Boxplot Extension

ThingWorx Boxplot Widget User Guide

Contents

Document Revision History

Revision Date	Description of Changes

Software Change Log

Version	Release Date	Changes
0.0.50	-	Beta release.
0.0.59	-	Add Frame Style Configuration.
0.0.67	-	Change IDE Design. Change Icon. Change scaling of the widget.
0.0.83	18.12.2017	Scaling Fix, Repeater Issue Fix, Add responsive feature
0.0.88	19.12.2017	More than 2 rows in repeater Fix.

Prerequisites

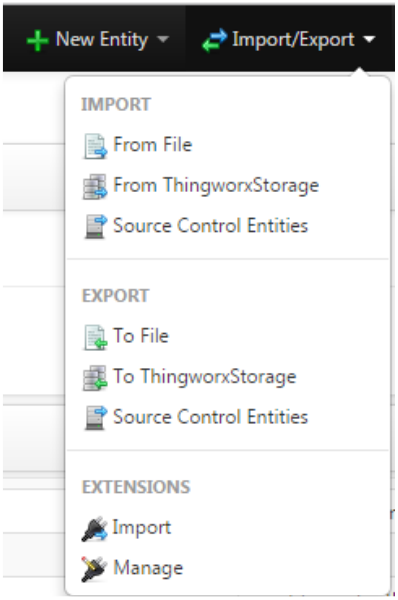
Prerequisites
ThingWorx 5.4.0 +
Browser with ECMAScript 5 compatibility - http://kangax.github.io/compat-table/es5/

Tested On

ThingWorx Version	Browser	Status
ThingWorx 8.0.4-b46	Google Chrome Version 61.0.3163.100 (Official Build) (64-bit)	**PASS**
ThingWorx 8.0.4-b46	Mozilla Firefox Version 46.0	**PASS**
ThingWorx 8.0.4-b46	Opera 48.0.2685.52 (PGO)	**PASS**
ThingWorx 8.0.4-b46	IE 11.1176	**PASS**
ThingWorx 8.0.4-b46	Edge 25.10586	**PASS**
ThingWorx 8.0.4-b46	Safari Version 10.1.2 (12603.3.8)	**PASS**

Installing the Widgets

1. From a web browser, launch ThingWorx.
2. Log into ThingWorx as an administrator.
3. Go to Import/Export > Import.



4. Click Choose File and select the **twx-wdg-knorrbremse-boxplot.zip** from wherever you have saved it.
5. Click Import. **NOTE**: If an Import Successful message does not display, contact your ThingWorx System Administrator.
6. Click Yes to refresh Composer after importing the extension.

Building Boxplot

As with any data-rendering widget in ThingWorx Composer, a Boxplot widget must be placed in a mashup and configured with incoming data bindings. To build a Boxplot:

1. Drag and drop Boxplot widget into responsive container on mashup.
2. On the right, add a data source entity and, from the **Returned Data**, drag **All Data** to the boxplot and bind it to the **Data** property. This binding defines where the data is loaded from, when the boxplot is launched.

   Note: It is required to bind data which is an InfoTable with defined DataShape.
3. Configure widget properties properly (see Properties chapter).
4. Save and View the completed mashup.

Properties

Property Name	Description	Base Type	Default Value	Bindable (Y/N)?
Id*	A unique identifier used internally by ThingWorx.	INTEGER	boxplot-<id>	N
Type*	The widget type.	n/a	Boxplot	N
DisplayName*	A user-defined name to identify the grid when displayed.	STRING	boxplot-<id>	N
Description*	A user-defined description.	STRING	n/a	N
Title	Title of Boxplot displayed in the top-middle position of Boxplot. • If the plot is bound to a data source, a filled arrow is displayed: ⬅ • If there is no data source, the arrow is unfilled: ⇦	STRING	Boxplot	Y
FrameStyle	Style definition for Frame of widget (CSS): • Foreground Color: border-color • Line Thickness: border-width • Line Style: border-style	STYLEDEFINITION	DefaultChartStyle	N
NumberOfSeries	Number of data series displayed in separate boxplots in 1 widget (minimum 1, maximum 10)	NUBMER	1	N
Data	Source of data that loads when the boxplot is launched. • If the plot is bound to a data source, a filled arrow is displayed: ⬅ • If there is no data source, the arrow is unfilled: ⇦	INFOTABLE	n/a	Y
Orientation	Specifies if data in boxplot should be rendered vertically or horizontally. Available options: 1. Vertical (see Figure 2 Vertical Boxplot) 2. Horizontal (see Figure 1 Horizontal Boxplot)	STRING	Vertical	N

BoxplotStyling	Specifies if data in boxplot should be rendered with additional information. Available options: 1. Standard – no additional information (see Figure 2 Vertical Boxplot) 2. Only Mean – add mean value to the plot (see Figure 3 Boxplot with mean) 3. Mean and Standard Deviation (see Figure 4 Boxplot with mean and standard deviation)	STRING	Standard	N
ShowLegend	Specifies if legend should be visible on boxplot widget	BOOLEAN	true	N
SeriesStyle1	Styling for Series Data number 1	STYLEDEFINITION	DefaultChartSeries1	N
SeriesStyle*[n-1]*	Styling for Series Data number *n-1*, where *n* is NumberOfSeries value. This property is generated automatically after changing NumberOfSeries property value to different than 1.	STYLEDEFINITION	DefaultChartSeries[n-1]	N
SeriesStyle*[n]*	Styling for Series Data number *n*, where *n* is NumberOfSeries value. This property is generated automatically after changing NumberOfSeries property value to different than 1.	STYLEDEFINITION	DefaultChartSeries[n]	N
SeriesLabel1	Label for Series Data number *n-1*, where *n* is NumberOfSeries value. This property is generated automatically after changing NumberOfSeries property value to different than 1.	STRING	n/a	Y
SeriesLabel*[n-1]*	Label for Series Data number *n-1*, where *n* is NumberOfSeries value. This property is generated automatically after changing NumberOfSeries property value to different than 1	STRING	n/a	Y
SeriesLabel*[n]*	Label for Series Data number *n*, where *n* is NumberOfSeries value. This property is generated automatically after changing NumberOfSeries property value to different than 1	STRING	n/a	Y
SeriesDataValueField1	Name of column from Data property, which will be assigned to series number 1. This property is available after assigning data with DataShape to Data property.	VALUEFIELD	n/a	N

SeriesDataValueField*[n-1]*	Name of column from Data property, which will be assigned to series number *n-1*, where *n* is NumberOfSeries value. This property is generated automatically after changing NumberOfSeries property value to different than 1. This property is available after assigning data with DataShape to Data property.	VALUEFIELD	n/a	N
SeriesDataValueField*[n]*	Name of column from Data property, which will be assigned to series number *n*, where *n* is NumberOfSeries value. This property is generated automatically after changing NumberOfSeries property value to different than 1. This property is available after assigning data with DataShape to Data property.	VALUEFIELD	n/a	N

Examples

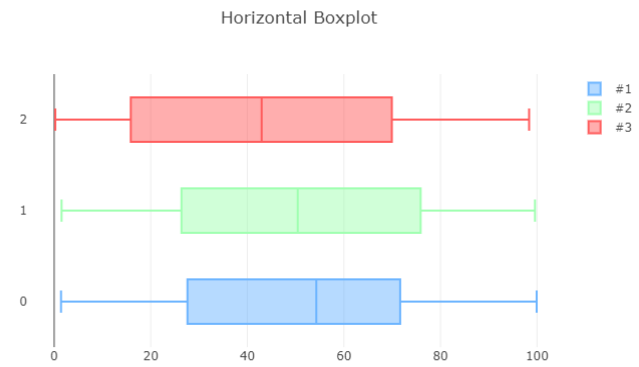


Figure 1 Horizontal Boxplot

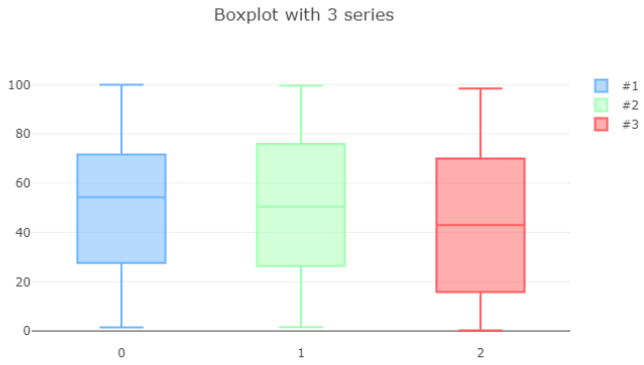


Figure 2 Vertical Boxplot

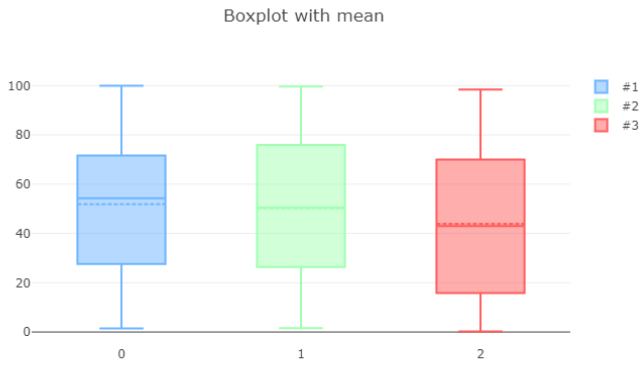


Figure 3 Boxplot with mean

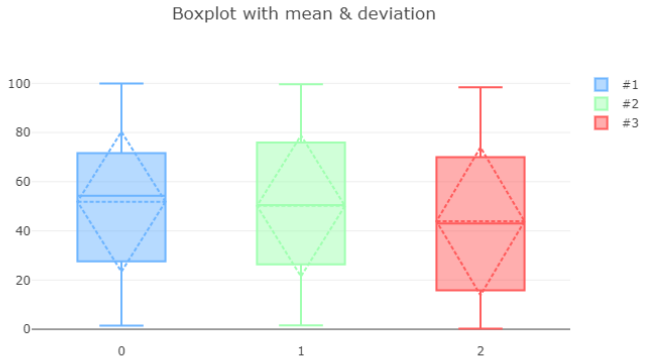


Figure 4 Boxplot with mean and standard deviation

Sample Data

1. From a web browser, launch ThingWorx.
2. Log into ThingWorx as an administrator.
3. Go to Import/Export > From File.
4. Choose **BoxplotSampleEntities.twx** file and Import it.
5. Navigate to the **BoxplotTestMashup**, open it and click on View Mashup.

Note:

Data is pushed to Plots thanks to the **BoxplotDataProviderMock** Thing and random data generator service written in Java Script.