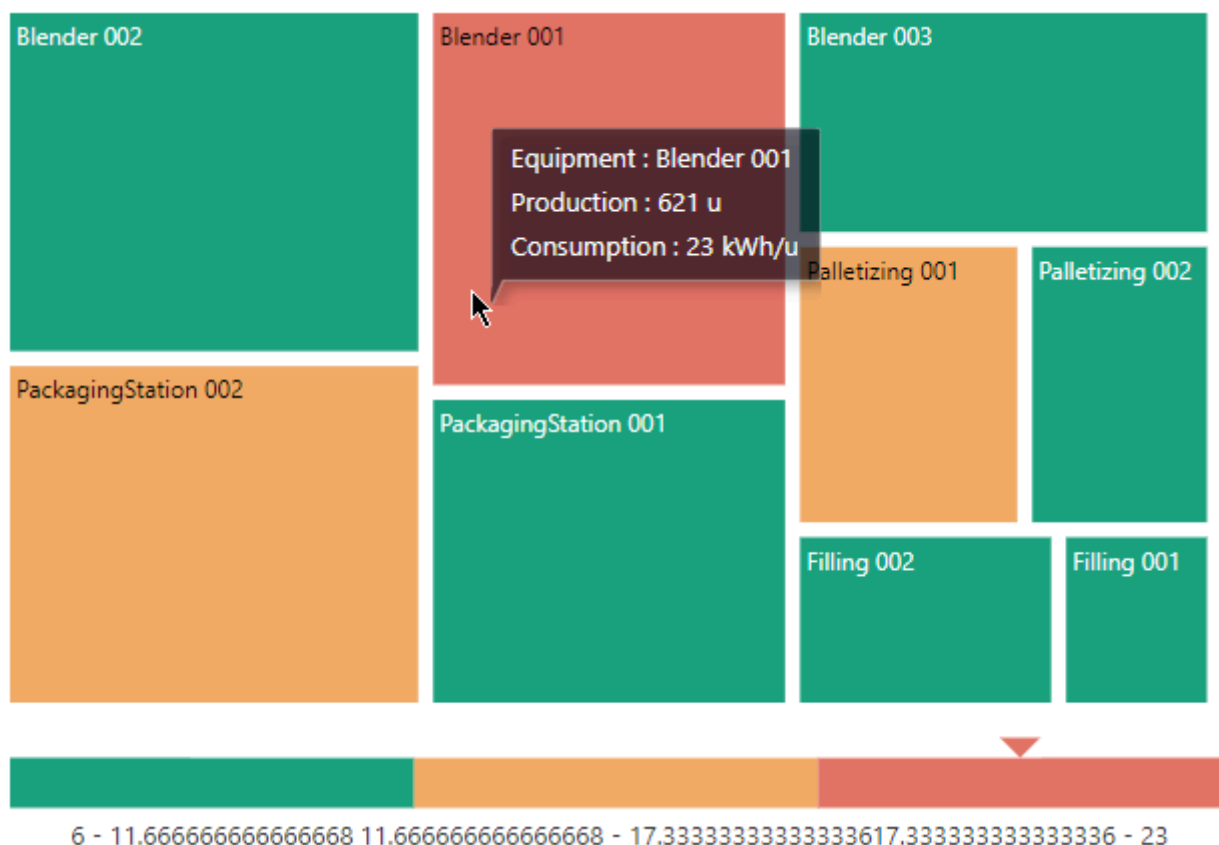


The TreeMap widget can be used as an extension of Proportion Charts : data is grouped in rectangles whose size and color can be bound to different data fields.

In a typical case, the size represents the importance, and the color represents the KPI itself. This can be useful typically for showing the efficiency of something combined with the quantity.

Example :

Each rectangle is grouped by Equipment. The size of rectangles is bound to the Quantity produced, the color is bound to the energy Consumption.



In the above TreeMap, we can see that the Blender_001 has very bad energy efficiency, but represents only a small fraction of the total production.

Last modified: 2019/05/14

2.7.5.1. Binding Data

Introduction

The following field containers are available for configuring data binding on a TreeMap :

Show

Weight

Sum

of

Production

Sorted

Asc

Value(s)

Sum

of

Consumption

Sorted

Asc

Group by

Group by

Value

of

Building

Sorted

Asc

Value

of

Zone

Sorted

Asc

Value

of

Equipment

Sorted

Asc

Value

of

Add optional field...

- *Weight* : the size of the rectangles. If no “Value” is configured, this is also the color.
- *Value* : the color of the rectangles
- *Group By* : how we group values together to create separate rectangles

Weight

This represents the quantity we want to display on the TreeMap, and will bind for rectangle size.
If no “*Value*” field is configured, this field also binds to the color of the rectangles.

This field **must be aggregated** : you must choose an aggregation function (SUM, AVG, ...)

This field is required.

Values

When a *Values* field is bound, it defines the color of the rectangles, while the first field now only defines the size of rectangles.

This field **must be aggregated** : you must choose an aggregation function (SUM, AVG, ...)

This field is optional.

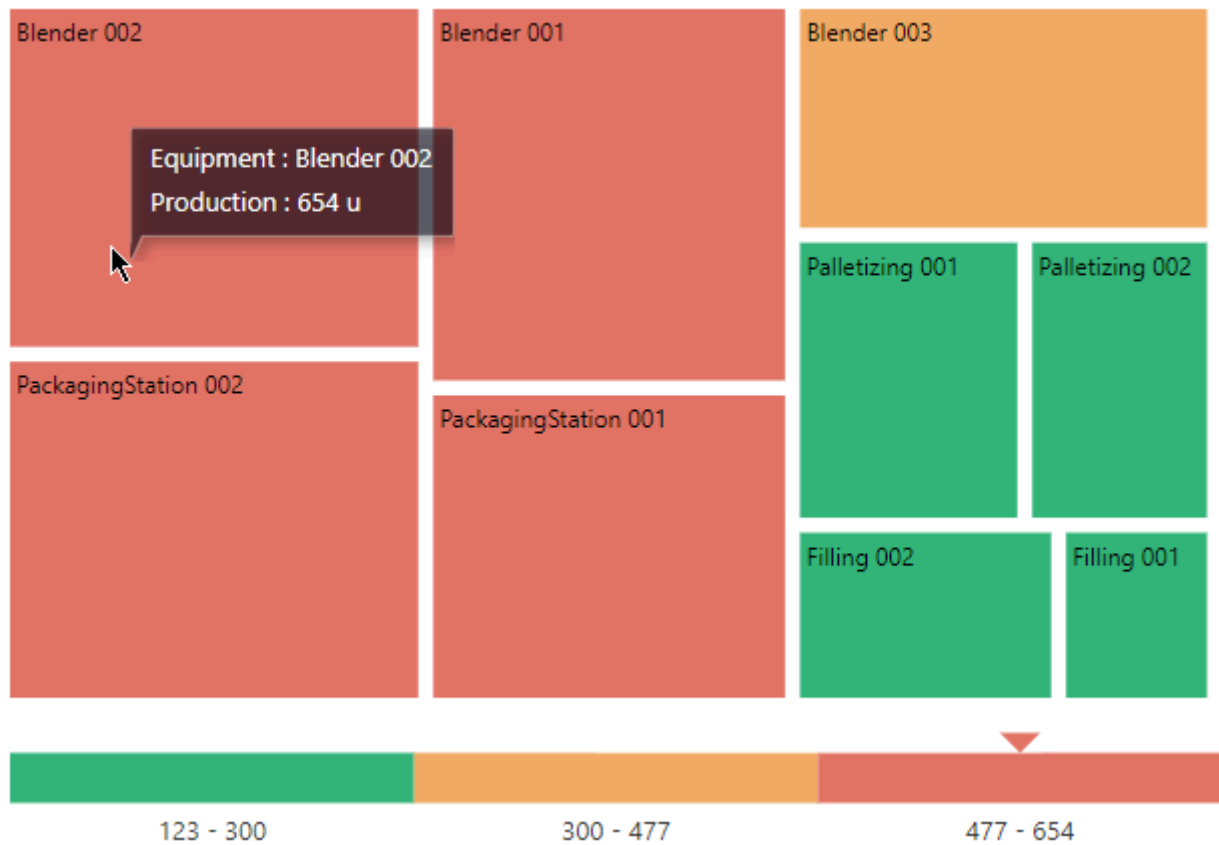
Example

For example, you configure a single field “Production” :

The image shows a configuration window for a TreeMap. It has two main sections: 'Weight' and 'Value(s)'. In the 'Weight' section, there is a dropdown menu set to 'Sum', followed by the word 'of', and another dropdown menu set to 'Production'. To the right of this is a close button (X). Below these is a 'Sorted' label followed by a dropdown menu set to 'ASC'. The 'Value(s)' section is below the 'Weight' section and is currently empty, showing a dashed border and a placeholder text 'Add optional field...'.

It defines both the size and color of rectangles.

So for example Equipment “Blender_002” has value 654 which is high (large and red) :



Now you configure as secondary field a “Consumption” :

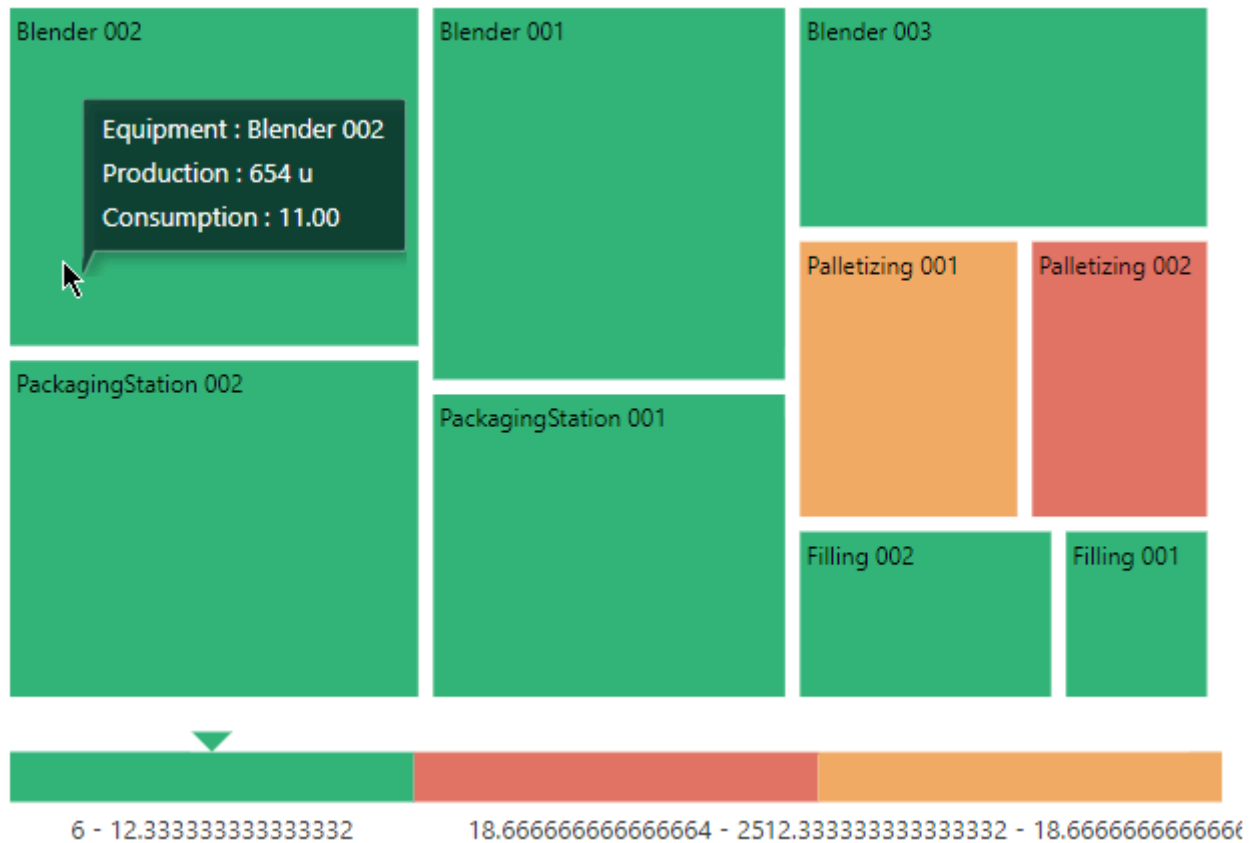
Weight

Sum of Production
Sorted ASC

Value(s)

Sum of Consumption
Sorted ASC

Now the rectangles keep the same size (the *Weight* field is the same), but the color is different. So for example Equipment “Blender_002” still has Quantity 654 which is high (large), but now the color is green because the secondary “Consumption” is low (11) :



Group By

This represents the category we want to use to separate rectangles.

These fields **cannot be aggregated** : they must have several separate values in order to group the “Values” aggregation.

Multiple fields : Grouping

When binding multiple fields at the same time, the rectangles will be grouped by the defined hierarchy.

For example, binding fields Continent>Country>Factory :

Group by

Value

▼

of

continent

▼

✕

Sorted

ASC

▼

Value

▼

of

Country

▼

✕

Sorted

ASC

▼

Value

▼

of

FactoryName

▼

✕

Sorted

ASC

▼

Value

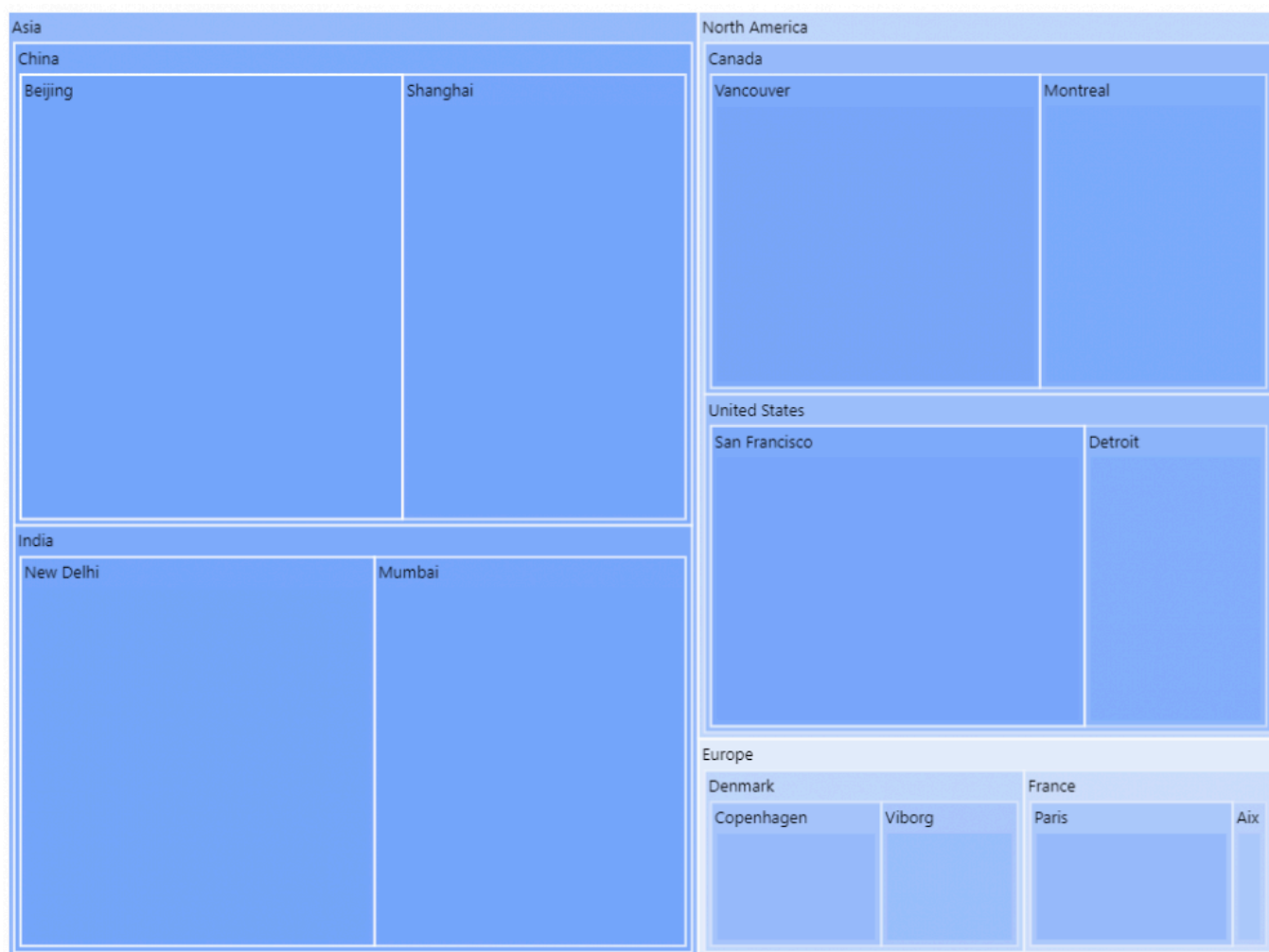
▼

of

Add optional field...

▼

... will group factories by country and the countries by continent :



Multiple fields : Drill-Down

When binding multiple fields at the same time, it is possible instead to define Drill-Down : the user will navigate the hierarchy by clicking.

Under Basic Settings, select Enable DrillDown :

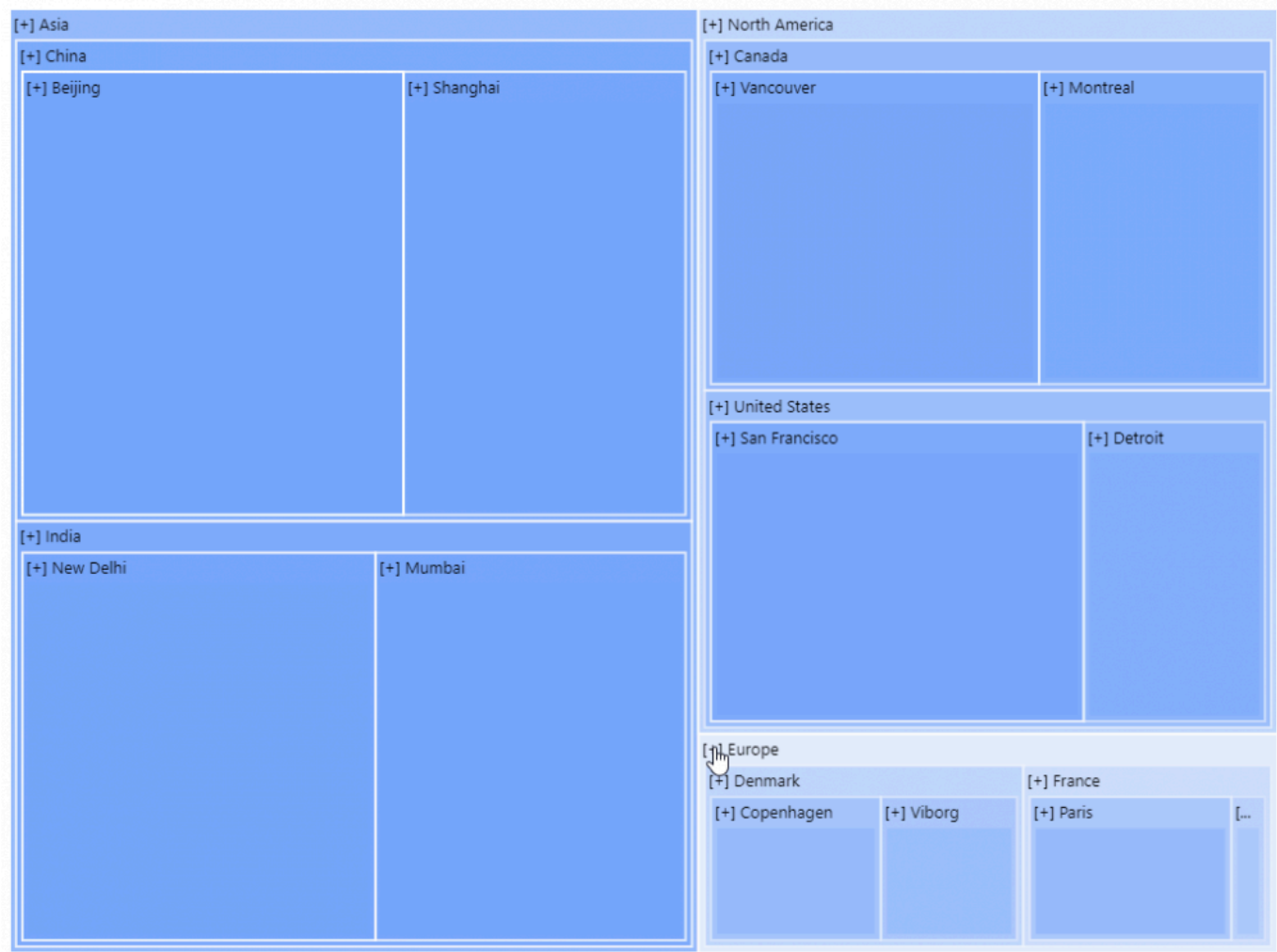


Now again the factories are grouped by country and the countries by continent. But in addition, this can be navigated by clicking on the [+] or [-] icon on shapes :

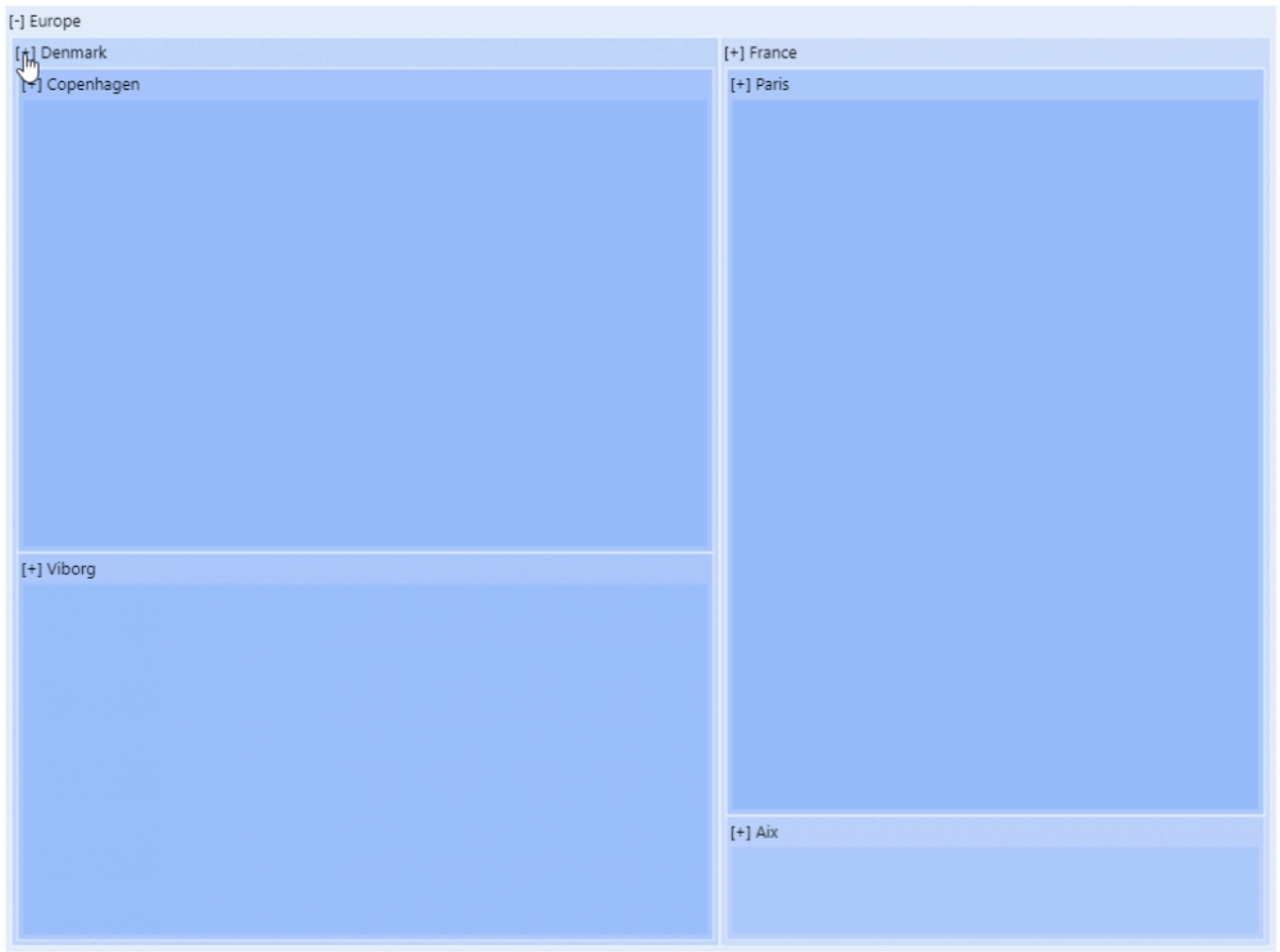


The User can click a Continent to drill down. For example, here clicking [+] on “Europe” brings focus to

this level :



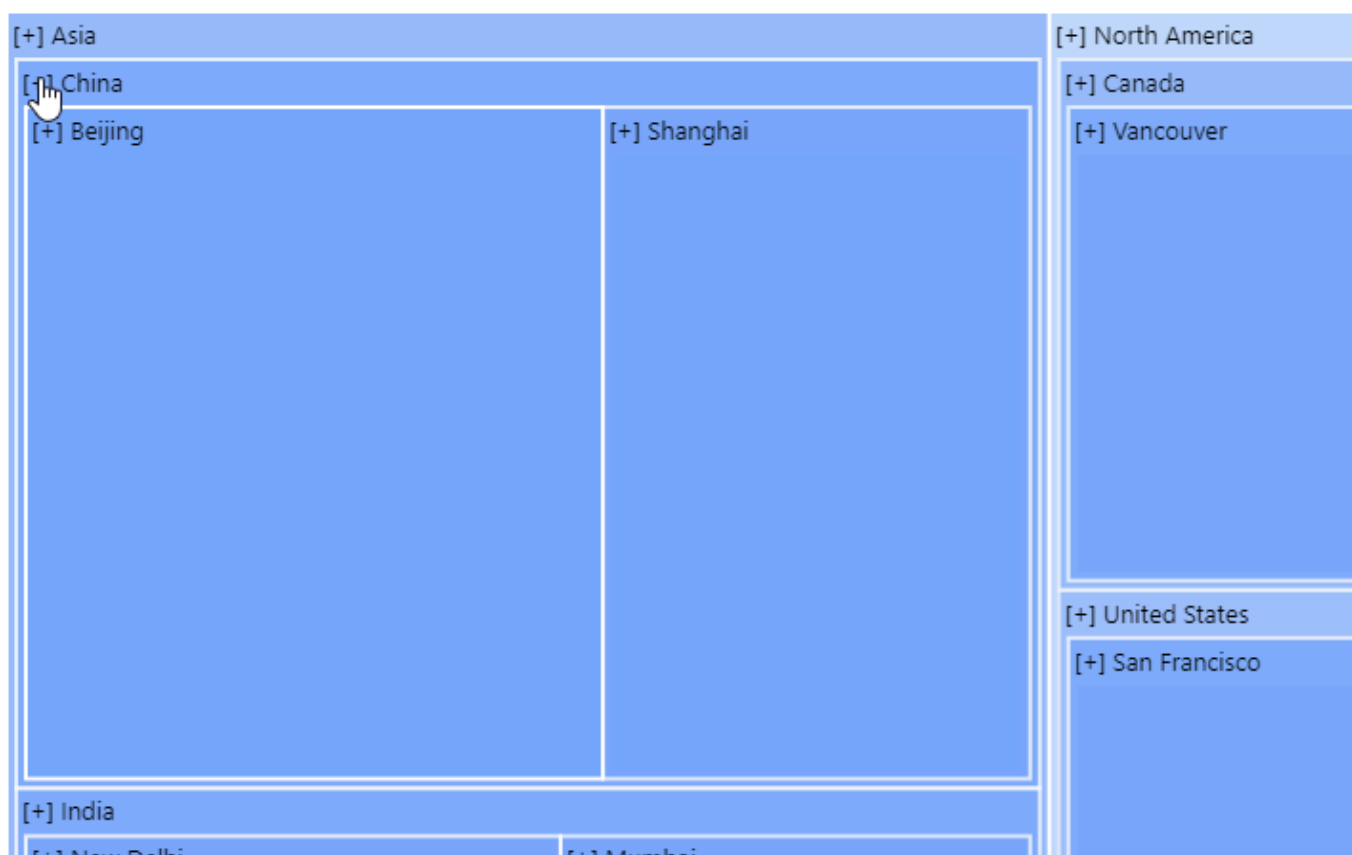
... and see a list of countries for that continent. Then clicking [+] on a Country brings focus on that Country :



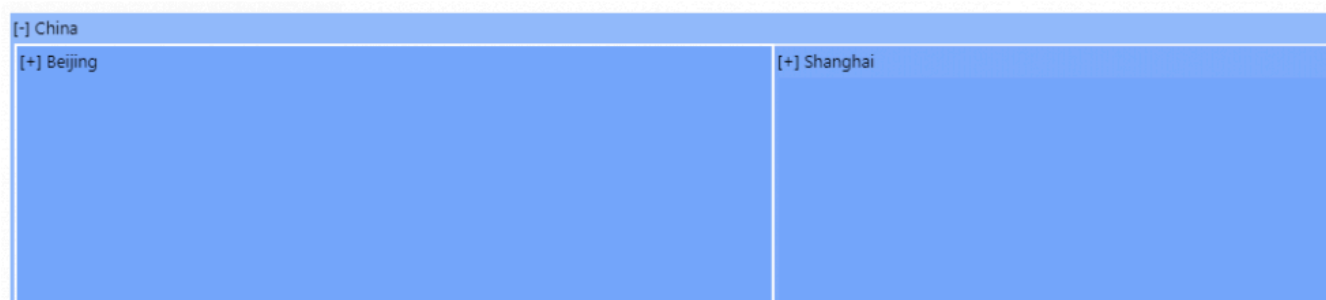
... and see a list of Factories for that Country :



It is possible to click the [-] to navigate up the DrillDown.



It is also possible to click on a lower level directly to navigate to it. For example, from top level, on can click directly a Country to drill down to this country and compare the Factories :



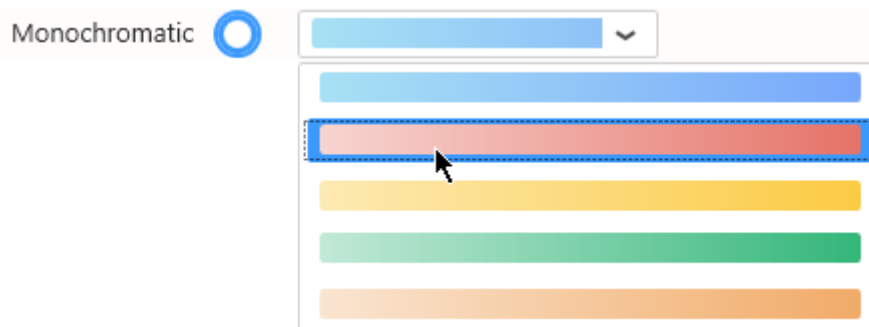
Last modified: 2019/05/14

2.7.5.2. Managing Colors

Monochromatic

By default, the TreeMap shows value intensity as a gradient of a single color.

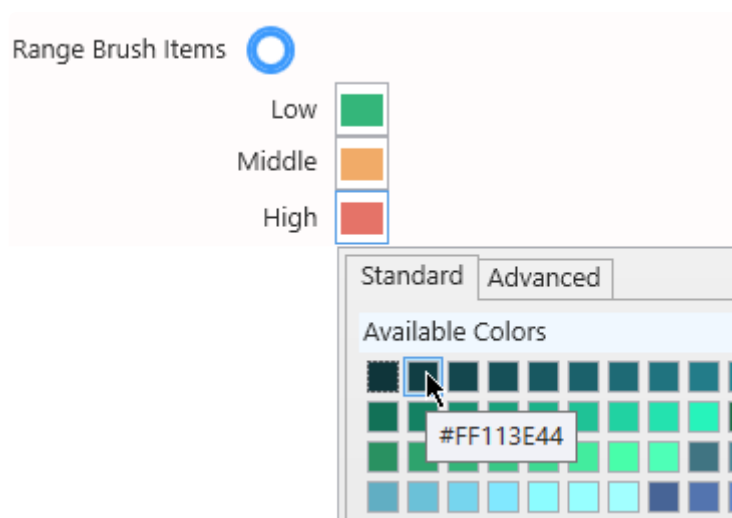
This *Monochromatic* Gradient can be set to a different color by using the dedicated combobox under *Range Color Settings* :



Range Brush

The TreeMap can be configured to show value intensity in ranges of single colors.

Select *Range Brush Items* and choose the colors of each interval :



The interval between the maximum of Value and the minimum of Value will be split in equal intervals and assigned a single color.



This can be useful for setting a color code like “low is red, medium is orange, high is green”.

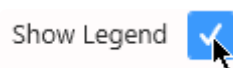
Last modified: 2019/05/14

2.7.5.3. Data Point information

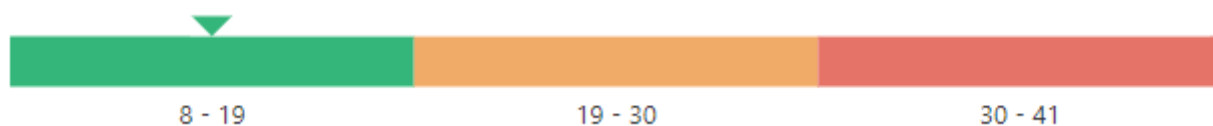
Legend

If using **Range Brush** colors, the TreeMap can display a legend.

Click on Show Legend under Basic Settings :



The scale is displayed with the value interval for each color, and the value for the currently selected shape is highlighted with an arrow :



Last modified: 2019/05/14

2.7.6. Gauge



The Gauge Widget can be used to display a single value as a gauge filling and optionally compare it to a target.

It can also be turned into a container for an indeterminate number of Gauges.

Last modified: 2019/05/06

2.7.6.1. Binding Data

The Gauge has 3 data fields available for binding :

- *Actual Value* : value displayed as a filling of the Gauge
- *Start Value* : dynamic minimum value
- *End Value* : dynamic maximum value
- *Target Value* : target displayed as an arrow on the Gauge
- *Series* : how we split series of different gauges in the same container

Show

Actual Value

Avg

▼

of

Consumption

▼

Sorted

by default

▼

×

Start Value

Min

▼

of

Min

▼

Sorted

by default

▼

×

End Value

Max

▼

of

Max

▼

Sorted

by default

▼

×

Target Value

Avg

▼

of

Target

▼

Sorted

by default

▼

×

Group by

Series

Value

▼

of

Equipment

▼

Sorted

ASC

▼

×

Actual Value

This represents the **primary quantity** we want to display on the Gauge, and will bind to the needle position and filling of the Gauge.

This field must be aggregated : you must choose an aggregation function (SUM, AVG, ...)

This field is required.

Actual Value

Avg

▼

of

Consumption

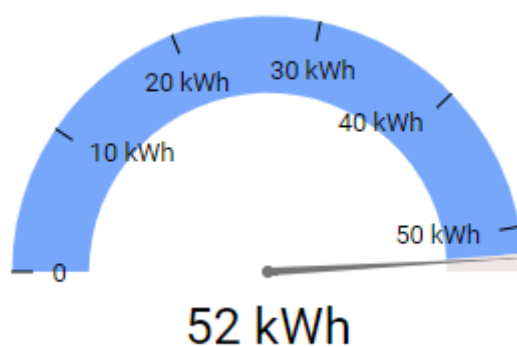
▼

✕

Sorted

by default

▼



Start Value

This represents the Minimum we want to display on the Gauge, for example the “MinEU” field from Wonderware Historian.

When this field is bound, it overrides manual configuration of the Custom Range Minimum.

If the Actual Value goes below this Start Value, the Gauge will appear empty.

This field must be aggregated : you must choose an aggregation function (SUM, AVG, ...).

This field is optional.

Start Value

Min

▼

of

Min

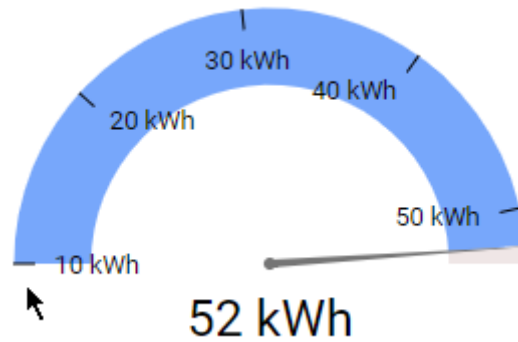
▼

✕

Sorted

by default

▼



End Value

This represents the Maximum we want to display on the Gauge, for example the “MaxEU” field from Wonderware Historian.

When this field is bound, it overrides manual configuration of the Custom Range Maximum.

If the Actual Value goes above this End Value, the Gauge will appear empty.

This field must be aggregated : you must choose an aggregation function (SUM, AVG, ...).

This field is optional.

End Value

Max

▼

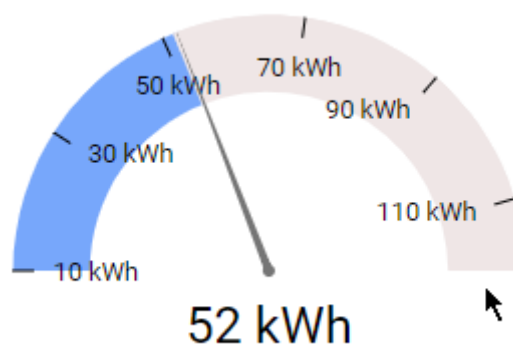
of

Max

▼

✕

Sorted by default ▼



Target Value

This represents the Target we want to display on the Gauge, and will bind to the position of a small arrow on the Gauge.

When this field is bound, an additional information is displayed as text : the distance from Actual Value to Target Value in units and in percentage.

The Target Value field must be aggregated : you must choose an aggregation function (SUM, AVG, ...). This field is optional.

Target Value

Avg

▼

of

Target

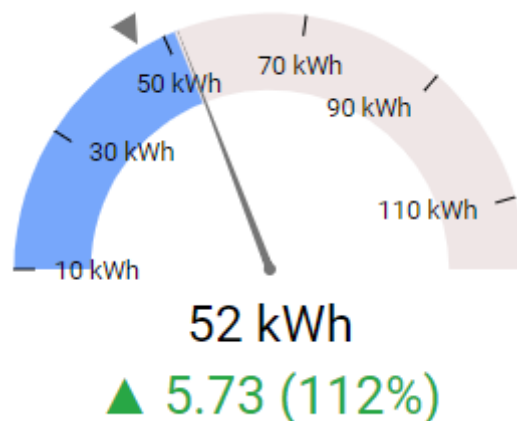
▼

✕

Sorted

by default

▼



Series

This represents the series field to generate a different contained Gauge for each different value in this Series field.

Each Gauge will take a portion of size of the widget depending on the container shape. In order to access the other values, the user may need to scroll.

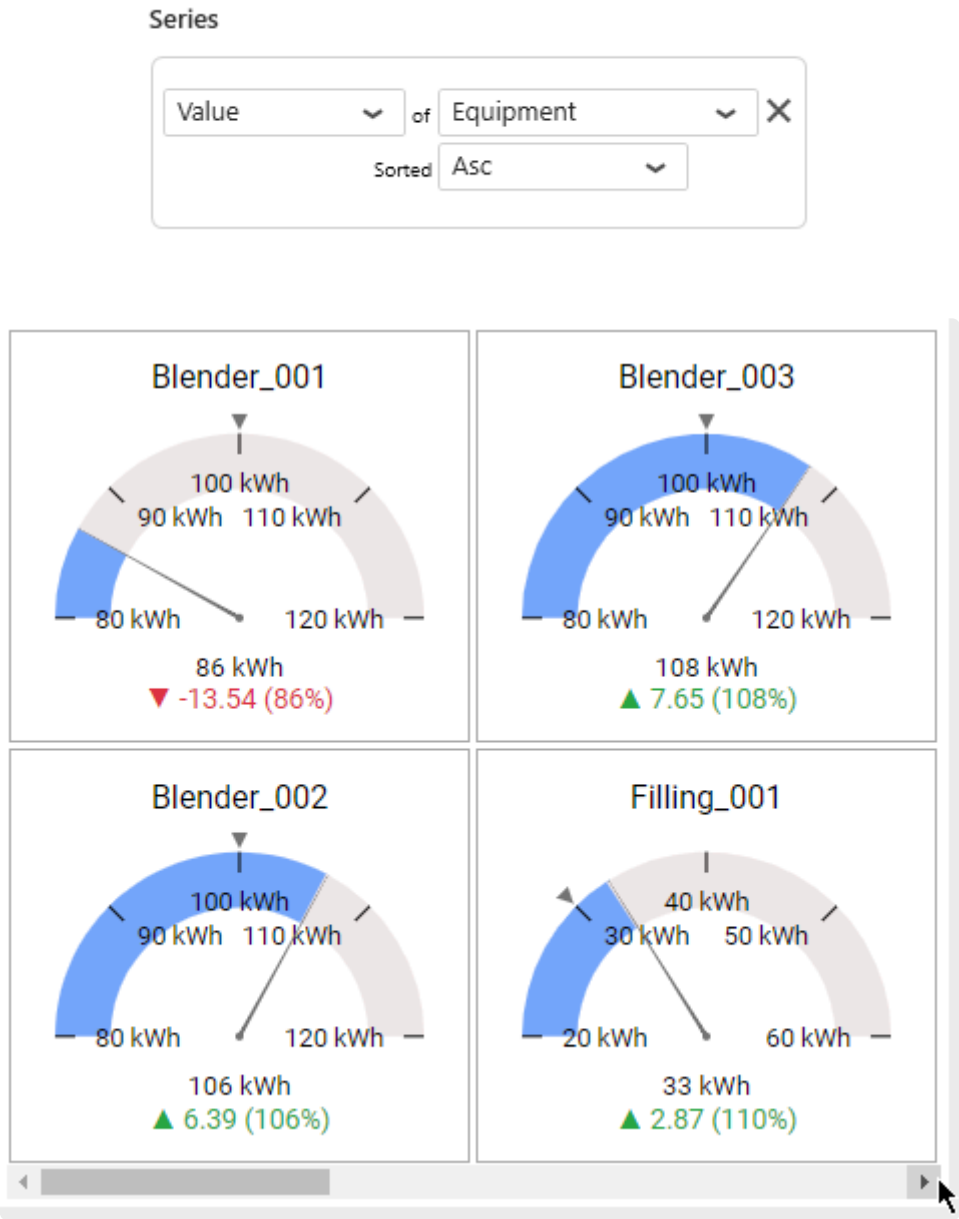
Please see [the corresponding chapter to configure the Gauge as Widget container](#).



Note :

This only works if your series are values into a single data field.

Example : If you have a field called “Equipment”, you may want to plot a completely separate Gauge for each value of Equipment :



Last modified: 2019/05/06

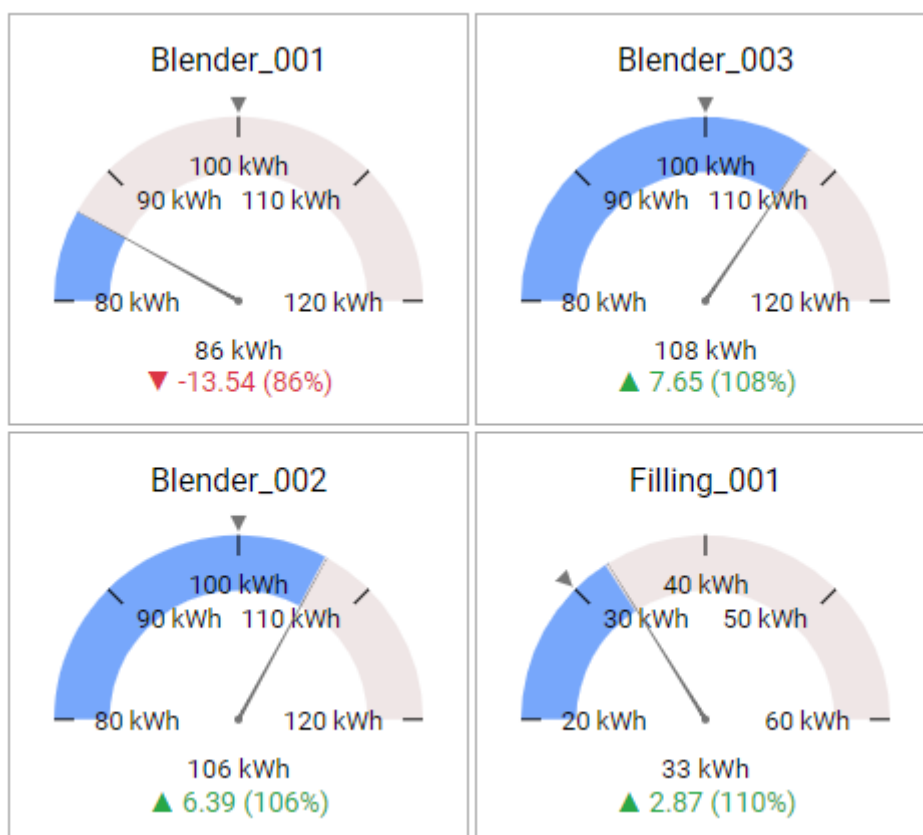
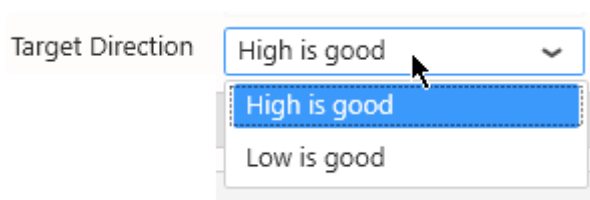
2.7.6.2. Value Display

Result interpretation

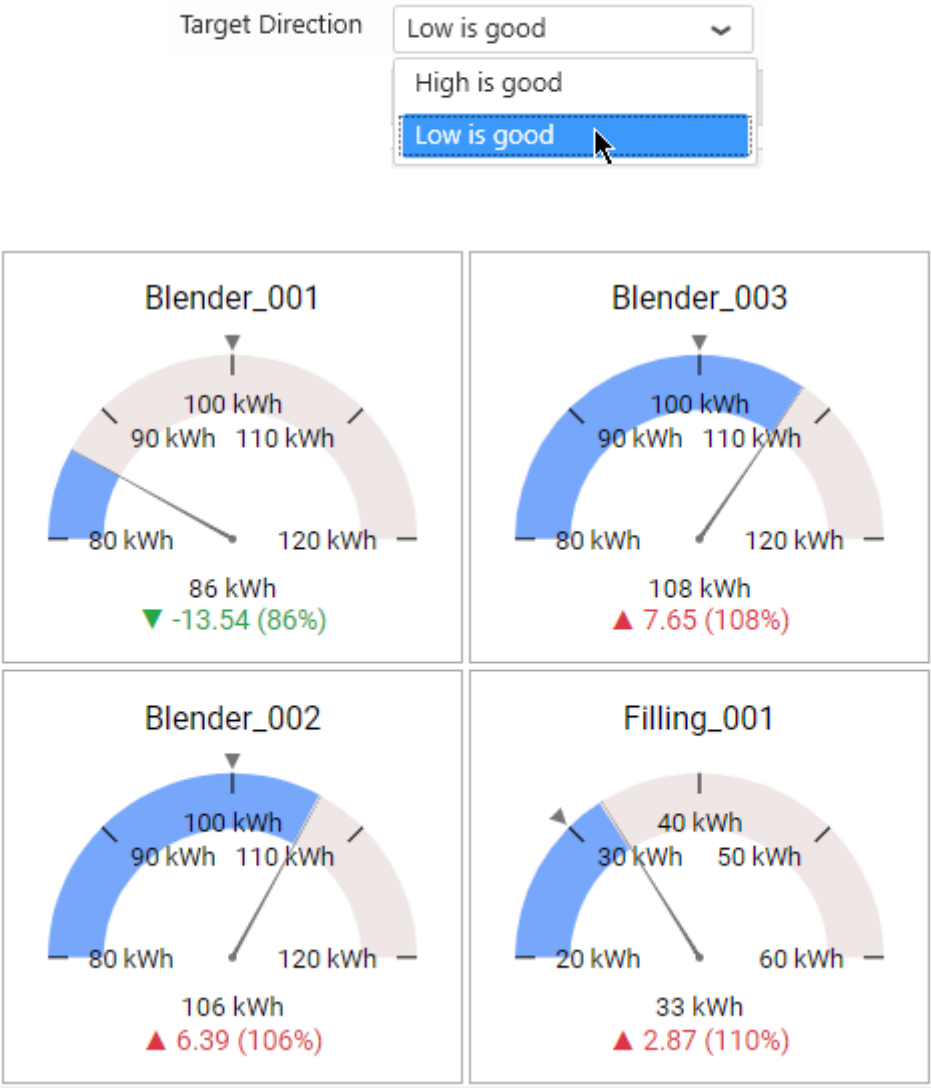
The font and symbol of the distance to target has a color :

- **green** for “good”
- **red** for “bad”

You can define high values as being “good” or “bad” using the *Target Direction* configuration in *Basic Settings* :



... reverts into :



Value Range

Dynamic value ranges can be defined by binding data fields to the [Start Value and End Value](#) configurations.

If instead you want static values for the boundaries of the value range, go under Basic Settings , check the desired part of the *Custom Range* and fill the number box :

Custom Range

☒

Minimum

☐

Maximum

This will override the dynamic data binding from Start Value and End Value.

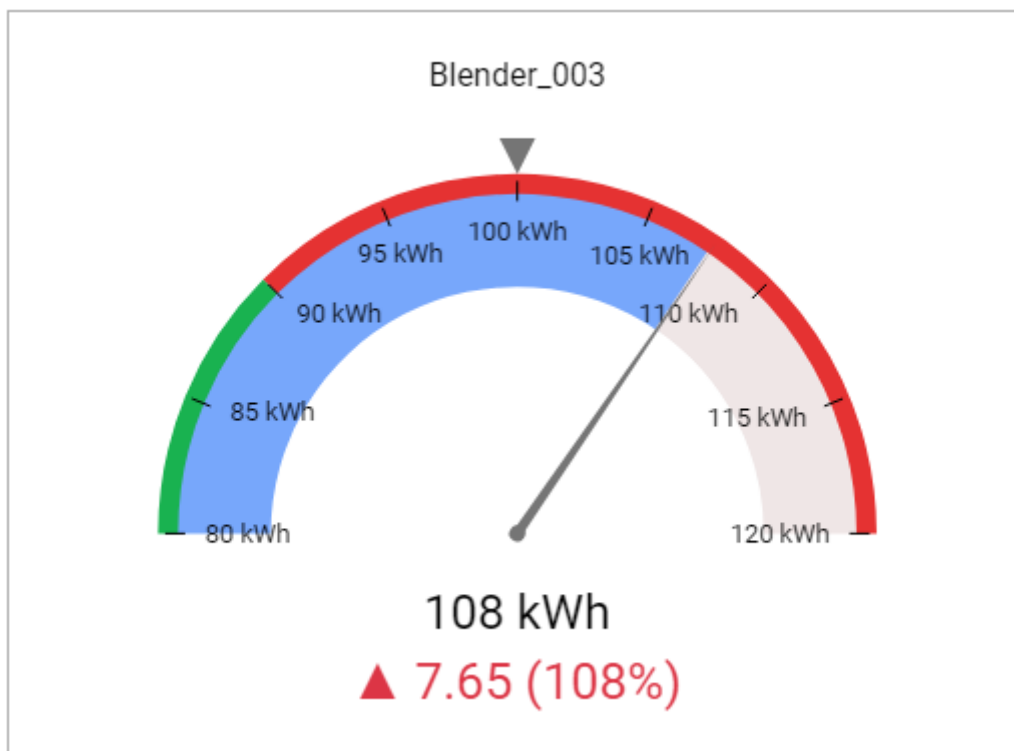
Value Threshold

Value of the Threshold

A value threshold can be displayed on the Gauge to define a minimum or maximum bad value. This is displayed as a background color that goes from green (good) to red (bad).

To enable this feature, check *Custom Threshold* under *Basic Settings* and set a *Threshold Value* :

Custom Threshold ☒ Threshold value



Note :

If a *Target Value* is configured (as in the screenshot above), the KPI still displays the distance to Target, and **not the distance to Threshold**.

Direction of the Threshold

By default, the low values are defines as good, so the green part is displayed on the low values.

If instead you want to define a low threshold and let low values be displayed as red, set the *Threshold Direction as High Is good* :

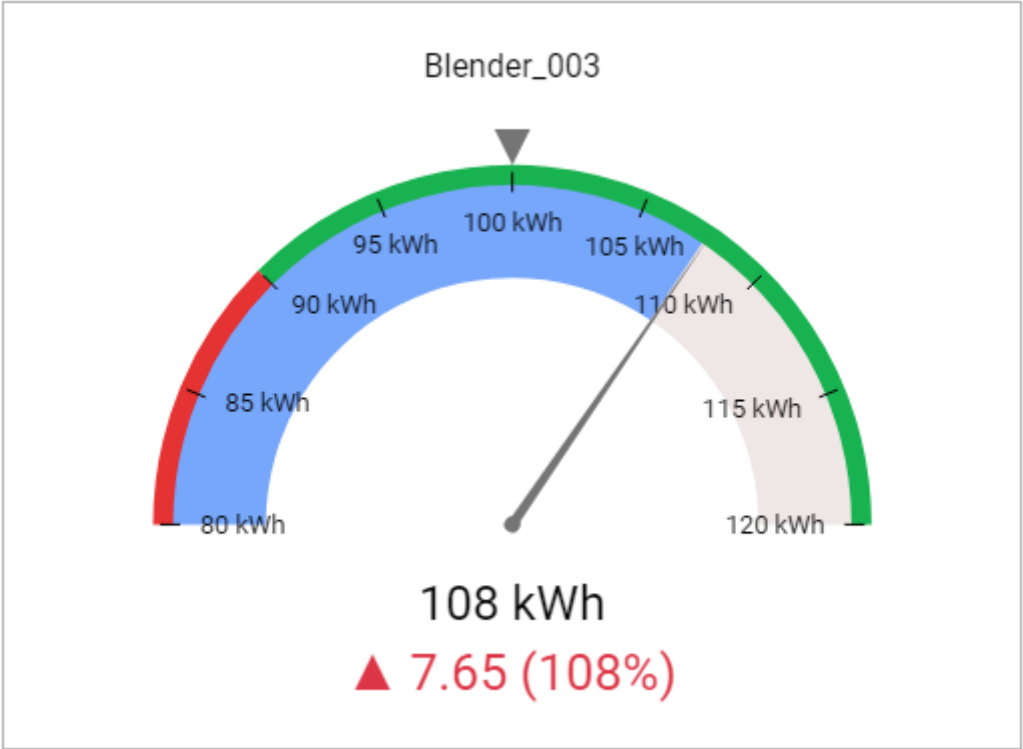
Threshold Direction

High is good

High is good

Low is good

High is good



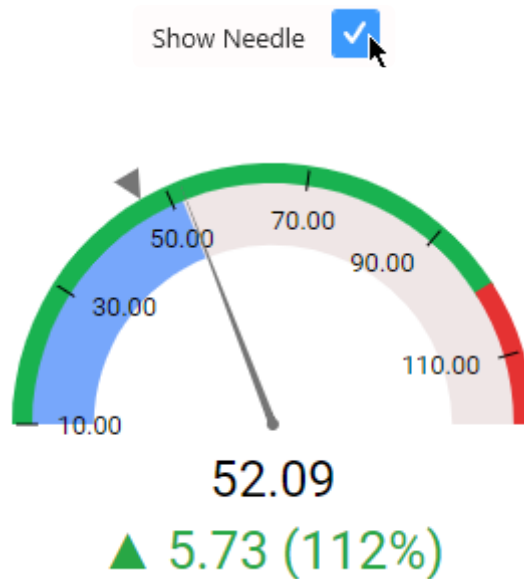
Last modified: 2019/05/06

2.7.6.3. Style and Appearance

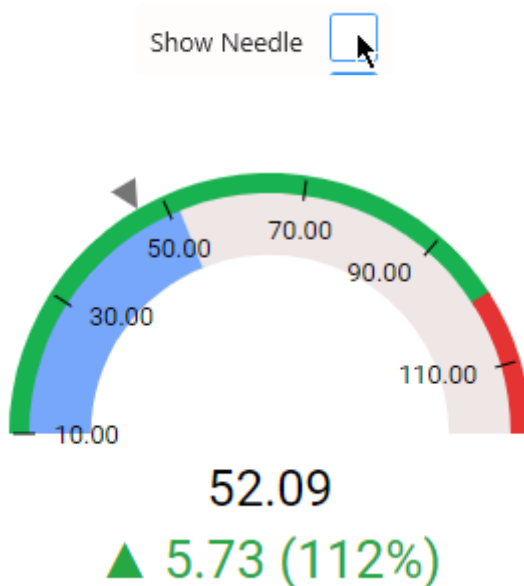
Needle

By default, the *Actual Value* is displayed using a gauge Needle.

This Needle can be shown or hidden by checking the option *Show Needle* under *Basic Settings* :

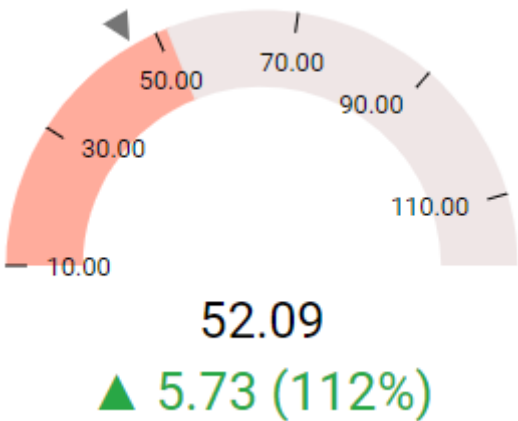
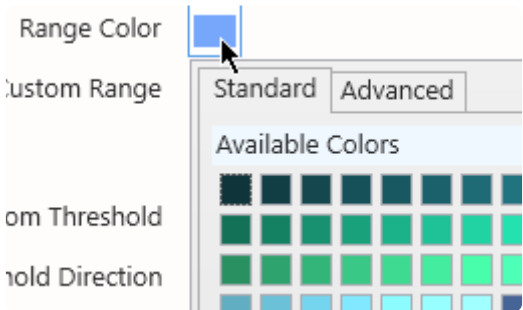


... turns into :



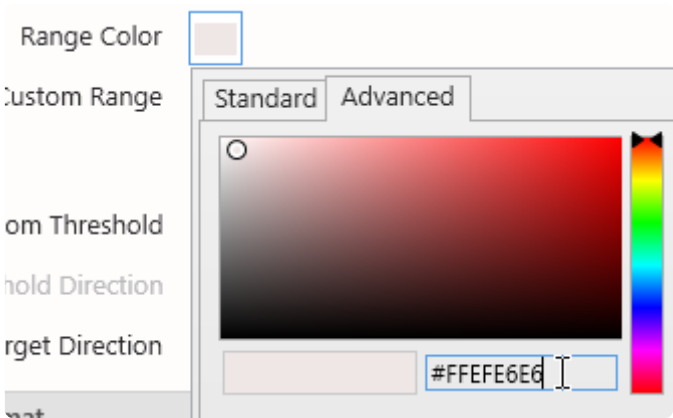
Gauge filling : *Range Color*

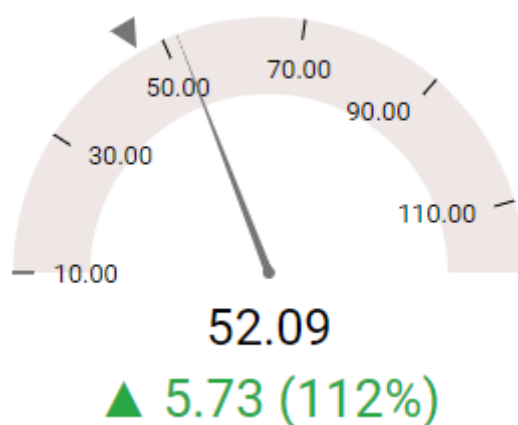
The Gauge displays the *Actual Value* as a color filling.
This color can be configured from *Range Color* under *Basic Settings* :



tip : hiding color

If you want to use only the Needle to display the *Actual Value*, you can hide the Range Color by setting it to the same color as the background color :

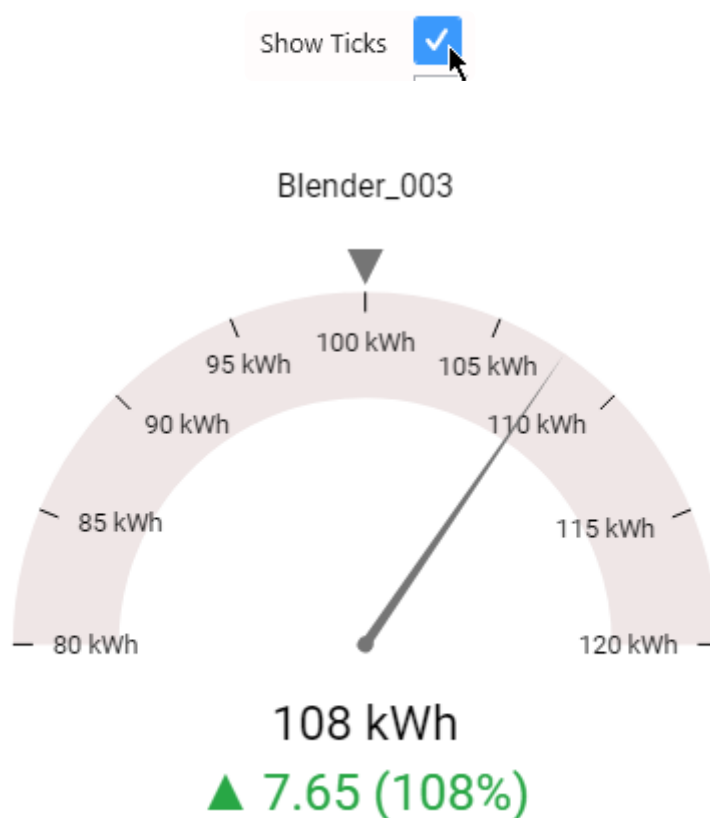




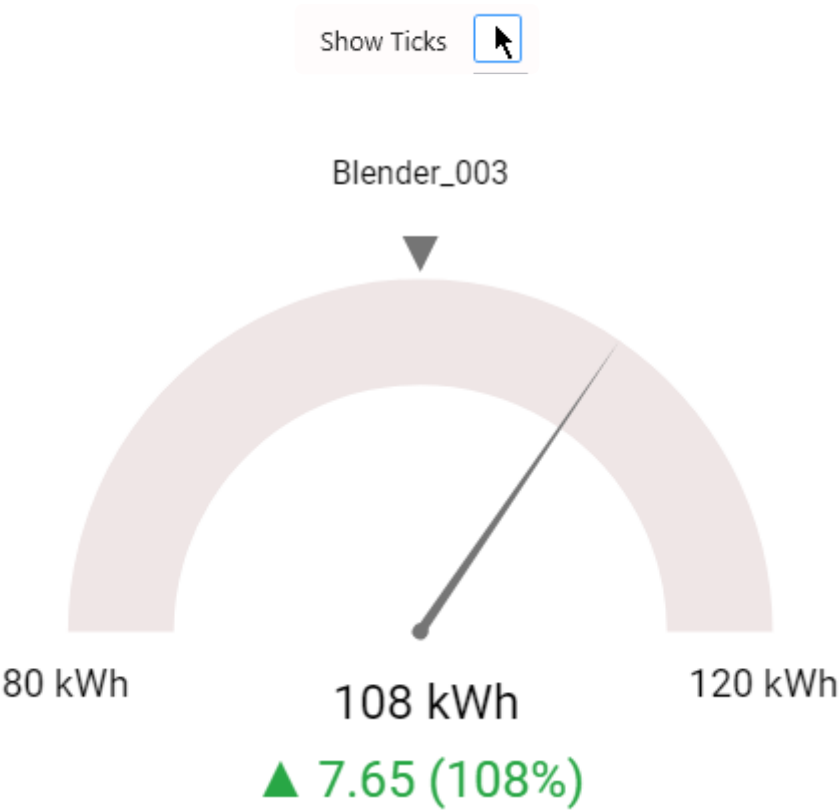
Range scale : *Ticks*

By default, the range of values is displayed as tick marks on a graduated scale.

When un-checking *Show Ticks* under the *Basic Settings*, these values are no longer displayed, and only the minimum and maximum are displayed :

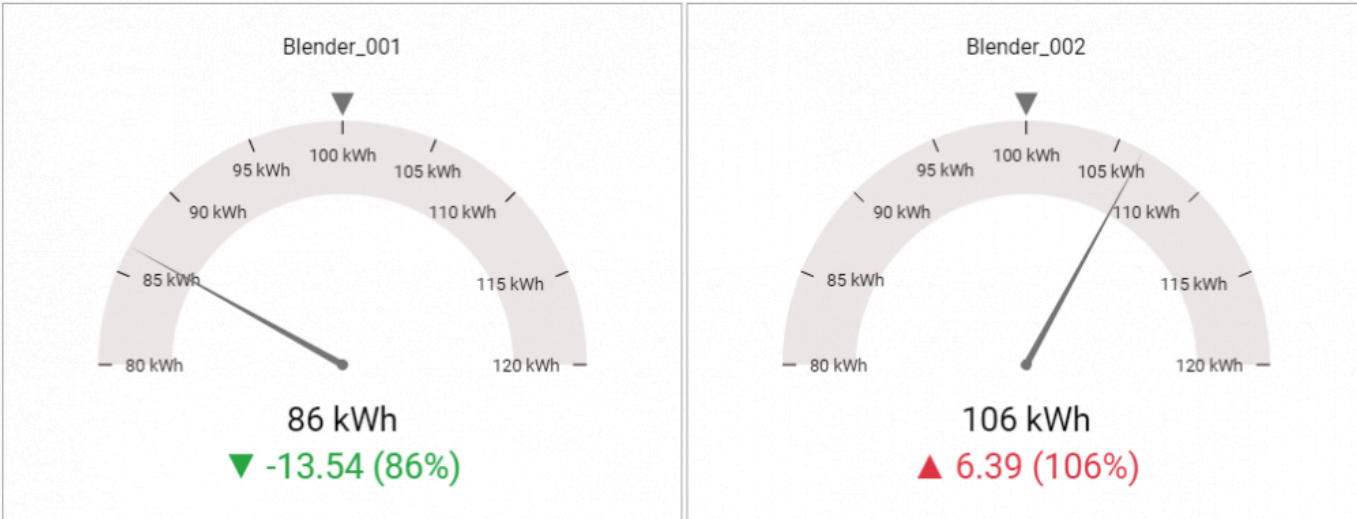


... turns into :



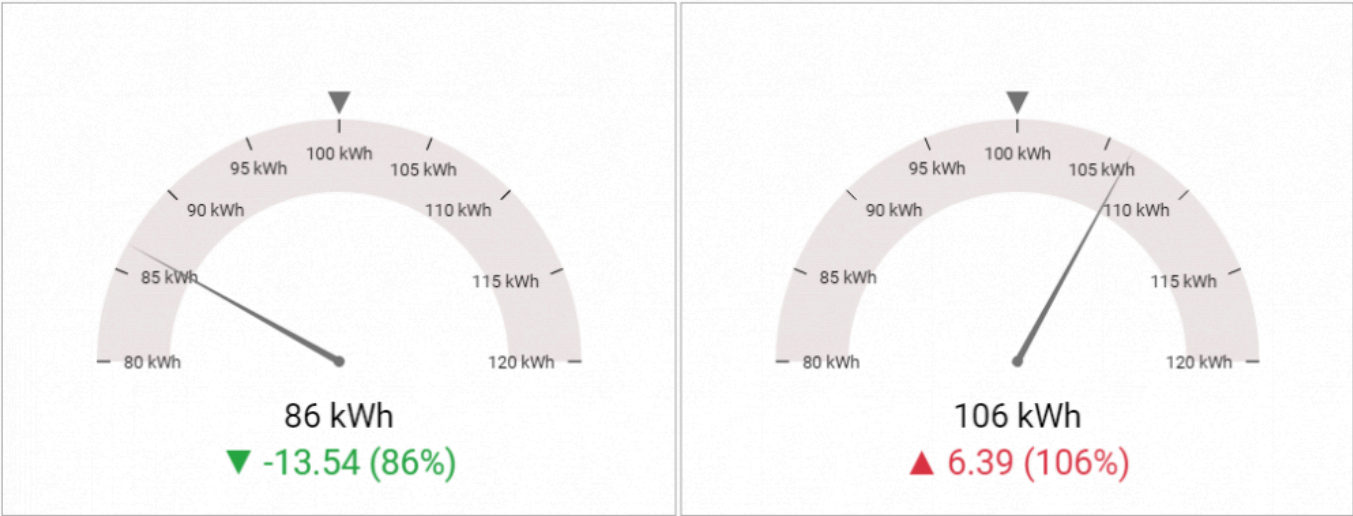
Series Titles

When the Gauge is configured with a *Series*, each Gauge in the series displays a Title :

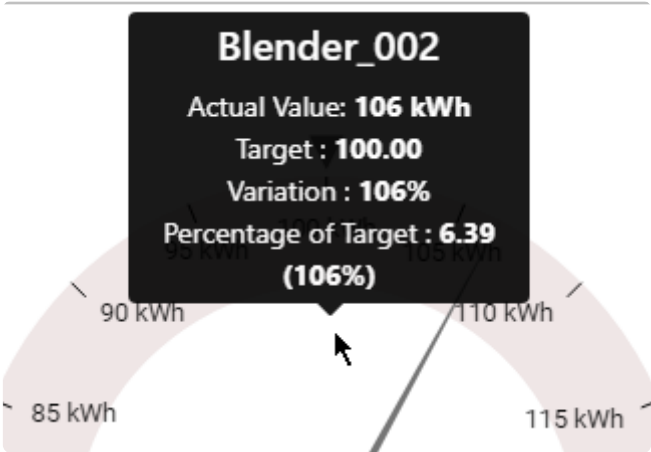


... turns into :





* Hint : when the title is hidden, you can still identify each Gauge by hovering it to display the tooltip :



Last modified: 2019/05/06

2.7.7. Card



The Card Widget can be used to display a single value as text and optionally compare it to a target.

It can also be turned into a container for an indeterminate number of Cards.

Last modified: 2019/05/06

2.7.7.1. Binding Data

The Card has 3 data fields available for binding :

- *Actual Value* : main value displayed on the Card
- *Target* : target displayed in the Card, compared to the Actual Value
- *Series* : how we split series of different Cards in the same container

Show
-

Actual Value

Sum
of
Consumption
X

Sorted
by default

Target Value

Sum
of
Target
X

Sorted
by default

Group by
-

Series

Value
of
Equipment
X

Sorted
ASC

Actual Value

This represents the **primary quantity** we want to display as text on the Card.

This field must be aggregated : you must choose an aggregation function (SUM, AVG, ...)

This field is required.

Show

Actual Value

Sum

of

Consumption

Sorted

by default

X

Consumption

573 kWh

Target Value

This represents the Target we want to display as text on the Card, and will be compared to the Actual Value.

When this field is bound, an additional information is displayed as text : a KPI indicating how the Actual Value compares to the Target Value, which can be configured.

This field must be aggregated : you must choose an aggregation function (SUM, AVG, ...).

This field is optional.

Show

Actual Value

Sum
of
Consumption
Sorted
by default

Target Value

Sum
of
Target
Sorted
by default

Consumption Vs Target

573 kWh

62.98

▲ 112.35%

Series

This represents the series field to generate a different contained Cards for each different value in this Series field.

Each Card will take a portion of size of the widget depending on the container shape. In order to access the other values, the user may need to scroll.

Please see [the corresponding chapter to configure the Card as Widget container](#).



Note :

This only works if your series are values into a single data field.

Example : If you have a field called “Equipment”, you may want to plot a completely separate Card for each value of Equipment :

Show

Actual Value

Sum

of

Consumption

Sorted

by default

Target Value

Sum

of

Target

Sorted

by default

Group by

Series

Value

of

Equipment

Sorted

ASC

<div>Blender_001</div> <div>86 kWh</div> <div>-13.54</div> <div>▼ 86.46%</div>	<div>Blender_003</div> <div>108 kWh</div> <div>7.65</div> <div>▲ 107.65%</div>	<div>Filling_002</div> <div>35 kWh</div> <div>4.74</div> <div>▲ 115.79%</div>
<div>Blender_002</div> <div>106 kWh</div> <div>6.39</div> <div>▲ 106.39%</div>	<div>Filling_001</div> <div>33 kWh</div> <div>2.87</div> <div>▲ 109.57%</div>	<div>Labeling_001</div> <div>36 kWh</div> <div>10.63</div> <div>▲ 142.54%</div>

Last modified: 2019/05/06

2.7.7.2. Value Display

Result interpretation

The font and symbol of the distance to target has a color :

- **green** for “good”
- **red** for “bad”

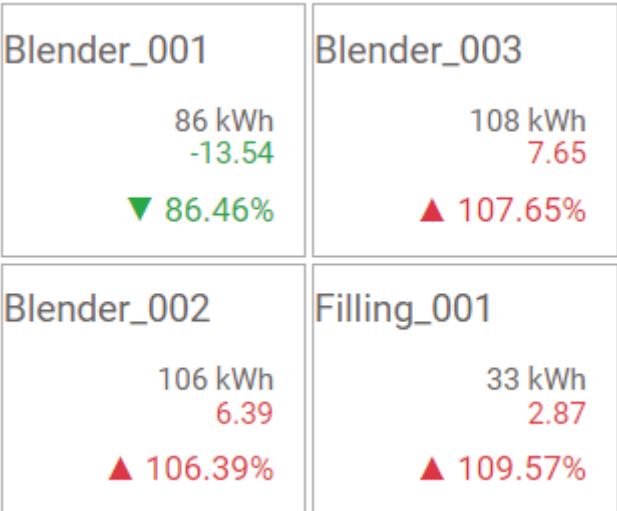
You can define high values as being “good” or “bad” by checking or un-checking the *High Value Is Good* configuration in *Behaviour* :

High Value Is Good ☒

Blender_001 86 kWh -13.54 ▼ 86.46%	Blender_003 108 kWh 7.65 ▲ 107.65%
Blender_002 106 kWh 6.39 ▲ 106.39%	Filling_001 33 kWh 2.87 ▲ 109.57%

... reverts into :

High Value Is Good ☐




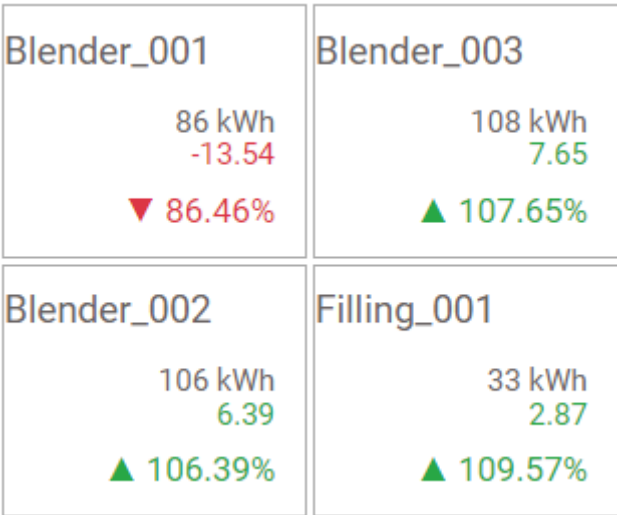
Full detail or KPI only

By default, when the Card has a Target Value, it displays both values as text, as well as a KPI with its value as text.


It is also possible to instead simplify this display and have only the KPI in a very readable way.

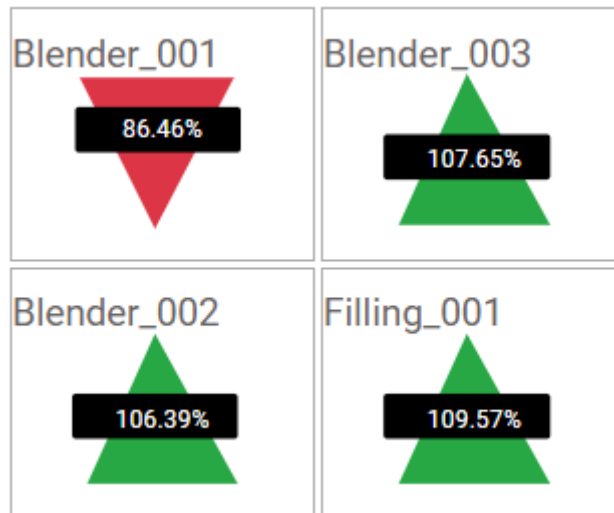
For this, select *Show Indicator Only* under *Behaviour* :

Show Indicator Only 



... turns into :

Show Indicator Only 



Note :

This setting is only available when a Target Value is configured.

KPI Calculation

What calculation is displayed for the KPI can be configured using the *Primary Value Type* selector under *Behaviour* :

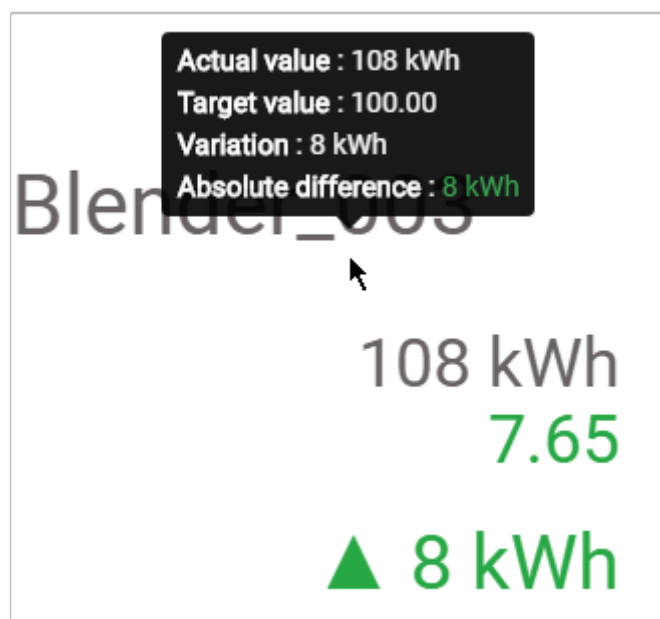
Primary Value Type

Percent of target
 Absolute difference
 Percent of difference
 Percent of target
 Actual Value

Absolute difference

Shows the difference between the Value and the Target.

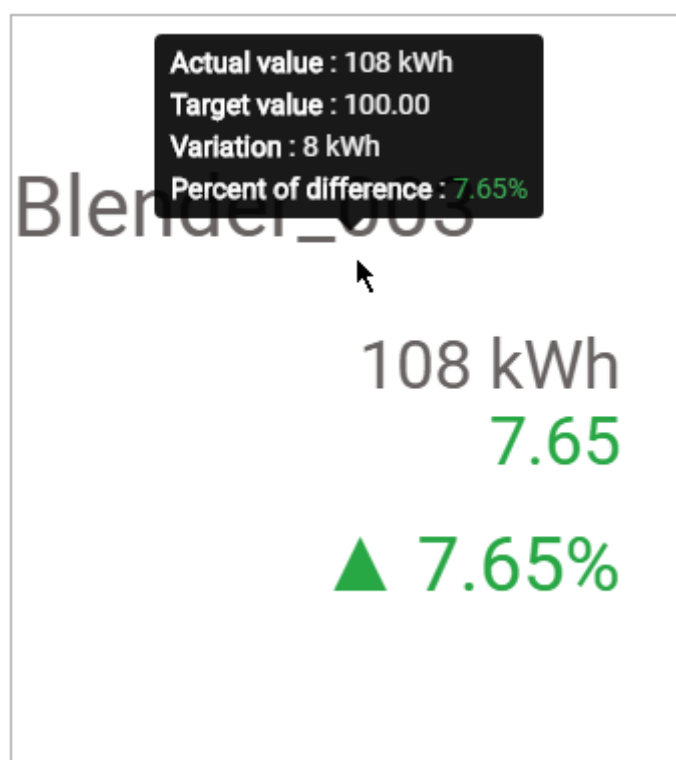
Calculation = Value - Target



Percent of difference

Shows the % of variation to target.

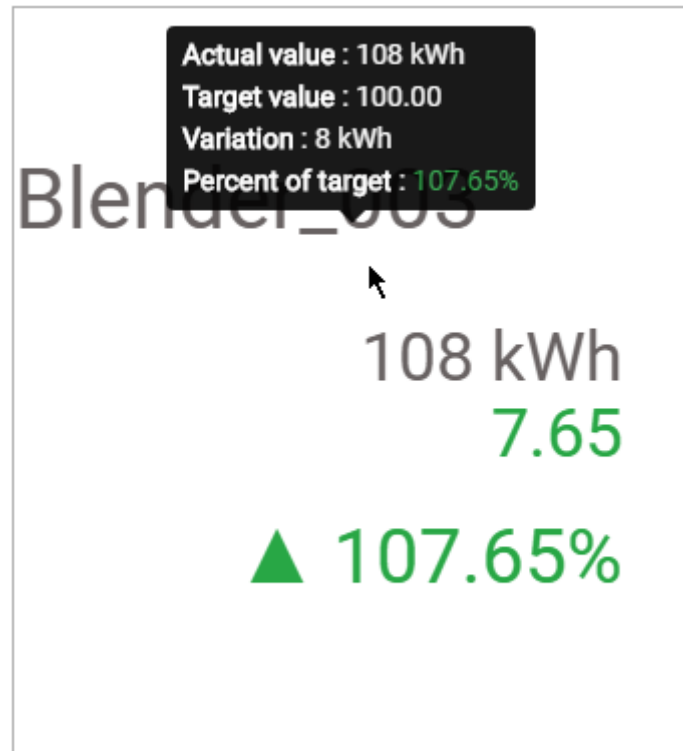
Calculation = $100 * (\text{Value} - \text{Target}) / \text{Target}$



Percent of Target (default)

Shows the % of value to target (hits is the default configuration).

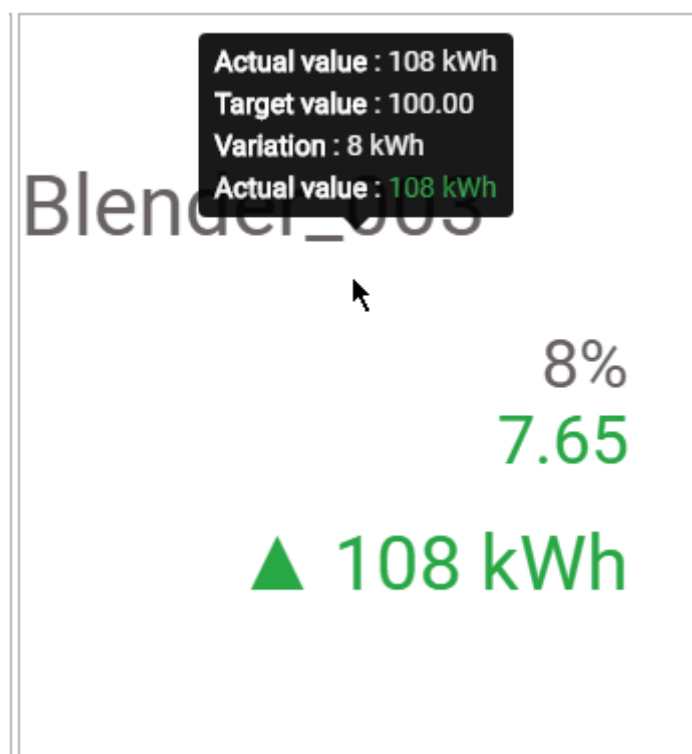
```
Calculation = 100 * Value / Target
```



Actual Value

Shows the Actual Value itself. This can be useful in configuration “Show Indicator Only”.

```
Calculation = Value
```



Last modified: 2019/05/06

2.7.7.3. Style and appearance

Titles

In addition to the [Widget Header](#) that all Widgets have, Cards can have a Title and a Subtitle :

Header

Header

Description

description

Basic Settings

-

Title ☒ Main Title

Title Alignment Left

Title Color

SubTitle ☒ Subtitle

SubTitle Foreground

Value Alignment Right

Header

i

description

Main Title

Subtitle

12%

62.98

▲ 573 kWh

Main Card Title

For single Card

By default, for a single Card (no Series), the Card title is automatically generated from the Actual/Target data field names, but can be customized with the Title text box :

Show

Actual Value

Sum

of

Consumption

Sorted

by default

Target Value

Sum

of

Target

Sorted

by default

Title

☒

Actual Vs Target

Consumption Vs Target

572.98

62.98

▲ 112.35%

... can be customized into :

Title

☒

Energy against Objective

Energy against Objective

573 kWh
62.98
▲ 112.35%

For multiple Cards (Series)

For multiple Cards (with Series), the Card title is automatically generated from the Series data **field values**.

Those cannot be changed.

Sub-Title

A Sub-Title can also be configured, including for Series :

SubTitle ☒ equipment

Blender_001 equipment 86.46 -13.54 ▼ 86.46%	Blender_003 equipment 107.65 7.65 ▲ 107.65%
Blender_002 equipment 106.39 6.39 ▲ 106.39%	Filling_001 equipment 32.87 2.87 ▲ 109.57%

Titles Style

Title Alignment and Color can be customized for Cards :

Title Alignment

Center

Title Color

SubTitle

✓

Subtitle

SubTitle Foreground



Value Style

Value Alignment and Color can be customized for Cards :

Value Alignment

Left

Actual Value Foreground



Last modified: 2019/05/06

2.7.8. Proportion Charts



Proportion Charts allow to see the relative importance of a quantity in different context values.

Proportion Charts are split into several Widget types, but it's possible to change type after configuring the widget :

- Pie
- Doughnut (Donut)
- Pyramid
- Funnel

They all have the same properties documented in this Chapter.

Last modified: 2019/05/15

2.7.8.1. Binding Data

Introduction

The following fields are available for configuring data binding on a Proportion Chart :

Show

Pie part

Part ☐ GoodParts

Sum

of

GoodParts

Sorted

by default


Vs

Sum

of

Add optional field...

As

 Pie

Group by

Use drill down ☐

Split into the chart

Value

of

Equipment

Sorted

ASC

Split into many charts

Value

of

Year

Sorted

ASC

- *Part* : the size of the portion (pie slice, doughnut slice, etc)
- *Split into the chart* : how we group values together to create separate portions
- *Split into many charts* : how we split series of different charts in the same container

Parts

This represents the *quantity* we want to display on the Proportion Chart, and will bind to the size of the portion (pie slice, doughnut slice, etc).

These fields **must be aggregated** : you must choose an aggregation function (SUM, AVG, ...)

At least one *Part* field must be configured, but it's also possible to configure several.

When a single *Part* field is used, the portions are separated by the values in the *Split into the chart* fields (see below).

When binding multiple *Part* fields at the same time, 2 behaviors are possible :

Multiple *Part* fields : multiple portions

If there is no *Split into the chart* field, each Value field will generate a portion :

Pie part

Part ☐ GoodParts ✕

Sum ▼ of GoodParts ▼

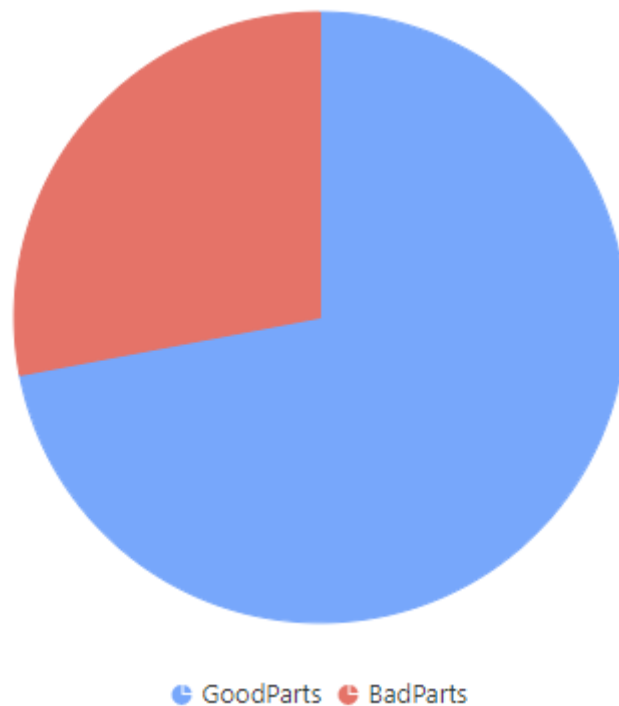
Sorted by default ▼

Vs

Part ☐ BadParts ✕

Sum ▼ of BadParts ▼

Sorted by default ▼



Multiple *Part* fields with *Split into the chart* : multiple series

If a *Split into the chart* field is bound, each *Part* field will generate a separate Proportion Chart :

Show

Pie part

Part

☐

GoodParts

Sum

of

GoodParts

Sorted

by default

Vs

Part

☐

BadParts

Sum

of

BadParts

Sorted

by default


Vs

Sum

of

Add optional field...

As

Pie

Group by

Use drill down

☐

Split into the chart

Value

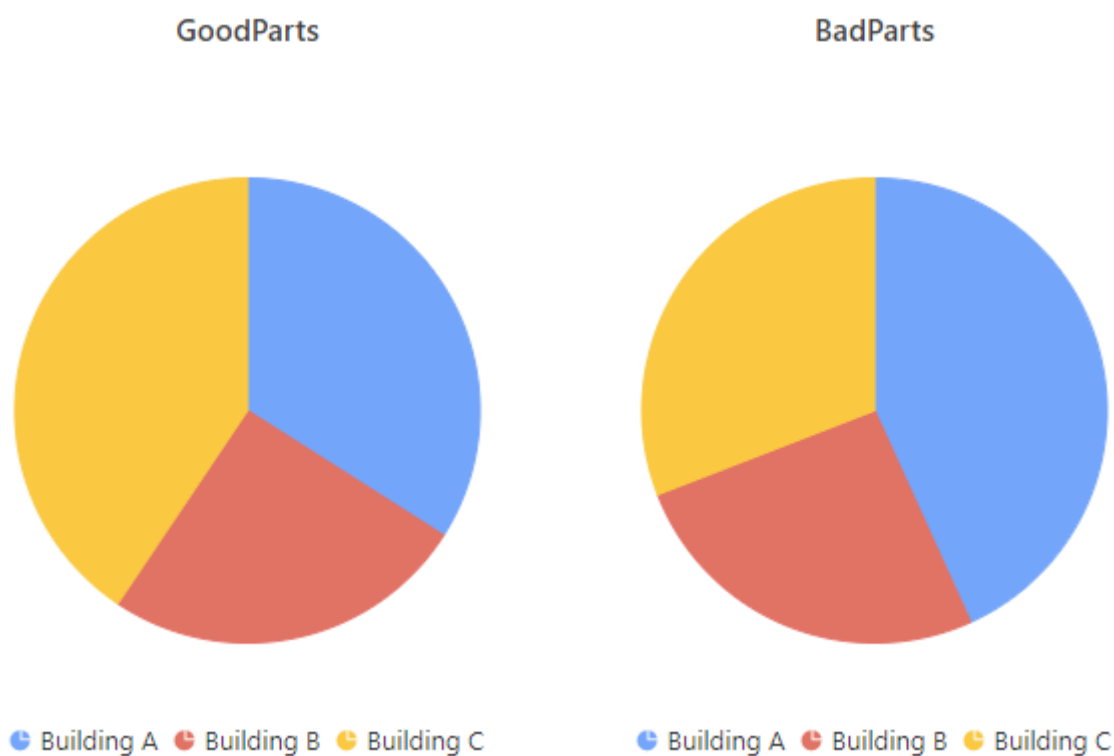
of

Building

Sorted

ASC

Page 478 of 761



Split into the chart

This represents the *category* we want to use to dynamically separate portions on the Pie/Doughnut/etc.

These fields **cannot be aggregated** : they must have several separate values in order to group the “Values” aggregation.

Show

Pie part

Part

GoodParts

X

Sum

▼

of

GoodParts

▼

Sorted

by default

▼

Vs

Sum


▼

of

Add optional field...

▼

As

Pie

▼

Group by

Use drill down ☐

Split into the chart

Value

▼

of

Building

▼

X

Sorted

ASC

▼



Multiple *Split into the chart* fields : Drill-Down

To enable Drill-Down, you can check the box *Use drill down* :

Use drill down ☒

This allows to bind any number of fields to drill-down levels, in the same order as they are in the list.
To change the order of fields, see [the corresponding chapter](#).

Example : With fields called “Building” and “Equipment”, the following configuration will configure the corresponding drill-down :

Group by

Use drill down

✓

Drill down levels

Value

▼

of

Building

▼

Sorted

ASC

▼

×

Then

Value

▼

of

Equipment

▼

Sorted

ASC

▼

×

Then

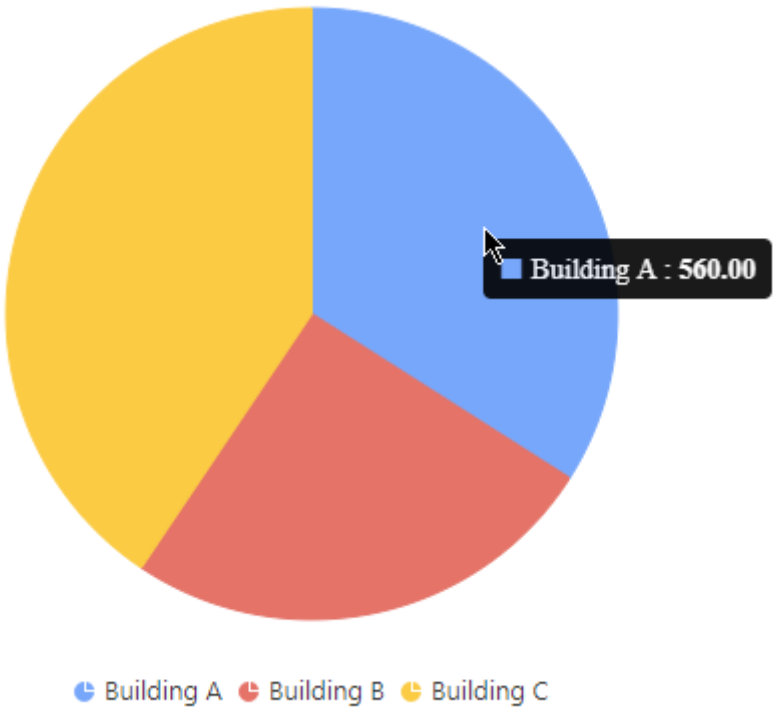
Value

▼

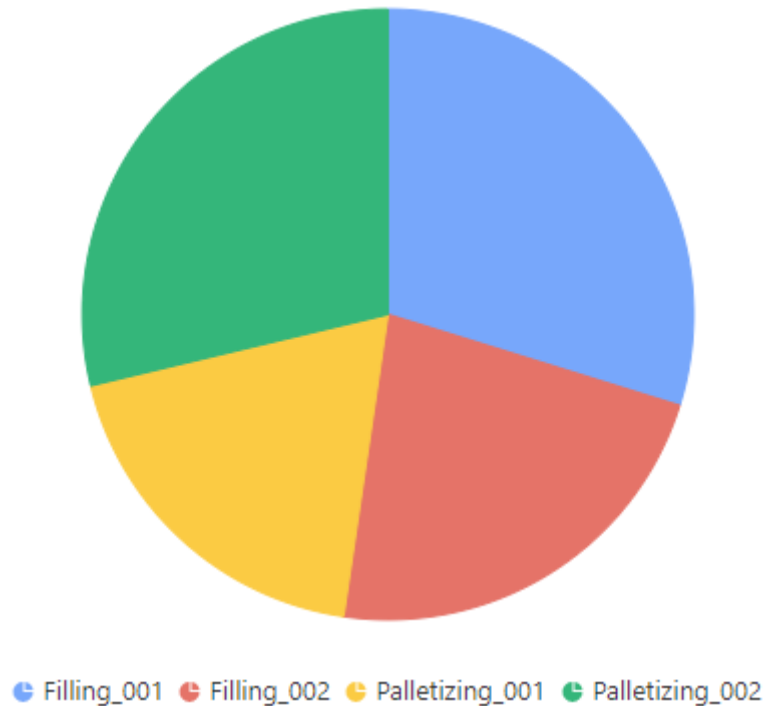
of

Add optional field...

▼



... drills down into :



Split into many charts

This represents the series field to generate a different contained Proportion Chart for each different value in this *Split into many charts* field.

This will turn the widget into a container, and generate a complete different chart for each value in the series field, contained in the Widget.

The layout of the charts inside the container is configurable.

Please see [the corresponding chapter to configure the view of the Widget container](#).

Split into many charts Series with Split into the chart as portions

Example : Your GoodParts are compared by Building. If you have a field called “Year”, you may want to plot a completely separate chart for each value of Year :

Show

Pie part

PartGoodParts

Sum

of

GoodParts

Sorted

by default

Vs

Sum

of

Add optional field...

As

Pie

Group by

Use drill down

Split into the chart

Value

of

Building

Sorted

ASC

Split into many charts

Value

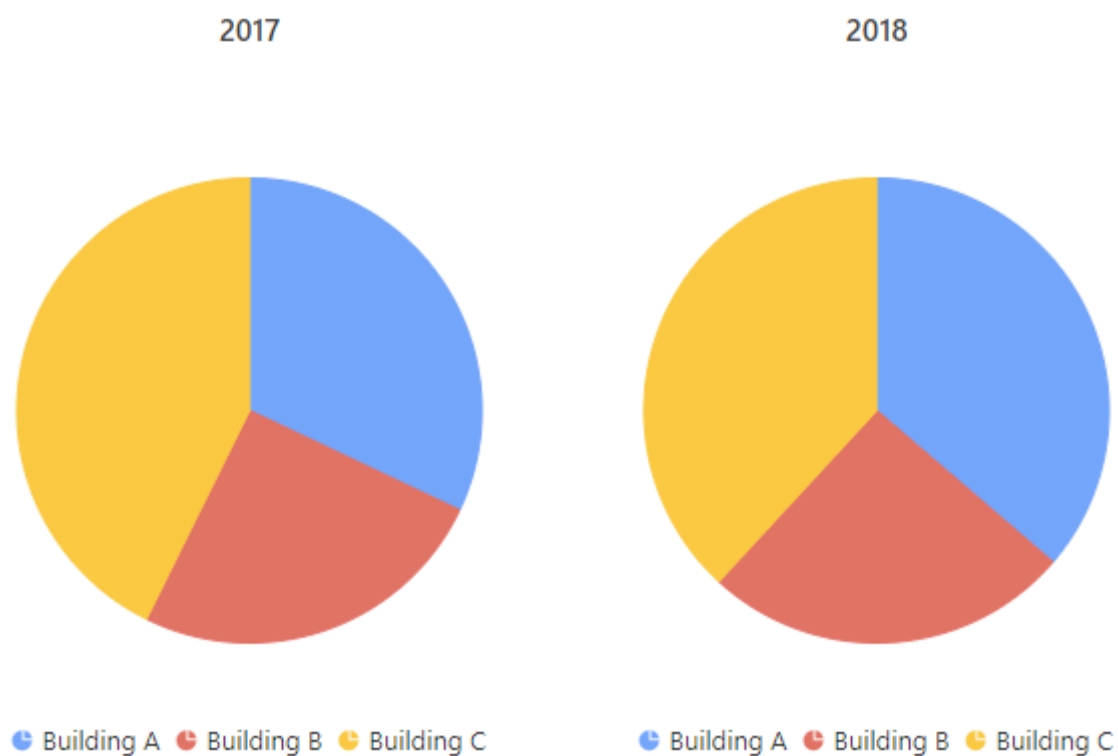
of

Year

Sorted

ASC

Page 484 of 761




Split into many charts Series with multiple *Parts* as portions

Example : Your GoodParts are displayed versus the BadParts. If you have a field called “Building”, you may want to plot a completely separate chart for each value of Building :

Show


Pie part

Part ☐ GoodParts  X

Sum of GoodParts

Sorted by default

Vs

Part ☐ BadParts  X


Sum of BadParts

Sorted by default

Vs

Sum of [Add optional field...](#)

As

 Pie

Group by

Use drill down ☐

Split into the chart

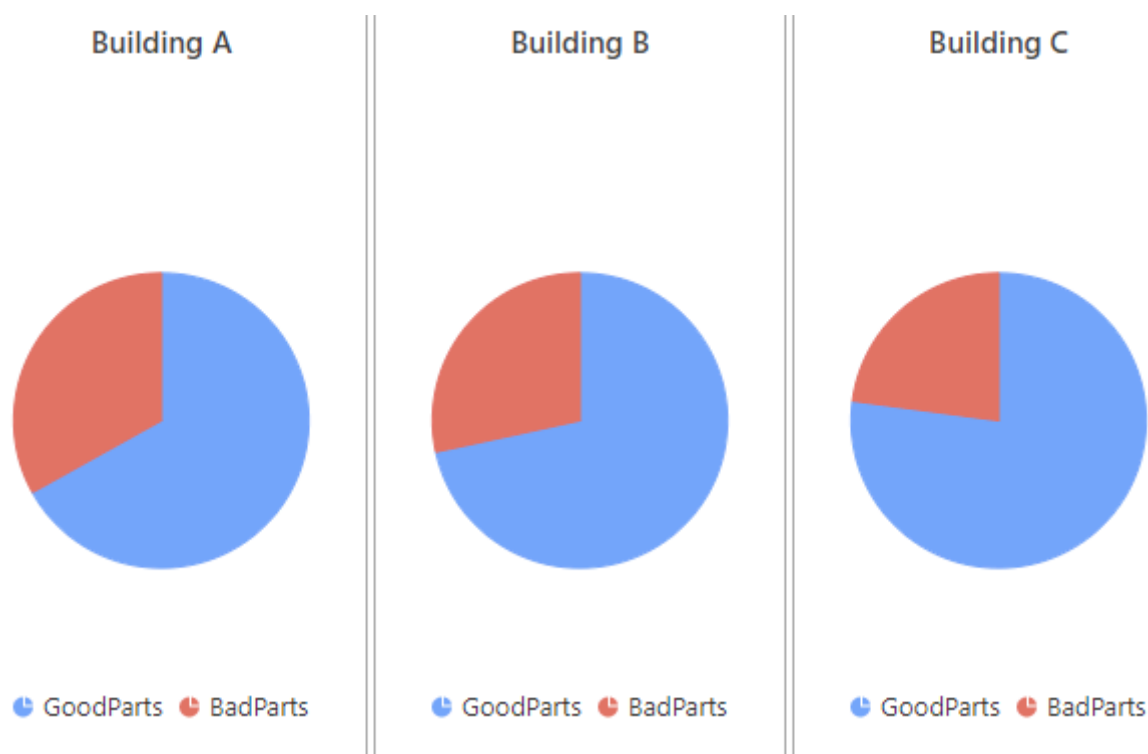
Value of [Add optional field...](#)

Split into many charts

Value of Building X

Sorted ASC

Page 486 of 761



Split into many charts Series multiple Parts with Split into the chart as portions

All three types of fields can also be configured together : each *Part* field will generate a separate chart for each separate value of the *Split into many charts* field, and the portions will be defined by the *Split into the chart* field.

Example : You want a separate chart for the GoodParts of each Year, and a separate chart for the BadParts of each Year, and on each chart you want to compare the Buildings values :

Show

Pie part

Part

☐

GoodParts

Sum of GoodParts

Sorted by default

Vs

Part

☐

BadParts


Sum of BadParts

Sorted by default

Vs

Sum of [Add optional field...](#)

As

 Pie

Group by

Use drill down

☐

Split into the chart

Value of Building

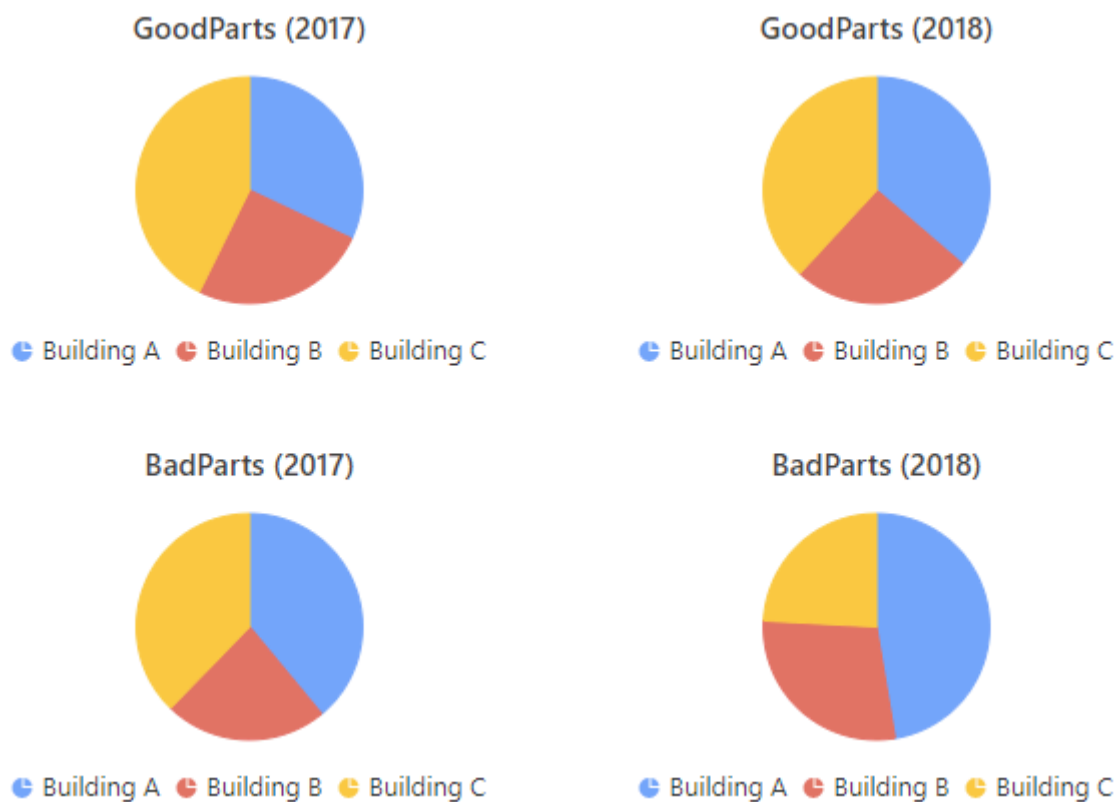
Sorted ASC

Split into many charts

Value of Year

Sorted ASC

Page 488 of 761



***Split into many charts* Series with synchronized Drill-Down**

If Drill-Down is configured, it becomes synchronized between each chart in the series.

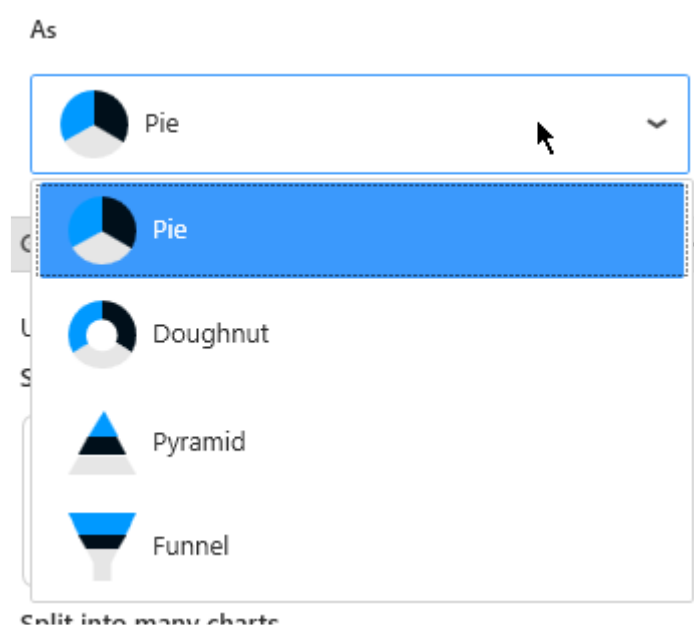
Last modified: 2019/05/16

2.7.8.2. Switching Chart type

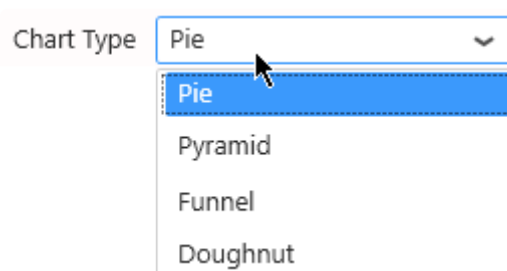
Proportion Chart type

Chart type can be changed on the fly while keeping the widget configuration.

To change the Chart type, use the selector under *As* in the data configuration :



Or use the setting *Chart Type* under *Basic Settings* :

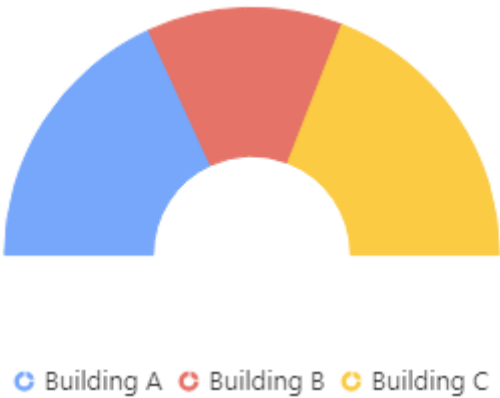


Or use the corresponding shortcut in the top toolbar :

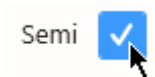


Semi

Pie and Doughnut types also can be displayed as Half-Pie or Half-Doughnut :



For this, check *Semi* under *Basic Settings* :



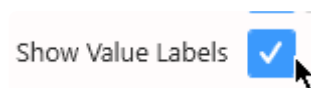
Last modified: 2019/05/15

2.7.8.3. Data Point information

Value Labels

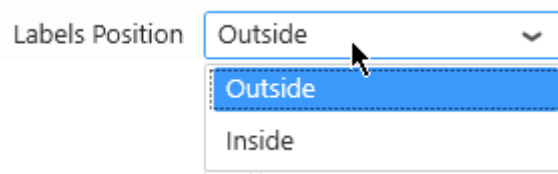
Text labels can be used to display the value next to each portion.

Under *Basic Settings*, check *Show Value Labels* :



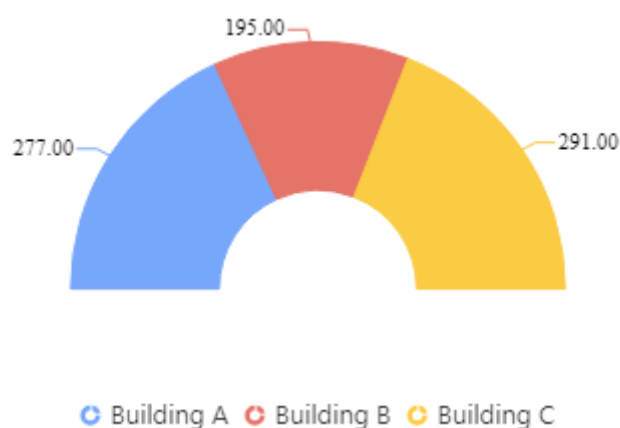
Position

The position of labels can be configured with the *Labels position* selector :



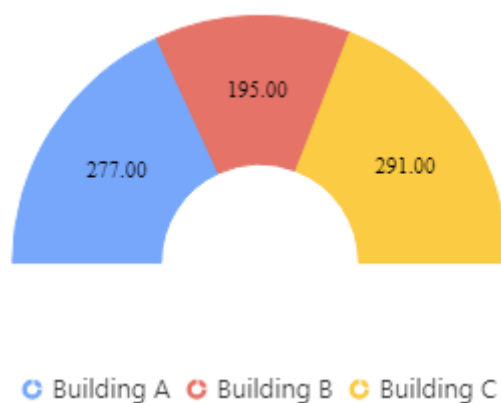
Outside (default)

By default, labels are displayed outside of the portions :



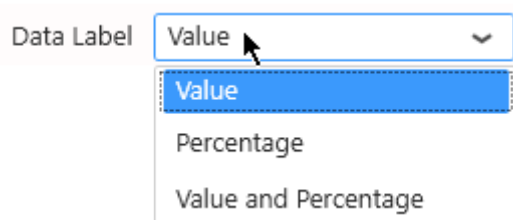
Inside

Labels can also be displayed inside the portion :



Type

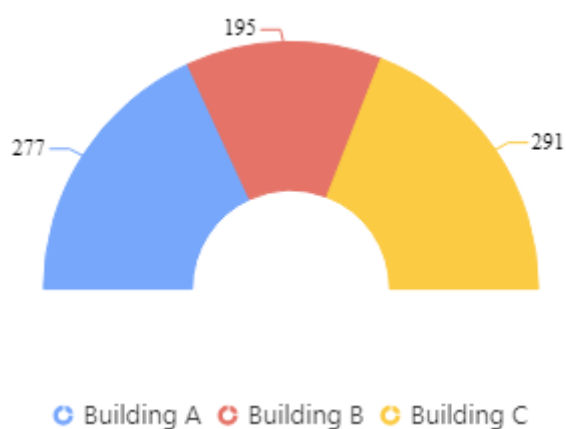
The type of labels can be configured with the *Data Label* selector :



- Value
- Percentage
- Value and Percentage

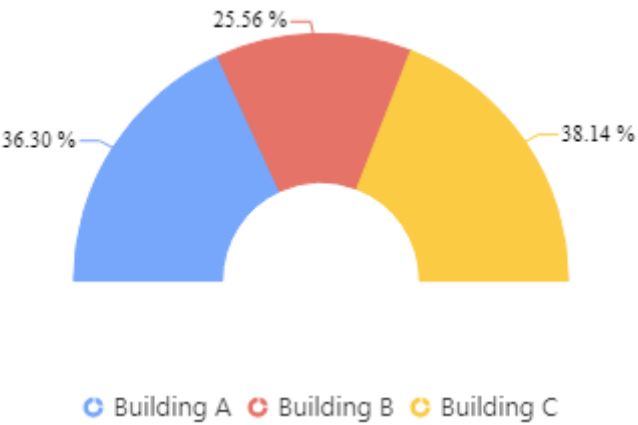
Value

Value labels show the aggregated Value field for the portion :



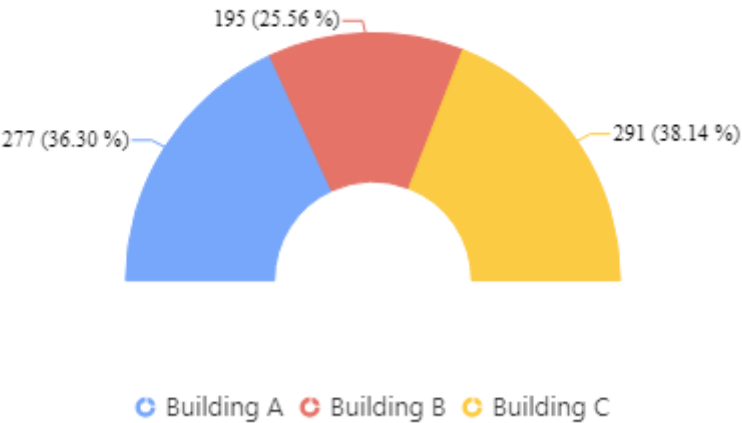
Percentage

Percentage labels show what percentage of the total the portion represents :



Value and Percentage

Value and Percentage can be combined in a single label :



Last modified: 2019/05/15

2.7.8.4. Legend

Portion Legend

The Legend shows the meaning of each portion :

 Building A  Building B  Building C

Hide/Show Legend

The chart Legend can be be hidden or displayed using the corresponding checkbox :

Show Legend ☒

Legend position

The chart Legend can be positioned on any side of the widget :

Show Legend ☒

Bottom

Bottom

Top

Left

Right

Customize Legend of portions

Legend items are automatically generated from the field names and values, but you can customize how they are generated under *Group Settings* :

Group Settings

Split into many charts

Title Format

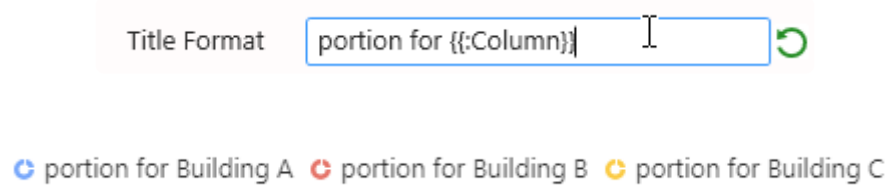
{{:Row}}

Split into the same chart

Title Format

{{:Column}}

With *Split into the same chart*, you can use the placeholder `{{:Column}}` to be replaced by the field value :

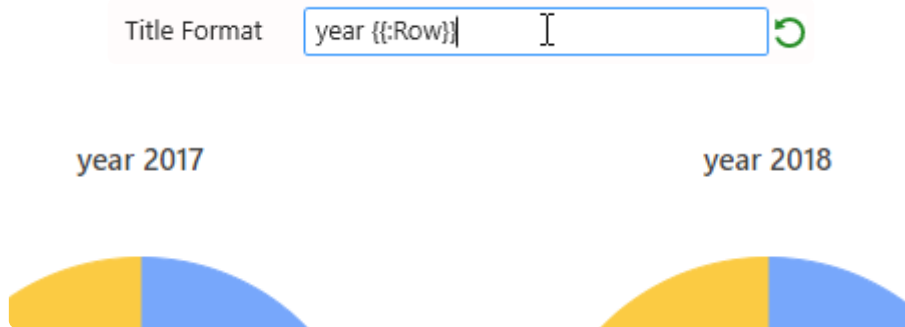


Customize Titles of contained charts

Group Settings also allows to configure the series titles when the Proportion Chart has a field bound to *Split into many charts*.

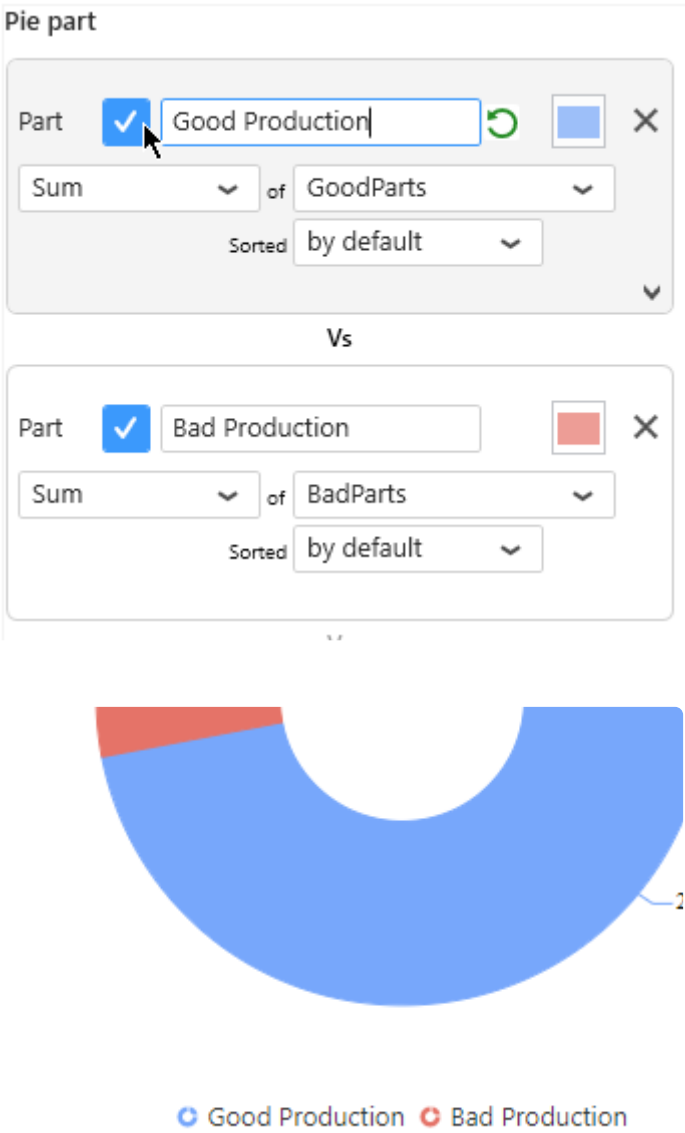
Depending on your data configuration, not all configurations may be available because they wouldn't have meaning.

With *Split into many charts*, you can use the placeholder `{{:Row}}` to be replaced by the field value :



Customize the name of *Part* fields

The legend and titles generated by the *Part* fields can be renamed by checking the text box next to the data field configuration :



Last modified: 2019/05/24

2.7.8.5. Managing Series Color

Indexed Colors (default)

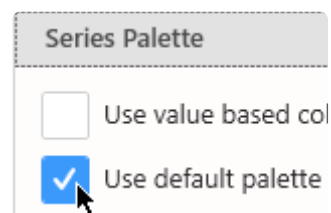
By default, Proportion Chart portions are assigned a color based on their index in the ordered list in which they appear (usually alphabetical order) :

Item1 in the values of the series (see Legend order) is assigned Color1, Item2 is assigned Color2, etc...






The palette has 15 colors. When there are more series items than 15, the colors assigned loop back to Color 1.

Default Palette (default)

To enable the Default Palette (it is enabled by default), check *Use Default Palette* under *Series Palette* :

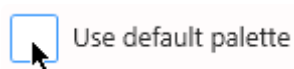


The list of colors in the default palette is below :

Color 1		#77A7FB
Color 2		#E57368
Color 3		#FBCB43
Color 4		#34B67A
Color 5		#F1AB68
Color 6		#6CC2D9
Color 7		#195962
Color 8		#E18878
Color 9		#1AA27D
Color 10		#BFAAC7
Color 11		#89695E
Color 12		#A5C77F
Color 13		#F48FB1
Color 14		#90CAF9
Color 15		#CC4452

Custom Palette

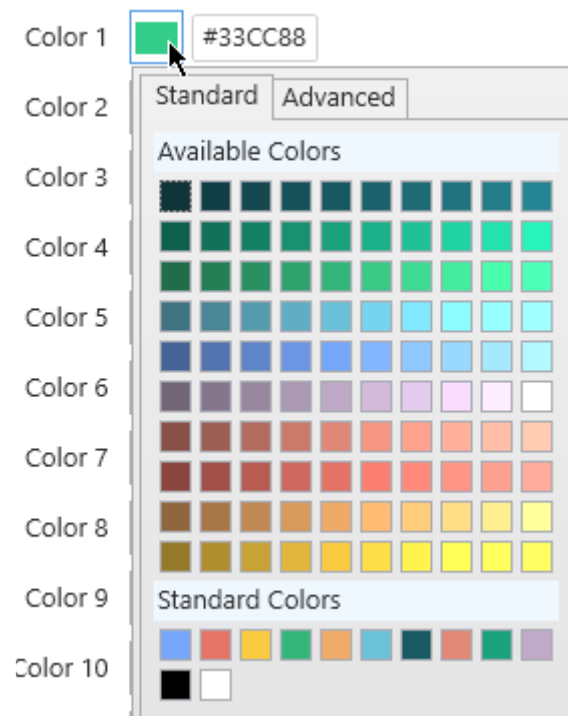
To start customizing the Default Palette, un-check *Use Default Palette* under *Series Palette* :



Then you can customize colors either by entering their hexadecimal code (#RRGGBB) :



... or by clicking the color and using the graphical color picker :



Value-based Colors

What are Value-based colors ?

But what if your series changes over time ?

When items are renamed, added, or deleted from the data, the index of existing items will change, and their color will change !

This could happen simply during filtering at runtime for example.

Instead of index-based, you may then choose **Value-based** colors : the same value in the Series field will always have the same color in the Chart.

For this, simply check *Use value-based colors* under *Series Palette* :



Customizing Value-based colors

Value-based colors are defined **at the Data Source level**.

This is in order to allow re-use of the colors everywhere the same data is used.

To configure value-based colors, see the corresponding chapter [Value-based Settings](#), under [Transforming Data](#).

Last modified: 2019/05/15

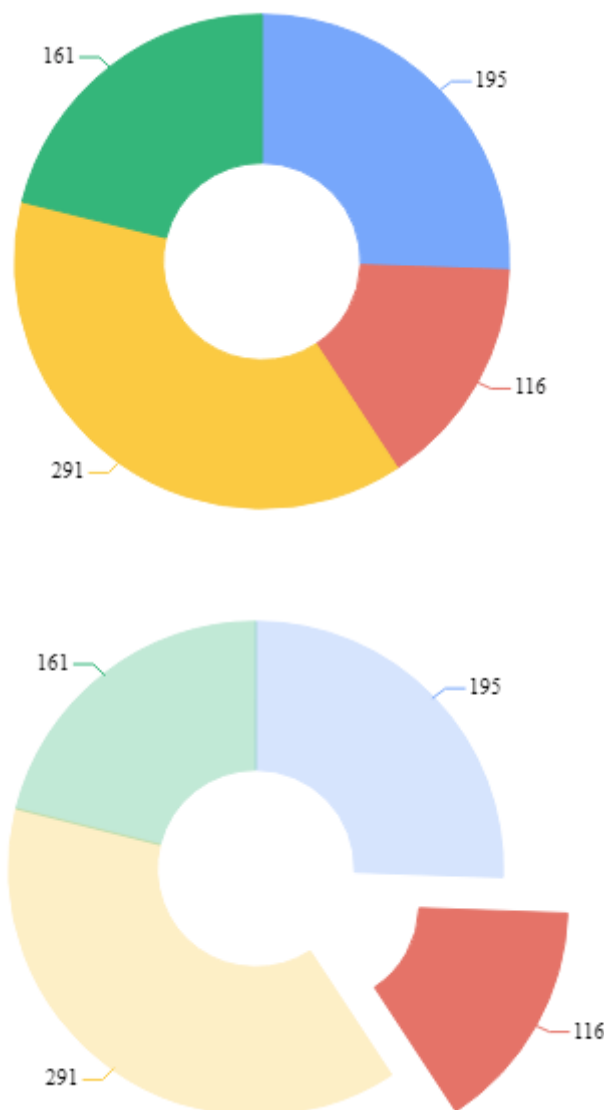
2.7.8.6. Interaction

Explode On Mouse Click

Check *Explode On Mouse Click* under *Series* :

Explode On Mouse Click ☒

At runtime, the user will be able to click on a portion to highlight it by slightly removing it from the chart :



Last modified: 2019/05/15

2.7.8.7. Style and Appearance

Opacity

The chart can be made semi-transparent to better integrate with a background.

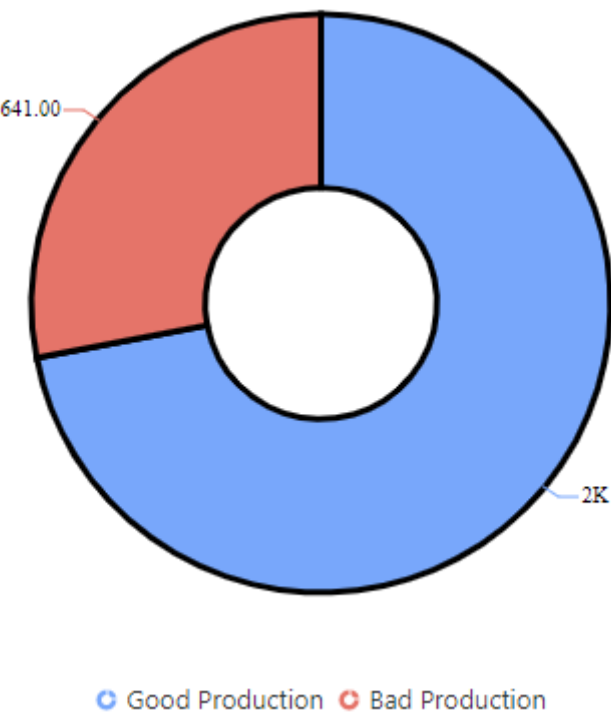
Under *Series*, configure the *Opacity* :



Border

Under *Series*, a *Border Thickness* and *Border Color* can be configured :

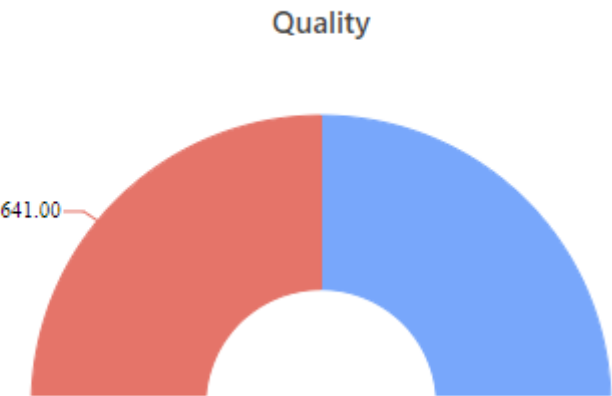




Custom Title

When the chart is not configured to show a series of several charts, a *Custom Title* can be configured under *Basic Settings* :







Custom Title ☒ 



Last modified: 2019/05/15

2.7.9. Filter Widgets

Even though all Widgets can be configured to filter other widgets, some are especially dedicated to filtering :

-  CheckBox
-  ComboBox
-  DatePicker
-  RadioButton
-  RangeNavigator
-  RangeSlider

Last modified: 2019/05/03

2.7.9.1. ComboBox



The Combobox presents multiple values in a foldable box and allows to make single or multiple selections.

Last modified: 2019/05/02

2.7.9.1.1. Binding Data

The Combobox has a 2 data fields for binding :

- *Using* : the value used internally for filtering
- *Displaying* : the value displayed for selection

The screenshot shows the configuration interface for a Combobox. It is divided into two main sections: 'Using' and 'Displaying'. Each section has a title bar with a minus sign. Below the 'Using' title bar, there is a 'Value' dropdown menu, followed by the word 'of', then a text field containing 'aaID' with a dropdown arrow, and a close button 'X'. Below this, there is a 'Sorted' label and a dropdown menu set to 'Asc'. The 'Displaying' section follows a similar layout, with a 'Value' dropdown menu, 'of', a text field containing 'EquipmentName' with a dropdown arrow, a close button 'X', and a 'Sorted' dropdown menu set to 'Asc'.

***Using* : filter value**

This represents the category we want to use internally to filter distinct values.

This field cannot be aggregated : it must have several separate values in order to be used as filter. This is usually an internal unique ID.

This is a required field.

***Displaying* : display value**

This represents the text displayed to the user when viewing the Combobox.

This field cannot be aggregated : it must have several separate values in order to be used as filter. Values can be duplicates because the unique values used for filtering are the ones configured for *Using*. This is an optional field.

✿ If no *Displaying* field is set, the values from the *Using* field are displayed instead.

Last modified: 2019/05/02

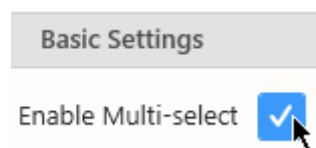
2.7.9.1.2. Configuring behavior

Multiple Selection

Configuring Multiple Selection

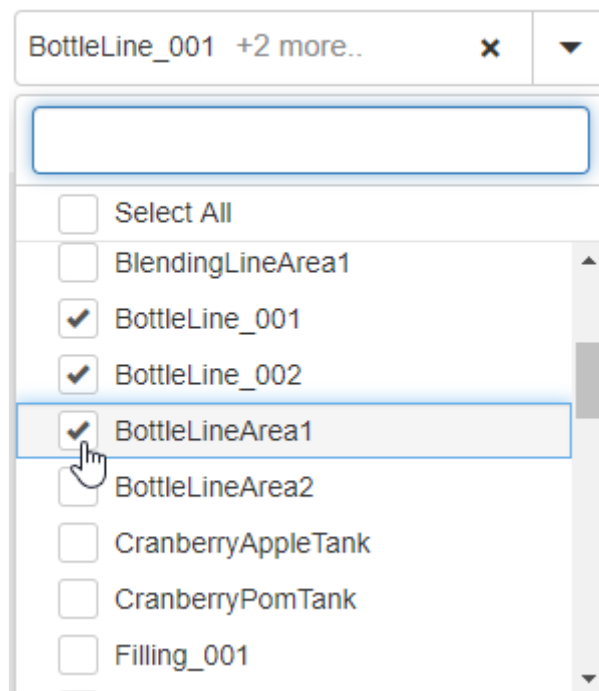
The Combobox can be configured to allow multiple selection.

Check the Enable Multi-Select box under Basic Settings :



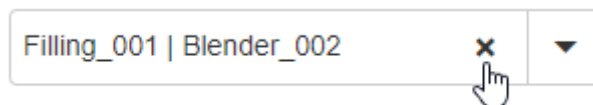
Selecting items at runtime

A checkbox appears in front of every item, allowing multiple selection :



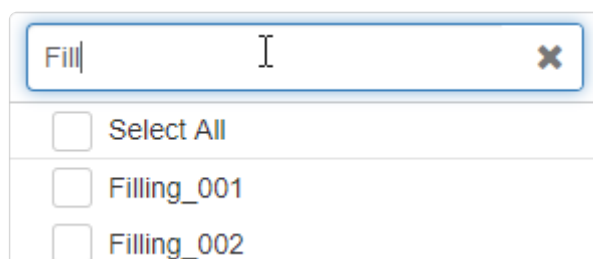
In addition, a new selection item called “*Select All*” (or “*Unselect All*”) is added at the top, to allow to select/unselect all items at once.

Clearing item selection at runtime

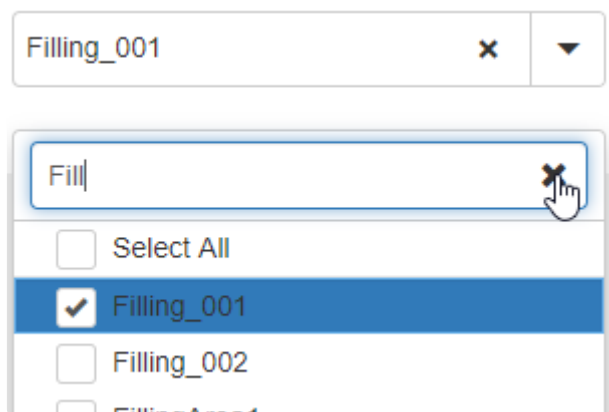


Item search

When Multiple Selection is configured, the Combobox contains a text box that allows the user to search for items



This search filters the content of the Commbobox incrementally and can be cleared using the cross button inside :



Last modified: 2019/05/02


2.7.9.1.3. Style and Appearance

Watermark

Under Basic Settings, the *Watermark* is the text displayed inside the Combobox when no value is selected.

This allows to invite the user to make a selection.

For example :

Watermark 

... at runtime will look like this :

Last modified: 2019/05/02

2.7.9.2. CheckBox



The CheckBox presents multiple values all visible in the widget and allows to make multiple selections.

Last modified: 2019/05/02

2.7.9.2.1. Binding Data

The Checkbox has a 2 data fields for binding :

- *Using* : the value used internally for filtering
- *Displaying* : the value displayed for selection

The screenshot shows a configuration window for a Checkbox. It has two main sections: 'Using' and 'Displaying'. Each section has a title bar with a minus sign. Below the 'Using' title bar, there is a 'Value' dropdown menu, followed by the word 'of', then another dropdown menu containing 'aaID'. To the right of this is a close button (X). Below these is a 'Sorted' dropdown menu set to 'Asc'. The 'Displaying' section follows a similar layout, with a 'Value' dropdown, 'of', a dropdown menu containing 'EquipmentName', a close button (X), and a 'Sorted' dropdown menu set to 'Asc'.

***Using* : filter value**

This represents the category we want to use internally to filter distinct values.

This field cannot be aggregated : it must have several separate values in order to be used as filter. This is usually an internal unique ID.

This is a required field.

***Displaying* : display value**

This represents the text displayed to the user when viewing the Checkbox.

This field cannot be aggregated : it must have several separate values in order to be used as filter. Values can be duplicates because the unique values used for filtering are the ones configured for *Using*. This is an optional field.

✿ If no *Displaying* field is set, the values from the *Using* field are displayed instead.

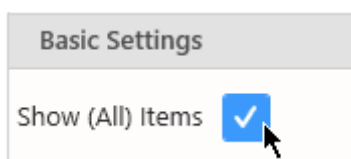
Last modified: 2019/05/02

2.7.9.2.2. Configuring behaviour

Show/Hide “All” item

By default, a item called “(All)” is added at the top, to allow to select/unselect all items at once.

You may show or hide this item from the Show (All) Items check box under Basic Settings :



Last modified: 2019/05/02

2.7.9.2.3. Style and Appearance

Multi-items Viewport

The layout of the checkboxes inside the container is configurable.

Please see [the corresponding chapter to configure the view of the Widget container](#).

Last modified: 2019/05/15

2.7.9.3. RadioButton



The RadioButton presents multiple values all visible in the widget and allows to make single selections.

Last modified: 2019/05/02

2.7.9.3.1. Binding Data

The Radiobutton has a 2 data fields for binding :

- *Using* : the value used internally for filtering
- *Displaying* : the value displayed for selection

The screenshot shows a configuration interface for a Radiobutton. It consists of two main sections: 'Using' and 'Displaying'. Each section has a title bar with a minus sign. Below the 'Using' title bar, there is a 'Value' dropdown menu, followed by the word 'of', then another dropdown menu containing 'aaID'. To the right of this is a close button (X). Below these is a 'Sorted' dropdown menu set to 'Asc'. The 'Displaying' section follows a similar pattern, with a 'Value' dropdown, 'of', a dropdown menu containing 'EquipmentName', a close button (X), and a 'Sorted' dropdown menu set to 'Asc'.

***Using* : filter value**

This represents the category we want to use internally to filter distinct values.

This field cannot be aggregated : it must have several separate values in order to be used as filter. This is usually an internal unique ID.

This is a required field.

***Displaying* : display value**

This represents the text displayed to the user when viewing the Radiobutton.

This field cannot be aggregated : it must have several separate values in order to be used as filter. Values can be duplicates because the unique values used for filtering are the ones configured for *Using*. This is an optional field.

✿ If no *Displaying* field is set, the values from the *Using* field are displayed instead.

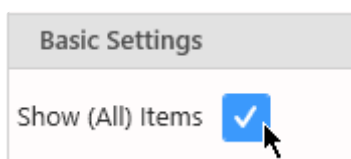
Last modified: 2019/05/02

2.7.9.3.2. Configuring behaviour

Show/Hide “All” item

By default, a item called “(All)” is added at the top, to allow to select/unselect all items at once.

You may show or hide this item from the Show (All) Items check box under Basic Settings :



Last modified: 2019/05/02

2.7.9.3.3. Style and Appearance

Multi-items Viewport

The layout of the radiobuttons inside the container is configurable.

Please see [the corresponding chapter to configure the view of theWidget container](#).

Last modified: 2019/05/15

2.7.9.4. RangeSlider

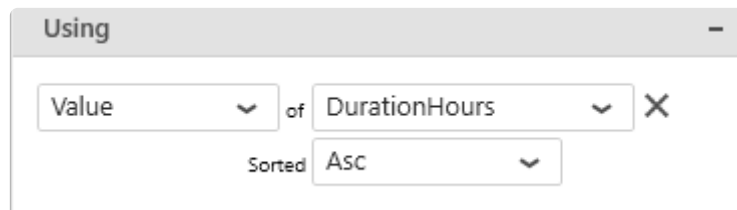


The Range Slider presents a slider and allows to select number ranges or single numbers, as well as dates.

Last modified: 2019/05/03

2.7.9.4.1. Binding Data

The Range Slider has a single data field for binding : *Using*



Using : filter value and display

This represents the category we want to use internally to filter distinct values.

This is also the text labels displayed to the user when viewing the RangeSlider

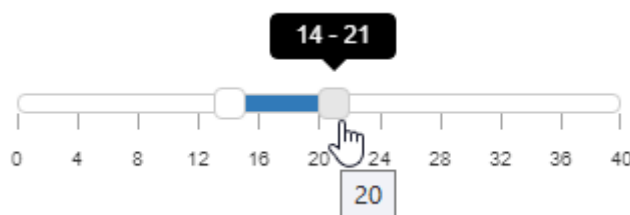
This field cannot be aggregated because it must have several separate values in order to be used as filter.

This is a required field.

Several data types are supported :

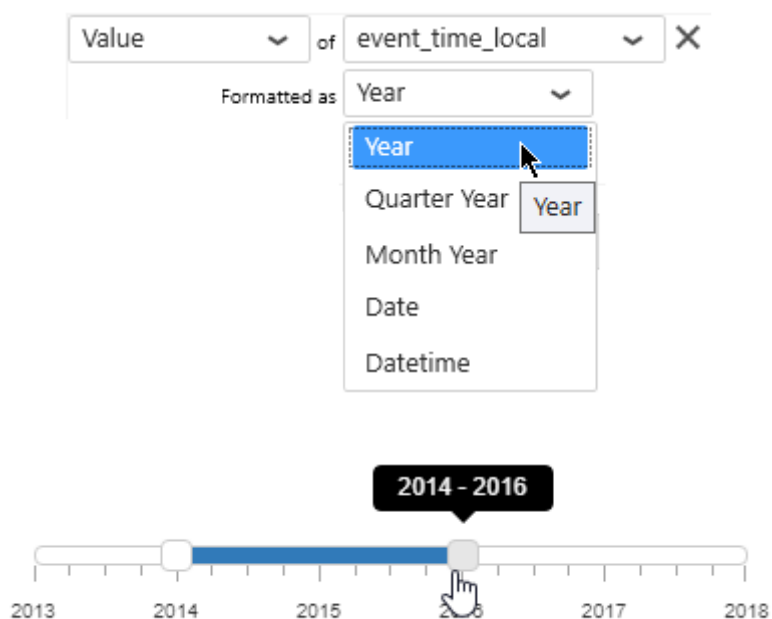
Number field

With a number field, the Range Slider displays numbers in order between the minimum value and the maximum value, and allows to select values :



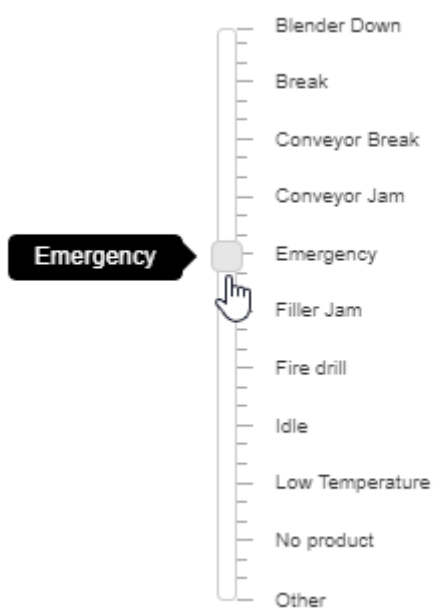
Date/Time field

With a date or time field, the Range Slider allows to select time intervals depending on the Date/Time formatting selected :



String field

The Range Slider could also be used to select strings, especially if using sorting (alphabetically) :



Last modified: 2019/05/03

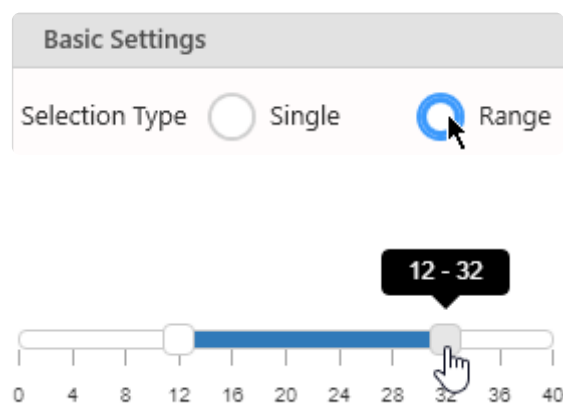
2.7.9.4.2. Configuring behaviour

Range Selection

Range

The Range Slider can be configured to select a range of values (default).

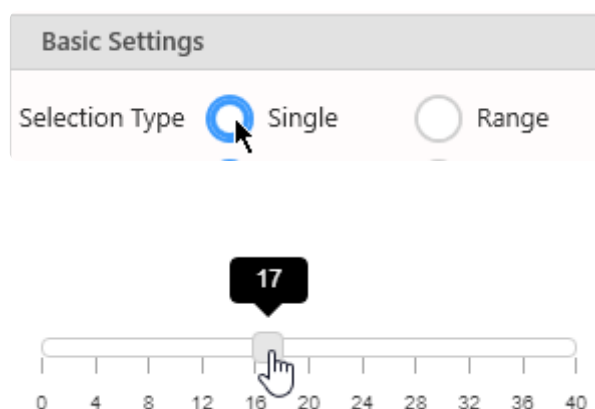
Under Basic Settings > Selection Type, select Range :



Single

The Range Slider can instead be configured to select a single value.

Under Basic Settings > Selection Type, select Single :



Last modified: 2019/05/03

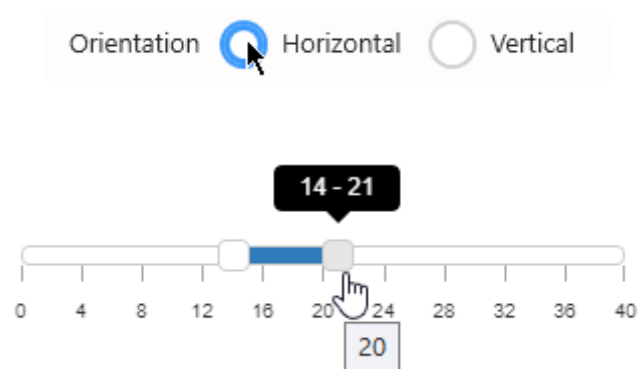
2.7.9.4.3. Configuring style

Orientation

Horizontal

The Range Slider can be oriented horizontally (default).

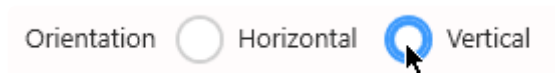
Under Basic Settings > Orientation, select Horizontal :

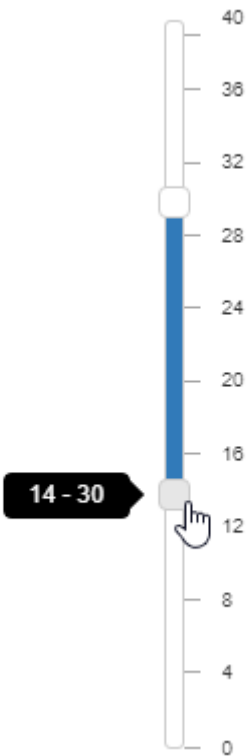


Vertical

The Range Slider can be oriented vertically.

Under Basic Settings > Orientation, select Vertical :





Last modified: 2019/05/03

2.7.9.5. DatePicker

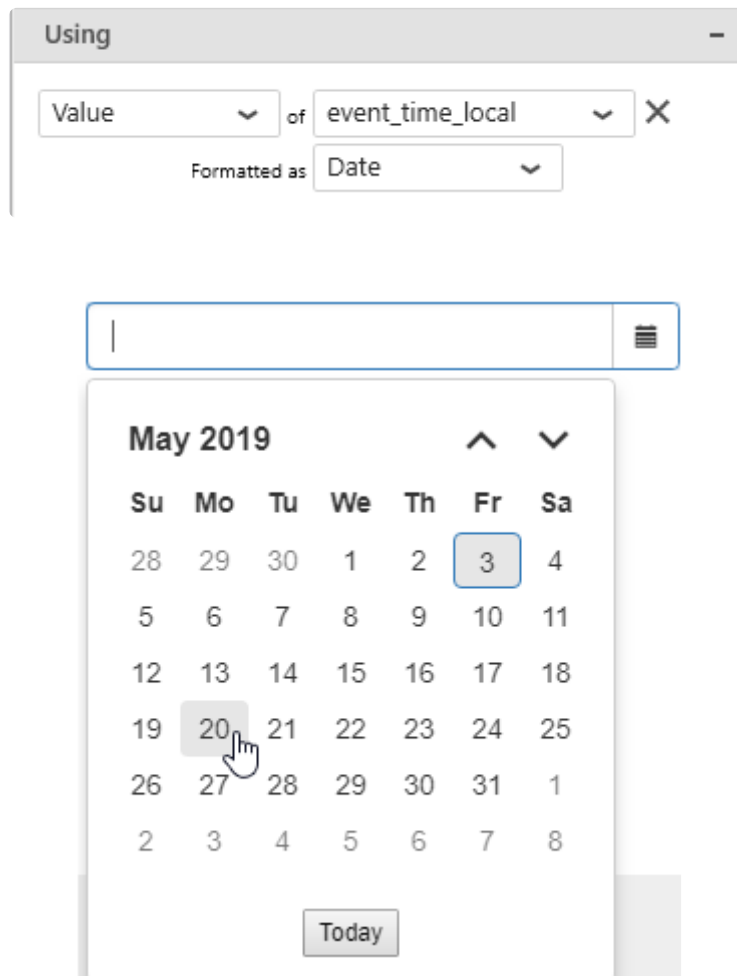


The DatePicker presents a calendar and allows to make single or multiple date selections.

Last modified: 2019/05/03

2.7.9.5.1. Binding Data

The DatePicker has a single data field for binding : *Using*



***Using* : filter value and display**

This represents the date field we want to use to present all the available distinct dates.

This field cannot be aggregated because it must have several separate values in order to be used as filter.

This is a required field.

The range of available dates will be the range between the oldest and the latest date in the date field.

Last modified: 2019/05/03

2.7.9.5.2. Configuring behaviour

Range Selection

The DatePicker can be configured to allow range selection.

Configure the Selection Type to Range :

Basic Settings

Selection Type ☐ Single ☒ Range

The calendar can now show two pages at the same time, to make it easier to select a Start Date and an End Date :

DatePicker

May 9, 2019 → Jun 17, 2019
40 days

< May 2019 >

Su	Mo	Tu	We	Th	Fr	Sa
28	29	30	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	1

< June 2019 >

Su	Mo	Tu	We	Th	Fr	Sa
26	27	28	29	30	31	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	1	2	3	4	5	6

Cancel

Apply

Time Selection

The Date Picker can be configured to allow time selection in addition to date.

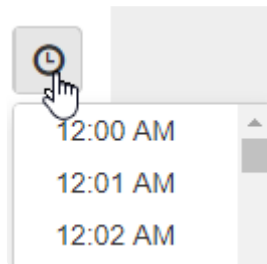
* If you configure *Time* selection with *Single* selection, the filter will try to match only the data that has the exact same minute. This usually gives no result, and you may want instead to use the *Range* selection with *Time*.

Enable Time selection

Select *Enable Time Value* :

Enable Time Value ☒

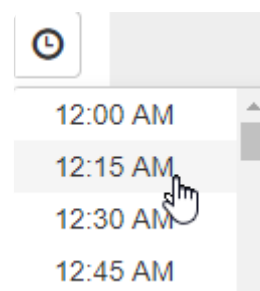
At runtime, this adds a time selector (or two, if Range is configured) :



Configure available times

By default, the time selector shows a list of times from midnight to midnight every 1 minute. This can be configured with *Choose Time Interval (minutes)* :

Choose Time Interval (minutes)



Default selection as Relative Date

What is Default selection as Relative Date

Like other Widgets, the DatePicker allows to configure its *default value* by selecting in the preview inside Alpana Designer. This default selection will be a fixed date range.

In addition, it is possible to configure a **relative date range as default selection**. This is described in [the corresponding chapter](#).

Last modified: 2019/08/29

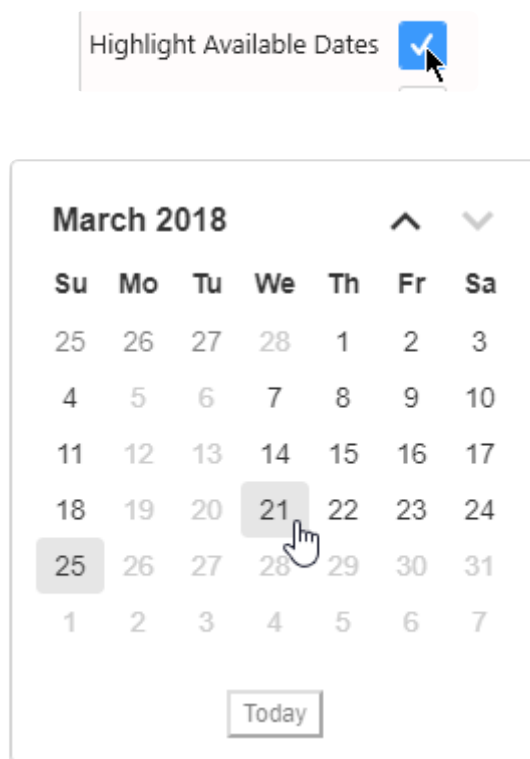
2.7.9.5.3. Configuring style

Highlight Available Dates

By default, all dates in the interval are available for selection. However, sometimes there is no actual data for a given date.

By checking Highlight Available Dates under Basic Settings, you can prevent users from selecting dates that don't have data.

The dates that don't appear in the configured date field are greyed-out and can't be selected :

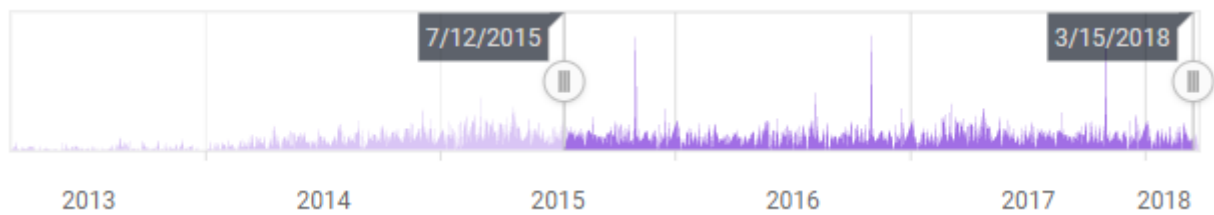


Last modified: 2019/05/03

2.7.9.6. RangeNavigator



The Range Navigator presents a chart as background and allows to make date range selections on top of it.



Last modified: 2019/05/03

2.7.9.6.1. Binding Data

The following field containers are available for configuring data binding on a Range Navigator :

The screenshot shows a configuration window titled 'Show' with a minus sign. It contains two main sections: 'Value' and 'Argument'.

Value Section:

- A dropdown menu set to 'Sum'.
- The word 'of'.
- A dropdown menu set to 'DurationHours'.
- An 'X' icon to clear the selection.
- Below the 'Sum' dropdown, the word 'Sorted' is followed by a dropdown menu set to 'by default'.

Argument Section:

- A dropdown menu set to 'Value'.
- The word 'of'.
- A dropdown menu set to 'event_time_local'.
- An 'X' icon to clear the selection.
- Below the 'Value' dropdown, the text 'Formatted as' is followed by a dropdown menu set to 'Year'.
- Below the 'Formatted as' dropdown, the word 'Sorted' is followed by a dropdown menu set to 'Asc'.

- Value : the height of the background chart
- Argument : the field used to filter data during selection, and also how we group values together to create points on the background chart

Value

This represents the quantity we want to display on the background chart, and will bind for example to the height of the Area (Area chart).

The Value field must be aggregated : you must choose an aggregation function (SUM, AVG, ...)

This is a required field.

Argument

This represents the category we want to use internally to filter distinct values.

This is how we group values together to create points on the background chart.

This field cannot be aggregated because it must have several separate values in order to be used as filter.

This is a required field, and must be a date field.

The date format (Date, MonthYear, ...) is used to group the data to display on the chart, but also during filter selection.

Last modified: 2019/05/03

2.7.9.6.2. Configuring behaviour

Default selection as Relative Date

What is Default selection as Relative Date

Like other Widgets, the RangeNavigator allows to configure its *default value* by selecting in the preview inside Alpana Designer. This default selection will be a fixed date range.

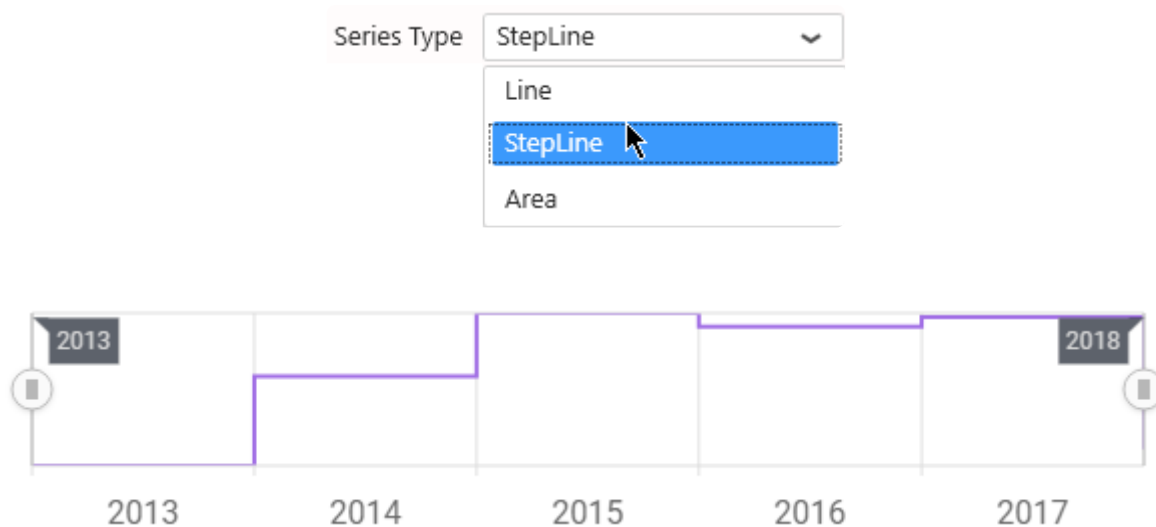
In addition, it is possible to configure a **relative date range as default selection**. This is described in [the corresponding chapter](#).

Last modified: 2019/05/24

2.7.9.6.3. Configuring style

Background Chart style

The style of the background chart can be changed by selecting from the Basic Settings > Series Type list :



Last modified: 2019/05/03

2.7.10. Static Widgets

Alpana allows to create some Widget types that are not bound to data.
The following chapters describe those *Static Widgets*.

Last modified: 2019/05/02

2.7.10.1. Image



The Image Widget allows to display static images on a dashboard, and handles transparency.

This can be useful for displaying a logo or a reference image for example.

Last modified: 2019/06/20

2.7.10.1.1. bind image

To configure, simply browse for a supported image file : .png, .jpg, .bmp, .jpeg, .gif, .emf, .tif, .tiff, .ico

Last modified: 2019/05/02

2.7.10.1.2. Image Modes

Images can be configured with an Image Mode which determines how the image is cropped/stretched to fit the Widget container

Default

In Default mode, the image is simply centered and cropped, but not resized.

✿ If you use a transparent image on the image, the margins will not be visible.





Fill

In Fill mode, the image is stretched to fit the container.

- * This can be useful to apply a background image and add Widgets on top of it : the Widget's position will be constant relative to the image, independently of the web page resizing.



Uniform

In Uniform mode, the image is resized to fit the container without being stretched or cropped. The

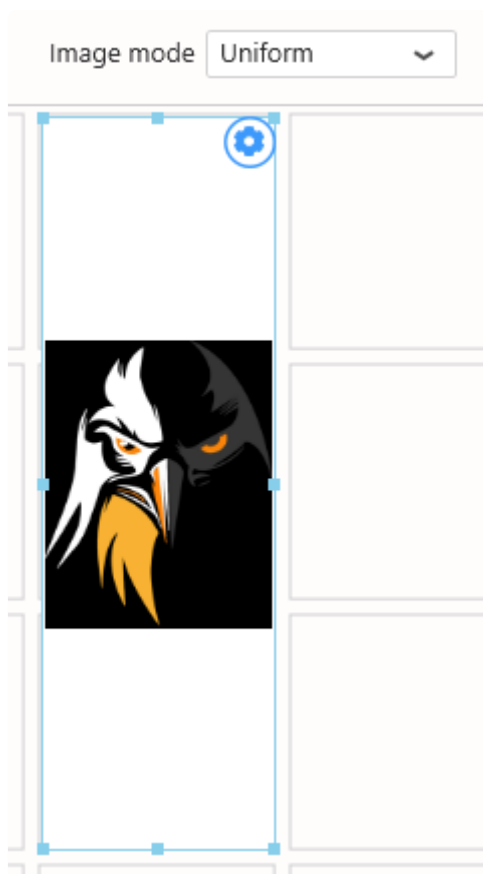
container is not filled completely.

This is useful to make sure the full image is always visible without being stretched.

✿ This is usually desired for a logo.

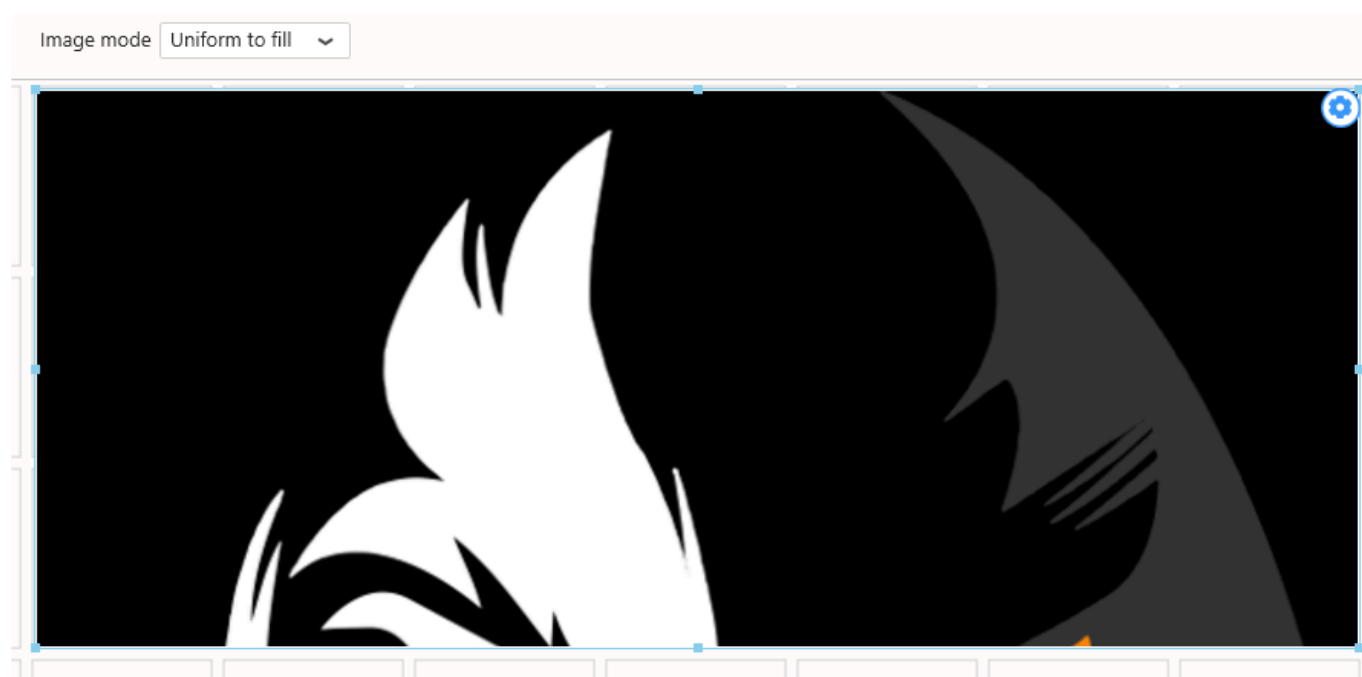
If you use a transparent image on the image, the margins will not be visible.

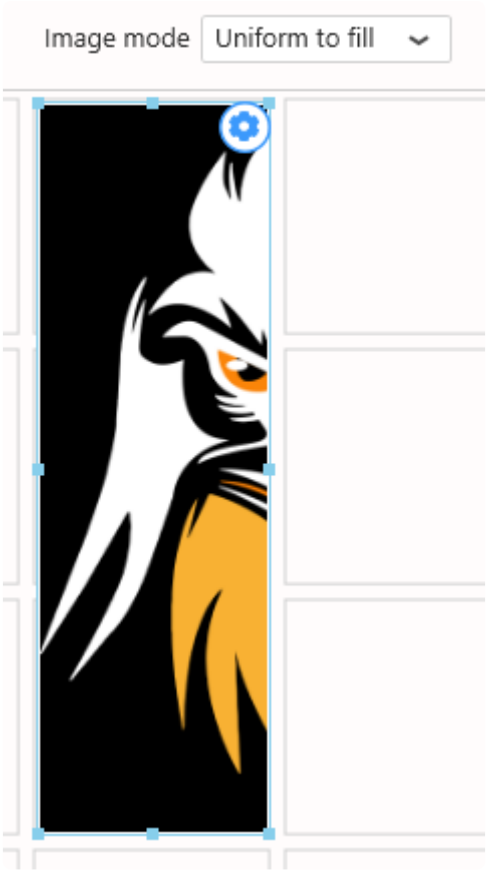




Uniform to fill

In Uniform to fill mode, the image is resized and cropped to fill the container without being stretched.





Last modified: 2019/05/02

2.7.10.2. Label

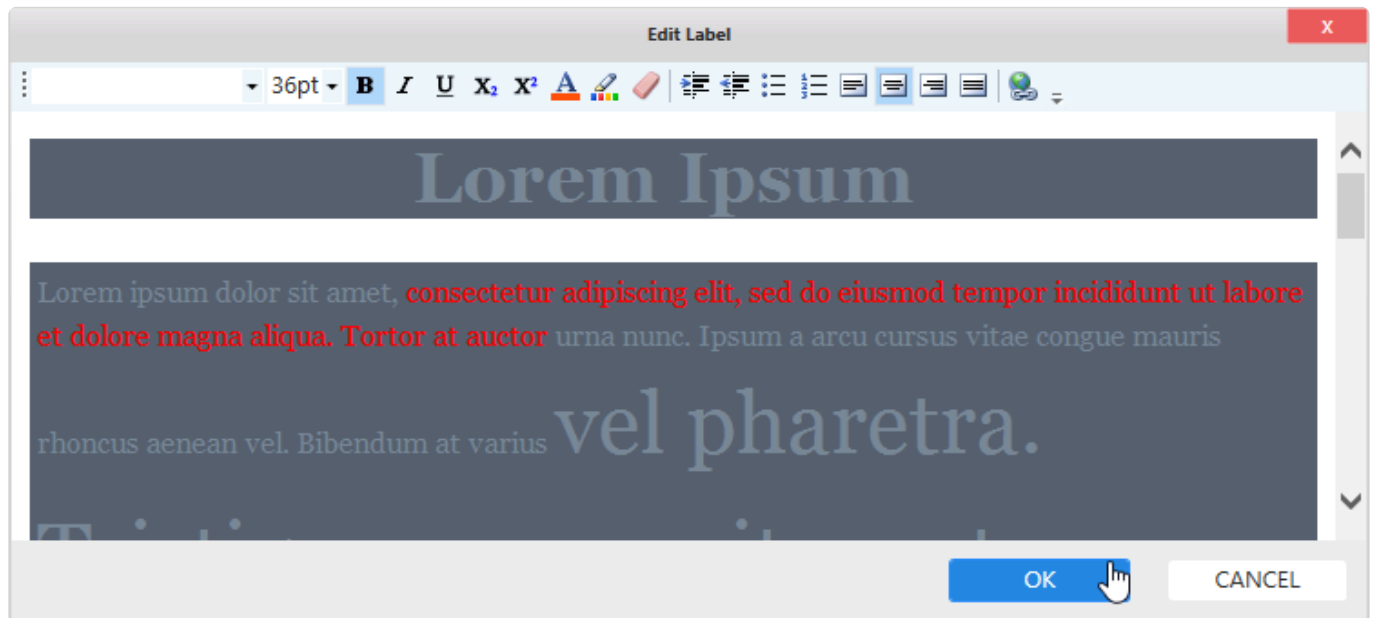


The Label Widget allows to insert rich text into a Dashboard.

Last modified: 2019/05/02

2.7.10.2.1. Configuration

The Label Widget can be configured like any rich text editor, with font type, size, style, color, alignment, bullet lists, hyperlinks, ...



Last modified: 2019/05/02

2.7.10.3. Embed



The *Embed* Widget is a widget dedicated to embedding other content into Alpana :

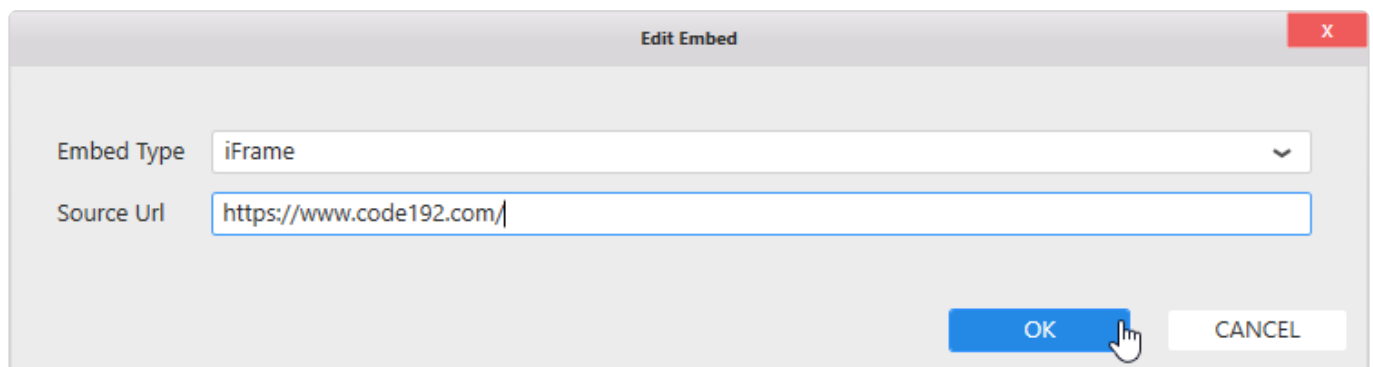
- other websites
- components from other applications (like Widgets from Aveva Insight)
- video or image streams from security cameras
- online documents published on other websites
- other web widgets, like streaming videos, calendars, etc...

Last modified: 2019/05/02

2.7.10.3.1. Configuration

Source Url

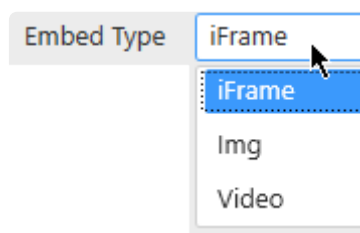
This is the Url that points to the content you wish to embed. Simply paste the Url in this text box.



The screenshot shows a dialog box titled "Edit Embed". It has a close button (X) in the top right corner. Inside the dialog, there are two fields: "Embed Type" with a dropdown menu currently showing "iFrame", and "Source Url" with a text input field containing "https://www.code192.com/". At the bottom right, there are two buttons: "OK" (highlighted with a mouse cursor) and "CANCEL".

Embed Type

The *Embed Type* determines how the content will be embedded :



iFrame

The content will be embedded "as-is".
This is usually the default.

Video

This is for video streams that require to be embedded inside `<video>` HTML markup.
This can be the case for some security cameras.

Img

This is for image streams that require to be embedded inside `` HTML markup.
This can be the case for some security cameras / webcams.

Last modified: 2019/05/02

2.7.10.3.2. Note : Cross-origin resource sharing

Embedding a content from one website (your content) into another website (Alpana) often requires to be allowed from the embedded website (your content).

This is called [Cross-origin resource sharing](#).

For example, the <https://www.google.com/> URL currently refuses to be embedded inside an iframe of another website. This is a normal thing, for security purposes.

In some cases, you can create an exception, please see with the documentation of your content website.

Last modified: 2019/05/02

2.8. Using Dashboard Parameters

Last modified: 2019/05/24

2.8.1. What is a Parameter ?

A **Parameter** is a proxy between actions and results in a *Dashboard*.

What is this for ?

This is the mechanism behind Dashboard Filters, dynamic Stored Procedure calls, URL Parameters, ...

How does it work ?

A Parameter acts as a **variable** : its value can be assigned to and read from.

Listeners can subscribe to update automatically when the Parameter value changes.

A Parameter can take 0, 1 or more values.

A Parameter can listen to one or more Master Widgets to update its value.

Examples

Dynamic filtering

Dashboard Widgets can listen to Parameters :
when the Parameter changes, the Widget is **filtered** accordingly.

Dynamic data calls

Stored Procedures can also be set by Parameters :
when the Parameter changes, the Stored Procedure is updated accordingly.

Getting values from the URL

Parameters values can also be given through URLs to change their default value.

Automatically generated Parameters

In some cases, Parameters are created automatically : see Chapter [Widget as Master Filter](#).

Last modified: 2019/05/24

2.8.2. Managing Parameters

Parameters are configured and managed in a central menu in the Parameters Editor.

Read the next chapters to know more.

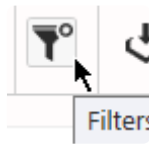
Last modified: 2019/05/24

2.8.2.1. Opening the Parameters Editor

How to open

From the Dashboard activity tab

In the top bar, use the Edit Filters icon :



From a “Use Parameter” configuration

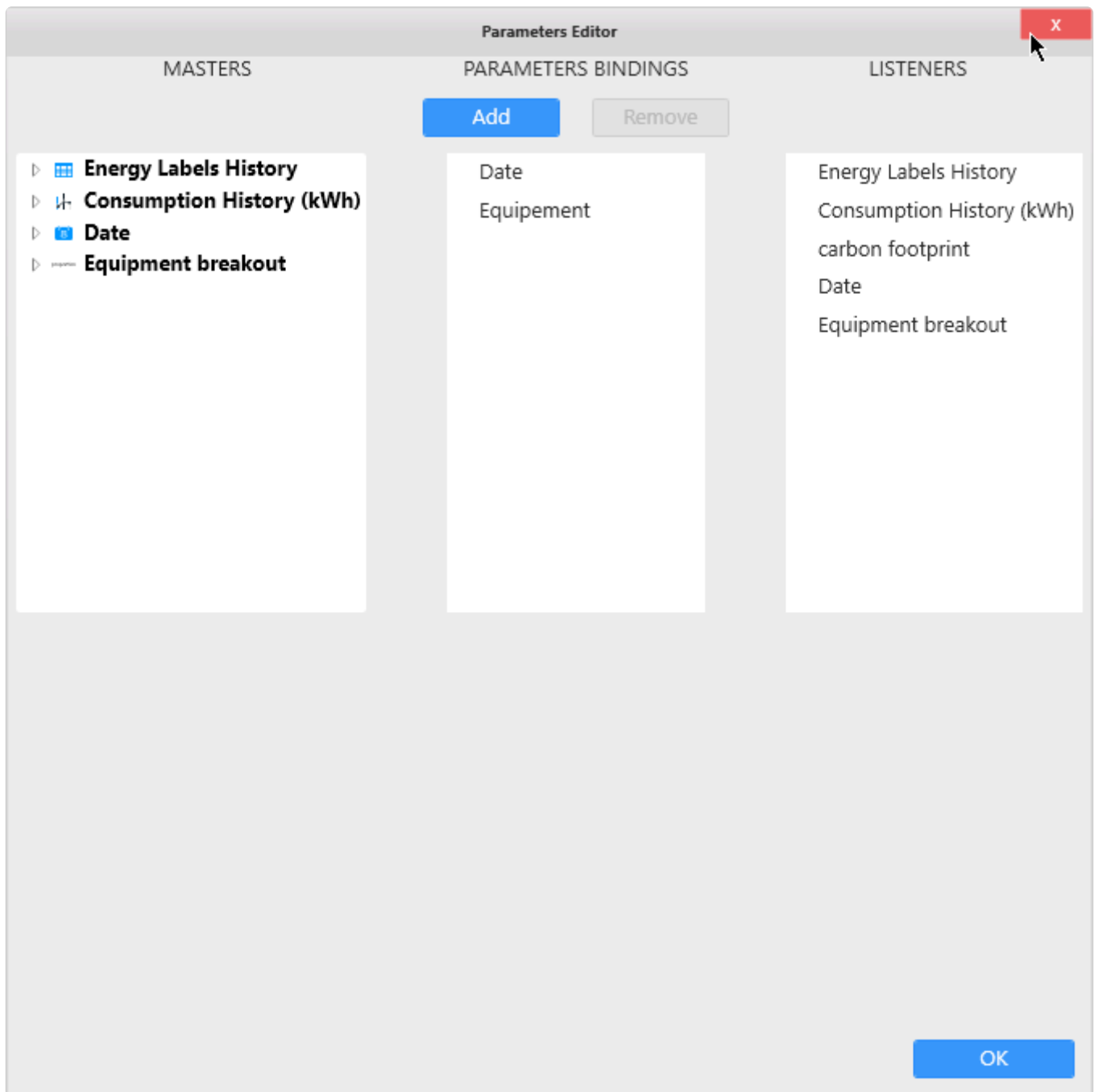
In many places where a Parameter can be used, a list icon acts as shortcut to open the Parameters Editor :



This appears for example in Data Source filters, in Stored Procedures parameters, etc...

Result : Parameters Editor

This opens the “Parameters Editor” dialog :



From there, you can manage Parameters :

- create Parameters
- rename Parameters
- remove Parameters
- change their default value
- change bindings between Widgets and Parameters

Masters

On the left is a list of Master Widgets and their selection events : this is a source of data for Parameters.

Parameters Binding

In the middle is a list of Parameters : this is the variable that contains the value.

Listeners

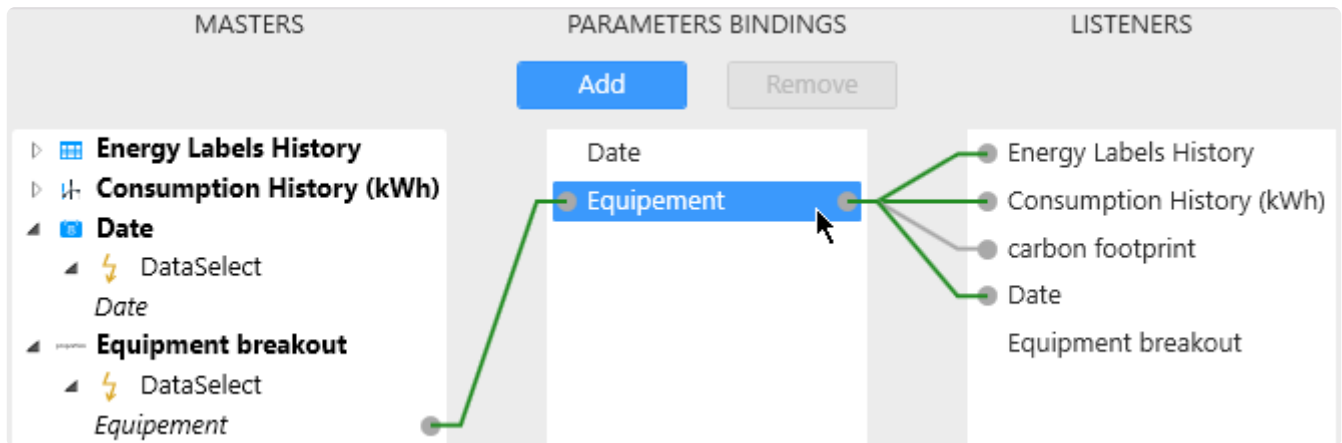
On the right is a list of Listener Widgets : this is a destination that will update according to the value.

Last modified: 2019/05/24

2.8.2.2. Visualizing Parameter binding

See a Parameter binding

To see which Widgets are connected to a Parameter, click on the Parameter name under the “Parameters Bindings” list :



List of Parameters

To the middle is a list of dashboard Parameters.

Master binding

To the left are the links to Master Widgets : this is the Widgets that can **send a value** into the Parameter.

Listener binding

To the right are the links to Listener widgets : this is the Widgets that will **receive a value** from the Parameter.

See Parameter-Listener relationship

To see how a Listener Widget will become filtered, click on the Listener Widget :

The screenshot shows the Alpana configuration interface. The top section is divided into three panels: **MASTERS**, **PARAMETERS BINDINGS**, and **LISTENERS**. The **MASTERS** panel contains two expandable sections: **Energy Labels History** and **Consumption History (kWh)**. Under **Energy Labels History**, there is a **Date** field with a **DataSelect** icon. Under **Consumption History (kWh)**, there is an **Equipment breakout** section with an **Equipment** field and a **DataSelect** icon. The **PARAMETERS BINDINGS** panel shows two parameters: **Date** and **Equipment**. The **LISTENERS** panel shows a list of listeners: **Energy Labels History**, **Consumption History (kWh)**, **carbon footprint**, **Date**, and **Equipment breakout**. A green line connects the **Date** parameter in the **PARAMETERS BINDINGS** panel to the **Energy Labels History** listener in the **LISTENERS** panel. Below the panels, there are two filter action configuration boxes. The first box is for the **Date** parameter, with a dropdown set to **Date**, a comparison operator dropdown set to **Between**, and another dropdown set to **Date**. It has an **Enable** checkbox checked and a **Use Field Format** checkbox unchecked. The second box is for the **Equipment** parameter, with a dropdown set to **Equipment**, a comparison operator dropdown set to **In**, and another dropdown set to **Equipment**. It has an **Enable** checkbox checked.

The link to Parameters is displayed in the middle.

At the bottom, is displayed a list of all filter actions configured for each Parameter :

- on the left : the field name from the Data Source
- on the middle : a boolean comparison operator
- on the right : the Parameter name
- at the bottom left : a checkbox to disable this filtering
- at the bottom right : an optional formatting applied to the Data Source field

Last modified: 2019/05/24

2.8.2.3. Editing Parameter bindings

To edit Parameters bindings, see the following chapters.

Last modified: 2019/05/24

2.8.2.3.1. Necessary connections

No connection is absolutely necessary :

- a Parameter that is not linked to a Master Widget can still receive values from a URL Parameter.
- a Parameter that is not linked to a Listener Widget can still send values into a Stored Procedure, a Custom Query Table, an Expression Value, a filter condition, ...

Multiple connections are possible :

- a Parameter can receive values from several Master Widgets
- a Parameter can send values into several Listener Widgets

Last modified: 2019/05/24

2.8.2.3.2. Removing a connection

Why remove a connection ?

By default, a Filter Widget becomes connected to all Listener Widgets using the same Data Source. Sometimes, you only want to filter some Widgets and not others.

This can also be useful for sending values into a Parameter without Listener Widgets :

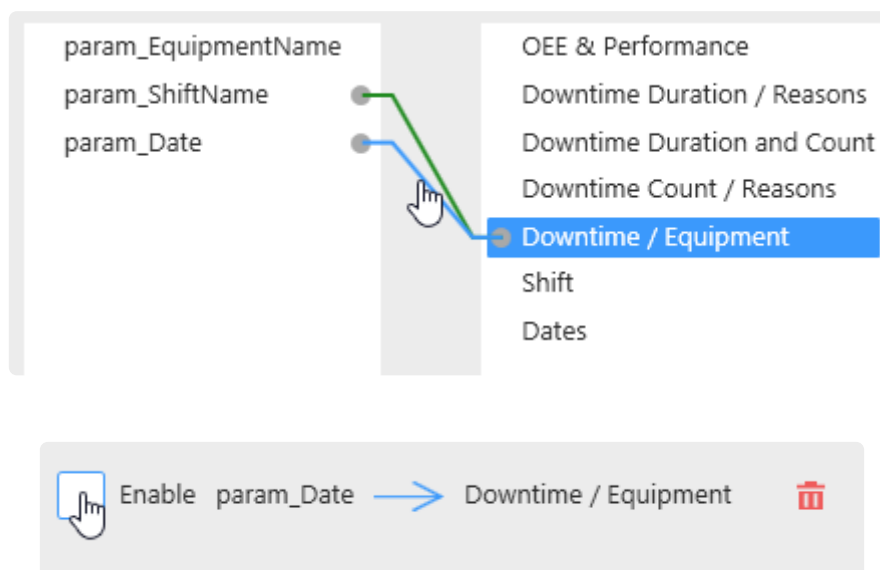
The Master Widget sends a value into the Parameter, and the Parameter is used as Stored Procedure parameter for example.

How to disable a connection

You can disable a connection without removing it.

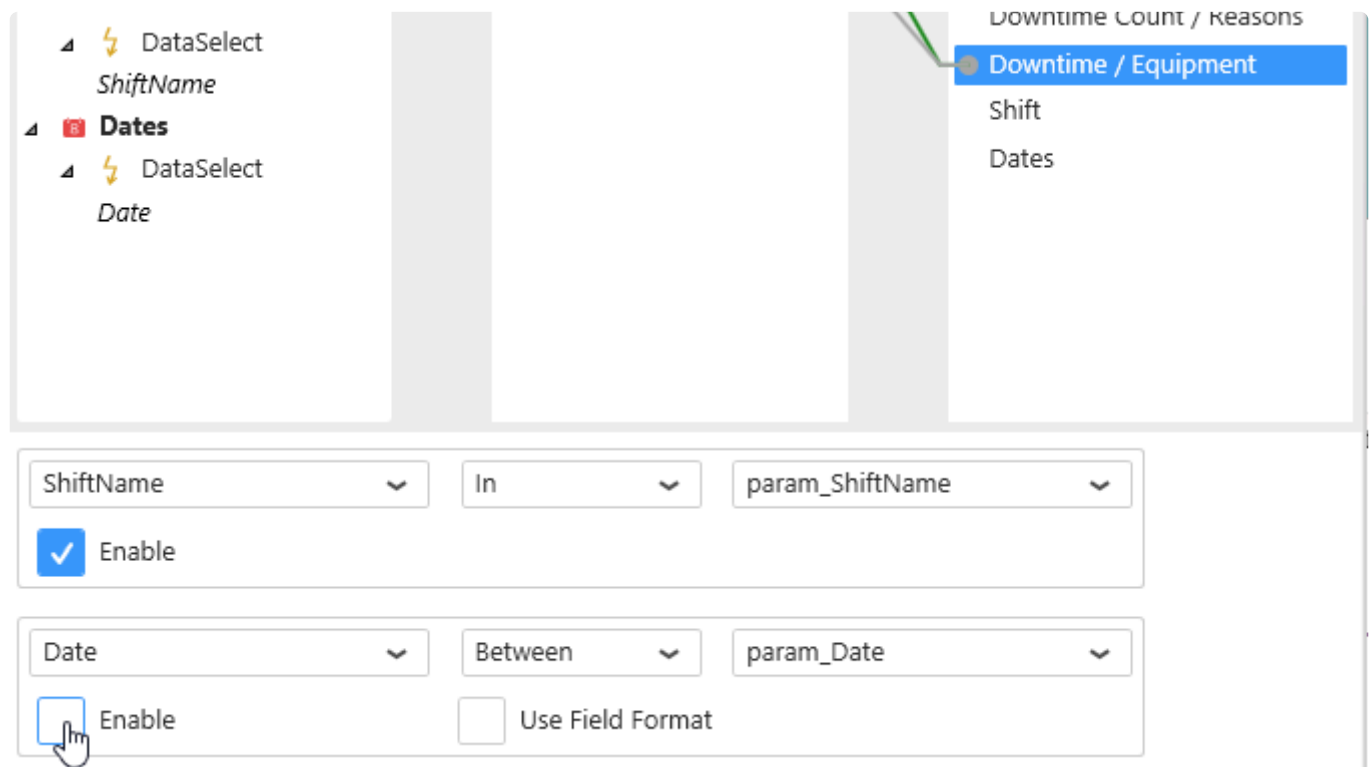
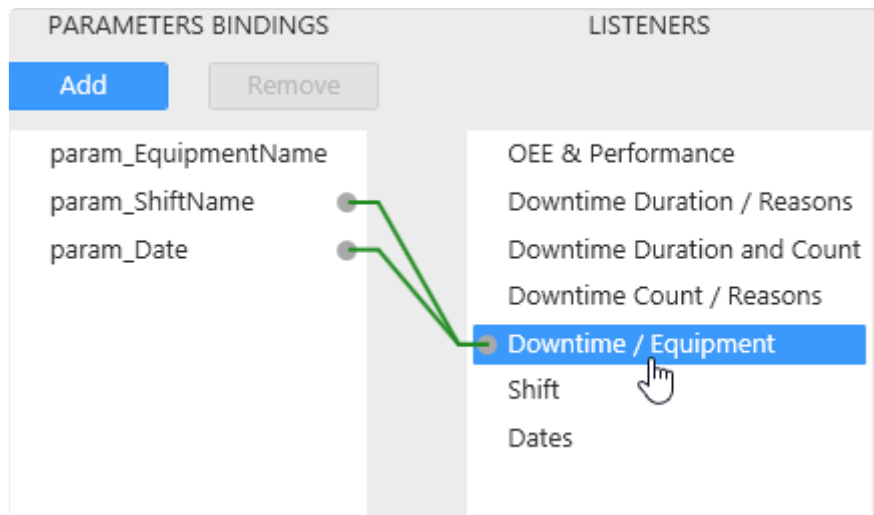
From the connection

Click on the connection, and un-check "Enable" :



From the Listener Widget

Alternately, you can click on a Listener Widget, and un-check "Enable" on the corresponding relationship :

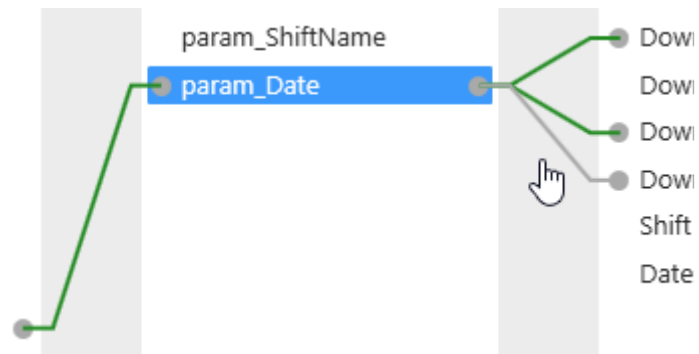


Re-enable connection

Similarly, you can re-enable the connection by checking again “Enable”.

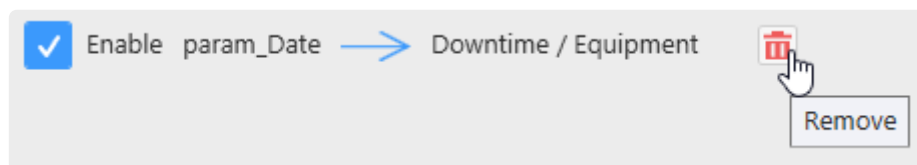
See disabled connections

A disabled connection is displayed as a greyed-out link :



How to remove a connection

To remove a connection, click on it, then click on the red bin button :



Last modified: 2019/05/24

2.8.2.3.3. Adding a connection

Why add a connection ?

This can be useful for **filtering across Data Sources** :

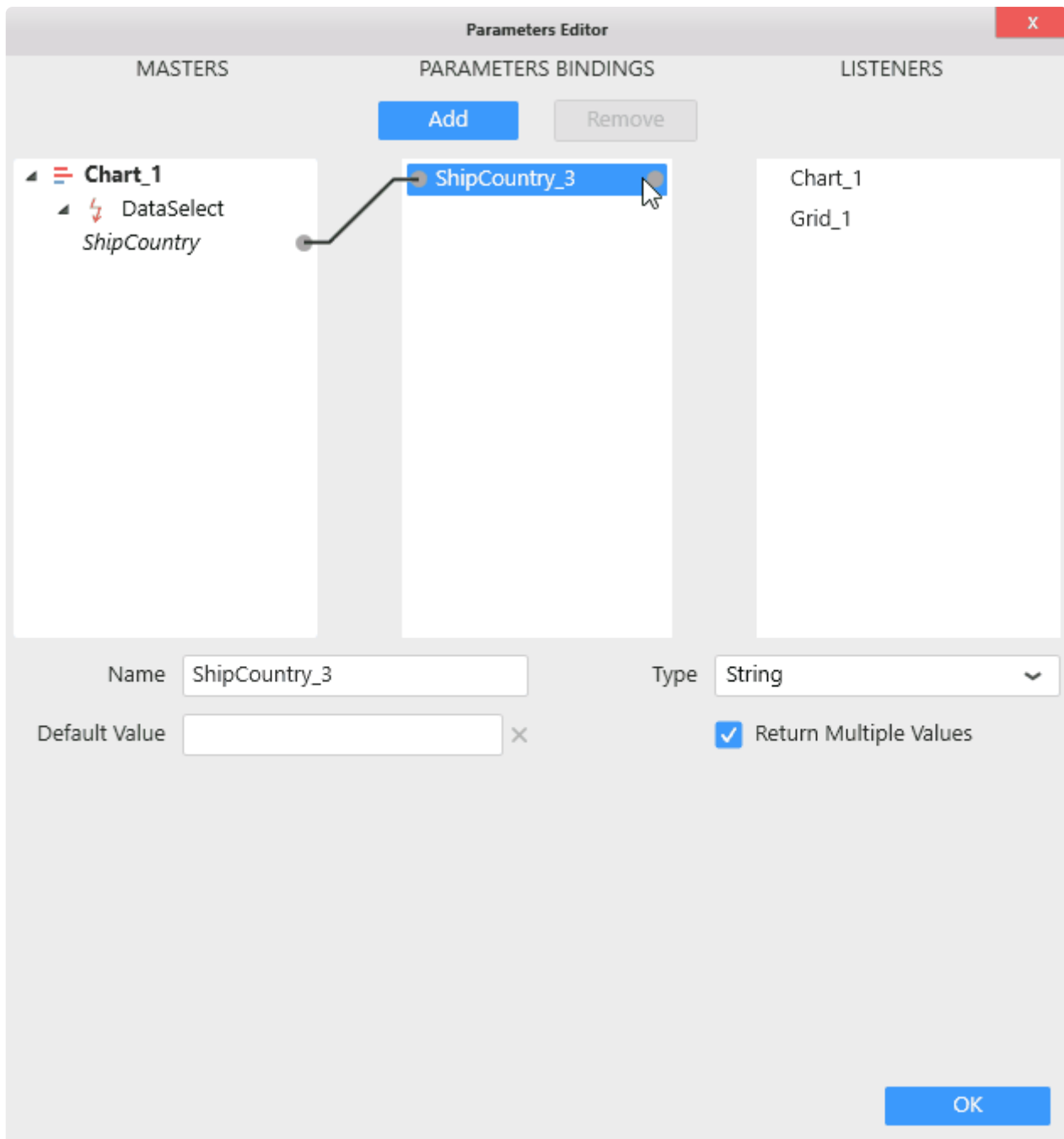
Only Widgets from the same Data Source are linked by default.

If you wish to use a Master Widget from DataSource_A to filter a Listener Widget from DataSource_B, you have to manually define the relationship : *what field in DataSource_B will become filtered by the Parameter and how ?*

This can also be useful for sending values into a Parameter without Listener Widgets. The Parameter can be used as Stored Procedure parameter for example.

How to add a connection

To add a connection, click on the right dot of the desired field to link from, and drag to the left dot on the field to be linked to :

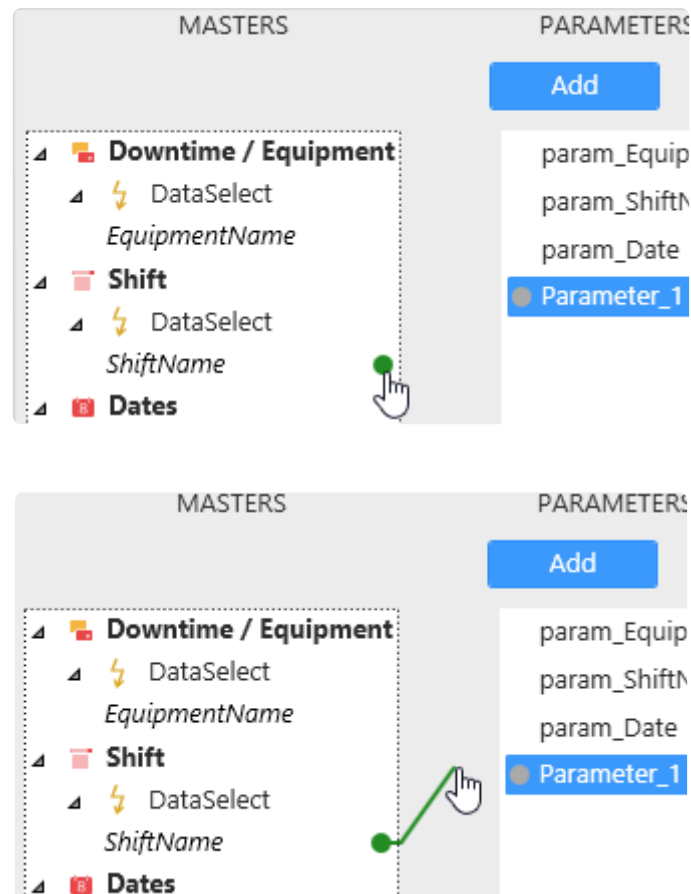


✿ A connection can only be created from left to right.

Connecting a Master to a Parameter

Browse the Master Widget on the left. Unfold the tree to find the desired DataSelect event (the name of the event is the name of the Data Source field selected).

Hover the DataSelect event. A grey dot appears on the right of the event name, this is where you will click and drag to draw a link :



Complete the link into the grey dot on the left of the destination Parameter :

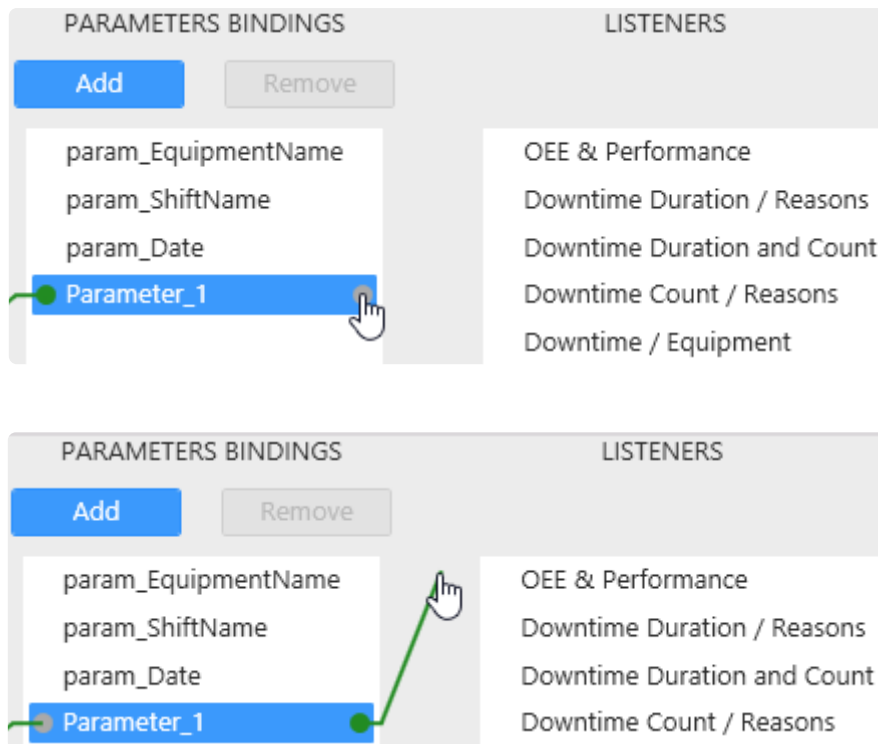


The Master Widget is now linked to this Parameter :

When a user will generate a DataSelect event for this field, the corresponding data value(s) will be sent into the Parameter.

Connecting a Parameter to a Listener

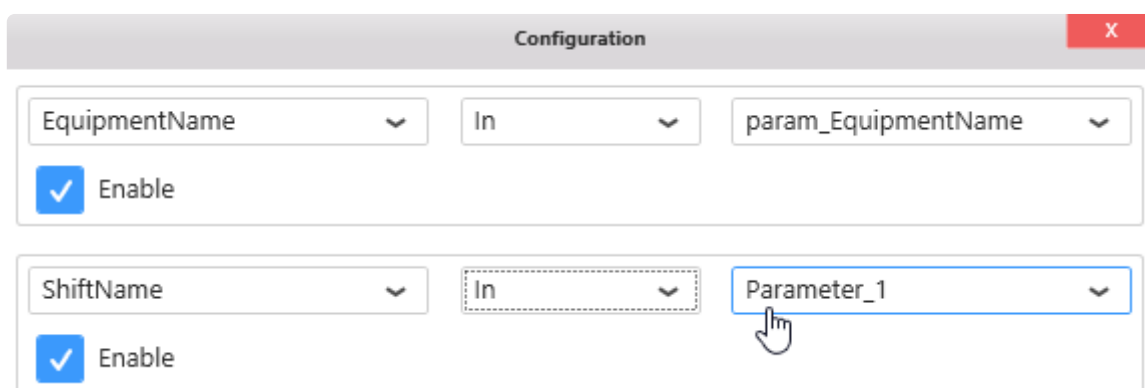
Select a Parameter. A grey dot appears on the right of the Parameter name, this is where you will click and drag to draw a link :



Complete the link into the grey dot on the left of the destination Listener Widget :



Upon completing the link, a relation ship editor appears to let you set the relationship between the Parameter value and the Data Source fields of the Listener Widget :



Set up the desired relationship, and close the dialog with the red cross.

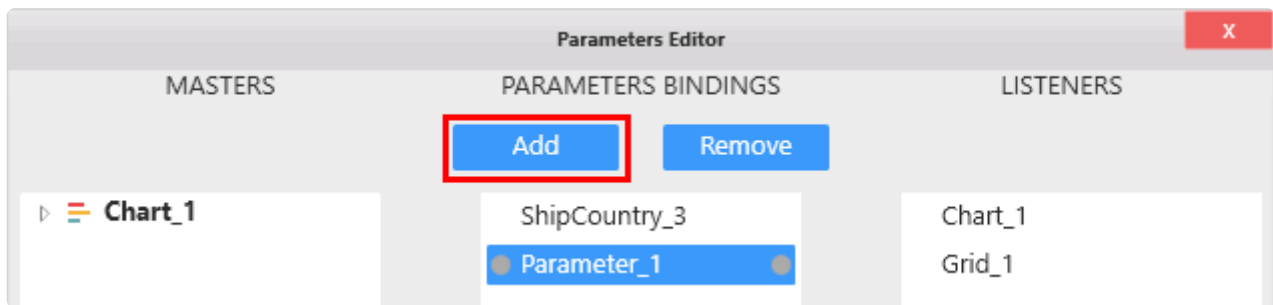
The relationship should read like a SQL statement. For example here : `WHERE ShiftName IN(<values of Parameter_1>)`

Last modified: 2019/05/24

2.8.2.4. Adding/Removing Parameter

Manually Adding a Parameter

To add a Parameter simply click on the Add button under “Parameter Bindings” :

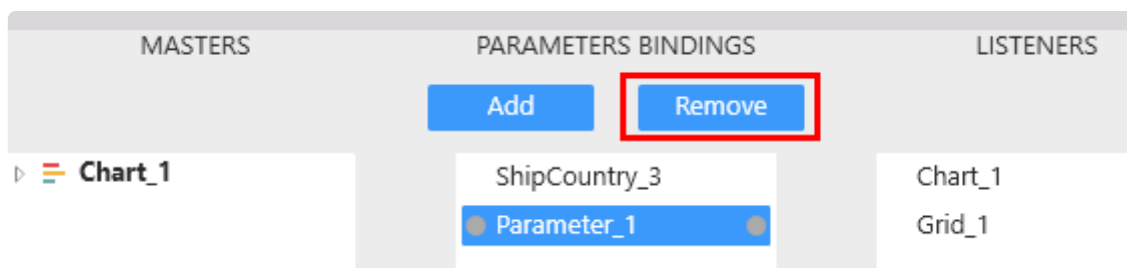


The new Parameter is added, with no links and default settings.

Manually Removing a Parameter

To remove a Parameter, first **delete all the connections** linked to it.

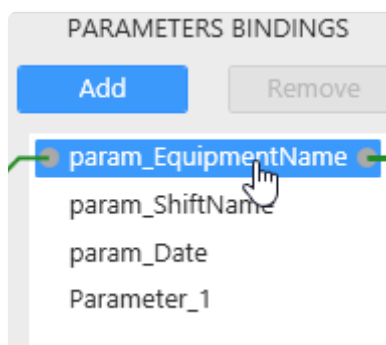
Then select the Parameter and click the *Remove* button :



Last modified: 2019/05/24

2.8.2.5. Edit Parameter properties

To edit a Parameter, click on it in the middle list :



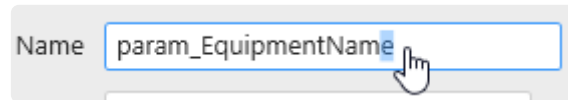
The Parameter properties are editable at the bottom :

Name	<input type="text" value="param_EquipmentName"/>	Type	<input type="text" value="String"/>
Default Value	<input type="text"/>	<input checked="" type="checkbox"/>	Return Multiple Values

Last modified: 2019/05/24

2.8.2.5.1. Parameter Name

You can give a custom name to the Parameter by using this text box :



A screenshot of a user interface element. It consists of a light gray rounded rectangle. Inside, on the left, is the label "Name" in a dark gray font. To the right of the label is a white rectangular text input field with a thin blue border. The text "param_EquipmentName" is entered into this field. A mouse cursor, depicted as a hand with a pointing finger, is positioned at the end of the text inside the input field.



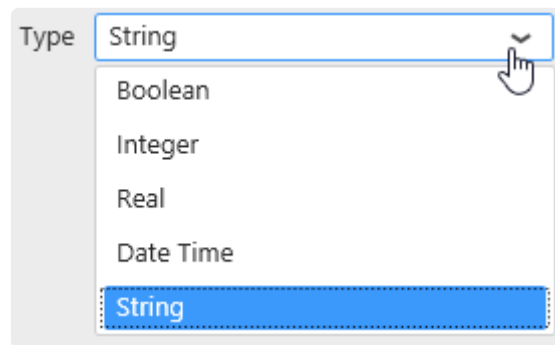
This is especially useful when using the Parameter as URL Parameter : the Parameter name will be used in the syntax to pass values to the Dashboard from the URL.

Parameter names are also listed in the Filter Overview at runtime.

Last modified: 2019/05/24

2.8.2.5.2. Value Type

To set the Type of the Parameter, select it from the “Type” listbox :



Values sent into this Parameter will be converted to this Type.

Last modified: 2019/05/24

2.8.2.5.3. Parameter Default Value

The value that the Parameter has by default is called the *Default Value*.

This is the value that the Parameter has before any manual filtering at runtime for example.

Setting Default Value from URL Parameters

[URL Parameters](#) actually replace the Default Value when the Dashboard loads.

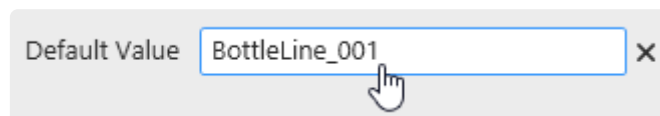
Setting Default Value from Widgets

It is possible to set a Parameter Default Value by making a selection in the Widget Preview of one of its Master Widgets.

This is explained in the [corresponding chapter](#).

Setting Default Value manually

Default Values can be set manually from the “Default Value” text box :



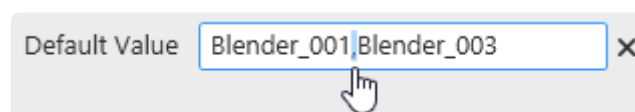
A screenshot of a user interface element. It consists of a light gray rounded rectangle. Inside, on the left, is the text 'Default Value'. To its right is a text input field with a blue border. The input field contains the text 'BottleLine_001'. To the right of the input field is a small 'x' icon. A hand cursor is pointing at the input field.

Text or number Default Values

Text or number default value can be set manually by writing the value in the text box.

Multiple Default Values

Multiple Default values can be set using a comma , as separator :



A screenshot of a user interface element, similar to the one above. It consists of a light gray rounded rectangle. Inside, on the left, is the text 'Default Value'. To its right is a text input field with a blue border. The input field contains the text 'Blender_001,Blender_003'. To the right of the input field is a small 'x' icon. A hand cursor is pointing at the input field.

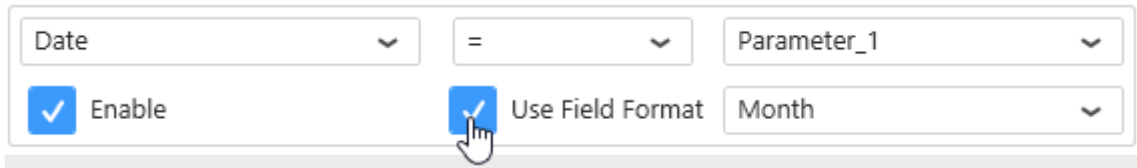
Date Default Values with Format

For Date type values, it is possible to use a date *Format*.

This will behave similarly as [Data Source Filters on formatted date fields](#).

Example :

In the Listener Widget relationship, use format “Month” :



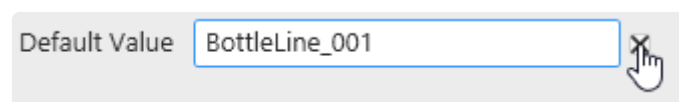
Set the Parameter as Integer and in the *Default Value* text box, you enter 2.

⇒ By default, the month of February (month number 2) is filtered.

Removing a Default Value

! Erasing the content of the *Default Value* text box does not remove the Default Value !
Actually it sets the Default Value to be the empty string “@@”.

To remove the Default Value, click the cross next to it :



Last modified: 2019/05/24

2.8.2.6. example : Multi-Data Source filtering

Multi-Data Source filtering

You can bind Parameters to any Listener Widget independently of its Data Source and independently of the corresponding field name.



This can be useful if you want to use for example a single date filter to apply to several Listener Widgets from different Data Sources.

Thus, you will avoid duplication and have a consistent filtering.

To achieve this, the Master Widget must be able to send the required values to the Parameter.

Example scenario 1

For example if DataSource1 has fewer values than DataSource2 (maybe a narrower date range, maybe not all equipments appear in this equipment list), then using the field from DataSource1 in the Master Widget will render you unable to send those missing values to the Parameter.

To solve this, you can use the field from DataSource2 which contains the most values (the widest date range, or the complete equipment list).

Note however that in this case you will try to filter Widgets from DataSource1 which may not contain the values and will return empty results.

Example scenario 2

If DataSource1 has some values that are not in DataSource2, **and** the other way around, then you cannot list all possible values in a Filter Widget by binding a field from one of your existing Data Source directly.

To solve this, you can create an additional Data Source dedicated to contain all the filter values you need to use. It could be a Custom Query with an UNION of both data sets, or it could come from one of your reference tables listing all equipments for example.

Note, once again, that when you will filter for values that exist only in one of the Data Sources, the Widgets bound to the other Data Source will return empty results.

Last modified: 2019/05/24

2.8.3. Widget as Master Filter

Alpana is dedicated to data exploration and interactive display of information.

Using a filter at runtime allows the user to focus the visualizations on a sub-set of the data.

Dashboard Parameters are the mechanism behind this dynamic filtering.

Dynamic filtering : Master Widget / Listener Widget

A **Master Widget** can be used at runtime to make selections that **apply a filter** to other Widgets.

Widgets that are configured to listen to this filter are called *Listener Widget*.

Internals

When the user clicks on a shape in a Master Widget, a DataSelect event is generated and sends the selected value into a Dashboard Parameter.

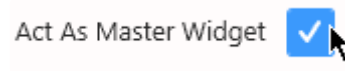
Listener Widgets receive the Dashboard Parameter value and add it as a filter to their data.

Last modified: 2019/05/24

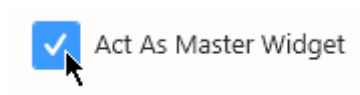
2.8.3.1. Configuring a Widget as Master

Configuration

To turn a Widget into a Master Widget and allow it to filter other Widgets, check *Act As Master Widget* under *Filter* :



There is also a shortcut in the top bar :



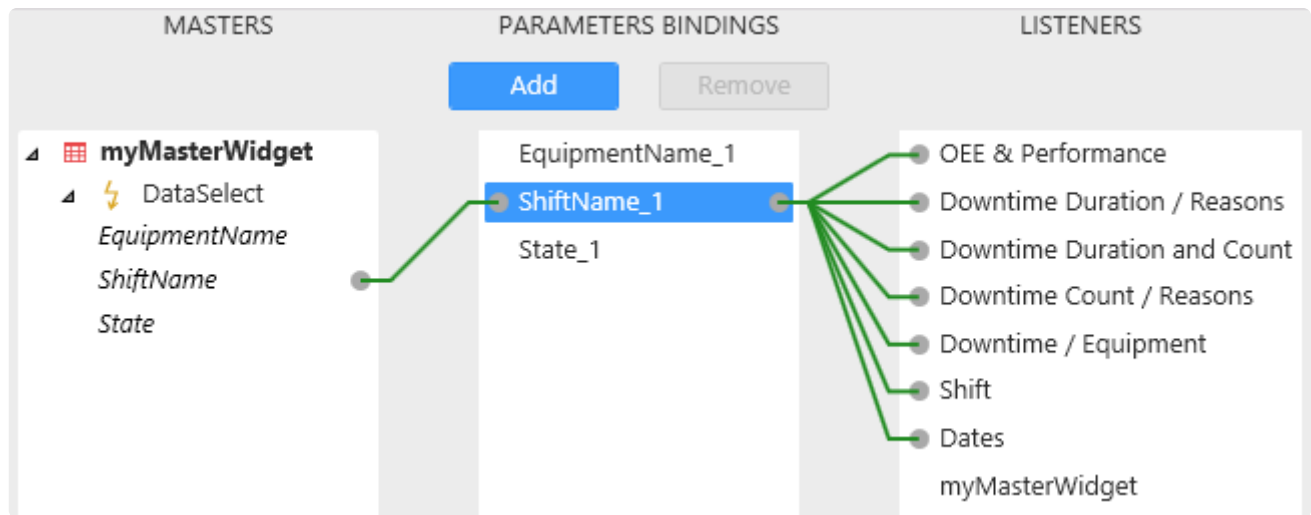
Master Widget by default

Some widgets do not require to perform this setting manually : all Widgets in category “Filter” are configured as Master Widget by default.

Internals

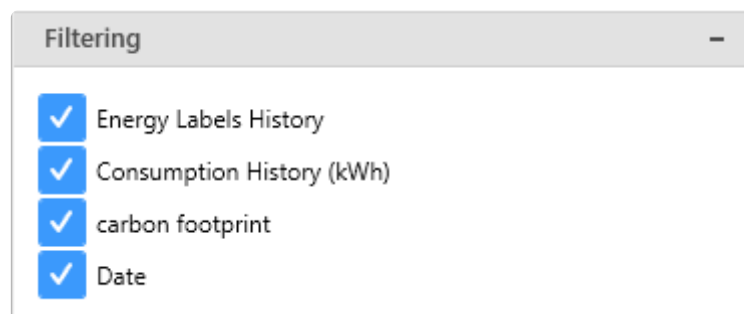
This will automatically :

- make the Widget and its DataSelect events available as “Master” to the left column in the “Parameters Editor” dialog
- generate as many Dashboard Parameters as there are DataSelect events, and name them like `FieldName_index` (example : `Equipment_1`)
- bind the events as Master to the corresponding Parameters
- bind to these Parameters all the Listener Widgets that use the same Data Source as the Master Widget



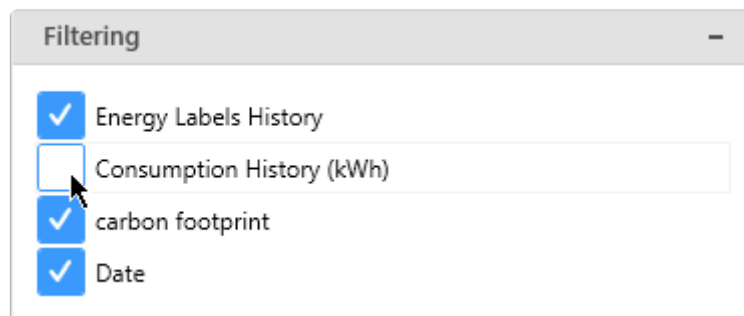
Viewing filtered widgets

When the Master Widget configuration is active, a new section appears at the bottom of the data panel on the left, called *Filtering* :



This is a list of Listener Widgets that are bound to this Master Widget.

The checkboxes are a shortcut to disable the Parameter linking to that listener.



Last modified: 2019/05/24

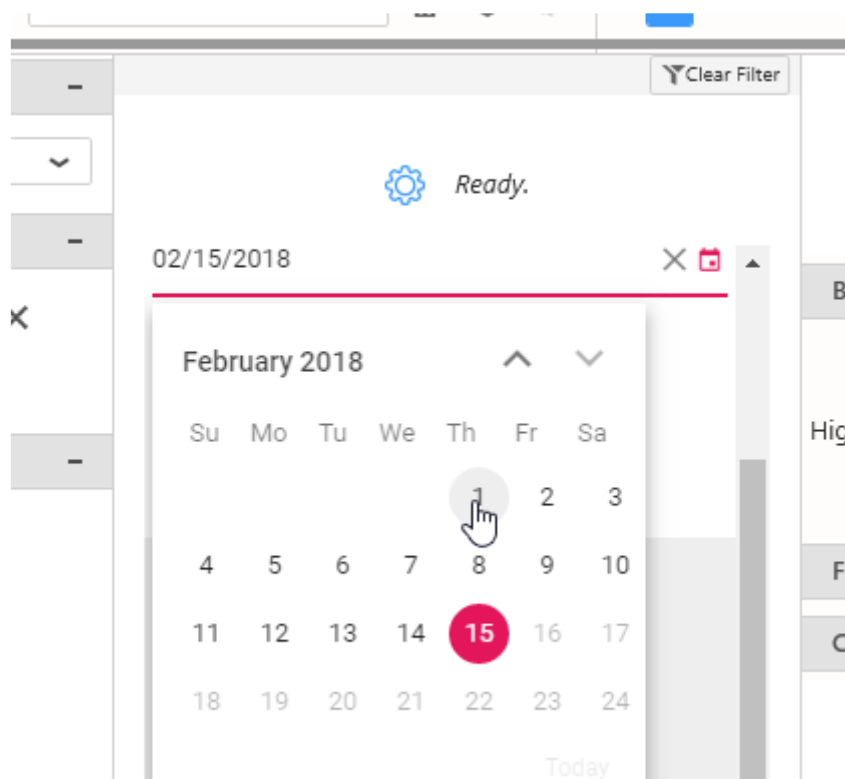
2.8.3.2. Setting a Widget Default Selection

The Master Widget can be configured to generate a selection by default.

This means that every time the Dashboard will load, the Widget will send this default selection into the [Parameter Default Value](#).

Setting a static default selection

In Alpana Designer, use the Widget preview in the middle part of the view to select the desired default value :



Default selection as Relative Date

In addition, some widgets (date/time filter widgets) allow to configure a **relative date range as default selection**.

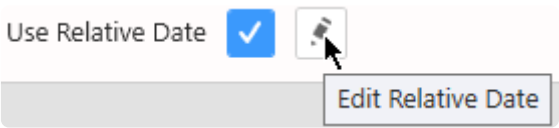


Note :

This is different from the notion of Widget data filtered based on a Relative Date. If you have a Widget that needs to have its data filtered based on a relative date, see instead the chapter on [Widget-level Filtering](#).

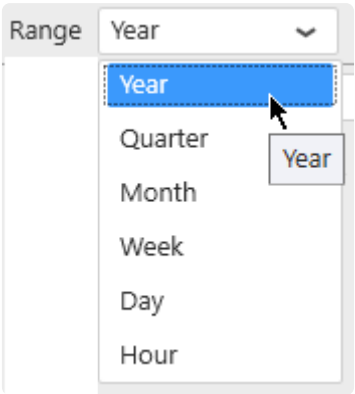
Enabling Relative Date for the Default selection

For this, check *Use Relative Date* under Basic Settings and click the pen icon to configure :

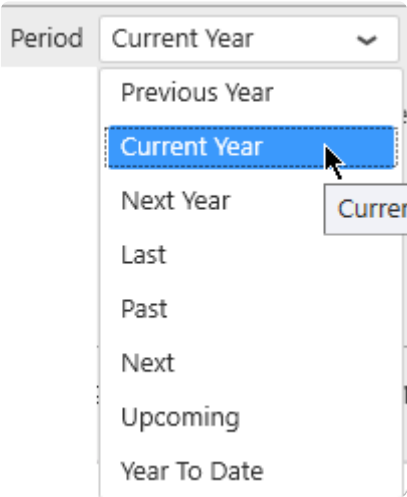


Configuring a Relative Date Period

Then for each time range :
Year, Quarter, Month, Week, Day, Hour



... a Period can be selected :
Previous 1, Current, Next 1, Last N, Past N, Next N, Upcoming N, To Date



A preview of the selected date range for the current time appears at the bottom of the window to help you configure :

1/1/2019 12:00:00 AM to 12/31/2019 11:59:59 PM



Last modified: 2019/05/24

2.8.3.3. Using filters at runtime

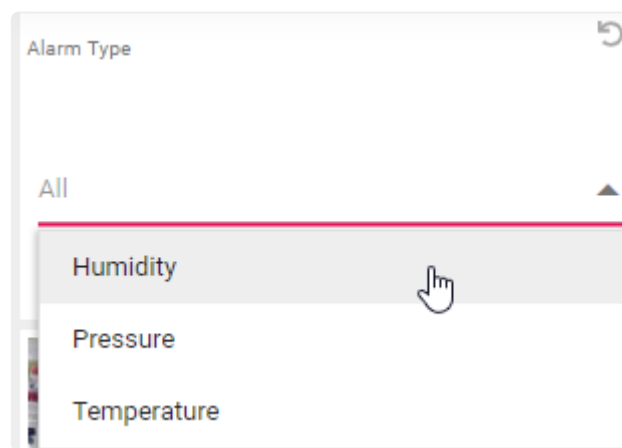
Default values


When a Dashboard loads, all Parameters have their Default Value.

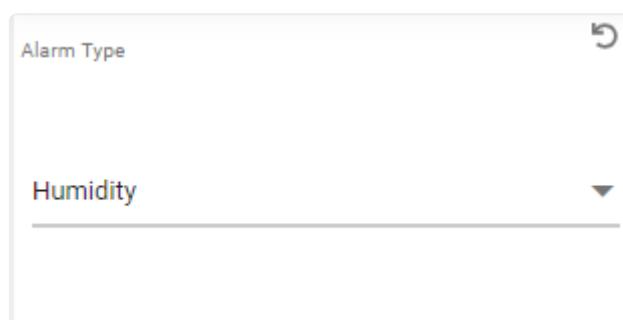
It may be empty (by default) or configured with values (see above in the [Managing Parameters chapter](#)).

Applying a filter

At runtime, clicking a shape in the Widget applies a filter and changes the corresponding Dashboard Parameter values :

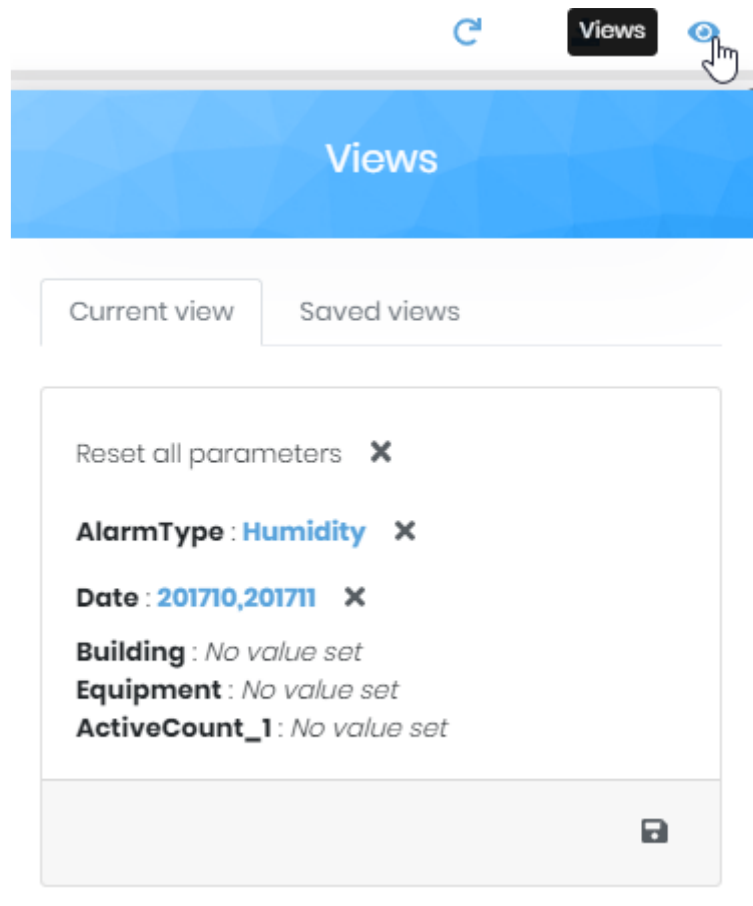


A reset icon  appears in the title bar of the Widget to indicate that the Widget is currently applying a filter :



Filters Overview

When filters are applied, the state of Dashboard Parameters can be listed by clicking **Views** on the top right of the Dashboard :




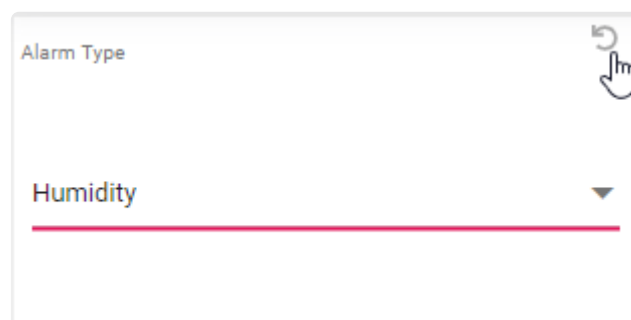
This shows a list of Parameter names and corresponding values.

Clearing filters

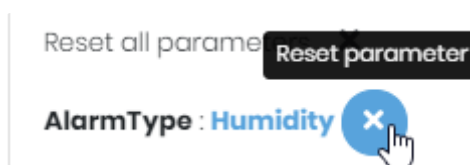
Parameters can be reset to their default values to remove the filter effect.

Clearing single filter

To clear a single filter, click on the reset icon  in the title bar of the Widget :

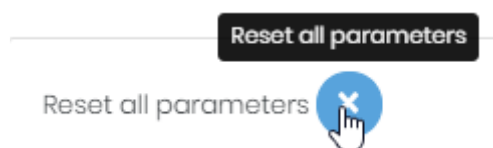


You can also click on the cross on the corresponding Parameter in the *Views* :



Clearing all applied filters

To clear all applied filters at once and reset all Parameters to their default value, open the *Views* and click *Reset All Parameters* :



Last modified: 2019/05/24

2.8.4. URL Parameters to receive values

The [default value](#) of any Parameter can be set from a **URL Parameter**.

This means that a particular set of filters can be used when opening the dashboard if the correct URL Parameters are configured.



This can be used to receive context from other tools that will simply link to a Alpana Dashboard and concatenate filter values as context.

See also Wikipedia for more information : https://en.wikipedia.org/wiki/Query_string

Last modified: 2019/05/24

2.8.4.1. Example

Your dashboard contains a Parameter called `Parameter1` and another called `Parameter2` which accepts multiple values.

The dashboard is published at `http://yoursite.com/viewer/dashboard/YourDashboardID`

By adding this :

```
http://yoursite.com/viewer/dashboard/YourDashboardID?Parameter1=ValueA&Parameter2=ValueB&Parameter2=ValueC&Parameter2=ValueD
```

It will give a default value for your Parameters :

- `Parameter1` takes the single value `ValueA`
- `Parameter2` takes the multiple values `ValueB, ValueC, ValueD`

Last modified: 2019/05/24

2.8.4.2. Reserved words



Important

The following names are reserved and cannot be used as URL Parameters. Please rename the corresponding Parameters.

- Id
- Category
- Comment
- View
- Dashboard
- Tab
- Viewid

Last modified: 2019/05/24

2.8.5. Parameter usage

Parameters are a fundamental concept used for many advanced configurations.
See some examples in the following chapters.

Last modified: 2019/05/24

2.8.5.1. Dynamic calls to Stored Procedures or Table-Valued Functions

Parameters can be used as arguments to **Stored Procedures** or **Table-Valued Functions** : when the Parameter changes, the Stored Procedure or Table-Valued Function is called again with the new values as arguments.

For more information, see in chapter [Stored Procedures](#).

Last modified: 2019/05/24

2.8.5.2. Dynamic calls to Custom Queries

Parameters can be used inside **Custom Queries** : when the Parameter changes, the Custom Query is called again with the new values.

For more information, see in chapter [Custom Query Tables](#).

Last modified: 2019/05/24

2.8.5.3. Dynamic Expression Values

Expressions can use Parameter values as “Value” : this allows to change the calculation depending on the value of the Parameter.

For more information, see in chapter [Calculating Expressions](#).

Last modified: 2019/05/24

2.8.5.4. Dynamic Data Source filters conditions

The value of a **Data Source filter** condition can be a Parameter value : this allows to change the Data Source-level filter depending on the value of the Parameter.

For more information, see in chapter [Filtering Data](#).

Last modified: 2019/05/24

2.8.5.5. Dynamic Widget filters conditions

The value of a **Widget-level filter** condition can be a Parameter value : this allows to change the Widget-level filter depending on the value of the Parameter.

For more information, see in chapter [Widget-Level Filtering](#)

Last modified: 2019/05/24

2.8.5.6. Dynamic Historian Queries

Parameters can be used inside Historian queries to let users change the query at runtime : when the Parameter changes, the Historian query gets updated.

For more information, see in chapter [Parameter-Based Settings](#)

Last modified: 2019/06/18

2.8.6. Widget-specific binding

Some specific Widget types use Parameters in their configuration.
See the following chapters for a reference.

Last modified: 2019/05/24

2.8.6.1. Grid conditional formatting

Grid conditional formatting conditions values can be a Parameter value : this allows to change the conditional formatting depending on the value of the Parameter.

(see in the chapter about [Grid conditional formatting](#))

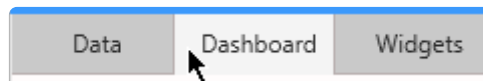
Last modified: 2019/05/24

2.9. Dashboard Settings and Publication

Once you prepared data, you can start to compose a Dashboard with Widgets.

There are several ways to proceed, as we will see in this chapter.

Since Alpana Designer v3.0, all the layout and dashboard-level properties are configured inside the *Dashboard* activity tab.



Other Dashboard-level settings are possible, such as :

- theme
- autorefresh
- title
- description
- disabling comments

Last modified: 2019/06/17

2.9.1. Dashboard Preview

When working in Alpana Designer, you can easily preview your work in the *Dashboard Preview*.



This is useful for testing interactions and making sure the dashboard layout is correct before Publishing.
This *Dashboard Preview* is local, so you don't need access to *Alpana Server* or Internet.

Starting the Dashboard Preview

To launch the current Dashboard in preview mode, click the *Preview* button in the top toolbar :



This starts your default web browser and displays the fully interactive dashboard as it would look when published.

Refreshing the Dashboard Preview

When the Dashboard is modified, the opened preview is not updated.
To update it, click again the *Preview* button to launch another Preview page.

Last modified: 2019/05/28

2.9.2. Create and Export / Import Dashboards


In Alpana Designer, a project is a Dashboard.

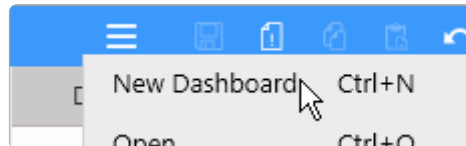
Continue reading to learn how to create/export/import Dashboards.

Last modified: 2019/05/27

2.9.2.1. Creating a New Dashboard

When Alpana Designer opens, an empty Dashboard is available for you to start working.

When a Dashboard is already open, you can replace it by a New Dashboard by clicking on the burger menu  at the top and selecting *New Dashboard* (or keyboard shortcut Ctrl+N) :



A confirmation dialog appears, allowing you to save your current dashboard, and a fresh dashboard is created in Alpana Designer.

Last modified: 2019/05/27

2.9.2.2. What is a Dashboard file


A complete Dashboard can be saved to a file with format `*.alpd`"

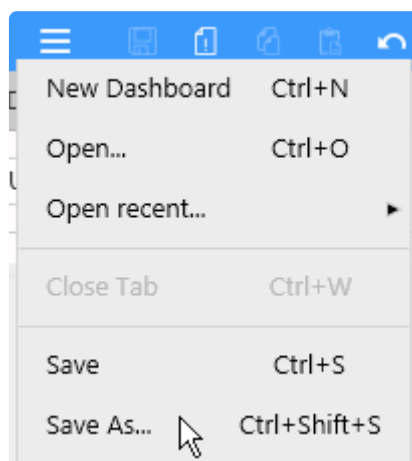
This Dashboard.alpd file contains :

- all Dashboard-level settings (theme, layout, filters, ...)
- all Widgets and their configuration
- all Data Sources
- all Connections, including their credentials
- a copy of the source data files for all file-based Connections
- the map GeoJSON content for all Map widgets
- the image files for all Image widgets

Last modified: 2019/05/27

2.9.2.3. Save a Dashboard as file

Click on the burger menu  at the top and use **Save** (keyboard shortcut Ctrl+S) to save the Dashboard with the same file name as currently, or use **Save As...** (keyboard shortcut Ctrl+Shift+S) to specify a file name and location :




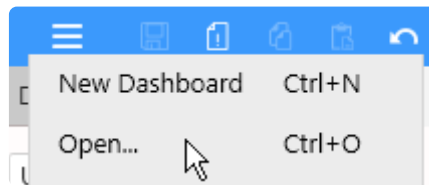
Last modified: 2019/05/27

2.9.2.4. Open a Dashboard file

A Dashboard.alpd file can be opened in a few ways :

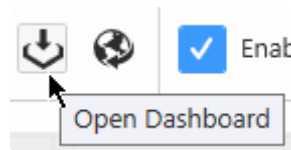
File > Open

To open a Dashboard.alpd file in the currently open Alpana Designer, click on the burger menu  at the top and use *Open...* :

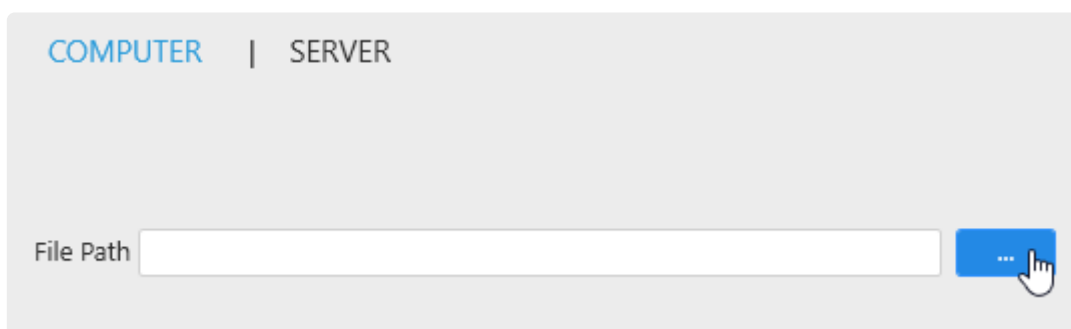


Open / Import

You can also use the top toolbar *Open Dashboard* button :




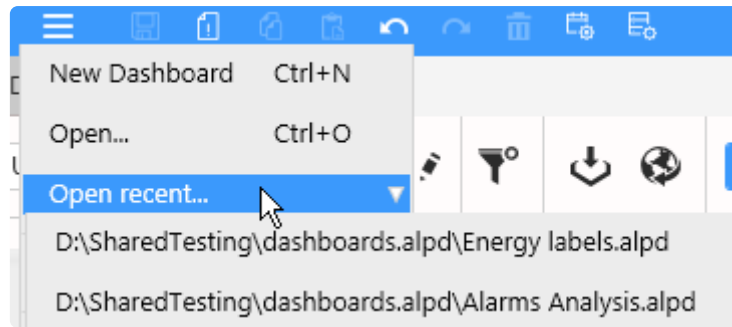
In the *Import Dashboard* dialog, under *Computer*, click ... to browse for a Dashboard file and click *Import* :



Open recent

Alpana Designer also maintains a list of recently opened Dashboards.

Click on the burger menu  at the top, hover the *Open recent...* item to pop-up a list of recent dashboards, and click an item to open it :



Double-click file in Windows

By double-clicking the file in Windows Explorer, a new instance of Alpana Designer opens this file. If Alpana Designer is already open, the window becomes focused and displays a confirmation message to close the current dashboard and open the new one.

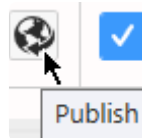
Last modified: 2019/05/27

2.9.2.5. Publishing a Dashboard to Alpana Server

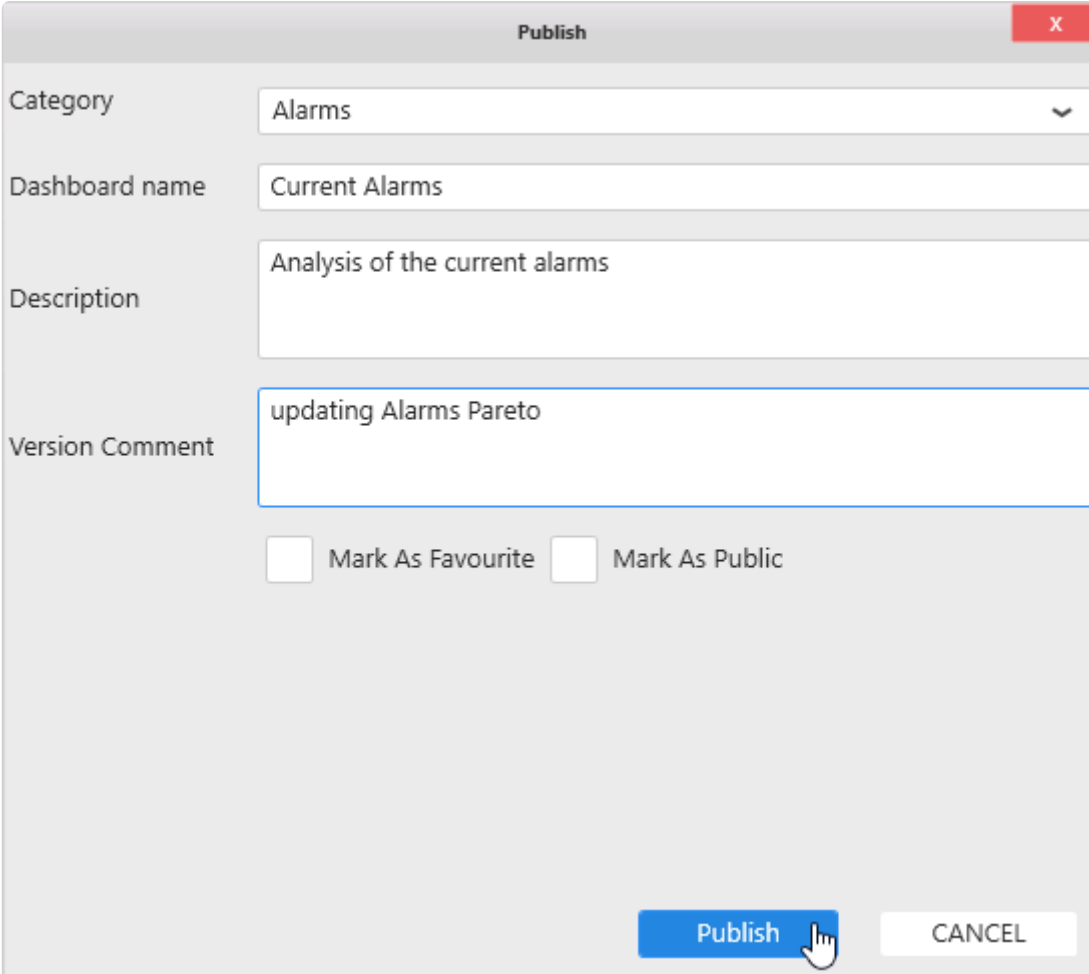
“Publishing” means pushing the Dashboard to a Tenant on Alpana Server to make it available to other users.

✿ In order to Publish a Dashboard to a Tenant on Alpana Server, you need to be [logged in](#) from Alpana Designer to that Alpana Server tenant, and have the necessary **permissions** to create Dashboards in a Category.

In the top toolbar, click the *Publish* button :



In the *Publish* dialog, fill in details and click *Publish* :



A screenshot of a 'Publish' dialog box. The dialog has a title bar with the word 'Publish' and a red close button. It contains four input fields: 'Category' with a dropdown menu showing 'Alarms', 'Dashboard name' with the text 'Current Alarms', 'Description' with the text 'Analysis of the current alarms', and 'Version Comment' with the text 'updating Alarms Pareto'. Below these fields are two checkboxes, 'Mark As Favourite' and 'Mark As Public', both of which are unchecked. At the bottom right, there are two buttons: a blue 'Publish' button and a white 'CANCEL' button. A mouse cursor is pointing at the 'Publish' button.

Publish

Category: Alarms

Dashboard name: Current Alarms

Description: Analysis of the current alarms

Version Comment: updating Alarms Pareto

☐ Mark As Favourite ☐ Mark As Public

Publish CANCEL

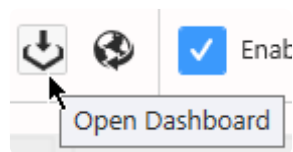
The Dashboard is now published to Alpana Server and available to other users depending on permissions.

Last modified: 2019/05/27

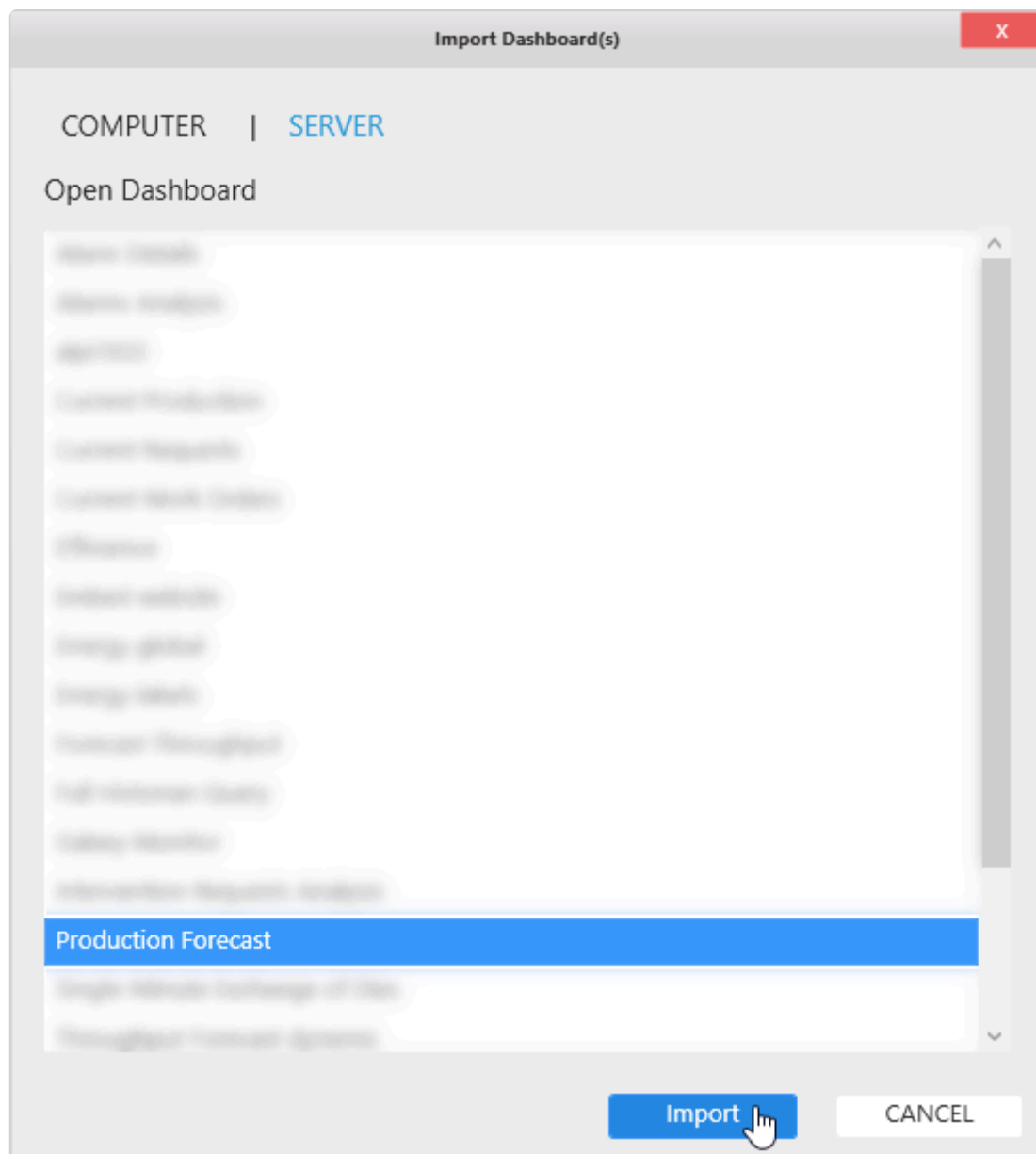
2.9.2.6. Importing a Dashboard from Alpana Server

* In order to Publish a Dashboard to a Tenant on Alpana Server, you need to be [logged in](#) from Alpana Designer to that Alpana Server tenant, and have the necessary **permissions** to access the Dashboard.

Click the top toolbar *Open Dashboard* button :



In the *Import Dashboard* dialog, click *Server*, select a published Dashboard and click *Import* :



The Dashboard is now open in Alpana Designer ready to be edited.

Last modified: 2019/05/27

2.9.3. Dashboard Layout

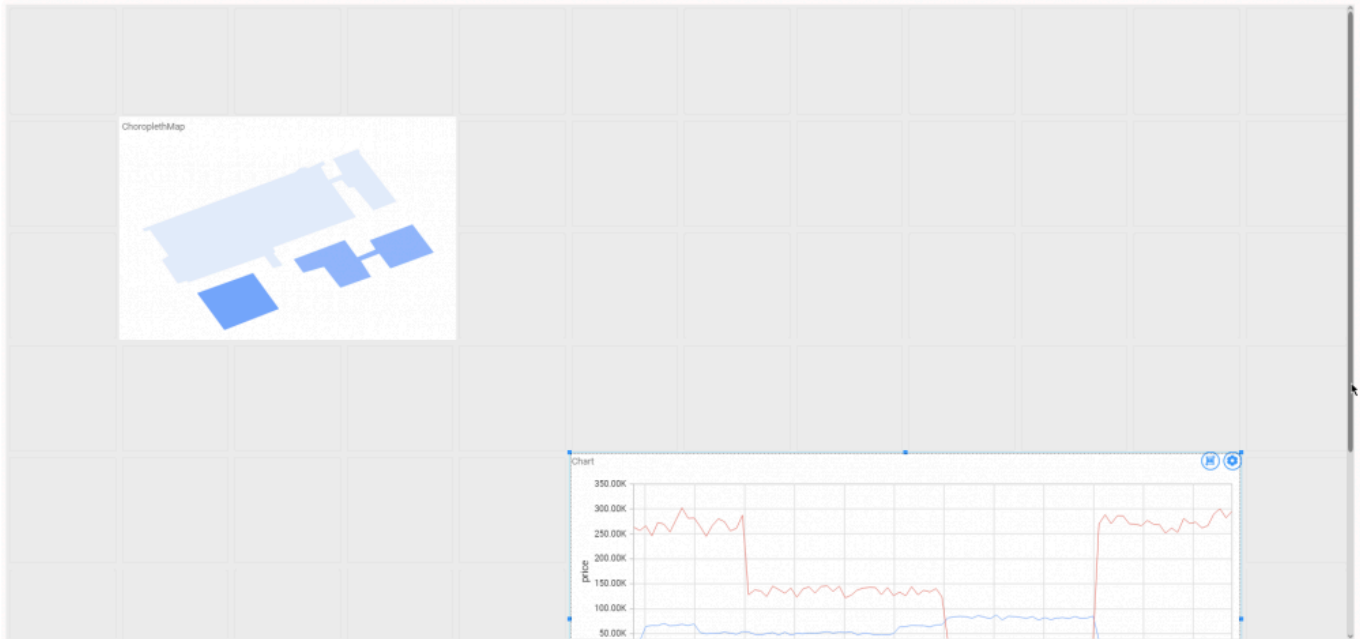
Last modified: 2019/05/27

2.9.3.1. Layout size and Responsiveness

Layout canvas

Widgets are laid out on a bi-dimensional (2D) canvas.

In Alpana Designer, the layout is configurable in the center part of the *Dashboard* activity tab :



This is where you will configure the Widget's position and size.

Layout Grid

To help for easy alignment and page composition, a layout grid is displayed in Alpana Designer. This grid is drawn in Alpana Designer only to guide the user, but doesn't appear at runtime.

When moving and resizing Widgets, they will snap to this grid.

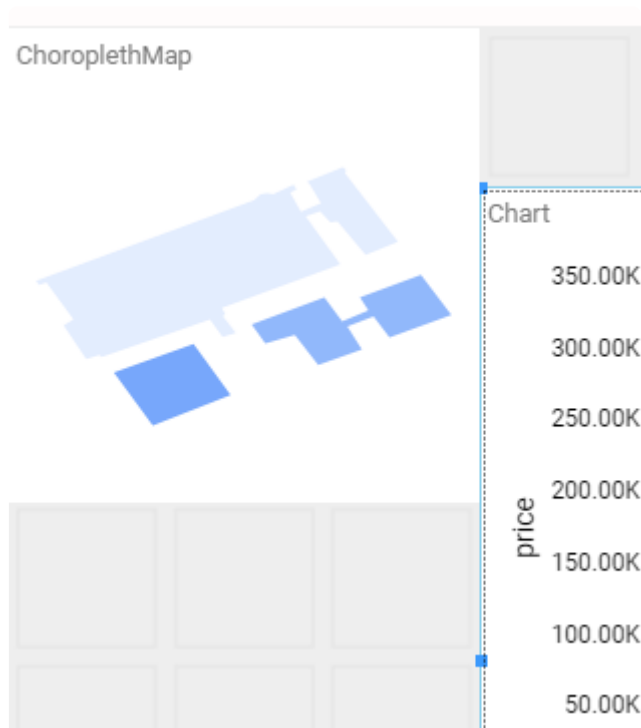
Grid Resolution

By default, the layout grid shows 12 grid tiles horizontally. (this corresponds to 6 vertically on most screens).

This *Grid Resolution* can be configured by clicking the dedicated button in the top toolbar :



For example, setting a higher resolution allows for finer Widget alignments and sizes :



Vertical size

Since Alpana v3.0.0, the layout size is not limited vertically : as long as you add Widgets at the bottom, the web page will be scrollable for any length as desired.

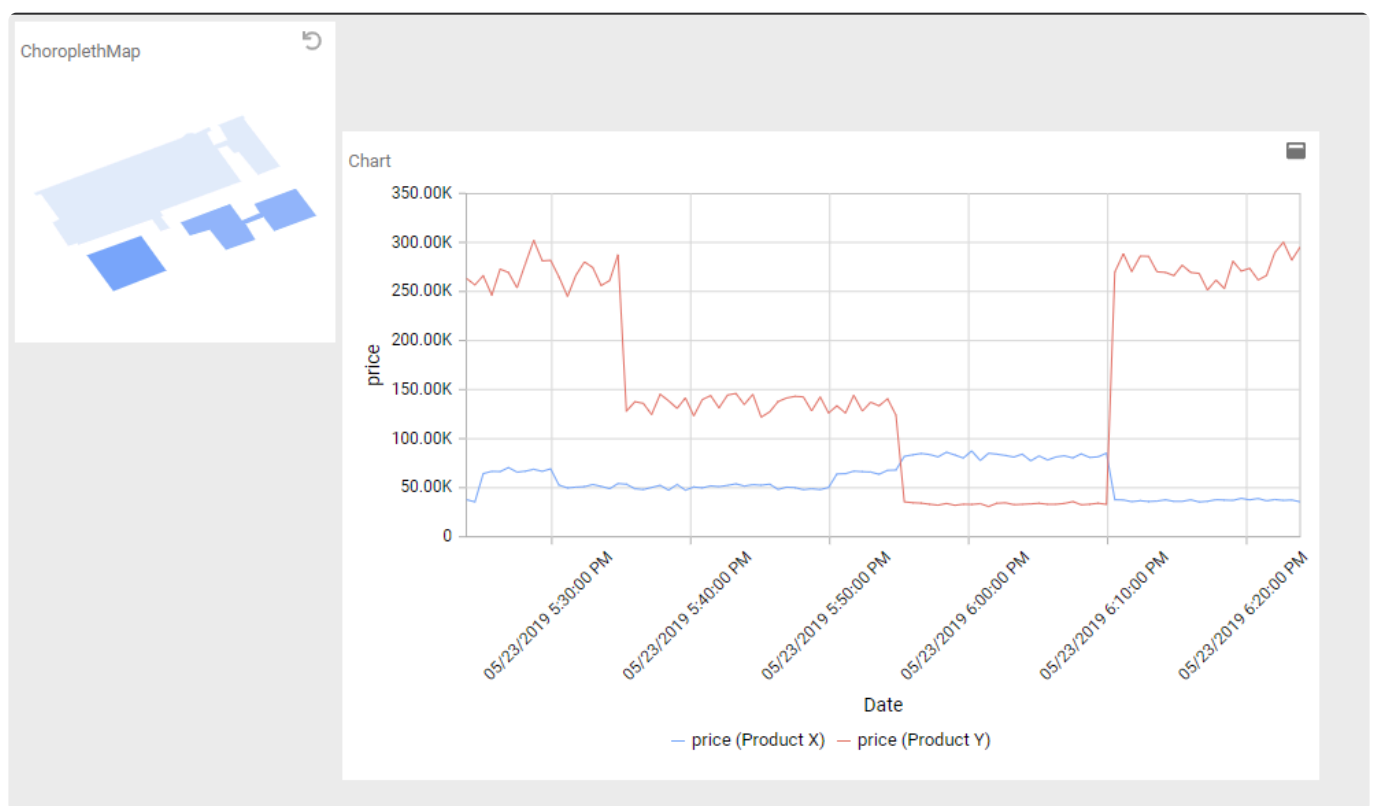
Responsiveness

This bi-dimensional layout allows to dispose Widgets horizontally and vertically.

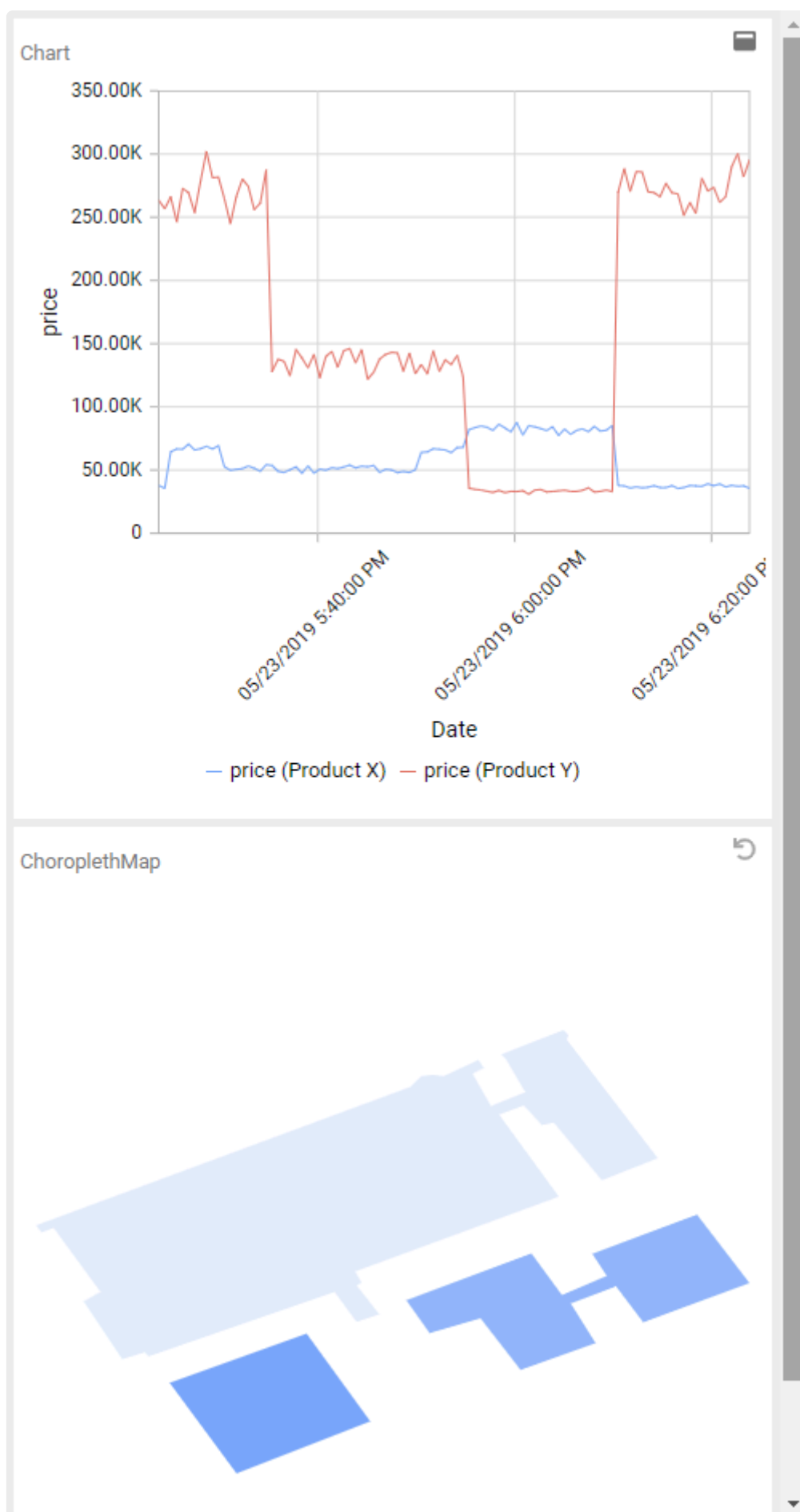
On many systems this is a comfortable way to view Dashboards.

However, on systems with small screens, it becomes preferable to have a linear layout, especially on hand-held devices.

This is why when the display size becomes too small (the size actually depends on system zoom and browser zoom too), the 2D layout turns into a linear layout : all Widgets are displayed in sequence in a vertically scrollable container :



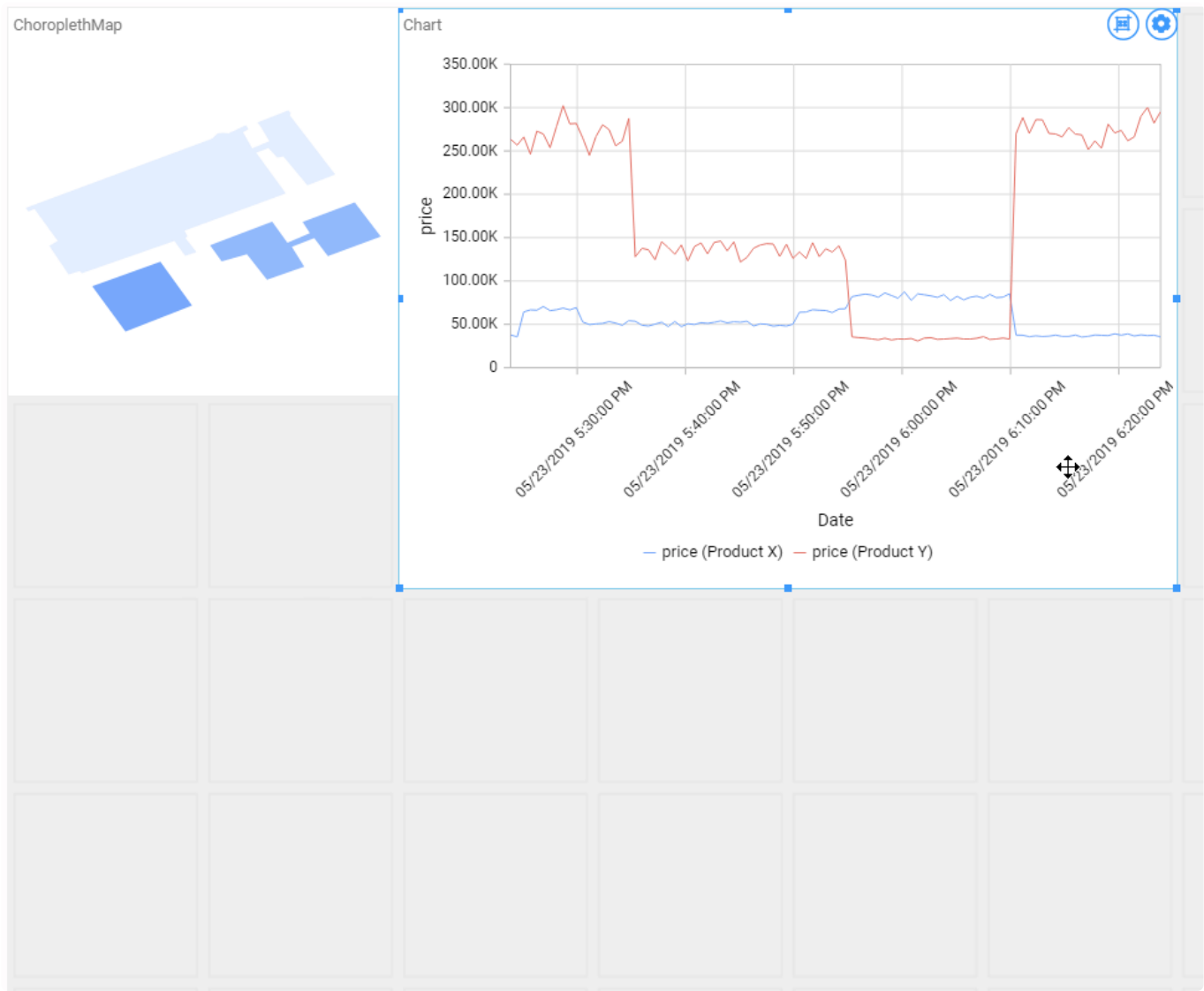
... when sized too small turns into :



Last modified: 2019/05/27

2.9.3.2. Moving Widgets

Widgets can be moved on the Dashboard layout by selecting and dragging them :

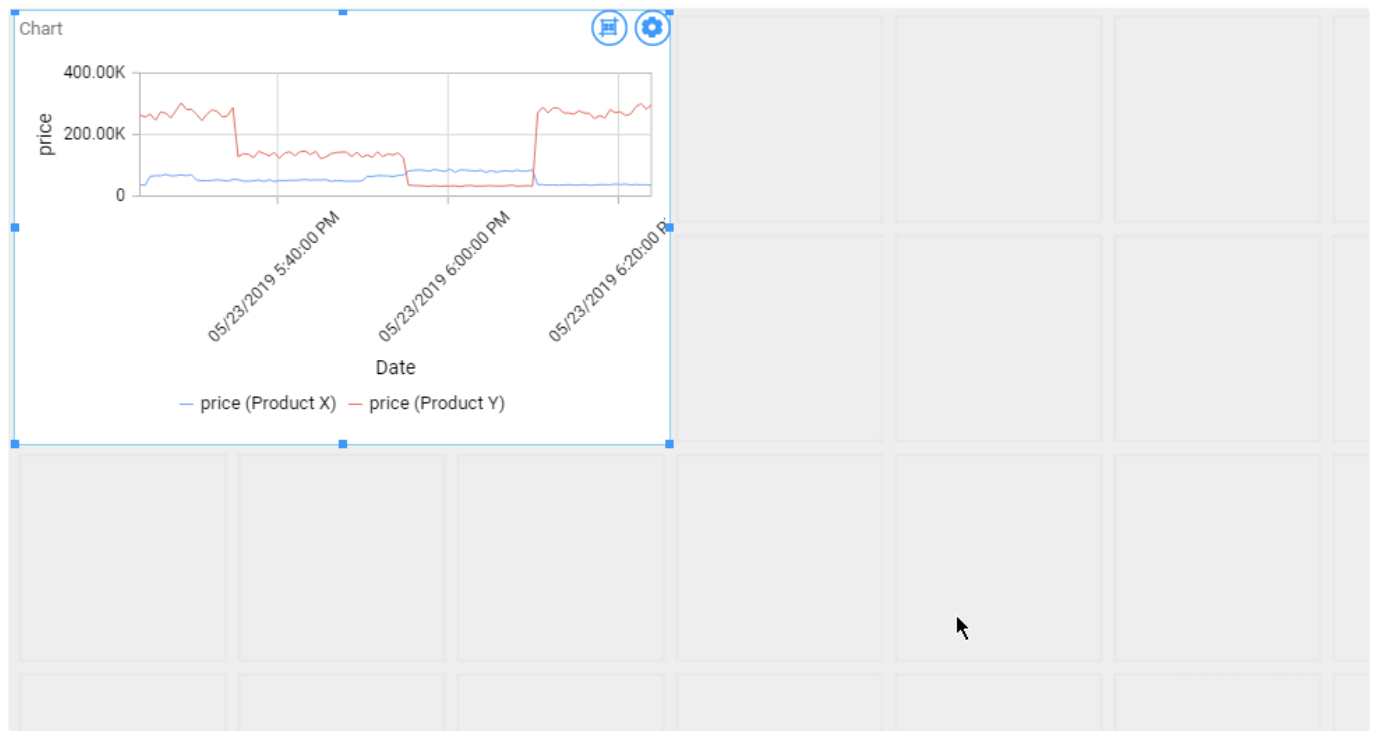


Last modified: 2019/05/27

2.9.3.3. Re-sizing Widgets

To resize a widget, select it to let the handles appear.

Then drag the corner or side handles to resize :



Last modified: 2019/05/27

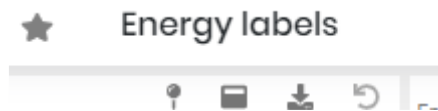
2.9.4. Dashboard-level Settings

Below are documented some settings that are configured at the Dashboard-level.

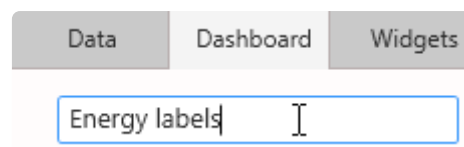
Last modified: 2019/05/28

2.9.4.1. Dashboard Title

Dashboards can have a Title displayed at runtime at the top :



To configure a Dashboard Title, edit the Title textbox on the top-left :



Last modified: 2019/05/28

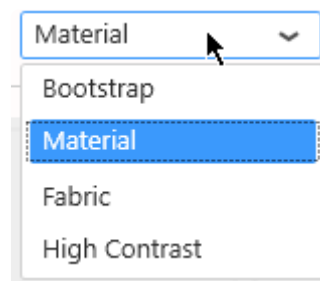
2.9.4.2. Dashboard Theme and Background

The default visual style for runtime widgets and controls is defined by the currently applied **Theme**.

* This Theme is Dashboard property and will only apply to the current dashboard (and all its contained Widgets).

Selecting a Theme

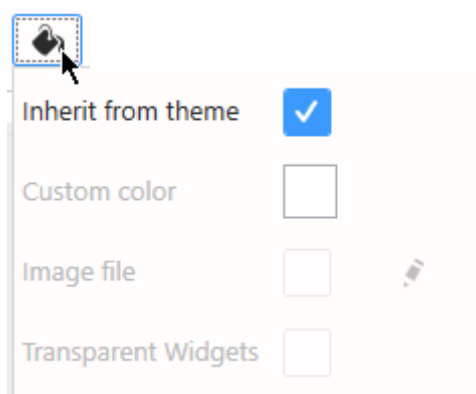
Use the theme Selector in the top toolbar :



Background Settings

The background settings are inherited from the currently selected Theme.

This can be overridden by using the *Background* button in the top toolbar :



Inherit from theme (default)

The background style is defined by the currently selected Theme.

Un-select this option to allow to customize the following options.

Custom Color

The Background Color can be configured manually using a color picker :

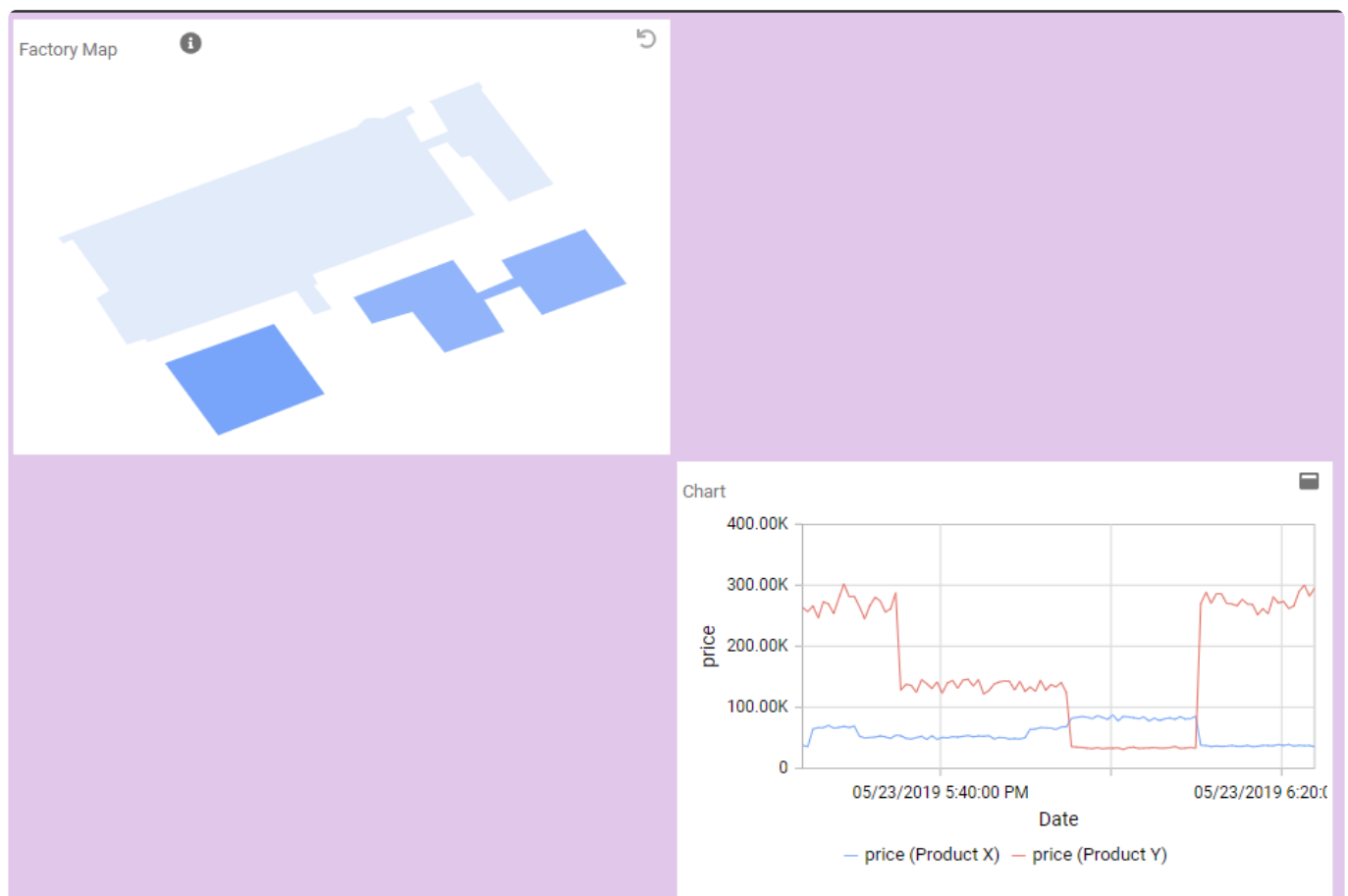
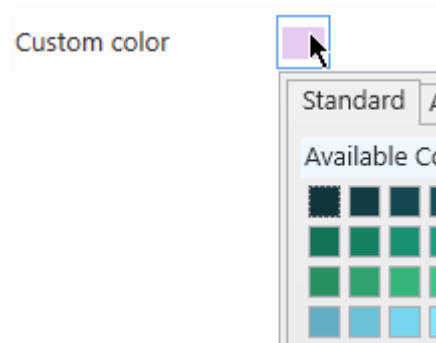


Image file

Instead of a plain background color, you can use an image as background.

Click on the pen icon to select a file.



The maximum size accepted is 1.00Mb

Supported formats are : *.png, *.jpg, *.bmp, *.jpeg, *.gif, *.emf, *.tiff, *.tif, *.ico

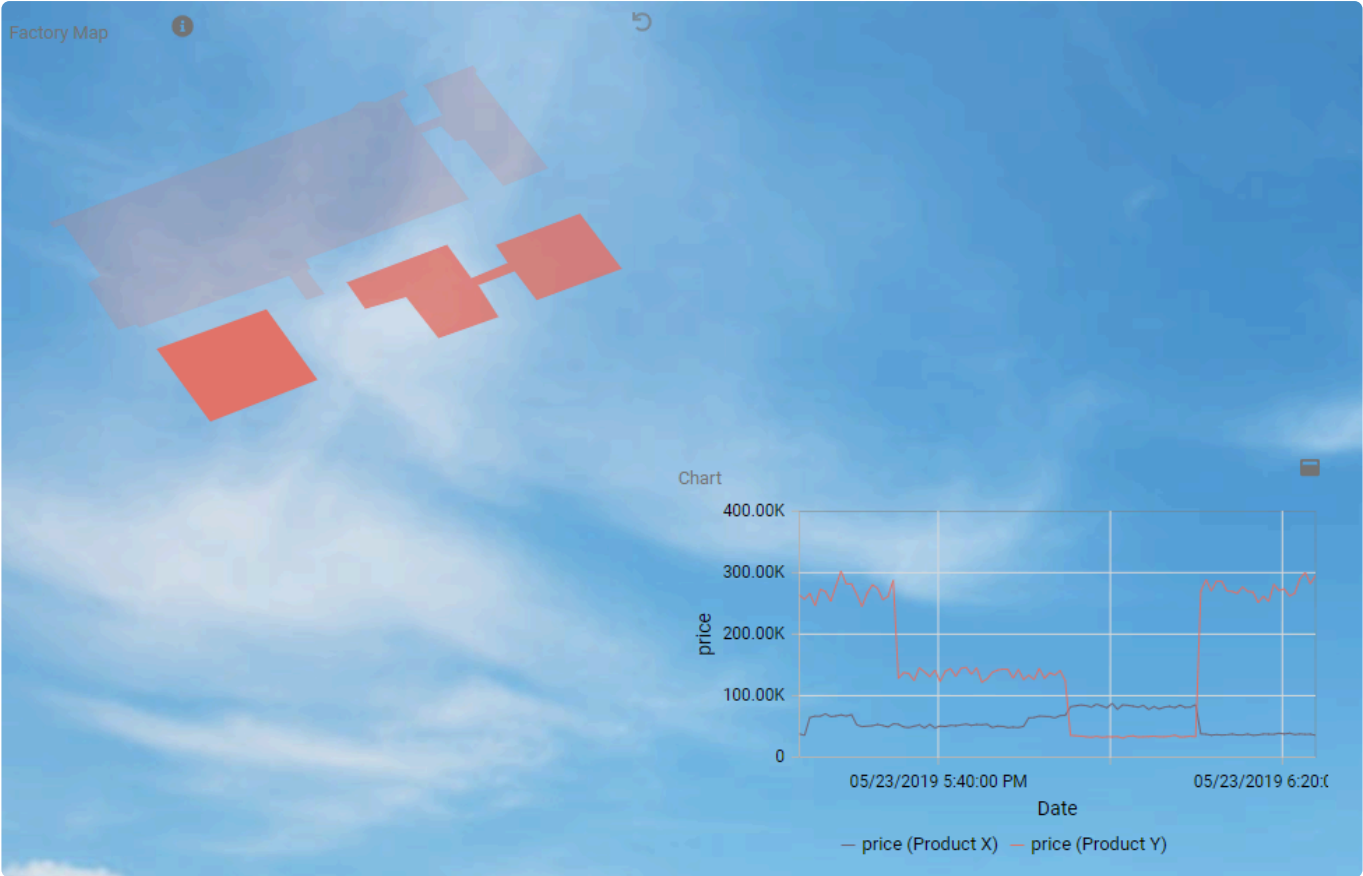


Transparent Widgets

Even when configuring a different background color for the dashboard canvas, the Widgets' background is still inherited from the Theme.

This is in order to maximize readability.

However, you may select *Transparent Widgets* to use the custom background (color or image) as widget background too :



Last modified: 2019/05/28

2.9.4.3. Widget Auto-refresh

What is Auto-Refresh

At runtime, when the Dashboard loads in the browser, data is fetched from the data sources and Widgets are drawn.

After this, if the user doesn't interact with the Dashboard, no new data is fetched and the data displayed can become outdated.

Instead, when Auto-refresh is configured, new data is fetched at regular intervals without needing user intervention.



This is useful for example when Dashboards are displayed on a TV screen without mouse and keyboard.

Auto-refresh ? Buffer refresh ?

Auto-refresh will perform the same action on data as a manual refresh of the web page.

This may be insufficient to get new data if your Connections are bufferized.

For more information on the Buffer and how to refresh it, see the [relevant Chapter](#) in the [Connecting to Data](#) chapter.

Important note on Performance



Setting a small Auto-refresh interval will make more queries to the data. Your system (databases, network, I/O, ...) needs to be able to support as many queries as $(\text{number of open widgets with Auto-refresh}) \times (\text{frequency of Auto-refresh})$. Make sure to do stress and performance tests for your application to scale appropriately.



The Auto-refresh system will perform decently with simultaneously 20 open Dashboards containing each 20 data measures (data points), and 10 Users, when using the recommended [hardware pre-requisites](#).

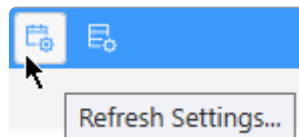
The minimum allowed Time Interval is 30s.

This is for performance reasons on the web server as well as the data server when many users will leave open an auto-refreshed dashboard.

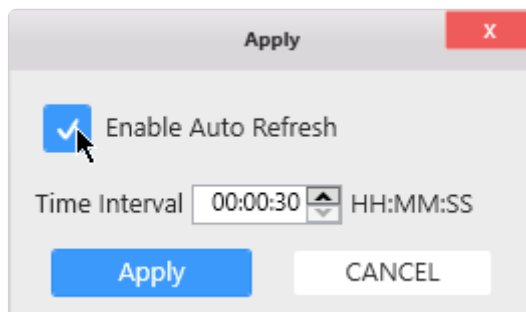
Use this setting sensibly, considering the number of queries that can result from many users leaving the dashboard open on different computers.

Configuring Auto-refresh

To configure Auto-refresh, use the title bar button *Refresh Settings...* :



In the configuration dialog, check Enable Auto Refresh and set a time Interval :

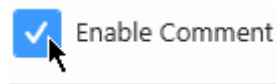


Last modified: 2019/07/23

2.9.4.4. Disabling Comments

Dashboard-level comments can be Disabled.

To Enable or Disable Dashboard-level comments, use the checkbox Enable Comment :



Last modified: 2019/05/28

2.9.5. Designer-level Settings

Alpana Designer maintains some application settings.
See the below chapters for more details.

Last modified: 2019/05/27

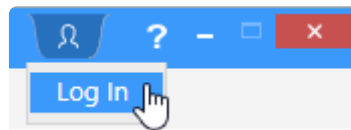
2.9.5.1. Login to Alpana Server

Alpana Designer can log in to Alpana Server.

This is useful for importing/exporting objects to Alpana Server from the Alpana Designer interface.

Logging in

To login, click the profile icon on the top right of the Alpana Designer Window and click *Log In* :



The *Login* window pops up and lets you enter your credentials :

A screenshot of the 'Login' dialog box. The title bar is grey with the text 'Login' and a red close button. The dialog contains the text 'Login with your dashboard server credentials'. Below this are four input fields: 'Server URL' with the value 'https://demo.alpana.io:444', 'Tenancy Name (leave empty if defined by subdomain)' which is empty, 'Username' with the value 'paul', and 'Password' which is masked with dots. At the bottom are two buttons: a blue 'OK' button and a white 'CANCEL' button. A mouse cursor is pointing at the 'OK' button.

* Once you login, you will stay logged in until you explicitly logout (see below). This includes closing Alpana Designer.

Server URL

The full URL to connect to the Alpana Server installation.

This must include :

- the right protocol (HTTP / HTTPS)
- the full name or IP address. If Alpana Server is configured to have Tenants as sub-domains, you can include the sub-domain here too.
- the port number of the Alpana backend website

! Remember : the Alpana Server backend website is usually running on a different port number than the frontend (which you use for viewing dashboards in your web browser)

Tenancy Name

The name of the Tenant you want to connect to.

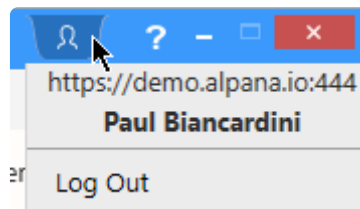
If Alpana Server is configured to have Tenants as sub-domains, you can leave this field empty.

Username / Password

Use the same credentials as for logging in to Alpana Server from the web interface (frontend).

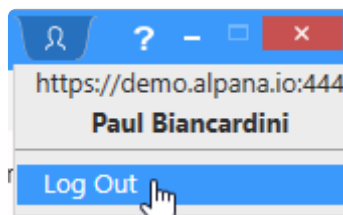
Checking login

To check if you are logged in, click the profile icon on the top right of the Alpana Designer Window and the details window pops up :



Logging out

To logout, click the profile icon on the top right of the Alpana Designer Window and click *Log Out* :



Last modified: 2019/05/27

3. (documentation in progress) Alpana Server

Securely share your Dashboards and Widgets to any modern web browser. Dashboards can be embedded in your Scada Application, your intranet or presented through our secured Server.

Read below the documentation of Alpana Designer to learn more.

Last modified: 2019/05/29

3.1. About Alpana server

Last modified: 2019/06/17

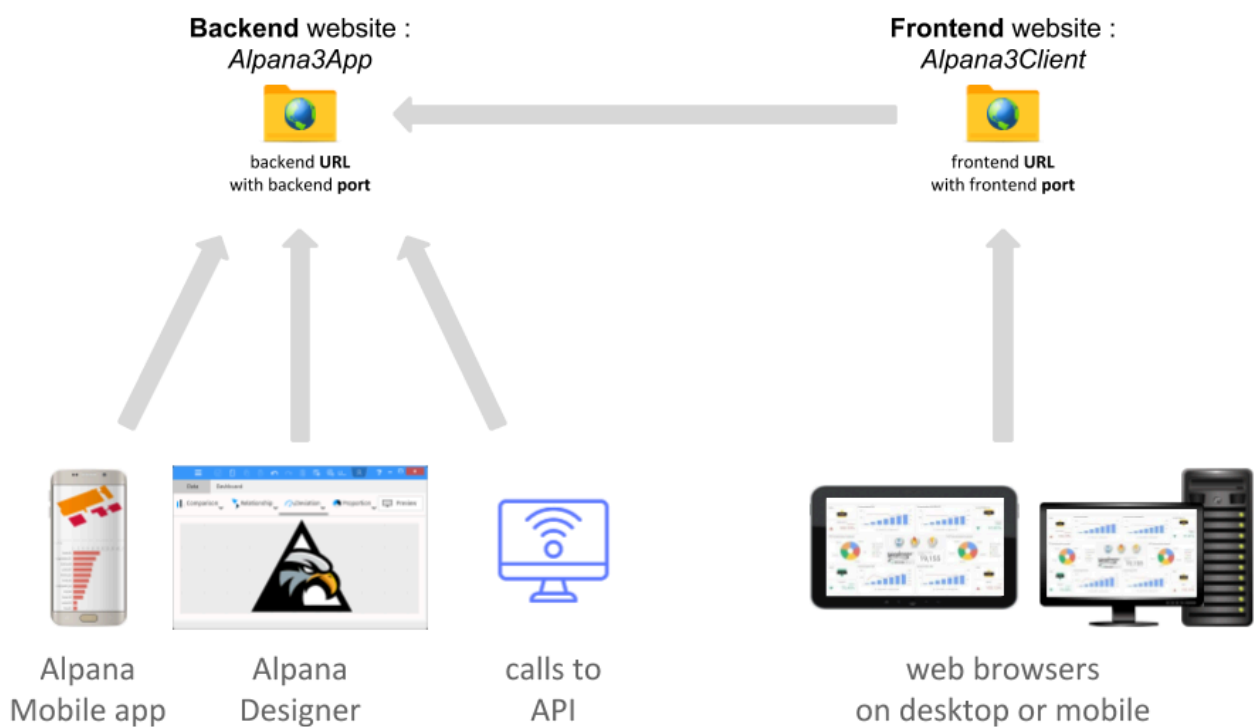
3.1.1. Architecture

Websites

Alpana Server runs two websites on Microsoft IIS :

- the *Back-End Site*
- the *Front-End Site*

Alpana Website architecture



The application is run under Microsoft IIS by the Windows User called `IIS APPPOOL\Alpana3App`

The *Back-End Site*

The *Back-End Site* is named `Alpana3App` in IIS.

This is where the application runs.

Any client application needs to connect to this Back-End :

- Users will connect to this site for publishing from Alpana Designer
- Users will log in to this site when using Alpana Mobile
- User-created applications will connect to this site when using the API
- actually, the *Front-End Site* also connects to the Back-End in the background

The *Front-End Site*

The *Front-End Site* is named `Alpana3Client` in IIS.

This is where the web portal is made available.

Users will connect to this site to view dashboards from a web browser.

Application Databases

Alpana Server also requires two types of databases :

- the actual Alpana Server database
- the Buffer database(s)

These databases are accessed by the Back-End application using SQL Windows Authentication under the Windows User called `IIS APPPOOL\Alpana3App`.

Alpana Server database

The Alpana Server database is named `AlpanaDb` by default.

It stores all information about the application (users, dashboards metadata, logs, ...)

Buffer database(s)

Each Tenant of Alpana Server has its own Buffer database called `AlpanaBuffer_X`, where `X` is the internal ID of the Tenant. (e.g. : on a single-tenant installation, the database is called `AlpanaBuffer_1`).

For more explanations about the Buffer, [see the dedicated chapter](#).

Access to user data

Dashboards point to data through [Connections](#).

When a Dashboard makes a new query to a Connection, new data is fetched by the Back-End application.

This means in particular :

- When using Windows Authentication in connections, they will be run by the corresponding IIS Windows user `IIS APPPOOL\Alpana3App`. So this user must have appropriate permissions to the data.
- This data access must be allowed by firewalls, port redirection, anti-virus, etc...

Folders

Website folders are by default located inside `C:\inetpub\alpana3\`.

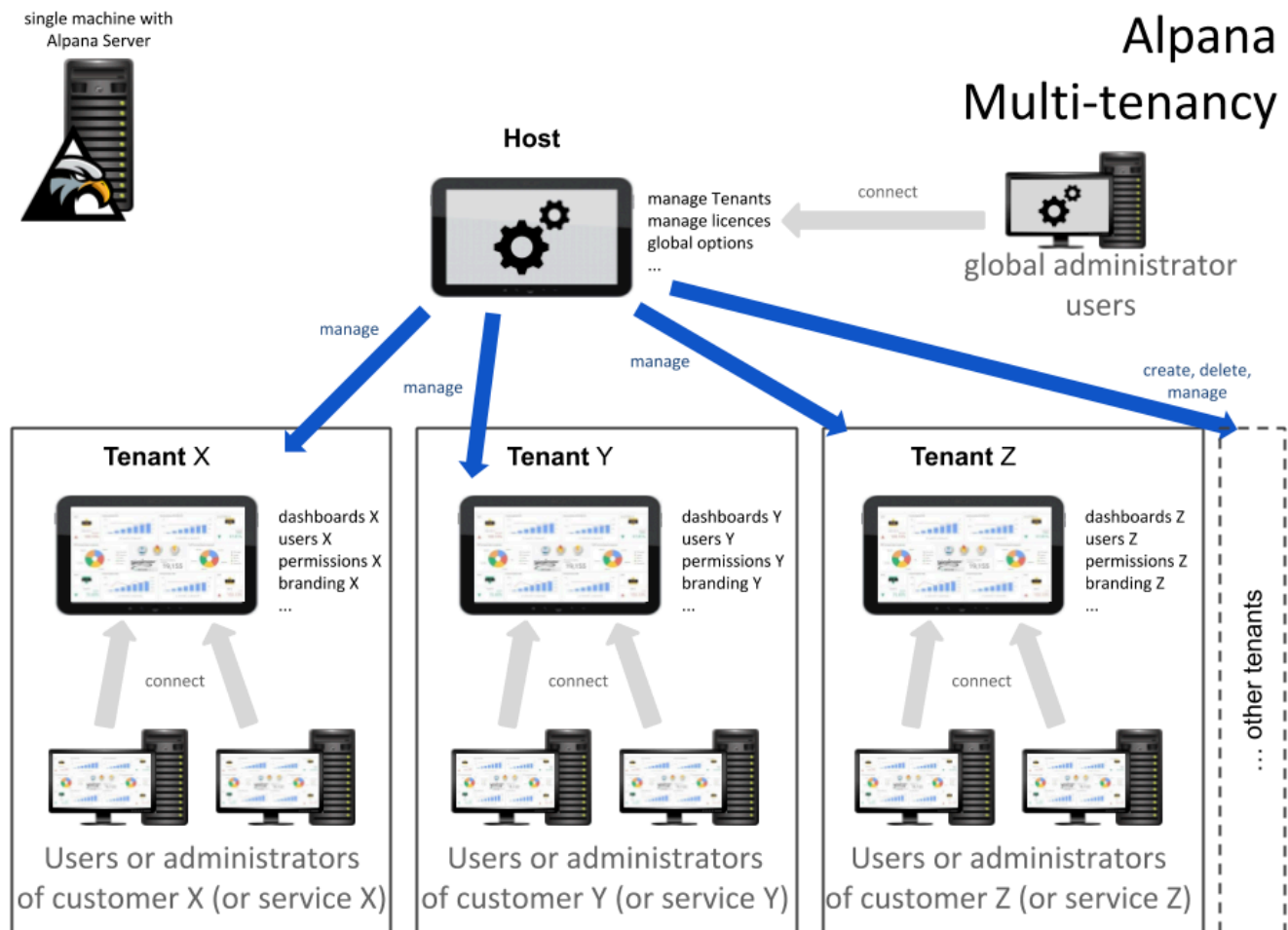
Under this folder, user data relevant for backup is located under `\app\App_Data\`.

In particular, the actual published Dashboards are located inside `\app\App_Data\Published\`.

Last modified: 2019/08/13

3.1.2. Multitenancy

What Is Multi-Tenancy?



The term “software multitenancy” refers to a software architecture in which a single instance of software runs on a server and serves multiple tenants. Systems designed in such manner are often called shared (in contrast to dedicated or isolated).

A tenant is a group of users who share a common access with specific privileges to the software instance. With a multitenant architecture, a software application is designed to provide every tenant a dedicated share of the instance – including its data, configuration, user management, tenant individual functionality and non-functional properties.

Multitenancy contrasts with multi-instance architectures, where separate software instances operate on behalf of different tenants.

([Wikipedia](#))

In short, multi-tenancy is a technique that is used to create **SaaS** (Software as-a Service) applications.

Host > Tenant

We define two notions for a multi-tenant system :

- *Tenant*: A customer which has its own dashboards, users, permissions, settings... and uses the application completely isolated from the other tenants.
A “*tenant user*” is a user owned by a tenant.
- *Host*: There is always a **single** Host. The Host is used for creating and managing tenants.
A “*host user*” is at a higher level : it is independent from all tenants and can control them.

Last modified: 2019/08/13

3.2. Installation

To install Alpana Server, please follow the below chapters.

[Pre-Requisites](#) are very important, in particular [IIS Configuration](#).

Last modified: 2019/06/18

3.2.1. Pre-Requisites

This section explains the system requirements to run Alpana Server.

Software

Operating System

Windows OS (x64) :

- Windows Server : 2012 R2, 2016
- Windows : 10 (Version 1607+)

Database

Microsoft SQL Server is required.

Microsoft SQL Server Express possible, but see limitations related to this edition of SQL (concurrency, database size, resource usage, etc).

During installation, you will need a login to SQL Server which has the permissions to create users and databases.

Other software

! Pre-requisites order is very important.
If you modify any IIS setting after installing ASP.NET Core, then you will have to Repair ASP.NET Core : see [Installation Troubleshooting](#).

Pre-requisites must be installed **in the following order** :

1. [Microsoft IIS](#) : this must be installed and configured **first**
2. PowerShell must be active
3. .NET Framework 4.8 can be found [here](#) then download the "Runtime". Alternately, an offline installer exists on the same page . **Installing this component requires to restart the machine.**
4. [ASP.NET Core 2.2](#) : select version "Runtime & Hosting Bundle". **Installing this component requires to restart IIS.**
5. [IIS URL Rewrite](#) : this should be installed after IIS

- ! You must restart IIS after installing ASP.NET Core.
- ! You must restart the computer after installing a new version of .NET Framework.

Hardware Requirements

- ! Hardware requirements highly depend on usage. Typically, SQL and IIS will take disk space, disk I/O, RAM and CPU depending on several factors like database usage, dashboards complexity and website traffic. Please see with your website and database administrators to size your application appropriately.

Hardware Minimum

- ! Those are the bare minimum that will get you a running installation. As soon as you will add content, add data, and increase traffic, **you will need more.**

- HDD : 1GB
- RAM : 3GB including OS
- CPU : any dual core CPU, x86 or x64 architecture

Hardware Recommended

To go beyond the minimal installation, you will need evaluate your requirements based on traffic, number of dashboards, number of tenants, other software and databases hosted on the same machine, etc...

Typically, it's recommended to have at least :

- SSD for fast I/O, 50GB+ with free space always available (usual system administrator advice apply)
- RAM : 8GB for a small/medium site with few databases and no other big software installed
- good server CPU, like recent Xeon

This can be much more if your application serves many dashboards to many users on many tenants.

Last modified: 2019/07/11

3.2.1.1. IIS Configuration pre-requisite

Required IIS configuration

! Pre-requisites order is very important.
If you modify any IIS setting after installing ASP.NET Core, then you will have to Repair ASP.NET Core : see [Installation Troubleshooting](#).

In order to install Alpana Server, Microsoft IIS must be installed and configured with the following options :

- Under *Application Development Features* :
 - *.NET Extensibility 3.5*
 - *.NET Extensibility 4.7*
 - *ASP.NET 4.7*
 - *ISAPI Extensions*
 - *ISAPI Filters*
 - *WebSocket Protocol*
- Under *Common HTTP Features* :
 - *WebDAV Publishing*
- Under *Performance Features* :
 - *Dynamic Content Compression*
- Under *Security* :
 - *Windows Authentication*
- Under *Web Management Tools* :
 - *IIS Management Console*

How to configure

On server editions of Windows (Server 2012 R2, Server 2016), IIS can be installed and configured by using the Server Manager, with the menu "Manage > Add roles and features".

On desktop editions of Windows (Windows 7, 8, 10), Microsoft IIS can be installed and configured by opening "Turn Windows features on or off"

Last modified: 2019/07/26

3.2.2. Installing

Important



Before installing, make sure you meet the [Pre-Requisites](#).



Alpana Server will create folders and SQL databases.

Download installer

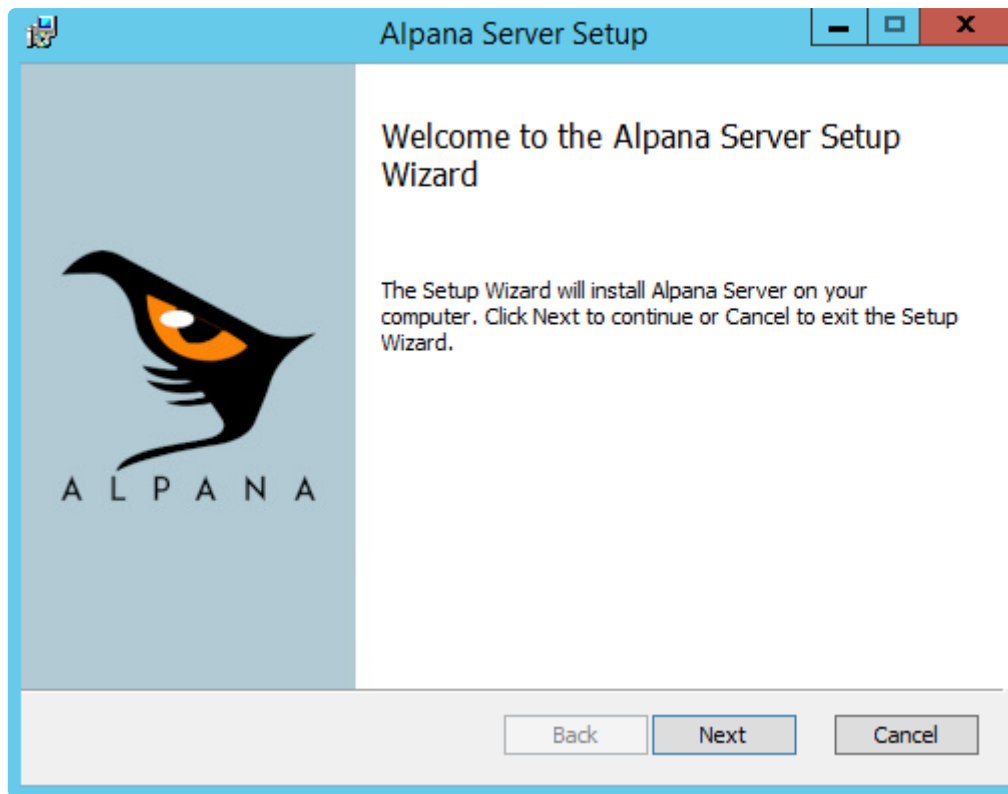
[See here to download the installer.](#)

Running the Alpana Server installer

Login with administrator rights to the machine where you want to install Alpana Server.

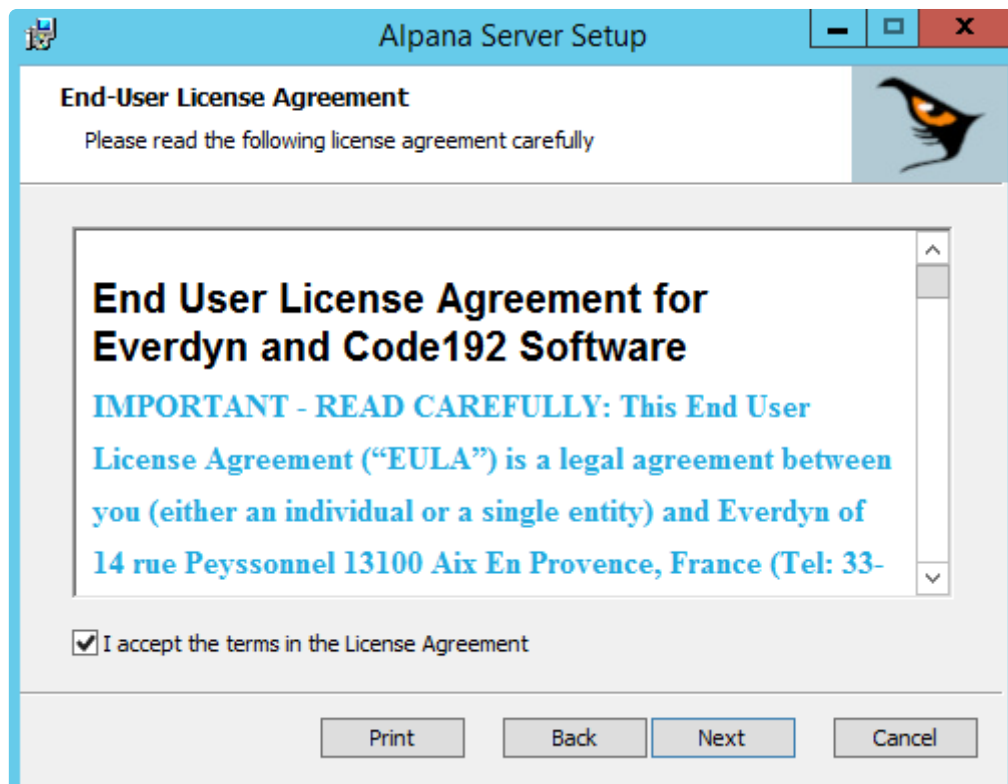
Start the Alpana Server installer `.msi` file to start the installation.

At the installer welcome message, click *Next* :



Accepting EULA

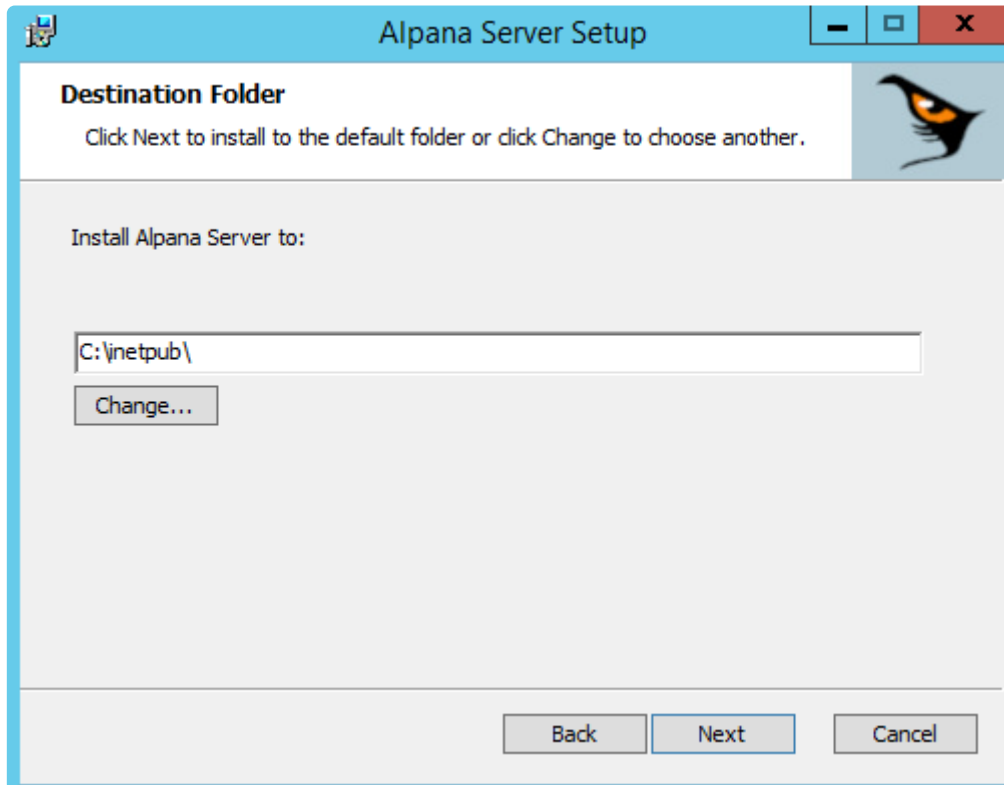
Accept the End User License Agreement and click *Next* :



Configuring website folder

Alpana Server will be installed as IIS websites in this machine.

Use this page to configure the folder on the disk where the website files will be hosted :



Configuring SSL or DNS

Use this page to optionally configure :

- SSL : insert a certificate to let users connect to your Alpana Server website using HTTPS
- DNS : bind a domain name to let users connect to your Alpana Server using a friendly name like `www.alpana.io`, or write a static IP address that is not currently available

Tenant names as subdomains

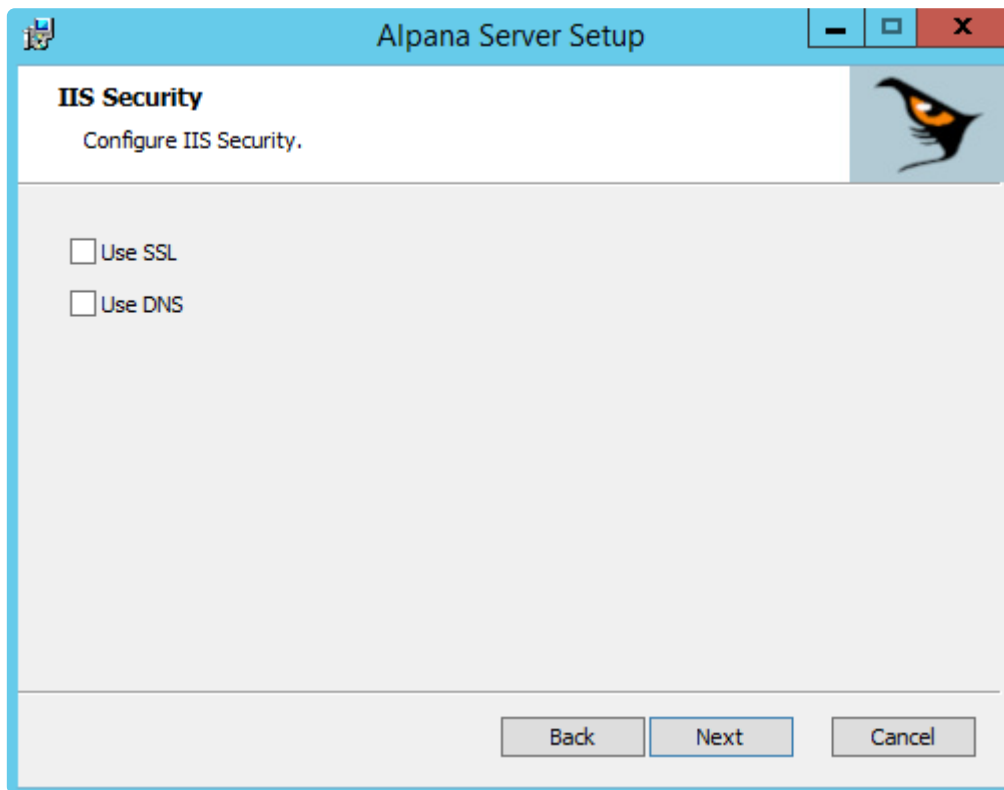
If you use DNS with a domain name, you will be able to create subdomains.

The installer can be configured to automatically generate a subdomain for each created Tenant.

For example, if your domain is "mydomain.com", and you have 3 Tenants, they each will have the following URL :

- tenant "Default" : URL `http://default.mydomain.com`

- tenant “CustomerX” : URL `http://customerx.mydomain.com`
- tenant “CustomerY” : URL `http://customery.mydomain.com`



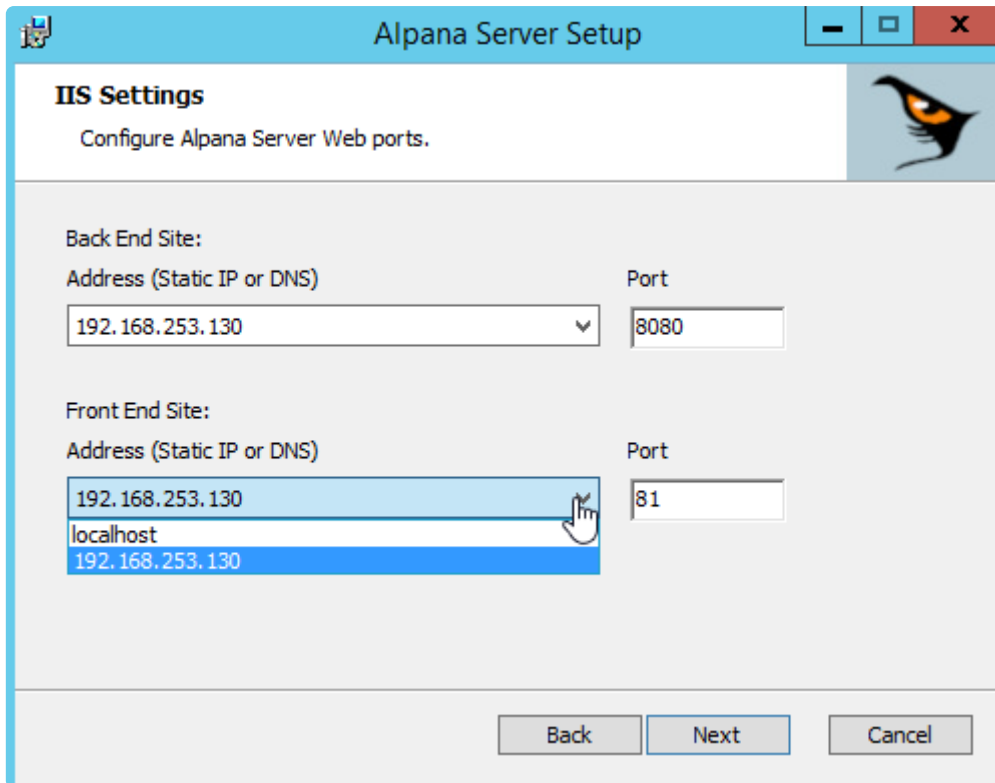
Configuring address and ports of websites

Alpana Server actually consists of two IIS websites :

- the *Back-End Site* is where the application runs. Users will connect to this site for publishing from Alpana Designer, for logging in using Alpana Mobile, or when using the API
- the *Front-End Site* is where the web portal is made available. Users will connect to this site to view dashboards from a web browser.

If a port is invalid or unavailable, select another port.

If you have several IP addresses (Network Interface Cards), select the desired IP address from the list, or “localhost”.



The image shows a Windows-style window titled "Alpana Server Setup" with a blue header bar. Below the header, the title "IIS Settings" is displayed in bold, followed by the instruction "Configure Alpana Server Web ports." in a smaller font. To the right of the text is a small icon of a stylized eye. The main area of the window is divided into two sections: "Back End Site:" and "Front End Site:". Each section has two input fields: "Address (Static IP or DNS)" and "Port". In the "Back End Site:" section, the "Address" field contains "192.168.253.130" and the "Port" field contains "8080". In the "Front End Site:" section, the "Address" field is open, showing a list of three options: "192.168.253.130", "localhost", and "192.168.253.130". The "localhost" option is currently selected and highlighted in blue. A mouse cursor is pointing at the "localhost" option. The "Port" field for the "Front End Site:" contains "81". At the bottom of the window, there are three buttons: "Back", "Next", and "Cancel".



When selecting "localhost", Alpana Server will not be reachable from another computer.



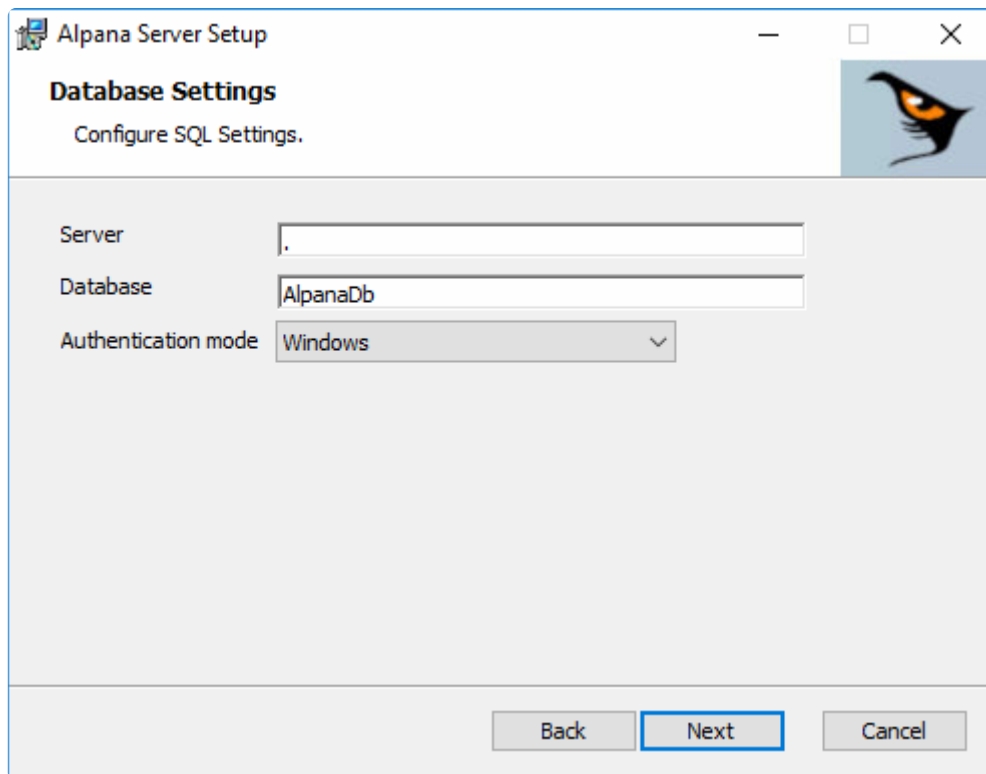
When using a domain name, Alpana will not respond with the IP address. See [Troubleshooting](#).

Configuring the SQL connection

Alpana Server stores its state inside a SQL database.

Use this page to configure the connection string to the database.

This connection string must allow the installer to create databases and give permissions to users.



* Two databases are created : the Alpana Server database for storing the state of Alpana Server, and the [Buffer database](#) for temporarily storing some dashboard data.

! The Alpana Server database is created by default with recovery mode *Simple*.

Server

This is the address of the SQL Server instance you need to access. Possible values are :

- . or localhost : local server
- <NetBIOSName> : server named with a NetBIOS name
- <IPv4Address> : server accessible through a IP v4 address
- <IPv6Address> : server accessible through a IP v6 address
- <server address>\<instance name> : connect to a specific instance. For SQL Express, it may be necessary to use <server address>\SQLEXPRESS
- <server address>,1433 : 1433 is the default port for SQL Server, but can be changed like this

Database

The desired name for the Alpana database.

Authentication Mode

“SQL Server” : will use SQL Server authentication with “User name” and “Password”.

“Windows” : will use Windows Authentication from the Windows user that generates the query.

Note on Windows Authentication

Please note that **two** Windows users will be using this connection string :

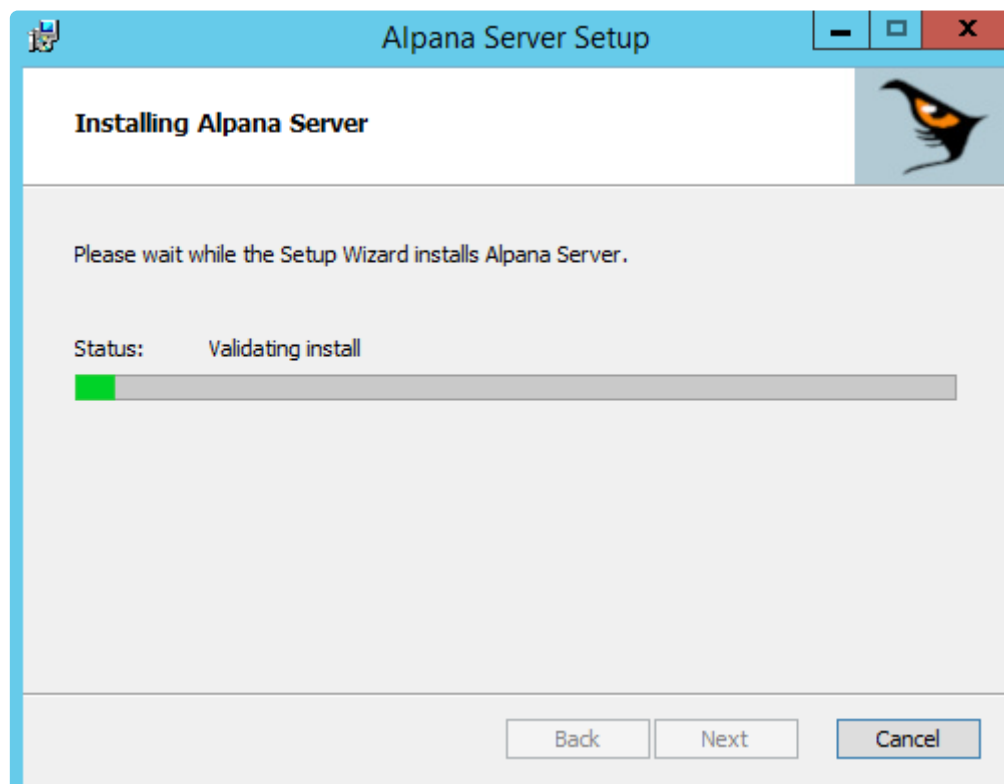
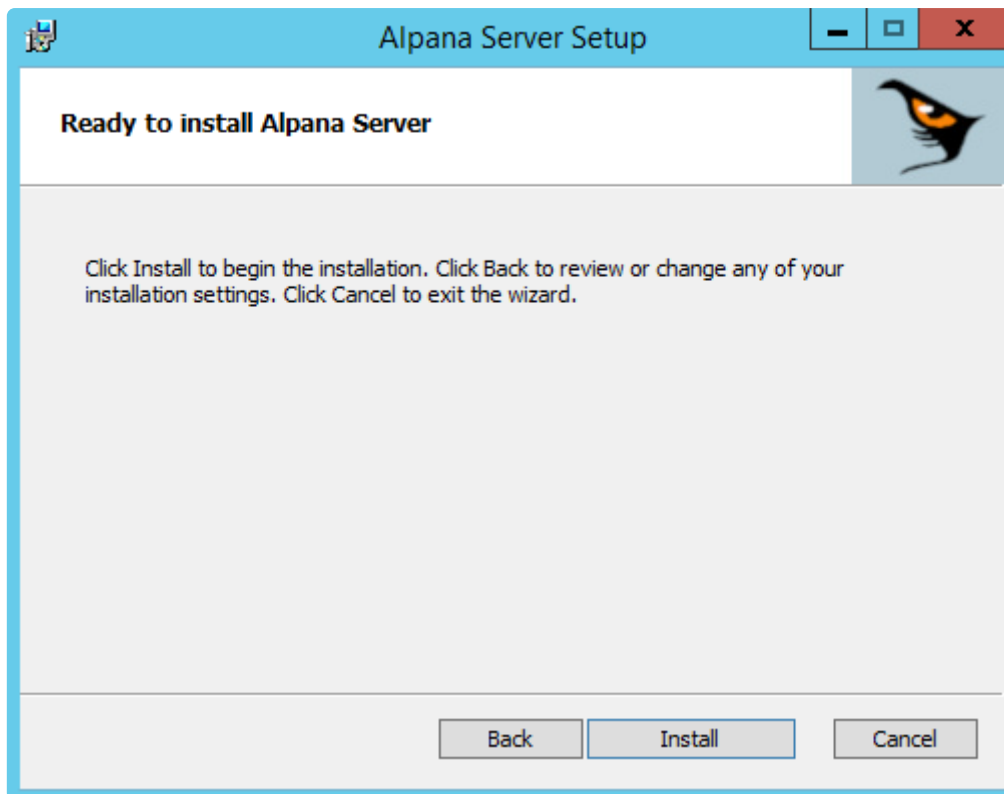
- The Windows user who runs the installer. The installers authenticates to SQL as this Windows user to create databases and transfers rights to the second user below :
- The Windows user who runs the Application Pool of the IIS Website for Alpana Server. Its name is `IIS APPPOOL\Alpana3App`.



Both Windows users must have the necessary permissions to SQL.
In particular, the Windows user you are using to run the Alpana Server installer must have admin privileges in SQL.

Completing the installation

Confirm the configuration by clicking *Install* :



Installation troubleshooting

If you encounter some issues during or after installation, see [Installation Troubleshooting](#) and contact

Support.

Next Steps

After successfully installing, you will have to [login to the Front-End](#) and [activate your license](#).



Application startup takes some time, this is normal.

When starting the IIS website, your browser should display a “loading” animation for some time (usually about 1min).

Last modified: 2019/07/25

3.2.3. Note : Upgrading from v3 to v3

To upgrade Alpana Server from v3 to a newer v3, please follow the below procedure :

1. read the corresponding [Release Notes](#)
2. uninstall any previous version of Alpana Server v3 using Windows Control Panel
3. make sure you meet the [Pre-Requisites](#)
4. start a new [installation](#) with the same information



When upgrading Alpana Server, please instruct all your users to clear their browser cache or do a hard refresh.

Last modified: 2019/08/27

3.2.4. First login

! Application startup takes some time, this is normal.
When starting the IIS website, your browser should display a “loading” animation for some time (usually about 1min).

Default password

When Alpana Server is started for the first time, you can login to the Host using the following credentials :

User name : admin

Password : 123qwe

! Important : As with any default password, please change this password as soon as possible !

A L P A N A

Current tenant: Not selected ([Change](#))

Log in

admin

...

☐ Remember me

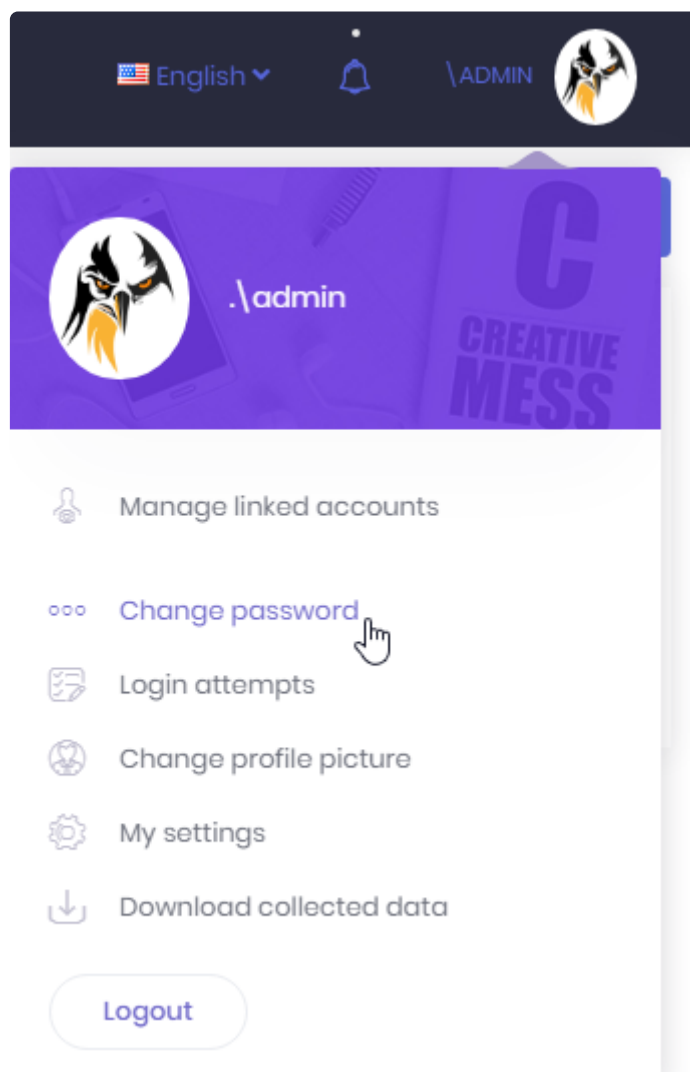
[Forgot password?](#)

Log in



Changing default password

Click on your “admin” username on the top right, and in the menu select *Change Password* :



Change password

Current password

New password

New password (repeat)

Cancel

Save

Last modified: 2019/07/05

3.2.5. Un-Installing

You can un-install Alpana Server using Windows Control Panel.

Tips

Don't lose your license !

Before un-installing, perform a [license de-activation](#) in order to be able to re-use your license later.

Avoid a reboot

You can usually avoid a reboot by first manually stopping the IIS websites for Alpana :

1. Windows > Start > Run program `inetmgr`
2. Select each Alpana website and click *Stop*.

Last modified: 2019/07/19

3.3. License Management

License is managed at the Host level for all Tenants.

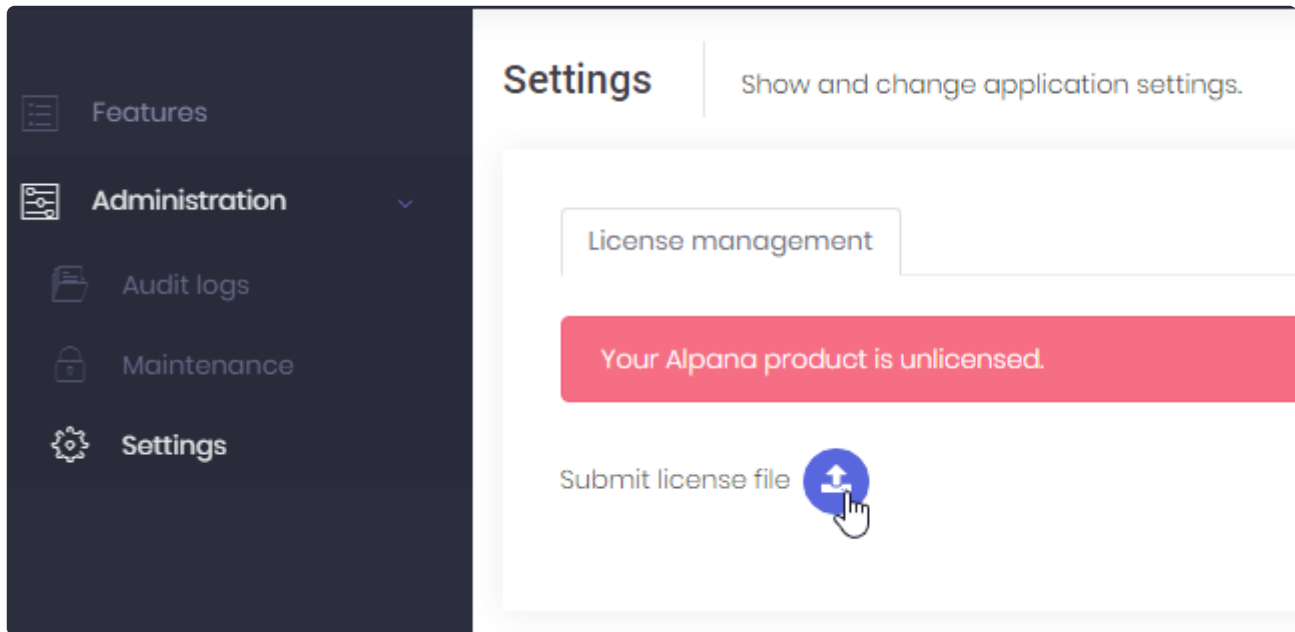
Login to the Host of your Alpana Server and navigate to *Administration > Settings*, tab *License Management*.

See the below chapters to manage your license.

Last modified: 2019/06/17

Activating the License File

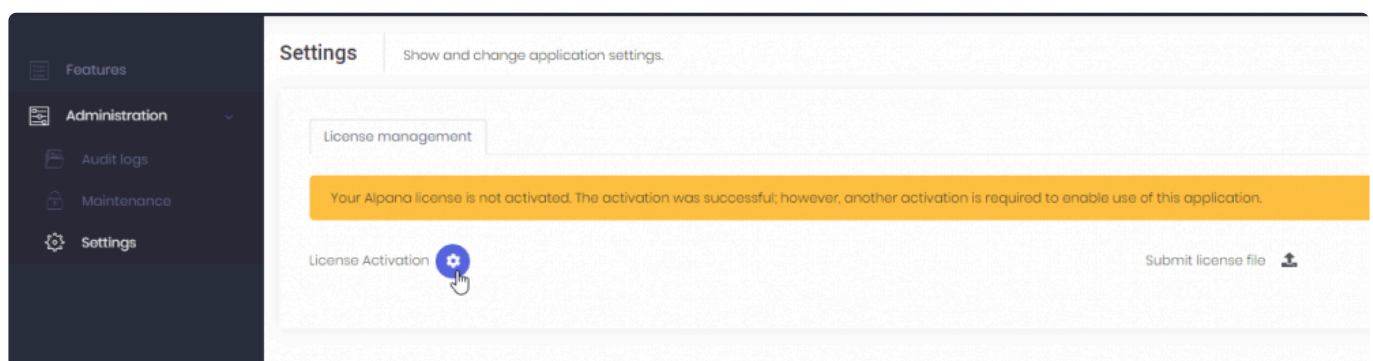
On the Host of your Alpana Server, navigate to *Administration > Settings*, tab *License Management*. Click *Submit License File* and upload the `AlpanaServer.lic` file :



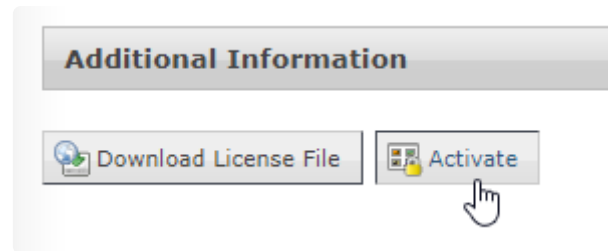
⇒ The first step of license activation is complete, but your license is not yet activated.

Activating the License

On the Host of your Alpana Server, click *License Activation* and leave the pop-up window open :



On the web License Portal, click *Activate* and leave the pop-up window open :



On the Host of your Alpana Server, copy the *User Code 1* and *User Code2* :

License Activation

User Code 1

XXXX

User Code 2

YYYY

Activation Code

...and paste them in the correct fields in the web License Portal, and click *Activate* :

*User Code 1: XXXX

*User Code 2: YYYY

Activate Cancel

⇒ As a result, the web License Portal displays an *Activation Code* :

Activation Code 1: ZZZZ


Copy the *Activation Code* and in the Host of Alpana Server, paste it and click to save :

License Activation ×

User Code 1
XXXX

User Code 2
YYYY

Activation Code
ZZZZ



⇒ The License should be activated.

* If the License Activation was successful, you may delete the `AlpanaServer.lic` file.

Last modified: 2019/06/17

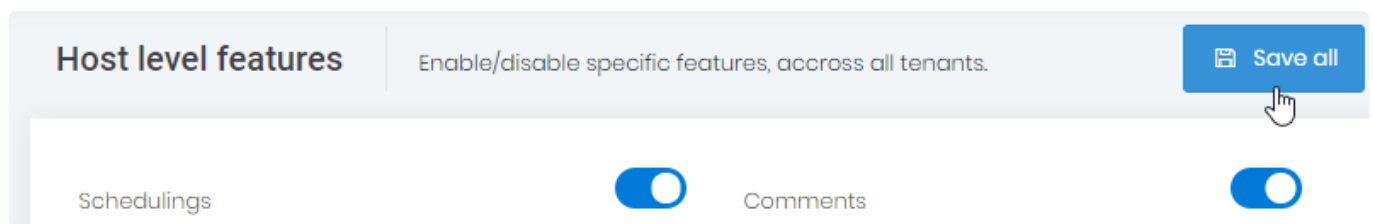
3.3.2. Host-level Features

Depending on your license type, some features may be activated at the Host level.

Login to the Host and browse to *Features* :



Check the required Features and click *Save* :



Then, in order for Tenants to benefit from the features, you must [allocate them](#).

Last modified: 2019/06/18

3.3.3. License Deactivation

Why Deactivate

You may want to return the license from a Host in order to decommission the Host and use the license on another Host.

How to Deactivate a License

! Warning : This will remove the license from the current Alpana Server installation. Until a new license is activated, you users won't be able to login or view dashboards.

✿ Note : You may at a later time activate again a license on this installation if needed.

On the Host of your Alpana Server, navigate to *Administration > Settings*, tab *License Management*.


Click *Deactivate License* :

Deactivate license 

Read the warning and confirm the deactivation :

Deactivate license ×

By clicking button below, your Alpana license will be definitively discarded. Please copy and keep the deactivation code that will be returned by the system.

DEACTIVATE 

Please copy and keep the *Deactivation code*

Deactivate license

By clicking button below, your Alpana license will be definitively discarded. Please copy and keep the deactivation code that will be returned by the system.

Deactivation code

XXXX



If you fail to keep your *Deactivation code*, you may not be able to re-use your license at a later time.

Contact Support

Contact Support with your *License ID* and *Deactivation Code* to allow to free your license.

Last modified: 2019/06/17

3.4. Troubleshooting

This chapters explains how to find error logs and how to fix common issues.

Last modified: 2019/07/25

3.4.1. Installation Troubleshooting

Generating install logs

In some cases, the installation of Alpana Server may fail and roll back the changes.

To get more information, you can start the installer with error output by following the below procedure.

Start cmd

Run Windows command prompt “cmd” **as administrator** and navigate to the folder where the installer .msi file resides (for example here `code192.Alpana.Server.Setup_3010.msi`).

Start installer with log output

From the command prompt, start the following command :

```
msiexec /l*vx Alpana_install.log /i code192.Alpana.Server.Setup_3010.msi
```

Replace `Alpana_install.log` with the desired log file name that will be created in the folder.

Replace `code192.Alpana.Server.Setup_3010.msi` with the file name of your Alpana Server installer.

Send log

When the installation has completed, you can send the zipped “Alpana_install.log” file to code192 support.

Don’t forget to include any relevant information about your installation (Windows/SQL/IIS versions ? What pre-requisites were configured and installed ? What configuration choices did you make during installation ? ...)

Post-installation issues

Website startup issues

If the Front-End doesn’t seem to start, try to open the URL of the Back-End directly in your web browser.

- If you see a login page on the back-end, then the Front-End should start, see [error logs](#) and contact Support.
- If instead you see a “Error HTTP 500” message or similar, then you are probably missing some [pre-requisites on IIS](#).

When changing the configuration to IIS, you must repair the ASP.NET Core component :

1. update IIS configuration
2. go to Windows Control Panel and for each component of ASP.NET Core, perform a "Repair".
Usually, this is done through right-click > Change, then select *Repair*.

License activation issues

When uploading the license file, if nothing seems to happen, here is a possible solution :

Check the host Audit Logs.

If a warning is present at that time, check the details.

If the details contain a message like "access denied" with a file location, then you may have permission issues in the IIS website folders.

Make sure that IIS and the Windows user running Alpana (`IIS APPPOOL\Alpana3App`) have permissions to create files in sub-folders of the installation folder.

Last modified: 2019/07/25

3.4.2. Log Files

Alpana Server Error Log Files

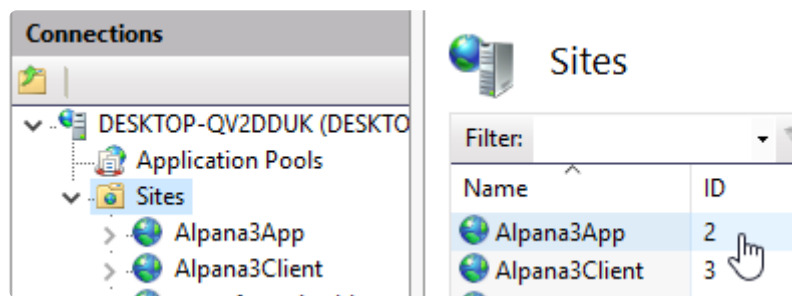
If for some reason you are unable to access the error logs from the web interface of the Front-End, you can find the logs as text files in the following folder :

(under your installation folder, by default `C:\inetpub\alpana3\) .\app\App_Data\Logs\`

IIS Requests Logs

If you need to investigate your network configuration to see the content of HTTP requests, you can find IIS log files :

1. Open IIS Manager (run `inetmgr`)
2. Click on the *Sites* list to get the ID of the desired website. For example in this screenshot, the Alpana3App ID is 2 :



3. With Windows file explorer, browse the following folder (replace the **2** with your desired ID from step 2. above) :

```
C:\inetpub\logs\LogFiles\W3SVC2\
```

4. This folder contains logs as space-separated text files ordered by date, containing all requests.

Browser errors

Sometimes, the web browser console gives indications on the nature of the problem.

See your web browser documentation to open the console.

For Google Chrome, press F12 and click the "Console" tab.

Last modified: 2019/07/25

3.4.3. Website Troubleshooting

Below are some answers to common issues with reaching or starting the websites.

Frontend website loading forever

You are trying to open the Frontend website, but you have a white page with a loading animation that never completes :



Wrong Frontend address used

Symptom

Check the web browser console for an error like :

```
Access to XMLHttpRequest at 'http://xxxxx.com:123//AbpUserConfiguration/GetAl1?d=1234567890' from origin 'http://yy.yy.yy.yy' has been blocked by CORS policy: Response to preflight request doesn't pass access control check: No 'Access-Control-Allow-Origin' header is present on the requested resource.
```

Check that you can reach the Backend from your same web browser on the same URL, and see a login page there.

Explanation

If so, a possible explanation is :

Your Backend at `http://xxxxx.com:123` is configured to not respond to queries from `http://yy.yy.yy.yy`.

For example, when you install Alpana Server and configure to use a domain name with `xxxxx.com:80` as Frontend and `xxxxx.com:123` as Backend, you won't be able to access the Frontend using its IP

address `yy.yy.yy.yy`.

That's for security reasons : the Frontend will only be allowed to talk to the Backend if it calls from the pre-defined origin (the domain name `xxxxx.com:80`).

That is configured through [CORS](#) rule internally.

Solving

You can solve this case by redirecting requests in IIS.

For example, if you wish to be able to reach the Frontend from IP address `yy.yy.yy.yy`, you need to let IIS listen to requests to `http://yy.yy.yy.yy` and redirect them to the accepted domain name `xxxxx.com:80`.

See [IIS documentation](#) examples on how to do this.

Last modified: 2019/07/25

3.5. Maintenance and Migration

This chapter describes commonly needed maintenance operations.

Last modified: 2019/06/17

3.5.1. Backup

Introduction

Two elements of Alpana Server must be backed up :

- the Alpana Server database, named `AlpanaDb` by default
- the dashboards folder located in the installation folder, by default `C:\inetpub\alpana3\`, inside `\app\App_Data\Published\`

Backup

- * To perform a backup, it is recommended to stop the websites first in order to ensure consistency between the database and the folders. (e.g. : a dashboard is uploaded between the database backup and the folder backup)
If this is not desired, one should at least backup the folder **after** the database.

Database backup

Locate the Alpana Server database, named `AlpanaDb` by default.

Use standard Microsoft SQL procedure to perform a full backup of the Alpana Server database.

- * The Alpana Server database is created by default with recovery mode *Simple*.
If you wish for a different setting, you are free to change it.

Folder backup

Locate the installation folder, by default `C:\inetpub\alpana3\`.

Copy and compress the `\app\App_Data\Published\` folder.

Restore

- * Before deleting any item, make sure you make a backup first.



Restoring requires to stop the websites, so this will incur downtime to your users.

Stop the websites

Open the IIS Management Console with : Windows > Run... `inetmgr`.
Select the two Alpana Server websites and click *Stop* for each of them.

Restore the folder

Locate the installation folder, by default `C:\inetpub\alpana3\`.

Replace the `\app\App_Data\Published\` folder by the backup.

Restore the database

Use standard Microsoft SQL procedure to perform a full backup of the Alpana Server database.



Use the same database name as previously on this installation.

Start the websites

Open the IIS Management Console with : Windows > Run... `inetmgr`.
Select the two Alpana Server websites and click *Start* for each of them.

Last modified: 2019/07/11

3.5.2. Migration

Purpose

Case 1 : Development → Staging → Production

You have several environments for your development cycle.

When development has been tested on one environment, you want to migrate the content (dashboards, users, etc) towards the Production environment.

Case 2 : Moving to another machine

You need to move your Alpana Server installation to a different machine (e.g. for architecture reasons).



In this case, it is possible to move the license by de-activating it on the first machine and using it later on the target machine.
Check with your support what is the most appropriate and see [License Deactivation](#).

Migration Procedure

To migrate from *MachineX* to *MachineY*, follow the below procedure :

1. Alpana Server must be installed on *MachineY*. If you are using a separate license, then the *MachineY* license must be activated too (though this can be done at a later time)
2. Perform a [backup](#) on *MachineX*
3. Perform a [restore](#) on *MachineY*

Migrated content

Note : all content from *MachineX* will be migrated :

- Users and their passwords
- Dashboards, Widgets, Data Sources
- Roles, Permissions, ...
- Server Settings, ...

Possible Issues

When migrating, some Dashboards may become unable to access their data.

This commonly happens when data is not accessible from the *MachineY* using the same Connection.

Example : database is not accessible on the new network.

The fix is then to open the Dashboard and edit the Connection accordingly.

Error logs should help you find and fix the issue.

Last modified: 2019/06/17

3.6. Tenant Management

It's run in multi-tenant or single-tenant modes. Multi-tenancy is disabled by default. We can enable it. there is a pre-defined tenant where the Tenancy Name is 'Default' .

Last modified: 2019/05/28

3.6.1. Create Tenant by Host Admin

There are clients / organisations interested in your application and would like to use it . But, the question is , how we can provide access to different organisations without any security breach and in an easy-to-follow way.

Admin should follow below steps to create a new tenant

Let us create a Tenant

- Click on the “**Tenant**” tab in left navigation
- Click on “**Create New Tenant**” button

Actions	Tenancy code name	Tenant name	Active	Creation time
Actions	Default	Default	Yes	12/17/2018
Actions	Harvest	Harvest	Yes	03/19/2019
Actions	test	test	Yes	03/28/2019
Actions	NewTest_Tenancy	NewTest_Tenant	Yes	03/21/2019
Actions	QA	QA Activity	Yes	03/06/2019

- Fill up the **create new tenant** form on the pop up

- Click on the **“Save”** button on the pop up

System will send tenant details to the registered email address . The registered user becomes the admin for the tenant. So, the user has all the permissions necessary to manage the tenant data.

✿ System will send tenant activation email with login credentials on registered Admin email address. User need activate the tenant by click on activation link

Last modified: 2019/05/28

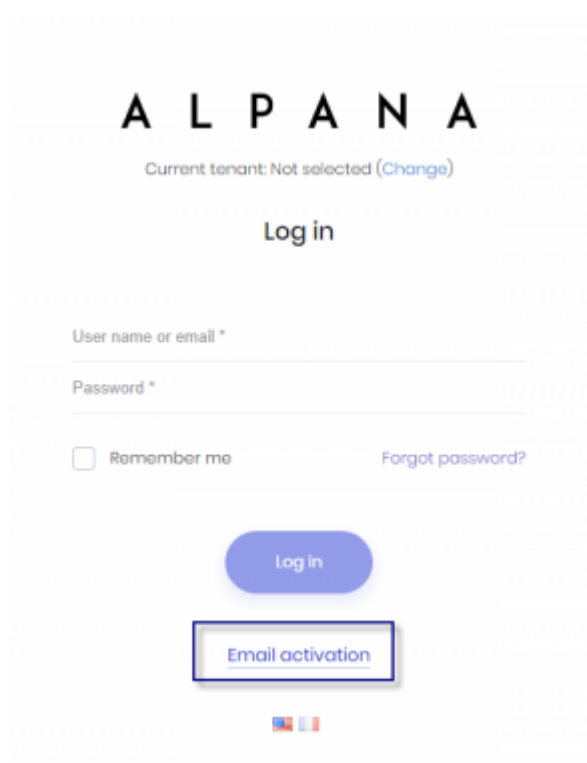
3.6.2. Create Tenant using Email Address

There is one provision for the tenant to register themselves in the application with ease and simplified way.

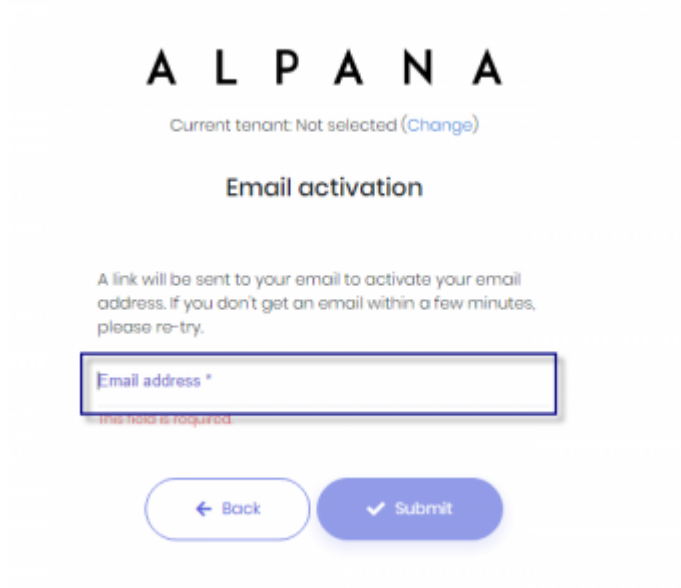
Follow below steps to registered as a tenant

Let's register

- Click on “**Email Activation**” Link

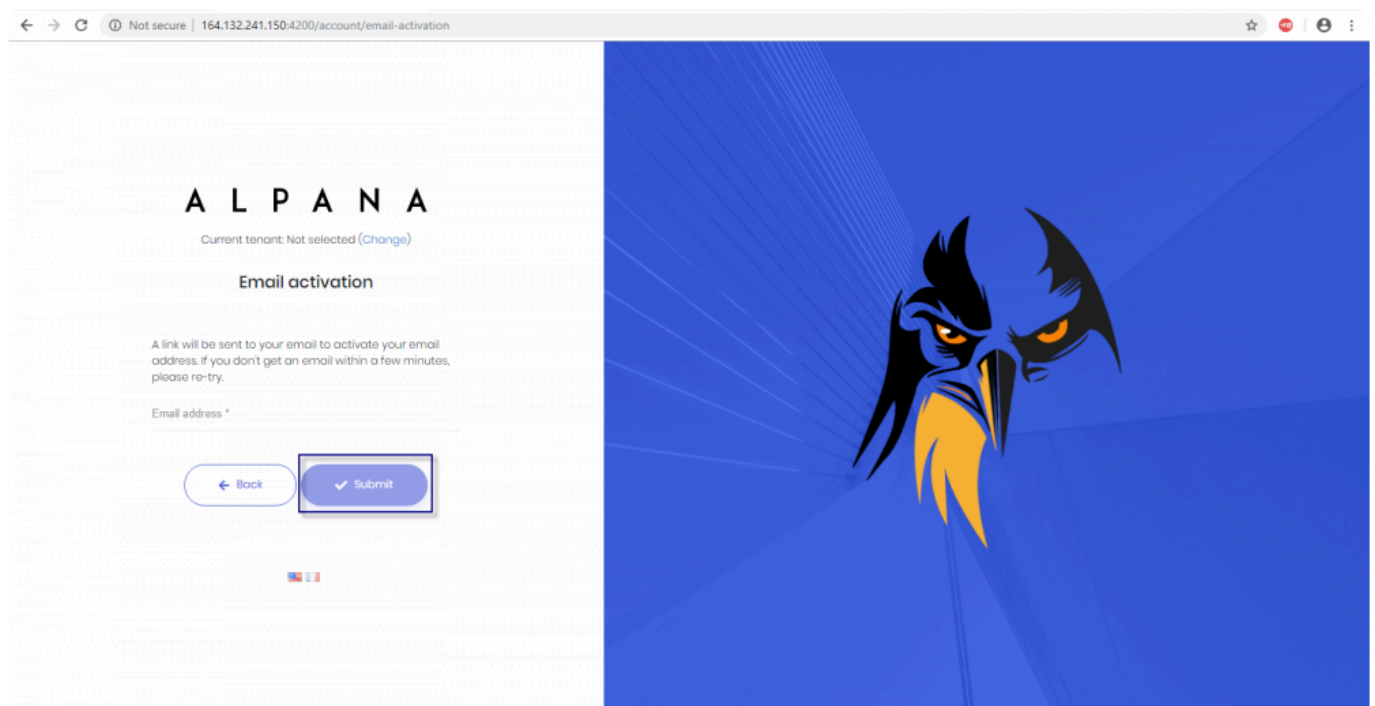
The image shows the Alpana application's login and registration page. At the top, the word 'ALPANA' is displayed in large, spaced-out letters. Below it, a message states 'Current tenant: Not selected (Change)'. A 'Log in' button is positioned below the message. The login section includes two text input fields: 'User name or email *' and 'Password *'. Below these fields are two links: 'Remember me' (preceded by an unchecked checkbox) and 'Forgot password?'. A blue 'Log in' button is located below the links. Below the 'Log in' button is a rectangular button labeled 'Email activation', which is highlighted with a red border. At the bottom of the page, there are three small social media icons.

- Enter a valid email address in text box



The image shows a web form for email activation. At the top, the word "ALPANA" is displayed in large, bold, black letters. Below it, the text "Current tenant: Not selected" is shown with a blue link "(Change)". The section is titled "Email activation". A message states: "A link will be sent to your email to activate your email address. If you don't get an email within a few minutes, please re-try." Below this is a text input field labeled "Email address *". A red error message "this field is required" is visible below the input field. At the bottom, there are two buttons: a blue "Back" button with a left arrow and a blue "Submit" button with a checkmark.

- Click on **Submit** Button



The image shows a web browser window displaying the email activation form. The browser's address bar shows "Not secure | 164.132.241.150:4200/account/email-activation". The form content is identical to the previous image, but it is now part of a larger page layout. On the right side of the page, there is a large blue banner featuring a stylized Alpana logo, which is a bird-like head with orange and yellow feathers and a black mask. The form is positioned on the left side of the page, and the "Submit" button is highlighted with a red border.

System will send activation link to the Email address. User needs to enter account details on “**Create Account**” form after opening the link received in the email . Once the user fills up the form , he is registered and can manage his account.

Last modified: 2019/05/28

3.6.3. Tenant Features

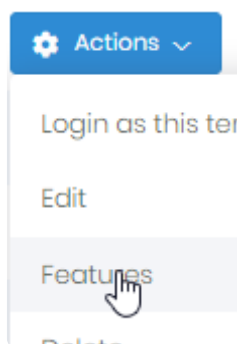
Depending on your license type, some features may be activated [at the Host level](#).

In addition, your license contains some limited resources (number of users, dashboards, etc)

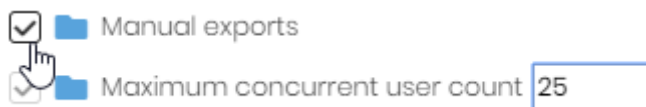
Then, in order to benefit from those features or resources, they must be allocated to each Tenant as desired :

On the Host, Navigate to *Tenants* management.

Then on the desired Tenant, select *Action > Features* :



In the *Features* window, check which Features to allocate, including resources, and click *Save* :

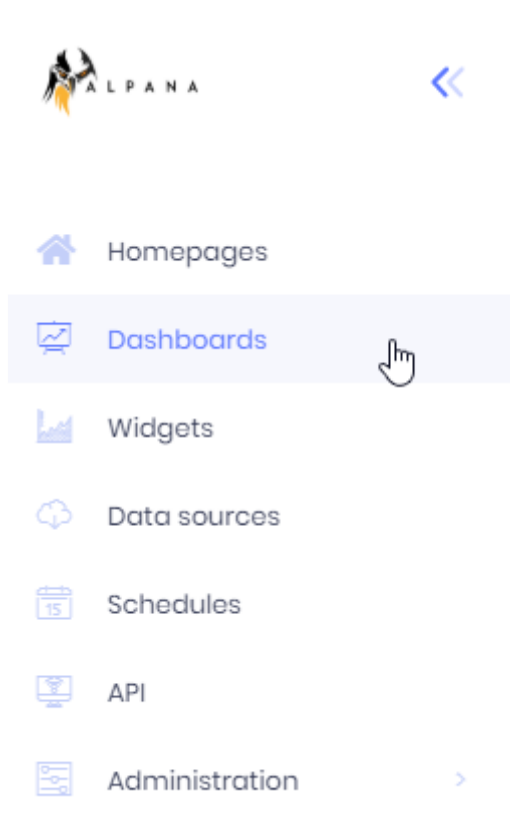


Last modified: 2019/06/18

3.7. Categories Management

Published Dashboards are organized in Categories.

To browse Dashboards and manage their Categories, use the *Dashboards* menu.



This Chapter documents the management of Categories.

Last modified: 2019/07/23

3.7.1. Creating Categories

* When you install Alpana Server or create a new Tenant, no Categories exist by default. You will need to create at least one Category for publishing.

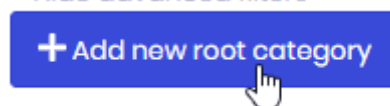
Categories are managed like a tree of folders.

There are two types of Categories :

1. *root Categories* are at the root of the tree structure.
2. *sub Categories* are nested under root Categories or other sub Categories

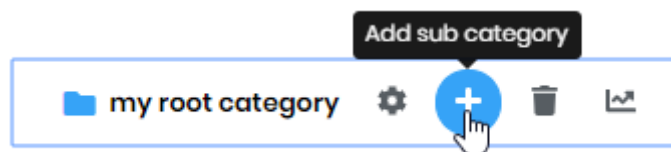
Creating a root Category

To create a root Category, click *Add new root Category* :



Creating a sub Category

To create a sub Category, click on the desired parent Category, then click the *Add sub Category* icon in from of it :



New Category menu

When creating a Category, fill in the desired name and description, and click *Save* :

New category

Name *

my category

Description

the description

Cancel

Save

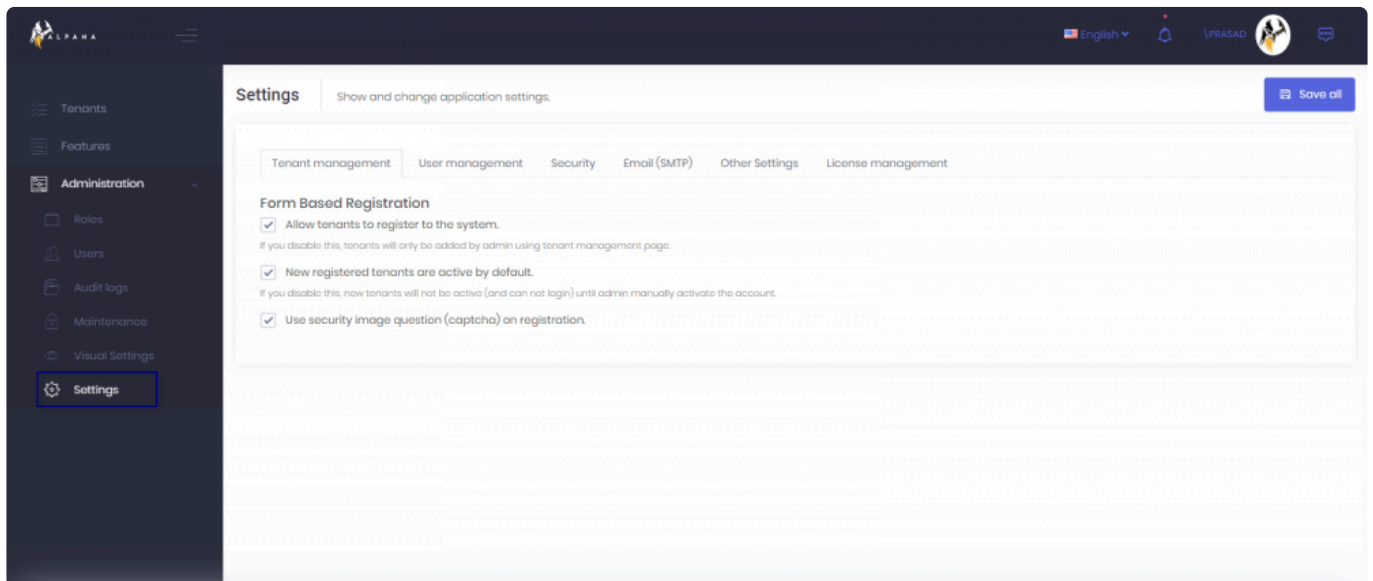
Last modified: 2019/07/23

3.8. Settings

Tenant Admin or the host admin is able to change the default settings.

Follow below steps for Default Settings Options

1. Click on “**Administration**” tab in the left navigation
2. Click on the “**Settings**” option



Last modified: 2019/05/28

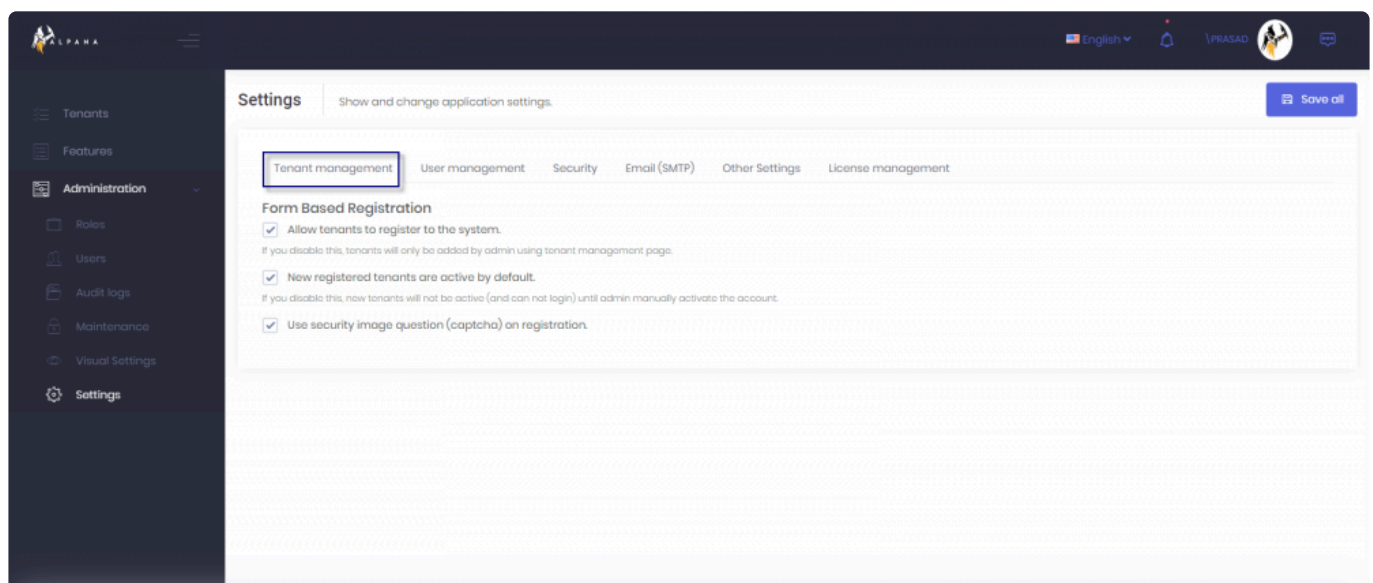
3.8.1. Settings-Tenant Management

As an organisation, you want to control the system, and Tenant management is very important as tenant is outside of the organisation. Let us understand the experience of tenant management.

Follow below steps to Manage a Tenant

Let us Manage the tenant

- Click on **Tenant Management** tab



- Below 3 options will be displayed to Configure/Manage the tenant. We just need to check/uncheck the boxes as per the requirement.
1. **Allow tenants to register to the system** – System will allow a tenant to get registered to the system by themselves
 2. **New registered tenants are active by default** – System will activate new registered tenants and they are able to login- If we uncheck then system will not activate new tenants(and tenant cannot login) until admin manually activates the account.

Last modified: 2019/06/21

3.8.2. Settings-User Management

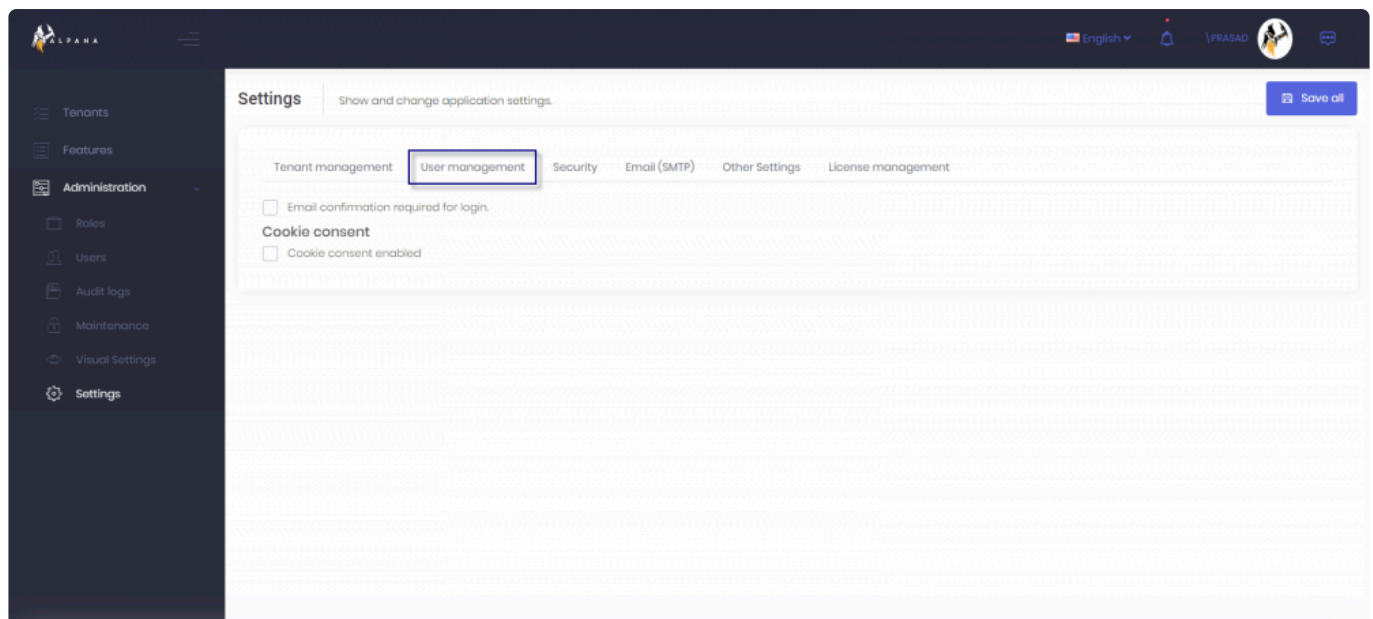
User Management settings are mostly related to login and cookie management .

Let us first understand the user management settings. System is given permission to set login verification and cookies management settings of the user to the admin . How does it work ?

1. **Email confirmation required for login** – User won't be able to login if they have never performed the activation procedure from e-mail.
2. **Cookie consent enabled** – The purpose of cookie consent is to alert the users of the website , about the cookies and get their consent for setting them up.

Follow below steps for user management settings

- Click on **User Management** tab



System will enable the settings If admin checked any of the option else it will be disabled

- Click on **Save all** button at top right corner

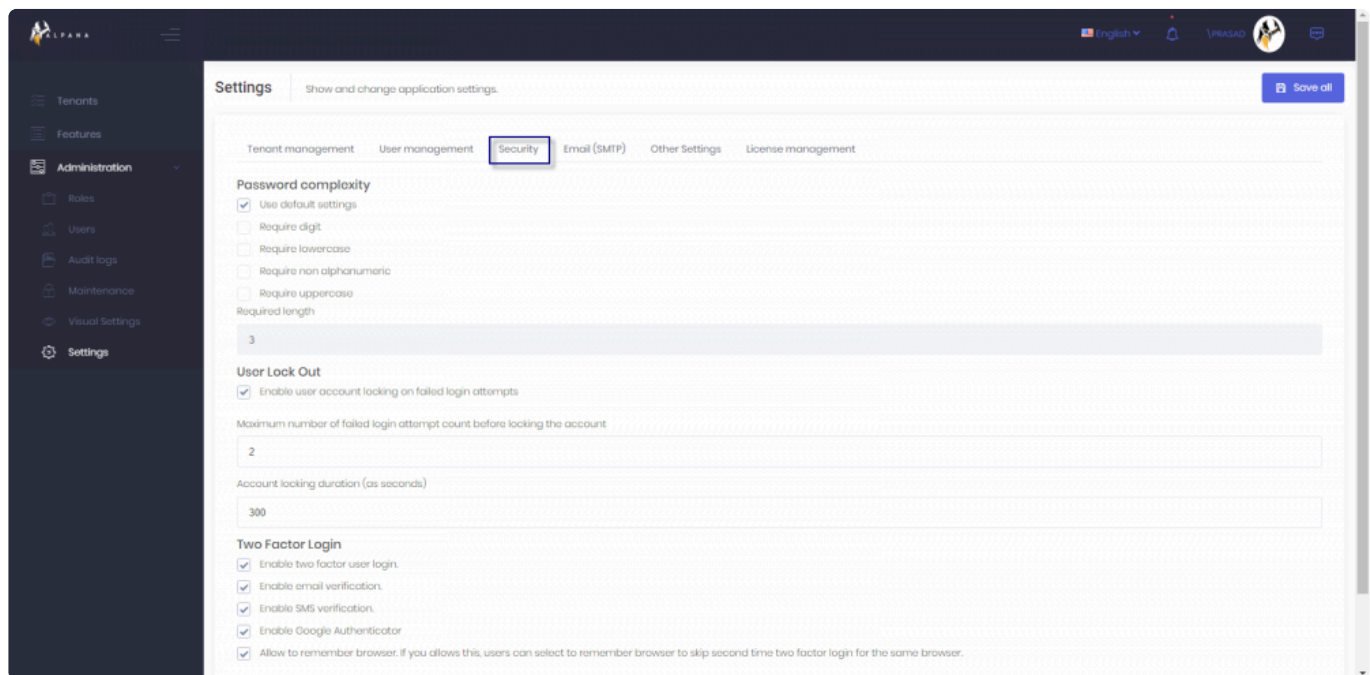
Last modified: 2019/07/02

3.8.3. Settings- Security

Two-factor (2FA) or multi-factor authentication (MFA) is an additional security layer for your business – helping to address the vulnerabilities of a standard password-only approach. Let us understand how these settings function.

Follow below steps for Security settings tab

- Click on **Settings** tab



- System will display below options for security settings. Let's understand them one by one.
1. **Password complexity** : This option will give option to admin to set password complexity for user while setting password. **for example** – Password should contain at least one number / special character etc.
 2. **User Lock Out** : as per these settings, user account gets locked after maximum number of login attempts have failed. Admin can set number for login attempts in the text box.
 3. **Two Factor Login**: Let us understand this as below :
 - a. Enable two factor user login : This option will send verification code for login either on email/ SMS/Google for login.
 - b. Enable Email Verification : This option will send email verification code to a registered email address
 - c. Enable SMS Verification: This option will send SMS on a registered mobile number
 - d. Enable Google Authentication: This option will send QR Code to your web app.

4. **Allow to remember browser** :If you allow this, users can select to remember browser to skip the second time two factor login for the same browser. : This option will allow user to remember password second time.

- Click on **Save all** button



Authentication code will be generated to send on email or SMS or google code after entering valid user name and password

Last modified: 2019/05/28

3.8.4. Setting-Email (SMTP)

Sending emails is a very common task for most applications. ASP.NET Boilerplate provides the basic infrastructure to send emails in a simple way. It also separates the email server configuration from the sending of emails

Follow below steps for Email (SMTP) settings

Let us set email settings

- Click on **Email (SMTP)** tab

The screenshot shows the 'Settings' page of the Alpana application. The 'Email (SMTP)' tab is selected and highlighted with a blue box. The settings are organized into sections: 'Default from (sender) email address' with the value 'admin@mydomain.com', 'Default from (sender) display name' with a 'Clear default' button, 'SMTP host' with the value 'smtp.gmail.com', 'SMTP port' with the value '587', 'Use SSL' checked, 'Use default credentials' unchecked, 'Domain name' with the value 'mydomain.com', 'User name' with the value 'admin@mydomain.com', and 'Password' with a masked input. At the bottom, there is a 'Test Email Settings' section with a 'Send Test Email' button.

In this Email (SMTP) tab you can set

- Set default **Sender Email address**
- Set default **Sender Name**
- set **SMTP host** and **SMTP port**

Last modified: 2019/05/28

3.8.5. Settings-Other Settings

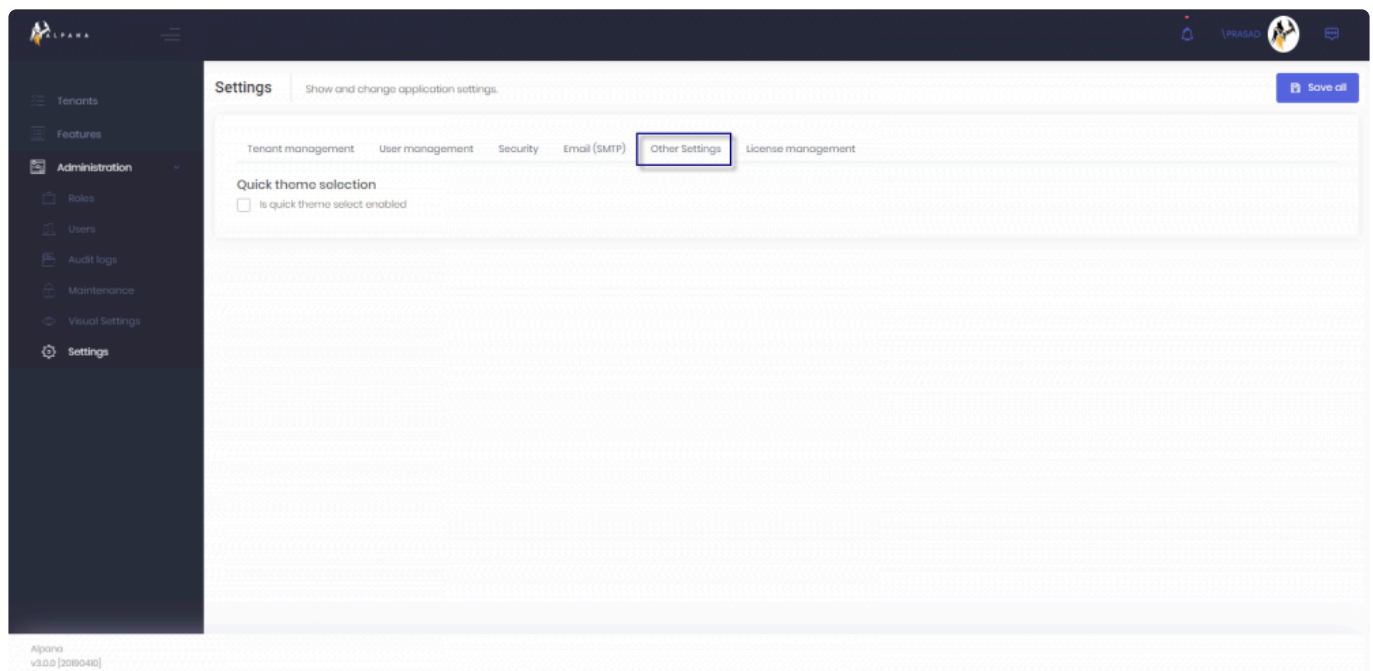
This will give an option to select the theme for the selected tenant. This will list down all the themes for selection in Visual settings.

User can select any of the theme and apply for any of the tenant.

Follow below steps for Other settings

Let us set the visual theme

- Click on Other Settings tab

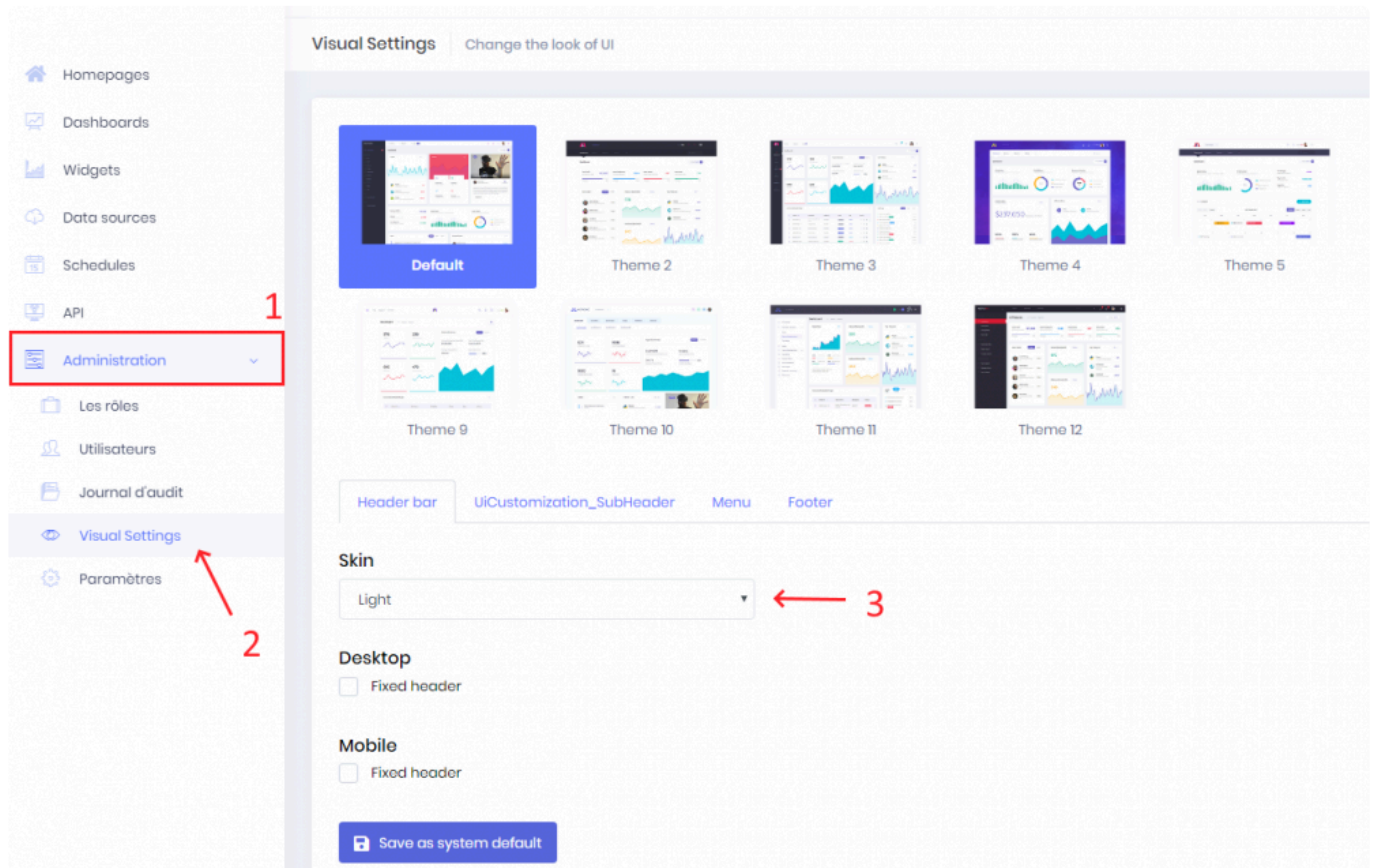


- Select the theme to apply for a tenant

Last modified: 2019/05/28

3.8.6. Settings-Appearence (Use Custom CSS)

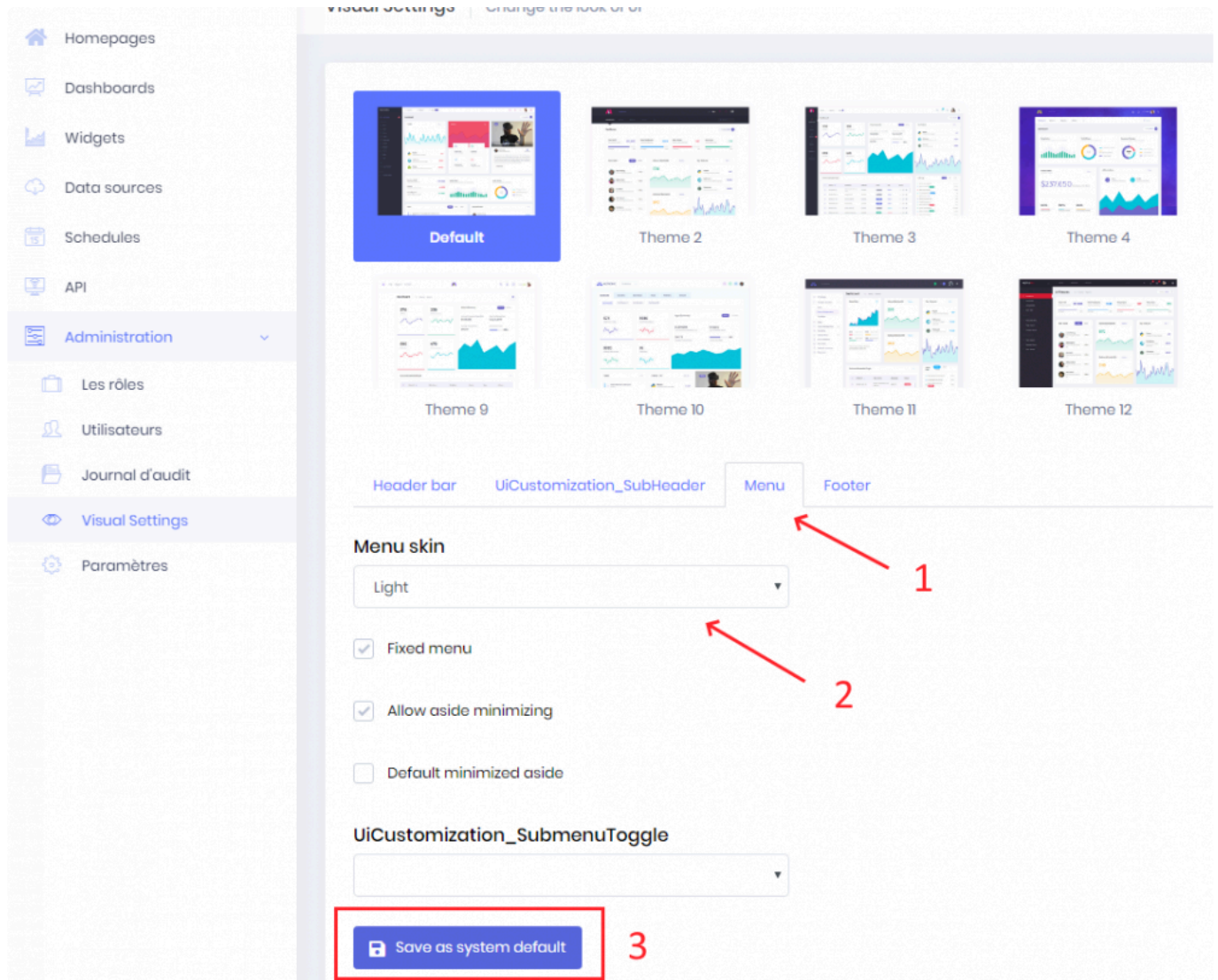
To apply your Custom CSS theme to Alpana Server please follow the below steps:



First, you need to go to the Visual Settings Page by clicking on Administration (Step 1 on screenshot), then click to Visual Settings (Step 2 on the screenshot).

Then, Select the Light Skin in the Skin Combobox. (Step 3 on the screenshot)

Once done, click on the menu tab like the step 1 on the following screenshot:



Then, select Light in the Menu Skin Combobox like Step 2 on the screenshot.

Once done, please ensure the DEFAULT theme is selected like on the screenshot and click on the “Save as system default” button which is Step 3 on the screenshot.

After this; you will need to prepare your custom CSS file.

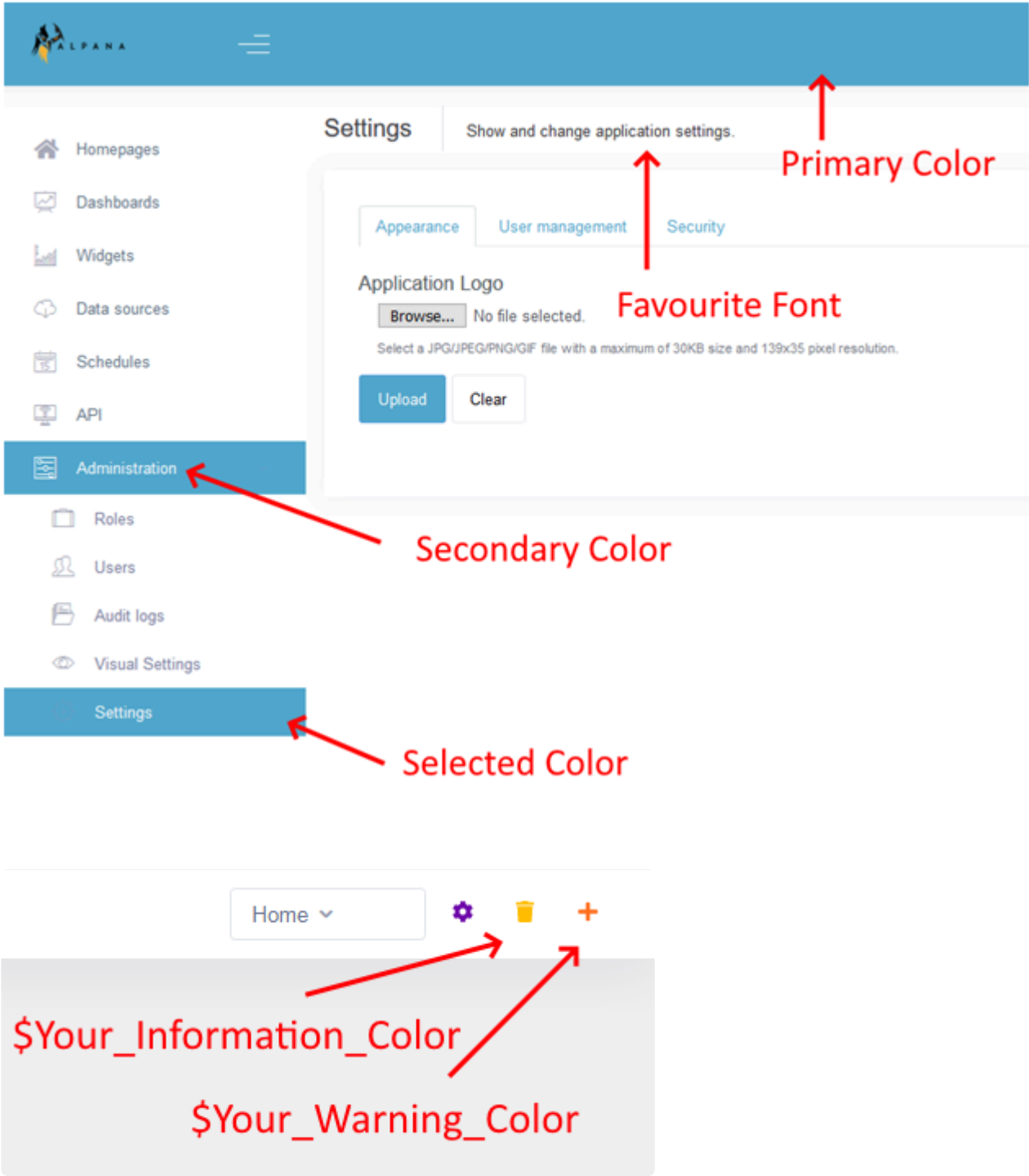
To help you with this step, we prepared a template file, please download the following file: “Alpana Server Custom CSS Placeholders File: <https://cdn.manula.com/user/15037/docs/alpana-custom-css-file-placeholders-file-3-1.css>

Then, you need to edit this file with your favorite text editor by changing the following placeholders by your custom values:

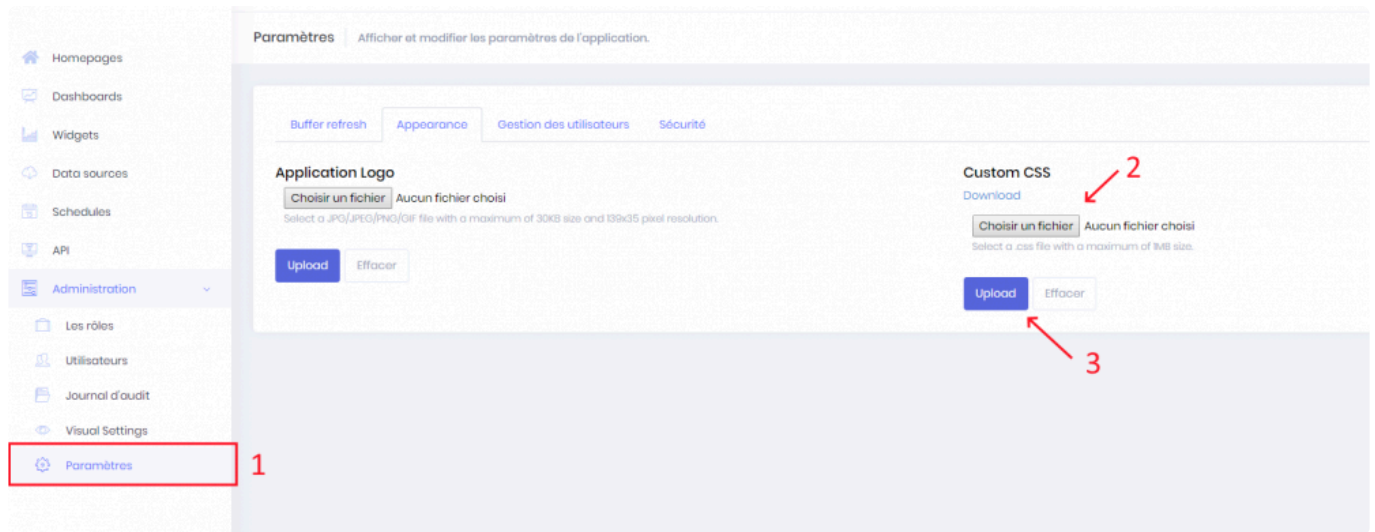
\$Your_Primary_Color
\$Your_Secondary_Color

\$Your_Selected_Item_Color
\$Your_favourite_font
\$Your_Information_Color
\$Your_Warning_Color
\$Your_Error_Color

Please take a look at the placeholders description in the following screenshots:



When it's done, you need to go to the Parameter Section shown in the screenshot below by clicking on Parameter (Step 1)



Then click on the Browse button (Step 2). From there, you need to select on your computer the previous edited file with your custom values and click “open” from the dialog box.

Finally click on the Upload button (Step 3) and you should have your selected theme applied.

Last modified: 2019/08/06

3.8.7. API Origins Whitelist

Why

Querying or embedding a content from Alpana into another website requires to be allowed from Alpana. This is called [Cross-origin resource sharing](#) , see also [Same-origin policy](#).

If you fail to do so, you won't be able to query or embed Alpana content from your application, and you will get a typical CORS error like :

```
Access to XMLHttpRequest at 'http://xxx.xxx.xxx.xxx:xxx' from origin 'http://yy.yyy.yyy.yyy:yyy' has been blocked by CORS policy: Response to preflight request doesn't pass access control check: No 'Access-Control-Allow-Origin' header is present on the requested resource.
```

This is a normal thing, for security purposes.

For example, the <https://www.google.com/> URL currently refuses to be embedded inside an iframe of another website.

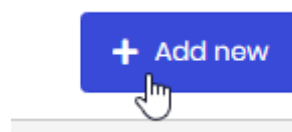
In Alpana, you can create an exception, to allow a specific “foreign website” to access Alpana content.

✿ If you are making queries from Javascript code in a local HTML file open in your browser, you won't be able to make a query and your browser will probably block you. If you really just want to test that your code works, then your browser may let you disable security for this.
Check your browser documentation or [internet](#).

Chrome.exe —disable-web-security

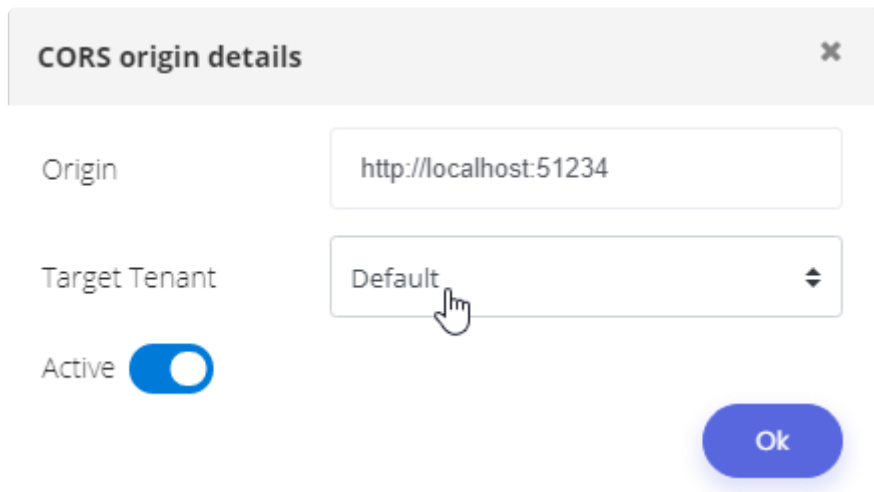
Creating an exception

On the Host of the Alpana Server; go under *Administration > Settings > API Origins Whitelist* and click *Add New* :



Configuration

In the configuration window, fill in the details :



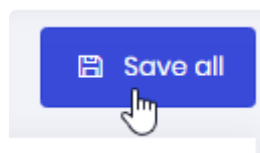
The screenshot shows a dialog box titled "CORS origin details". It contains the following fields:

- Origin**: A text input field containing "http://localhost:51234".
- Target Tenant**: A dropdown menu with "Default" selected.
- Active**: A toggle switch that is currently turned on (blue).
- Ok**: A blue button at the bottom right.

- **Origin** : this is the URL of your application. The format is (protocol) : // (site) : (port) with no trailing slash /. Examples : https://www.google.com:80, http://123.456.789:81, http://localhost:1234
- **Target Tenant** : the tenant that will accept the queries or embedding
- **Active** : the exception is Active or not

Click OK.

At the top of *Administration > Settings > API Origins Whitelist*, don't forget to click *Save All* :



Last modified: 2019/07/05

3.9. Manage Users and Permission

The User entity represents a user of the application. It should be derived from the AbpUser class .

The AbpUser class defines some base properties like :

1. **UserName**: Login name of the user. Should be unique for a tenant.
2. **EmailAddress**: Email address of the user. Should be unique for a tenant.
3. **Password**: Hashed password of the user.
4. **IsActive**: True, if this user can login to the application.
5. **Name** and **Surname** of the user.

AbpUser class also contains properties like **Roles**, **Permissions**, **Tenant**, **Settings**, **IsEmailConfirmed** etc. Check the AbpUser class for more information.

Last modified: 2019/05/28

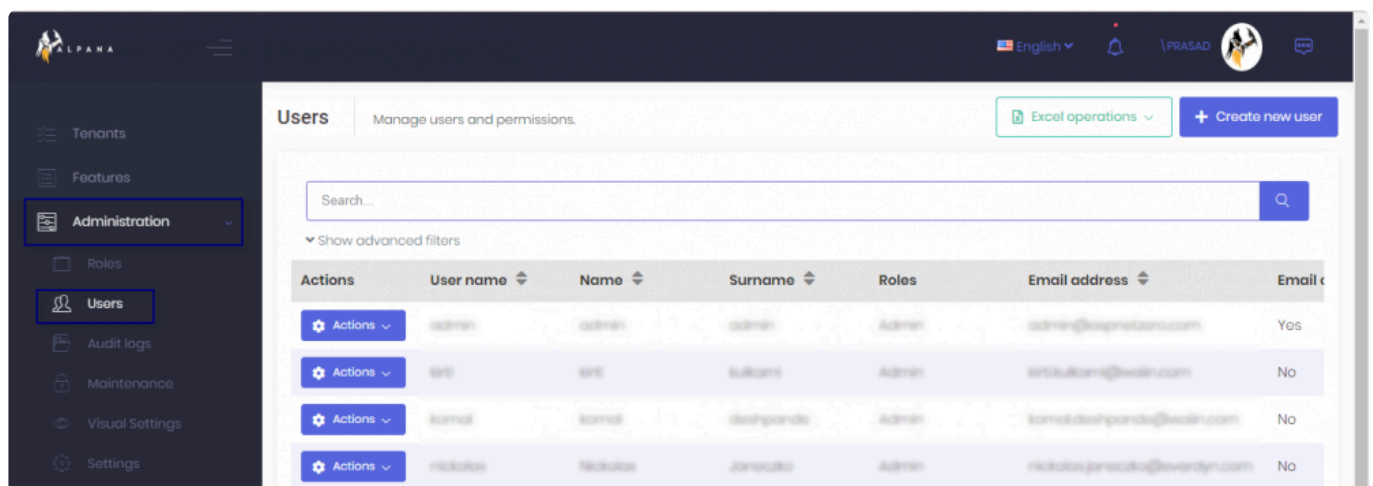
3.9.1. Create User

After login into the System as admin. it's time to add valid users who can access the application and perform their task as per role assigned.

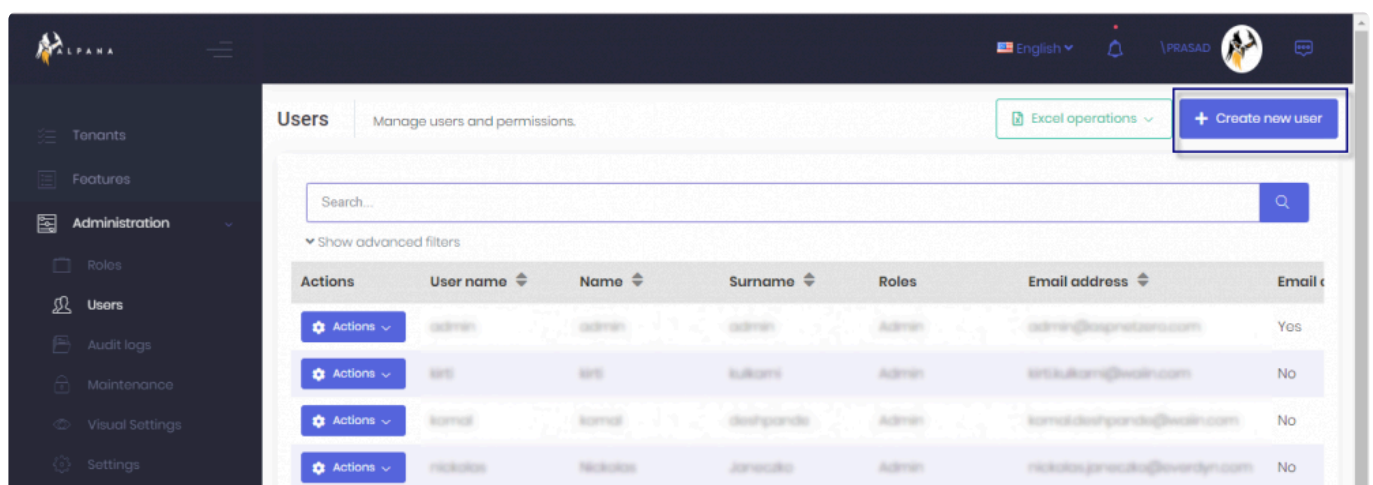
Create a new user

Now let's add a user:

- Click on “**Administration**” tab in left navigation
- Click on “**User**” option



- Click on **Create New User**




- Fill the create user form on the pop up

Create new user

User informations

Roles1



Name *

This field is required.

Surname *

Email address *

This field is required.

Phone number

User name *

☒ Set random password.

☒ Should change password on next login.

☒ Send activation email.

☒ Active

☒ Two factor authentication enabled


☒ Lockout enabled

Cancel

Save

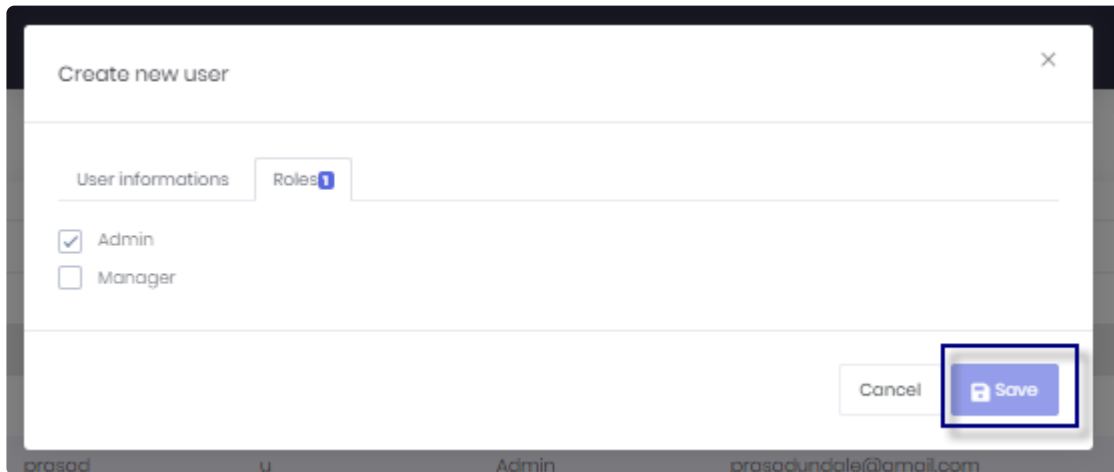
- Click on the “**Roles**” tab on Create User pop up and select the Role for user

Page 696 of 761



The screenshot shows a 'Create new user' dialog box. It has a title bar with a close button (X). Below the title bar, there are two tabs: 'User informations' and 'Roles'. The 'Roles' tab is selected and highlighted with a blue box. Under the 'Roles' tab, there are two checkboxes: 'Admin' (checked) and 'Manager' (unchecked). At the bottom right of the dialog, there are two buttons: 'Cancel' and 'Save'. The 'Save' button is highlighted with a blue box. The dialog box is set against a background that shows a user interface with a header bar containing the text 'prasad', 'u', 'Admin', and 'prasadundale@gmail.com'.

- Click on the **“Save”** button on the pop up



This screenshot is identical to the one above, showing the 'Create new user' dialog box with the 'Roles' tab selected and the 'Save' button highlighted.

User will receive the login details on a registered email address with account activation link. The user needs to click on the activation link and then enter login credentials.

That is how you can create a new user and access the application

✿ User needs to contact the administrator , in case of any difficulty in login.

Last modified: 2019/05/28

3.9.2. Edit User

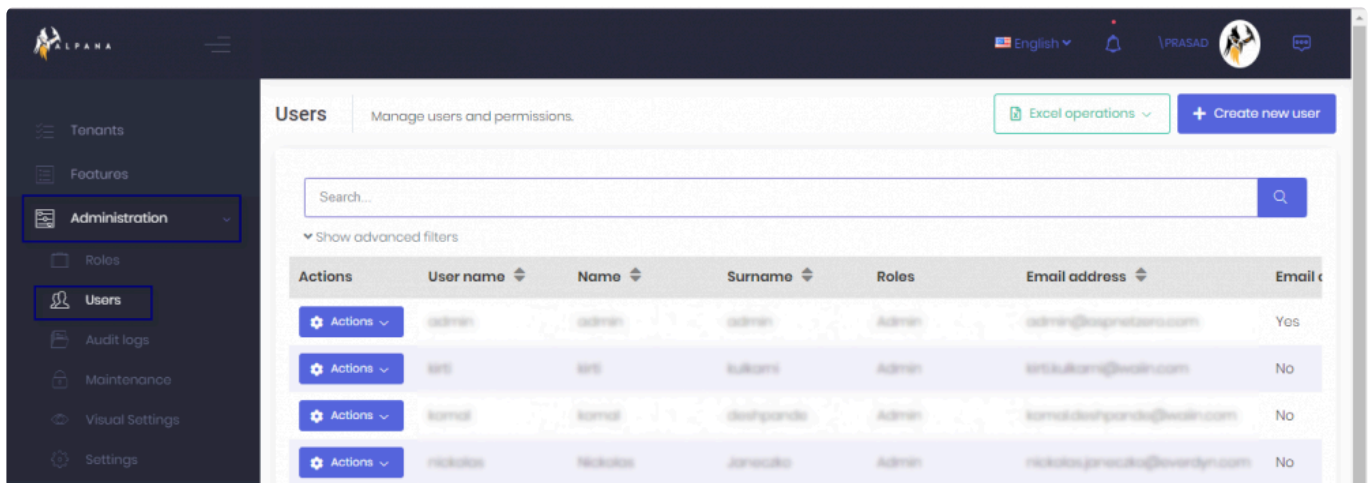
There are number of reasons for the admin to edit user details. Below are a few reasons:

1. Update the personal details
2. Set new password
3. Activate or Deactivate the user
4. Update security related to login

Follow Below steps to edit a user

Let's Edit a user

- Click on “**Administration**” tab in left navigation
- Click on “**User**” option



The screenshot shows the 'Users' management page in the Alpana application. The left sidebar has 'Administration' and 'Users' tabs. The main content area has a search bar and a table of users. The table has columns: Actions, User name, Name, Surname, Roles, Email address, and Email status. There are four users listed: admin, kirti, karnal, and nikolas. Each user has an 'Actions' button next to it.

Actions	User name	Name	Surname	Roles	Email address	Email status
Actions	admin	admin	admin	Admin	admin@alpana.com	Yes
Actions	kirti	kirti	kulkarni	Admin	kirtikulkarni@alpana.com	No
Actions	karnal	karnal	dasgupta	Admin	karnaldasgupta@alpana.com	No
Actions	nikolas	Nikolas	Janesukla	Admin	nikolas.janesukla@alpana.com	No

- Click on the “**Action**” button on Manage Users and permission page
- Click on “**Edit**” button

Users

Manage users and permissions.

Excel operations

Create new user

Search...

Show advanced filters


Actions	User name	Name	Surname	Roles	Email address	Email c
Actions	Admin	...	Yes
Actions	Admin	...	No
Login as this user	Admin	...	No
Edit	Admin	...	Yes
Permissions	Admin	...	No
Unlock	Admin	...	No
Delete	Admin	...	No
Actions	Admin	...	No
Actions	Admin	...	Yes

- Update required details on the **Edit User** pop up

Edit user: prasadu

User informations

Roles1



Name *

Surname *

u

Email address *

Phone number

User name *

☐ Set random password.

Password

Password (repeat)

☐ Should change password on next login.

☐ Send activation email.

☒ Active

☐ Two factor authentication enabled

☒ Lockout enabled

Cancel

Save

- Click on the “**Roles**” tab and change the role (This is optional step)



Dialog box titled "Edit user: prasadu" with a close button (X) in the top right corner.

Section: User informations

Roles button (highlighted with a blue box)

Checkboxes:

- ☒ Admin
- ☐ Manager


Buttons: Cancel, Save

- Click on “**Save**” button on the pop up to save details

Edit user: prasadu

User informations

Roles1



Name *

Surname *

Email address *

Phone number

User name *

☐ Set random password.

Password

Password (repeat)

☐ Should change password on next login.

☐ Send activation email.

☒ Active

☐ Two factor authentication enabled

☒ Lockout enabled

Cancel

Save



System will send an email to the user only if user password is updated. For other updates, there is no system generated notification.



User Name can not be updated

Last modified: 2019/05/28

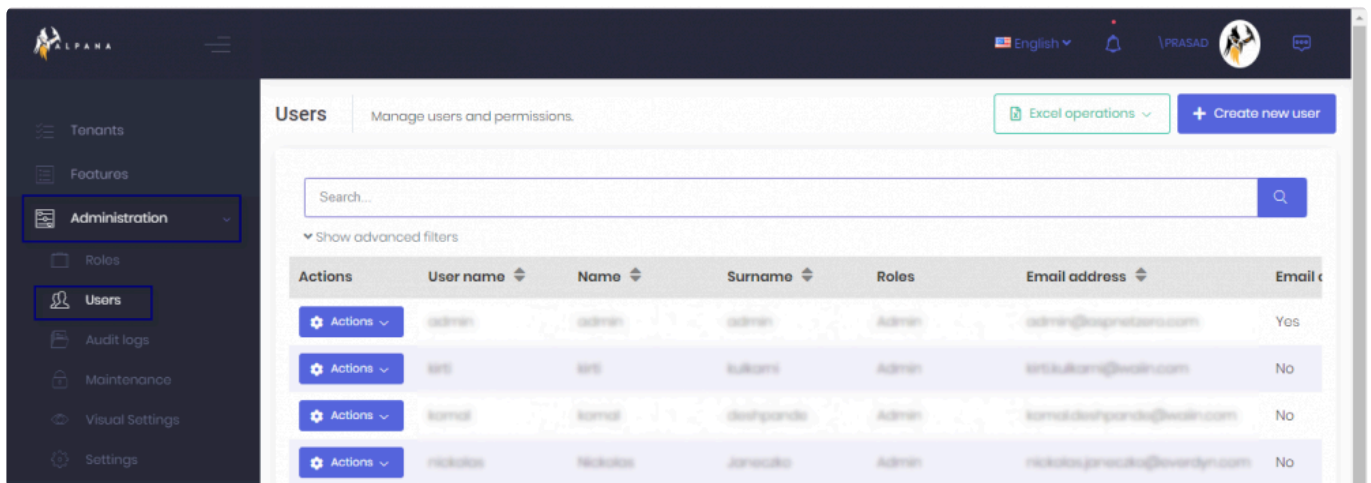
3.9.3. Delete User

Deleting a user is the last thing that an admin can do, if he wants to remove the system access of the user. The admin has the option to deactivate a user . However, in case, a particular user has left the organisation or does not belong to the application, the admin can delete the user permanently.

Follow Below steps to delete user

Let's delete the user

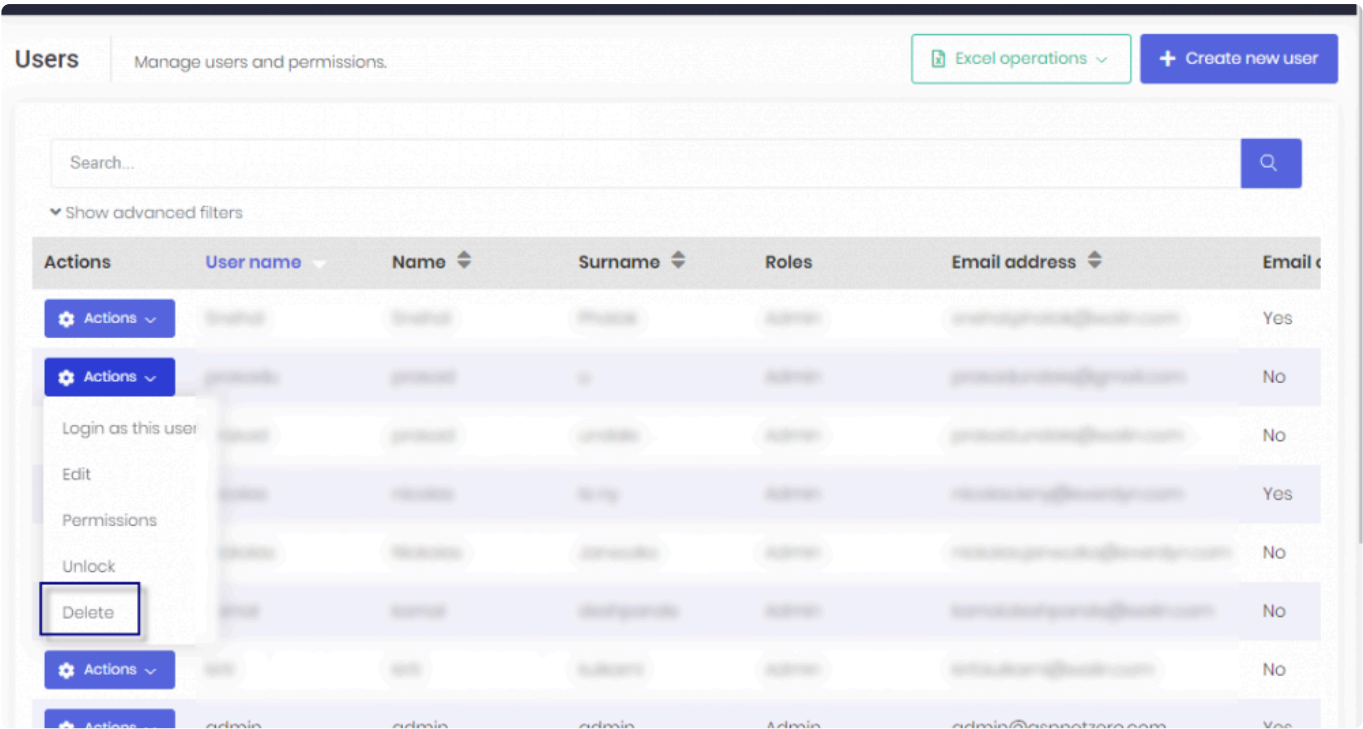
- Click on “**Administration**” tab in left navigation
- Click on “**User**” option



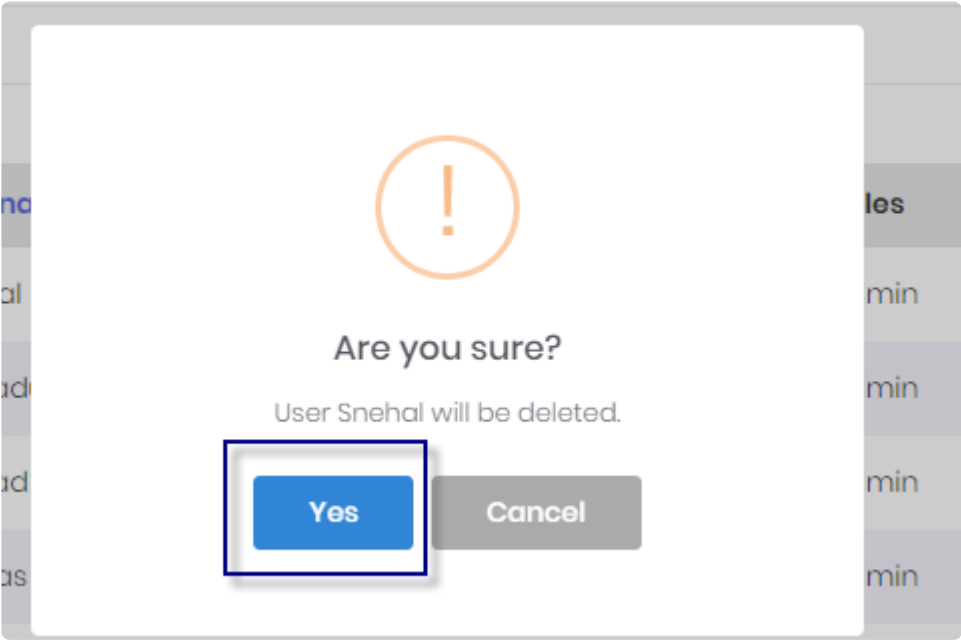
The screenshot shows the 'Users' management page in the Alpana application. The left sidebar has 'Administration' and 'Users' highlighted. The main content area shows a table of users with the following data:

Actions	User name	Name	Surname	Roles	Email address	Email status
Actions	admin	admin	admin	Admin	admin@alpana.com	Yes
Actions	test	test	kulkarni	Admin	testkulkarni@alpana.com	No
Actions	karnal	karnal	deshpande	Admin	karnal.deshpande@alpana.com	No
Actions	nikolas	Nikolas	Janecek	Admin	nikolas.janecek@alpana.com	No

- Click on the “**Action**” button on Manage Users and permission page
- Click on “**Delete**” button



- After click on “Delete” button, confirmation pop will displayed
- Click on “Yes” button on confirmation pop up
- System shall delete the user



✿ Default admin (registered user while creating tenant) of the application cannot be deleted



There is no direct mechanism to revert back deleted user.

Last modified: 2019/05/28

3.9.4. Excel Operations

Last modified: 2019/05/28

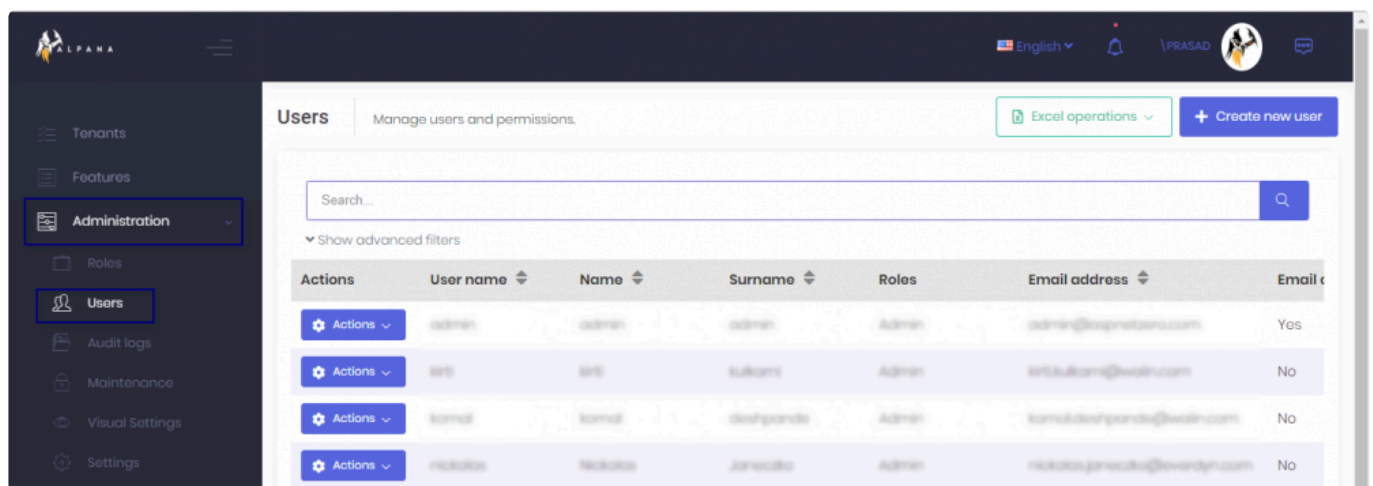
3.9.4.1. Export users details

There is always a need to keep offline backup of system data. Particularly, User data is very crucial to an application. The application has a very easy-to-use feature to back up user data i.e. exporting data to excel with a single click.

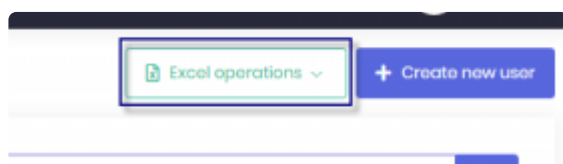
Follow the steps below to create user

Let's export the user data

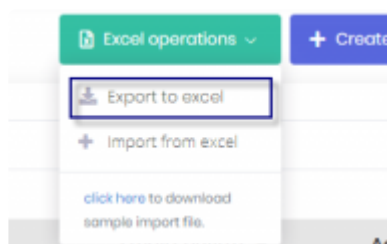
- Click on “**Administration**” tab in left navigation
- Click on “**User**” option



- Click on **Excel Operations**



- Click on **Export to excel**



System will download the excel file with below user information

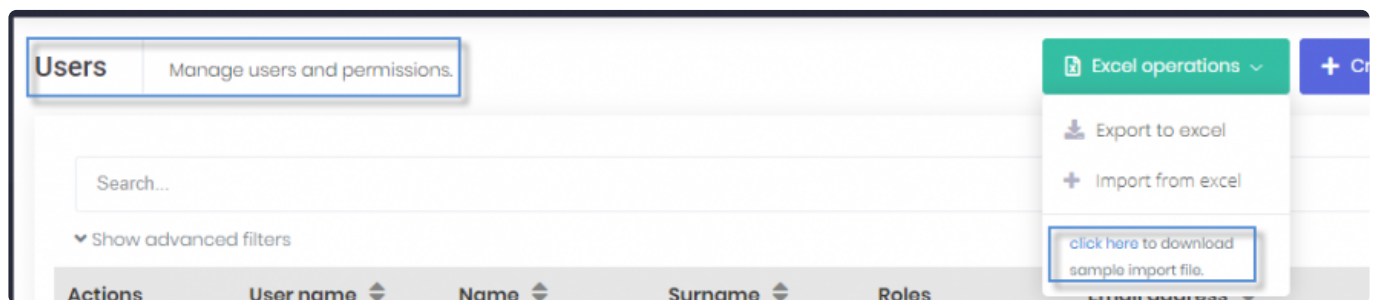
1. **Name**
2. **Surname**
3. **User Name**
4. **Phone Number**
5. **Email Address**
6. **Email Confirm**
7. **Roles**
8. **Active**
9. **Creation Time**

Last modified: 2019/05/28

3.9.4.2. Import User Details

it is obvious that, a system is accessible to multiple users, and users need to be added into system all the time. But, it is very difficult to add users one by one or migrating from any other system if user count is high. For this reason, the application has provided a performance driven feature to upload large number of users in one go by importing an excel file.

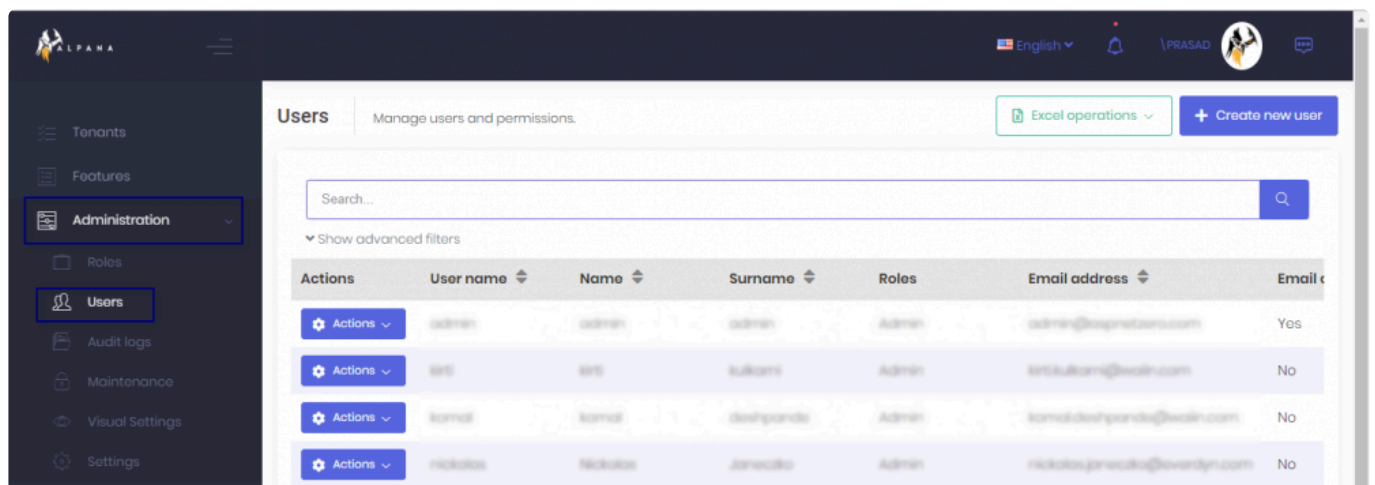
First, the admin needs to download a sample excel file from **Excel operations** option on **Manage Users and permissions** page and accordingly prepare the user list.



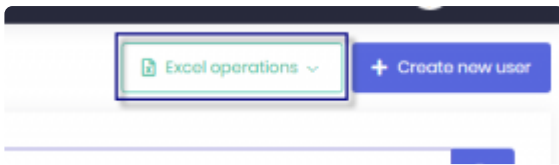
Follow below steps to import users

Let's import the users

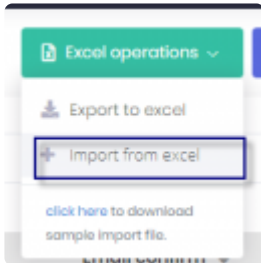
1. Click on **"Administration"** tab in left navigation
2. Click on **"User"** option



3. Click on **Excel Operations**



4. Click on **Import to Excel** and upload the excel file from local drive



System allows to import the following user details

1. **User Name**
2. **Name**
3. **Surname**
4. **Email Address**
5. **Phone Number**
6. **Password**
7. **Assigned Role Name** – This is optional. Role management can be done later from application itself.

✿ All the imported users are active by default.

Last modified: 2019/05/28

3.10. Manage Roles

Roles are stored in the `AbpRoles` table in the database. You can add your own custom properties to the `Role` class (and create database migrations for the changes).

`AbpRole` defines some default properties. The most important are:

Name: Unique name of the role in the tenant.

DisplayName: Shown name of the role.

IsDefault: Is this role assigned to new users by default?

IsStatic: Is this role static? (setup during pre-build, and can not be deleted).

Roles are used to group permissions. When a user has a role, then he/she will have all the permissions of that role. A user can have multiple roles. The permissions of this user will be a merger of all the permissions of all assigned roles.

Last modified: 2019/05/28

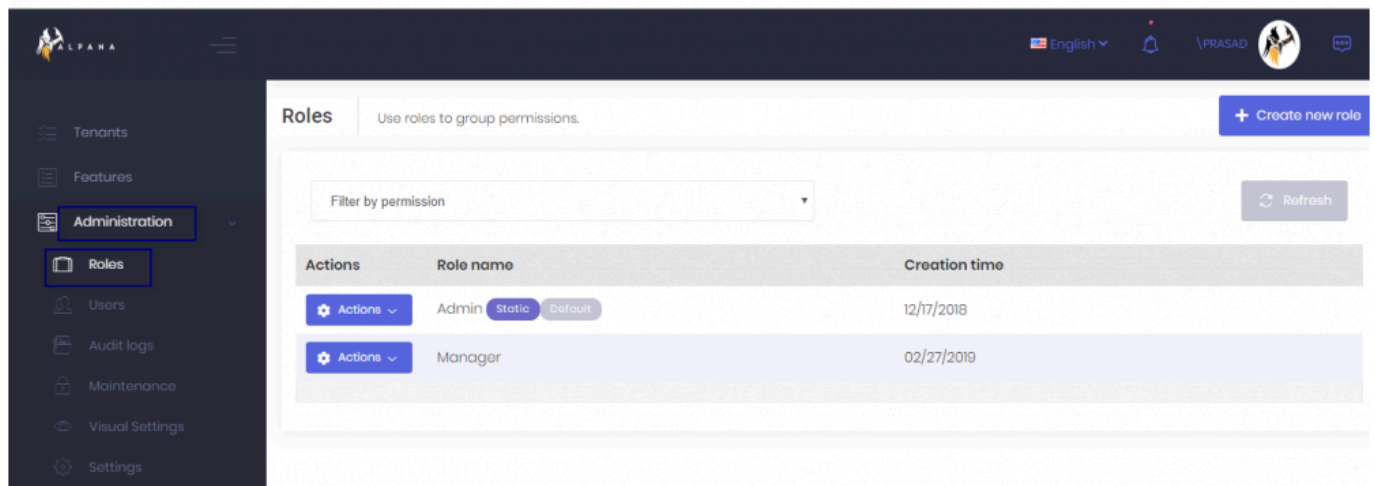
3.10.1. Create Role

In any organisation there are different responsibilities and roles carried out by individuals or group. Also, as a security measure, system should not expose all the features to all users. An application admin can create a role and assign different application responsibilities/features to that role. This way, the user can access specific features and get permissions as per roles.

Follow below steps to create a Role

Let's create a role

- Click on “**Administration**” tab in left navigation
- Click on “**Roles**” option



- Click on “**Create New Role**” button on **Manage Roles** page to add custom role
- Enter Role Name on **Create New Role** pop up
- If “**Default**” option is selected then, role will be assign to to the new user by default

Create new role

Role name

Permissions

Role name *

This field is required.

☐ Default

Assign to new users as default.

If you are changing your own permissions, you may need to refresh page (F5) to take effect of permission changes on your own screen!

Cancel

Save

- Click on the **Permissions** tab and select the permissions needed to assign to the Role.

Create new role

Role name

Permissions

Search...

☒

Pages

☒

Administration

☐ Audit logs

☐ Maintenance

☒ Roles

☒ Creating new role

☒ Deleting role

☒ Editing role

☐ Settings

☐ Users

☐ Changing permissions

☐ Creating new user

☐ Deleting user

☐ Editing user

☐ Login for users

☐ Visual Settings

☐ Tenants

☐ Changing features

☐ Creating new tenant

☐ Deleting tenant

☐ Editing tenant

☐ Login for tenants

If you are changing your own permissions, you may need to refresh page (F5) to take effect of permission changes on your own screen!

Cancel

Save

- Click on **Save** button

Page 714 of 761

Create new role

Role name Permissions

Search...

- ☒ Pages
 - ☒ Administration
 - ☐ Audit logs
 - ☐ Maintenance
 - ☒ Roles
 - ☒ Creating new role
 - ☒ Deleting role
 - ☒ Editing role
 - ☐ Settings
 - ☐ Users
 - ☐ Changing permissions
 - ☐ Creating new user
 - ☐ Deleting user
 - ☐ Editing user
 - ☐ Login for users
 - ☐ Visual Settings
 - ☐ Tenants
 - ☐ Changing features
 - ☐ Creating new tenant
 - ☐ Deleting tenant
 - ☐ Editing tenant
 - ☐ Login for tenants

If you are changing your own permissions, you may need to refresh page (F5) to take effect of permission changes on your own screen!

Cancel Save

That's it. Role is created !!

✿ Admin Role is created by Default

✿ All the permissions assign to the role shall be granted to the user when role is assign to the user

Last modified: 2019/05/28

3.10.2. Edit Roles

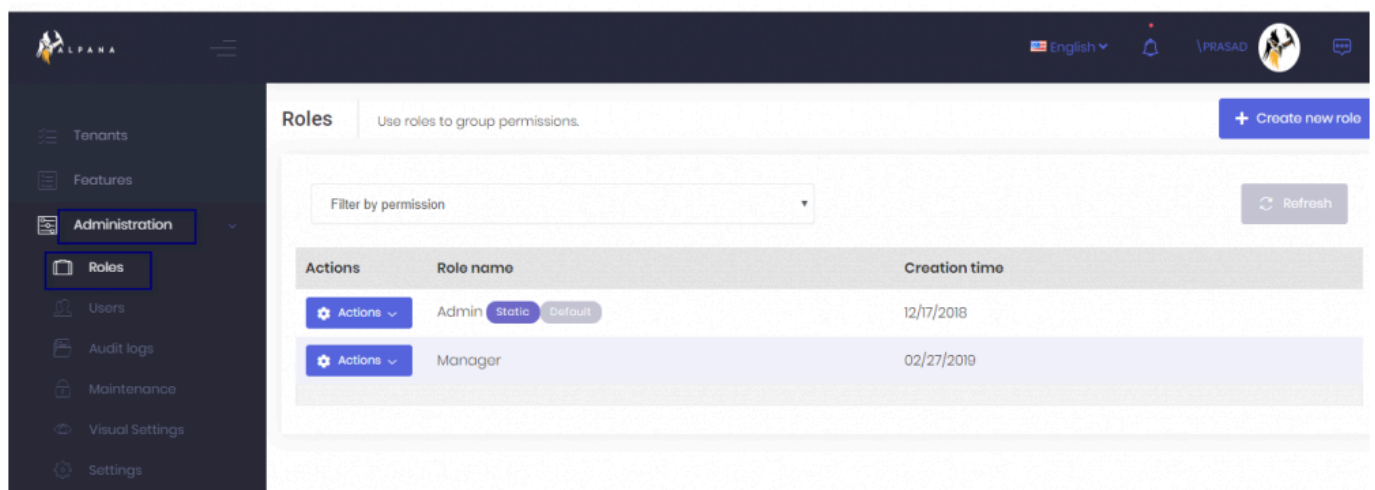
There are a number of reasons where the admin might want to edit role details. Below are a few reasons:

1. To change **Role Name** or **Description**
2. To change default status to assigned role to user by default or not
3. To change the permission for the roles

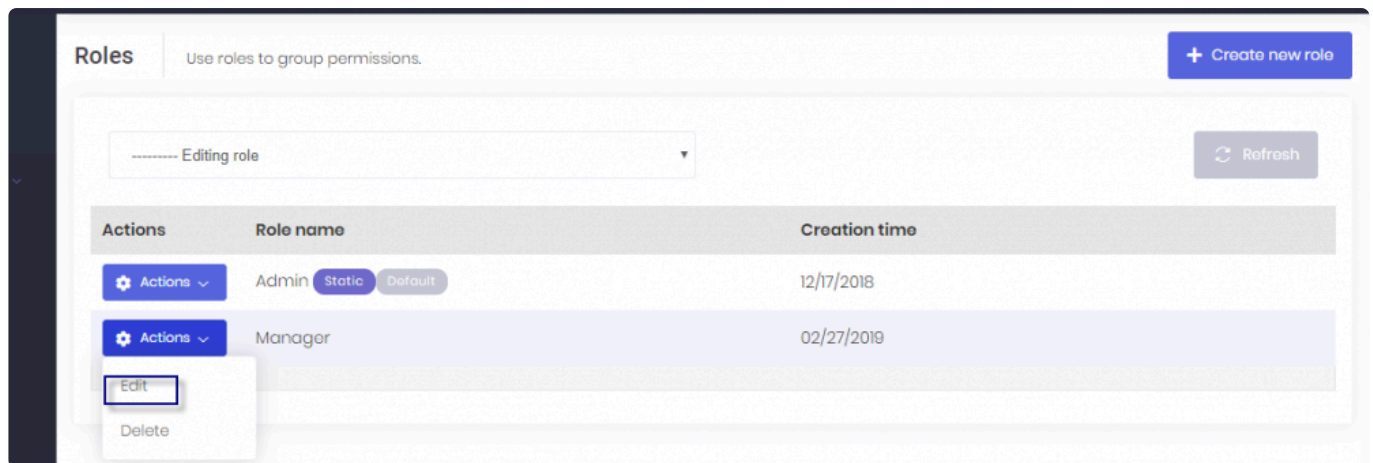
Follow below steps to edit a role

Let's Edit a Role

- Click on “**Administration**” tab in left navigation
- Click on “**Roles**” option



- Click on the “**Action**” button on **Manage Roles** page
- Click on “**Edit**” option in the **Actions** drop-down



- Admin can update Role Name and Default status on **Roles** tab

Edit role: Manager

Role name

Permissions

Role name *

Manager

☐ Default

Assign to new users as default.

If you are changing your own permissions, you may need to refresh page (F5) to take effect of permission changes on your own screen!

Cancel

Save

- Admin can change the permission in **Permissions** tab by selecting the check boxes on the next page

Edit role: Manager

Role name

Permissions

Search...

☒

Pages

☒

Administration

☐ Audit logs

☐ Maintenance

☐

Roles

☐ Creating new role

☐ Deleting role

☐ Editing role

☐ Settings

☒

Users

☒ Changing permissions

☒ Creating new user

☐ Deleting user

☒ Editing user

☒ Login for users

☐ Visual Settings

☐

Tenants

☐ Changing features

☐ Creating new tenant

☐ Deleting tenant

☐ Editing tenant

☐ Login for tenants

If you are changing your own permissions, you may need to refresh page (F5) to take effect of permission changes on your own screen!

Cancel

Save

- Click on **Save** button

Edit role: Manager

Role name

Permissions

Search...

☒ Pages

☒ Administration

☐ Audit logs
 ☐ Maintenance

☐ Roles

☐ Creating new role
 ☐ Deleting role
 ☐ Editing role

☐ Settings

☒ Users

☒ Changing permissions
 ☒ Creating new user
 ☐ Deleting user
 ☒ Editing user
 ☒ Login for users

☐ Visual Settings

☐ Tenants

☐ Changing features
 ☐ Creating new tenant
 ☐ Deleting tenant
 ☐ Editing tenant
 ☐ Login for tenants

If you are changing your own permissions, you may need to refresh page (F5) to take effect of permission changes on your own screen!

Cancel

Save



System will update the permission and grant access to the application , as per updated permission to the user, to whom the updated role is assigned.

Last modified: 2019/05/28

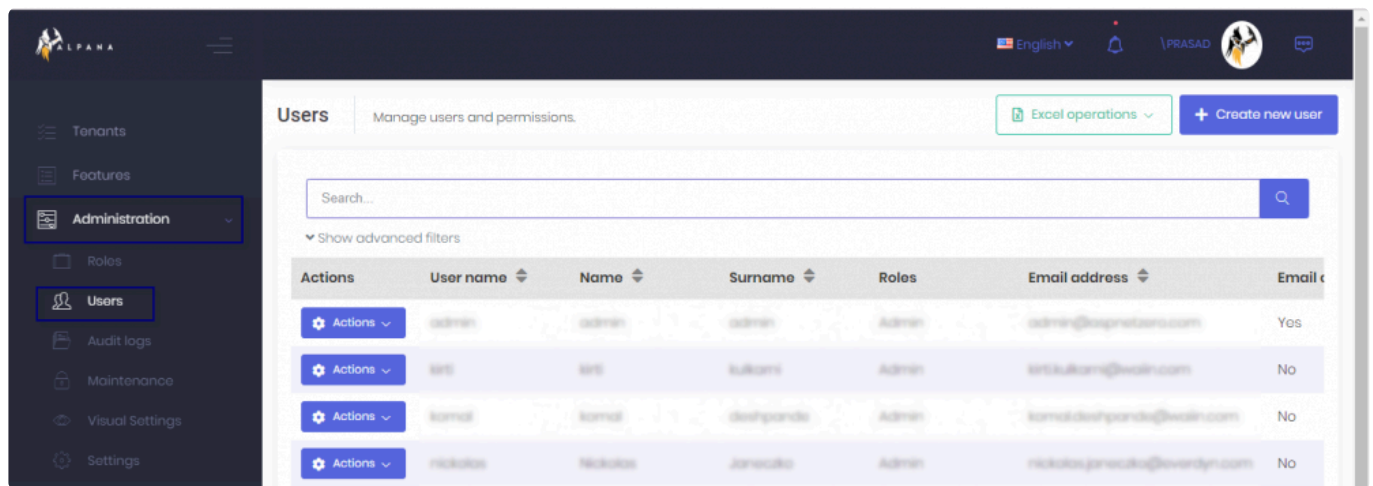
3.10.3. Delete Role

Deleting role is one of the feature provided by the application. So, we can delete the role(s) which are not relevant or got created by any human error

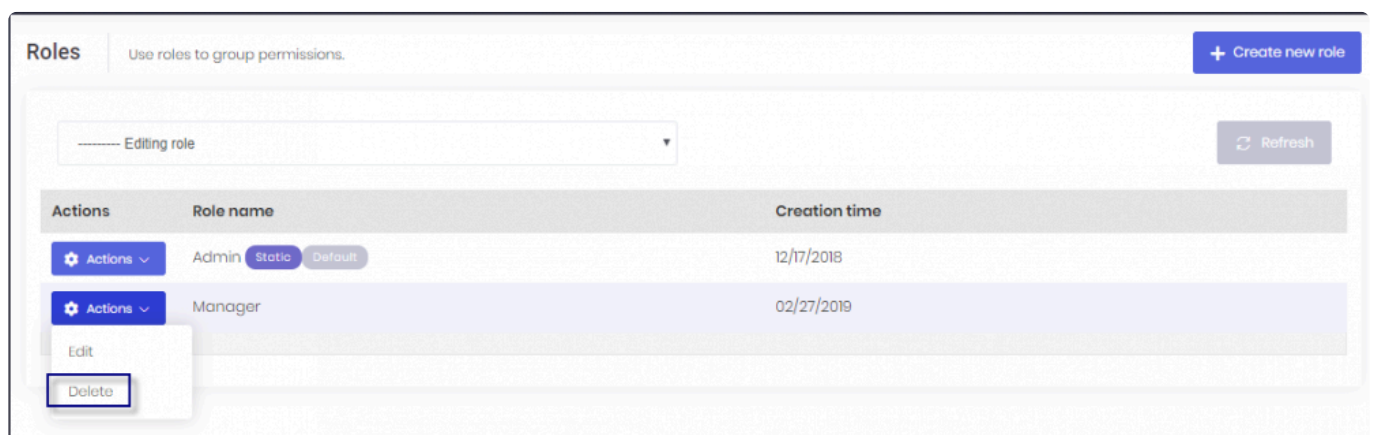
Follow below steps to delete role

Let's delete a role

- Click on “**Administration**” tab in left navigation
- Click on “**Roles**” option

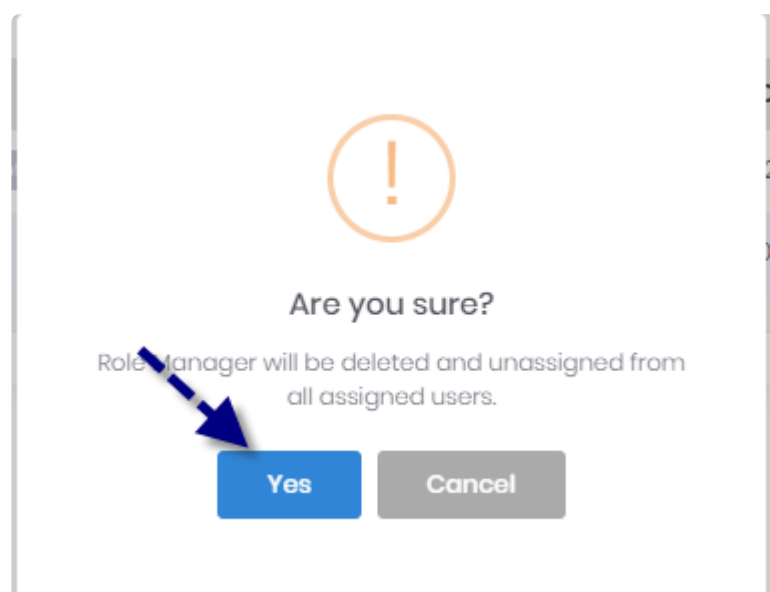


- Click on the “**Actions**” button on **Manage Roles** page
- Click on “**Delete**” button



- After click on “**Delete**” button, confirmation pop will displayed
- Click on “**Yes**” button on confirmation pop up

- System shall delete the role



! Deleted Role will be unassigned from all users

* Admin Role cannot be deleted

Last modified: 2019/05/28

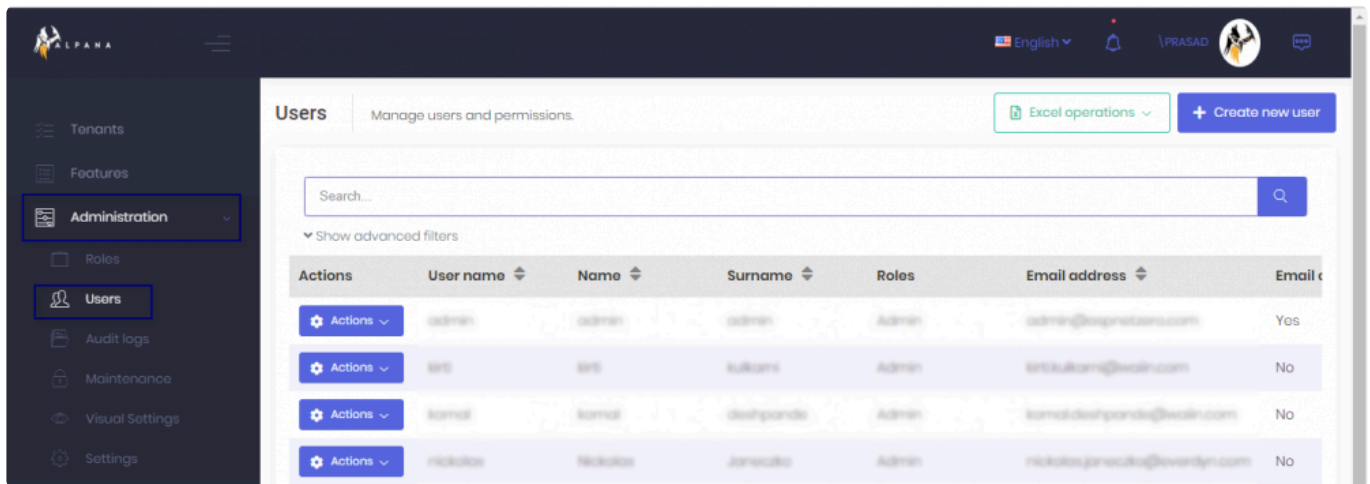
3.10.4. Assign Role to User

Default roles get assigned while creating a user. The “**Assign Role**” feature is specifically used for changing a user role or for assigning any custom role to the user.

Follow below steps to Assign roles to the user

Let's Assign role to a user

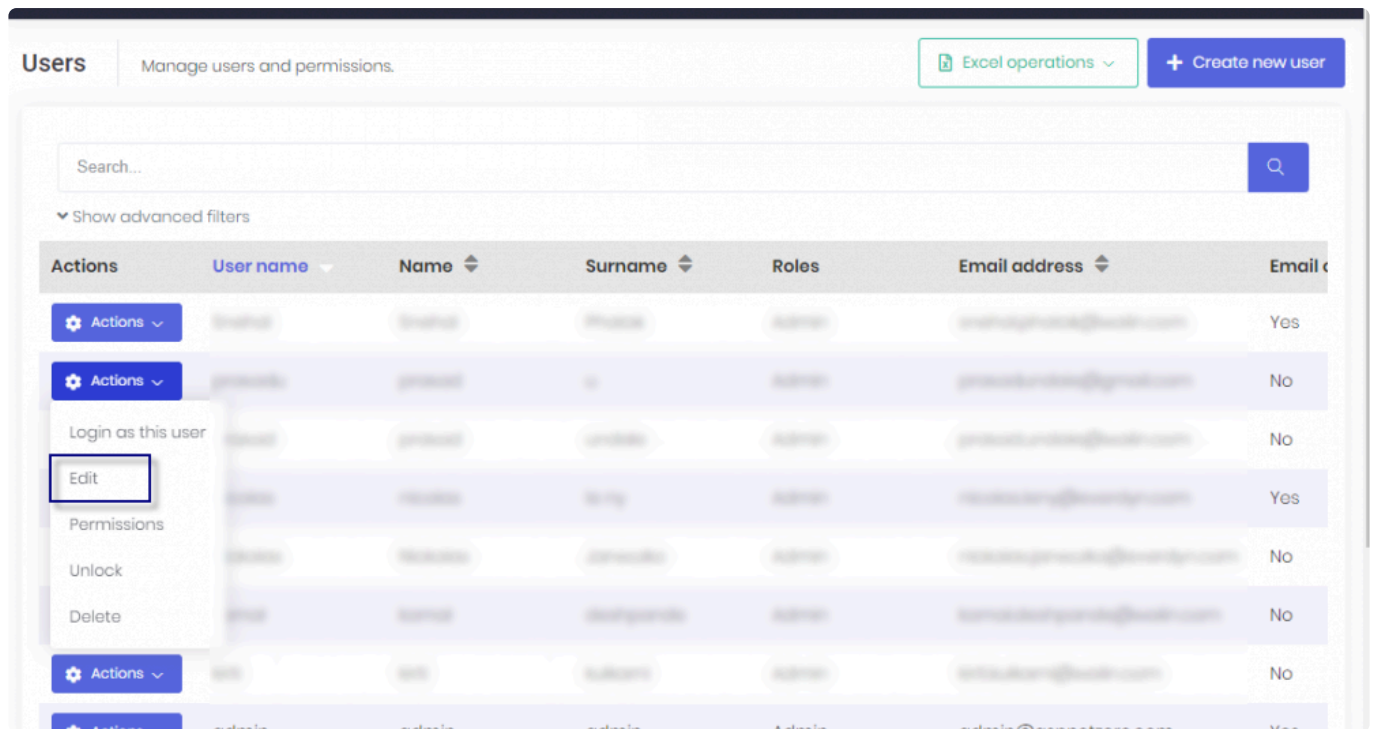
- Click on “**Administration**” tab in left navigation
- Click on “**User**” option



The screenshot shows the 'Users' management interface. The left sidebar contains a navigation menu with 'Administration' and 'Users' highlighted. The main content area is titled 'Users' and includes a search bar, a 'Show advanced filters' dropdown, and a table of users. The table has columns for 'Actions', 'User name', 'Name', 'Surname', 'Roles', 'Email address', and 'Email status'. The 'Users' button in the sidebar is highlighted with a red box.

Actions	User name	Name	Surname	Roles	Email address	Email status
Actions	admin	admin	admin	Admin	admin@alpana.com	Yes
Actions	test	test	kulkarni	Admin	testkulkarni@alpana.com	No
Actions	karnal	karnal	dashpanda	Admin	karnal.dashpanda@alpana.com	No
Actions	nikolas	Nikolas	Janezuko	Admin	nikolas.janezuko@alpana.com	No

- Click on the “**Action**” button on **Manage Users and permissions** page
- Click on “**Edit**” button



- Click on the **Roles** tab



- Unassign a currently assigned role by deselecting the the checkbox . Similarly a new role can be assigned by selecting the checkbox for required role and saving the changes.

- ✿ After changing the role, system will reassign all the permission to the selected user as per the new role.

Last modified: 2019/05/28

3.11. API

Alpana Server exposes a web API that exposes many actions and properties.
The following chapters document how to use this API.

Last modified: 2019/07/23

3.11.1. General API help and troubleshooting

Permissions

To work with the API, the user must have the required permissions on the Tenant :

- API permissions
- desired object permissions : for example, for API queries on Schedules, the user must have permissions on Schedules

API location

The API responds on the [Back-End website](#) (Alpana3App).

Resources

API Code snippets on Alpana Server

On your Alpana Server tenants, users with permissions to the API have access to auto-generated code snippets to help you make common requests.

So use this utility, login to the desired tenant (to the Front-End) and navigate to menu *API* :

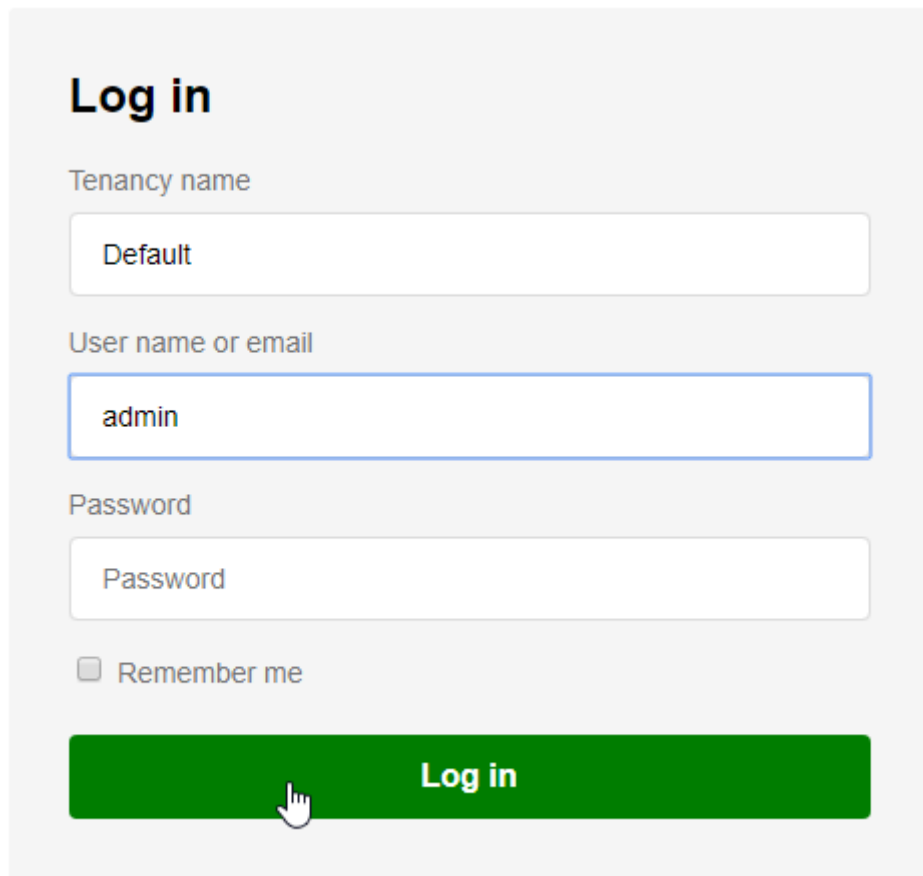


Swagger UI : Automatic documentation and test

Swagger UI allows you to access the full list of API endpoints, along with some auto-generated documentation, and an interface for testing queries.

To access it, simply use your web browser to navigate on the Back-End website of Alpana Server.

Login with your tenant credentials, and use the Swagger web page to browse endpoints, unfold their documentation, and try them from the browser.



Log in

Tenancy name

Default

User name or email

admin

Password

Password

☐ Remember me

Log in

Testing with Postman

You can easily test your API queries with a third-party tool called [Postman](#) .

This allows you to configure all parameters of the queries, and generate code in your programming language.

Common issues

If your application is a web application running in a browser, it will need to be allowed to make queries to the API. Otherwise, you will encounter CORS errors.

In order to do this, see [API Origins Whitelist](#).

Last modified: 2019/07/23

3.11.2. Authentication to the API

Getting code snippets for authentication

Authentication to the API is documented in each Tenant under the menu *API > Authentication*

Perform the documented query with a valid username/password combination in order to get an **authentication token**.

API Generate code templates to use the remote API.

AuthenticationRemote actionsUpdate Dependencies

Request access token

Supply account credentials to get access token

Perform a POST request to following endpoint

`https://[REDACTED]/api/TokenAuth/Authenticate`

Include following body in request

```
{  "userNameOrEmailAddress": "myUserName",  "password": "myPassword"}
```

Set the following request headers

`Abp.TenantId: 1`

The response result has the following structure. Extract the access token and the expiring information.

```
{  "success": "true | false",  "error": {    "code": "number",    "message": "string",  }
```

* It's up to you to keep the access token safe. If you're running the request from a website, consider storing it in an encrypted cookie or at server-side ; from a desktop application or middleware, follow the best practices and guidelines for your development platform.

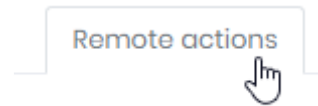
Important note on Security

If you need to perform authenticated queries programmatically from your application, you will need to consider the best practices for handling these credentials safely, without storing passwords for example.

Last modified: 2019/07/23

3.11.3. Remote Actions

Remote Actions are some commonly used API actions that trigger actions to clients. You can find the corresponding snippets by navigating to *API > Remote actions*.



Last modified: 2019/07/23

3.11.3.1. Refreshing / Updating Filters

Scenarios

Refreshing

You want to remotely force a refresh on any widget from any dashboard currently open in a browser.

This can be because you want event-based refresh.

This can be because you want very fast refresh (almost “live”) on some widgets.

You may want to filter target users who see this refresh.

This can be because you only want to force a refresh on the TV screen on the plant floor, and not disturb other users.

Updating Filters

You want to remotely send Parameters values inside any dashboard currently open in a browser.

This can be because you embedded the Alpana dashboard inside your application, and you want to send filter values from your application.

You may want to filter target users who see this filtering.

This can be because you want only the user currently browsing the application to see the filtering.

Code snippets

You can find code snippets by navigating to *API > Remote Actions* and selecting Action *Refresh / Filter* :

Action

Select API action to generate code for

Refresh / Filter



Fill in the desired information as *Target* on the left pane (like the desired Dashboard, Parameters, etc) :

Authentication

Remote actions

Update Dependencies

Target


Perform actions against following target

☒ Dashboard ☐ Published widget

Refresh_SysTime

Select dashboard

Select widgets to refresh (select none to refresh whole dashboard)

Grid 2 Header 

Perform actions against target displayed by following connected users

☐ Filter users by role

☐ Filter users individually

Then copy to your application the code snippets generated on the right.

Last modified: 2019/07/23

4. Alpana Mobile

- [Installation](#)
- [Usage](#)

Last modified: 2019/04/12

4.1. Installation

- [How to get Alpana Mobile](#)
- [Setup](#)

Last modified: 2019/04/12

4.1.1. How to get Alpana Mobile

You can download Alpana mobile through the usual providers:

- [Google Play Store](#) for Android devices
- [App Store](#) for Apple devices

Alpana Mobile is part of the Alpana software solution and as such is free to download.

Last modified: 2019/07/01

4.1.2. Setup

Address

At first launch you will be prompted to enter Alpana Sever address.

The format of your input should exactly match the address set for your server. For instance, if your server has been set to ignore the “www”, you should ignore them in your input.

✿ It is also important to start with “http://” (“https://” if your server has a security certificate).

Some examples:

- http://www.company.com
- https://company.com
- http://subdomain.company.com

Your server address could also be an simple I.P address “http://123.150.XX.XX”

If your server has been set for [Multitenancy](#) with tenancy sub-domains, you can take it into account in your input (“https://tenantX.company.com” , “https://tenantY.company.com”...).

In that case, the application will recognize the subdomain and set it as tenancy name for you.

Port

The port number to input is the one that has been set for your backend Alpana IIS server.

Last modified: 2019/04/12

4.2. Usage

- [Login](#)
- [Tenancy](#)
- [Dashboard Page](#)

Last modified: 2019/04/12

4.2.1. Login

Login screen

The login screen allows you to enter your login credentials. From there, you can also change the tenancy name and reset the server address.

Login behavior

After your first connection, the application will securely save your credentials and will automatically log you into the server at application startup.

However, if you haven't logged in the mobile application or the desktop web application for an extended period of time, your session can expire. When that happens, the mobile application will invite you to log out. You can then log in again and your session and your rights will be renewed.



Note that you cannot login to the application without a tenancy name set. See [Tenancy](#)

You can, through the side menu or the settings menu, log out. Doing so will end your session and navigate you to the login screen. The log out action will clear your saved password but tenancy name and server address will remain saved.

Last modified: 2019/04/12

4.2.2. Tenancy

Single tenancy

In the case of single tenancy, the application will try to set “Default” as tenancy name at first launch as is it the default tenant in the Alpana solution.

If your company changed the “Default” name for something else you will be prompted to change it. You can tap the change tenant box on the login screen and input the tenancy name that your company set. Provided that the tenancy name is valid, it will be set as tenant and saved.

Multitenancy

In the case of multitenancy, the application will set your tenancy name according to the sub domain name provided in the server address. You can, however, change the tenancy name at any time via the change tenant box on the login screen.



Please note that the mobile application does not allow to connect without tenancy name or “Host” mode.

Last modified: 2019/04/12

4.2.3. Dashboard Page

Once logged in, the application navigate you to the dashboard page. Here, you can browse through all the dashboard saved into your server.

This page is divided into five different tabs in appearing order:

- Favorites
- Recent
- Categories
- Public
- All

The dashboard list for each page is fetched once when the page first appear. You can however refresh them manually by pulling down the list. This features doesn't apply for subcategories navigation. If you want to refresh a dashboard list seen in a subcategory, you can go back then reopen said subcategory to refresh its content.

Last modified: 2019/04/12

5. Alpana Update File Dependency

Alpana dashboards rely on *dependencies* when they use an external file.

Sometimes, you need to update one such dependency. For example : when you need to update the content of an Excel file on a published dashboard without having to re-publish the dashboard.

You have two options :

- write an external application that makes use of Alpana Server API to push the updated file
- use the utility called “Alpana Update File Dependency” which implements this for common cases

Last modified: 2019/08/12

5.1. Installation

Installation of Alpana Update File Dependency is light on resource and pretty straightforward. Proceed to the next chapters to complete the installation.

Last modified: 2019/08/12

5.1.1. Pre-Requisites

This section explains the system requirements to run Alpana Update File Dependency.

Software Requirements

The following minimum software requirements are necessary to run Alpana Update File Dependency :

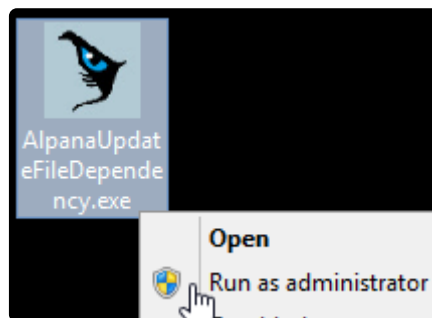
- .NET Framework 4.8 can be found [here](#) then download the “Runtime”. Alternately, an offline installer exists on the same page . **Installing this component requires to restart the machine.**
- Alpana Server version 3.1.x or greater is required
- Alpana Advanced license, to support API calls

Last modified: 2019/08/12

5.1.2. Installing

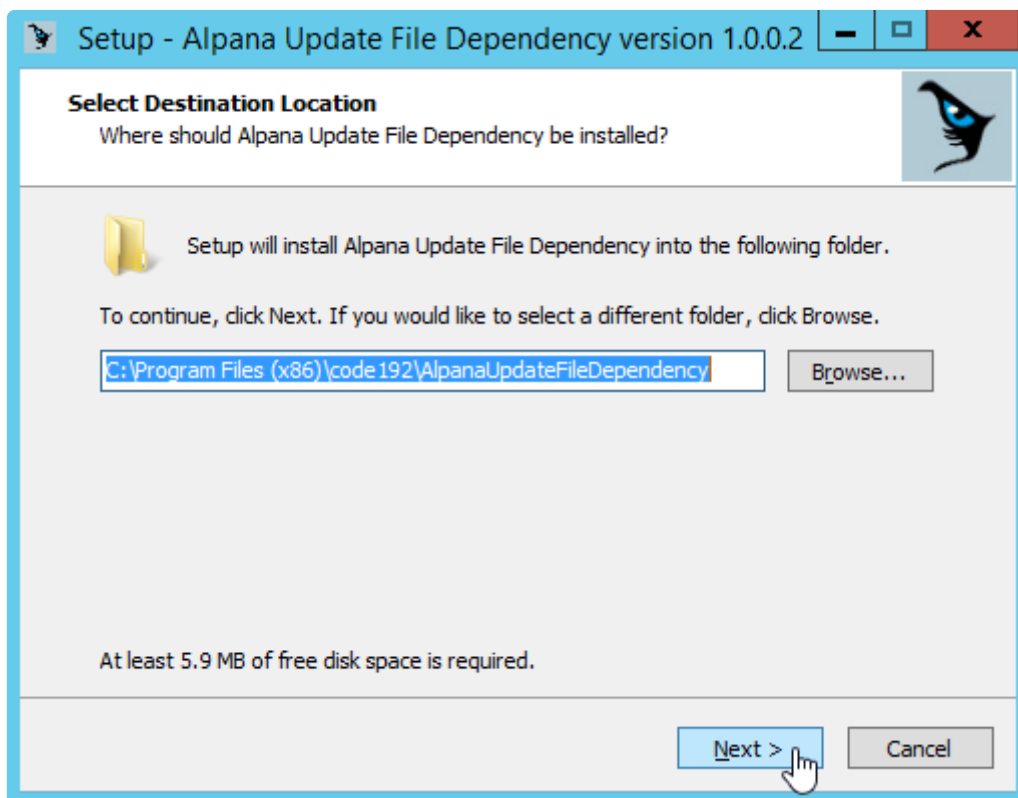
Run installer

Double-click the installer to start installation :



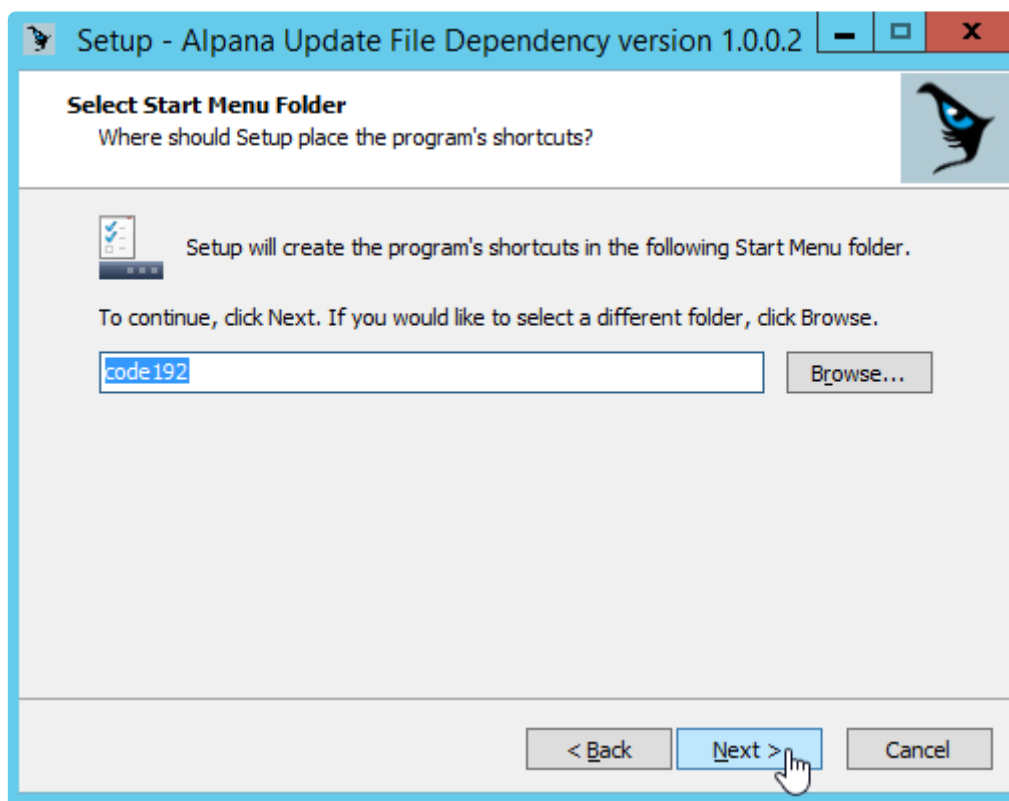
Select folder

Select the installation folder and click *Next* :



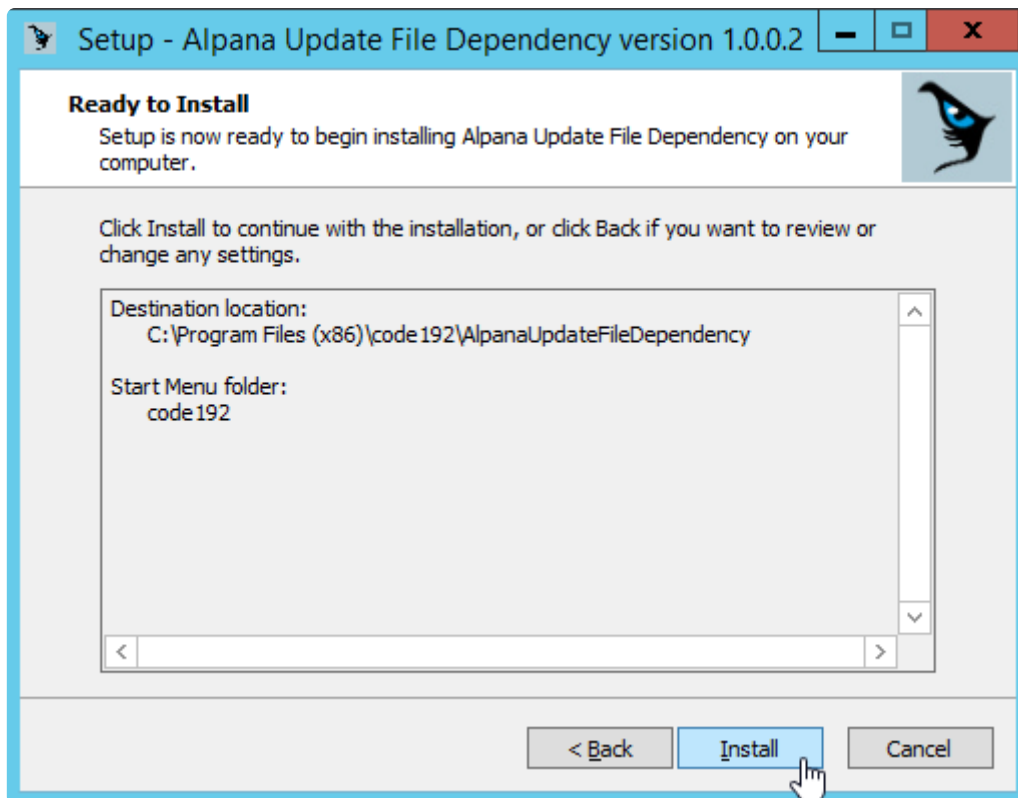
Start Menu folder

Select start menu folder and click *Next* :

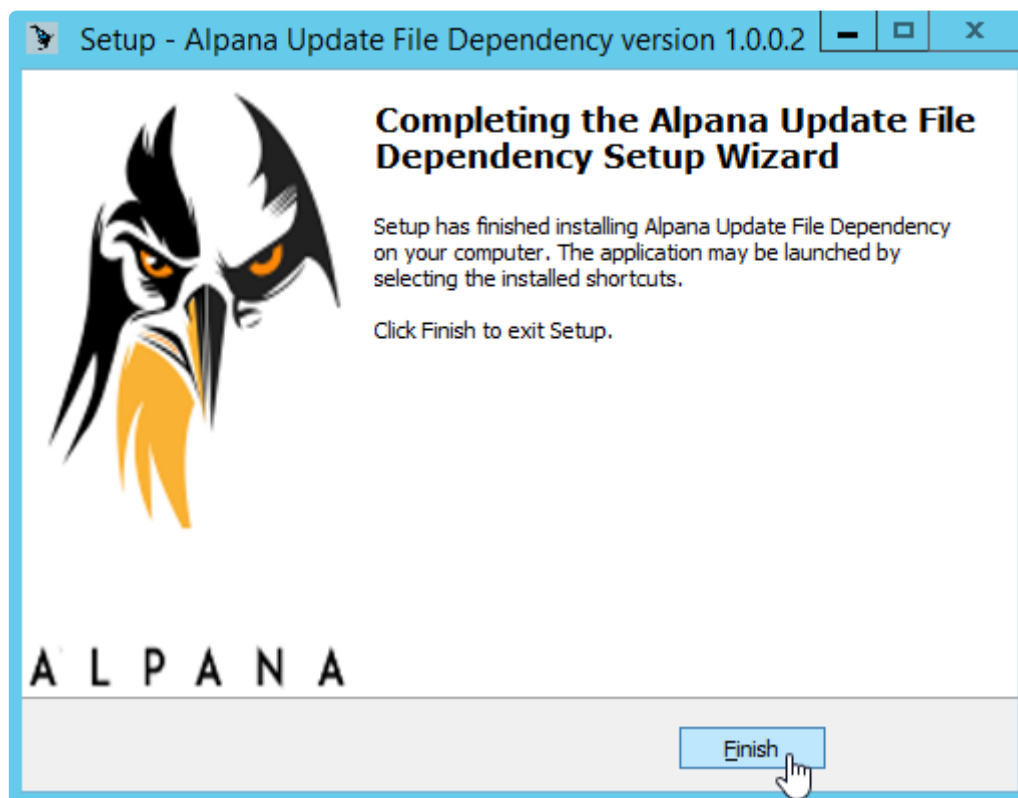


Complete installation

Review options, click *Install* :



Wait for completion and click *Finish* :

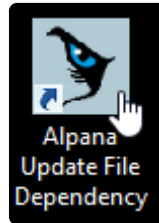


Last modified: 2019/08/12

5.2. Running the application

The application runs as a Windows service called “AlpanaDependencyUpdate”.
This allows to run jobs even when no user is logged in.

To start configuring Jobs and start/stop the Service, run the “Alpana Update File Dependency” shortcut :



When finished configuring, you can close the application, or even the Windows session, as the Jobs will be run by the Windows service.

When starting for the first time, the application requires to [start the service](#).

Last modified: 2019/08/12

5.3. Settings

Opening Settings

To manage application settings, click the tab *Settings* :



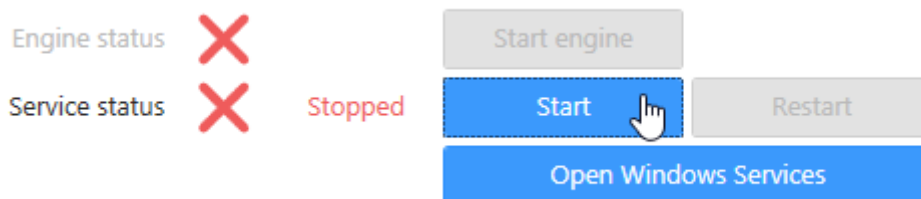
Starting the Service and the Engine

When you start the application for the first time, you need to start the Windows service, and the update engine.

The *Settings* window also allows to see the status of the service and engine.

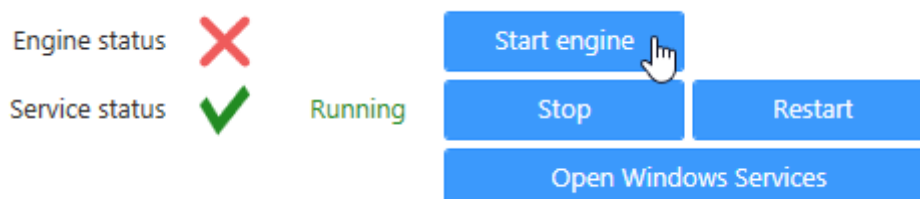
Start Service

Click *Start Service* to start the Windows service :



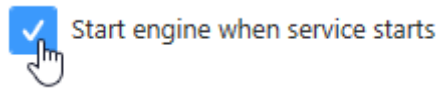
Start Engine

Click *Start Engine* to start the Engine :



Start Automatically

You can also configure the Engine to start automatically when the Service starts



Last modified: 2019/08/12

5.4. Connecting To Alpana

To log in or log out to your Alpana Server, click the tab *Configuration* :



Log in

Fill in the login information and click *Connect* :

Host	<input type="text" value="http://192.168.1.37"/>
Backend Port	<input type="text" value="8080"/>
Tenancy Name	<input type="text" value="Default"/>
Username	<input type="text" value="admin"/>
Password	<input type="password" value="•••••"/>

Connect

These credentials will be remembered by the application and will be used to run the Jobs.

Host

URL of the Alpana Server with which you want to communicate.
Include “http” or “https” as relevant.



A license “Alpana Advanced” must be active on the server, and the Feature “API” must be activated.

Backend Port

Network port to the **Backend** website of the Alpana Server.

Tenancy Name

Name of the Tenant where your dashboards are published.

✿ The Feature “API” must be activated on the Tenant.

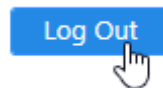
Username and Password

Login information for a User of the Alpana Server with the necessary permissions.

The User must have API permissions, and permissions to the dashboards that will be targeted by the application.

Log out

To log out, simply click *Log Out* on the top right :



Last modified: 2019/08/12

5.5. Jobs

The application will run Jobs that update the file dependencies of Dashboards, based on conditions.

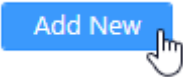
To access the Jobs configuration, click the tab *Configuration* and [log in](#) if necessary :



Last modified: 2019/08/12

5.5.1. Creating a Job

To create a Job, click the *Add New* button :



This creates an un-configured Job :

Target Dependency		Source File		Update Mode		Enabled	Upload	Remove
(Not Set)	...	(Not Set)	...	Manual Only	...	<div></div>	<div></div>	<div></div>

Last modified: 2019/08/12

5.5.2. Configuring a Job

When a new Job is created, it is not configured.

You can also configure an existing job.

To configure a Job, configure each property :

- *Target Dependency*
- *Source File*
- *Update Mode*

Last modified: 2019/08/12

5.5.2.1. Target Dependency

Selecting the Target Dependency means selecting which file you want to update from which existing Dashboard.

To configure the Target Dependency, click the “...” button :

Target Dependency

(Not Set)

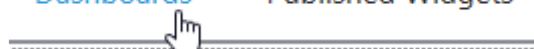


Dashboard or Widget

You can select a published Dashboard or Widget by clicking at the top of the *Select Target Dependency* window :

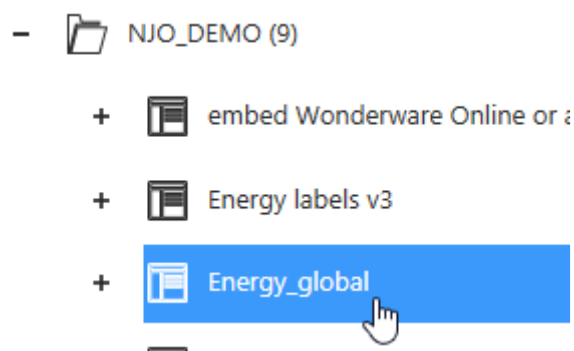
Dashboards

Published Widgets



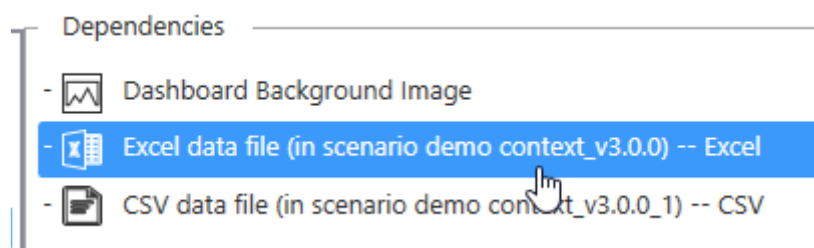
Browse Dashboard

Browse Categories and Dashboards in the left pane :



Browse Dependency

After clicking on a Dashboard on the left pane, the right pane lists all its dependencies :



Click *OK* to validate :

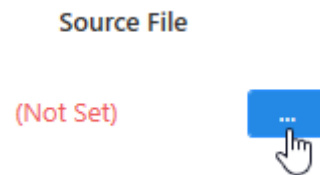


Last modified: 2019/08/12

5.5.2.2. Source File

Selecting the Source file means selecting the path where you want to fetch the new file that will replace the current dependency.

To configure the Source File, click the “...” button and browse for a desired file :



Important : If you plan to schedule automatic updates, choose well this file path so that it can be accessed by the Windows user running the Service.
See [Troubleshooting](#) for more information.


Last modified: 2019/08/12

5.5.2.3. Update Mode

Selecting the update Mode means choosing when you want the update to happen.

To configure the Update Mode, click the “...” button :

Update Mode

Manual Only 

Select mode :

These modes can be combined and selected by checking the corresponding boxes :

- ☒ Manual Only
- ☐ Scheduling Timer
- ☐ Whenever Source Changed

Manual Only

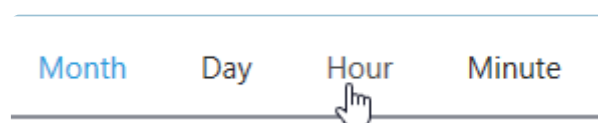
With *Manual Only*, the update will only happen when a user will trigger it manually inside the Alpana Update File Dependency application.

This mode doesn't require additional configuration.

Scheduling Timer

With *Scheduling Timer*, the update will happen at selected time intervals.

Select the time interval type at the top of the window :



For the desired interval, configure more precisely the desired schedule :

The screenshot shows a scheduling configuration interface with the following options:

- ☒ Any Day
- ☐ Every Day(s) starting on
- ☐ Every Day(s) starting on the of the month.
- ☐ Specific Day of Week
 - ☐ Sunday
 - ☐ Monday
 - ☐ Tuesday
 - ☐ Wednesday
 - ☒ Thursday
 - ☐ Friday
 - ☐ Saturday
- ☐ Specific Day of Month

<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3	<input type="checkbox"/> 4	<input type="checkbox"/> 5	<input type="checkbox"/> 6	<input type="checkbox"/> 7	<input type="checkbox"/> 8	<input type="checkbox"/> 9	<input type="checkbox"/> 10	<input type="checkbox"/> 11
<input type="checkbox"/> 12	<input type="checkbox"/> 13	<input type="checkbox"/> 14	<input type="checkbox"/> 15	<input type="checkbox"/> 16	<input type="checkbox"/> 17	<input type="checkbox"/> 18	<input type="checkbox"/> 19	<input type="checkbox"/> 20	<input type="checkbox"/> 21	<input type="checkbox"/> 22
<input type="checkbox"/> 23	<input type="checkbox"/> 24	<input type="checkbox"/> 25	<input type="checkbox"/> 26	<input type="checkbox"/> 27	<input type="checkbox"/> 28	<input type="checkbox"/> 29	<input type="checkbox"/> 30	<input type="checkbox"/> 31		
- ☐ On the last day of the month
- ☐ On the last week day of the month
- ☐ On the last of the month
- ☐ day(s) before the end of the month

Whenever the Source Changed

With *Whenever the Source Changed*, the service will watch the file for any change and will trigger the update.

This mode doesn't require additional configuration.

Last modified: 2019/08/12

5.5.3. Running jobs

Run Automatically

Jobs can be run automatically when Enabled.

✿ Jobs are not Enabled by default.

To Enable a Job, check the *Enabled* box :

Enabled



To Disable a Job, uncheck the *Enabled* box :

Enabled



Run Manually

To manually force the update now, click the *Upload* button :

Upload



Last modified: 2019/08/12

5.5.4. Deleting a Job

To delete a Job, click the *Remove* button in front of it :

Remove



! This will remove the Job permanently.
If instead you want to disable it temporarily, [juste Disable it.](#)

Last modified: 2019/08/12

5.6. Troubleshooting

Logs

Application logs are available in the tab *Logs* :



They contain the history of run jobs and useful messages upon failure.

Filter by log level

All

Refresh

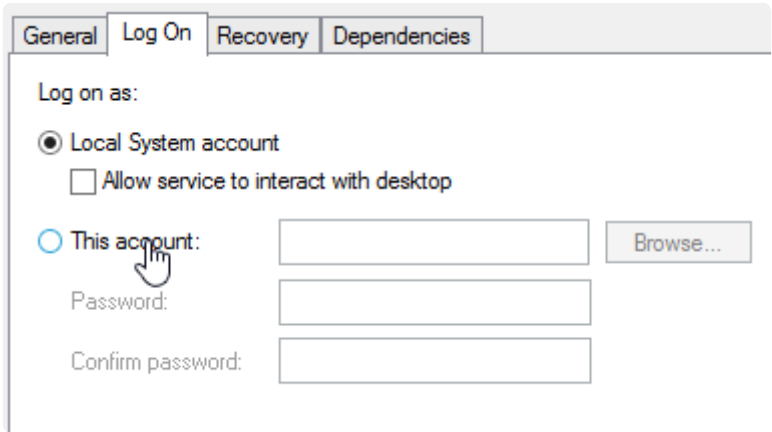
Delete All

Date/Time	Type	Message	Remove
7/24/2019 5:19:15 PM	Info	Update engine started	X

Common Issues

I/O Errors and file permissions

The Windows Service must have access to the source files.
By default, the Service is run by a Local System user.
Check if this user has correct permissions, or change what Windows account is used to run the Service :



Last modified: 2019/08/12