

Unified-E App Designer

Software Version 2.2.0.0, Last Update: July 2019

Content

1	General	3
1.1	Definition: Operator App	3
1.2	Definition: Endpoint	4
1.3	Definition: Datapoint	4
1.4	Gateway Communication vs Direct Communication	5
1.4.1	Gateway Communication	5
1.4.2	Direct Communication	5
2	App Properties	5
2.1	Display-specific Settings	6
2.2	Common Properties	6
3	Configure Endpoints	8
3.1.1	Select Adapter and Configure	8
3.1.2	Datapoint Table	8
3.1.3	Create Formulas	10
3.1.4	User-specific Datapoints	10
4	Create Views	11
4.1	Define Default Appearance for View Elements	11
4.2	Create a New View	12
4.3	View Structure	13
4.4	Add Images	14
4.5	Create and Use Types	15
4.5.1	General	15
4.5.2	Use Types	15
4.5.3	Type Categories	17
4.6	View Editor	18
4.6.2	View	25
4.6.3	Panels	27
4.6.4	Elements	32
5	Configure Push Notifications	41
5.1	General	41
5.2	Add Push Notifications	41
5.3	Push Notifications Table	42

5.4	Further Push Notification properties	42
6	Configure User Roles	43
6.1	Define User Roles	43
6.2	Activate Access Control	44
7	Test the App.....	44
7.1	Testing with Direct Endpoint Connection	44
7.2	Testing with Simulated Datapoint Values	45
8	App Publishing	46
8.1	Create App Package File.....	46
8.2	Publish App Package to Gateway	46
8.3	Publish App Package for Direct Communication	47
8.4	Update Existing App.....	48
9	Multilingual Apps	48
9.1	General	48
9.2	Add a New Language.....	49
9.3	Modify Texts in the Language Editor	49
9.4	Set App Language on Smartphone	50
10	Installed Endpoint Adapters	50
11	Import and Export Data.....	51
11.1	Data Exchange with Microsoft Excel	51
11.2	Data Exchange with CSV- and Text Files.....	52
12	Handle Endpoint Messages	52
12.1	Define Datapoints for Messages.....	52
12.2	Define Message texts with a Status Text Type	52
12.3	Add Message List to View	52
13	Curve Recordings and Time Charts	53
13.1	Create a Curve Recording Datapoint.....	54
13.1.1	Add Curve Recording	54
13.1.2	Dialog "Configure Curve Recording"	55
13.1.3	Edit Curve Recording	56
13.2	Create Time Chart.....	56

1 General

A brief description of the entire Unified-E product can be found at <http://www.unified-e.com/en/GettingStarted>. The “Unified-E App Designer” software is only one part of this product.

App project:

With Unified-E App Designer (also just called “App Designer” in the following) you can create individual operator apps for machines. All configuration data of an operator app will be saved in a “Unified-E App Project” – also just called “Project” in the following.

Start object editor:

All project's main objects are listed in the left area called “Project navigation”. By “double-clicking” on the object you can open the according editor.

Editors can also be displayed in a new window by dragging the mouse on the respective editor tab.

Example projects:

Example projects are already included in the App Designer Setup.

They will be located in the < Documents>Unified-E App DesignerApp Projects folder by default.

Select „File -> Open Project...“, to open an example project.

1.1 Definition: Operator App

Operator app vs. the app from store called Unified-E:

The operator app created in the App Designer is published as follows:

- 1) Create app project in App Designer
- 2) Generate app package based on the app project
- 3) Publish app package
 - a) For gateway communication:
 - i) install app package in App Manager (separate software on the gateway)
 - ii) Register the operating app in the Unified-E App from App Store
 - b) For direct communication:
 - i) Use app package directly and register in Unified-E App from App Store

The operator app is therefore not published directly in a public app store (such as Google Play), but it is executed within the Unified-E app.

For the different systems Android, Windows, and iOS, there is the Unified-E App, which can be found under the name "Unified-E" in each store. There, the operator app created by you can then be registered directly or via App Manager.

Example:

There are two operator Apps registered in the “Unified-E” App on a smartphone.



1.2 Definition: Endpoint

An endpoint must meet the following criteria:

- Must be able to communicate with the App Manager (e.g. via Ethernet, WLAN)
- Monitor:
Must be able to provide values by a unique ID or address (e.g. temperature, stock, typically a PLC variable). These values (also called datapoints) will be visualized in an operator app view.
- Control:
Must allow to set values by a unique ID

Endpoint examples:

- PLC, controller
- Database, ERP
- Web server

Endpoint adapter:

An Endpoint Adapter for the required endpoints is needed in order for the App Designer or App Manager to be able to communicate with the endpoints. For common endpoints adapters are already being provided. Not all adapters are supported for direct communication without App Manager (see also website).

Configure endpoints:

An operator app communicates with one or multiple endpoints. An endpoint typically has an IP address and further communication parameters. All endpoints parameters can still be adjusted after the operator app has being installed at the App Manager.

1.3 Definition: Datapoint

A value of an endpoint which can be identified by an ID (address, Plc variable) is called a datapoint.

Identification of a value takes place depending on the endpoint:

- By symbol: If you do not specify an address explicitly in the datapoint table then the datapoint name is used for addressing.
- By address: Contains all information required to determine the value.

Datapoint examples:

- With Beckhoff controllers: datapoint name refers to a variable e. g. of the variable table of a TwinCAT project.
- With Siemens controllers: datapoint address contains all information (including type as DINT) for writing/reading the variable.

1.4 Gateway Communication vs Direct Communication

In gateway communication, the operator device communicates with the endpoint using a gateway PC on which the App Manager application is installed.

In the case of direct communication, the Unified-E App communicates directly with the endpoint on the operator device (smartphone); the App Manager application is not required for this.

1.4.1 Gateway Communication

Communication of the operator devices via the gateway requires a PC, which acts as a link between the operator devices and endpoints.

Advantages:

- Push notifications are sent by the App Manager
- The App Manger manages the users so that only authorized users have access to the plant/machine
- App Manager functions such as curve recording can be used
- All delivered endpoint adapters can be used

1.4.2 Direct Communication

In direct communication, the operator panel can directly access the endpoint (controller). Currently (as of October 2018) the following adapters are available:

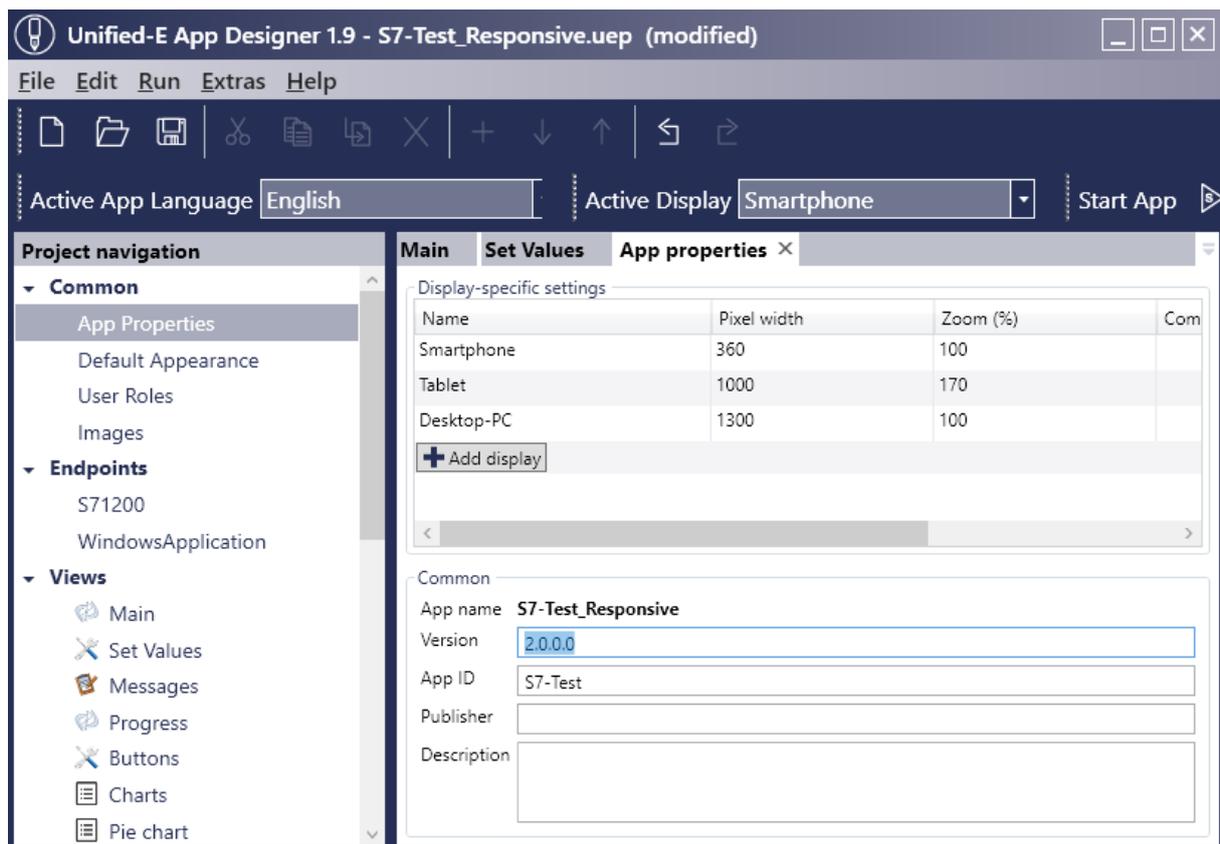
- Allen Bradley adapter
- Siemens S7 adapter
- Modbus adapter
- OPC UA adapter

Advantages:

- No PC required

2 App Properties

Once a new project is created or an existing project has been opened, you can navigate to the app properties by double-clicking on “App properties” in the project navigation.



2.1 Display-specific Settings

Views can be configured with different layouts depending on the display size (e. g. tablet portrait or tablet landscape).

The operator app selects the layout of the corresponding display group based on the current display pixel width of the device.

Each display group has the following properties (see table above):

- Name of the display group.
- Pixel width: The logical minimum width (also CSS or HTML width) in pixels. This width is also used in the view editor.
- Zoom (%): The zoom factor which should be applied for all views if the display group is active.
- Comment: Any comment.

The layout of a view for a display group is defined in the properties of the view, see also chapter 4.6.2.3.

2.2 Common Properties

App name:

This is always the file name of a project and will be suggested when installing.

Version:

The version is displayed in the App Manager's app properties and has no technical meaning so far.

App ID:

The app ID should be as unique as possible as it is used for remote updates of an app in the App Manager.

Publisher (optional):

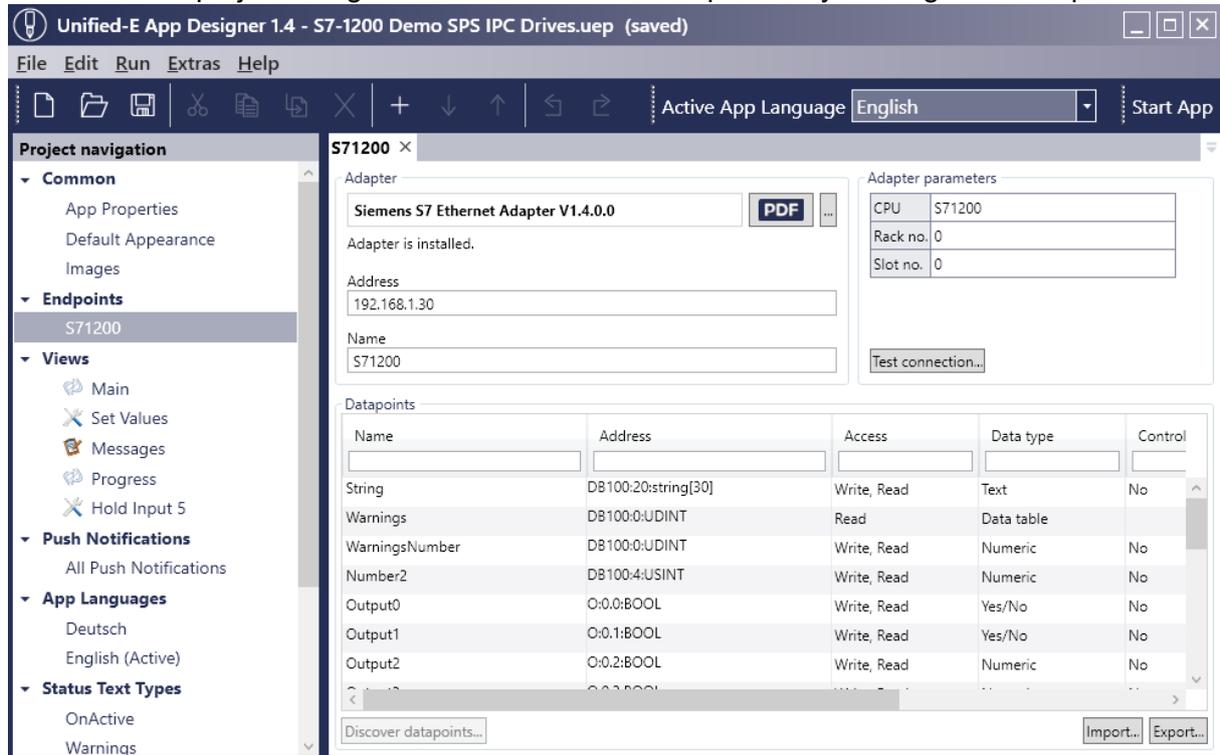
Name / Company of the app

Description (optional):

Brief description of the app, which will be displayed in the App Manager.

3 Configure Endpoints

By default, a new project is created with an endpoint object. Further endpoints can be created in the project navigation's context menu "Endpoints" by clicking "Add Endpoint".



3.1.1 Select Adapter and Configure

First, select the appropriate adapter by clicking the "...". Depending on the adapter the according adapter parameters are displayed in the right table. You will find details for setting the adapter address and the adapter parameters in the according adapter documentation (click on the PDF button). How to install an endpoint adapter is described in Chapter 9.

3.1.2 Datapoint Table

3.1.2.1 Add Datapoints

"Discover datapoints..." button:

This button imports datapoints. It is only activated when this feature is supported by the adapter.

"Add curve recording..." button:

You can add a curve recording datapoint here. Details are described in chapter 13.

3.1.2.2 Datapoint Table Columns

Name	Address	Access	Data type	Control PIN	Def. sim. value	Value	Group	Comment
AlarmTrigger	AlarmTrigger	Write, Read	Numeric	No	0	0		
ValveOpened	ValveOpened	Write, Read	Yes/No	No	0	1		
CurrentAction	CurrentAction	Write, Read	Numeric	No	1	1		

“Name” column:

Identification of datapoint. The label has to be unique within the table.

“Address” column:

This is where the unique address is set (e.g. registration number, symbol name, variable name). When there is symbolic addressing the value in “Name” is automatically used as address, if this column remains empty.

“Access” column:

Read: Such datapoint can only be read

Write, Read: This datapoint can be read and written. Input elements for entering can only set datapoints of the type “Write, Read”.

“Data type” column:

Data type defines how the datapoint value is to be interpreted.

Numeric: A numeric value

Yes / No: Can have the following values: 0 (No) and 1 (Yes)

Text: Any text

Data table: The value is a data table where the cell values are organized in columns and rows. You can display the table content in the list panel, for further information, see 12.

Example:

A datapoint returns the value „1.23“. On computers with a comma as decimal separator this value is displayed as „1,23“, if data type is „Numeric“. For data type “Text”, the value is directly displayed as „1.23“.

“Control password” column:

Yes: To set a datapoint by an input element a control password is required – you will receive your control password when registering the operator App. For critical operations you should always set “Yes”.

No: To set a value no control password is required.

“Def. sim. Value” column:

This is where the standard value has to be entered which is used in the App simulator with simulated endpoints at startup. If the cell is empty, a default value will be set.

“Value” column:

This column can only be seen if the operator App has been started for testing. There, the current datapoint value is displayed.

When starting the operator App simulator with simulated endpoints, the value can be changed for testing purposes.

“Group” column:

In this column you can set the group name for grouping datapoints. It is possible to filter groups. The column may remain empty.

“Comment” column:

You can enter a comment for the datapoint here, but the comment is only visible in the table.

3.1.3 Create Formulas

A formula can be used to create simple arithmetic and logical expressions that contain datapoints as operands. The formula itself is also a datapoint, which can be used wherever datapoints of the type Text, Numeric or Yes/No are possible.

In the case of the formula operands, the operand datapoints can originate from different endpoints; the formula datapoint itself can be added at any endpoint.

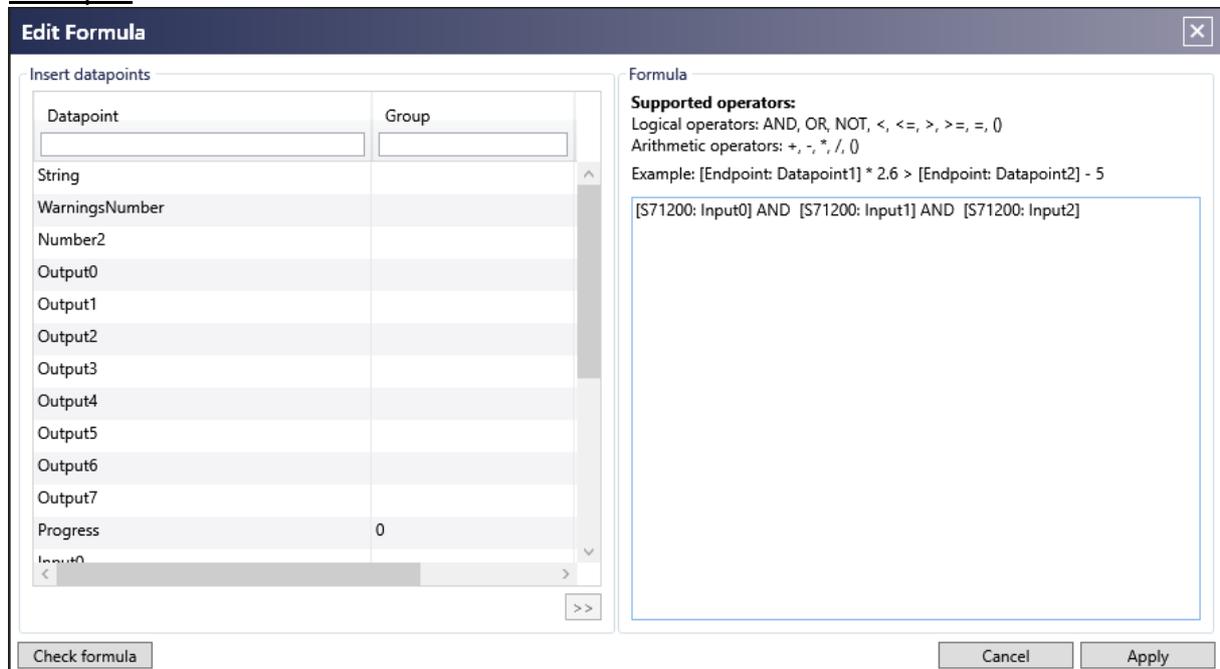
Add new formula:

Select the "Add formula..." button in the desired datapoint table. A new datapoint is created and the dialog "Edit formula" appears.

Edit existing formula:

In the datapoint table, select the "..." button in the "Address" column at the formula datapoint. Rework the formula in the "Edit Formula" dialog.

Example:



Formula guidelines:

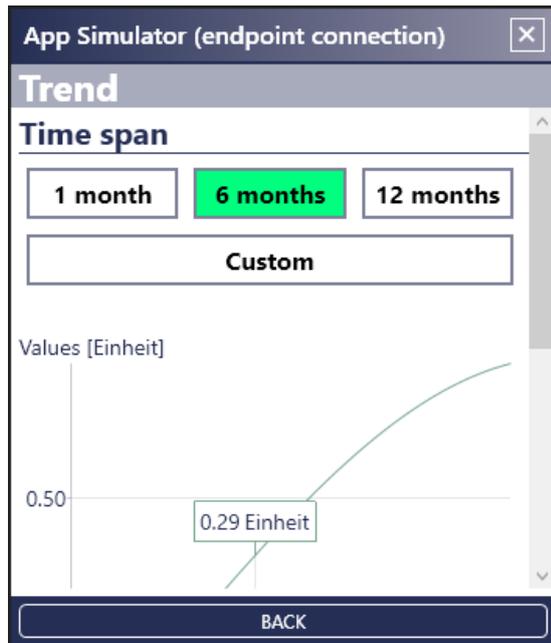
- For logical operands: 0 means "No", 1 means "Yes"
- Use the decimal point "." as separator

3.1.4 User-specific Datapoints

More and more often, there is the desire that not only machine states are visualized in the operator app, but also user-specific information is to be displayed:

Examples:

- Diagram or list display: Users want to see a custom range of data.
- Current production orders of the user.
- Current performance characteristics of the user.



To enable such features as well, datapoint contexts have been introduced.

Global datapoint context:

A data point in the global context always has the same value for all app users. Endpoint adapters for PLC controllers only support this context.

User-specific datapoint context:

This context is currently supported for WindowsApplication and WebHttp adapters. Depending on the app user, the endpoint must supply the corresponding datapoint value. The endpoint therefore receives the user from whom the request originates.

Language-specific datapoint context:

This context is currently supported only for WindowsApplication and WebHttp adapters and makes sense if all datapoint values are basically the same for all users. Only for "Text" datapoints the endpoint delivered the language-dependent text.

You can find more information in the WindowsApplication and WebHttp adapter user manuals.

4 Create Views

An operator app consists of one or more views. Using view links, you can navigate between views (like in a website). Home view is the first view that appears right after starting the operator app.

4.1 Define Default Appearance for View Elements

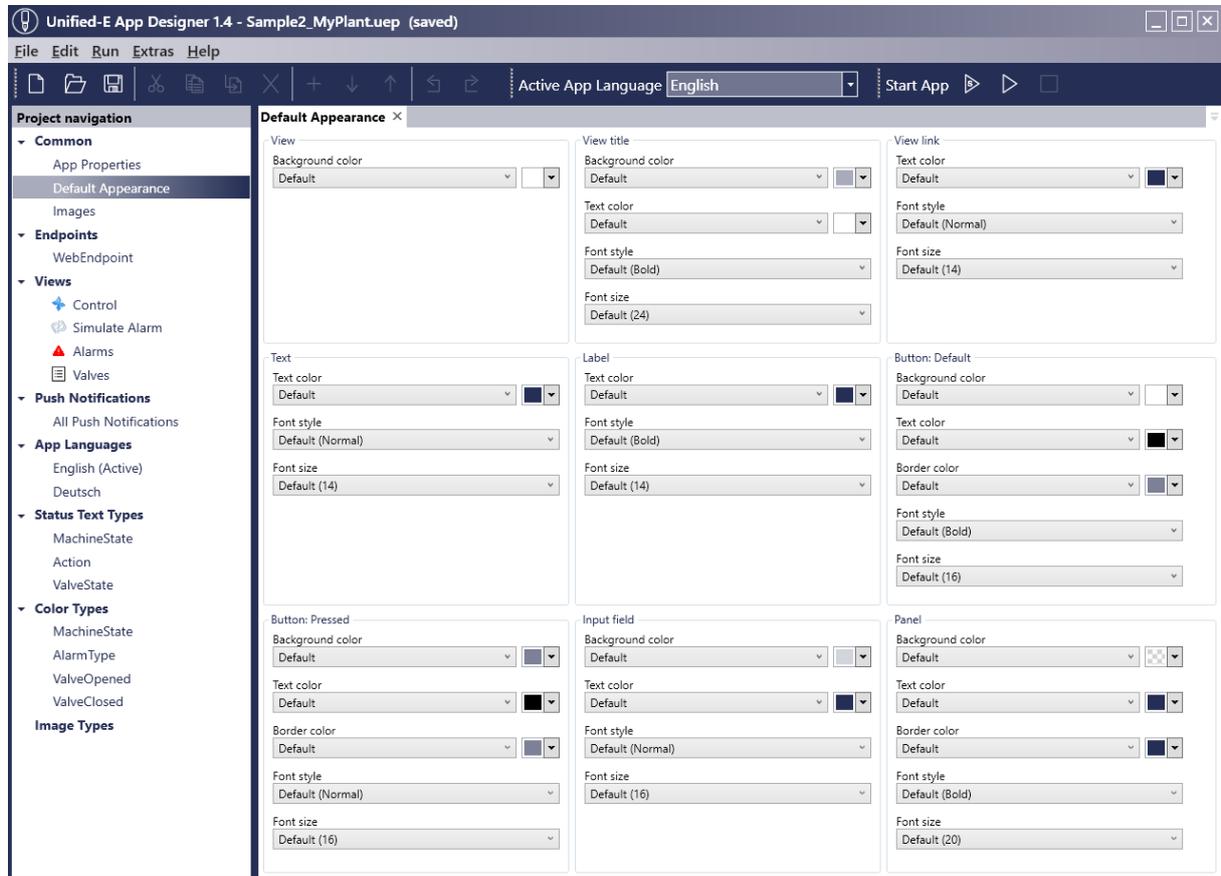
You can define the default appearance of the view elements in the project navigation in Common -> Default Appearance.

The following view element properties can be modified:

- Background color
- Text color
- Border color
- Font style

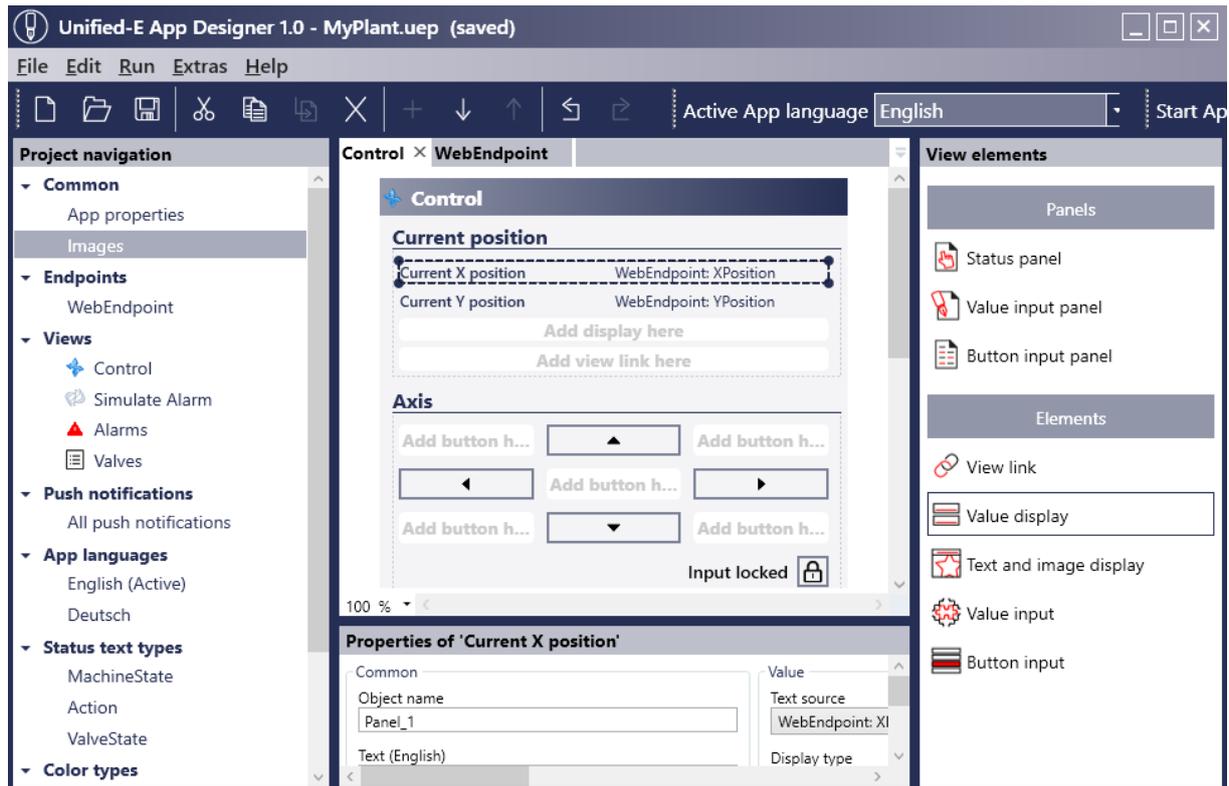
- Font size

Additionally, you can override the default appearance for each element in the properties section of the view editor.



4.2 Create a New View

The views of the operator app you create are listed in the project navigation in the “Views” section. In the context menu of “Views” a new view can be added by selecting “Add View”.

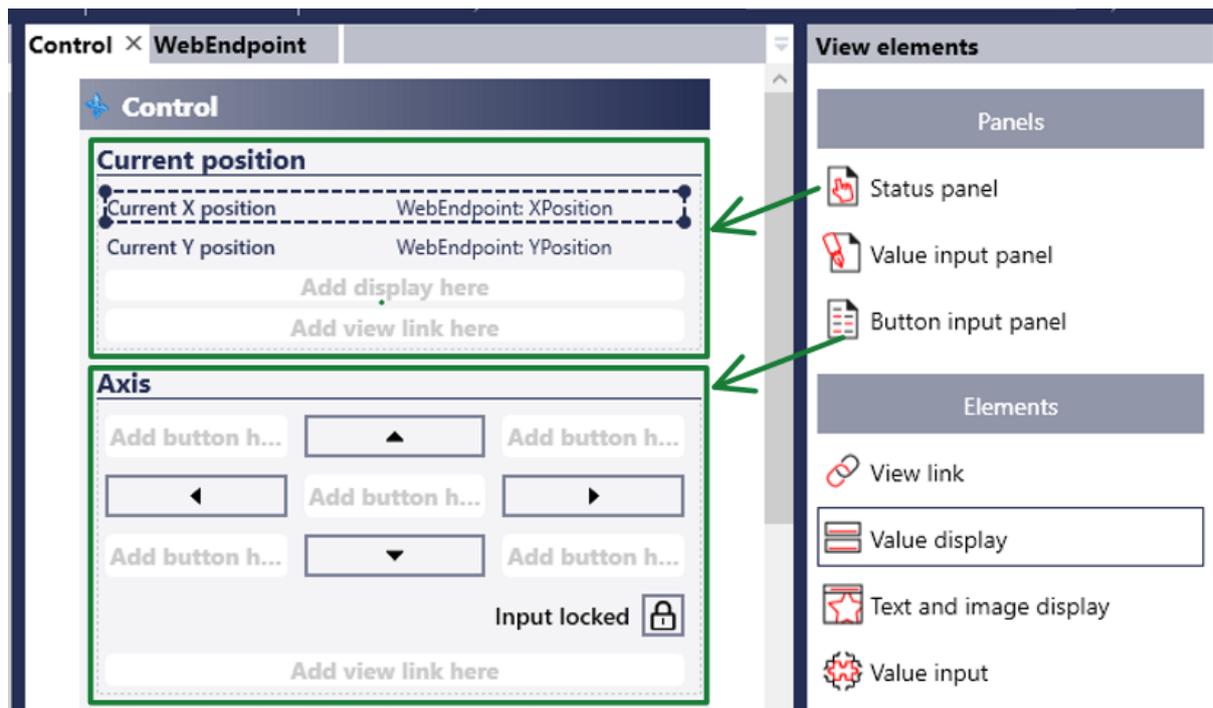


Start view editor:

By double-clicking on the views object in the project navigation the view editor will be opened. There, the view can be configured – description in Chapter 4.5.

4.3 View Structure

A view includes view elements: Panels and elements. Only panels can be placed directly in a view. Panels in turn contain elements. It is not possible to place an element directly in a view. Panels define a logical group of panel elements. For example, a chart series may only be placed into a chart panel. The elements are dragged with the mouse to the corresponding placeholders in the view.



The different view elements are described in more detail below.

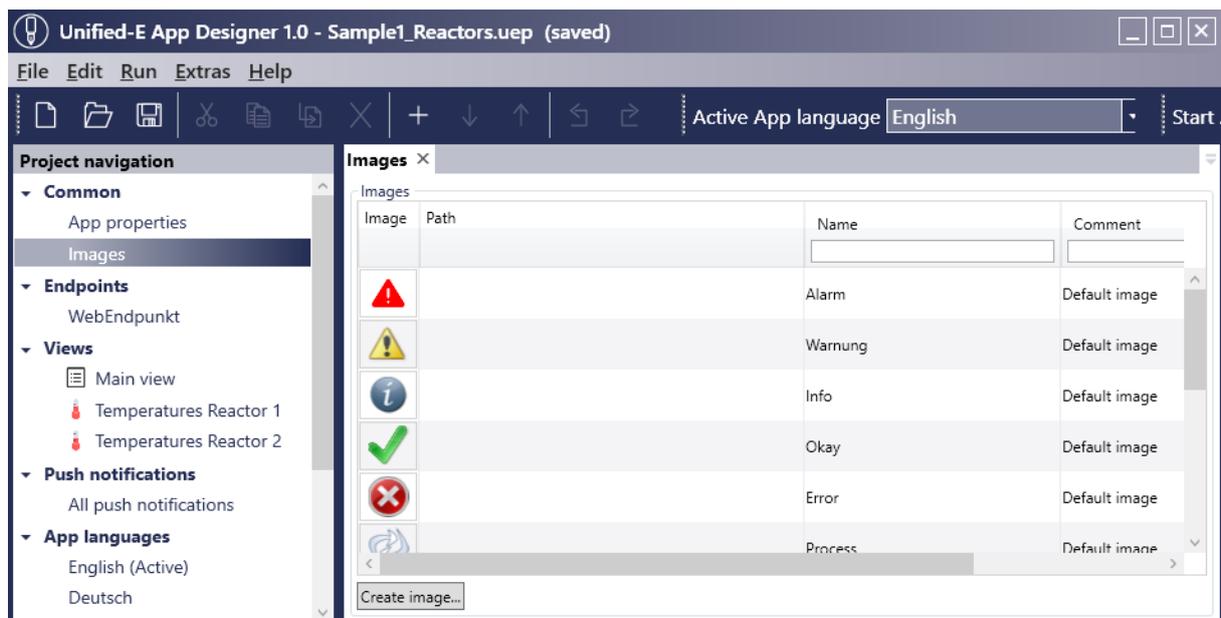
4.4 Add Images

Certain view elements can display images. All images have to be added in the images editor first. View elements and image types (described below) reference these image objects. Images will be saved directly in the App project file.

Recommended image format:

Image format: SVG (vector graphics), PNG or JPEG

Size for icons: 48x48



Tip:

You can also find many free and commercially usable images at www.iconfinder.com.

4.5 Create and Use Types

4.5.1 General

Types are used in the view elements to define the visualization (e. g. frame color) depending on a datapoint value.

Each type selection requires to select a datapoint which is used for value comparison.

What is a "Type":

A type defines visualization at a certain datapoint value.

There are three type categories:

- Status text types: Defines the text that should be displayed at a certain datapoint value (or value range).
- Color types: Defines the color that should be used for a certain datapoint value (or value range).
- Image types: Defines the image that should be displayed at a certain datapoint value.

Create a new type:

You will have to select the according category in the project navigation (e. g. "Color types").

There you can select "Add <Type>" in the context menu to create a new type.

4.5.2 Use Types

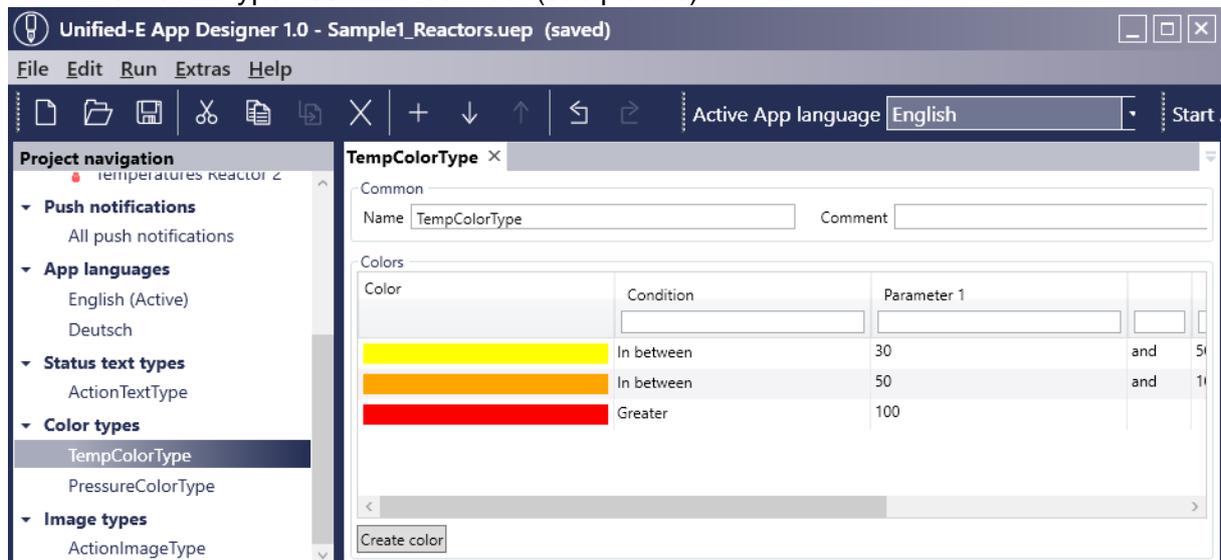
When configuring the view elements the types you created will be used in the "Properties" window when configuring view elements. When setting a type there always an associated datapoint has to be set. The datapoint value then will be used to choose the color, for example.

Example:

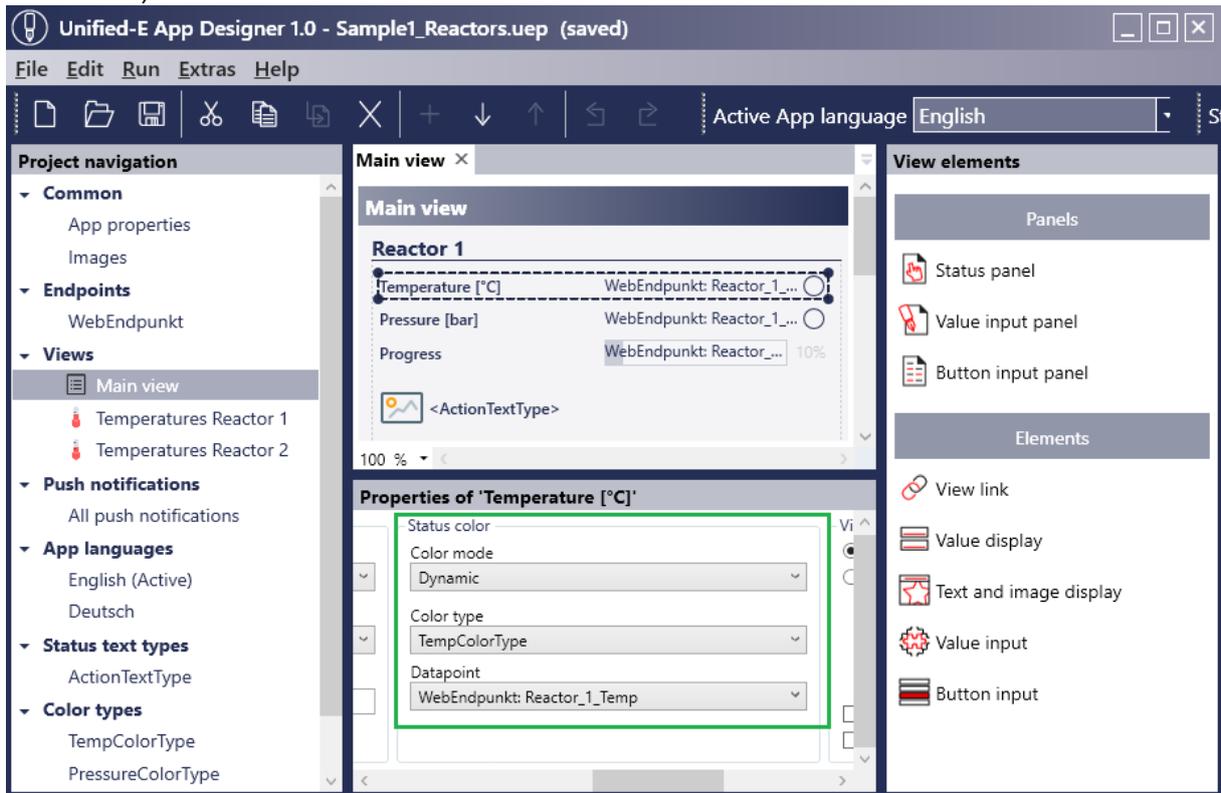
The temperature display should visualize status color "red" as soon as the value has exceeded 100°C.

Procedure:

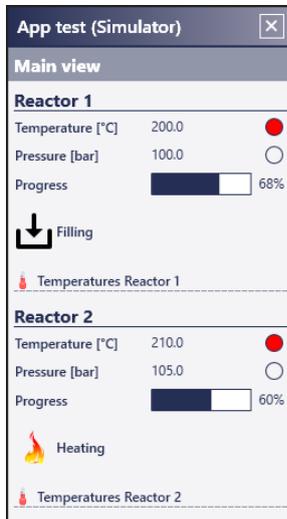
1. A new color type has to be defined. (see picture)



- Set the color type and the comparative datapoint in the properties (see green frame area below).



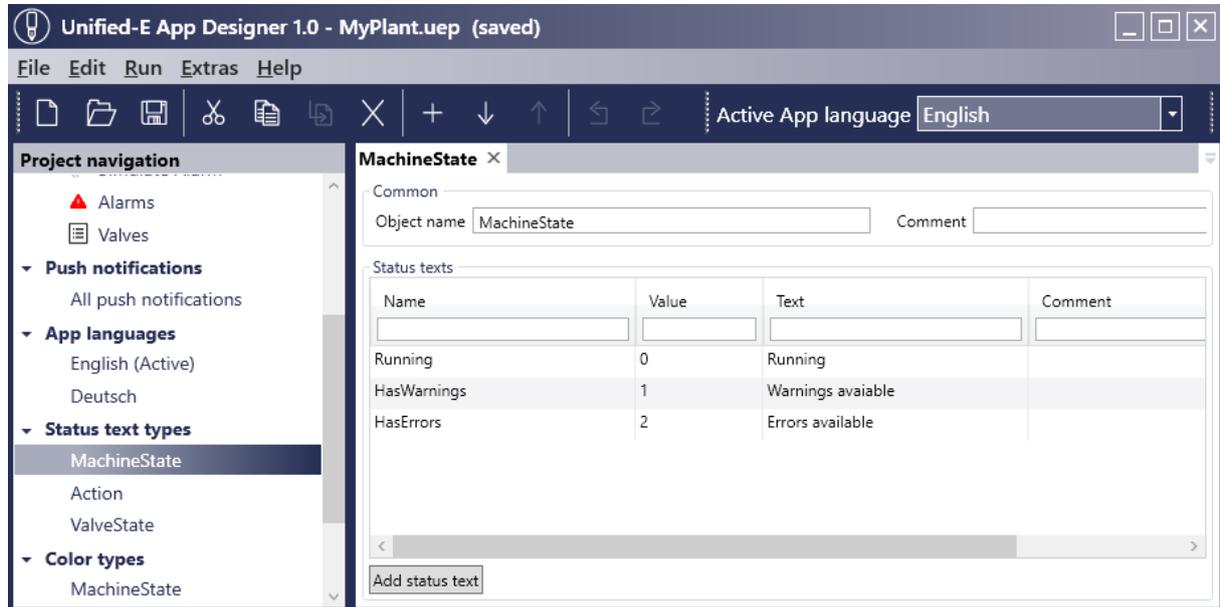
Result:



4.5.3 Type Categories

4.5.3.1 Status Text Types

The status text type defines the multilingual text at a particular datapoint value.

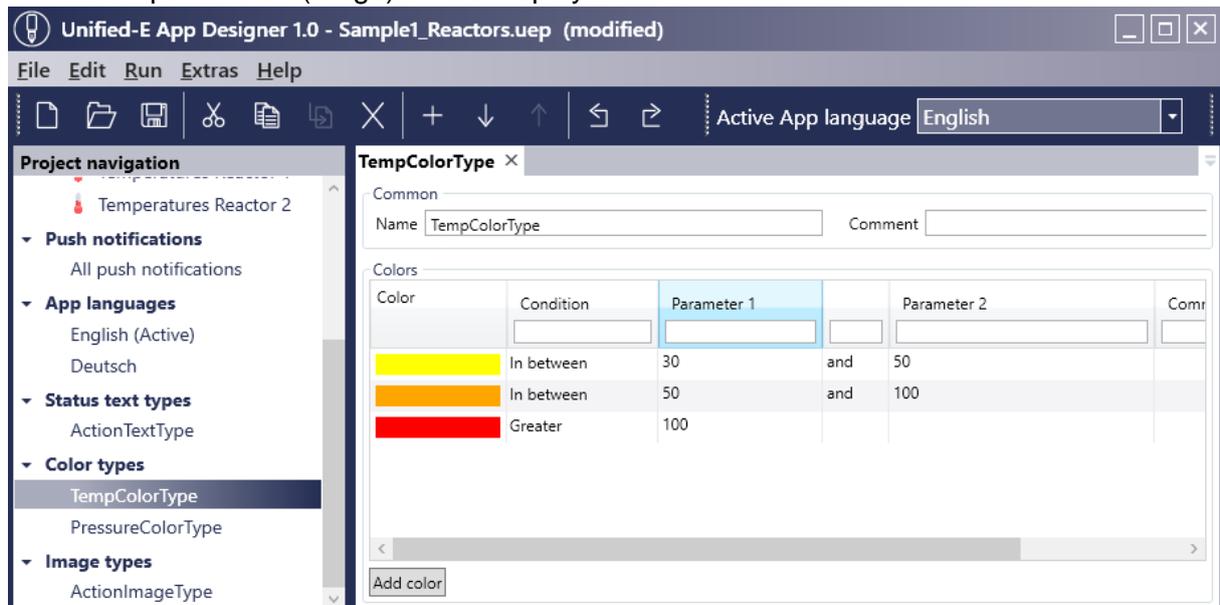


“Value” column:

Contains a numeric value. If this value matches the datapoint value, the text from the “Text” column will be used.

4.5.3.1 Color Types

There are conditions registered for a color type. The conditions determine which color for which datapoint value (range) will be displayed.



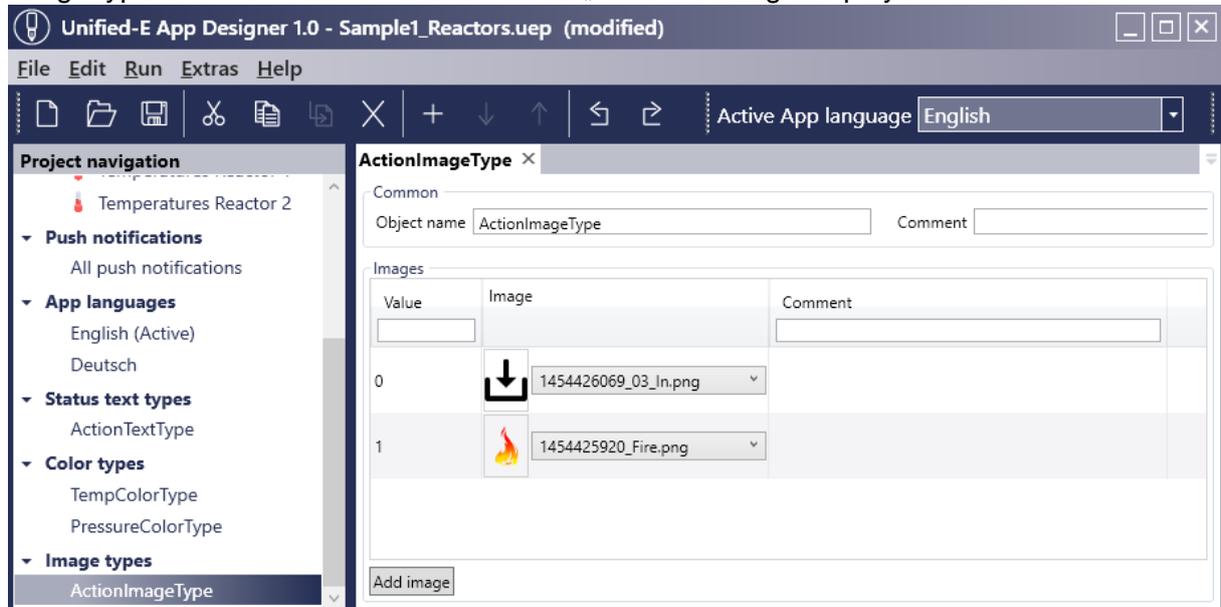
Example (see picture above):

The color “orange” is to be used when “within range 50 and 100”. Which means, if the stored datapoint value has the value ≥ 50 and ≤ 100 , the “orange” color will be used.

If there isn't any color defined for a datapoint value “no color” will be used.

4.5.3.2 Image Types

By using the image type an image can be displayed depending on the datapoint value. The image type will be used in the view element „Text and image display“.



Hint:

The images used in the image type are linked to pictures in the image editor.

“Value” column:

Contains a numeric value. If this value matches the datapoint value an image from the “Image” column will be used.

4.6 View Editor

In the view editor a view is created or edited.

The following elements can be selected in a view:

- The view itself (click on the view name)
- View elements (panels and elements)
- Placeholders for adding new view elements

Add view elements:

View elements can be dragged with the mouse on placeholders (e. g. “Add display here”) from the “View elements” area on the right.

4.6.1.1 General Property Groups

Each object in the view editor (view, view element) has properties. Some properties, such as the object name, can be found on each object.

You can find property groups that are present in almost all objects listed below.

4.6.1.1.1 „Common“ Property Group

Object name:

An object's name (must be unique within the group).

Comment:

An optional comment about the element.

4.6.1.1.2 "Description" Property Group

This property group is present on all view elements with a language-dependent name.

Display type:

“None” display type:

No text will be displayed.

"Text" display type:

A static text will be displayed. The text has to be entered in the "Display text" field.

Display text:

The display text is multilingual. The active language can be set in the toolbar under “Active App Language”.

„Status text“ display type:

Depending on status text type and datapoint the associated text will be displayed.

Datapoint:

The datapoint, of which the value determines the status text.

Status text type:

The status text type must be set in the selection list.

„Text with datapoint parameters“ display type:

The text, which should be displayed, has to be set in in the „Display text“ field.

Display text:

The display text is multilingual. The string „{0}“ has to be used as placeholder for the datapoint parameter with index 0. The active language can be set in the toolbar under “Active App Language”.

Table "Datapoint parameters*":

Here the datapoint parameters for the placeholders are configured.

First column: Contains the data point index.

Second column: Contains both the reference to the datapoint and the format of how the value is to be formatted.

Format:

For numeric datapoint values, the format for the decimal places is set; for date/time values, the display for the date/time is specified (also see chapter 4.6.1.1.11).

Horizontal Alignment:

The alignment of the text within the object.

Possible values: Left, Right, Center

Allow line break:

If the check box is set and the text cannot be displayed in the available width, then the text will be displayed in multiple lines.

4.6.1.1.3 “Image” Property Group

The image mode defines, which image will be displayed.

Image mode:

“None” image mode:

No image will be displayed.

“Static” image mode:

A static image will be displayed, which has to be selected in the “Static image” selection list.

Image position:

You must select the image position “Left” or “Right” in the selection list.

“Dynamic” image mode:

Depending on the image type and datapoint value the according image will be displayed.

Datapoint:

The datapoint, of which the value determines the image.

Image type:

The image type must be selected.

“View image” image mode:

The image of the linked view is used.

Image position and image size:

Image position:

You must select the image position “Left” or “Right” in the selection list.

Width (px):

The width in pixels reserved for display.

Height (px):

The height in pixels reserved for display.

Distance from text (px):

The distance of the image from the text (if any).

4.6.1.1.4 „Visibility“ Property Group

In this group you decide whether the view element should always be visible or only if a certain condition is met. Furthermore you can set here, whether a paragraph should be added before or after an element.

Properties:

„Always visible“ button:

If this button is set, the element will always be visible.

“Visible on condition” button:

If this button is set, a condition can be set that controls visibility of the element.

Datapoint:

The value of the selected datapoint is used to verify if the condition is satisfied.

Condition:

The specific compare condition, e. g. "Equals", "Greater".

Parameter 1:

The first parameter for the condition.

Parameter 2:

The second parameter for the condition.

„Has paragraph before“ checkbox:

If this checkbox is set a paragraph will be set before the element. This, for example, is to group elements logically within a paragraph.

„Has paragraph after“ checkbox:

If this checkbox is set a paragraph will be set after the element.

4.6.1.1.5 "Appearance" Property Group

The default appearance can be overridden which is defined in the default appearance editor.

For a view element, there are one or more sub sections each of them with a specific "Appearance" property group.

For colors, you can select the modes as follows:

Mode "Default":

No changes are applied the element appears as defined in the default appearance.

Mode "Individual":

A specific, fix color can be applied.

Mode "Dynamic":

The color which should be applied depends from a datapoint value. You have to select both a datapoint and color type.

Example: A text should be colored "red" if the machine is in alarm state.

4.6.1.1.6 "Access Control" Property Group

In this group, access protection is activated so that this object is only displayed for granted users (or user roles).

Properties:

"Activate access control" checkbox:

If set, the object is displayed only for granted users / user roles.

"Granted user roles" table:

The permissions are to be configured there. The object is only displayed for user roles with the checkbox set.

4.6.1.1.7 "Layout of elements" Property Group

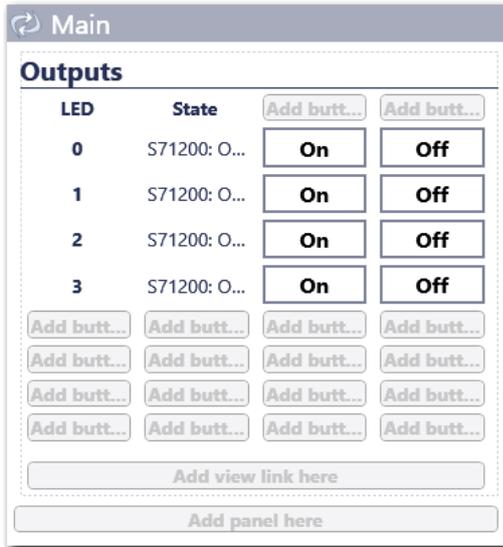
You can configure the layout of the elements in the following panels:

- Standard panel
- List panel (for layouting of a list row)

Layout type:

"Grid" layout:

Layouting is done within a grid layout. Each element must be placed on a grid cell. The column width can be configured in the "Layout" property group so an element might occupy multiple cells.



The number of columns and rows can be configured in the "Layout" property group of the element.

Furthermore, the sizes of the individual columns and rows can be configured individually.

Type: Fixed pixel length:

The size of the row or column is predefined by entering the logical pixel length.

Type: Automatic:

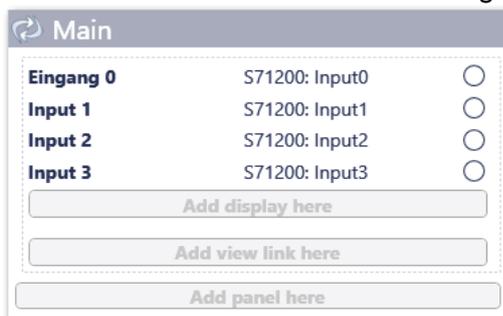
The size is automatically determined by the size of the largest element.

Type: Fill:

The remaining space in the panel, if available, will be filled out.

"Flow" layout:

The elements are stacked in one single column.



Element spacing:

Element spacing above/below (px):

Describes the vertical distance of the elements in the panel.

Element spacing left/right (px):

Describes the horizontal distance of the elements in the panel.

Distances can also be defined in the element properties "Borders & margins".

4.6.1.1.8 "Position & size within panel" Property Group

Span within grid:

These properties are only visible if "Raster" has been selected in the Layout type property of the panel. It is necessary to specify how many rows and columns the element should occupy in the grid.

Span (column count):

Number of occupied columns for the element.

Span (row count):

Number of occupied rows for the element.

Fixed size:

Normally, the element sizes are automatically calculated and no further sizes are required. On request, however, fixed sizes can be assigned here.

Width (px) (optional):

The element width in pixels.

Horizontal alignment:

The horizontal orientation of the element if more space has been allocated in the panel.

Height (px) (optional):

The element height in pixels.

Vertical alignment:

The vertical orientation of the element if more space is available in the panel.

4.6.1.1.9 "Borders & margins" Property Group

This allows individual distances and borders for the element or panel.

Vertical alignment:

Sets the vertical alignment of the content. This property is only applied if the element is larger than its content.

Border line visibility:

This lets you show or hide border lines around the element.

Outer margins:

Defines the distances / indents of an element.

Inner margins:

Defines the distances between element border and content.

Border style:

Radius Corners (px):

Defines how the corners of the frame should be rounded.

Thickness (px):

Sets the borderline thickness.

4.6.1.1.10 "Positioning in view (display)" Property Group

Here the position and size of a panel can be determined display-specific. The displayed values and entries always apply to the currently set active display.

The configuration is analogous to the positioning of elements in the range, see also 4.6.1.1.8.

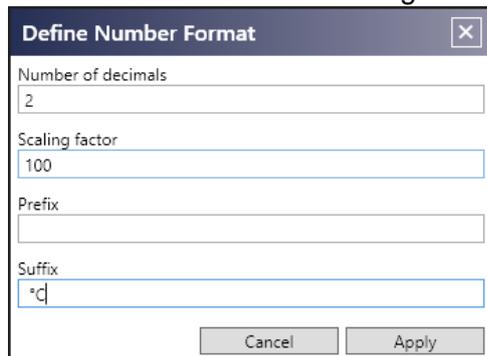
4.6.1.1.11 Format Property

Some elements have a section for formatting and scaling numeric values or for formatting the date/time value.

Number format:

You can configure the number format in a dialog. Select "..." button next to the format field for this.

"Define Number Format" dialog:



The app displays the datapoint value multiplied with the scaling factor.

Example:

The datapoint returns the value "0.2".

The displayed value is "20.00 °C".

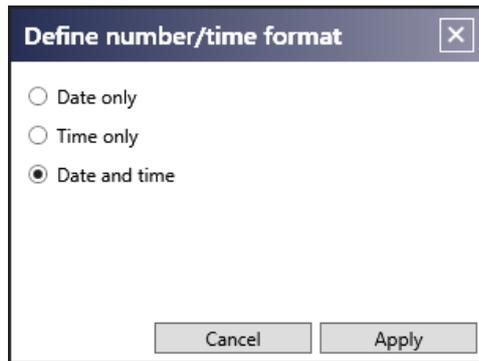
Depending on a smartphone's country settings the corresponding decimal separator is used.

Date/Time:

You can use date/time values with datapoints of type "Text". The endpoint always handles the text/string as follows:

- The date/time value must always contain both the date and time part. The date/time must be provided as UTC time.
- The date/time must be provided in the following format:
MM/dd/yyyy HH:mm:ss

"Define Date/Time" dialog:



Depending on the country settings of the smartphone, the date or time is displayed accordingly.

4.6.2 View

A view contains arbitrary panels. By clicking the view name, you will receive the properties of the view in the properties window.

4.6.2.1 "Display Options" Property Group

"Is start view" checkbox:

If the checkbox is set, this view will be the home view of the operator app and will be displayed first. This property can be set individually for each display. For example, you can set another start view for a smartphone then for a tablet.

"Hide view title" checkbox:

If the checkbox is set, this view title will not be displayed.

"Disable periodic update of datapoint values. The view is automatically updated after user input:

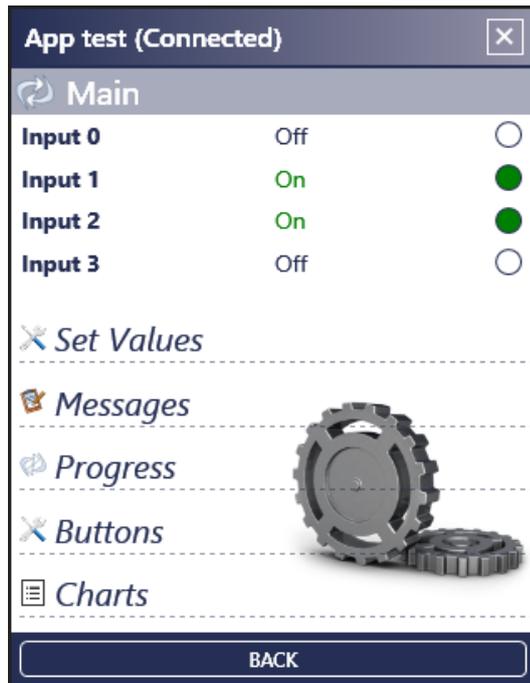
If checked, the current datapoint values for the view are only read when displayed or after a user input. Otherwise, the data point values will be read periodically, which will increase the load on the endpoint and App Manager.

Image:

An image can be assigned to the view. This image is used in both the view title and the view link.

4.6.2.2 "Background image" Property Group

Example:



Static image:

Select the display image. The optimal resolution of the image depends on the displayed image width and the desired sharpness. To fill the full view width, the image should have a minimum pixel width of 500 pixels.

Horizontal alignment:

Possible values are left, center and right. This property is only applied if the image is smaller than the width of the view.

Vertical alignment:

Possible values are bottom, center, top.

Image width (%)

This value describes the image width in percent. 100% corresponds to the width of the view.

4.6.2.3 "Layout of panels (display)" Property Group

Here, the basic layout of the panels within a view for the active display is adjustable.

The active display can be set in the toolbar under "Active Display".

Layout type "Grid":

The areas are arranged in a grid (or in a table). Each area is to be placed in a table cell. The number of occupied columns can be adjusted in the panel property group "Display-specific layout", so that one panel can occupy several cells.

Furthermore, the sizes of the individual columns and rows can be configured individually.

Type: Fixed pixel length:

The size of the row or column is predefined by entering the logical pixel length.

Type: Automatic:

The size is automatically determined by the size of the largest element.

Type: Fill:

The remaining area on the display, if available, will be filled out.

Layout type "Flow":

The areas are arranged one below the other in a single column.

4.6.2.4 "Visible panels (display)" Property Group

Here you can hide panels that are optimized for other display groups.

4.6.2.5 "Initialize input datapoints before displaying" Property Group

Here, you can select datapoints that should be set to a specific value as soon as an operator opens the view.

The desired data points with the initial values are to be configured in a table.

"Datapoint" column: Reference to the datapoint to be reset when displaying.

"Value" column: Contains the reset value.

4.6.3 Panels

4.6.3.1 Standard Panel

The default panel allows you to configure individual view sections that are suitable for both input and display purposes.

Permitted elements:

- Display
- View link
- Value input
- Button input
- Image

4.6.3.1.1 "Button input lock" Property Group

By default, the operation of buttons is explicitly enabled by the user for security reasons. This is done by pressing the lock symbol.

Properties:

"hide element for button input lock" checkbox:

If set, then the lock symbol is not visible. A button operation does not have to be explicitly activated by the operator.

Margins:

This sets the edges of the button lock element (lock icon).

4.6.3.1.2 "Send input values" Property Group

Entered values can be sent immediately after input or by explicitly pressing the "Apply" button.

"Send trigger" selection list (optional):

The send trigger datapoint will be set on the trigger value after clicking the "Apply" button as soon as all values from the value input panel have been send. For instance, this trigger can be used in a PLC to read the new datapoint values consistently.

Trigger value:

Contains the trigger value that should be set when activating the send trigger.

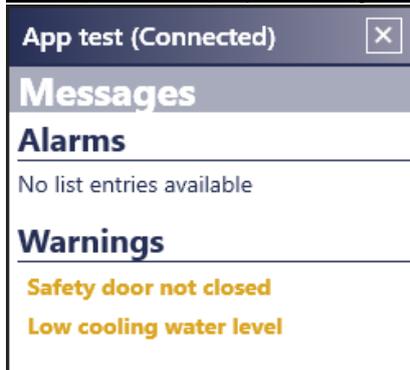
Margins:

This sets the margins of the "Apply" button.

4.6.3.2 List Panel

This panel contains a list. The list content is based on a datapoint of type "Data table".

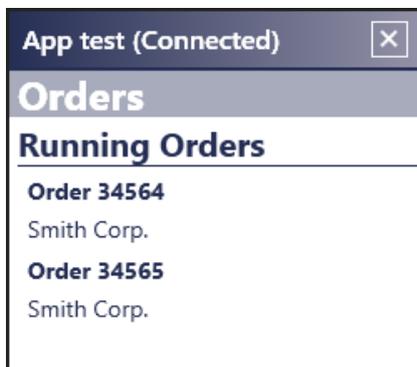
Example 1: Display messages of a controller:



If you want to set up a message list for a PLC you can find more details in chapter 12.

Example 2: Display running orders including details:

Lists with more than one value per entry are currently only supported by the WebHttp and Application endpoint adapter. These adapters allow data table datapoints with multiple columns.



4.6.3.2.1 Design the Template for List Entries

Each row in the data table of the datapoint associates with a list entry in the displayed list. A list entry can be configured by using one or multiple display elements (see chapter 4.6.4.3).

The display (e. g. the value) is not configured with a datapoint from an endpoint but with a column index (starting from 0) of the data table. In this way, you can also configure dynamic images and colors.

4.6.3.2.2 "List" Property Group

Datapoint (data table):

There you have to select the datapoint of the data type "Data table" whose contents are to be displayed in the list.

Max. number of entries at the start:

The max. number of list entries at the beginning. The operator has the option of scrolling or expanding the list (paging).

4.6.3.2.3 "Selection" Property Group

It is possible to configure to navigate into a detailed view when the user selects a list entry. This feature is only possible for endpoint adapters with user-specific datapoints (currently WebHttp and WindowsApplication adapter). The current selection is stored in a user-specific datapoint (see below).

Entry ID (column index):

The used column index of the datatable column is defined here which uniquely identifies the list entry.

Datapoint for selection transfer:

The user-specific datapoint which is set on selection with ID of the list entry (specified by "Entry ID (column index)").

Details view:

Defines the details view for the list entries. The details view typically uses user-specific datapoints since the displayed values should depend from the list selection (depending from the datapoint value specified in "Datapoint for selection transfer").

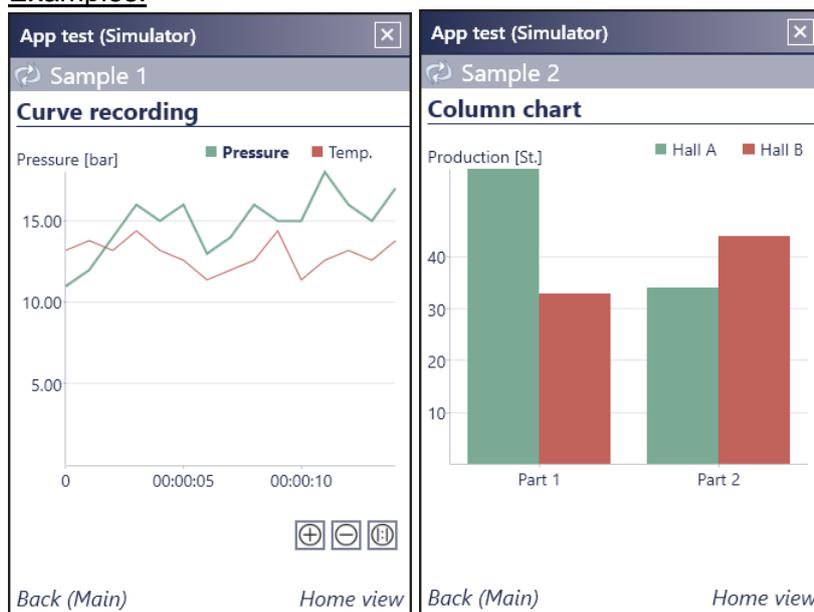
4.6.3.3 Chart Panel

In this panel, you can display one or more chart series within a chart area.

Supported chart types:

- Time chart
- Column chart
- Line chart
- Pie chart

Examples:



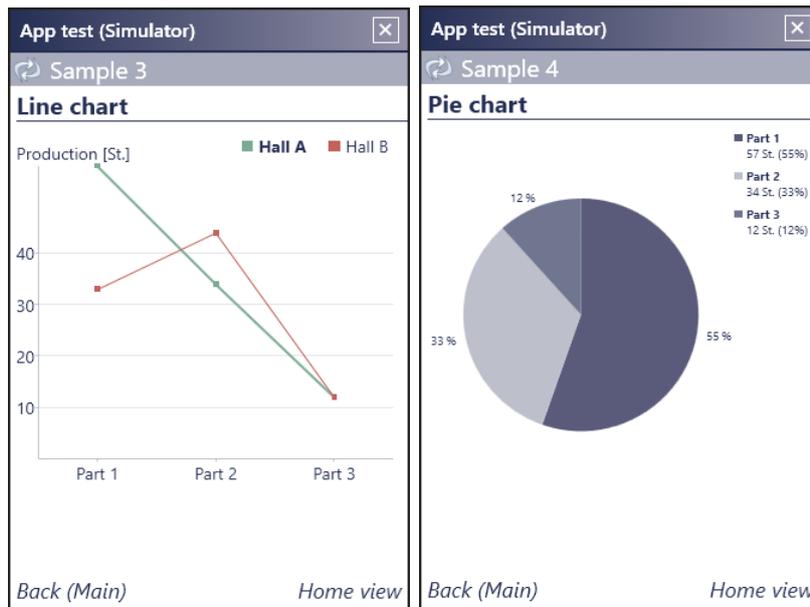


Table of values:

For visualization, the defined chart series are used. Each chart series has a table of values, e. g. x/y values for the time line. The table of values are provided from a datapoint with "data table" datatype.

The pie chart supports only one chart series. All other chart types support multiple chart series.

Permitted elements:

- View link
- Chart series

Hint:

To completely configure the chart, you must set the properties of the chart series as well (see chapter Chart Series4.6.4.7).

4.6.3.3.1 "Chart" Property Group

In this group, you define the chart type.

"Chart type" dropdown list:

Select the chart type here.

"Time axis" dropdown list:

Selection is only possible for time axis.

Absolute date/time values: The time axis displays the absolute date/time values.

Numerical values: The time axis displays numerical time values, this is elapsed time since start of recording.

"Define custom colors for categories" checkbox:

Only possible for column charts with a single series. If set you can define individual colors for each column bar.

"Display category names in legend" checkbox:

Only possible for column charts with a single series. If set the column bar names are displayed in a legend box. This is useful for long column bar names to avoid text overlapping.

4.6.3.3.2 "Time axis (X axis)" Property Group

This group is only visible for time charts.

Configuration mode:

Static configuration: The properties are set to constant values in the Unified-E App Designer.

Dynamic configuration with datapoints: The properties are set by means of datapoints at runtime.

Unit:

The unit of the values. This is only supported for "Numerical values" time axis.

Number/time format:

Describes, how the numerical time value should be formatted. If you choose "{T}" for time format, it is assumed that the numerical value is in seconds. The value is then displayed like "hh:mm:ss". This is only supported for "Numerical values" time axis.

Min. range:

The minimum range, which should be displayed. If you use an "absolute date/time" time axis, the range is in hours.

Max. range:

The maximum range, which should be displayed. If you use an "absolute date/time" time axis, the range is in hours.

4.6.3.3.3 "Categories" Property Group

This group is not applicable for time charts.

Configuration mode:

Static configuration: The properties are set to constant values in the Unified-E App Designer.

Dynamic configuration with datapoints: All categories data must be provided with the table of values of the chart series.

Categories table:

This table is only visible for static configuration mode.

Column and line chart:

Categories are the X axis texts. A category consists of a name and a multi-lingual display text.

Pie chart:

There is a category for each pie segment. The category consists of a name, color and a multi-lingual display text.

4.6.3.3.4 "Common Y Axis" Property Group

Line and column charts have only one common Y axis description which is applicable for all chart series. This does not apply for time charts, there each chart series has its own Y axis description.

Configuration mode:

Static configuration: The properties are set to constant values in the Unified-E App Designer.

Dynamic configuration with datapoints: The properties are set by means of datapoints at runtime.

Display text:

The multi-lingual display text for the Y axis.

Unit:

The unit of the Y values.

Number format:

Defines the format, which should be used for displaying the values (see also Chapter 4.6.1.1.11).

Minimum:

The minimum value for the Y axis. If not set the smallest value from the table of values will be used.

Maximum:

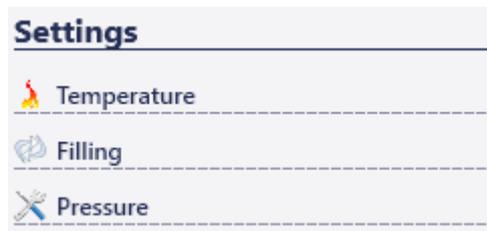
The maximum value for the Y axis. If not set the largest value from the table of values will be used.

4.6.4 Elements

4.6.4.1 View link

The view link can be compared with a hyperlink. By clicking on this link, the linked view opens.

Example: Multiple Links in one display panel:



4.6.4.1.1 "Display options" Property Group

Horizontal alignment:

Describes the horizontal alignment of the content (text and image).

"Underline view link" checkbox:

If set, the text is underlined with dashed lines.

4.6.4.1.2 "Link" Property Group

Linked view:

The view that opens when clicking on the link.

4.6.4.2 Hyperlink

The hyperlink can be used to create a link to an external website. The URL of the website can be configured statically or dynamically via a datapoint.

4.6.4.2.1 "Display options" Property Group

Horizontal alignment:

Describes the horizontal alignment of the content (text and image).

"Underline hyperlink" checkbox:

If set, the text is underlined with dashed lines.

4.6.4.2.2 "Web address" Property Group

Here the URL is configured.

Option "Static configuration":

The field "Address (URL)" has a predefined URL. This website is opened by clicking on the hyperlink in the standard browser.

Option "Dynamic configuration with datapoints":

The URL is read from a datapoint of type "Text". The data point is to be selected in the field "Address (URL)".

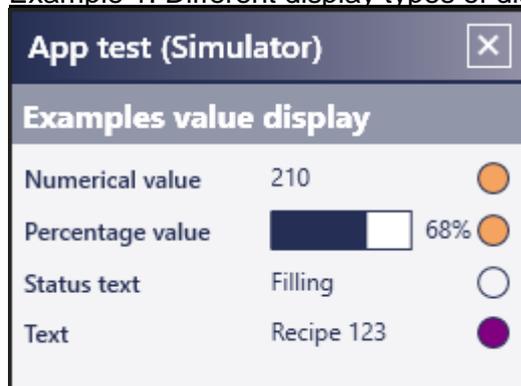
4.6.4.3 Display

The display is used to visualize a datapoint's value.

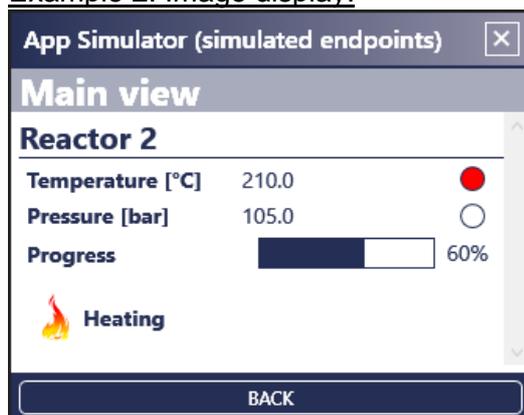
A display can contain up to four elements. All elements can be hidden, if desired:

- Description: The multilingual text or some text based on a datapoint value
- Value: Displays the value of a datapoint
- Image: Displays an image which can be dynamic
- Status indicator: A circular symbol which can be filled with a dynamic color

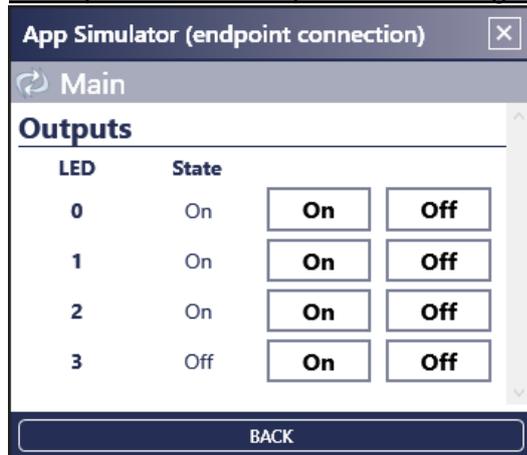
Example 1: Different display types of display:



Example 2: Image display:



Example 3: Standard panel containing display elements (raster display):



4.6.4.3.1 “Value” Property Group

Display type:

Defines how the value will be displayed. Depending on the display type further properties will appear.

“None” display type:

The value is hidden.

“Text” display type:

The datapoint's value will be displayed directly.

“Numeric value” display type:

By selecting this type a numeric value can be displayed. The output format can be edited in the “output” column.

Format:

Defines, how many decimals will be shown and if the datapoint value should be scaled. (see Chapter 4.6.1.1.11)

Format datapoint:

This field is only visible if you selected "Dynamic configuration with datapoints". The selected text datapoint provides the format.

„Status text“ display type:

Depending on the datapoint value and status text the according text will be displayed.

Status text type properties:

The status text type has to be set in the selection list.

“Percentage value” display type:

Datapoint value will be shown in percentage.

“Date/Time” display type:

The datapoint value is used for date/time display.

Format:

Defines which date/time parts should be used for display (see also Chapter 4.6.1.1.11).

Format datapoint:

This field is only visible if you selected "Dynamic configuration with datapoints". The selected datapoint provides the format.

Datapoint:

Sets the datapoint of which the value should be visualized.

Width (%):

Sets the width of the value representation. The smaller this value, the more space is available for the name.

Horizontal Alignment:

The alignment of the text within the object.

Possible values: Left, Right, Center

4.6.4.3.2 "Status Color" Property Group

In addition to the value a status color (colored circle) can be displayed.

"None" color mode:

No status color will be displayed.

"Static" color mode:

It is always a fixed status color displayed. The color mode has to be set in the selection list.

"Dynamic" color mode:

The status color will be selected and shown depending on the color type and datapoint value.

Color type:

An existing color type needs to be selected.

Datapoint:

The datapoint, of which the value determines the color.

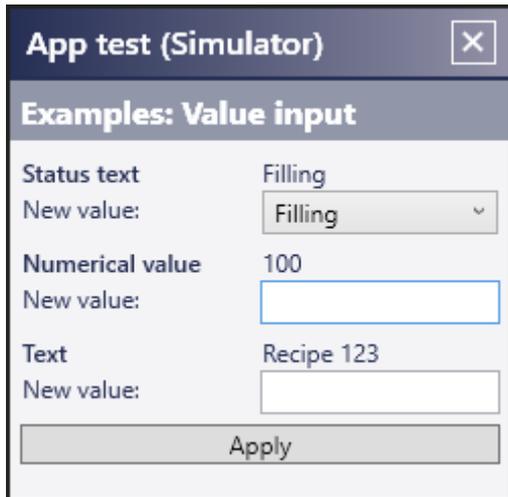
4.6.4.4 Value Input

4.6.4.4.1 General

With the "Value input" element datapoint values will be set. That is how the endpoint can be controlled e. g. a machine or plant.

Example:

Different input types for value input:



The screenshot shows a window titled "App test (Simulator)" with a close button. Below the title bar is a section titled "Examples: Value input". This section contains three input groups:

- Status text:** The current value is "Filling". Below it, "New value:" is followed by a dropdown menu currently showing "Filling".
- Numerical value:** The current value is "100". Below it, "New value:" is followed by an empty text input field.
- Text:** The current value is "Recipe 123". Below it, "New value:" is followed by an empty text input field.

At the bottom of the form is a button labeled "Apply".

4.6.4.4.2 "Input" Property Group

"Numeric value" input type:

This input type allows you to enter numeric values and the specification, in which range (minimum/maximum) the input value must be located.

Format:

Defines, how many decimals will be shown and if the datapoint value should be scaled (see also Chapter 4.6.1.1.11).

"Range of values" selection list:

Fix range of values: The minimum and maximum values are defined with constant values.

Range of values from datapoints: The minimum and maximum values are retrieved from the selected datapoints.

Minimum value:

When this value is set, it is ensured that the input value of the operator is not going to be smaller than the minimum value.

Maximum value:

When this value is set, it is ensured that the input value of the operator is not going to be greater than the maximum value.

"Text" input type:

You can do text/string inputs.

Maximum length:

Specifies the maximum number of characters that can be entered in the text field.

"Date/Time" input type:

The user can enter a date and/or time value.

Format:

Defines if the user should enter a date and/or the time value (see also Chapter 4.6.1.1.11).

Input type "Slider":

Numeric values can be entered via a slider control.

This input type has identical configuration options as the "numeric value" input type, see above.

„Dropdown list (dynamic)“ input type:

This allows the dropdown list to be filled based on a data table datapoint.

List (data table):

The table datapoint is to be selected. The table can have 1 or 2 columns.

Table with a single column: The column represents both the display text and the text which should be set on selection.

Table with two columns: The first column is the ID and is set to the input datapoint, the second column is the corresponding display text.

„Dropdown list (status texts)“ input type:

By selecting the status text, the operator indirectly selects the value in a selection list.

Status text type:

The status text type has to be selected.

Input datapoint:

The datapoint for which the new value has to be set needs to be selected. In the selection list only datapoints with access "Write; Read" are listed.

4.6.4.4.3 "Input options" Property Group

Input mode:

"Keyboard" mode: The user input is done via soft keyboard.

"Barcode scanner" mode: The user input is done by scanning in a barcode or QR code.

The following codes are currently supported:

- AZTEC, CODABAR, DATA-MATRIX, MSI, PLESSEY, ITF, Maxi-Code
- Code 39, 93, 128
- EAN 8, 13
- QR-Code
- UPC A, UPC E, UPC EAN Extension
- PDF 417
- RSS 14, RSS Expanded

Others:

"Initialize input field with current value" checkbox:

If set, then the input field will be preassigned with the current value when opening the view.

"Hide current value" checkbox:

If set, the current value of a datapoint will not be displayed.

"Use full line for input" checkbox:

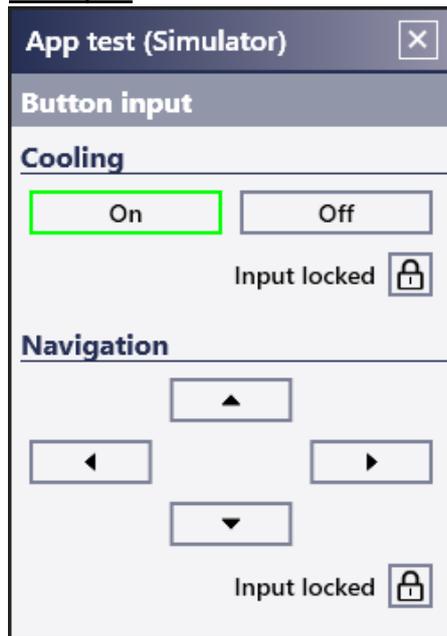
If set the text "New value" will not be displayed at the input field but the input field uses the full line for user input.

4.6.4.5 Button Input

With the "button input" element datapoints can be set – both by pressing and releasing.

The button input element can display a description and an image.

Example:



4.6.4.5.1 "Display options" Property Group

Horizontal alignment:

Describes the horizontal alignment of the content (text or image) within the button.

4.6.4.5.2 "Press Action" Property Group

This group will determine which datapoint value will be set once by pressing. In addition, an "Activate Alive trigger" can be set cyclically. For example, after the timeout period the PLC program could trigger an alarm or go in a safe state (e. g. stop the axis movement).

Datapoint:

The datapoint that needs to be set. (To make sure your datapoint is listed, it must be configured for access "Write, Read" in the datapoint table.)

Setpoint value:

The value that will be set when pressing.

"Alive trigger" property:

"Activate Alive trigger" checkbox:

When checkbox is set it can be configured, which datapoint should be set periodically while the operator is pressing (holding down) the button.

Datapoint:

The datapoint that should be used as alive trigger.

Setpoint value:

The trigger value that should be set periodically.

Trigger period [ms]:

The wait time between setting the trigger value.

4.6.4.5.3 "Released Action" Property Group

In this group you have to determine which datapoint value will be set by releasing the button.

“Activate release action” checkbox:

When this checkbox is set release action will be activated and you can configure the set point for release.

Datapoint:

The datapoint that needs to be set when releasing.

Setpoint value:

The value that needs to be set on the datapoint when releasing.

4.6.4.6 Image

"Image alignment" property group:

Horizontal alignment:

Describes the horizontal image orientation within the element.

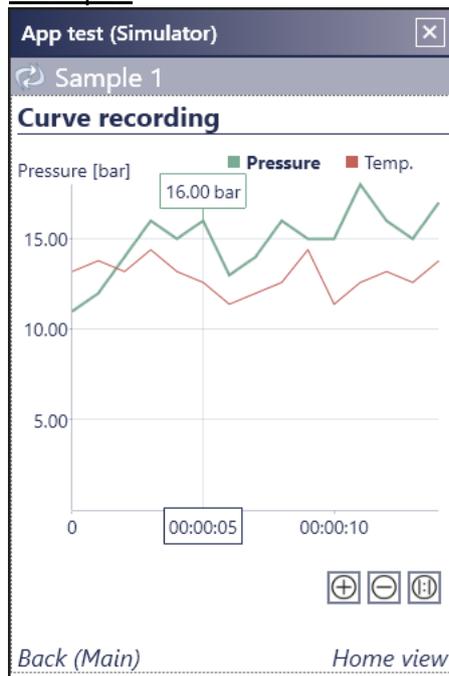
Image width (%):

Describes how much of the width should be used for the display (in percentage).

4.6.4.7 Chart Series

A chart can display multiple chart series.

Example:



The time chart in the example displays two chart series.

Chart series "Pressure":

This chart series contains all data for displaying the "Pressure" time line. The "Pressure" series is selected so the Y axis of this chart series is displayed.

Chart series "Temp.":

This chart series contains all data for displaying the "Temp." time line.

When the user clicks the legend (series name) then the associated time line is highlighted and the Y axis of the clicked series is displayed (see image above).

4.6.4.7.1 "Table of Values" Property Group

In this group, you must select the datapoint which provides the table of values. The datapoint must be of data type "data table".

The table of values contains the values for visualizing the chart series, e. g. timeline or column heights. Depending on the properties and chart type the structure of the table of values varies.

Examples:

Table of values with columns "date/time", "value":

This table of values is used for time charts with absolute time values for the X axis. Only WebHttp, OPC-UA and WindowsApplication adapters can provide such a data table.

However, you can define a curve recording datapoint for a measurement value. Recording is done in the Unified-E App Manager after app installation. So, curve recording is possible for all endpoints (see chapter 13).

Table of values with a single column "value":

This table of values is used for line, column and pie charts when the categories are predefined. This table type is supported by all endpoint adapters (ARRAY data type). You will find more details in the documentation of the specific endpoint adapter.

Limitations:

Transferring the table of values from the Unified-E App Manager to the smartphone causes a higher traffic than transferring only simple values.

Therefore, only a limited number of rows is transferred:

- "Firewall-friendly" communication mode (relay messaging): max 1000 rows
- All other communication modes: max. 5000 rows

4.6.4.7.2 "Display Options" Property Group

Here you configure common display options for the chart series.

Configuration mode:

Static configuration: The properties are set to constant values in the Unified-E App Designer.

Dynamic configuration with datapoints: The properties are set by means of datapoints at runtime.

Color:

The color for the columns or line.

Unit:

The unit of the values in the table of values. This property is only supported for pie charts.

Number format:

Defines the format of the pie chart values values (see also Chapter 4.6.1.1.11). This property is only supported for pie charts.

4.6.4.7.3 "Y axis" Property Group

This property group is only active for time charts. There each chart series describes the Y axis individually. The chart displays the Y axis of the selected chart series.

The properties of the group are identical with the group described in chapter 4.6.3.3.4. Exception: For the Y axis text the chart series "Description" group is used instead of the display text.

5 Configure Push Notifications

5.1 General

Push notifications are messages that will be sent to the operator app on the smartphone as soon as a specific event occurs.

By activating (pushing) the push notification on the smartphone the linked view opens in the operator app.

Push notification examples:

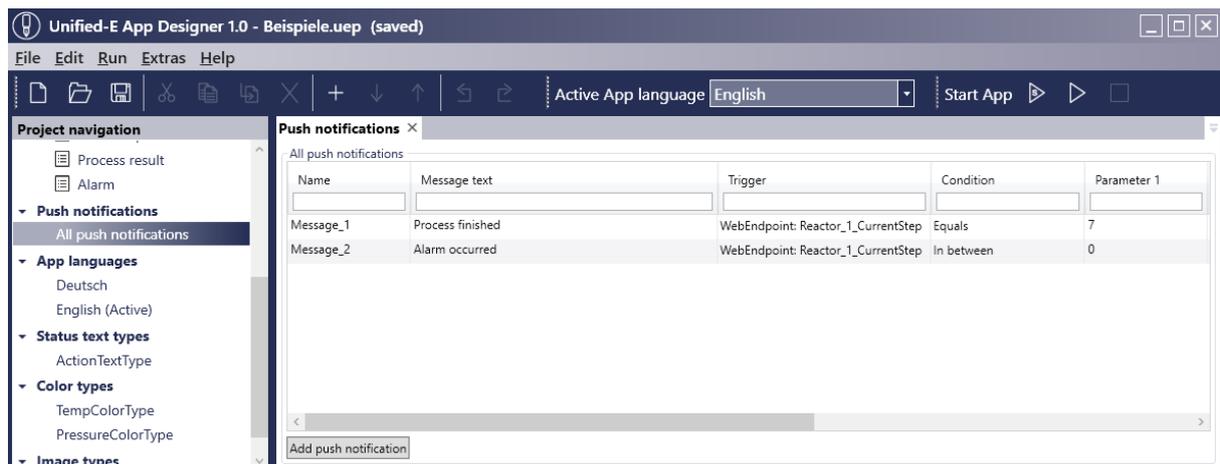
- Alarm occurred
- Machine maintenance necessary
- Process completed
- The order in the ERP has reached status "X"

Push message elements:

- Condition: Defines when the message will be triggered
- Message text: Defines the message which should be displayed
- Linked view: Defines which view should be started when the push notification is being activated

Start push notification editor:

The push notification editor has to be started by double-clicking "All push notifications" in the project navigation.



5.2 Add Push Notifications

By pressing the "Add push notification" button a new push notification is created and added to the table.

5.3 Push Notifications Table

Push notifications ×

All push notifications

Name	Message text	Trigger	Condition	Parameter 1	Parameter 2	Affected view	Comment
Message_1	Process finished	WebEndpoint: Reactor_1_CurrentStep	Equals	7		Process result	
Message_2	Alarm occurred	WebEndpoint: Reactor_1_CurrentStep	In between	0	and 8	Alarm	

Name:

The push notification object's name. This name is used as ID in the language table, to be able to easily identify the multilingual text.

Message text:

The multilingual text message, as seen by the app user.

Trigger:

The trigger datapoint, of which the value is used for the trigger condition. The trigger conditions (defines when a message text will be sent) have to be set in the columns "Condition", "Parameter 1", and "Parameter 2".

Condition:

Describes the trigger condition. The associated parameters have to be defined in the following columns.

Parameter 1:

The first parameter for the condition.

Parameter 2:

The second parameter for the condition, if necessary.

Linked view:

The view which will be displayed when the app user activates the message text.

Access control:

Describes for which user roles the push notification should be sent if access control is enabled.

Comment:

An optional comment, which is only visible in the push notification table.

5.4 Further Push Notification properties

The further properties of a push notification can be set in the properties area below the table.

Message text:

Display type "Text":

A static, multilingual text is displayed. This must be entered in the "Display text" field.

Display type "Text with datapoint parameters":

A text is displayed that contains one or multiple datapoint values in addition to the language-dependent text. The datapoint value may also include the complete message text.

Evaluation rate [s]:

Describes the interval how often the trigger condition for sending the push message should be evaluated in App Manager. The smaller the number, the higher the load at the end point.

6 Configure User Roles

User roles can be defined in the user role editor. Permissions for protected objects can then be assigned to each user role.

If there are no user roles, access protection is generally disabled and all users have the same rights.

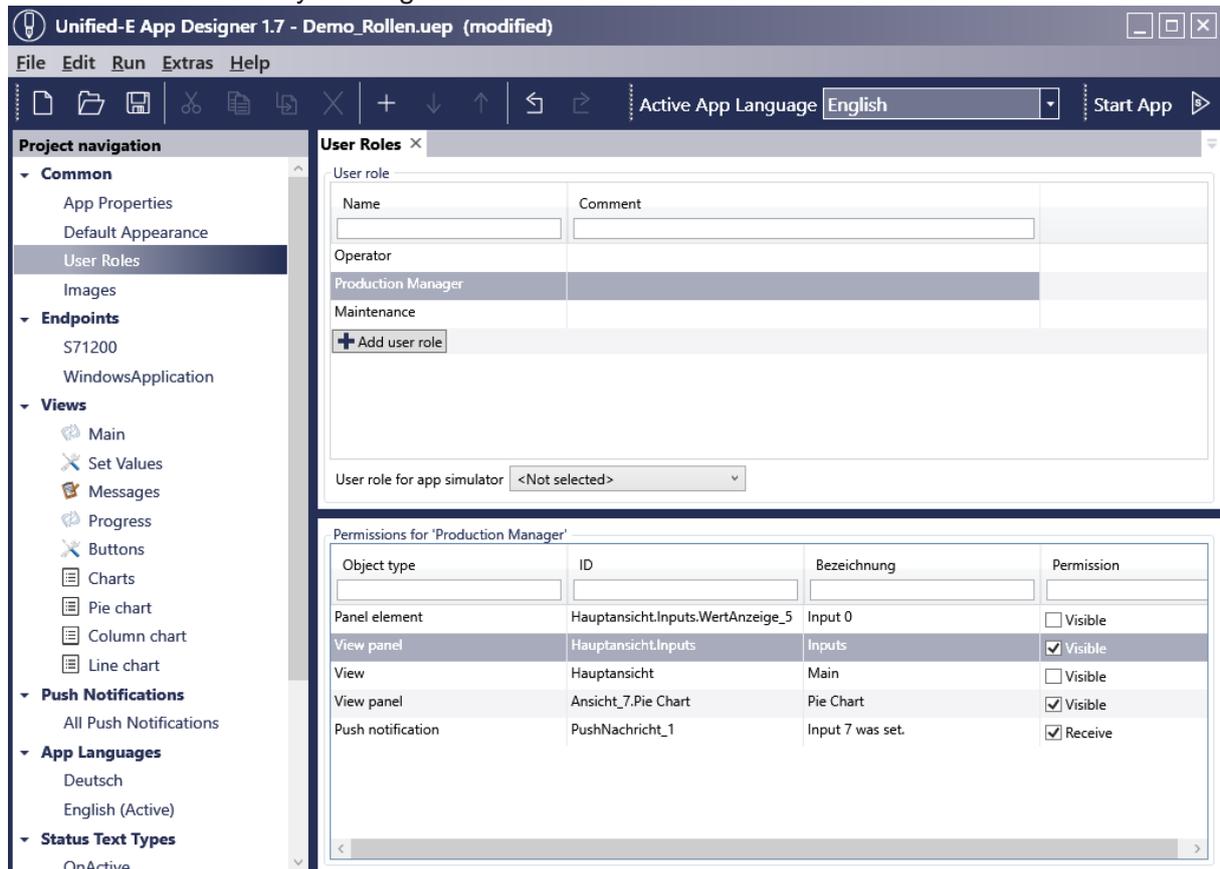
Each operator app user will be given a user role (if any). This is done in the App Manager during user registration.

6.1 Define User Roles

The user role editor is started by double-clicking the "User roles" entry in the project navigation.

Add user role:

Add a new user role by clicking "Add user role".



Permissions table:

All objects with access protection are listed here. The permissions for the selected user role in the user role table can be configured in the "Permission" column. Note: In order to assign permissions for an object, access protection must first be activated for the desired objects (see next chapter).

6.2 Activate Access Control

The following objects are currently available with access protection:

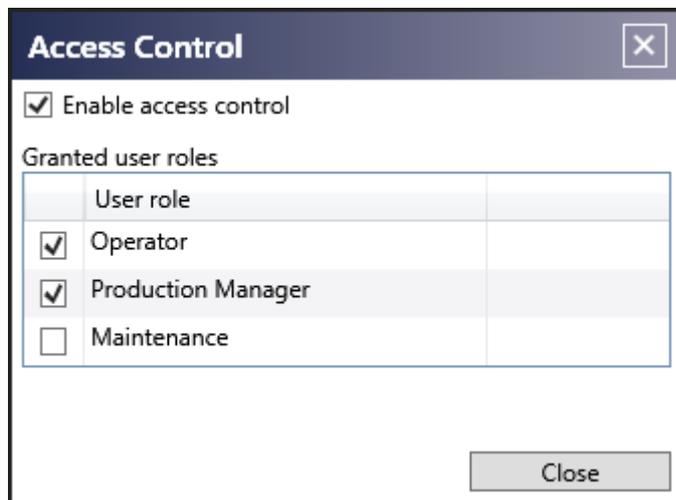
- View: If granted, all associated view links are visible
- Panel: If granted, the panel is visible
- Panel element: If granted, the panel element is visible
- Push notification: If granted, this push notification can be received

Activate access control within view editor:

In the view editor, the access protection can be activated in the object properties (see chapter 4.6.1.1.6).

Activate access control within push notification editor:

1. Select the desired push notification
2. Press the "..." button in the "Access control" column
3. Activate access control in the "Access Control" dialog box by setting the checkbox "Activate access control"
4. Assigning permissions: Set the button for granted user roles



7 Test the App

Operator apps can already be tested in the App Designer - description below. But push notifications cannot be tested in the App Designer yet.

However, it is recommended to test a published operator app in the App Manager again, with the smartphones that will be used. We suggest installing a second app instance in the App Manager, which is only for testing purposes.

7.1 Testing with Direct Endpoint Connection

Start:

Start the operator app with a direct connection to the configured endpoints by selecting "Run > Run" or via the "Run" toolbar button.

Toolbar button: 

Please note that the configured adapter parameters will be used to connect to the endpoints.

Testing:

After starting the operator app opens in a new window "App Simulator (endpoint connection)" and can be tested.

In addition, the current datapoint values can be seen in the datapoint table of the endpoint.

The current datapoint value is displayed in the "Value" column.

Name	Address	Access	Data type	Control PIN	Def. sim. value	Value
AlarmTrigger	AlarmTrigger	Write, Read	Numeric	No	0	1
ValveOpened	ValveOpened	Write, Read	Yes/No	No	0	0
CurrentAction	CurrentAction	Write, Read	Numeric	No		0
XPosition	XPosition	Read	Numeric			<Datapoint not found>
YPosition	YPosition	Read	Numeric			<Datapoint not found>

Note: Datapoint values cannot be set directly in the table, for safety reasons.

7.2 Testing with Simulated Datapoint Values

In this test no connection to the endpoints is required, but datapoint values can be set directly in the datapoint table.

This test mode has to be used if the endpoints in the project cannot be reached by the PC of the app developer via a network connection.

Start:

Start the operator app by selecting "Run-> Run (simulated endpoint)" or via the "Run (endpoint connection)" toolbar button.



The initial datapoint value after starting is the value set in "Def. sim. Value" column.

If the "Def. sim. Value" cell is empty a default value is used:

- 0 for numeric datapoints
- „“ for text datapoints
- 0 for Yes/No datapoints

Testing:

After starting the operator app opens in a new window "App Simulator (simulated endpoints)" and can be tested.

Unlike in testing with endpoint connection the cells in "Value" column can now be edited to simulate different datapoint values.

Name	Address	Access	Data type	Control PIN	Def. sim. value	Value	Group
XPosition	XPosition	Read	Numeric			22	
YPosition	YPosition	Read	Numeric			0	
AlarmTrigger	AlarmTrigger	Write, Read	Numeric	No	0	0	

8 App Publishing

8.1 Create App Package File

Generally, an app package is required to publish an operator app. Depending on the type of communication, the app package is then installed on the App Manger or directly in the Unified-E App on the operating device (smartphone).

For direct communication, an additional signing of the app package is required.

App package files have the file extensions ".uea".

„Internal“ app package components:

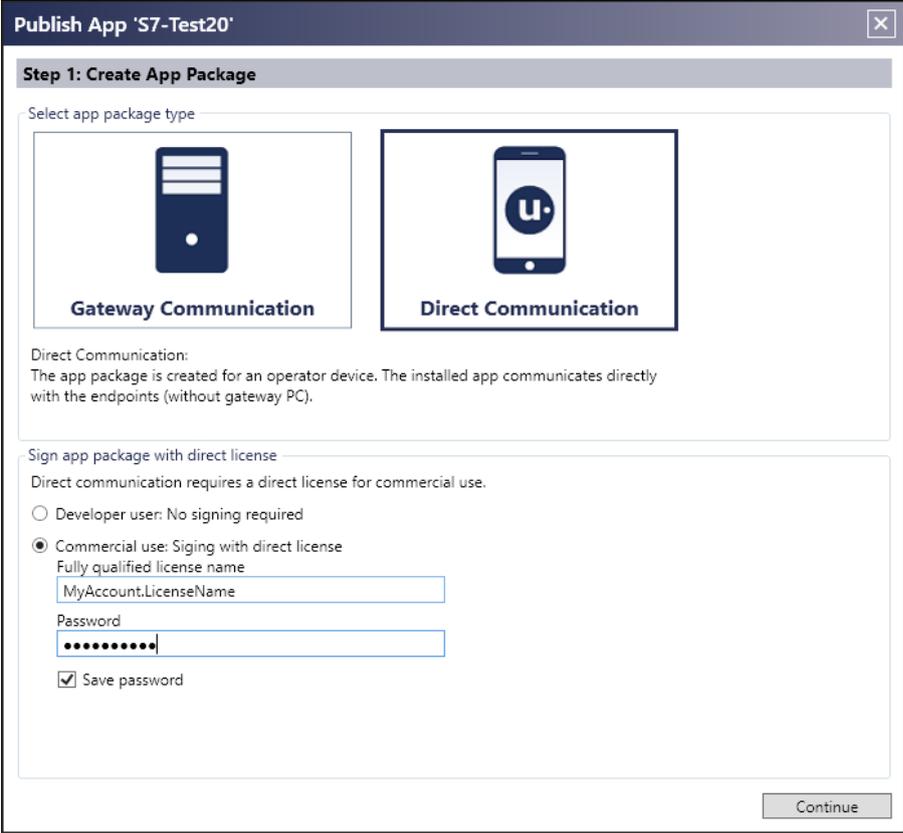
Basically the app package will be created based on the app project. Depending on the settings made under "Settings- > General Settings...", the app project will be integrated.

„App package“ group in the settings dialog:

App package

- Include project in App package
- Save project before deploying

The creation and publishing of the app package is started under "Run" -> "Publish App Package".



Publish App 'S7-Test20'

Step 1: Create App Package

Select app package type

Gateway Communication

Direct Communication

Direct Communication:
The app package is created for an operator device. The installed app communicates directly with the endpoints (without gateway PC).

Sign app package with direct license
Direct communication requires a direct license for commercial use.

Developer user: No signing required

Commercial use: Signing with direct license

Fully qualified license name

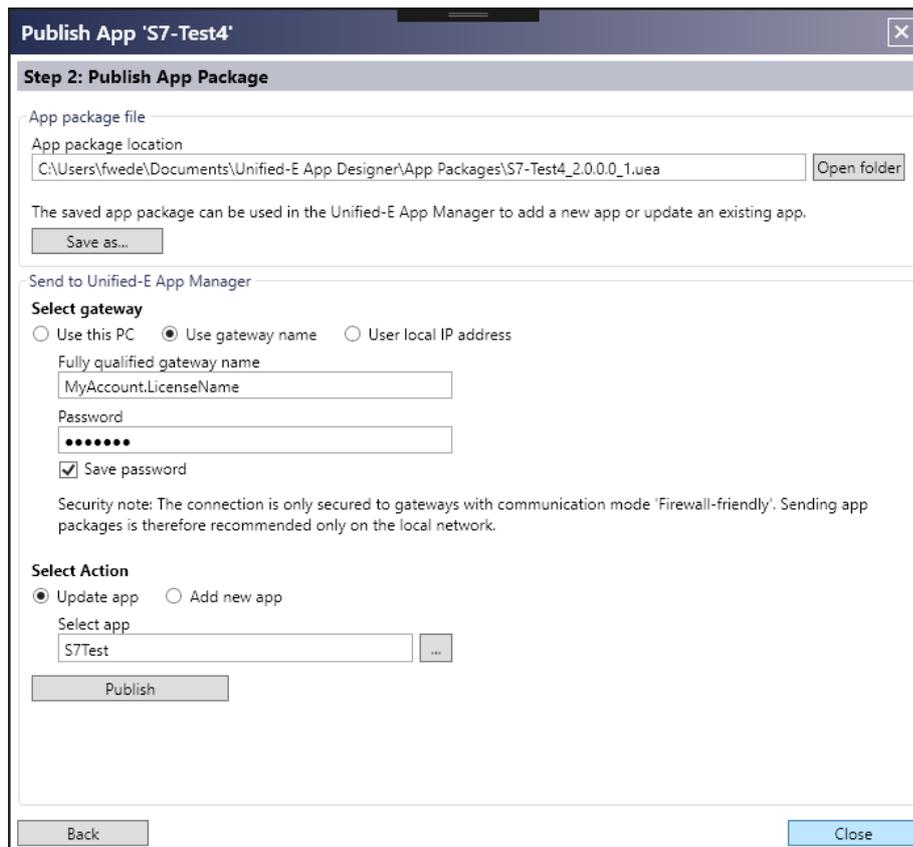
Password

Save password

Continue

8.2 Publish App Package to Gateway

The app package does not need to be signed because licensing is handled through the gateway license.



There are two ways to install the app package.

Manually install app package file:

The saved app package file can be installed in App Manager:

- 1) Select "Installed Apps" tab
- 2) Select the "Install App ..." button
- 3) Select the generated app package file
- 4) After install the app can be registered by the app users

Send app package file remotely to App Manager:

With this variant, you can install or update an app directly from App Designer without having to switch to App Manager.

To enable this function, the remote installation via App Designer must be enabled in the App Manager under "Settings" -> "Security".

8.3 Publish App Package for Direct Communication

Direct communication requires no internet. The license check is carried out via the signed app package, for the signing a direct license is required.

If the app package is not signed, then the developer/test operation is assumed. The operator app is then only a limited time available.



Open app package file with Unified-E App:

In addition to downloading the app package with a download address, the file can also be sent to the smartphone e. g. via mail or WhatsApp. You can open it with the Unified-E App. Afterwards, the app package file can be opened with the Unified-E App and then registered there.

8.4 Update Existing App

Updating is the same as installing an app package file. Similar to installing, the update can be done remotely in the "Publish App Package" dialog or manually in the App Manager application.

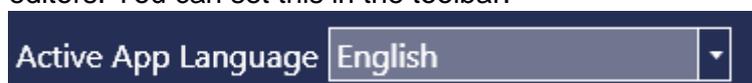
9 Multilingual Apps

9.1 General

All display texts can be set multilingual. There are input fields for multilingual texts in various editors, e. g.

- Display text input field in a property window of a view element
- Message text column in a push notification editor
- Text column in the status text type editor

The "Active App Language" determines the current language of the display texts in the editors. You can set this in the toolbar:



9.2 Add a New Language

You can add a new language as follows:

1. Open the context menu “App languages”
2. Select “Add Language”
3. By double-clicking on the new item (e. g. “Language_1”) the app language editor opens.

9.3 Modify Texts in the Language Editor

„Common“ group:

Language:

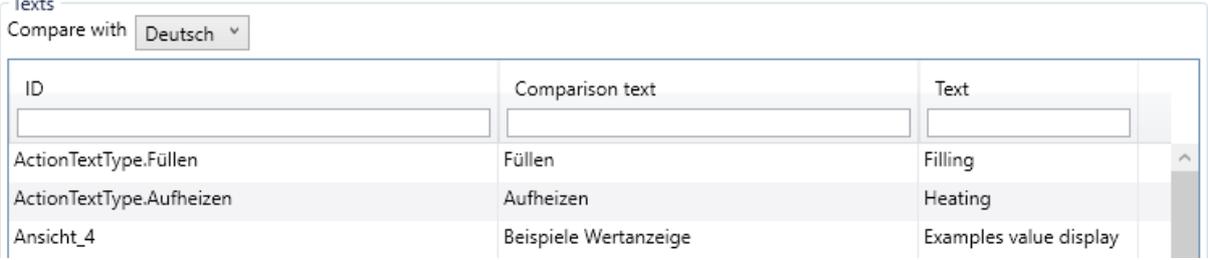
There you can set the system language. If the app user has set a language, which is defined in the language column, the operator app will be started automatically in this language.

Name:

An alternative name for the language.

“Texts“ group:

The text table lists the ID and text of the language of all objects with text display.



ID	Comparison text	Text
ActionTextType.Füllen	Füllen	Filling
ActionTextType.Aufheizen	Aufheizen	Heating
Ansicht_4	Beispiele Wertanzeige	Examples value display

“Compare with” selection list:

You can select a comparison language in the selection list. This, for example, is useful when translating.

„ID“ column:

Unique name of the object with the display text, based on the set name or object name.

„Comparison text“ column:

The text in the comparative language, if set in the selection list “Compare with”.

Text:

The text that needs to be edited in the current language of the language editor.

9.4 Set App Language on Smartphone

The user has the possibility to set the language of a specific operator app:

1. Select the context menu in the operator app in the home screen
2. Select „Configure“
3. Set the desired language in the setup window in the “Language” group and press “Apply”

Setup screen:

Language

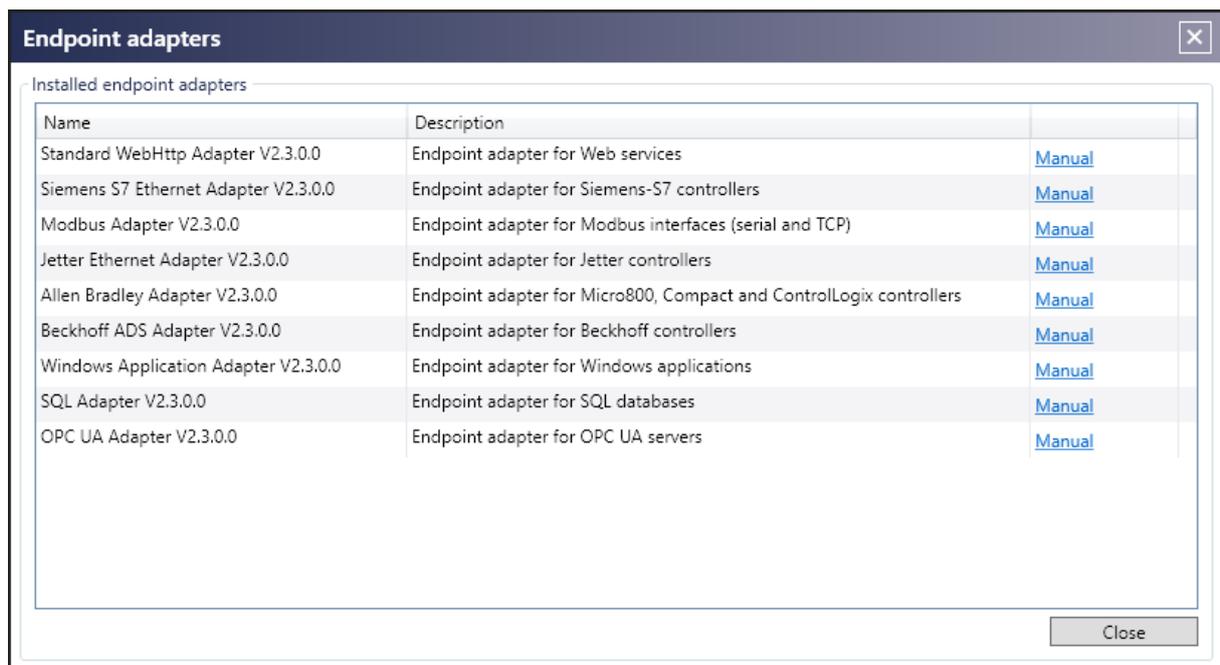
English

Apply

10 Installed Endpoint Adapters

Endpoint adapters are needed because different endpoints (e. g. Siemens PLC, Beckhoff PLC) define an own communication protocol each - such as how datapoint values can be written or read.

All endpoint adapters are shipped with the App Designer and are listed under "Settings" -> "Endpoint Adapters ...".



The "Manual" link opens the manual of the selected adapter.

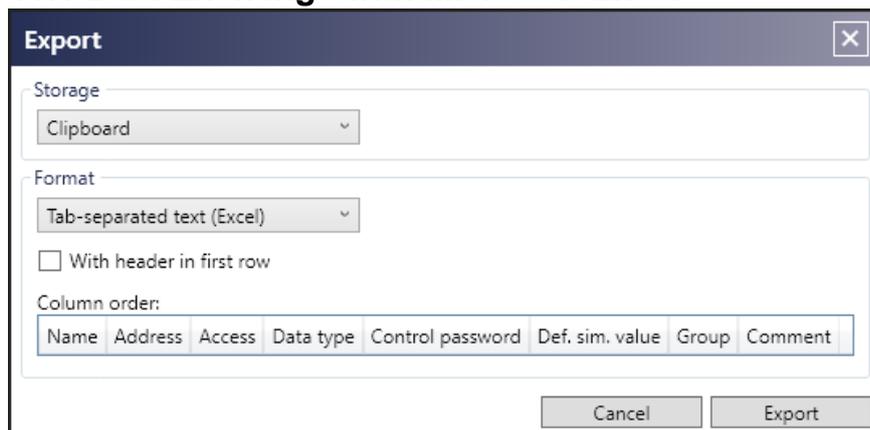
11 Import and Export Data

The following tables support importing/exporting of the content:

- Datapoint table
- Status text type editor
- Language editor

The action is started with the "Import"/"Export" button.

11.1 Data Exchange with Microsoft Excel



The easiest way to exchange data with Excel is via clipboard. For that you must select "Tab-separated text (Excel)" in the "Format" selection list.

Example:

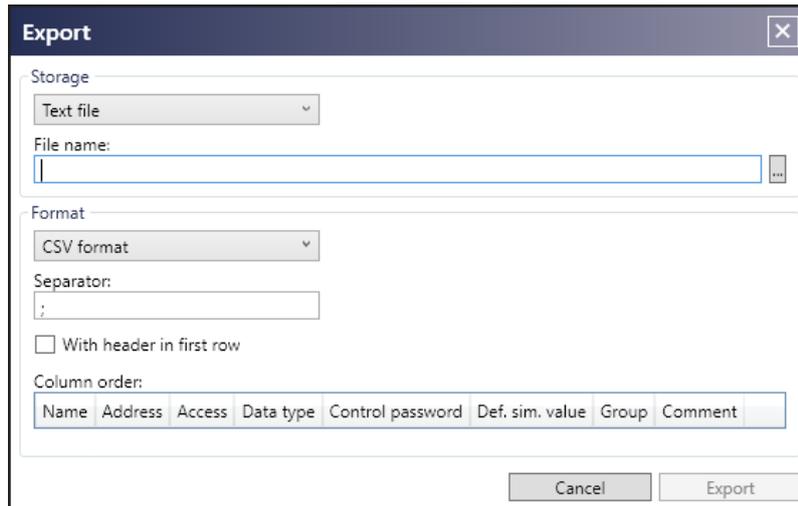
You want to create similar datapoints based on existing datapoints already configured with the Unified-E App Designer.

Solution:

- Export the datapoints with the "Export" button as described above. Afterward paste the datapoints into Excel from clipboard and duplicate/edit the new datapoints.
- Import the datapoints back to the datapoint table: Copy all datapoint rows and columns of the new datapoints to clipboard. Afterward import these datapoints with the "Import" button above the datapoint table.

11.2 Data Exchange with CSV- and Text Files

Choose the "Text file" in the "Storage" section. You can select an individual separator for CSV format.



12 Handle Endpoint Messages

Messages or message lists for PLC endpoints can be configured as follows:

1. Define a PLC register/variable with an integer type, e. g. WORD
2. Every bit (trigger bit) of this variable is assigned a specific message, e. g. for the first bit: 0 = "Emergency Stop"
3. Create a new status text type in the Unified-E App Designer. Assign a message for each bit index.
4. The operator app will display all messages in a "List panel" element.

Messages are supported from all PLC endpoint adapters (also Modbus, OPC UA).

12.1 Define Datapoints for Messages

1. Goto datapoint table: Define a new datapoint which addresses the message variable in the PLC.
2. Set the data type to "Data table" in the column "Data type".

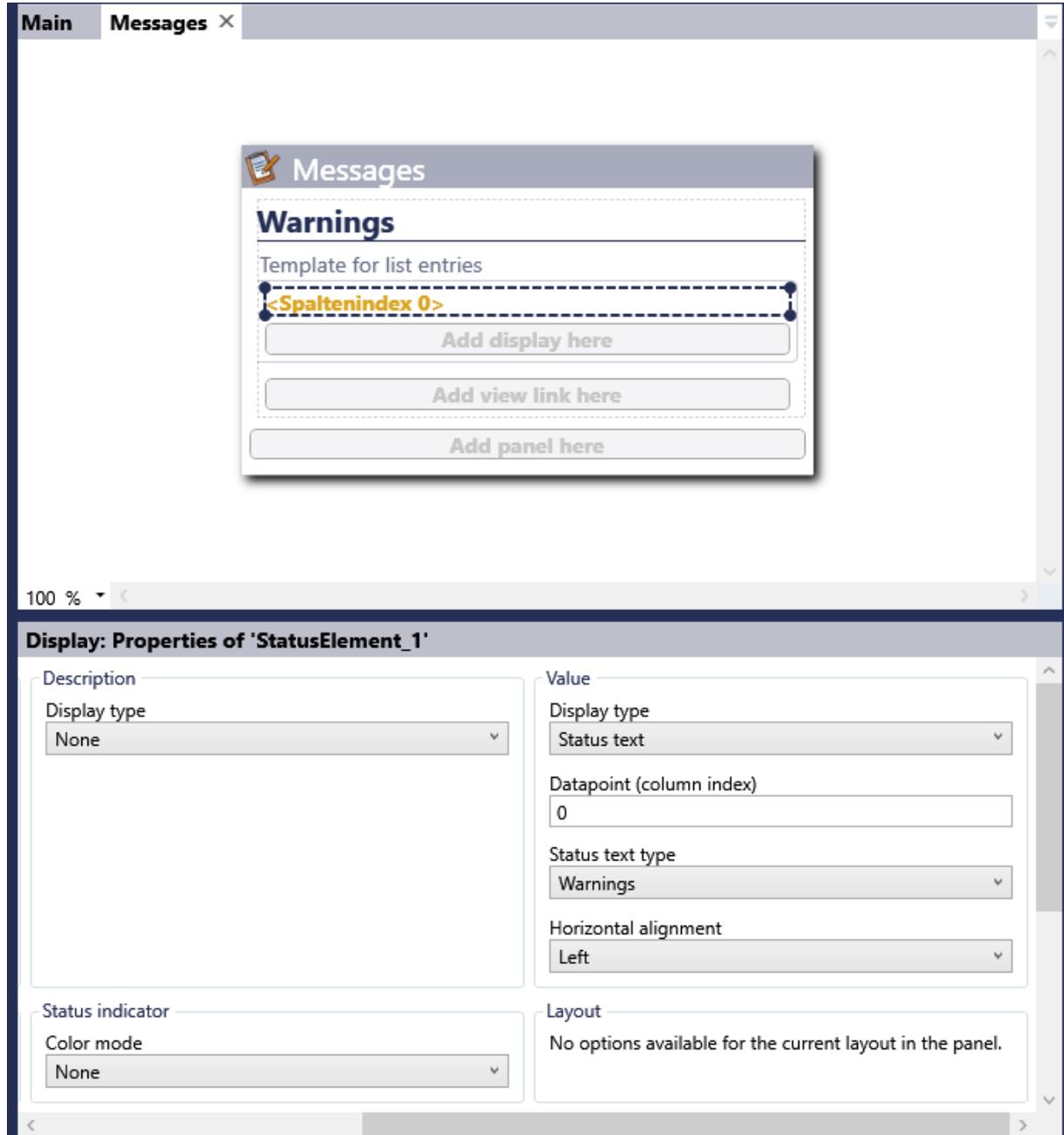
12.2 Define Message texts with a Status Text Type

1. Create a new status text type, goto "Status Text Types" in navigation context.
2. Create a new message text for each trigger bit with the "Add status text" button. The "Value" column is used for the bit index which starts at 0.

12.3 Add Message List to View

1. Open the view editor where you want to add the message list
2. Drag a "List panel" from the right editor panel to the view
3. Set the properties of the list panel as follows (see also 4.6.3.2)

- a. Set the datapoint of the message variable which must be of type "Data table"
- b. Configure the display element for the list entry template
 - i. Set display type to "Status text"
 - ii. Set "Datapoint (column index)" to 0
 - iii. Apply the status text type containing the messages



13 Curve Recordings and Time Charts

Curve recording means that a measurement value (datapoint value) is periodically requested and stored. The saved value pairs (time value / measurement value) can be displayed in a time chart.

Examples:

- Time line for a temperature for the last 7 days

- Time line for measurement values of a process

Recording value:

The endpoint must provide a datapoint of "numerical" data type. This datapoint is then used for recording.

Recording and storage:

Both recording and saving the measurement values are done in the Unified-E App Manager. The recorded values are saved in an XML file. This file contains a signature. Therefore, manual modification on that file are recognized.

13.1 Create a Curve Recording Datapoint

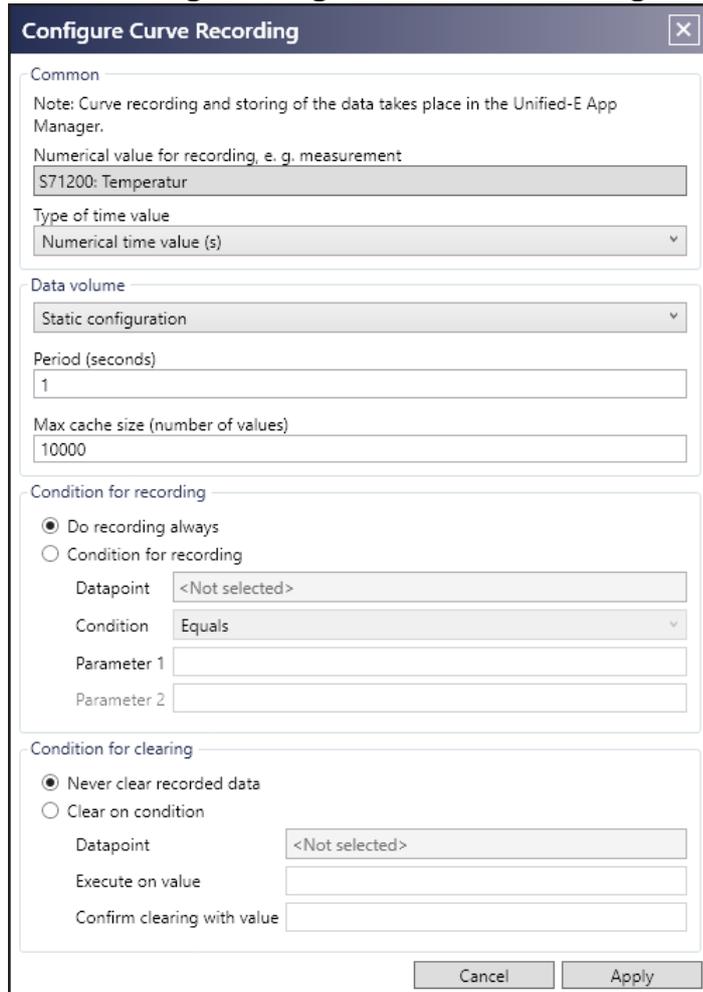
General:

All view elements get the variable data via datapoints. This applies also for a chart chart or chart series. The table of values of the recorded data must be provided from a datapoint with data type "data table".

13.1.1 Add Curve Recording

1. Assumption: You already added a numerical datapoint which represents the recording value (measurement value).
2. Open the datapoint table of an arbitrary endpoint.
3. Choose "Add curve recording..." either via the context menu or with the button below the table.
4. A new curve recording datapoint is created with data type "data table". The datapoint is special since it does not have any address.
5. The "Configure Curve Recording" dialog is opened. There you can configure the recording.

13.1.2 Dialog "Configure Curve Recording"



"Common" property group:

Numerical value for recording, e. g. measurement:

There you must select the numerical datapoint which provides the current value for recording.

Type of time value:

Absolute date/time: The curve recording datapoint provides the table of values with absolute time values.

Numerical time value (s): The curve recording datapoints provides the table of values with numerical (relative) time values. The time values are the elapsed seconds since start of recording.

"Data volume" property group:

Configuration mode:

Static configuration: The properties are set to constant values in the Unified-E App Designer.

Dynamic configuration with datapoints: The properties are set by means of datapoints at runtime.

Period (seconds):

The recording period in seconds.

Max. cache size (number of values):

The maximum cache size for the recorded values. The oldest values are deleted if required.

"Condition for recording" property group:

Here you can configure the start and the end of a recording.

„Do recording always“ button:

If this button is set, recording is always active.

“Condition for recording” button:

If this button is set, a condition can be set that controls the recording activity.

Datapoint:

The value of the selected datapoint is used to verify if the condition is satisfied.

Condition:

The specific compare condition, e. g. "Equals", "Greater".

Parameter 1:

The first parameter for the condition.

Parameter 2:

The second parameter for the condition.

"Condition for clearing" property group:

"Never clear recorded data" button:

If this button is set, clearing is not configured.

"Clear on condition" button:

If this button is set, a clear condition can be configured.

"Datapoint" field:

The selected datapoint is used to verify if the clear condition is satisfied. This datapoint must have both read and write access since it is also used to confirm clearing.

"Execute on value" field:

If this value equals to the datapoint value, then the recording is cleared.

"Confirm clearing with value" field:

After clearing the value of this field is set to the datapoint.

13.1.3 Edit Curve Recording

1. Select the curve recording datapoint in the datapoint table.
2. Klick the "..." button in the "Address" column.
3. The dialog "Configure Curve Recording" is opened for editing.

13.2 Create Time Chart

You can use the curve recording datapoint in a time chart as follows:

1. Add a chart panel to the view.
 - a. Set the "Diagram" property to the value "time chart".
 - b. Set the "Time axis" property: This must accord with the "Type of time value" property of the curve recording datapoint.

2. Select the chart series and set properties.
 - a. Go to "Table of values" property group and set the curve recording datapoint.
 - b. Set other chart series properties

