COPA-DATA
do it your way

zenon manual

Screens

v.7.00

© 2012 Ing. Punzenberger COPA-DATA GmbH

All rights reserved.

Distribution and/or reproduction of this document or parts thereof in any form are permitted solely with the written permission of the company COPA-DATA. The technical data contained herein has been provided solely for informational purposes and is not legally binding. Subject to change, technical or otherwise.

Contents

1. Welcome to COPA-DATA help 9

2. Screens 9

3. Context menu project manager 11

4. Graphics quality 12

4.1 Error treatment DirectX 16

5. Detail view of toolbar and context menu screens 18

6. Edit Screens Toolbar 20

7. Toolbar elements 22

8. Context menu elements 24

9. XML-Export and XML-Import 28

10. Screen elements 30

10.1 .NET Controls 37

10.2 ActiveX 39

10.3 Bar Graph 41

10.4 Command Input Element 43

10.5 Screen Alarm Button 43

10.6 Button 43

10.6.1 Animate graphics 46

10.6.2 Bitmap Button (up to version 6.22 only) 46

10.6.3 Invisible button (up to version 6.22 only) 49

10.7 Combined element 51

10.7.1 Assistant 54

10.7.2 Clickable buttons in any form 59

10.7.3 Statuses 63

10.7.4 Test mode 77

10.7.5 Formula editor 77

10.7.6 Display of variable name and variable identification ... 88
10.8 Combo/List Box ... 88
10.9 Dynamic Text ... 96
10.10 Ellipse ... 97
10.11 Circle ... 97
10.12 Arc of a circle ... 98
10.13 Segment of a circle ... 99
10.14 Line ... 100
10.15 Message element ... 101
10.16 Multi-binary Element ... 103
10.16.1 Multi-binary element ... 104
10.17 Polygon ... 110
10.18 Polyline ... 112
10.19 Rectangle ... 113
10.20 Square ... 114
10.21 Tube ... 114
10.22 Switch ... 116
10.23 Static Text ... 116
10.24 Move symbol (up to version 6.22 only) ... 118
10.25 Trend Element ... 118
10.26 Clock ... 120
10.27 Universal slider ... 122
10.28 WPF-Element ... 126
10.28.1 Basics ... 126
10.28.2 Manual for designer ... 133
10.28.3 Engineering in zenon ... 146
10.29 Numerical value ... 191
10.30 Indicating instrument ... 192
10.31 Status Element ... 193

11. Edit screen element ... 194

11.1 Graphic actions ... 194
11.1.1 Truncations ... 195
11.1.2 Effects for screen elements ... 196
11.1.3 Move element ... 210

11.1.4 Color gradient and transparency ... 210
11.1.5 Transfer format ... 213
11.1.6 Copy, Insert and Move ... 215
11.1.7 Add and delete dots ... 216
11.1.8 Arrows in vector elements ... 217
11.1.9 Grid ... 218
11.1.10 Font ... 219
11.1.11 Selecting ... 220
11.1.12 Scale, center and zoom ... 220
11.1.13 Keyboard shortcuts ... 221
11.1.14 Assign key to control element ... 224
11.2 Replace link of variables and functions ... 225
11.2.1 Naming conventions ... 226
11.2.2 Replace link in Editor screen ... 227
11.2.3 Replace link at screen switch ... 229
11.2.4 Replace indices ... 233
11.2.5 Example replace variable rule-based ... 235
11.3 Background graphics ... 236
11.4 Functions Selection dialog ... 237
11.5 Variable selection dialog ... 238

12. Frames ... 242

12.1 Frame detail view toolbar and context menu ... 244
12.2 Frame editor ... 245
12.2.1 Opening the frame editor ... 246
12.2.2 Display frames in frame editor ... 247
12.2.3 Create new frame ... 247
12.2.4 Positioning in Runtime ... 249
12.2.5 Deleting frames ... 252
12.3 Define background graphics for frames ... 253
12.4 Distributed engineering with frames ... 253
12.5 Main frame ... 254
12.6 Effects for frames ... 254
12.6.1 Highlight frame ... 254

13. Color palettes ... 257

13.1 Palettes toolbar and context menu detail view ... 258
13.2 Create and edit colors ... 260
13.3 Create and edit colors ... 261
13.4 Use palette in the Editor ... 264
13.5 Create Runtime files ... 265
13.6 Switch color palette function ... 266
13.7 Export and import ... 267
13.8 Error messages ... 269
13.9 Example for Editor and Runtime ... 269

14. Fonts ... 272

14.1 Font lists ... 272
14.1.1 Creating a new font list ... 274
14.2 Default font and system font ... 274
14.2.1 Creating a new font ... 275
14.3 Naming and numbering fonts and font lists ... 277
14.4 Linking fonts to elements ... 279
14.4.1 Embedded and linked fonts ... 280
14.5 ClearType ... 281

15. Symbols ... 282

15.1 Symbol detail view toolbar and context menu ... 286
15.2 Configure size and background color ... 288
15.3 Release properties from link ... 288
15.4 Create and delete your own symbols ... 291
15.4.1 Transfer own symbols to the library ... 294
15.5 Add symbols to screen ... 295
15.5.1 Embedded symbols ... 296
15.5.2 Linked symbols ... 297
15.5.3 Symbol selection in the Combined element ... 300
15.6 Symbol list ... 303
15.7 XML export and XML import ... 303

16. Visibility levels ... 305

17. Screen types ... 306

17.1 Special screen types 307
17.1.1 Create special screen types 310
17.1.2 Template for standard screens and special screen types 310
17.2 Screen-specific functions 317
17.3 Screen filter 317
17.3.1 Screen filter dialog 317
17.3.2 Creating an Alarm Message List filter screen 327
17.3.3 Create a screen of the type CEL Filter 336
17.3.4 Time filter screen 345
17.4 HTML 349
17.4.1 Creating a screen of the type HTML 349
17.4.2 Function Screen switch - HTML 349
17.4.3 Operating during Runtime 351
17.5 Notebook 352
17.5.1 Creating screen Notepad 352
17.5.2 Function screen switch to Notepad 353
17.5.3 Operating during Runtime 354
17.6 Video 355
17.6.1 Creating a screen of the type Video 356
17.6.2 Screen switch - video 356
17.6.3 Operating during Runtime 361

18. Screen Functions 362

18.1 Screen switch 363
18.2 Index screen 366
18.3 Close screen 366
18.4 Screen back 368
18.5 Focus: Activate input to the element with the focus 369
18.6 Set focus to template 369
18.7 Move focus 372
18.8 Focus: Delete from frame 372
18.9 Show menu 372
18.10 Monitor assign 373
18.11 Close template 374
18.12 Set point input for keyboard screen 377
18.13 Display overview window 378

18.14 Screen: Move center 381

1. Welcome to COPA-DATA help

GENERAL HELP

If you miss any information in this help chapter or have any suggestions for additions, please feel free to contact us via e-mail: documentation@copadata.com (mailto:documentation@copadata.com).

PROJECT SUPPORT

If you have concrete questions relating to your project, please feel free to contact the support team via e-mail: support@copadata.com (mailto:support@copadata.com)

LICENSES AND MODULES

If you realize that you need additional licenses or modules, please feel free to contact the sales team via e-mail: sales@copadata.com (mailto:sales@copadata.com)

2. Screens

Screens are central elements of a project. They display the configured equipment, inform, and provide user elements.
Screens consist of vector elements and/or dynamic elements. These are linked with variables or functions. Screens are configured in the main window of the Editor.

zenon provides a large number of pre-defined screen types, such as alarm information lists, trend screens etc. The advantage of this type of screen is that the desired functionality is connected to the

screen type. You must create the screen type and and you have a variety of pre-defined functions available. You can find an overview of screen types in the Types of screen (on page 306) section.

License information

Part of the standard license of the Editor and Runtime.

MAIN WINDOW

In the main window documents such as screens, reports, etc. are displayed and edited. The main window is the only window that cannot be hidden.

OPEN MULTIPLE DOCUMENTS PARALLEL.

You can display multiple screens in the main window at the same time. This function can be activated via: *Options -> main window with tabs* . The screens can be selected with tabs at the top of the main window and can be positioned via options in menu item **Windows**.

To close open screens, use the keyboard shortcuts `Ctrl+F4`, the context menu or the command *Window -> Close*.

CONTEXT MENU DOCUMENTS

Right-click on the tab of a document in order to open the context menu.

Parameters	Description
Save	Saves the document.
Close	Closes the document.
Close all others	Closes all other documents.

POSITION ELEMENTS IN THE SCREEN

Activate elements by clicking on the symbol in the Toolbar elements and open them with the mouse in the main window.

There are multiple possibilities for moving the elements:

- Move quickly: Click element and position it with the mouse

- Position exactly: Click element and position it with the arrow keys
- Turning: Turn element with the "handle" above the upper left corner as you like it.

POSITIONING OF ELEMENTS IN THE EDITOR AND IN RUNTIME

Elements can be positioned at different levels in the Editor and in Runtime. Individual elements can also be hidden (on page 305) in the Editor

Attention: Control elements are always displayed at the uppermost position. That also applies if they are covered by other elements in the Editor. This ensures that they can always be reached.

ADJUSTMENT TO RESOLUTION

Screens and their elements are automatically adjusted to the screen resolution in Runtime. This can be prevented. to display screen elements in the size and position configured in the Editor, activate the `Do not adapt element to screen resolution` property for the corresponding screen in the `Size` group.

3. Context menu project manager

Menu item	Action
New screen	Creates a new screen with the standard name 'Picture' as a standard screen type.
Export XML all ...	Exports all entries as an XML file.
Import XML...	Imports entries from an XML file. **Note:** Existing frames are not replaced during XML import using the **`Screens`** node. Existing frames are overwritten via the **`Frames`** node during XML import. In this case, all screens and their elements are adapted to the new frame.
Editor profile	Opens the drop-down list that includes pre-defined editor profiles.
Help	Opens online help.

4. Graphics quality

In zenon the quality of the displayed graphics can be tuned to the resources for the system. This setting is made using the project setting `Graphics quality` in group `Graphical design`. Possible options are:

- `Windows Basic`: Basic graphics settings. Recommended for resource-weak hardware.
- `Windows Enhanced`: Provides enhanced functions for the graphical display - needs more resources.
- `DirectX Software`: Graphics calculations are done by the CPU. Depending on the graphic you can use more than one CPU. Using **`DirectX Software`** may cause a high CPU workload.
- `DirectX Hardware`: A part of the graphic calculation is done by the graphics card by which the performance is increased. If this setting is not supported by the used system, zenon automatically switches to `DirectX Software`. Generally speaking `DirectX Hardware` is preferable and `DirectX Software` should only be used if necessary.

> **Attention**
>
> - **`Windows Enhanced`** and **`DirectX`** are not available under Windows CE.
> - DirectX display is only supported in the Runtime.

DIRECTX

`DirectX` allows a higher graphic quality than `Windows Basic` or `Windows Enhanced`. Thus in contrast to `Windows Basic/Enhanced`, DirectX can turn on/off properties `Use antialiasing` and `ClearType fonts` regardless of the settings in the operating system. The antialiasing quality of DirectXis higher than the quality of `Windows Enhanced`.

To use **`DirectX`**, several requirements must be fulfilled:

Requirement	Description
`DirectX Hardware` or `DirectX Software` must be activated.	In the project settings you must select `DirectX Hardware` or `DirectX Software` for property `Graphical design`.
The operating system must support DirectX 11.	`DirectX Hardware` and `DirectX Software` only works on operating system supporting DirectX11: ▸ Microsoft Windows Vista SP2 incl. Plattform Update and Windows 7 ▸ Windows XP and Windows CE are not supported If the system does not support DirectX 11, it automatically switches to `Windows Enhanced`.
The current DirectX Runtime must be installed.	For zenon it is installed together with the setup. For the web client it must be installed manually.
The screen or element must support DirectX.	Only elements or screens which support DirectX can be displayed with DirectX.
For `DirectX Hardware` the minimum requirements must be fulfilled.	For details see minimum requirements graphics card. If they are not fulfilled, it automatically switches to `DirectX Software`. If the operating system does not provide any hardware acceleration (e.g. Remote Desktop), it may not be possible to use `DirectX Hardware`.

MINIMUM REQUIREMENTS GRAPHICS CARD FOR DIRECTX HARDWARE

For the use of `DirectX Hardware` the following minimum requirements must be fulfilled:

Parameters	Minimum requirements	Recommended
Graphics card:	Dedicated DirectX 11 AMD or nVidia graphics card	Dedicated DirectX 11 AMD or nVidia graphics card
Graphics memory: (Actually needed size depends on the number of the called up screens and displayed elements.)	512 MB VRAM	1 GB VRAM
Driver graphics card:	Most current driver of the graphics card manufacturer	

To check the graphics card support for `DirectX Hardware`, you can use the Windows operating system tool **`dxdiag.exe`**. With this the DirectX hardware capability of the graphics card and the driver

can be checked under **Display** with the value **DDI Version**. The value 11 for example means DirectX 11.

DIFFERENCES BETWEEN DIRECTX AND WINDOWS ENHANCED

DirectX differs from Windows Enhanced regarding quality, performance and resource requirements. Some of the most important differences:

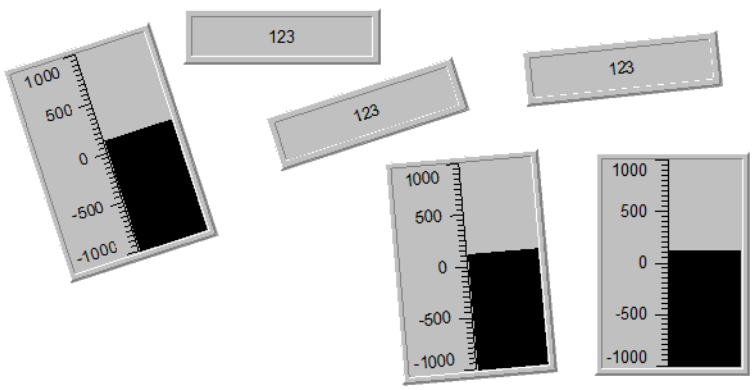
- Scaling, contortions and rotations of objects are displayed with antialiasing. Antialiasing also works with transformations.
- Shadows are displayed as a correct likeness of the actual element. Even texts and semi-transparent bitmaps have a completely correct shadow. Dynamic changes are considered: If a button is clicked, the shadow also changes.
- Glow effect enables the glowing of an element via the likeness of the actual element. Even texts and semi-transparent bitmaps have a correct glow effect. Dynamic changes are considered: If a button is clicked, the glow effect also changes.

> **Info**
>
> *The use of WMF and EMF graphics with DirectX may cause performance losses.*

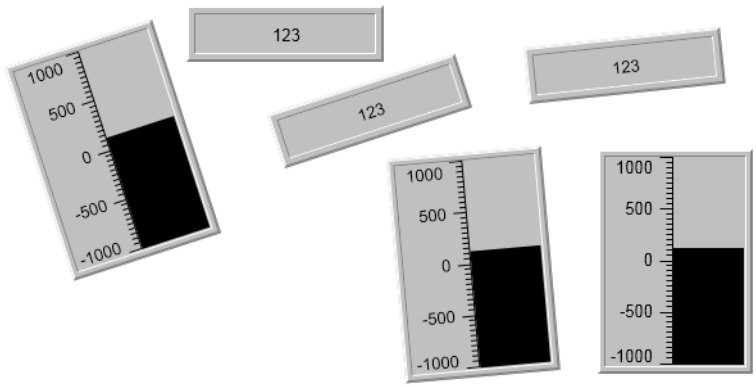
EXAMPLES

ROTATED DYNAMIC ELEMENTS

With Windows Enhanced:



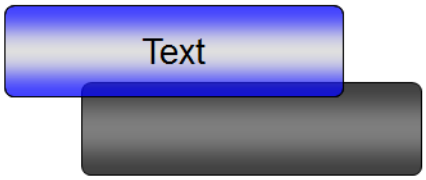
With `DirectX`:



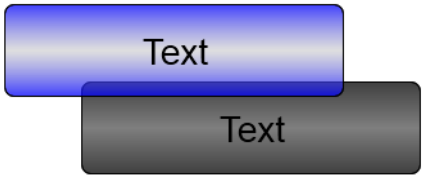
The forming of ridges at the edges is reduced, texts are positioned correctly.

SEMI-TRANSPARENT BUTTON WITH SHADOW

With `Windows Enhanced`:



With `DirectX`:

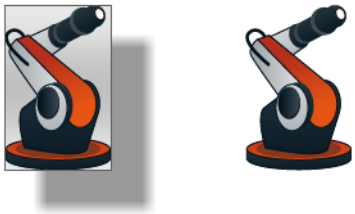


Correct shadow with visible text.

SHADOW OF A BUTTON WITH BITMAP

On the left a bitmpa button with shadow is displayed, on the right the same screen with property `Transparent` active.

With `Windows Enhanced`:



With `DirectX`:

4.1 Error treatment DirectX

Errors are displayed in the Diagnosis Viewer.

DIRECTX IS NOT INITIALIZED.

If DirectX cannot be initialized during the call up of a screen or in continuous operation,

- a progress bar is displayed
- it is tried to initialize DirectX again
- the Runtime continuous normally as long as the progress bar is displayed
- you can try to solve the problem (driver installation, ...)

The progress bar can be canceled by the user by clicking on button **`Cancel`**. After the cancellation the Runtime is closed and an error message is displayed which can only be closed by clicking **`OK`**.

Note: For errors which make it impossible to reinitialize DirectX, an error message is displayed immediately.

DIAGNOSIS VIEWER

DirectX provides individual messages in the Diagnosis Viewer with:

- an error message
- possible reasons for the errors
- Error codes

The display take place in an own module **DirectX**. For errors the error text is displayed in text field **Error**. For warnings and debug information the corresponding messages are displayed in text field **General**:

- For the successful call up of a screen at least one DEBUG success message is displayed.
- Warning point out events which can influence the operation.
- If an error occurs which cannot be fixed, an error message is displayed.

> **Info**
>
> *If a message is not completely displayed in the table in the Diagnosis Viewer, open the entry via double click to display the whole message.*

CHECK LIST FOR ERROR CHECKING

- An appropriate check whether DirectX works can be carried out by activating the debug message in the Diagnosis Viewer. For successful support, an appropriate message is displayed.
- Are DirectX-specific warnings or error messages displayed in the Diagnosis Viewer?
- Is property `DirectX Hardware` or `DirectX Software` activated in project setting `Graphics quality`?
- Are the minimum requirements for DirectX fulfilled? For details about minimum requirements see chapter Graphics quality (on page 12).
  - Is DirectX 11 supported by the operation system?
  - Is the current DirectX Runtime installed?
  - Is DirectX Hardware supported by the graphics card?

  To check the graphics card support for `DirectX Hardware`, you can use the Windows operating system tool **dxdiag.exe**. With this the DirectX hardware capability of the

graphics card and the driver can be checked under **`Display`** with the value **`DDI Version`**. The value `11` for example means DirectX 11.

- Is DirectX supported by the screen or the element?
- Does the display work properly with `Windows Basis` or `Windows Enhanced`?
- Does the display work with another driver?
- Does the display work with a graphics card of another manufacturer?

5. Detail view of toolbar and context menu screens

CONTEXT MENU

Menu item	Action
New screen	Adds a new screen to the list and puts the focus onto this entry.
Open screen	Opens the main window for the screen highlighted.
Create standard function	Opens the assistants to select a picture and for the configuration of details, in order to create a suitable function automatically. The action is documented in the output window.
Linked elements	Opens drop-down list with option **Jump back to start element**. Click to go to the element from which you jumped to the screen.
Copy	Copies selected entries to the clipboard.
Paste	Pastes the contents of the clipboard. If an entry with the same name already exists, the content is pasted as "Copy of".
Delete	Deletes selected entries.
Export selected XML	Exports selected entries as an XML file.
Import XML	Imports XML files. Note: Existing frames are not replaced during XML import using the **Screens** node. Existing frames are overwritten via the **Frames** node during XML import. In this case, all screens and their elements are adapted to the new frame.
Create template for screen type	Opens dialog for creating a new template (on page 313) for adding control elements in the screen type.
Remove all filters	Removes all filter settings.
Edit selected cell	Opens the selected cell for editing. The binocular symbol in the header shows which cell has been selected in a highlighted line.
Replace text in selected column	Opens the dialog for search and replace texts in the selected language.
Properties	Opens the **Properties** window for the selected entry
Help	Opens online help.

6. Edit Screens Toolbar

A toolbar is available for editing screens. Per default, it is displayed below the menu bar.

Symbol (from left to right)	**Function**
Save screen	Save the current screen.
Save all	Saves all changed screens.
Cut	Cuts the selected element and stores it in the clipboard; works across projects, too. Only available if an element has been selected.
Copy	Saves a copy of the current screen to the clipboard. Only available if an element has been selected.
Paste	Inserts (i.e. pastes) a screen from the clipboard. Only available if the clipboard contains data.
Redraw screen	Refreshes the display.
Edit mode	Switches from **Zoom** to **Edit mode** .
Undo	Allows undoing up to 100 actions. By default, 10 actions can be undone. Define the number of actions: *Options -> Settings ->Settings -> number of undoable actions*. Enter a number between 1 and 100.
Zoom	Provides two zoom modes: ▸ Variable zoom: A predefined value can be selected from the combobox, or you can directly enter any value between 15% and 400%. ▸ Zoom tool: You can zoom directly in the screen with the two magnifier symbols (**+** and **–**). The CTRL key switches between the different modes. The defined zoom is saved for each single screen. Close zoom mode: Click on the **Edit mode** symbol in the **Elements** toolbar.
Print	The whole screen content is printed on the default printer. Set printer properties via *File -> Standard configuration -> standard.*
Help	Opens the online help
Symbol bar options	Clicking on the arrow opens the submenu: Active: Toolbar is displayed. If the toolbar is not displayed, it can be activated using the Menu

	Options -> Toolbar.

7. Toolbar elements

no.	Symbol	Function
01	Rectangle	With frame and fill color, type of frame and fill pattern
02	Line	With definable line thickness, color and type
03	Rounded rectangle	Rectangle with freely definable rounded corners
04	Ellipse/Circle	With frame and fill color, type of frame and fill pattern (circle)
05	Polygon	With frame and fill color, type of frame and fill pattern
06	Polyline	With line type and color (also filling pattern and color for an area description)
07	Arc of a circle	With definable line thickness, color and type
08	Segment of a circle	draw circle segment with an opening angle of 180º (changeable), definable line thickness, color and type
10	Static text	With font color and selection of the type of font
09	Tube	With frame and fill color, type of frame and fill pattern
11	Bar Graph	Display values as bar graph
12	Numerical value	Display values numerically
13	Dynamic Text	Display limit value texts
14	Trend Element	Simple line graphics
15	Indicating instrument	Value in instrument display
16	Status Element	Take colors from status configuration
17	Button	Operating field in button display
18	Switch	Simple command initiation or set value input element
19	Multi-binary element	Display several variables in one screen point in color and symbol (string)
20	Screen Alarm Button	Only available with a license for SICAM 230
21	Message element	Display texts from a text table
22	Clock	Date and time

23	Command Input Element	Only available with a license for SICAM 230
24	Combined element	Display one or more variables in color or symbol (chained characters)
25	Screen alarming	Only available with a license for SICAM 230
26	ActiveX element	Insert ActiveX control
27	Universal slider	Slider control
28	Combo/List boxes	Link values of variables with text messages
29	WPF Element	Displays valid WPF XAML files.
30	Edit mode	Switches from zoom mode to edit mode
	Symbol bar options	Clicking on the arrow opens the submenu: Active: Toolbar is displayed. If the toolbar is not displayed, it can be activated using the Menu *Options -> Toolbar*.

8. Context menu elements

Commands adapted to the situation are available to you in the context menu. You can also get to these commands using menus and toolbars.

Command	Description
Symbol	Opens drop-down list with commands. The following are available depending on symbol status: ▸ Create embedded symbol ▸ Break up ▸ Changing to individual editing mode/leaving individual editing mode ▸ Insert in existing embedded symbol ▸ Convert linked symbol to embedded symbol ▸ Editing in the symbol editor ▸ Inserting in the symbol library
▸ **Create embedded symbol**	Creates an embedded symbol.
▸ **Break up**	Breaks up an embedded symbol into its screen elements.
▸ **Changing to individual editing mode/leaving individual editing mode**	Switches to the individual editing mode or back to the symbol editing mode. The individual editing mode enables you to edit individual elements of a symbol.
▸ **Insert in existing embedded symbol...**	Inserts symbol into an existing embedded symbol.
▸ **Convert linked symbol to embedded symbol**	Converts a linked symbol to an embedded symbol.
▸ **Edit in symbol editor**	Opens a linked symbol in the symbol editor to edit it there.
▸ **Inserting in the symbol library**	Opens dialog (on page 294) to add symbols from a screen of a symbol library.

Element position	Opens drop-down list for changing the position of the element: ▸ Foreground: Moves selected elements into foreground ▸ Background: Moves selected elements into background ▸ Forward: Moves selected elements one layer up ▸ Backward: Moves selected elements one layer down.
Arrange	Opens drop-down list for rearranging elements.
Linked elements	Opens drop-down list with dynamically linked elements, such as: Variables, functions, fonts, symbols etc.
Replacing links	Find/replace function e.g. tank 1 temperature is replaced by tank 2 temperature
Cut	Cuts the selected objects and stores them in the clipboard.
Copy	Copies selected objects
Paste	Pastes copied or cut objects form the clipboard.
Paste in same position	Pastes copied or cut objects form the clipboard.
Delete	Deletes selected objects
Transfer format (on page 213)	Transfers the properties of a selected element to another one / multiple elements. ▸ Transfer to an element: *Click on element with original properties* *-> Click on* **Transfer properties** *symbol or command in the context menu* *-> Click on target element: Properties are transferred* ▸ Transfer to multiple elements: *Select source element* *-> Select target element with the Ctrl key held down (source and target element are highlighted)* *> Click on the* **Transfer properties** *symbol or command in the context menu* *-> Properties of the source element are transferred to the target elements.* Source and target object determine which properties are transferred.

	If several elements were chosen as source, the properties of the first selected element are transferred.
Full screen mode	The full-screen hides all remaining windows and shows only the main window with all opened documents (screens, reports, ...) in full-screen mode. The full-screen mode can be closed by: ▶ the displayed button for closing the full-screen mode ▶ the short key `Shift+F9` ▶ the key `Esc`
Redraw screen/symbol	Refreshes the display.
Display grid	Switches the grid on or off.
Use grid	Switches usage of the grid on or off. On: All objects are automatically aligned with the grid.
Use magnetic points	Switches magnetic points on or off. Use: If you move an object near the handling point of another object, it snaps in place at the handling point.
Zooming	Changes the resolution in fixed steps between 15% and 400%.
Screenshot print current screen/symbol	The current screen is printed on the standard printer. Set printer properties via *File -> Standard configuration -> standard.*
Insert vector graphics	Opens the dialog for inserting an external vector graphic.
Properties	Opens the property window for the selected element.
Help	Opens online help.

9. XML-Export and XML-Import

EXPORTING SCREENS

The included variables and functions are exported with the screens. The export file for the screens (on page 9) has the following sections:

- Screens
- Frames
- Symbols
- Variables
- Functions

To export screens:

1. highlight the desired screens
2. select **Export selected XML...** in the context menu of the detail view
   Alternatively: select, in the context menu of the **Screen** node, the **XML export all...** command
3. the selection dialog for the saving location is opened
4. select the desired saving location
5. assign a name
6. pay attention to file type XML
7. confirm by clicking on the Save button

The selected screens are exported to an XML file and can be imported in this or in other projects at any time.

IMPORTING SCREENS

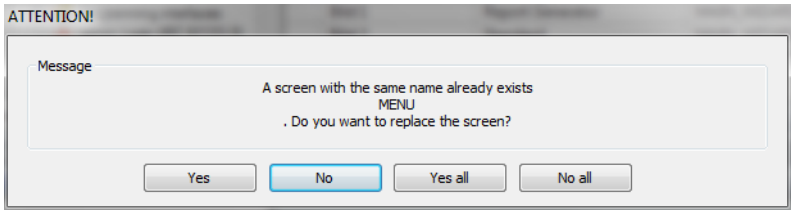
To import screens:

1. in the context menu of node **Screens** or in the detail view **Screens** select the **Import XML** command
2. the selection dialog for the saving location is opened
3. navigate to the saving location of the desired XML file
4. Select the desired file
5. confirm the import by clicking on the **Open** button

The screens are imported.

NAME CONFLICTS DURING THE IMPORT

At the import of a screen the name (attribute **ShortName**) which is stored in the XML file is used for the name of the screen which is created. If a screen with the same name already exist, the import is halted and an error message is displayed:



Parameters	Description
Yes	The screen described in the error message is replaced by the screen from the XML file at the import. If a new conflict occurs, the error message is displayed again.
No	The screen described in the error message is not imported. The existing screen is maintained. If a new conflict occurs, the error message is displayed again.
Yes, all	The screen described in the error message is replaced by the screen from the XML file at the import. This setting is used automatically when errors due to using the same name occur. All affected screen are replaced.
No, all	The screen described in the error message is not imported. The existing screen is maintained. This setting is used automatically when errors due to using the same name occur. All affected screen are maintained.

Note: Only the **ShortName** is used. The file name does not matter for the name of the screens which should be imported. Thus it cannot be used to solve the conflict.

VARIABLES AND FUNCTIONS IN SCREENS

Dynamic elements (on page 30) and Frames (on page 242) are automatically imported together with the screens. Variables and functions have to be imported manually before from the same file.

Only variables and functions of the first level are treated, i.e. variables and functions that are directly linked to the screen. As on the one hand variables can be linked to functions (e.g. setting values) but on the other hand functions can be linked to variables (e.g. limit functions), it can be necessary to first import the variables, then the functions, and then the variables again. Then all links should work correctly.

Example: A screen contains a button with a function **`Send value to hardware`** to a variable. The function is imported with the screen but not the variable.

> **Info**
>
> *Here we recommend using the XML Import Wizards.*

10. Screen elements

You use two types of screen elements in zenon:

1. static vector elements
2. Dynamic elements

You have many key combinations available when editing screen elements. You can find details in the help, in the graphics key combinations (on page 221) section.

In this section, you can find information on:

- Insert into the main window
- Define properties
- Change element size
- Rotate element
- Sequence in Runtime

- Vector elements in zenon
- Dynamic elements in zenon

INSERT INTO THE MAIN WINDOW

The selected element can be selected several time in the main window and arranged there. Which element is selected and therefore active is shown by a symbol under the peak of the mouse pointer. To draw the template:

1. press the left mouse button
2. move the mouse
3. release the mouse button

Exception: Polyline, Polygon and tube

- A mouse click inserts a new supporting position
- A double-click or the `Esc` key stops pulling

Elements can be freely rotated with the mouse; to do this:

1. click on the element with the mouse
2. rotate it with the yellow sizing handle

The touch point and therefore the rotation point can be set in `Reference point` properties.

DEFINE PROPERTIES

For elements in the main window, the properties are set in Window Properties. The corresponding element must be activated in the main window to do this.

You can also select several elements at the same time and assign these common properties. For this, the following applies:

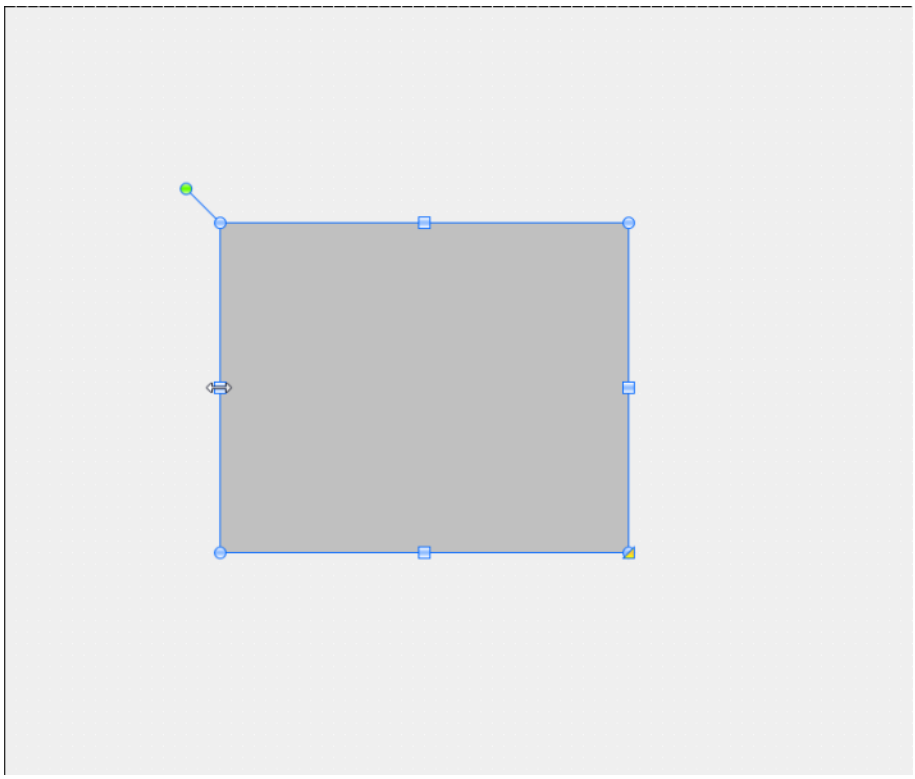
- Different values for different properties are marked in red.
- Properties which cannot be changed jointly are deactivated.

You also receive help instructions for the properties when the elements are created. To do this, activate the help window in zenon.

CHANGE ELEMENT SIZE

In order to change the size of an element, you can use the mouse, keyboard, properties or a dialog:

- With the mouse:
  a) Click on one of the handles of the element frame
  b) drag the element to the desired size
- With the keyboard:
  - Activate the desired sizing handle (mouse pointer must be over the sizing handle)
  - Drag it with the cursor keys:
  - By pressing a cursor key you change the position by 1 pixel.
  - With the `Shift` key: + cursor: By pressing a cursor key you change the position by 10 pixel.
- With the property window: In the property group `Position` you can enter the position and the size of the element accurate to the last pixel.
- With a dialog: Double clicking a handle opens a dialog in which you can enter the position accurate to the last pixel.



ROTATE ELEMENT

Elements can be rotated freely. The pivot point is set with the properties in the group `Reference point`.

You set the rotation angle as follows:

- With the mouse: Click on the element and rotate it with the help of the green handle.
- With the property window: With property `Rotation angle [°]` in group `Position`.
- With a dialog: A dialog is opened, in which you can enter the rotation angle, by double clicking the green handle.

> **Info**
>
> *Window-based screen elements such as ActiveX, lists or combobox/listbox cannot be rotated in Runtime. Configuration in the Editor is possible for ActiveX, but this has no effect on the display in Runtime.*

SEQUENCE IN RUNTIME

In Runtime, the following sequence is generally applicable for the display of elements, from top to bottom:

- WPF element, always in the foreground
- ActiveX Element, always in the foreground, except if it is overlaid by a WPF element
- Dynamic elements and vector elements

COVERED ELEMENTS

The following applies for overlaid elements:

- If a vector element overlays a dynamic element:
  - the dynamic elements or the overlaid part is not displayed
  - The dynamic elements can be operated, even if they are fully covered.
- If several dynamic elements are overlaid, only the visible (clickable) parts can be operated.

> **Info**
>
> *Elements that are covered in Runtime can be controlled with the keyboard. For details on configuration, see the Runtime manual, Defining sequences in frames chapter.*

VECTOR ELEMENTS IN ZENON:

You create vector elements in zenon with the character editor. Select the desired element in the Elements toolbar (on page 22) or in the drop-down list and place it in the main window using the mouse. Configure the properties with the **Properties** window. The element must be selected in the main window to do this.

Element	Properties
Ellipse (on page 97)	With frame and fill color, type of frame and fill pattern
Circle (on page 97)	Select the ellipse symbol and hold down the Shift key when drawing; with frame and fill color; type of frame and fill pattern
Arc of a circle (on page 98)	With definable line thickness, color and type
Segment of a circle (on page 99)	With definable line thickness, color and type
Line (on page 100)	With definable line thickness, color and type
Polygon (on page 110)	With frame and fill color, type of frame and fill pattern
Polyline (on page 112)	With line type and color (also filling pattern and color for an area description)
Square (on page 114)	select Rectangle symbol, hold down the Shift key when drawing; with frame and fill color; type of frame and fill pattern
Rectangle (on page 113)	with frame and filling color, type of frame and fill pattern
Rounded rectangle (on page 113)	with frame and filling color, type of frame and fill pattern
Tube (on page 114)	With frame and fill color, type of frame and fill pattern
Static text (on page 116)	With font color and selection of the type of font

IMPORTING YOUR OWN VECTOR GRAPHICS

Vector elements that have been created in external programs can be imported as a screen and included. To do this, go to *Files-> Graphics -> Add file in the project manager*.

DYNAMIC ELEMENTS IN ZENON

You create dynamic screen elements in zenon with the character editor. Select the desired element in the Elements toolbar (on page 22) or in the drop-down list and place it in the main window using the mouse. Configure the properties with the Properties window. The element must be activated in the main window to do this. Graphic elements can be activated directly here through variables. For many elements, a configuration dialog is shown when they are being created.

Dynamic elements in zenon:

Element	Function
ActiveX (on page 39)	Inserts any desired ActiveX elements. These must already be installed on the computer.
Bar Graph (on page 41)	Displays the size of a signal in bar form. In doing so, the length of the bar changes with the signal size.
Command Input Element (on page 43)	Makes it possible to send commands for the Command Input module. Requires license forSICAM 230.
Screen Alarm Button (on page 43)	Enables alarming with color and flashing via the screen. Requires license forSICAM 230.
Bitmap Button (on page 46)	Links a graphics file (pixel graphics: *.bmp, *.jpg, *.gif, or *.png or vector graphics: *.wmf) with a button. **Attention!** This element is only available if the project property `Create RT files for` has been set to lower than 6.50. This function has been integrated into the element button from version 6.50.
Button (on page 43)	Enables actions such as execute function or target value to be instigated. It can display text and graphics and be created as a transparent object via other elements. **Attention!** If the project property `Create RT files for` is set to lower than 6.50, only text output is displayed in runtime. Graphics and the `invisible` function must then be configured with the **`bitmap button`** and the **`invisible`** button elements. You then find these two elements at the bottom of the **`elements`** drop-down list.
Combined element (on page 51)	Displays statuses of variables via symbols, screens and texts in graphic form also be used as a switch or as a button.

Combo/List Box (on page 88)	Displays a text in a combo box or a list box regardless of variable value. When an entry is selected, the attendant value is sent or the attendant function is executed.
Dynamic Text (on page 96)	Displays the value of a string variable in alphanumeric form or the current limit text of numeric variables.
Message element (on page 101)	Reads text from a text file, depending on two variables, and displays this.
Multi-binary element (on page 103)	Enables, independently of variable values, a graphic to be displayed or symbols to be colored and a status text to be provided.
Switch (on page 116)	Displays values of a binary variable and modifies these.
Move Symbol (on page 118)	Enables a symbol to be moved, rotated and changed in size regardless of variable values.
Trend Element (on page 118)	Displays values in the form of trend curves.
Clock (on page 120)	Displays the current time and date.
Universal slider (on page 122)	Provides sliders in different graphical forms.
Invisible Button (on page 49)	Transparent object, with which actions such as execute function or define set value can be triggered. **Attention!** This element is only available if the project property `Create RT files for` has been set to lower than 6.50. This function has been integrated into the **`button`** element from version 6.50.
WPF Element	Displays WPF-XAML files in zenon.
Numerical value (on page 191)	Displays the value of a variable in numerical form.
Indicating instrument (on page 192)	Displays the value of a variable in the form of an analog measuring tool with a indicator as display element.
Status Element (on page 193)	Transfers properties from linked variables to a symbol and displays statistical limit texts.

DEFINE PROPERTIES

For many dynamic elements, a configuration dialog opens to select variables and/or functions that define the elements in runtime. The properties can also be set independently of the configuration dialog in the properties window. They can be changed at any time here. The corresponding element must be activated in the main window to do this.

You can also select several elements at the same time and assign these common properties. For this, the following applies:

- Different values for different properties are marked in red.
- Properties which cannot be changed jointly are deactivated.

You also receive help instructions for the properties when the elements are created. To do this, activate the properties list in zenon.

DISPLAY IN THE MAIN WINDOW

You can obtain information on a dynamic element in the main window by:

Tool tip when the mouse is moved over it:

- Element type
- Element name
- Function name/function type
- Linked variables

In the status bar on activation:

- Mouse pointer position
- Element type
- Element name
- Start point
- Width/height

10.1 .NET Controls

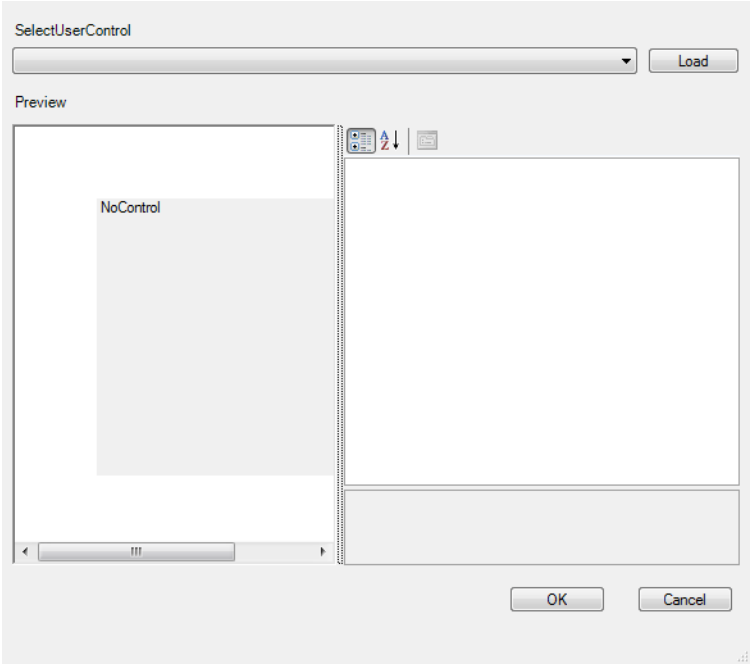
The ActiveX control `CD_DotNetControlContainer.Container` makes it possible to use any .NET Windows Forms Controls as dynamic elements in zenon. In doing so, all functions implemented in the .NET control are supported. It can be used with all zenon versions from 5.50 on which support ActiveX.

Note: .NET Framework 3.5 must be available.

> **Info**
>
> *You can read more about the use of .NET with zenon in the Programming interfaces manual in the .NET chapter.*

To use the ActiveX control **CD_DotNetControlContainer.Container**:

1. create a dynamic element of the type ActiveX (on page 39)
2. select **CD_DotNetControlContainer.Container** from the list of available controls
3. click on button **Properties**
4. the dialog for configuring the control opens



Parameters	Description
Load	Opens the file manager for selecting a **.NET Control Assembly**. Note: The **.NET Control Assembly** must be in the same folder as Runtime. It is always the absolute path to the **.NET Control Assembly** that is saved, e.g.: C:\Controls\Assembly.dll.
SelectUserControl	Selection of the **.NET Controls** from the ones available in the selected **.Net Control Assembly**.
Preview	Displays the absolute path to the **.NET Control Assembly**. **Field left below**: Preview of the control. **Field right below**: Property list for the selected **.NET Control**.
Symbol: By categories	Displays properties sorted by categories.
Symbol: Alphabetical	Display properties in alphabetic order.
Symbol: Property pages	Displays property pages.
OK	Confirms configuration. This is then saved in the zenon ActiveX element as XML stream.
Cancel	Cancel configuration.

10.2 ActiveX

You connect ActiveX elements in projects using the ActiveX dynamic element.

> **Info**
>
> *You can read more about the use of ActiveX with zenon in the Programming interfaces manual in the ActiveX chapter.*

To create ActiveX in a screen:

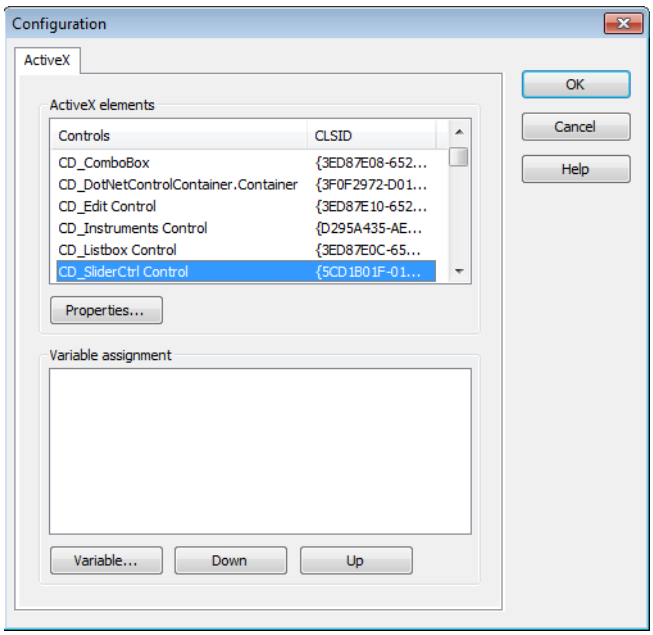
1. select the ActiveX symbol in the **Elements** tool bar or in the **Elements** drop-down list
2. select the start point in the main window
3. pull the control element open while pressing and holding the left mouse button

4. a configuration dialog opens, in which you can select the element and configure it
5. The shape, size and position can be changed at any time by pulling the vertexes or moving the element
6. if you press and hold `alt` while pulling the outer corner points, the change is carried out symmetrically

You define individual properties of the element in the property window. In order to do this, the element must be active (mouse click) in the main window. To subsequently select or change controls, open the configuration dialog by double-clicking on the properties menu item in the context menu.

> **Attention**
>
> *Only ActiveX elements that are installed and registered on the computer are shown. Please ensure that all ActiveX elements are also installed on all Runtime systems on which you use the project.*
> *You can also use Remote Transport to register ActiveX elements on remote systems. See: Define files (copy&register)*

ACTIVEX CONFIGURATION



Element	Function
ActiveX elements	List of all ActiveX elements available.
Properties	Opens a configuration dialog for the ActiveX highlighted, if available. You can obtain help and information about the configuration from the manufacturer of the respective ActiveX control.
Linking variables	List of variables linked to the ActiveX element.
Variable	Opens the zenon Dialog to select variables if it is possible to link variables.
Down	Sorts highlighted variables downwards.
Up	Sorts highlighted variables upwards.

POSITION CHANGE IN RUNTIME

ActiveX controls work with untransformed mouse coordinates in Runtime. If an element is rotated, stretched or otherwise changed in terms of its size or position, the control no longer fits and continues to be executed at the original position. The transformed element can therefore no longer be operated at its transformed position using the control.

Info

Regard the difference between windowed and windowless ActiveX elements.

Windowed ActiveX elements (e.g. all Microsoft ActiveX Controls) are always in teh foreground.

In contrast you can place for example a dynamic element over windowless ActiveX elements.

10.3 Bar Graph

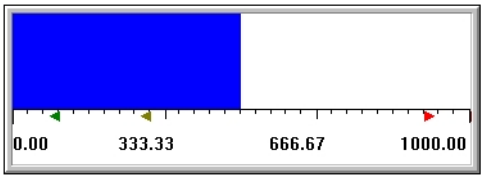
You show the numerical size of a signal with the **Bar graph** dynamic element. The length of the bar changes with the signal size in runtime.

To create **Bar Graph** in a screen:

1. select the Bar graph symbol in the **Elements** tool bar or in the **Elements** drop-down list

2. select the start point in the main window
3. pull open the element while pressing and holding the left mouse button
4. the dialog to assign a variable (on page 238) opens
5. The shape, size and position can be changed at any time by pulling the vertexes or moving the element
6. if you press and hold **alt** while pulling the outer corner points, the change is carried out symmetrically

You define individual properties of the element in the **Properties window**. In order to do this, the element must be active (mouse click) in the **Main window**.



To edit the linked variable:

1. open the variables in the Project Manager Detail view,
2. drag the desired variable onto the element while holding the left mouse button
3. the previous variable is substituted by the new one

COLOR-CODED DISPLAY OF THE LIMIT VIOLATION

At limit violations the color from the variable can color the whole area or only the area starting from the limit violation. This functionality is controlled via property `Explicit` in group `Representation`.

This property affects only the display of values with defined limits from variables. Limits from reaction matrices are not considered.

`Active`: If a limit of the linked variable is violated, only the part of the display which reaches into the limit violation is displayed in the color of the respective limit.
If the variable has several defined limits, the respective areas of the display are always displayed in the color of the corresponding limit.

`Inactive`: If a limit of the linked variable is violated, the complete display is displayed in the color of the violated limit.

Default: `inactive`

10.4 Command Input Element

Makes it possible to send commands for module Command. Requires license forSICAM 230.

You can find instructions on how to use elements in the help at SICAM 230.

10.5 Screen Alarm Button

Attention: The **`screen alarm button`** requires a license for SICAM 230. You can find further instructions on how to use elements in the help at SICAM 230.

10.6 Button

You create a freely adjustable button the the dynamic element `button` which you can use interactively in the screen to either carry out a function or as a switch for a binary variable.

The button can

- display a graphics file
- animate a graphics file
- be displayed invisibly

> **Info**
>
> *Buttons are subdivided into text button, bitmap button and invisible button in versions up to and including version 6.22. From version 6.50, all properties that can be configured with buttons are together in the element button.*

To create the dynamic element button in a screen:

1. select the Button symbol in the **Elements** tool bar or in the **Elements** drop-down list
2. select the start point in the main window
3. pull open the element while pressing and holding the left mouse button
4. The Selection dialog functions (on page 237) open
5. select the desired function
6. Define the desired properties in the properties window

   Hint: You can round (on page 195) the corners as you wish
7. You add variables by dragging & dropping

LABELING

To label the button:

- Click in the element.
- enter the text
- define the text properties in the Representation node

UNDERLINE

To underline text, place a & in front of the text to be underlined.

&Text becomes:

If you would like to use the character '&' in text, enter it twice:
1 && 2 becomes 1 & 2.

LINE BREAK

To define a line break, use the string **\n**.

Example:

Line 1 \n

Line 2

BUTTON AS SWITCH

To use the button as a switch:

1. activate the `Switch` property in the `Write set value` node
2. You activate settings to use the button as a momentary switch with the `Momentary switch` property
3. Link the button with a binary variable (node `Variable / function`, property `Variable`)

BUTTON TO EXECUTE A FUNCTION

To link a button to a function:

1. deactivate the `Switch` property in the `Write set value` node
2. Link the button with a binary variable (node `Variable / function`, property `Function`)

BUTTON WITH GRAPHICS

To put graphics on the button:

1. activate the element
2. define the desired graphics for the different switch states in the `Representation` node via `Graphics file` properties and "Pressed button graphics". You can use the following graphics files: Pixel graphics: *.bmp, *.jpg, *.gif, or *.png or vector graphics: *.wmf.

A button with graphics has a thin 3D frame line as standard, which covers a pixel on each side of the graphic. If the graphics are to be visible, then:

- In the button properties, select the `Transparency border color [%]` property in the `Color` group
  - with the value `100`

or

- Activate the `Transparent` property to deactivate background color and frame color

This causes the 3D frame line to not be displayed and the whole graphic is visible.

INVISIBLE BUTTON

to define an interactive, transparent user interface:

1. activate the element
2. definite the `Transparent` property in the `Color` node
3. remove all labeling from the button
4. Access to Runtime:
   - The mouse pointer changes when the mouse is moved over the element
   - over a defined key combination (`Runtime` node, `Key combination` property)

> **Info**
>
> *The status of the variable cannot be displayed for the invisible button.*

10.6.1 Animate graphics

GIF graphics files can be animated in the dynamic element button:

1. assign a GIF file to a button in the `Representation` node via Property `Graphics file`
2. select the property `GIF animation` in the same node
3. choose between `Always` or animation only if a Boolean variable has a value of 1: To do this, link the properties of both properties `Variable`

10.6.2 Bitmap Button (up to version 6.22 only)

Links a graphics file (pixel graphics: *.bmp, *.jpg, *.gif, or *.png or vector graphics: *.wmf) with a button.

If the project property Create RT files for is set to small 6.50, the element button can only display texts in runtime. Graphics must be configured by means of the bitmap button in this case.

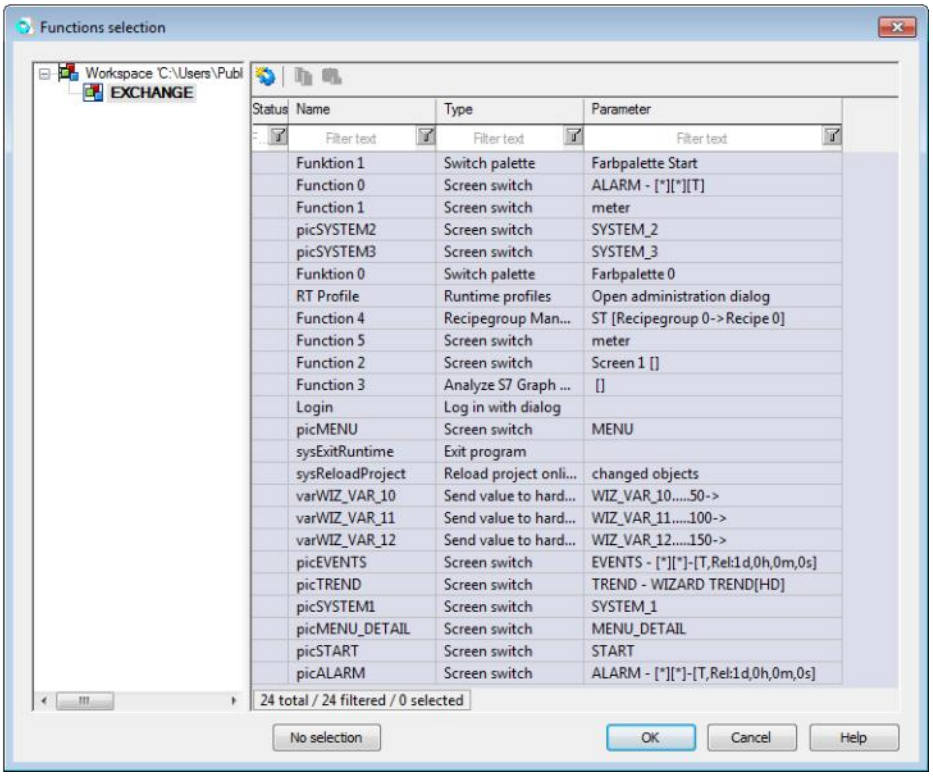
Engineering buttons for different Runtime versions:

Button for	Create RT files for < 6.50	from 6.50
Text	Button	Button
Graphics	Bitmap Button	Button
invisible	Invisible Button	Button

to activate a bitmap button:

- select the Bitmap button symbol in the **Elements** tool bar or in the **Elements** drop-down list
- select the start point in the main window
- pull open the element while pressing and holding the left mouse button
- a dialog opens, in which you can link the desired function to the element
- The shape, size and position can be changed at any time by pulling the corner points or moving the element
- if you press and hold the alt key while pulling the outer corner points, the change is carried out symmetrically
- You define individual properties of the element in the property window. In order to do this, the element must be active (mouse click) in the main window.

In order to select functions, a filtered dialog is opened:



Element	Description
Project tree	Definition of the project from which the function should be selected.
Selection window	Selection of the function.
No selection	Depending on the element: ▶ the dialog is canceled ▶ already linked functions are deleted

Hint: The size of this dialog can be adjusted. The dialog size and position are saved.

CHANGE LINKED FUNCTION

1. manually
   - in the Project Manager´s Detail view open the functions
   - drag the desired function on the element while holding the left mouse button
   - the previous function is substituted by the new one

2. automatic
    - Select the entry Replace links from the context menu. For more details see chapter Substitution of variables and functions in dynamic elements (on page 225).

GIF graphics files can be animated in the dynamic element button:

1. assign a GIF file to a button in the `Representation` node via Property `Graphics file`
2. select the property `GIF animation` in the same node
3. choose between `Always` or animation only if a Boolean variable has a value of 1: To do this, link the properties of both properties `Variable`

10.6.3 Invisible button (up to version 6.22 only)

If the project property `Create RT files for` is set to small 6.50, the element button can only display texts in runtime. Invisible buttons must be configured with the invisible button element in this case.

Engineering buttons for different Runtime versions:

Button for	**Create RT files for < 6.50**	from 6.50
`Text`	Button	Button
`Graphics`	Bitmap Button	Button
`invisible`	Invisible Button	Button

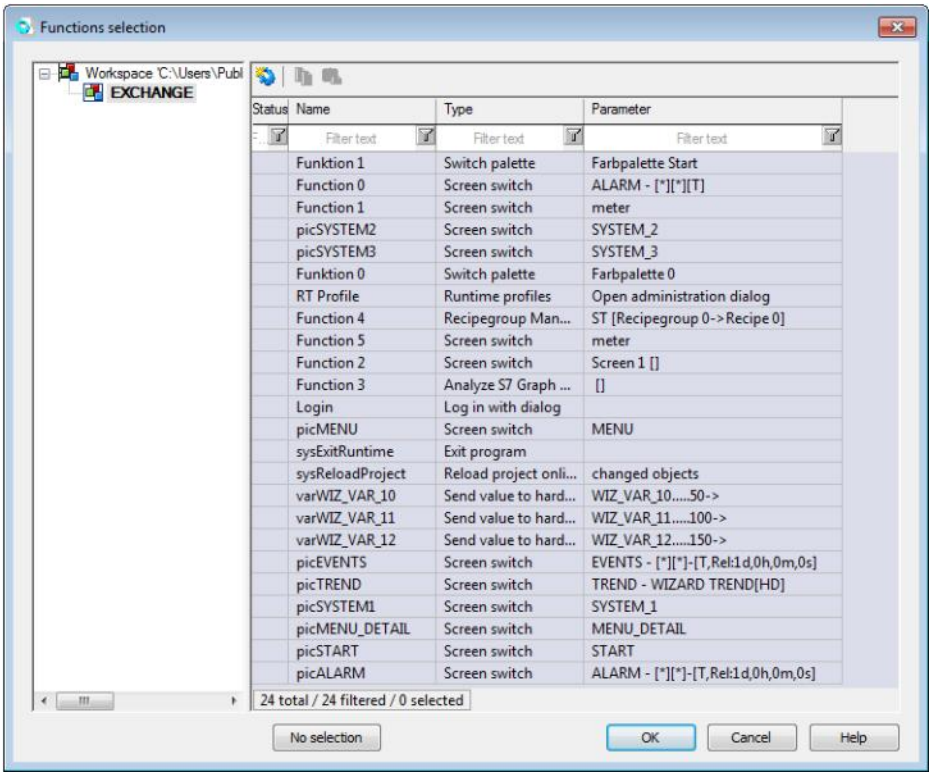
The dynamic element Invisible Button offers the possibility to define an interactive transparent user interface which is linked to a function call.

To activate an invisible button:

- select the `Invisible Button` symbol in the **Elements** tool bar or in the **Elements** drop-down list
- select the start point in the main window
- pull open the element while pressing and holding the left mouse button
- a dialog opens, in which you can link the desired function to the element

- The shape, size and position can be changed at any time by pulling the vertexes or moving the element
- if you press and hold the alt key while pulling the outer corner points, the change is carried out symmetrically
- You define individual properties of the element in the property window. In order to do this, the element must be active (mouse click) in the main window.

In order to select functions, a filtered dialog is opened:



Element	Description
Project tree	Definition of the project from which the function should be selected.
Selection window	Selection of the function.
No selection	Depending on the element: ▸ the dialog is canceled ▸ already linked functions are deleted

Hint: The size of this dialog can be adjusted. The dialog size and position are saved.

CHANGE LINKED FUNCTION

1. manually
    - in the Project Manager´s Detail view open the functions
    - drag the desired function on the element while holding the left mouse button
    - the previous function is substituted by the new one
2. automatic

Select the entry Replace links from the context menu. For more details see chapter Substitution of variables and functions in dynamic elements (on page 225).

> **Info**
>
> *The status of the variable cannot be displayed for the invisible button.*

10.7 Combined element

The combined element is a dynamic element that can be used universally and can adopt the most varied graphic characteristics:

- Graphics
- Symbol from library
- Symbol embedded in screens
- Text

The form of the display is defined through statuses. Variable values of the main value, values of further values and status information from all linked variables can all serve as statuses. Links can be created as desired via formulas.
Statuses of the main variables (limit attributes) can be transferred directly to the combined element. The combined element also supports the setting of set values in the form of a switcher/momentary switch as well as the sending of and sending functions and serves to implement technical process-related elements for Automatic Line Coloring.

If set values are set via command input and a response variable is set in the combined element dynamic element, it can be set regardless of the setting of the `Setting values active` property. All action

buttons in the command input screen that trigger a direct modification of the response variable are then set to invisible.

CREATE A COMBINED ELEMENT

To create a combined element in a screen:

1. Click on the Combined Element symbol in the Elements toolbar (on page 22) and drag the element to the main window with the mouse.
2. The variable selection dialog opens, filtered for numeric variables.
3. Select the desired variable
4. The Assistant (on page 54) for combined element settings opens and guides you through the basic settings. If you prefer to define all settings manually, click on cancel now. You can find details in the Assistant. (on page 54) chapter

When setting up manually, define the `Type of display` in the `Representation` node. The symbol from the library is given as a default. You can define screen data and text according to the settings.

You define additional statuses either in the assistant via the New Status or in Combined Element Properties in the display with property node `Configuration and test`.

Functions and variables can be assigned to the combined element by dragging & dropping. In doing so, the respective variables/functions are each replaced by the new ones.

> **Attention**
>
> *Symbols in a "Combined" element cannot be operated.*
>
> *Exception: Symbols from the library can be created as clickable buttons (on page 59).*

LINK RULES

If you have selected `Symbol from Library` as a display type, you can enter link rules. You open the dialog for entering the element via the `Configuration and test` property. After selecting the symbol via the Symbol Select button, you can create and edit linking rules. These rules are saved for every status.

Example

Variables/functions contained in the symbol:
Var_SWITCH_0010
Var_TEMPERATURE_0010
Fct_ShowTrend_0010
Fct_Help_0000

The variables and functions of the symbol are now replaced with the following rule:
Source: *_0010
replace with: _0020

result of linking rule:
Var_SWITCH_0020
Var_TEMPERATURE_0020
Fct_ShowTrend_0020
Fct_Help_0000

With the help of the linking rule, all variables and functions can be exchanged. If a variable or function does not fit into the scheme, that link cannot be replaced.

When the runtime files are created, the respective links are set for every status, according to the rule.

Example

You would like to graphically display if a motor is switched on and in which direction it is turning.

Information about run direction and on/off state comes from the variables on the control side. The status text displays whether the motor is turning and how it is turning.

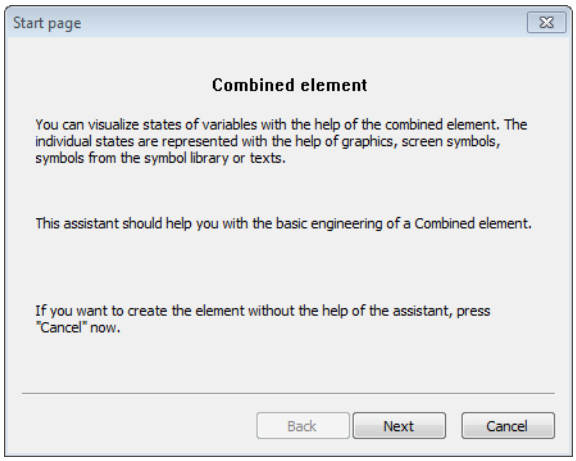
- Select Properties -> Display Type -> Status Text.
- Open the Enter Element dialog by clicking on `Configuration and test`.
- Enter the status text which you want to allocate to the respective value.

It is subsequently displayed in your motor during runtime, for example: 'left', 'right' or no text at all if the motor is not turning.

You will find more information about the evaluation of the status bits in the chapter Statuses (on page 63).

10.7.1 Assistant

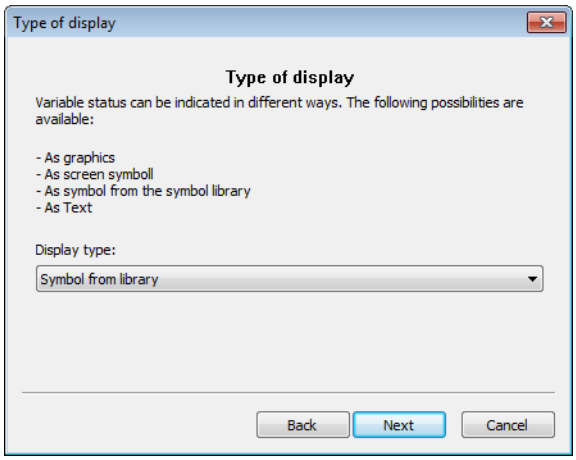
An assistant is available for the basic configuration of the combined element. This opens as soon as you have selected a variable for the combined element.



In the first step, you select the display type. The following display types are available:

- Symbol from Library
- Status Text and Screen Symbol
- Bitmap and Screen Symbol
- Screen Symbol Only

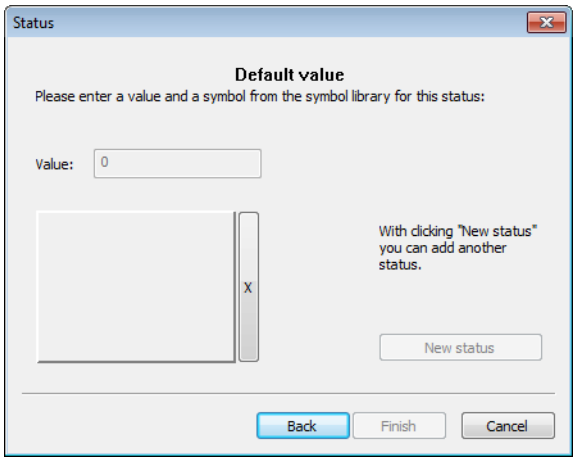
> **Info**
>
> *You can define additional statuses in all status selection windows. The New Status button is available for this.*



Default: Symbol from library.
This setting is also displayed as standard in the properties if you create the combined element without the assistant.

You can define statuses as soon as you have selected a type of display. The first status is always the default status.
Hint: A value cannot be entered for the default status; a value must be entered for all other statuses. You can only edit one status at a time.

SYMBOL FROM LIBRARY

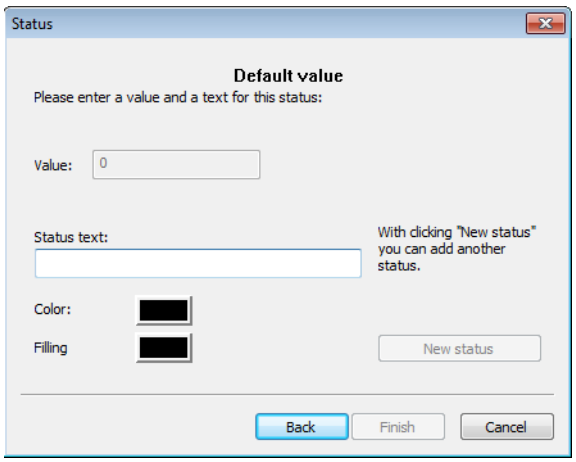


Click on the empty window at the bottom left to select a symbol from the global symbol library or from the project library. You can delete the selected symbol again by clicking on the X next to the window.

Setting	Description
Value	Numerical value at which the status becomes active. Default: no value From 2nd status: Entry required.
New status	Creates a new status for the combined element.

> **Info**
>
> *For `Symbol from library`, clickable buttons (on page 59) can be created in any form.*

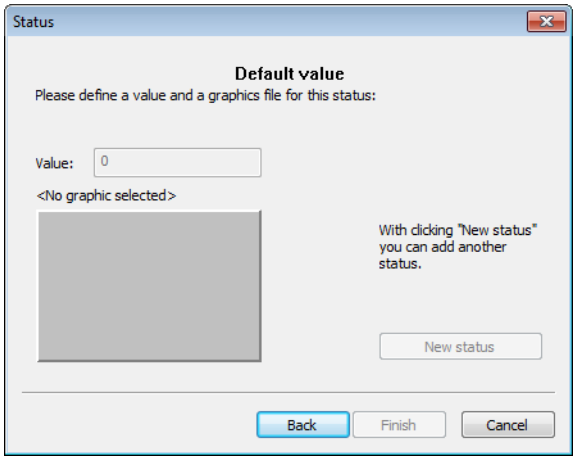
STATUS TEXT AND SCREEN SYMBOL



Each condition can be labeled with a text and with a foreground and background color.

Setting	Description
Value	Numerical value at which the status becomes active. Default: no value From 2nd status: Entry required.
Status text	Text as it is displayed in runtime.
Color	Font color of the status text.
Filling color	Background color of the status text.
New status	Creates a new status for the combined element.

BITMAP AND SCREEN SYMBOL



Click on the <No graphic selected> button at the bottom left to select a screen file (*.bmp, *.gif, *.jpg, *.png etc.). You have two options for screen selection:

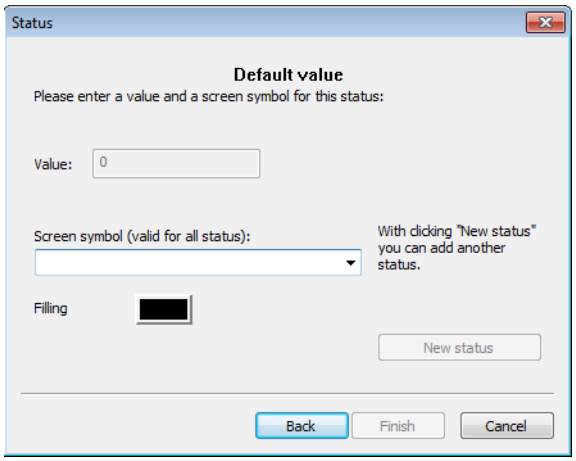
1. Pictures from the project:

   Pictures which you have already stored in the graphics directory of a project (see chapter on Editor -> Project Manager -> Files) are offered to you automatically by the assistant.

2. Absolute path to other directories:
   - Activate direct file selection for this
   - The window switches to direct selection
   - Click on the button ...
   - Select the corresponding file with the explorer

Setting	Description
Value	Numerical value at which the status becomes active. Default: no value From 2nd status: Entry required.
New status	Creates a new status for the combined element.

SCREEN SYMBOL ONLY



You can give a common symbol for all combined element statuses.
This symbol:

- must already be present on the screen
- applies for all statuses
- can be labeled for each status with its own filling color

Setting	Description
Value	Numerical value at which the status becomes active. Default: no value From 2nd status: Entry required.
Screen symbol	Drop-down list to select a symbol. If no symbols are displayed, then no symbols are present in the current screen.
Filling color	Names the allocated status.
New status	Creates a new status for the combined element.

10.7.2 Clickable buttons in any form

In the combined element, clickable buttons can also be created in any desired form.

> **Info**
>
> *This option is available from version 6.51. For projects for earlier versions, the property is treated as* `inactive`.

To create clickable buttons in any form desired:

1. Create a combined element in the **`Symbol from library`**attributes
2. In combined element properties, activate the `Symbol form defines the click area` property in the `Representation` node

> **Info**
>
> This property does not work under Windows CE, because the required functionality is not available under Windows CE.

ELEMENTS

There can be further symbols or combined elements in the symbol. These are triggered correctly corresponding to the click area. For symbols, the element limits are used completely normally, with a

combined element this depends on the `Symbol form defines the click area` property, either the individual click areas of the elements or the environment rectangle of the combined element.

The following elements have their own special click area:

- Rounded rectangle (present as its own vector element up to version 6.50)
- Button
- Combined element with "Symbol form for click area" property
- Circle
- Segment of a circle
- Arc of a circle
- Line
- Pipeline
- Polygon
- Polyline
- Rectangle
- Static text
- Symbol (correct click areas of the symbol elements are used)

All other elements have their surround as a click area.

SPECIAL ATTRIBUTES

Special attribute	Effect
Are never treated specially: ▸ Type of line ▸ Filling pattern ▸ Transparencies of all type (except text element)	Areas are always clickable.
Transparency for static test: Elements behave as though the element did not exist. As a result, it is possible to name elements without unintentionally enlarging the click area.	Does not influence the click area. Hint: If a transparent text element is to influence the click area: ▸ Deactivate the `Transparent` property ▸ Place a transparent element over or under this manually
Button	Is used for the click area; `GIF animation` for "pressed" is not available here. This corresponds to the behavior with an inactive `Symbol form defines the click area` property.
Symbol protrudes from the combined element.	Symbol is displayed, but clicks are only interpreted within the combined element. **Comment:** If elements protrude outside symbol limits or symbols protrude out from the limits of a combined element, this can lead to display problems and blurring effects.

CLICK AREAS

In the Editor, areas larger than the click area are sometimes defined, in order to make these easier to click on. For example, a line with 1 pixel thickness in the editor can be selected using a width of 8 pixels.

The display of the click area is exact in Runtime. For example, a line of 1 pixel thickness is also only selectable from a width of 1 pixel.

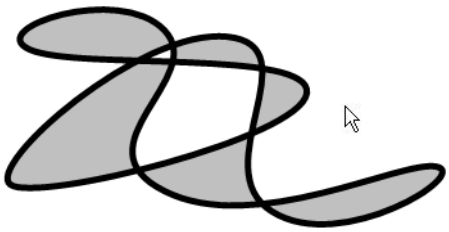
The calculation of the click area for the combined element works regardless of the `Graphics quality` property's setting. However, a different click area is calculated for the elements regardless of this setting.

Info

If the mouse pointer is above the combined element, it is analyzed whether the cursor is in the click area of the element. The load placed on the CPU can then be high with a large number of elements in the combined element symbol.

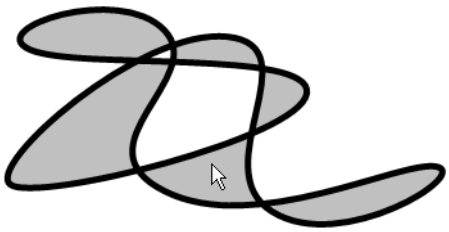
Example clickable buttons

EXAMPLE 1



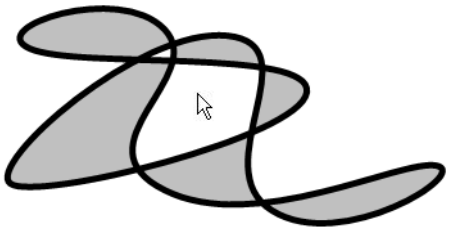
status Symbol form defines the click area	Action
active	No click
inactive	Click

EXAMPLE 2



status Symbol form defines the click area	Action
active	Click
inactive	Click

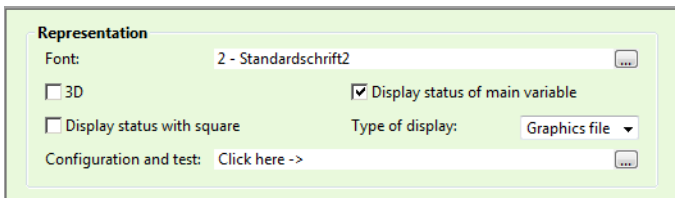
EXAMPLE 3



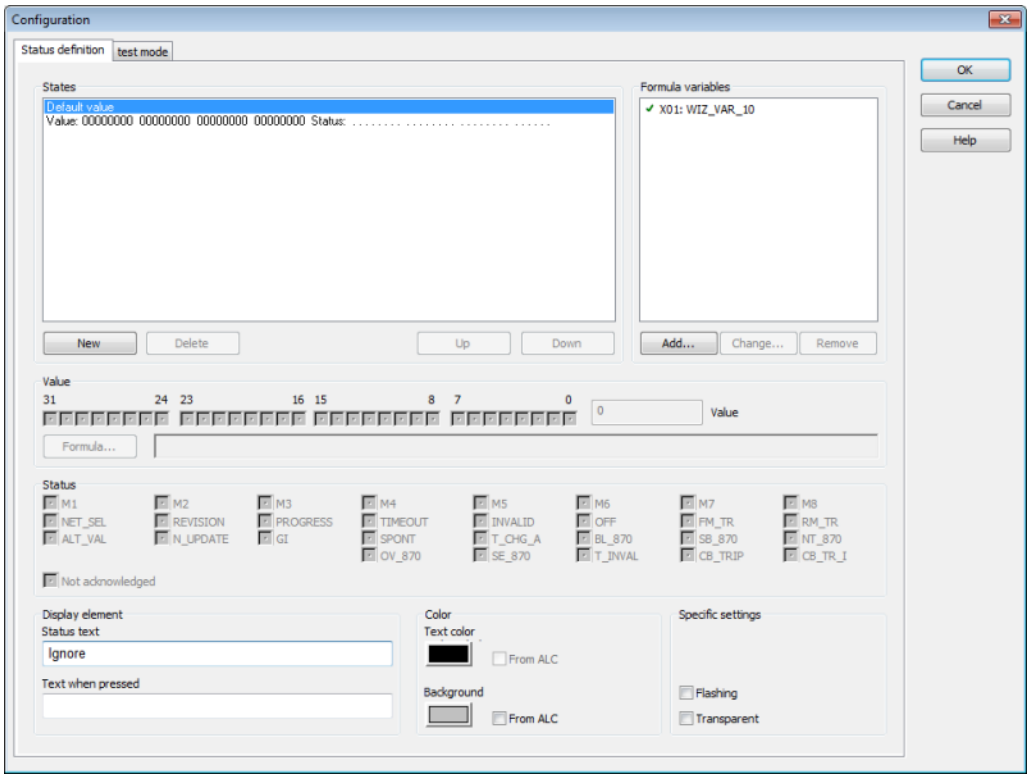
status Symbol form defines the click area	Action
active	No click
inactive	Click

10.7.3 Statuses

You get to the statuses window via the Configuration and test property in the Representation node.



Clicking on the button or on the text "Click Here" opens the window:



Here

- you edit existing statuses
- add the new statuses
- specify in which direction statuses are to be processed.

The settings in the upper part of the window are the same for all types of display. In the lower third, individual settings for the four different types of display are made.

COMMON SETTINGS

STATUS DEFINITION

Lists all defined statuses. A status can consist of values and status. The element is processed from the first status up to the last entry. The first complete match is shown. Exception: The Overlapping property (for symbols from the library) or Chained (for texts) is active.

Parameters	Description
Default status	The status defined when creating the element. It is always in the first position and can longer be changed or deleted.
Value	Shows the defined (32 bit) value.
Status	shows the defined status. An "F" at the end of the status display indicates a formula.
New	Defines new statuses. in doing so, values, status and formulas of the status currently highlighted are taken on as initial values.
Delete	Deletes the selected status. The default status cannot be deleted.
upwards/downwards	changes the order in which statuses are processed. The default status is always in the first position.

FORMULA VARIABLES

shows all variables that were defined for input in formulas. Existing variables are marked with a green tick, non-existent variables are marked with a red cross. Non-existent variables can be replaced by exiting variables via Change. The Add and Change buttons open the dialog to select variables (on page 238).

VALUE

Enables

- input of a value for a status via a matrix or numerically
- the definition of a formula

Parameters	Description
Value	Defining the value via a preset bit structure
Value	Enter the numeric value.
Formula	Opens the window to enter formulas (on page 83). In the formula variable area, defined variables are available. Linking is performed by logical, bitwise and comparative operators.

> **Attention**
>
> *For non-binary variables, the value is entered in the signal resolution. Settings in the measuring range or the non-linear value adjustment are not supported in this element.*

STATUS

Enables status information to be requested.

Attention: Most statuses can only be requested if the driver supports the status processing.

no.	Short term	Long name	Comment
0	M1	User defined status 1	
1	M2	User defined status 2	
2	M3	User defined status 3	
3	M4	User defined status 4	
4	M5	User defined status 5	
5	M6	User defined status 6	
6	M7	User defined status 7	
7	M8	User defined status 8	
8	NET_SEL	Select in the network	
9	REVISION	Revision	
10	PROGRESS	In process	
11	TIMEOUT	Timeout exceeded	
12	MAN_VAL	Hand value	Only available for formulas
13	M14	User defined status 14	Only available for formulas
14	M15	User defined status 15	Only available for formulas
15	M16	User defined status 16	Only available for formulas
16	GI	General interrogation	
17	SPONT	Spontaneous	
18	INVALID	Invalid	
19	T_CHG_A	Time change announcement	
20	OFF	Switched off	
21	T_EXTERN	Real time external	Only available for formulas
22	T_INTERN	Real time internal	Only available for formulas
23	N_SORTAB	Not sortable	Only available for formulas
24	FM_TR	Fault message transformer value	
25	RM_TR	Run message transformer value	

26	INFO	Information for the variable	
27	ALT_VAL	Alternative value	
28	RES28	Reserved	
29	N_UPDATE	Not updated	
30	T_STD	Standard time	Only available for formulas
31	RES31	Reserved	
32	COT0	Cause of transmission bit 1	Only available for formulas
33	COT1	Cause of transmission bit 2	Only available for formulas
34	COT2	Cause of transmission bit 3	Only available for formulas
35	COT3	Cause of transmission bit 4	Only available for formulas
36	COT4	Cause of transmission bit 5	Only available for formulas
37	COT5	Cause of transmission bit 6	Only available for formulas
38	N_CONF	Negative acceptance of Select by device (IEC60870 [P/N])	Only available for formulas
39	TEST	Test bit (IEC 60870 [T])	
40	WR_ACK	Writing acknowledged	Only available for formulas
41	WR_SUC	Writing successful	Only available for formulas
42	NORM	Normal status	Only available for formulas
43	N_NORM	Deviation from normal status	Only available for formulas
44	BL_870	IEC 60870 Status: blocked	
45	SB_870	IEC 60870 Status: substituted	
46	NT_870	IEC 60870 Status: not topical	
47	OV_870	IEC 60870 Status: overflow	
48	SE_870	IEC 60870 Status: select	
49	T_INVAL	Time invalid	Only available for formulas
50	RES50	reserved	
51	RES51	reserved	

52	RES52	reserved	
53	RES53	reserved	
54	RES54	reserved	
55	RES55	reserved	
56	RES56	reserved	
57	RES57	reserved	
58	RES58	reserved	
59	RES59	reserved	
60	RES60	reserved	
61	RES61	reserved	
62	RES62	reserved	
63	RES63	reserved	

Info

The short terms for the status bits have been unified for all languages since version 6.50. You can find more details and how you can use the old short terms in chapter Status bits - new short terms.

Not acknowledged: If option Flashing is active for the limit of the variable, the bit nor acknowledged is set. With the option `Acknowledge flashing also acknowledges alarm` in the alarm configuration, the alarm is acknowledged and the bit mot acknowledged is set to 0 (see also chapter Acknowledge flashing for alarms).

INDIVIDUAL SETTINGS

Individual settings for the 4 types of display of statuses is available in the subchapters:

Symbol from Library (on page 71)

Status Text and Screen Symbol (on page 73)

Bitmap and Screen Symbol (on page 75)

Screen Symbol Only (on page 76)

CONFIGURING THE STATUS FOR VARIABLES THAT HAVE NOT YET BEEN LINKED

Sometimes you must define a combined element but you do not yet have a real, existing variable available. For example, if the combined element is supposed to be used in a symbol of the symbol library and it the variable link will be created only after pulling it into the screen. In this case, you first work with a 'dummy variable', which you then subsequently replace with an existing variable. You can then define all statuses in the combined element if existing variables have not yet been linked. You have no access with dummy variables in the test mode only.

This is how you create a 'dummy' variable:

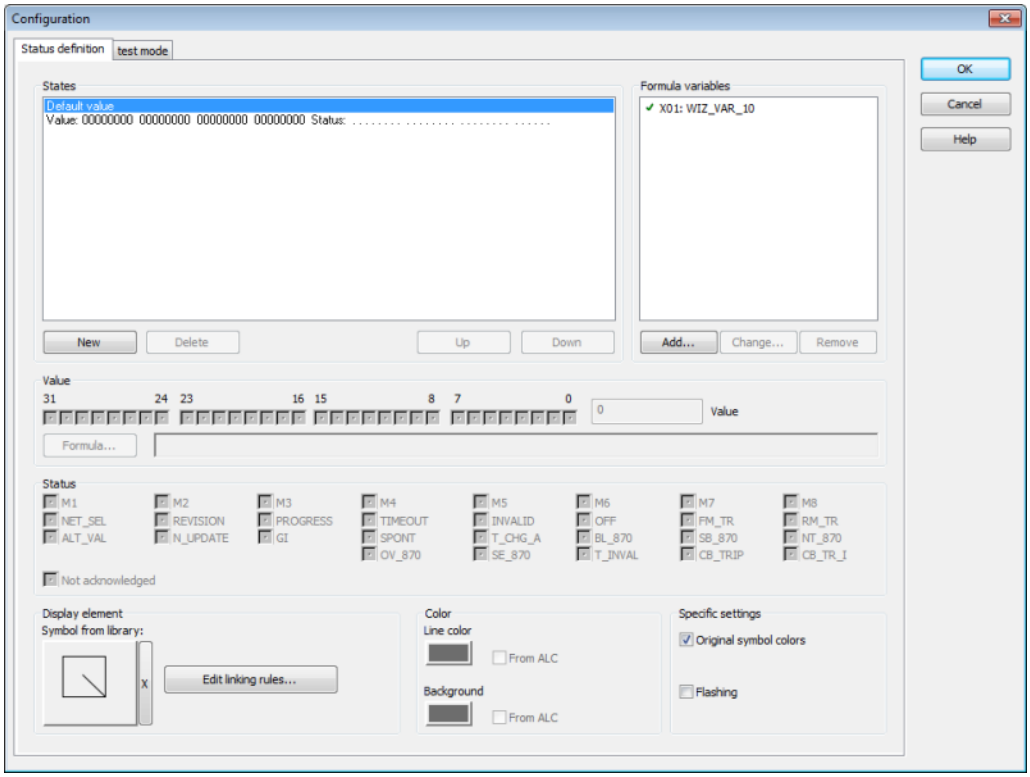
- Enter a symbolic name into the properties of the combined element in the `Variable` input field.
- Click in the `Representation` area in the `Configuration and test` field: The page statuses in the element input are opened.
- In the list of formula variables, existing variables are marked with a green check mark. Variables that do not exist are marked with a red X. You can add further variables:
- Use the Add button for existing variables.
  With Change, you can replace existing and non-existing variables with other variables.
- To add a non-existent variable, click on Add and then on No selection. You get a formula variable named after the following scheme: X00: <empty>. (00 stands for a two-digit number.)
  Highlight the variable and click in the <empty> area; assign a label.
  If you enter the name of an existing variable, the link is established and the mark changes from a red X to a green check mark.

FORMULAS

Read how a formula is created in the chapter on the Formula Editor (on page 77).

Symbol from library

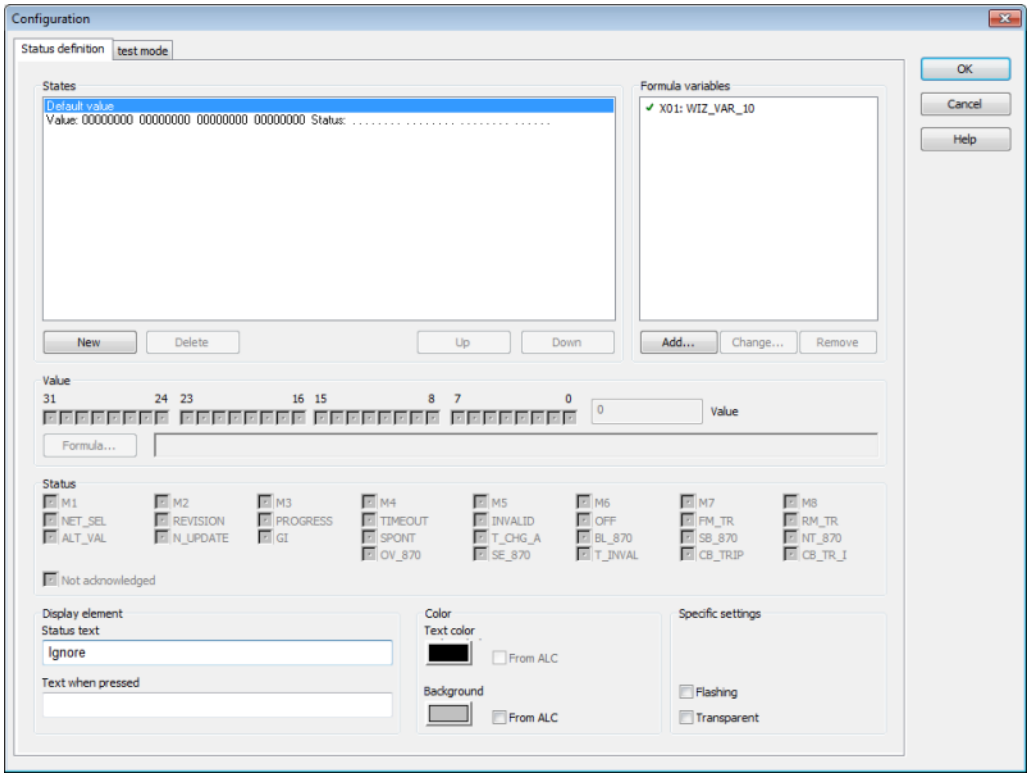
The general settings for all display types can be found in the chapter on statuses (on page 63).



Parameters	Description
Display element	
Symbol from Library	Clicking on the symbol opens the dialog to select a symbol. Clicking on the X next to the symbol deletes the selected symbol.
Edit linking rule	Variables and functions contained in the symbol can be exchanged with the help of linking rules. **Attention:** If a variable or function does not fit into the scheme, that link cannot be replaced.
Colors	Only active if the Original Symbol Colors property in Specific Settings is turned off.
Color	Choice of color.
Filling color	Choice of filling color.
From ALC	The color of the linked source is used.
Specific settings	
Original Symbol Colors	Active: Transfers original color from symbol. Deactivates Take color of main variable property.
Flashing	Active: Symbol flashes in runtime if status has been achieved.

Status Text and Screen Symbol

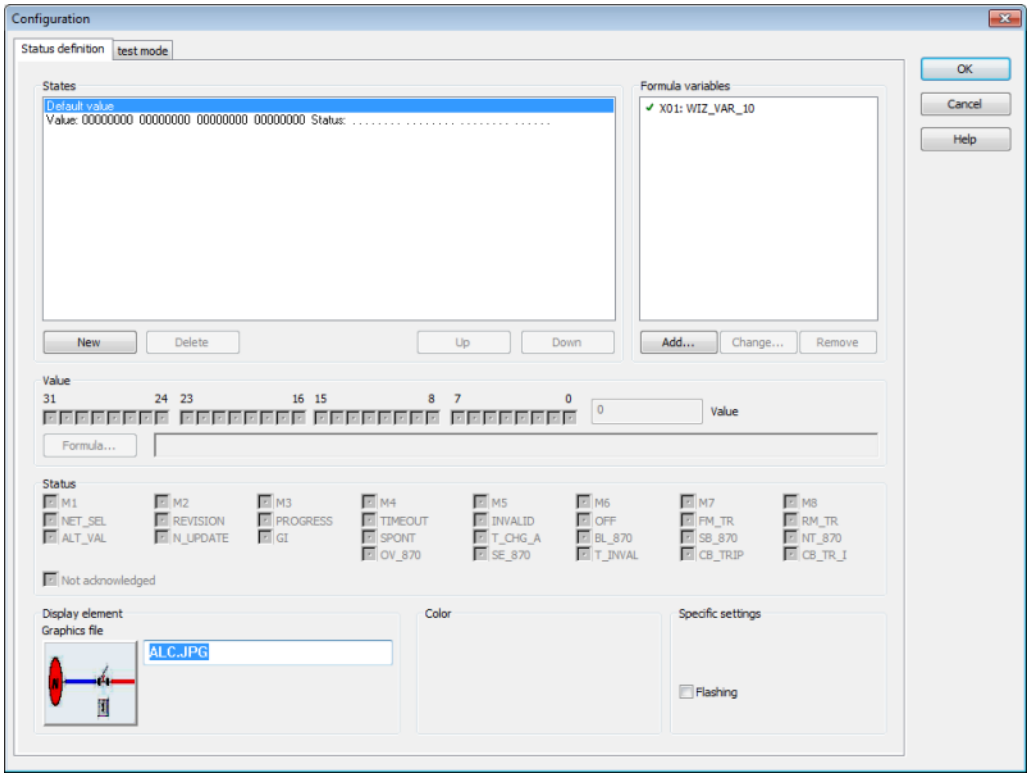
The general settings for all display types can be found in the chapter on statuses (on page 63).



Parameters	Description
Display element	
Status Text	Text for the status.
Text when pressed	Text with pressed Symbol.
Colors	
Color	Choice of color.
Filling color	Choice of filling color.
From ALC	The color of the linked source is used.
Specific settings	
Flashing	Active: Symbol flashes in runtime if status has been achieved.
Transparent	Active: Filling color is set to transparent.

Graphic and screen symbol

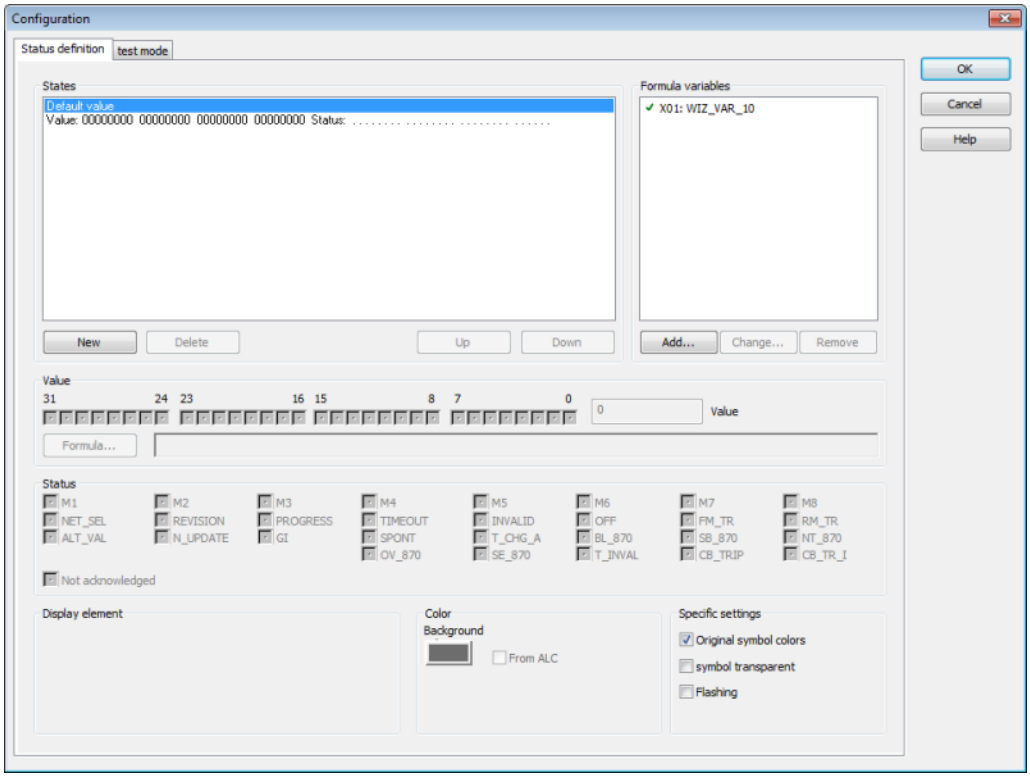
The general settings for all display types can be found in the chapter on Statuses (on page 63).



Parameters	Description
Display element	
Graphics	Clicking on Graphics opens the graphics selection dialog. Delete: Remove name from text line.
Specific settings	
Flashing	Active: Symbol flashes in runtime if status has been achieved.

Only Screen Symbol

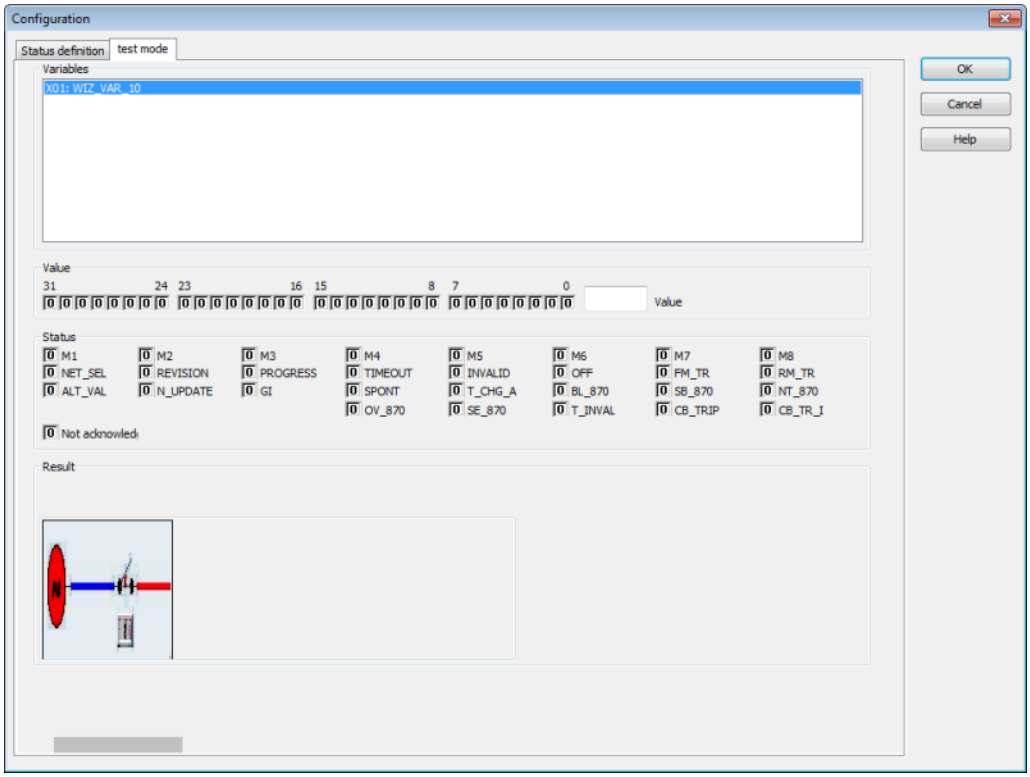
The general settings for all display types can be found in the chapter on statuses (on page 63).



Parameters	Description
Colors	
Color	Choice of color.
From ALC	The color of the linked source is used.
Specific settings	
Original Symbol Colors	Active: Transfers original color from symbol.
Transparent Symbol	Active: Symbol color is set to transparent.
Flashing	Active: Symbol flashes in runtime if status has been achieved.

10.7.4 Test mode

The test mode simulates the Runtime: You can test the change of a value and see how your Combined Element would behave during the Runtime. You can only activate this tab if all links refer to existing variables.



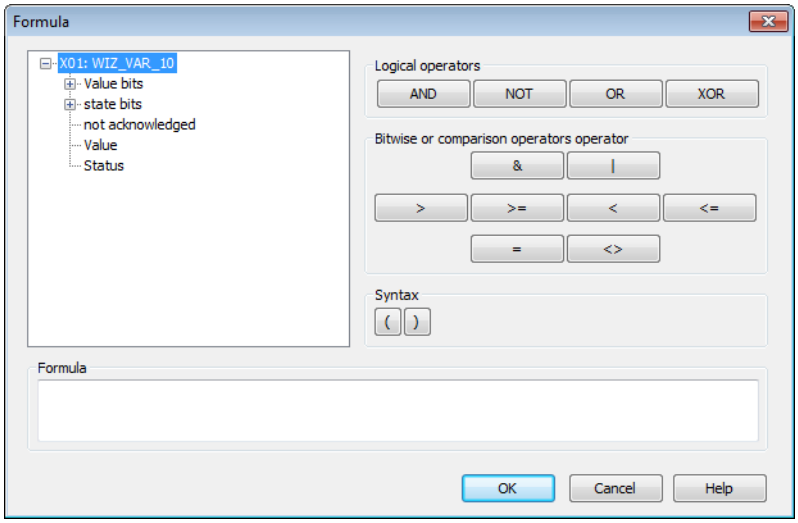
Note: In test mode, the symbol is displayed in a lower resolution than on the screen in Runtime .

10.7.5 Formula editor

The formula editor provides support when creating formulas with logical or comparative operators with a combined element, for interlockings and commands. If additional variables are required for a formula, create these in the **formula variables** (on page 63) area of the status window by clicking on the **Add** button. existing formulas are displayed in the status list with the letters**F** .

CREATING A FORMULA

Click on the **Formula** button in the status window The formula editor opens



You select the bits for your formula in the left screen.

On the right, you find the operators for logical and comparative operations.

The formula created is displayed in the **Formula** area.

Info

Up to 99 variables can be linked in one formula. X01 to X99. The length of the formula may not exceed 4096 characters.

THE MEANING OF THE BITS:

Parameters	Description
value bits	32 value bits (von 0 -31) are available. They describe the variable value bit by bit. For BOOLEAN variables, only bit 0 is of importance, for SINT and USINT only the bits from 0–7, etc. **Note:** The value refers to the raw value (signal range) of the variables and not to the converted measuring range.
status bits	Here you find the most commonly used status bits. You find the exact definition and use of the status bits in the Status Bits List (on page 81).
not acknowledged	Not acknowledged is treated like a usual status bit. But here it is listed separately, because it does not belong to the classical variable statuses.
value and status	In the formulas, all values (value bits and status bits) are treated as binary values and can be logically linked with AND, OR, etc. The total value and overall status are an exception to this. In order to arrive at a Boolean expression, this total value has to be ORed bitwise with a constant (on page 85). For this, we use the operator &. For the result 0 (false) of this logical ORing we get the binary value 0 (false), otherwise 1 (true). Example: see chapter onbitwise ORing example (on page 85)

Info

The status bits NORM and N_NORM are only available in the formula editor here. They cannot be engineered using the status.

If other settings outside the formula are set for the current status, they are combined with the formula with a logical AND.

Refer to the examples (on page 87) section for examples.

Info

Formulas with binary X values and bitwise linking can be used with a maximum of 2 binary values. If more values are required, the linking must be carried out without binary X values.

Example:

`X01.Value & X02.Value` -> works

`X01.Value & X02.Value & X03.Value` -> does not work

But:

`X01.00 AND X02.00 AND X03.00 AND X04.00 AND X05.00` -> works

List of status bits

Bit number	Short term	Long name	zenon Logic label
0	M1	User defined status 1	_VSB_ST_M1
1	M2	User defined status 2	_VSB_ST_M2
2	M3	User defined status 3	_VSB_ST_M3
3	M4	User defined status 4	_VSB_ST_M4
4	M5	User defined status 5	_VSB_ST_M5
5	M6	User defined status 6	_VSB_ST_M6
6	M7	User defined status 7	_VSB_ST_M7
7	M8	User defined status 8	_VSB_ST_M8
8	NET_SEL	Select in the network	_VSB_SELEC
9	REVISION	Revision	_VSB_REV
10	PROGRESS	In operation	_VSB_DIREC
11	TIMEOUT	Timeout exceeded	_VSB_RTE
12	MAN_VAL	Hand value	_VSB_MVALUE
13	M14	User defined status 14	_VSB_ST_14
14	M15	User defined status 15	_VSB_ST_15
15	M16	User defined status 16	_VSB_ST_16
16	GI	General interrogation	_VSB_GR
17	SPONT	Spontaneous	_VSB_SPONT
18	INVALID	Invalid	_VSB_I_BIT
19	T_CHG_A	Time change announcement	_VSB_SUWI
20	OFF	Switched off	_VSB_N_UPD
21	T_EXTERN	Real time external	_VSB_RT_E
22	T_INTERN	Real time internal	_VSB_RT_I
23	N_SORTAB	Not sortable	_VSB_NSORT
24	FM_TR	Fault message transformer value	_VSB_DM_TR

25	RM_TR	Run message transformer value	_VSB_RM_TR
26	INFO	Information for the variable	_VSB_INFO
27	ALT_VAL	Alternative value If no value was transferred, the defined alternate value is used otherwise the last valid value is used.	_VSB_AVALUE
28	RES28	Reserved for internal use (alarm flashing)	_VSB_RES28
29	N_UPDATE	Not updated	_VSB_ACTUAL
30	T_STD	Standard time	_VSB_WINTER
31	RES31	Reserved for internal use (alarm flashing)	_VSB_RES31
32	COT0	Cause of transmission bit 1	_VSB_TCB0
33	COT1	Cause of transmission bit 2	_VSB_TCB1
34	COT2	Cause of transmission bit 3	_VSB_TCB2
35	COT3	Cause of transmission bit 4	_VSB_TCB3
36	COT4	Cause of transmission bit 5	_VSB_TCB4
37	COT5	Cause of transmission bit 6	_VSB_TCB5
38	N_CONF	Negative acceptance of Select by device (IEC60870 [P/N])	_VSB_PN_BIT
39	TEST	Test bit (IEC 60870 [T])	_VSB_T_BIT
40	WR_ACK	Writing acknowledged	_VSB_WR_ACK
41	WR_SUC	Writing successful	_VSB_WR_SUC
42	NORM	Normal status	_VSB_NORM
43	N_NORM	Deviation from normal status	_VSB_ABNORM
44	BL_870	IEC 60870 Status: blocked	_VSB_BL_BIT
45	SB_870	IEC 60870 Status: substituted	_VSB_SP_BIT
46	NT_870	IEC 60870 Status: not topical	_VSB_NT_BIT

47	OV_870	IEC 60870 Status: overflow	_VSB_OV_BIT
48	SE_870	IEC 60870 Status: select	_VSB_SE_BIT
49	T_INVAL	Time invalid	not defined
50	CB_TRIP	Breaker tripping detected	not defined
51	CB_TR_I	Breaker tripping detection inactive	not defined
52	RES52	reserved	not defined
53	RES53	reserved	not defined
54	RES54	reserved	not defined
55	RES55	reserved	not defined
56	RES56	reserved	not defined
57	RES57	reserved	not defined
58	RES58	reserved	not defined
59	RES59	reserved	not defined
60	RES60	reserved	not defined
61	RES61	reserved	not defined
62	RES62	reserved	not defined
63	RES63	reserved	not defined

Info

In formulas all status bits are available. For other use the availability can be reduced.

You can read details on status processing in the Status processing chapter.

Logical Operators

Logical links: Variables will only be checked for the logical value '0'; if the value does not equal '0', it will be considered as '1'.

In contrast to bit formulas, the technical range can be modified by a stretch factor -> (not equal '0' or '1').

Operator	Meaning
AND	logical 'AND'
NOT	Negation
OR	logical 'OR'
XOR	logical 'EXCLUSIVE OR'

The operators have the following priority in the formula calculation:

Priority	Operator
1	& (operator for bit formulas (on page 84))
2	NOT
3	AND
4	XOR/OR

Info

Up to 99 variables can be linked in one formula. X01 to X99.

Info

The status bits NORM and N_NORM are only available in the formula editor here. They cannot be engineered using the status.

Bit formulas

Bit formulas only have a logical high or low state. In contrast to logical formulas, the raw value is already predefined (0,1).

Operator	Description
&	AND
|	OR

Example oring bit for bit

You want to find out if one of the user status bits 1-8 (M1 ... M8) of the variable X01 is set.

USUAL FORMULA:

```
X01.M1 OR X01.M2 OR X01.M3 OR X01.M4 OR X01.M5 OR X01.M6 OR X01.M7 OR X01.M8
```

This request can be made much easier by using the logical ORing of the (complete) status.

LOGICAL ORING:

```
X01.Status & 0xFF
```

The constant can be entered in hexadecimals, as described above:

`0xFF` corresponds to decimal `256`; these are the first eight status bits (binary 11111111). If one of these bit is set to `1`, the result of this bitwise ORing is `1` (true), otherwise it is `0` (false).

If, for example, all user status bits except the user status bit M7 should be queried, the binary statement for this would be: 10111111. Bit 7 is not of interest and is thus set to `0`. This corresponds to 0xBF in hexadecimal. The expression for the formula is then: **`X01.Status & 0xBF`**.

Instead of ORing bitwise with a constant, the value can also be directly compared to a decimal number. If the comparison is wrong, the binary value is `0` (false) otherwise it is `1` (true).

Example:

You want to find out if the value is equal to the constant `202`: The formula is:

```
X01.value = 202
```

If the value is equal to the constant `202`, the result of the comparison is `1` (true) otherwise it is `0` (false).

Note: The bitwise ORing works with the OR character (|) in a similar manner to this example.

Comparative operators

Comparison operators serve for the direct comparison of two numeric values. The result of this comparison is a binary value. „0“ if the condition is not fulfilled and „1“ if the condition is fulfilled.

Operator	Description
<	smaller
>	greater
<=	Lower or equal
>=	greater or equal
=	Equal
<>	unequal

To the left and to the right of the comparison operator, there has to be a (total) value or a (total) status, single bits cannot be used with these comparison operators.

There can also be a constant to the right of the comparison operator. (the constants can only be integers; a comparison to a floating point number is not possible.)
These constants are entered as hexadecimal values or decimal values in the combined element. Hexadecimal figures are automatically converted to decimal values by clicking on ox (for example, 0x64 is in decimal figures 100).

Example

X01.value >= X02.value
The result is 1, if the value of X01 is higher than or equal to the value of X02

X01.value = 0x64
The result is 1, if the value of X01 is exactly equal to the numeric value 100 (= hex 0x64)

(X01.value = 0x64) OR (X01.value = 0x65)
The result is 1, if the value of X01 is exactly equal to the numeric value 100 or 101 (= hex 0x64 and hex 0x65)

Example for formulas

SIMPLE LOGICAL AND LINKING BETWEEN TWO BIT VALUES

Example

Formula: X01.03 AND X02.03

This formula has the status TRUE, if both `bit 3` of variable 1 and `bit 3` of variable 2 both have the value 1.

COMPARISON OF AN ANALOGUE VALUE OR STATUS OF A VARIABLE

Example

(X01.Value> X02.Value)

COMPARE ANALOG VALUES WITH EACH OTHER ON A LOGICAL BASIS

Example

(X01.Value> X02.Value) AND (X01.Value = X02.Value)

COMPARE WITH VALUE BITS AND STATUS BITS

Example

(X01.Value> X02.Value) AND (X01.Value = X02.Value) OR (X01.03 = X02.03)

COMPARE A VALUE WITH A DECIMAL OR HEXADECIMAL VALUE

Example

Formula: (X01.Value = 111)

Formula: (X01.Value = 0x6F)

If a hexadecimal values is used, this is later transferred to decimal by clicking on OK. If a decimal value is entered and confirmed, the value continues to be displayed as a decimal value after reopening.

Info

It is not possible to use a comma or a period when entering values.

10.7.6 Display of variable name and variable identification

You can display variable names and identifications with the the help of the combined element. Proceed in the following way:

1. Create a combined element.
2. As type of display, select `Status text & screen symbol`.
3. Create a default state.
4. Enter the following placeholder as status text

fill-in	Description
%n	shows the variable name
%n,1,2	shows the variable name starting at position 1 for 2 characters
%N	shows the variable name
%N,3,2	Shows the variable name minus the first 3 and the last 2 characters
%l	shows the variable identification
%l,1,2	shows the variable identification starting at position 1 for 2 characters
%L	shows the variable identification
%L.3.2	Shows the variable identification minus the first 3 and the last 2 characters

10.8 Combo/List Box

You link values of variables with texts using the **`Combo/List boxes`** dynamic element. You use this in two ways:

1. If a text is selected in runtime, the value of the variable changes to the value that is linked to this text.
2. If the value reaches the value of the defined value, the linked text is displayed in the combo/list box with the current value.

CREATE A COMBO/LIST BOX

To create the **Combo/List box** dynamic element in a screen:

1. select the **Combo/List box** symbol in the **Elements** tool bar or in the **Elements** drop-down list
2. select the start point in the main window
3. pull open the element while pressing and holding the left mouse button
4. a Dialog (on page 238) opens in which you can link the desired variable to the element.
   The following variable types are supported:
   - Binary
   - Numerical
   - string

> **Info**
>
> *Variables can also be linked to the element by Drag&Drop. Drag a variable onto a combo/list box and this then replaces the variables that were there before.*

The combo/list box element supports the same additional functionalities as the numerical value (on page 191) element:

- Lock, Status, Visibility, Dynamic Colors.
- Dynamic colors: The background color is only accepted for existing entries. The part of the element that is not filled with entries is displayed in the standard background color.

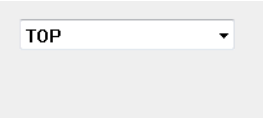
Linking of macros to element events (such as LeftClickUp ...) is not supported.

> **Attention**
>
> *It is not possible to modify colors and fonts in the combo/list box under Windows CE.*

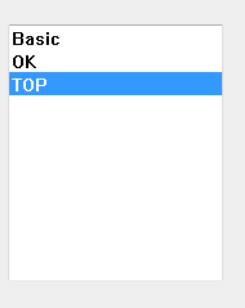
Comboboxes/listboxes can be created dynamically or statically. The ID_PROP_COMBOLIST_DYNCONTENT_USE property switches between static and dynamic operation. Fixed entries are defined in the Editor during static operation. In dynamic operation, a content variable is linked and its value is interpreted in Runtime.

DIFFERENCE BETWEEN COMBOBOX AND LIST BOX

The texts are opened individually in a window in the **Combo box**.



The values are displayed in a list and the current value is marked in the **List box**. If there are more statuses than can be displayed in the list, a scroll bar is displayed.



Info

If the variable has a value which has not been defined in the combo box/list box entries, no text is displayed or marked.

STATIC COMBOBOX OR LIST BOX

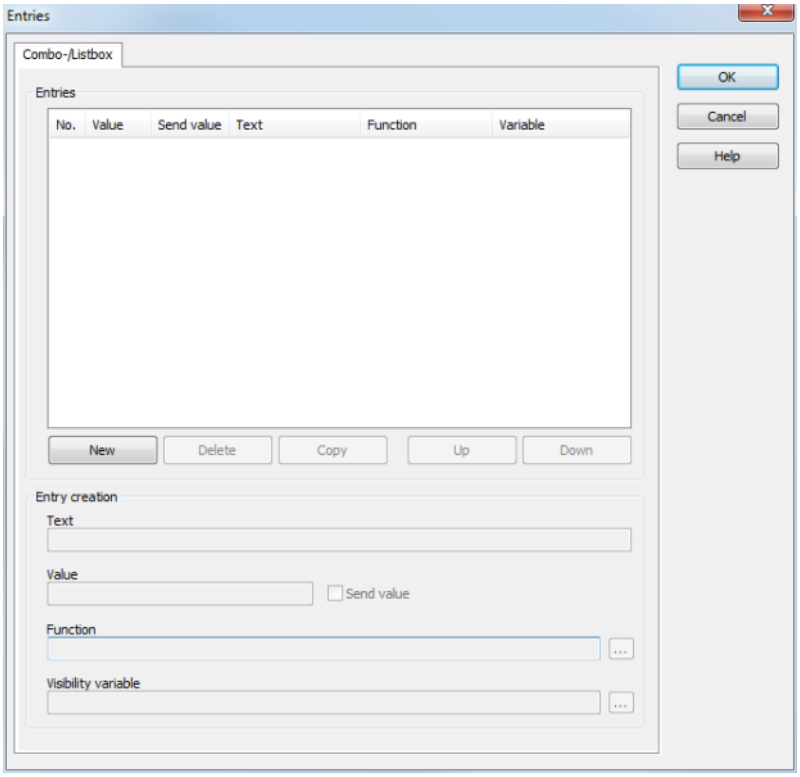
Fixed entries are configured in the Editor for a static box. These are displayed and used in Runtime.

To enter values:

1. Open the `Representation` group.
2. Select the display type using the `Type` property: `Combobox` or `List box`.

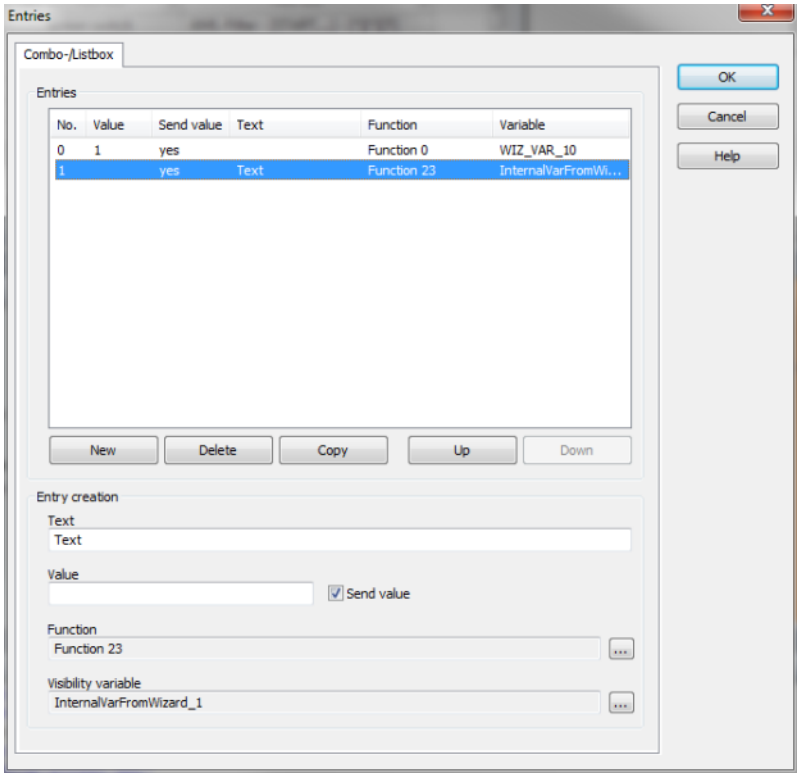
Default value `Listbox`

3. Deactivate the `Entries from string variable` property
4. Click on the `...` button in the `Entries` property
5. The dialog for configuration of the entries is opened



6. enter the values for statuses
7. Link the values with the texts for statuses

ENTRIES DIALOG



Parameters	Description
Entries	List of entries already defined.
no.	Consecutive number, is assigned automatically.
Value	Value that was **assigned** under **Entries/**Value.
Send value	Yes: Send value is active. No: Send value is inactive.
Text	Character string that was entered into**Entries/**Text .
Function	Function that was linked to the **Entries/**Function field.
Variable	Linked visibility variable.
New	Creates a new entry with: ▸ consecutive number ▸ Set active value
Delete	Removes the selected entry. Only one entry can be removed; it is not possible to select several entries.
Copy	Copies selected element and adds it at the end of the list as new element. The settings of the copied elements are taken over.
Upwards/downwards	Moves the selected entry upwards or downwards and adapts the consecutive numbering to suit.
Entry setting	Configuration of entries in the list.
Text	Entry of the character string which is linked to the value.
Value	Numerical value. The values that can be entered depends on the data type of the variables.
Send value	activates or deactivates set value: Active**:** The value is set as a set value. Inactive: The value entered only serves to index the selection. **Attention:** Set value only affects the value, not the function. The function is always sent. The Set active value must be inactive for a value without linked variable!

Function	Function that is executed when the entry is selected in Runtime. Click on the ... button to open the dialog for selecting a function (on page 237). **Attention:** The function is only executed if you select the entry via the combo box/list box during runtime. If the triggering value changes and reaches the level of the variable, the function is not executed. Set value must be inactive for a function call without linked variable.
Visibility variable	Variables for which the value changes and thus limit violations can be evaluated in Runtime. If there is a limit breach of the visibility variables, the Visibility property is evaluated by this limit breach. It determines if the entry is visible in the list or not. The entry is displayed in the list as standard (there is no limit breach). Click on the ... button to open the dialog for selecting a variable (on page 238). **Attention:** A value change during operation can lead to, with comboboxes, an open combobox being closed.
OK	Applies settings and closes dialog.
Cancel	Discards changes and closes dialog.
Help	Opens online help.

SORTING ENTRIES

The entries can be sorted using:

- **Upwards** and **Downwards** buttons: Highlight the desired element and move it with the buttons.
- Drag & drop: Drag the element to the desired place with the mouse. A red line displays the point where the element is inserted.

DYNAMIC COMBOBOX OR LIST BOX

For dynamic operation, a string variable that defines the content must be linked using the Variable property. Any static entries that may be configured at the same time in Runtime are ignored. The value of the content variable is evaluated and interpreted. To do this, the value of the variables must correspond to a valid layout in Runtime.

LAYOUT

The string value displays a list of entries. Each entry consists of a value and a text that is separated with a comma ",". Entries are concluded with a semi-colon ";". If the text part of an entry contains an "@" character, the text part is translated with the help of the existing language table.

Example of a valid string value:

```
1,one;2,two;3,three;
```

Example of a valid string value with translation:

```
1,@one;2,@two;3,@three;
```

CONFIGURATION

To link a variable:

1. Open the Representation group.
2. Select the display type using the Type property: Combobox or List box.

   Default value Listbox
3. Activate the Entries from string variable property
4. Select, using the Variable property, the desired variable
   (clicking on the ... button opens the dialog for variable selection (on page 238))
5. Select the string variable with the configured entries

MARK ELEMENT IN LIST WHEN SELECTED

The elements displayed in the combo box or list box can be marked in color in runtime. To do this:

- navigate to Group Color in properties
- select Properties in the group Colors static
  - Text color when selected for the text color
  - Background color for selected entry for the text background color

These colors are then used during runtime. Operating system settings are ignored, as are the colors of linked variables.

Info

The colors of the operating system were used up to version 6.50. When converting the project:

- the background color is set to that configured in the operating system
- if the text color is set as a statically defined color

On computers where the colors have been changed in the operating system, this may lead to different behavior during runtime.

TROUBLESHOOTING

Errors are displayed as log entries in the Diagnosis Server:

Message	Level	Meaning
The value '%s' of the content variable '%s' is not valid, entries might be incorrect or missing!	Error	The string variable linked in the `Variable` property has a value with invalid layout. The text is evaluated as far as possible. The current value is also logged to make error analysis in Runtime easier.

10.9 Dynamic Text

With the **`Dynamic Text`** dynamic element, you display the current limit value text in the event of the limit being breached or display the value of a string variable in alphanumeric form. If there is no limit violation, the text defined in the Editor is displayed.

To create a **`dynamic text`** in a screen:

1. select the **`Dynamic Text`** symbol in the **`Elements`** tool bar or in the **`Elements`** drop-down list
2. select the start point in the main window
3. pull open the element while pressing and holding the left mouse button
4. the dialog to assign a variable (on page 238) opens
5. select the variable you want to assign

6. Define the desired properties in the properties window

10.10 Ellipse

To draw an ellipse:

1. select the Ellipse/Circle symbol in the **Elements** tool bar or in the **Elements** drop-down list
2. select the start point in the main window
3. pull open the ellipse while pressing and holding the left mouse button
4. The shape, size and position can be changed at any time by pulling the vertexes or moving the element
5. if you press and hold alt while pulling the outer corner points, the change is carried out symmetrically
6. define the desired properties in the **properties window**

10.11 Circle

In order to draw a circle:

1. select the Ellipse/Circle symbol in the **Elements** tool bar or in the **Elements** drop-down list
2. select the start point in the main window
3. hold down the shift key
4. pull open the circle while pressing and holding the left mouse button
5. The shape, size and position can be changed at any time by pulling the vertexes or moving the element
6. if you press and hold alt while pulling the outer corner points, the change is carried out symmetrically

7. if you want the circle to retain its shape press and hold `Shift` while pulling
8. define the desired properties in the **`Properties window`**

10.12 Arc of a circle

In order to draw an arc of a circle:

- select the `Arc of Circle` symbol in the **`Elements`** tool bar or in the **`Elements`** drop-down list
- select the start point in the main window
- pull open the arc of the circle while pressing and holding the left mouse button
- pulling upwards to the right opens the arc of the circle to the left; all directions work along the same lines
- to change the opening direction, move a central touch point over the opposite line
- The shape, size and position can be changed at any time by pulling the vertexes or moving the element
- if you press and hold `alt` while pulling the outer corner points, the change is carried out symmetrically
- hold down the `shift` key whilst moving it and the aspect ratio is maintained
- define the desired properties in the **`properties window`**

> **Info**
>
> *The arc of the circle is always displayed as a quadrant and can be extended by copying and pasting.*
> *Filling patterns and filling colors are not available for arcs of circles.*

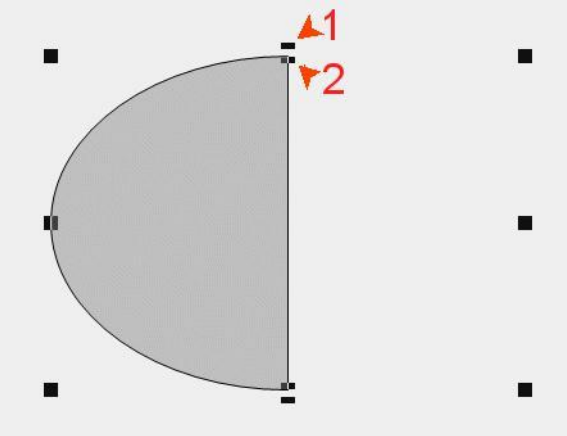
10.13 Segment of a circle

> **Attention**
>
> *With the segment of a circle, you draw the desired part of a circle or ellipse.*

To draw a segment of a circle:

1. Select the Segment of Circle symbol in the Elements tool bar or in the Elements drop-down list
2. select the start point in the main window
3. pull on the segment of the circle pressing and holding the left mouse button
4. pulling upwards to the right opens the segment to the left and vice versa
5. to change the opening direction, move a central touch point over the opposite line
6. The shape, size and position can be changed at any time by pulling the vertexes or moving the element
7. if you press and hold Alt while pulling the outer corner points, the change is carried out symmetrically
8. hold down the shift key whilst moving it and the aspect ratio is maintained
9. you have two possibilities to change the opening angle or to open the circle segment upwards or downwards:
   - move the mouse over one of the two divided touch points until the mouse pointer on the outermost of the two points turns into a black cross with arrows and move the point into the desired direction
     or
   - Change the angle settings properties under `Opening angle`. e.g.:
     - for opening downwards: `End angle [°]` 180° and `Start angle [°]` to 0°
     - for opening downwards: the reverse of the above
     Angle data can have a maximum of two decimal points.
     Note: Because the corner points calculated from angle data are pixels, and these must be whole numbers, it can be the case that the angle display deviates from the data somewhat. For example 45° does not give whole number of pixels. It is rounded off internally in zenon and the the angle is displayed as 44.75°.

10. define the desired properties in the Properties window



Key:

1) touch points for opening angle

2) touch point for height

> **Info**
>
> *If you would like to use the segment of a circle as an arc of a circle, change the* `Filling pattern` *property to the value of transparent. Only the outer line is displayed with the transparent display type.*
> *The opening angle must always be > 0° and < 360°.*

> **Attention**
>
> *The Segment of a circle element is not available in Windows CE.*

10.14 Line

To draw a line:

1. select the `Line` symbol in the **`Elements`** tool bar or in the **`Elements`** drop-down list
2. select the start point in the main window
3. pull out the line while pressing and holding the left mouse button
4. The shape, size and position can be changed at any time by pulling the vertexes or moving the element
5. if you press and hold `alt` while pulling the outer corner points, the change is carried out symmetrically
6. define the desired properties in the **`properties window`**

> **Info**
>
> *Filling patterns and filling colors are not available for lines.*

10.15 Message element

With the **`Message Element`** dynamic element, you can show text in runtime depending on the statuses of process variables. The source for texts are ASCII text files, made up as tables. You can configure these freely.

To create the **`Message Element`** in a screen:

1. select the **`Message Element`** symbol in the **`Elements`** tool bar or in the **`Elements`** drop-down list
2. select the start point in the main window
3. pull open the element while pressing and holding the left mouse button
4. the dialog to Assign a variable (on page 238) opens
5. select the desired variables
6. Define the desired properties in the properties window

CONSTRUCTION OF THE TABLE

- Create an ASCII text document
- you must save it in Unicode format
- use the character |(ASCII-Code 124) as a column separator
- Example:

1 | Off | On | Diff | Fault

2 |Off | On | True | False

4 | Text1 | Text2 | Text3 | Text4

- the linked variables define lines and columns of the text to be shown

FIRST VARIABLE

The first variable in the list must be a word variable and defines the column position from which the text is taken. At the same time, it controls the type of processing.

bit set	effect
Bit 15 (bit with the highest value	Message element blocked; nothing is displayed any more.
Bit 14	Update of the screen output switched off; last active text is displayed.

Note: For a 16-bit variable a maximum of 14 columns is possible. The addressing of the column takes place via the bit number and not via the analog value.

SECOND AND FURTHER VARIABLES

The other variables define the line position from which the text is taken. The significance arises from

Value	Row
1	Line 1
2	Line 2
4	Line 3

Each line variable is responsible for a maximum of 16 lines. If more than 16 lines are needed, further line variables can be defined.

If line 1 and line 2 are to be displayed alternately, the significance of the line variable must be 3. If all 16 bits are set, lines 1 to line 16 are displayed alternately.

10.16 Multi-binary Element

You can define several elements with the **Multi-binary** dynamic element:

- display graphics depending on variable values
- Color symbols
- Issue status text
- link numerical values and binary values
- display a switch or key for the first variable, provided the first variable is a binary variable

CREATE THE MULTI BINARY ELEMENT

To create a **multi-binary** in a screen:

1. select the **Multi-binary** symbol in the **Elements** tool bar or in the **Elements** drop-down list
2. select the start point in the main window
3. pull open the element while pressing and holding the left mouse button
4. the dialog to Assign a variable (on page 238) opens
5. select the desired variables
6. Define the desired properties in the properties window

DEFINE STATUSES

You define the settings for variable statuses to be displayed in the dialog to enter elements:

1. click on the `Representation` node
2. on the `Configuration` property
3. the dialog for status definition (on page 104) opens

4. defined statuses are processed from top to bottom
5. The settings for the first applicable status are displayed

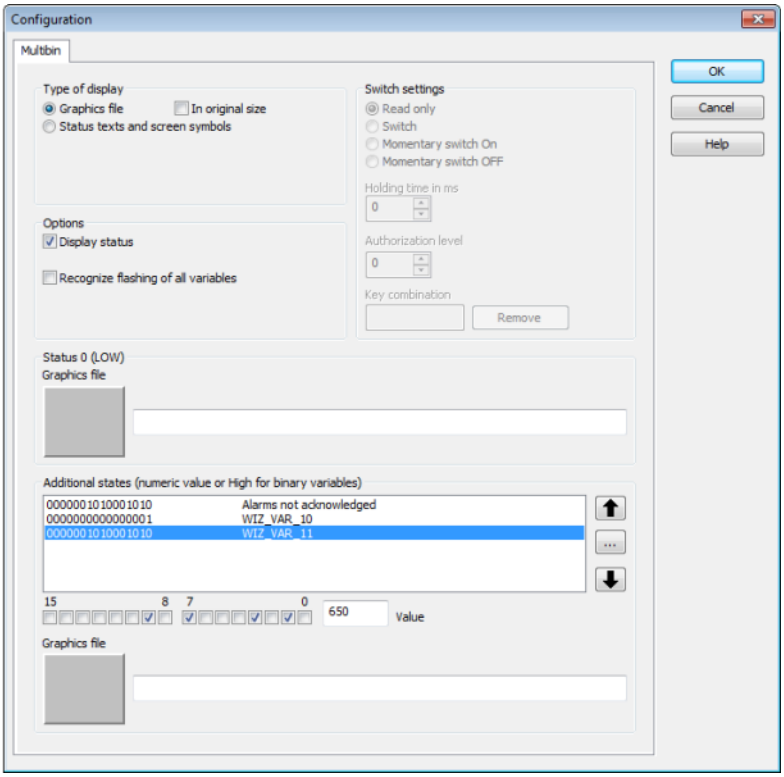
10.16.1 Multi-binary element

To assign the **`multi binary element`** to certain statuses:

1. click on the `Representation` node
2. on the `Configuration` property
3. the dialog for inputting elements (on page 104) opens
4. select the desired display type:
   - Graphics: Representation of the information in pixel graphics files; this also enables moved screens
   - Screen symbol/status text: Display of symbols and text information
5. Defined statuses processed from top to bottom and the setting for the first applicable status are displayed.

Graphic

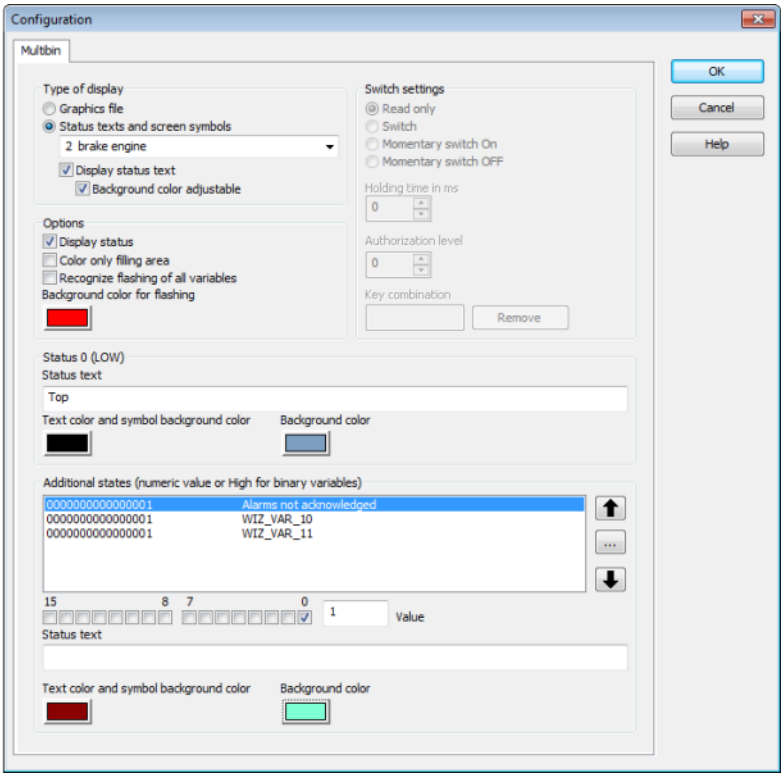
Display of information in pixel graphics files; this also enables moved screens.



Property	Description
In original size	Active: Graphics are displayed in original size. Inactive: Graphics are stretched or compressed to the display size.
Options	
Display status	Highlights the element in runtime with a red dot at exception status.
Recognize flashing of all variables	Active: Sum alarming of all linked process variables. The graphics or color are determined by the current status. Active flashing states remain kept until quitting.
Status 0 (LOW)	Settings for the current status in line with display type.
Graphics	Selection of graphics. A click on the ... button opens the file selection dialog. ▸ To select a file from a project, this must already be present in the Files/graphics node in the project manager. ▸ to select a file directly from any desired folder on the computer, the Direct file selection property must be activated.
Further statuses	Select further variables (on page 238) via the ... button and sorting of statuses via the **Upwards** and **Downwards** buttons. For each of the linked variables, there must be a discrete status defined; this is defined by entering the values for multibit information and numerical values. Multibit values must be assigned several times: Number of statuses minus 1; example double message=3. If a discrete state has occurred, it is displayed. If none of the indicated values is present, the status 0 [LOW], globally valid for the element, is displayed. **Note:** for non-binary variables, the value is entered in the signal resolution.

Screen symbol/status text

Display of symbols and text information. The color of the current status is given at the same time.

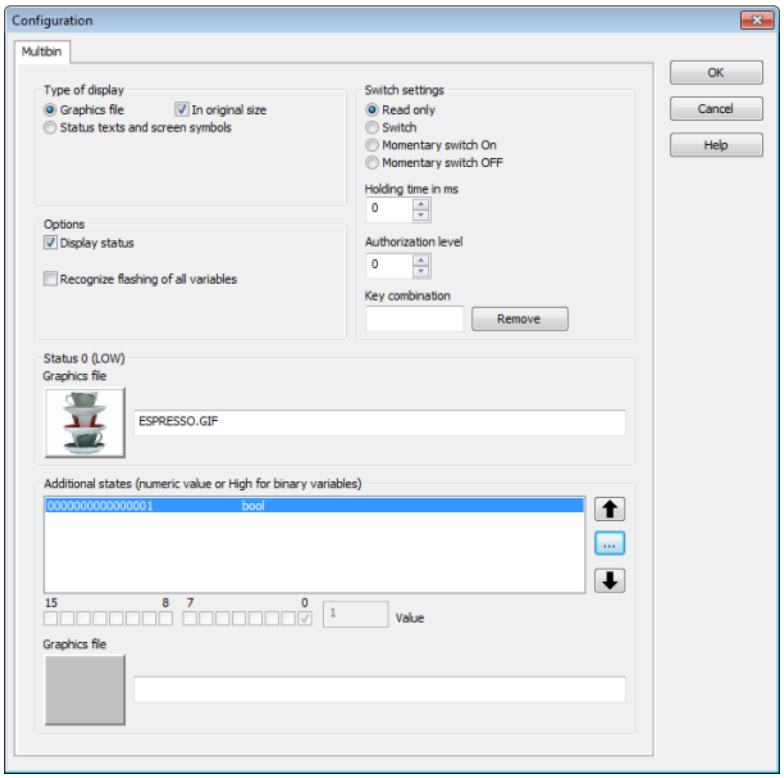


Property	Description
Screen symbol/ Status text	Select a symbol. The symbol must already be present on the screen The symbol shown in the drop-down list flashes on the screen until the **Enter element** dialog is ended.
Display status text	Active: Status texts are displayed in runtime.
Background color can be set	Active: The background color for the text can be freely chosen.
Options	
Display status	Highlights the element in runtime with a red dot at exception status.
Only color the filling color	only the filling area of the static elements is colored, the border stays the same
Recognize flashing of all variables	Active: Sum alarming of all linked process variables. The graphics or color are determined by the current status. Active flashing states are maintained until quitting.
Background color when flashing	Background color when flashing
Status 0 (Low)	
Status text	Text that it is displayed in runtime. Input is only possible if the Display status text property is active.
Symbol color/text color	Select the color for the symbol and status text.
Text background color	Selection of the background color for the status text. Only available if the configurable background property is active.
Further statuses	Select further variables (on page 238) via the ... button and sorting of statuses via the **Upwards** and **Downwards** buttons. For each of the linked variables, there must be a discrete status defined; this is defined by entering the values for multibit information and numerical values. Multibit values must be assigned several times: Number of statuses minus 1; example double message=3. If a discrete state has occurred, it is displayed. If none of the indicated values is present, the status 0 [LOW], globally valid for the element, is

	displayed. **Note:** for non-binary variables, the value is entered in the signal resolution.

Switch settings

If a binary variable is linked as the first variable, this can be used as a switch or momentary switch.



Property	Description
Read only	It is not possible to input setpoints in runtime.
Switch	Element acts as a switch for setpoint input.
Key ON	Writes setpoint HIGH. The preferred position for the switch goes to **Off**.
Key OFF	Writes setpoint LOW. The preferred position for the switch goes to **On**.
Holding time in ms	Holding time of target state
Authorization groups	Defines the necessary authorization level for set value element.
Key combinations	Key combination for set value input

10.17 Polygon

To draw a polygon:

1. select the Polygon symbol in the **Elements** tool bar or in the **Elements** drop-down list
2. In the main window, select the start point by clicking the left mouse button
3. pull the first section
4. place an initial corner point with a simple mouse click (left)
5. draw another line and place the corner points
6. set the end point by double clicking or pressing the escape key
7. The shape, size and position can be changed at any time by pulling the vertexes or moving the element
8. hold down the shift key when pulling; this way, only horizontal or vertical movements away from the original position are possible
9. define the desired properties in the Properties window

Hint: You define the rounding (on page 195) of the lines with the Rounding property. To do this, select a value between 0 (no rounding) and 1 (maximum rounding).

ADDING CORNERS:

To add corners (corner points):

- move the mouse cursor to a line of the polygon:
- right-click
- select menu item **`Adding a node in the selected element`** from the context menu

or:

- press `Ctrl` and `Shift` simultaneously
- move the mouse cursor to a line of the polygon:
- the mouse cursor changes to an arrow with a plus-symbol
- left-click in order to add a corner

REMOVING CORNERS:

- move the mouse cursor to a corner of the polygon:
- right-click
- select menu item **`Deleting the node in the selected element`** from the context menu

or:

- press `Ctrl` and `Shift` simultaneously
- move the mouse cursor to a corner of the polygon:
- the mouse cursor changes to an arrow with a plus-symbol
- left-click to delete the corner

Info

The `With brightness values` property is not available for polygons.

10.18 Polyline

To draw a polyline:

1. select the Polyline symbol in the **Elements** tool bar or in the **Elements** drop-down list
2. In the main window, select the start point by clicking the left mouse button
3. pull the first section
4. place an initial corner point with a simple mouse click (left)
5. draw another line and place the corner points
6. set the end point by double clicking or pressing the escape key
7. The shape, size and position can be changed at any time by pulling the vertexes or moving the element
8. hold down the shift key when pulling; this way, only horizontal or vertical movements away from the original position are possible
9. define the desired properties in the Properties window

Hint: You define the rounding (on page 195) of the lines with the Rounding property. To do this, select a value between 0 (no rounding) and 1.

ADDING CORNERS:

To add corners (corner points):

- move the mouse cursor to a line of the polygon:
- right-click
- select menu item **Adding a node in the selected element** from the context menu

or:

- press Ctrl and Shift simultaneously
- move the mouse cursor to a line of the polygon:
- the mouse cursor changes to an arrow with a plus-symbol

- left-click in order to add a corner

REMOVING CORNERS:

- move the mouse cursor to a corner of the polygon:
- right-click
- select menu item **Deleting the node in the selected element** from the context menu

or:

- press Ctrl and Shift simultaneously
- move the mouse cursor to a corner of the polygon:
- the mouse cursor changes to an arrow with a plus-symbol
- left-click to delete the corner

> **Attention**
>
> *If you make a Polyline into a polygon, the touch points for the start and end of the line are invisible where they overlap if the invert background option is activated. You deactivate this option under Option Menu-> Settings-> Corner Points-> Invert Background.*

10.19 Rectangle

To draw a rectangle:

1. select the Rectangle symbol in the **Elements** tool bar or in the **Elements** drop-down list
2. select the start point in the main window
3. drag the rectangle while pressing and holding the left mouse button
4. The shape, size and position can be changed at any time by pulling the vertexes or moving the element

5. if you press and hold `alt` while pulling the outer corner points, the change is carried out symmetrically
6. define the desired properties in the **`properties window`**

Hint: You can round (on page 195) the corners at will in order to created a rectangle with rounded corners

10.20 Square

In order to draw a square:

1. select the `Rectangle` symbol in the **`Elements`** tool bar or in the **`Elements`** drop-down list
2. hold down the `shift` key
3. select the start point in the main window
4. pull open the square while pressing and holding the left mouse button
5. The shape, size and position can be changed at any time by pulling on the corner points or moving the element;
   if the 'square' shape is to be retained hold down the `shift` key whilst pulling on the corner points
6. if you press and hold `alt` while pulling the outer corner points, the change is carried out symmetrically
7. define the desired properties in the **`properties window`**

10.21 Tube

To draw a tube:

1. select the `Tube` symbol in the **`Elements`** tool bar or in the **`Elements`** drop-down list
2. In the main window, select the start point by clicking the left mouse button
3. pull the first section

4. place an initial corner point with a simple mouse click (left)
5. draw another line and place the corner points
6. set the end point by double clicking or pressing the `esc` key
7. The shape, size and position can be changed at any time by pulling the vertexes or moving the element
8. hold down the `shift` key when pulling; this way, only horizontal or vertical movements away from the original position are possible
9. define the desired properties in the Properties window

Hint: You define the rounding (on page 195) of the lines with the `Rounding` property. To do this, select a value between 0 (no rounding) and 1 (maximum rounding).

ADDING CORNERS:

To add corners (corner points):

- move the mouse cursor to a line of the polygon:
- right-click
- select menu item **`Adding a node in the selected element`** from the context menu

or:

- press `Ctrl` and `Shift` simultaneously
- move the mouse cursor to a line of the polygon:
- the mouse cursor changes to an arrow with a plus-symbol
- left-click in order to add a corner

REMOVING CORNERS:

- move the mouse cursor to a corner of the polygon:
- right-click
- select menu item **`Deleting the node in the selected element`** from the context menu

or:

- press `Ctrl` and `Shift` simultaneously
- move the mouse cursor to a corner of the polygon:
- the mouse cursor changes to an arrow with a plus-symbol
- left-click to delete the corner

10.22 Switch

You show the value of a binary variable in graphic form and modify it with the **`Switch`** dynamic element. The element can also be used as a switch for one-stage unsecured command input.

To create **`Switch`** in a screen:

- select the **`Switch`** symbol in the **`Elements`** tool bar or in the **`Dynamic Elements`** drop-down list
- select the start point in the main window
- pull open the element while pressing and holding the left mouse button
- the dialog to select a binary variable (on page 238) opens
- select the variable you want to assign
- define the desired properties in the properties window

To use the element as a switch, activate the `Momentary switch` property in the `Write set value` node

10.23 Static Text

To enter statistical text:

1. select the `Statistical Text` symbol in the **`Elements`** tool bar or in the **`Elements`** drop-down list
2. select the start point in the main window
3. pull the text field open while pressing and holding the left mouse button

4. The shape, size and position can be changed at any time by pulling the vertexes or moving the element
5. if you press and hold the alt key while pulling the outer corner points, the change is carried out symmetrically
6. define the desired properties in the Properties window

To enter text:

1. click on the text field with a delayed double click

or: select Edit in the context menu

2. write the desired text

> **Info**
>
> *The text is automatically wrapped as standard, depending on the size of the text field. To turn off the wrapping, deactivate the property `Word wrap` in the `Representation` node.*
>
> ***Hint:** If a `Text angle [°]` (`Representation`) is defined in the property, it is not possible to wrap the text. An angle display with text wrapping can also be achieved by rotating the whole element (including the frame lines, if activated).*

UNDERLINE

To underline text, place a & in front of the text to be underlined.

&Text becomes:

If you would like to use the character '&' in text, enter it twice:
1 && 2 becomes 1 & 2.

LINE BREAK

To define a line break, use the string **\n**.

Example:

Line 1 \n

Line 2

> **Info**
>
> *Input is no longer limited to 80 characters from version 6.51 onwards.*

10.24 Move symbol (up to version 6.22 only)

With the **Move Symbol** dynamic element, you can move elements you wish to move horizontally or vertically or change their size (zoom) regardless of the status of other variables. To do this, connect the elements to a symbol.

To create **Move Symbol** in a screen:

- select the **Move Symbol** symbol in the **Elements** tool bar or in the **Elements** drop-down list
- select the start point in the main window
- pull open the element while pressing and holding the left mouse button
- select the desired variable and define the desired properties in the properties window
- You can only use symbols that are already in the screen

The **Move Symbol** dynamic element is available up to version 6.22. From version 6.50, use dynamic properties of an element such as `Position`, `Size and rotation dynamic` or `Color` for all dynamic actions.

10.25 Trend Element

With the **Trend Element** dynamic element, you display all values in runtime in the form of trend curves, where several process variables can be displayed at the same time.

In order for the **Trend Element** to display values, the `Harddisk data storage active` property in the `Harddisk data storage` node of variable properties must be active. `Cycle time [s]` and `Number of values` must concur with `Update time` and the`Time of representation`

in **Trend Element** , otherwise the screen to draw curve will be restarted each time it is shown. Each change can have an effect on other **Trend Elements** that use the same process variables.

CREATE A TREND ELEMENT

To create a **Trend Element** in a screen:

1. select the **Trend Element** symbol in the **Elements** tool bar or in the **Elements** drop-down list
2. select the start point in the main window
3. pull open the element while pressing and holding the left mouse button
4. the dialog to assign several numeric variables (on page 238) opens
5. select the desired variables
6. all curves are automatically created in the **Trend Element**

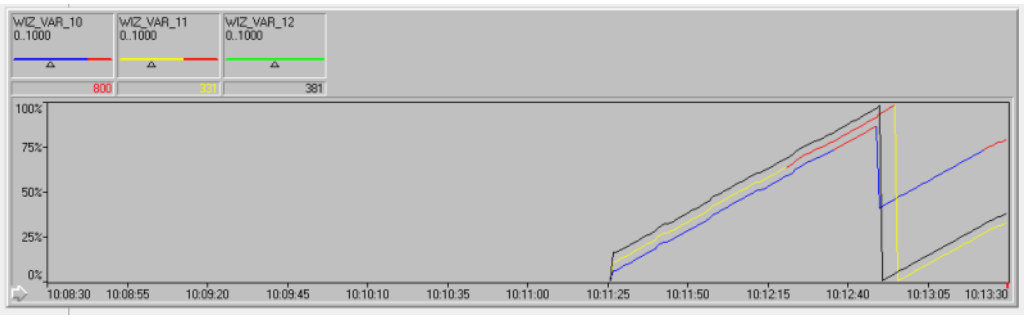
To add a curve later:

- In properties, select the {Curve new} property in the Curves node

or:

- drag the desired variable onto the **Trend-Element** while holding the left mouse button

7. Define the desired properties in the properties window
   - You define Graphic properties of the curves in the Curves node of properties
   - You determine cycle time and number of values in the properties of the respective variable in the Harddisk data storage node



Info

Functions and variables can also be linked to the Element by Drag&Drop.

10.26 Clock

With the **Clock** dynamic element, you show the current time in Runtime or you convert numerical values into a time. The date/and or the time can be displayed as an absolute time or a time difference.

ADVERTISEMENT

The display in Runtime depends on whether a variable was linked.

- If no variable is linked to the element, the current system time is displayed in absolute time in Runtime.
  To do this, set, in the `Representation` group, the `Type of display` property to the value `System time/absolute time`.
- If a variable is linked, then its value can be displayed as:
  - `Time difference`: The variable must contain the number of seconds.
    The integer values in the seconds area `-2147483648 (min(DINT))` to `+ 4294967295 (max(UDINT))` can be displayed.
    It is therefore also possible to display negative values: A variable value of `-60` is displayed in `mm:ss` format as `-01:00`. Thus if `-01:40` is this displayed, the resulting variable value is `-100`.
  - `Absolute time`: The Unix time - the seconds passed since January 1, 1970 00:00 UTC in the positive range.

  Only positive values up to 2147483647 can be displayed. A variable value outside this area leads to an incorrect display: The `hh:mm` part remains `00:00` or `01:00` in winter time.

CONFIGURATION

To create a **Clock** in a screen:

1. select the **Clock** symbol in the **Elements** tool bar or in the **Elements** drop-down list
2. select the start point in the main window

3. pull open the element while pressing and holding the left mouse button
4. If a variable is to be linked, then:
   Link the desired variable in the `Variable / function` section or by dragging & dropping the element
   - All variables can be linked, with the exception of string and Boolean variables
   - Fractions of float variables (from data types REAL and DATA_AND_TIME for example) are not taken into account.
5. Define the desired properties in the properties window, most of all:
6. Select the `Type of display` property in the `Representation` group
   - `System time/absolute time:` The time is displayed as an absolute time.
     Note: Context menu and tooltip are not supported.
   - `Time difference:` The time difference is given in minutes and seconds The linked variable must contain the number of seconds.
     The integer values in the seconds area `-2147483648 (min(DINT))` to `+4294967295 (max(UDINT))` can be displayed. It is therefore also possible to display negative values.
     Examples in `mm:ss` format: The variable value `50` is displayed as `00:50`, the variable value `100` is displayed as `01:40.`
     The variable value `-60` is displayed as `-01:00`. Thus if `-01:40` is this displayed, the resulting variable value is `-100`.
7. Select the desired display using the `Formating` property

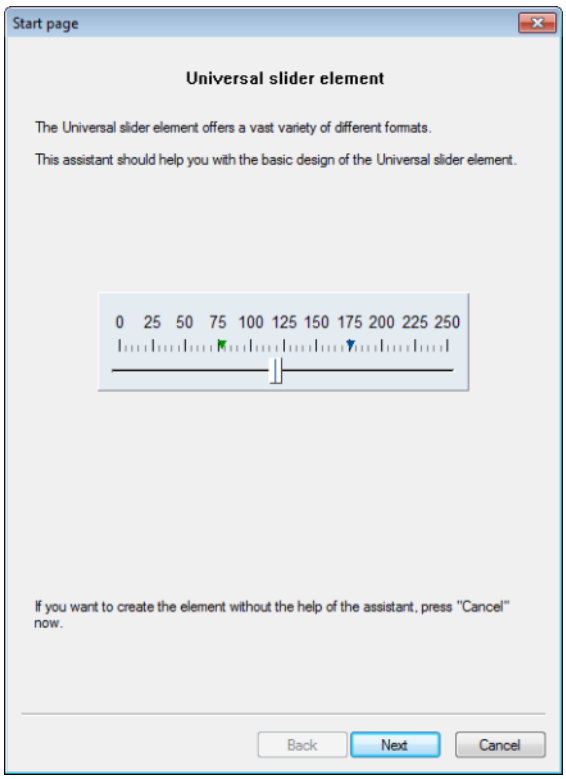
> **Info**
>
> *With `absolute time` with the `Write set value via element`, the input field has `time difference` formatting.*
>
> *For example:*
>
> - Absolute time formatting: `hh:mm:ss dd.mm.yy`
>   is displayed in the element as: `dd:hh:mm:ss`
> - `dd.mm.yyyy` (for example 17.12.2008)
>   as `ddddd` (14230 days since 1. 1. 1970).

10.27 Universal slider

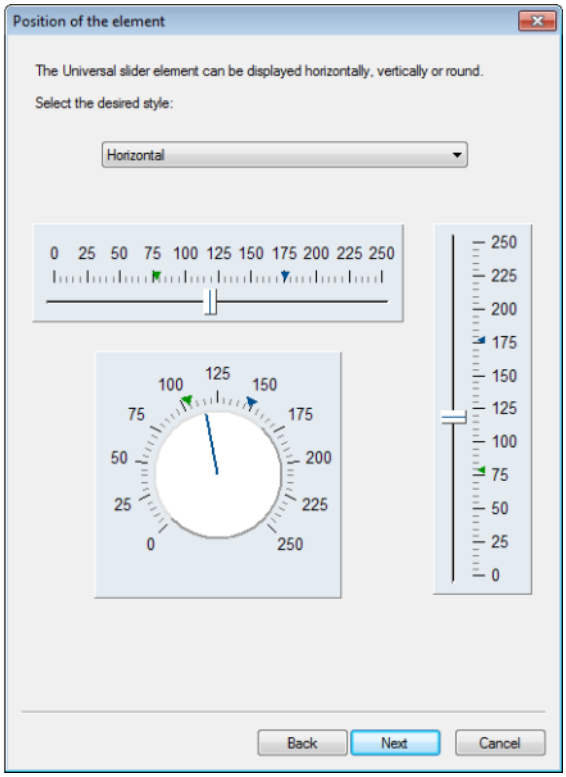
With the **Universal slider** element, you show variables in the form of sliders, bar graphs, LED bar displays or any other sliders you wish. The sliders allow set value elements to be set in runtime.

To create the **universal slider** in a screen:

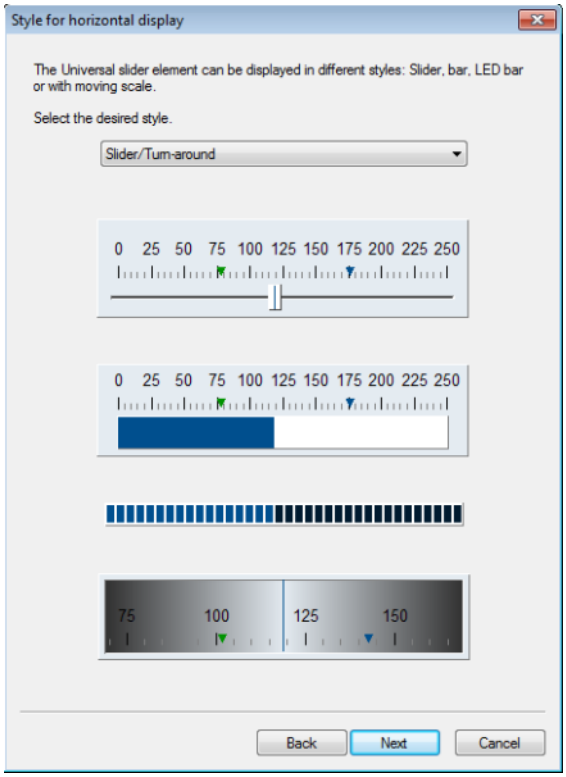
1. select the **Universal Slider** symbol in the **Elements** tool bar or in the **Elements** drop-down list
2. select the start point in the main window
3. pull open the element while pressing and holding the left mouse button
4. the dialog to assign a variable (on page 238) opens
5. select the variable you want to assign
6. the universal slider configuration assistant opens



7. select the alignment:
   horizontal, vertical or round

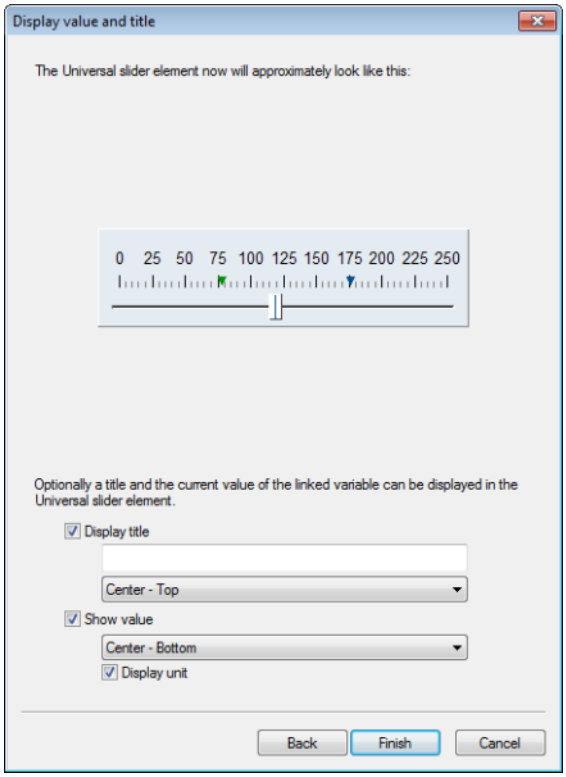


8. select the shape:
   slider/turn-around, bar, LED bars or moving scale



9. You a given a preview:

   Define the title display and position of the value display.



10. You define other properties in the properties window

Hint: You change the display and shape of a universal slider in properties in the `Universal slider` node

> **Info**
>
> A drag indicator can display the minimum or maximum value of the variables within a certain period of time. Double-clicking the element in runtime resets the marking of minimum and maximum.

COLOR-CODED DISPLAY OF THE LIMIT VIOLATION

At limit violations the color from the variable can color the whole area or only the area starting from the limit violation. This functionality is controlled via property `Bar explicit` in group `Universal slider`.

This property affects only the display of values with defined limits from variables. Limits from reaction matrices are not considered.

`Active`: If a limit of the linked variable is violated, only the part of the display which reaches into the limit violation is displayed in the color of the respective limit.
If the variable has several defined limits, the respective areas of the display are always displayed in the color of the corresponding limit.

`Inactive`: If a limit of the linked variable is violated, the complete display is displayed in the color of the violated limit.

Default: `inactive`

10.28 WPF-Element

With the `WPF` dynamic element, valid WPF/XAML files in zenon can be integrated and displayed.

> **Info**
>
> *All brand and product names in this documentation are trademarks or registered trade marks of the respective title holder.*

10.28.1 Basics

XAML

XAML stands for **`Extensible Application Markup Language`**. The description language developed by Microsoft and based on XML defines the syntax in Silverlight applications and WPF user interfaces. XAML makes it possible to separate design and programming. The designer prepares the graphical user interface and creates basic animations that are then used by the developers/project planners. The project planner can control these .xaml files in a purposeful manner and animate them accordingly.

WPF

WPF stands for Windows Presentation Foundation and describes a graphics framework that is part of the Windows .NET framework:

- WPF displays the programming environment.
- XAML describes, based on XML, the interface hierarchy as a markup language. Depending on the construction of the XAML file, there is the possibility to link properties, events and transformations of WPF elements with variables and functions of CD_PRODUCTNAME<.
- The framework unites the different areas of presentation such as user interface, drawing, graphics, audio, video, documents and typography.

According to Microsoft, WPF is supplied with Windows 7, Windows Vista and Windows Server 2008 and is available for Windows XPSP2 or later and Windows Server 2003. Microsoft .NET 3.5 is required for execution.

WPF in process visualization

XAML makes different design possibilities possible for zenon. Display elements and dynamic elements can be adapted graphically regardless of the project planning. For example, laborious illustrations are first created by designers and then imported into zenon as an XAML file and linked to the desired logic. There are many possibilities for using this, for example:

DYNAMIC ELEMENTS IN ANALOG-LOOK

Graphics no longer need to be drawn in zenon, but can be imported directly as an XAML file. This makes it possible to use complex, elaborately illustrated elements in process visualization. Reflections, shading, 3D effects etc. are supported as graphics. The elements that are adapted to the respective industry environment make intuitive operation possible, along the lines of the operating elements of the machine.

INTRICATE ILLUSTRATIONS FOR INTUITIVE OPERATION

The integration of XAML-based display elements improves the graphics of projects and makes it very easy to display processes clearly. Elements optimized for usability make operation easier. A clear display of data makes it easier to receive complex content. The flexible options for adapting individual elements makes it easier to use for the operator. It is therefore possible for the project planners to determine display values, scales and units on their own.

CLEAR PRESENTATION OF DATA AND SUMMARIES

Grouped display elements make it possible to clearly display the most important process data, so that the equipment operator is always informed of the current process workflow. Graphical evaluations, display values and sliders can be grouped into an element and make quick and uncomplicated control possible.

INDUSTRY-SPECIFIC DISPLAYS

Elements such as thermometers, scales or bar graphs are part of the basic elements of process visualization. It is possible, using XAML, to adapt these to the respective industry. Thus equipment operators can find the established and usual elements that they already know from the machines in process visualization at the terminal.

ADAPTATION TO CORPORATE DESIGN

Illustrations can be adapted to the respective style requirements of the company, in order to achieve a consistent appearance through to the individual process screen. For example, the standard operation elements from zenon can be used, which can then be adapted to color worlds, house fonts and illustration styles of the corporate design.

Transfer of values from zenon to WPF

zenon always works internally with the **`double`** or **`string`**. These are sent to the WPF element. The WPF element is embedded in a .NET container. It usually needs to be converted so that the data type can be used. This conversion can automatically be carried out by .NET.

The values are sent in accordance with the following rules:

- If the .NET type (**`System.Object`**) for zenon is not evident, the value is sent as it is to .NET. .NET must take care of the display or conversion itself.
- If the .NET type is a Boolean type (**`System.Boolean`**), then zenon writes according to the .NET convention `0` or `-1`.
- If the .NET type is known, a check is carried out to see if .NET can convert the value. The converter from .NET is used for this.
  - Yes: The value is sent.
  - No: The value is sent nevertheless. If .NET reacts with an error message, the value of zenon is converted into a string and sent again.

Referenced objects

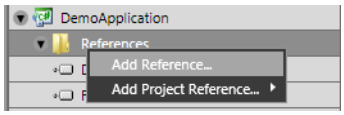
In WPF not only standard objects such as rectangles, buttons, text fields, etc. can be used, but also WPF user controls, which are referenced as assemblies.

WPF user controls are individually created objects. For example, this element can look like a tacho and provide special properties and optical effects, such as a "Value" property, which causes the pointer of the tacho to move and display the value when it is set.

The workflow for this:

- The appearance of a user controls is labeled with standard objects, which are offered by WPF.
- The properties and interactions are programmed.
- The whole package is compiled and present in the form of a .NET assembly.

This assembly can also be used for WPF projects. To do this, it must be referenced (linked) in the WPF editor (for example: Microsoft Expression Blend). To do this, select the assembly in the zenon file selection dialog:



From this point in time, the WPF user controls of the assembly in the tool box can be selected under **Custom user controls** and used in the WPF project.

USED REFERENCED ASSEMBLIES IN ZENON

To use an assembly in zenon, this must be provided as a file.
Collective files in **.cdwpf** format administer these independently; no further configuration is necessary. Assemblies must be added to the Files folder for .xaml files:

- Click on Files on the project tree
- Select Other
- Select **Add file...** in the context menu
- The configuration dialog opens
- Insert the desired assembly

When displaying a WPF file in the **WPF element** (Editor and Runtime), the assemblies from this folder are loaded. It is thus also ensured that that when the Runtime files are transferred using **Remote Transport**, all referenced assemblies are present on the target computer.

A collective file (**.cdwpf**) can exist alongside an XAML file with the same name. All assemblies (*.dll) from all collective files and the Other folder are copied to the work folder. Only the highest file version is used if there are several assemblies with the same name.

> **Attention**
>
> *Assemblies are only only removed after loading when the application is ended. That means:*
>
> *If a WPF file with a referenced assembly in zenon is displayed, then this assembly is loaded is in the memory until zenon is ended, even if the screen is closed again. If you would like to remove an assembly from the `Files/Other` folder, the Editor must first be restarted, so that the assembly is removed.*

Allocation of zenon object to WPF content

zenon objects are allocated to WPF content using the `name` of the WPF object. In doing so, note:

Visual objects do not have a **`RuntimeNamePropertyAttribute`** property. Therefore at the time when the WPF content is loaded and created, the additional information of `name` is not available.

Thus a clear allocation of zenon objects to WPF objects is not possible. Therefore only logical objects are listed in the configuration dialog of zenon. Which WPF objects the **`RuntimeNamePropertyAttribute`** has available is visible in MSDN or on the Microsoft website.

WORKAROUND

Nevertheless, the following workaround is possible to animate visual objects:

For visual elements, the animateable property is linked to the text property of an invisible text box using a data connection.

Because the text box as a logical object supports the **`name`** property, this is displayed in zenon.

The textbox property can also be animated with zenon.

This visual object is also indirectly animated as a result.

Workflows

The WPF/XAML technology makes new workflows in process visualization possible. The separation of design and functionality ensures a clear distinction of roles between the project planners and designers; design tasks can be easily fulfilled by using pre-existing designs, which no longer need to be modified by the project planner.

The following people are involved in the workflow to create WPF elements in zenon:

- Designer
  - illustrates elements
  - takes care of the graphics for MS Expression Design
- MS Expression Blend operator
  - Animates elements
  - Creates variables for the animation of WPF elements in zenon, which project planners can access
- Project planner
  - Integrates elements into zenon:
  - stores logic and functionality

We make a distinction:

- Workflow with Microsoft Expression Blend (on page 132)
- Workflow with Adobe Illustrator (on page 133)

Workflow with Microsoft Expression Blend

When using Microsoft Expression Blend, a WPF element is created in four stages:

1. Illustration of elements in **MS Expression Blend** (on page 134)
2. Open element in **MS Expression Design** and export as WPF
3. Animation in **MS Expression Blend** (on page 134)
4. Integration into zenon (on page 177)

You can find an example for creating a WPF elements with Microsoft Expression Blend in the Create button as XAML file with Microsoft Expression Blend (on page 134) chapter.

Workflow with Adobe Illustrator

Based on traditional design processes with **Adobe Illustrator** the following workflow is available:

1. Illustration of elements in **Adobe Illustrator** (on page 139)
2. Import of **.ai** files and preparation in **MS Expression Design** (on page 141)
3. WPF export from **MS Expression Design** (on page 141)
4. Animation in **MS Expression Blend** (on page 143)
5. Integration into zenon (on page 185)

You can find an example for creation in the Workflow with Adobe Illustrator (on page 138) chapter.

10.28.2 Manual for designer

This section informs you how to correctly create WPF files in Microsoft Expression Blend and Adobe Illustrator. The tutorials on Creating a button element (on page 134) and a bar graph element (on page 138) show you how fully functional WPF files for zenon can be created from pre-existing graphics in a few steps.

The following tools were used for this:

- Adobe Illustrator CS3 (AI)
- Microsoft Expression Design 4 (ED)
- Microsoft Expression Blend 4 (EB)
- zenon 6.51

> **Info**
>
> *If referenced objects (assemblies) are used in WPF, note the instructions in the Referenced objects (on page 129) chapter.*

Workflow with Microsoft Expression Blend

With Microsoft Expression Blend, a WPF element:

- is illustrated
- is converted into WPF format using **`MS Expression Design`**
- animated

The following example shows the illustration and conversion of a button element into an XAML file.

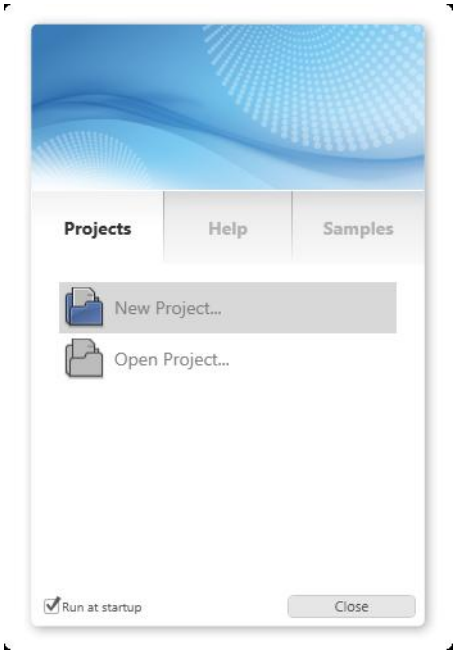
Note: A test version of "Microsoft Expression Blend" can be downloaded from the Microsoft website.

Create button as XAML file via Microsoft Expression Blend

CREATE BUTTON

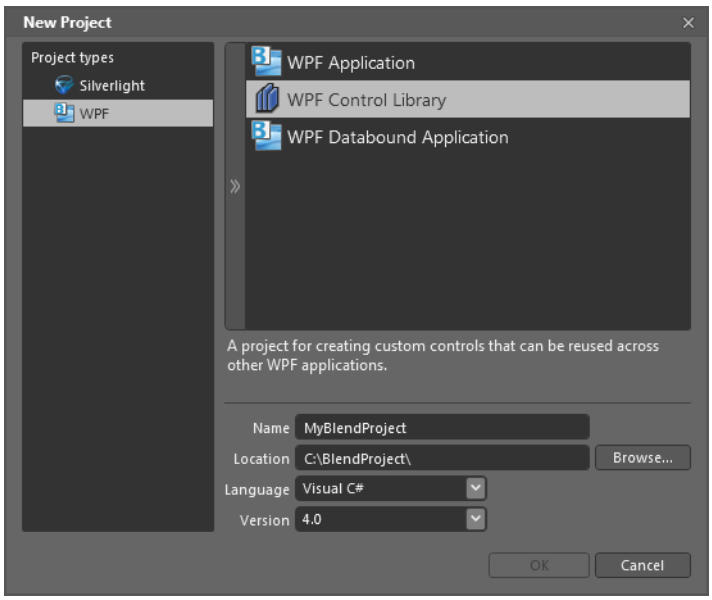
1. Start Expression Blend

2. select the **`New Project`** option



3. Select `WPF` as project type

4. give it a path and name of your choice (MyBlendProject, for example)



The **`Language`** and **`Version`** settings can be ignored, because no functionality is to be programmed.

5. After the dialog has been confirmed with **OK**, Microsoft Blend creates a new project with the chosen settings. Expression Blend adds an empty XAML file which already contains a class reference.

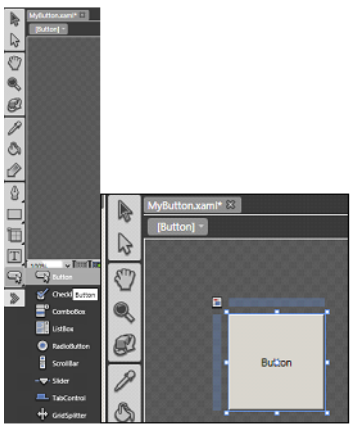
6. Delete the CS file that belongs to the XAML file using the context menu.

7. Rename the XAML file **`MainControl.xaml`** to **`MyButton.xaml`**.

8. The development size of the file is set at 640 x 480 pixels as standard and must still be changed:
   a) switch to **XAML** view
   b) correct the size to 100 x 100 pixels
   c) Delete the class reference x:Class="MyBlendProject.MyButton"

```
MyButton.xaml
1  <UserControl
2      xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
3      xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
4      xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
5      xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
6      mc:Ignorable="d"
7      x:Name="UserControl"
8      d:DesignWidth="100" d:DesignHeight="100">
9
10     <Grid x:Name="LayoutRoot" />
11
12 </UserControl>
```

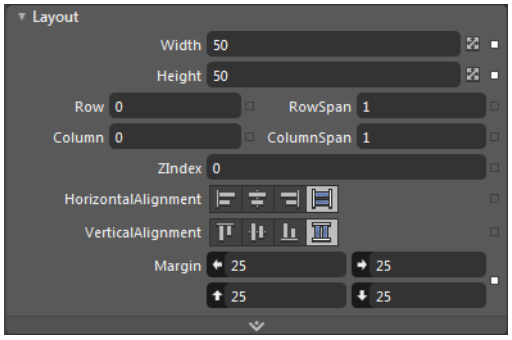
9. switch to **`Design`** view



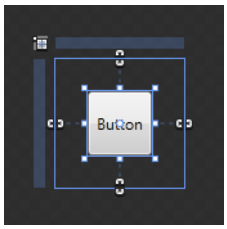
10. add a button via the tool bar

11. define the properties
    - Width: 50
    - Height: 50

- Margins: 25



The button is therefore at the center of the control.



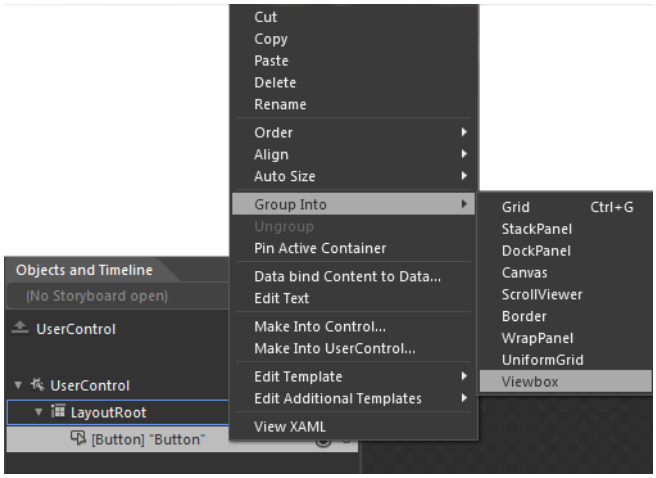
12. Save the changes and open the file in Internet Explorer to check it. You will see that the button is displayed in a size of 50 x 50 pixels.

MAKE BUTTON SCALABLE

If you integrate this status into zenon, the button will always have the exact size of 50 x 50 pixels. Because the button can be implemented as a scalable button, switch to Expression Blend again:

1. select the button in the tree view
2. select the *Group Into->Viewbox* button in the context menu
3. the button is inserted into a **`Viewbox`**
4. Define the properties of the viewbox
   - Width: Auto
   - Height: Auto

5. save the file



6. If you now open the file in Internet Explorer, the button is automatically scaled when the IE window size is changed. This file will now also automatically adapt to changes in the size of the `WPF element` in zenon.

CHANGE NAME

Before you can integrate the file into zenon, you must give the `WPF element` a name. The `WPF elements` are not named in Expression Blend as standard, and are labeled with square brackets and their type. zenon content is assigned to WPF content via the name of the `WPF elements`:

- in tree view, change the name
  - of the button on **`MyButton`**
  - of the ViewBox to **`MyViewBox`**

This button can now be integrated in zenon (on page 177) as an XAML file.

Workflow with Adobe Illustrator

When **`Adobe Illustrator`** is used, a WPF element:

- is illustrated in **`Adobe Illustrator`**
- is converted into a WPF in **`MS Expression Design`**
- is animated in **`MS Expression Blend`**

The following example shows the illustration and conversion of a bar graph element into an XAML file.

Bar graph illustration

A bar graph is created in Adobe Illustrator.

1. AI: Starting element for bar graph

Illustrated in Adobe Illustrator CS3.

2. AI: Path view of bar graph in Adobe Illustrator

- All effects must be converted (`Object -> Convert appearance`)
- All lines are transformed into paths (`Object -> Path -> Contour line`)
- Do not use filters such as shading, blurring etc.

NOTES ON COMPATIBILITY

Illustrations that were created with Adobe Illustrator are in principle suitable for WPF export. However, not all Illustrator effects can become corresponding effects in Expression Design/Blend. Note:

Effect	Description
Clipping masks	Clipping masks created in Adobe Illustrator are not correctly interpreted by Expression Design. These are usually shown in Blend as areas of black color. We recommend creating illustrations without clipping masks.
Filters and effects	Not all Adobe Illustrator filters are transferred into Expression Design accordingly: Thus blurring filters, shading filters and corner effects from Illustrator do not work in Expression Design. Solution: ▸ Most effects can be converted so that they can be read correctly by Expression Design using the **Object -> Convert appearance** command in Adobe Illustrator. ▸ Corner effects from Adobe Illustrator are correctly interpreted by MS Design if they are converted to AI in paths.
Text fields	To be able to link text fields with code, these must be created separately in Expression Blend. „**Labels**" are required for dynamic texts; simple "**text fields**" are sufficient for static information. There is no possibility to create text labels in MS Design. These must be directly created in MS Blend.
Transparencies and group transparencies	There can be difficulties in Adobe Illustrator with the correct interpretation of transparency settings, in particular from group transparency settings. However MS Expression Blend and MS Expression Design do offer the possibility to create new transparency settings.
Multiply levels	These level settings in Adobe Illustrator are not always correctly displayed by MS Expression Blend. However, there is the possibility to "**Multiply levels**" directly in Expression Design.
Indicating instruments and standard positions	To prepare the graphics optimally for animation, the indicator and slider must always be set to the starting position, usually 0 or 12:00 o'clock. Thus the position parameters for rotations etc. are also correct in Blend and an animation can be implemented without conversion of position data.

WPF-Export

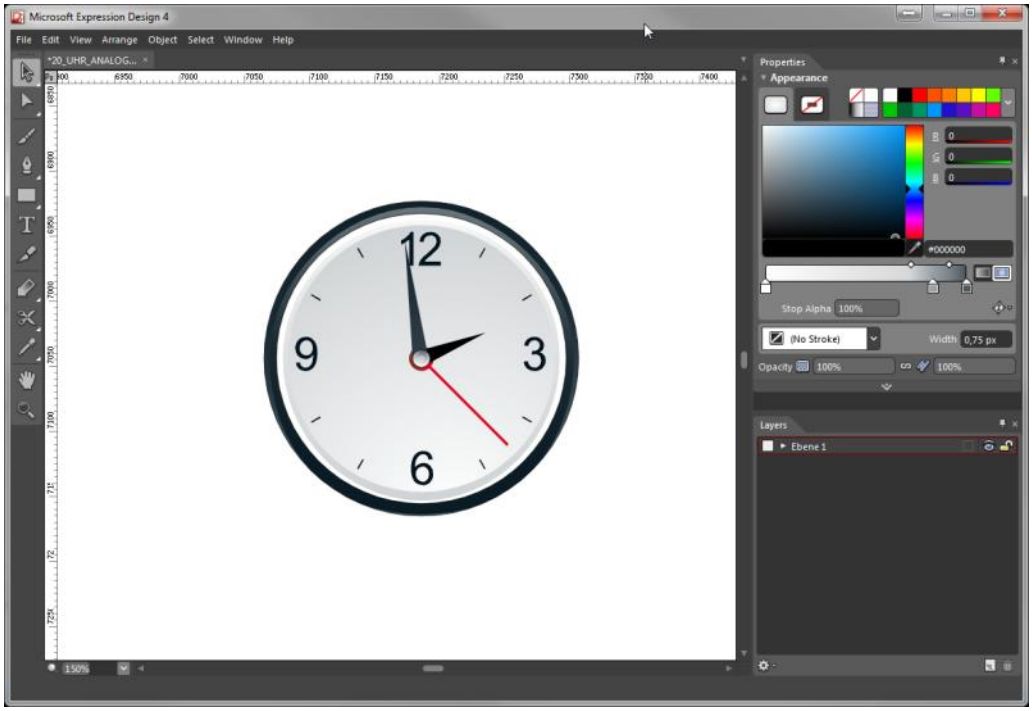
WPF files are required for animation in Microsoft Expression Blend. We recommend Microsoft Expression Design for this export, because it provides good results and most Illustrator effects are correctly interpreted.

Note: There is a free plug-in for the direct export of WPF files from Adobe Illustrator available on the internet. This plug-in provides a quick, uncomplicated way of exporting from Illustrator, however it is less suited to the current application because it lead to graphical losses. Even color deviations from the original document are possible.

Files in `.ai` format can regularly be imported into Expression Design; the paths are retained in the process.
Attention: Some common Illustrator effects cannot be displayed by Expression Design correctly however (see Illustration (on page 139) chapter).

We export the pre-created bar graph element in 5 stages:

1. ED: Import
   - Import the prepared Illustrator file (on page 139) in **`Microsoft Expression Design`** via *File -> Import*
2. ED: Optimization



- If the starting file is not correctly displayed in MS Expression Design, it can still be subsequently edited and optimized here

3. ED: Selection

- Highlight the element for WPF export with the **`direct selection`** arrow in MS Expression Design; in this case it is the whole clock

4. ED: Start export

- Start the export via *File -> Export*
- the dialog for configuring the export settings opens

5. ED: Export settings

- Enter the following export settings:

a) `Format: XAML Silverlight 4 / WPF Canvas`

`Always name objects`: Activate with tick

`Place the grouped object in an XAML layout container`: Activate with tick

b) `Text: Editable text block`

c) `Line effects: Rasterize all`

The exported file has **`.xaml`** file suffix. It is prepared and animated (on page 143) in MS Expression Blend in the next stage.

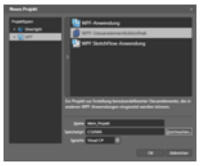
Animation in Blend

With MS Expression Blend:

- static XAML files from MS Expression Design are animated
- Variables for controlling effects that can be addressed by zenon are created

In thirteen steps, we go from a static XAML to an animated element, that can be embedded in zenon:

1. EB:create project

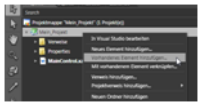


a) Open Microsoft Expression Blend
b) Create a new project
c) Select the **`Project type`** of *WPF- >WPF Control Library*
d) Give it a name (in our tutorial: **`My_Project`**)
e) Select a location where it is to be saved
f) Select a language (in our tutorial: C#)
g) Select Framework Version 3.5

2. EB: delete MainControl.xaml.cs

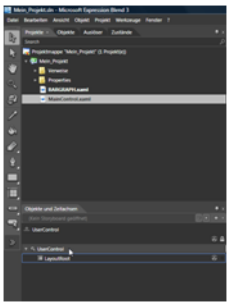
a) Navigate to **`MainControl.xaml.cs`**
b) Delete this file using the **`Delete`** command in the context menu

3. EB: Open exported XAML file



a) Open the context menu for **`My_Project`** (right mouse button)
b) Select **`Add existing element`**…
c) Select the XAML file exported from Microsoft Expression Design, in order to open this in Microsoft Expression Blend

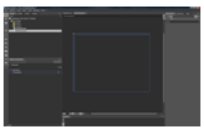
4. EB: Open MainControl.xaml



a) Open the automatically created **MainControl.xaml**

b) In the **Objects and Time axes** area, navigate to the **UserControl** entry

5. EB: Adapt XAML code



a) Click on **UserControl** with the right mouse button

b) Select **Display XAML** in the contextual menu.

c) Delete lines 7 and 9 in the XAML code:

```
x:Class="My_Project.MainControl"
d:DesignWidth="640" d:DesignHeight="480"
```

6. EB: check XAML code

- The XAML code should now look like this:

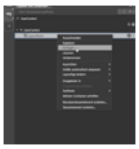
```
<UserControl
    xmlns=http://schemas.microsoft.com/winfx/2006/xaml/presentation
    xmlns:x=http://schemas.microsoft.com/winfx/2006/xaml
    xmlns:d=http://schemas.microsoft.com/expression/blend/2008
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"


    mc:Ignorable="d"
    x:Name="UserControl">
```

```
    <Grid x:Name="LayoutRoot"/>
</UserControl>
```

7. EB: Copy elements

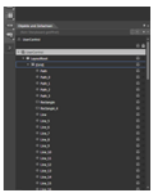
a) Open the XAML file imported from Expression Design
b) Mark all elements
c) Select **`Delete`** in the context menu
d) Change back to the automatically created XAML file

8. EB: Insert element



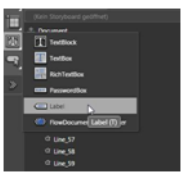
a) Click on **`Layout Root`** with the right mouse button
b) Select **`Insert`**

9. EB: Adapt layout type



a) Click on *Layout root -> Change layout type -> Viewbox* with the right mouse button
b) The structure should now look like this: **`UserControl -> LayoutRoot -> Grid -> Elements`**
c) Give a name for **`LayoutRoot`** and **`Grid`** by double-clicking on the names

10. EB: Texts and values



- Dynamic and static texts are labeled with text fields

- Values (numbers) are issued with **`Labels`**

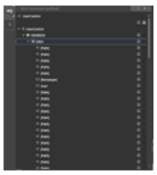
11. EB: Insert labels

- Labels replace numbers that are to be subsequently linked using INT variables (must be carried out for all number elements)

12. EB: Set property

- To display 100%, set the bar graph element's **`MaxHeight`** property to `341` (the maximum height of the indicator element is `340`)

13. EB: prepare for use in zenon



a) Delete all name labels (names may only be given for elements that are to be addressed via zenon)

b) Save the XAML file with any desired name

c) Integrate the XAML file into zenon (on page 185)

A tip for checking: If the XAML file is displayed with no problems in Microsoft Internet Explorer and the window size of Internet Explorer adapts to it, it will also be correctly used in zenon.

10.28.3 Engineering in zenon

To use WPF with zenon, Microsoft Framework 3.5 must be installed on both the editor computer and on Runtime.

CONDITIONS FOR WPF DISPLAY IN ZENON

The animation is currently available for simple variables; arrays and structures cannot be animated. Therefore the following WPF functions can be implemented in zenon:

- Element properties that correspond to simple data types, such as `String`, `Int`, `Bool` etc.
- Element properties of the "Object" type, which can be set with simple data types
- Element events can be used with functions; the parameters of the events are not however available in and cannot be evaluated in zenon
- Element transformation, for which a **`render transform`** is present for the element in the XAML file

Attention: if the content is outside of the area of the WPF element during transformation, this part of the content is lost or is not labeled

> **Attention**
>
> *If the Runtime files were created for a project for a version **before** 6.50, existing `WPF elements` are not included into Runtime screens.*

CREATE WPF ELEMENT

To create a WPF element

1. in the elements toolbar, select the symbol for `WPF element` or the **`Elements`** entry in the menu
2. select the start point in the main window
3. pull open the element with the mouse
4. In properties, select `Representation` the property`XAML file` in the group
5. the file selection dialog opens
6. Select the desired file
   Files of the following formats are valid
   - *.xaml: Extensible Application Markup Language
   - *.cdwpf: WPF collective file, also shows preview image

(the file must already be present in the Project Manager under **Files/graphics** or created in the dialog.)

7. configure the link (on page 148)

> **Info**
>
> *If referenced objects (assemblies) are used in WPF, note the instructions in the Referenced objects (on page 129) chapter.*

CDWPF FILES (COLLECTIVE FILES)

Rules for the use of collective files:

- The files can be in the ZIP file directly or in a joint folder.
- The name of the XAML file should correspond to the names of the collective file.
- Only one XAML file may be contained.
- The preview graphic should be small and no more than 64 pixels high.
  Name of the preview file: **preview.png** or the name of the XAML file with the suffix **png**.
- Any number of assemblies can be used. The distinction is made on the basis of the file version in numerical form.
- Collective files do not need to contain an assembly.
- All folders are searched and only ***.dll**, ***.xaml** and ***.png** files are taken into account.

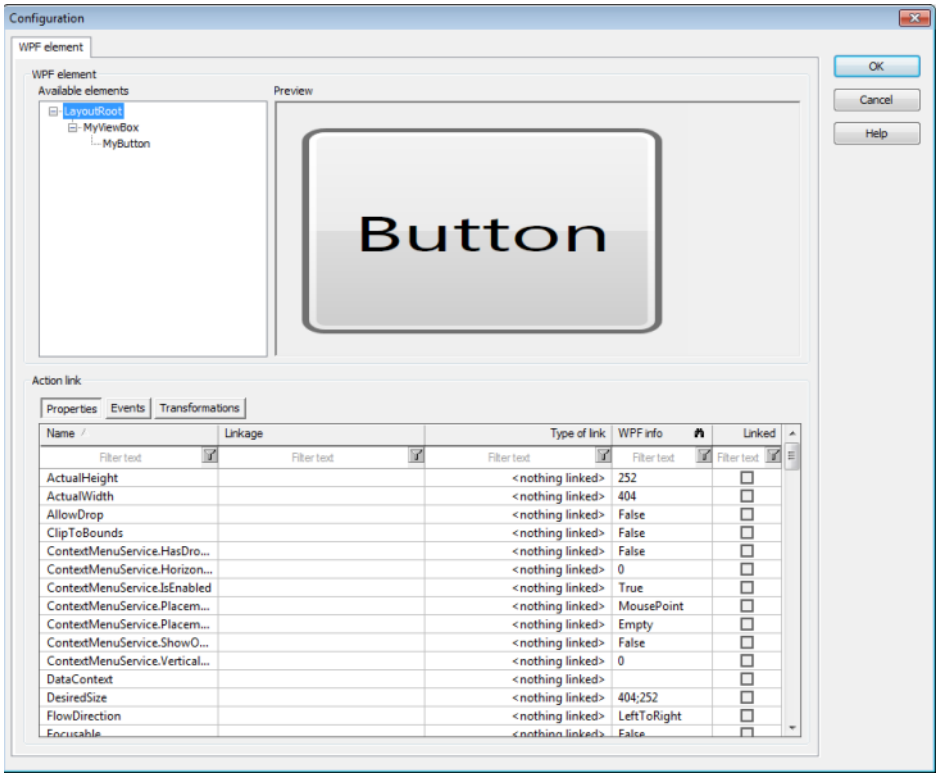
DISPLAY UNDER WINDOWS VISTA/WINDOWS 7

If a WPF screen contains a slider and Windows Vista or Windows 7 Aero Effects are used, this may lead to refresh problems in zenon Editor.

Configuration of the link

To configure a WPF element

1. In properties, select the `Configuration` property in the Display group

2. The dialog with three tabs opens with a preview of the XAML file and the elements present in the file 

Parameters	Description
Existing elements	Shows the named file elements in a tree structure. The selected element can be linked with process data. WPF is assigned to process data based on the element name. Therefore elements are only shown if they and the attendant elements have a name. Allocations are configured and shown in the **Properties**, **Events**, **Transformations** tabs.
Preview	The selected element is shown flashing in the preview.
Properties (on page 150)	Configuration and display of properties (variables, authorizations, interlockings, linked values).
Events (on page 158)	Configuration and display of events (functions).
Transformations (on page 160)	Configuration and display of transformations.
Name	Name of the property.
connection	Selection of link.
Link type	Type of link (variable, authorization, function)
WPF info	Shows the current value for properties in WPF content. For the user, it is directly visible what type of property it is (Boolean, string, etc.).
Linked	Shows if a property is currently being used. Not contained by default in the view, but can be selected using *Context menu->Column selection*.

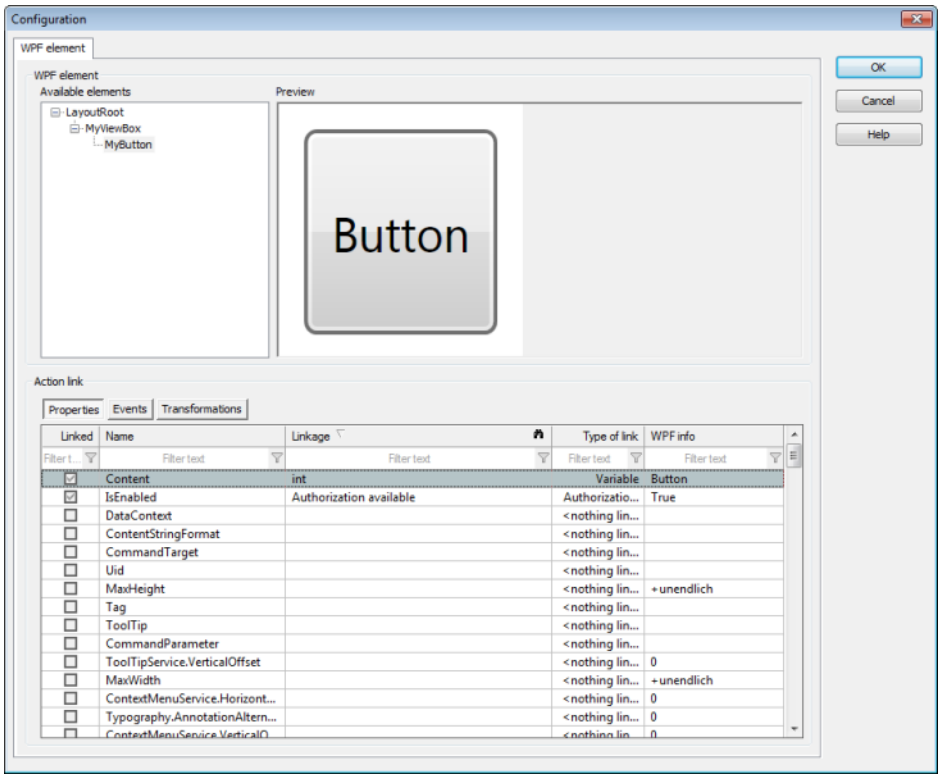
Info

Only logical objects can be displayed in the configuration dialog. Visual objects are not displayed. You can read about backgrounds and how visual objects can be animated in the Allocation of zenon object to WPF content (on page 131).

Properties

The properties enable the linking of:

- Variables (on page 153)
- Values (on page 154)
- Authorizations and interlockings (on page 156)



Parameters	Description
Name	Name of the property.
connection	Linked variable, authorization or linked value. Clicking in the column opens the respective selection dialog, depending on the entry in the **Link type** column.
Link type	Selection of linking.
WPF info	Shows the current value for properties in WPF content. For the user, it is directly visible what type of property it is (Boolean, string, etc.).
Linked	Shows if a property is currently being used. Not contained by default in the view, but can be selected using *Context menu->Column selection*.

CREATE LINK

To create a link:

1. Highlight the line with the property that is to be linked
2. Click in the **Link type cell**
3. select the desired link from the drop-down list.

Available are:

- <not linked> (deletes an existing link)
- Authorization/interlocking
- Variable
- Value linking

4. Click in the **Link** cell
5. The dialog for configuring the desired link opens

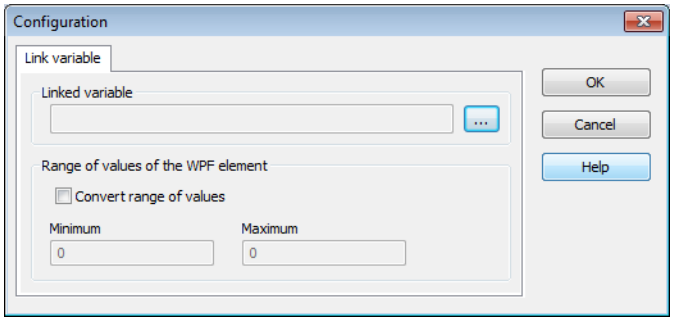
> **Info**
>
> *Properties of WPF and zenon can be different. If, for example the `visibility` property is linked, there are three values available in .NET:*
>
> *0 - visible*
>
> *1 - invisible*
>
> *2- collapsed*
>
> *These values must be displayed via the linked zenon variable.*

Link variable

To link a variable with a WPF property:

1. Highlight the line with the property that is to be linked
2. Click in the **`Link type cell`**
3. Select from the **`Variable`** drop down list
4. Click in the **`Link`** cell
5. The dialog for configuring the variables opens

This dialog also applies for the selection of variables with transformations (on page 160). The configuration also makes it possible to convert from zenon into WPF units.



Parameters	Description
Linked variables	Selection of the variable to be linked. A click on the ... button opens the selection dialog.
Value range of WPF element	Data to convert variable values into WPF values.
Convert value range	Active: WPF unit conversion is switched on. **Effect on Runtime:** The current zenon value (incl. zenon unit) is converted to the WPF range using standardized minimum and maximum values. **For example:** The value of a variable varies from 100 to 200. With the variables, the standardized range is set to 100 - 200. The aim is to display this change in value using a WPF rotary knob. For this: ▸ for **Transformations**, the RotateTransform.Angle property is linked to the variables ▸ Adjust value activated ▸ a WPF value range of 0 to 360 is configured Now the rotary knob can be turned at a value of 150, for example, by 180 degrees.
Minimum	Defines the lowest WPF value.
Maximum	Defines the highest WPF value.
OK	Accepts settings and ends the dialog.
Cancel	Discards settings and ends the dialog.
Help	Opens online help.

Link values

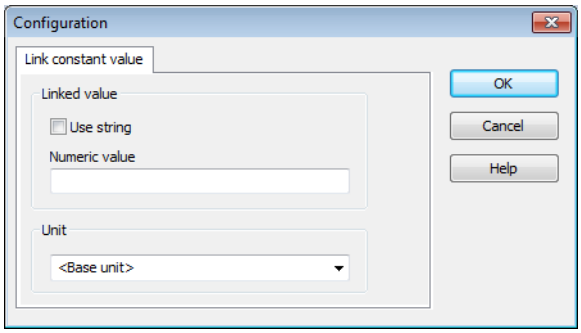
Linked values can either be a **string** or a numerical value of the **double** type. When selecting the screen, the selected value is sent in WPF content after loading the WPF content.

Attention

The data type of the WPF property need not necessarily be **double** or **string**. However only values of the string type or double are sent by zenon. These must be converted to .NET on the WPF page. For details see the Value transfer from zenon to WPF (on page 129) chapter.

To link a value with a WPF property:

1. Highlight the line with the property that is to be linked
2. Click in the **Link type cell**
3. Select from the **value links** drop down list
4. Click in the **Link** cell
5. The dialog for configuration of value linking opens



Parameters	Description
Linked value:	Entry of a numerical value or string value.
Use string	Active: A string value is used instead of a numerical value. The language of string values can be switched. The text is translated in Runtime when the screen is called up and sent in WPF content. If the language is switched whilst the screen is opened, the string value is retranslated and sent.
String value/numerical value	Depending on what is selected for the Use string property, a numerical value or a string value is entered into this field. For numerical values, a unit can also be selected.
Unit:	Selection of a unit from the drop down list. You must have configured this in unit switching beforehand. The unit is allocated with the numerical value. If the units are switched in Runtime, the value is converted to the new unit and sent to WPF content.
OK	Accepts settings and ends the dialog.
Cancel	Discards settings and ends the dialog.
Help	Opens online help.

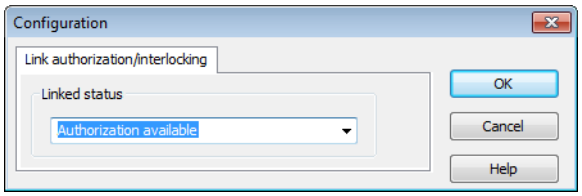
Link authorization or interlocking

Authorizations cannot be granted for the whole WPF element. The element is allocated a user level. Authorizations are granted within the user level for individual controls. If an authorization is active, the value 1 is written to the element.

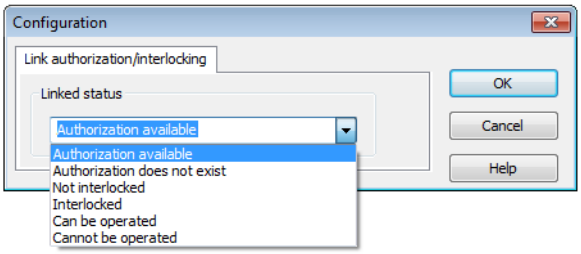
To link an authorization or interlocking with a WPF property:

1. Highlight the line with the property that is to be linked
2. Click in the **Link type cell**
3. Select **Authorization/interlocking** from the drop down menu
4. Click in the **Link** cell

5. The dialog for configuring the authorizations opens



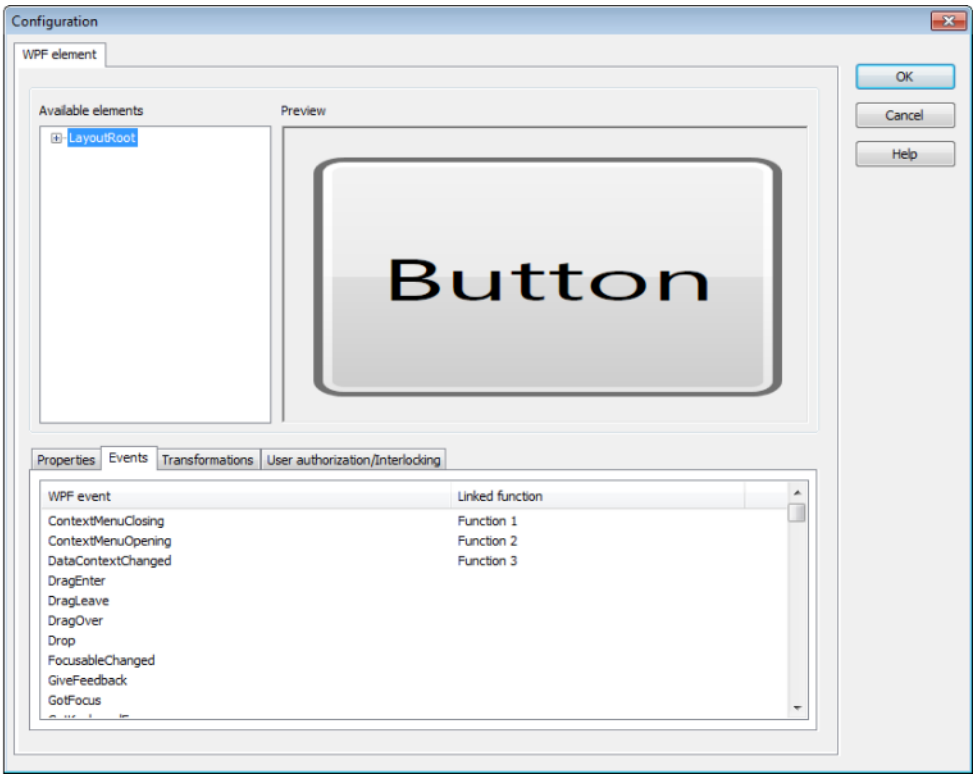
Parameters	Description
Link authorization/interlocking	Setting the authorizations.
Linked status	selection of an authorization that is linked to a WPF control from the drop down list. For example, visibility and operability of a WPF button can depend on a user's status.



Permission	Description
Authorization present	If the user has sufficient rights to operate the WPF element, a value of 1 is written to the property.
Authorization not present	If the user does not have sufficient rights to operate the WPF element, a value of 1 is written to the property.
Not locked	If the element is not locked, the value 1 is written to the property.
Locked	If the element is locked, the value 1 is written to the property.
Operable	If authorization is present and the element is not locked, then a value of 1 is written to the property.
Not operable	If authorization is not present or the element is not locked, then a value of 1 is written to the property.

Events

Events make it possible to link zenon functions to a WPF element.

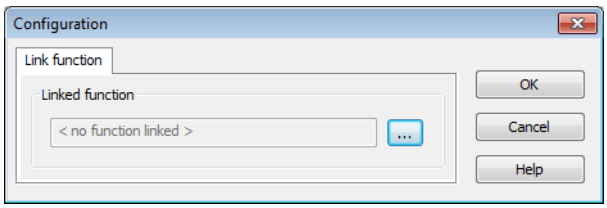


Parameters	Description
Name	Name of the property.
connection	Linked function. Clicking in the cell opens the configuration dialog .
Link type	Selection of linking. Clicking in the cell opens the selection dialog.
WPF info	Shows the current value for properties in WPF content. For the user, it is directly visible what type of property it is (Boolean, string, etc.).
Linked	Shows if a property is currently being used. Not contained by default in the view, but can be selected using *Context menu->Column selection*.

LINK FUNCTIONS

To create a link:

1. Highlight the line with the property that is to be linked
2. Click in the **Link type cell**
3. Select from the drop down list function
4. Click in the **Link** cell
5. The dialog for configuring the function opens

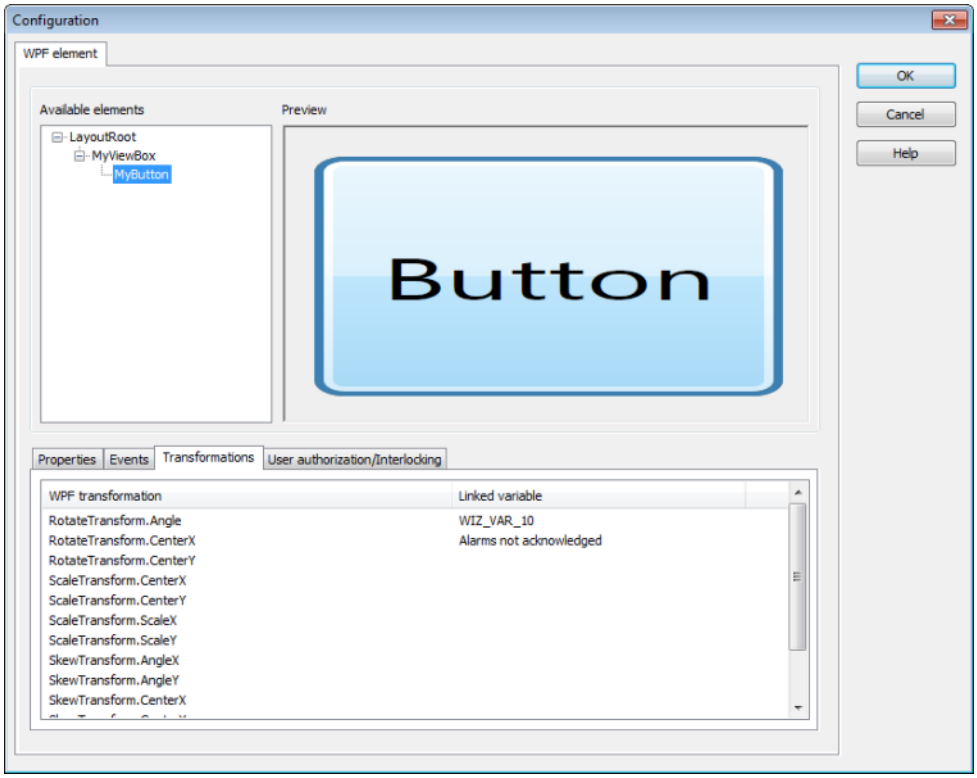


Parameters	Description
Linked function	Selection of the function to be linked. Clicking on the ... button opens the dialog for Function selection (on page 237).
OK	Accepts selection and closes dialog.
Cancel	Discards changes and closes dialog.
Help	Opens online help.

Transformation

The WPF element does not support rotation. If, for example, the WPF element is in a symbol and the symbol is rotated, the WPF element does not rotate with it. Therefore there is a different mechanism for **Transformation** with WPF to turn elements or to otherwise transform them. These transformations are configured in the **Transformation** tab.

Attention: If the content is outside of the WPF element area, this part of the contents is lost, i.e. it is not shown.



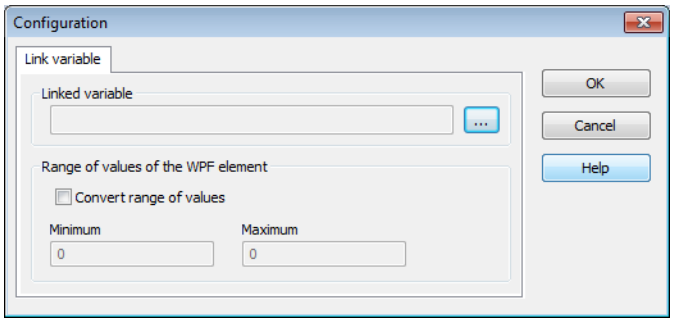
Parameters	Description
Name	Name of the property.
connection	Selection of the linked variables. Transformations are displayed in XAML as transformation objects with their own properties. If an element supports a transformation, then the possible properties of the transformation object are displayed in list view. (more on this in: Integrate button as WPF-XAML in zenon (on page 177) For example, if the linked variable is set at the value of `10`, then this value is written as a WPF target and the WPF element is rotated by 10°.
Link type	Selection of transformation link type.
WPF info	Shows the current value for properties in WPF content. For the user, it is directly visible what type of property it is (Boolean, string, etc.).
Linked	Shows if a property is currently being used. Not contained by default in the view, but can be selected using *Context menu->Column selection*.

LINK TRANSFORMATIONS

To link a transformation with a WPF property:

1. Highlight the line with the property that is to be linked
2. Click in the **Link type cell**
3. Select from the **Transformation** drop down list
4. Click in the **Link** cell
5. The dialog for configuring the variables opens

The configuration also makes it possible to convert from zenon into WPF units.



Parameters	Description
Linked variables	Selection of the variable to be linked. A click on the ... button opens the selection dialog.
Value range of WPF element	Data to convert variable values into WPF values.
Convert value range	Active: WPF unit conversion is switched on. **Effect on Runtime:** The current zenon value (incl. zenon unit) is converted to the WPF range using standardized minimum and maximum values. **For example:** The value of a variable varies from 100 to 200. With the variables, the standardized range is set to 100 - 200. The aim is to display this change in value using a WPF rotary knob. For this: ▸ for **Transformations**, the RotateTransform.Angle property is linked to the variables ▸ Adjust value activated ▸ a WPF value range of 0 to 360 is configured Now the rotary knob can be turned at a value of 150, for example, by 180 degrees.
Minimum	Defines the lowest WPF value.
Maximum	Defines the highest WPF value.
OK	Accepts settings and ends the dialog.
Cancel	Discards settings and ends the dialog.
Help	Opens online help.

Validity of XAML Files

XAML files are valid subject to certain requirements:

- correct name space
- no class references

- Scalability

CORRECT NAME SPACE

The `WPF element` can only display WPF content, i.e.:

Only XAML files with the correct WPF namespace can be displayed by the `WPF element`. Files that use a Silverlight namespace cannot be loaded or displayed. However, in most cases it is suffice to change the Silverlight namespace to the WPF namespace.

WPF namespace:

```
xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
```

NO USE OF CLASS REFERENCES

Because the XAML files can be loaded dynamically, it is not possible to use XAML files that contain references to classes ("class" key in header). Functions that have been programmed in independently-created C#- files cannot be used.

SCALABILITY

If the content of a `WPF element` is adjusted to the size of the `WPF element`, then the controls of the `WPF element` are interlaced in a control that offers this functionality, such as a **`view box`** for example. n addition, care must be taken to ensure that the **`height`** and **`width`** elements are configured as **`automatic`**.

CHECKING AN XAML FILE TO SEE IF IT IS CORRECT

To check if an XAML file has the correct format:

- Open the XAML file in Internet Explorer
  - If it can be opened without additional plug-ins (Java or similar), then it can be assumed with a high degree of certainty that this file can be loaded and displayed by zenon
  - if problems occur during loading, these are then shown in Internet Explorer and the lines in which problems arise can be clearly seen

The scaling can also be tested in this manner: If the file has been created correctly, the content will adjust to the size of the Internet Explorer window.

ERROR MESSAGE

If an invalid file is used in zenon, then an error message is displayed in the output window when loading the file in the WPF element.

For example:

"error when loading `xaml-Datei:C:\ProgramData\COPA-DATA\SQL\781b1352-59d0-437e-a173-08563c3142e9\FILES\zenon\custom\media\UserControl1.xaml`

`The attribute "Class" cannot be found in XML namespace "http://schemas.microsoft.com/winfx/2006/xaml". Line 7 Position 2."`

Pre-built elements

zenon is already shipped with several WPF elements. More are available for download in the web shop.

All WPF elements have properties which determine the graphical design of the respective element (Dependency Properties). Setting the values via an XAML file or linking the property via zenon can directly change the look in the Runtime. The following tables contain the respective Dependency Properties, depending on the control.

Elements:

- Round display (on page 166)
- Progress bar (on page 171)
- Vertical bar graph (on page 172)
- Temperature control (on page 173)
- Analog clock (on page 174)
- Universal slider (on page 175)

Circular gauge control

Property	Function	Value
CurrentValue	Current value which should be displayed.	Double
IsReversed	Scale orientation - clockwise or anti-clockwise	Boolean
ElementFontFamily	Element font.	Font
MinValue	Minimum value of the scale.	Double
MaxValue	Maximum value of the scale.	Double
ScaleRadius	Radius of the scale.	Double
ScaleStartAngle	Angle at which the scale starts.	Double
ScaleLabelRotationMode	Alignment of the scale caption.	Enum: ▸ None ▸ Automatic ▸ SurroundIn ▸ SurroundOut
ScaleSweepAngle	Angel area which defines the size of the scale.	Double
ScaleLabelFontSize	Font size of the scale caption.	Double
ScaleLabelColor	Font color of the scale caption.	Color
ScaleLabelRadius	Radius on which the scale caption is orientated.	Double
ScaleValuePrecision	Accuracy of the scale caption.	Integer
PointerStyle	Shape of the pointer displaying the value.	Enum: ▸ Arrow ▸ Rectangl

		e ▸ Triangle Cap ▸ Pentagon ▸ Triangle

MajorTickColor	Color of main ticks on the scale.	Color
MinorTickColor	Color of sub ticks on the scale.	Color
MajorTickSize	Size of main ticks on the scale.	Size
MinorTickSize	Size of sub ticks on the scale.	Size
MajorTicksCount	Number of main ticks on the scale.	Integer
MajorTicksShape	Shape/type of main ticks on the scale.	Enum: ▸ Rectangle ▸ Trapezoid ▸ Triangle
MinorTicksShape	Shape/type of sub ticks on the scale.	Enum: ▸ Rectangle ▸ Trapezoid ▸ Triangle
MinorTicksCount	Number of sub ticks on the scale.	Integer
PointerSize	Size of the pointer.	Size
PointerCapRadius	Size of the pointer fastening point.	Double
PointerBorderBrush	Color of pointer border.	Brush
PointerCapStyle	Shape/type of pointer fastening point.	Enum: ▸ BackCap ▸ FrontCap ▸ Screw
PointerCapBorderBrush	Color of pointer fastening point.	Brush

PointerBrush	Color of pointer.	Brush
GaugeBorderBrush	Color of the element border.	Brush
GaugeBackgroundBrush	Color of element background.	Brush
PointerCapColorBrush	Color of pointer fastening point.	Brush
GaugeMiddlePlate	Radius of the element background middle plate.	Double
PointerOffset	Offset of the pointer (displacement).	Double
RangeRadius	Radius of the total range display.	Double
RangeThickness	Thickness of the total range display.	Double
RangeStartValue	Start value of the total range display.	Double
RangeEndValue	End value of the total range display.	Double
RangeMinEndValue	End value of the minimum range.	Double
RangeMidEndValue	End value of the middle range.	Double
RangeMinColorBrush	Color of the minimum range.	Brush
RangeMidColorBrush	Color of the middle range.	Brush
RangeMaxColorBrush	Color of the maximum range.	Brush
ScaleOuterBorderBrush	Color of the scale border.	Brush
ScaleBackgroundBrush	Color of scale background.	Brush
ValueTextFrameStyle	Shape/type of value display.	Enum: ▸ LargeFrame ▸ SmallFrame ▸ None
ValueTextContent	Content of the value display.	Enum:

		▸ Text ▸ TextValue ▸ Value
ValueTextSize	Font size of the value display.	Double
ValueTextColor	Font size of the value display.	Color
IsGlasReflection	Activate the glass effect on the element.	Boolean
GaugeOffsett	Lowering the rotation point of the whole element.	Double

Progress bar - ProgressBarControl

Property	Function	Value
CurrentValue	Current value which should be displayed.	Double
MinValue	Minimum value of the value area.	Double
MaxValue	Maximum value of the value area.	Double
ProgressbarDivisionCount	Number of divisions of the progress bar.	Integer
VisibilityText	Visibility of the value display.	Boolean
TextSize	Font size of the value display.	Double
TextColor	Color of the value display.	Color
ProgressBarBoxedColor	Color of the border of the progress bar.	Color
ProgressBarMarginDistance	Distance of the progress bar box from the element edge (left, top, right, down).	Double
ProgressBarInactiveBrush	Indicator color not active.	Brush
ProgressBarActiveBrush	Indicator color active.	Brush
ProgressBarPadding	Distance of the progress bar from the progress bar box (left, top, right, down).	Double
ElementBorderBrush	Color of the element border.	Brush
ElementBackgroundBrush	Color of element background.	Brush

Bar graph vertical - VerticalBargraphControl

Property	Function	Value
CurrentValue	Current value which should be displayed.	Double
MinValue	Minimum value of the scale.	Double
MaxValue	Maximum value of the scale.	Double
MajorTicksCount	Number of main ticks on the scale.	Integer
MinorTicksCount	Number of sub ticks on the scale.	Integer
MajorTickColor	Color of main ticks on the scale.	Color
MinorTickColor	Color of sub ticks on the scale.	Color
ElementBorderBrush	Color of the element border.	Brush
ElementBackgroundBrush	Color of element background.	Brush
ElementGlassReflection	Activate the glass effect on the element.	Visibility
ElementFontFamily	Element font.	Font
ScaleFontSize	Font size of the scale.	Double
ScaleFontColor	Font color of the scale.	Color
IndicatorBrush	Bar graph filling color.	Brush
BargraphSeparation	Number of bar graph dividion.	Integer
BargraphSeparationColor	Color of the scale division.	Color

Temperature indicator - TemperatureIndicatorControl

Property	Function	Value
CurrentValue	Current value which should be displayed.	Double
MinValue	Minimum value of the scale.	Double
MaxValue	Maximum value of the scale.	Double
MajorTicksCount	Number of main ticks on the scale.	Integer
MinorTicksCount	Number of sub ticks on the scale.	Integer
TickNegativColor	Color of the negative main tick (gradient to TickPositivColor).	Color
TickPositivColor	Color of the positive main tick (gradient to TickNegativColor).	Color
MinorTickColor	Color of the sub ticks.	Color
ElementBorderBrush	Color of the element border.	Brush
ElementBackgroundBrush	Color of element background.	Brush
ElementGlassReflection	Activate the glass effect on the element.	Visibility
ElementFontFamily	Element font.	Font
IndicatorColor	Color of the indicator filling color.	Color
IndicatorBorderColor	Color of the indicator border.	Color
MajorTickSize	Size of main ticks on the scale.	Size
MinorTickSize	Size of sub ticks on the scale.	Size
ScaleLetteringDistance	Distance of the scale caption (vertical), each x. main tick should be captioned.	Integer
IndicatorScaleDistance	Distance between indicator and scale (horizontal).	Double
ScaleFontSize	Font size of the scale.	Double
ScaleFontColor	Font color of the scale.	Color
Unit	Unit.	String

ElementStyle	Shape/type of element.	Enum: ▸ SmallFrame ▸ Unit ▸ None

Analog clock - AnalogClockControl

Property	Function	Value
ElementStyle	Shape/type of element.	Enum: ▸ SmallNumbers ▸ BigNumbers ▸ No
ElementBackgroundBrush	Color of element background.	Brush
ElementGlassReflection	Activate the glass effect on the element.	Visibility
Offset	Value in hours (h) which displays the time lag to the system clock.	Int16
OriginText	Text which is displayed in the clock (e.g. location).	String

Universal slider - UniversalReglerControl

Property	Function	Value
CurrentValue	Current value which should be displayed.	Double
ElementFontFamily	Element font.	Font
MinValue	Minimum value of the scale.	Double
MaxValue	Maximum value of the scale.	Double
Radius		Double
ScaleRadius	Radius of the scale.	Double
ScaleStartAngle	Angle at which the scale starts.	Double
ScaleLabelRotationMode	Alignment of the scale caption.	Enum: ▸ None ▸ Automatic ▸ SurroundIn ▸ SurroundOut
ScaleSweepAngle	Angel area which defines the size of the scale.	Double
ScaleLabelFontSize	Font size of the scale caption.	Double
ScaleLabelColor	Font color of the scale caption.	Color
ScaleLabelRadius	Radius on which the scale caption is orientated.	Double
ScaleValuePrecision	Accuracy of the scale caption.	Integer
ElementStyle	Display type of the element	Enum: ▸ Knob ▸ Plate ▸ None
MajorTickColor	Color of main ticks on the scale.	Color
MinorTickColor	Color of sub ticks on the scale.	Color

MajorTickSize	Size of main ticks on the scale.	Size
MinorTickSize	Size of sub ticks on the scale.	Size
MajorTicksCount	Number of main ticks on the scale.	Integer
MajorTicksShape	Shape/type of main ticks on the scale.	Enum: ▸ Rectangle ▸ Trapezoid ▸ Triangle
MinorTicksShape	Shape/type of sub ticks on the scale.	Enum: ▸ Rectangle ▸ Trapezoid ▸ Triangle
MinorTicksCount	Number of sub ticks on the scale.	Integer
BackgroundBorderBrush	Color of the element border.	Brush

BackgroundBrush	Color of element background.	Brush
PointerCapColorBrush	Color of pointer fastening point.	Brush
GaugeMiddlePlate	Radius of the element background middle plate.	Double
ValueFontSize	Font size of the value display.	Double
ValueFontColor	Font size of the value display.	Color
IsGlasReflection	Activate the glass effect on the element.	Boolean
KnobBrush	Color of the knob.	Brush
IndicatorBrush	Color of the indicator.	Brush
IndicatorBackgroundBrush	Background color of the inactive indicator.	Brush
KnobSize	Diameter of the knob.	Double
KnobIndicatorSize	Indicator size of the knob.	Size
ElementSize	Size of the element.	Size
VisibilityKnob	Activating of the knob.	Boolean
ValuePosition	Position of the value display.	Double
ValueVisibility	Activating the value display.	Boolean

Examples: Integration of WPF in zenon

You can see how XAML files are created and integrated as WPF elements in zenon from the following examples:

- Integrate button as WPF-XAML in zenon (on page 177)
- Integrate bar graph as WPF XAML in zenon (on page 185)

Integrate button as WPF XAML in zenon

Example structure:

- Creating a button (on page 134) in Microsoft Expression Blend
- Integrate into zenon
- Link to a variable and a function
- adjust the button to the size of the element
- Create button

As a first step, create a button as described in the Create button as XAML file with Microsoft Expression Blend (on page 134) chapter. To be able to use the XAML file in zenon, insert this in the project tree in the **Files/graphics** folder.

INTEGRATE BUTTON

Note: A zenon project with the following content is used for the following description:

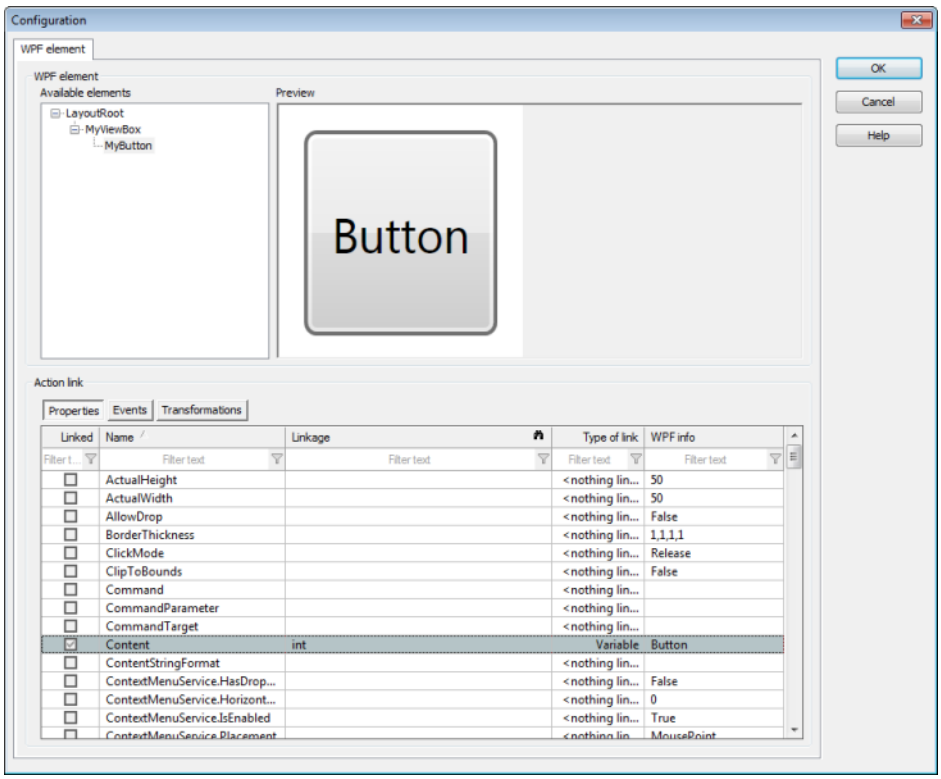
- An empty screen as a start screen
- an internal variable **int** of type **Int**
- a function **Function_0** of type **Send value to hardware** with:
  - **Direct to hardware** option activated
  - Set was set to 45

To integrate the button:

1. open the empty screen
2. place a WPF element in the screen
3. select XAML file in the properties window
4. Select the desired XAML file (for example **MyButton.xaml)** and close the dialog
5. select the Configuration property

CONFIGURE THE BUTTON

The configuration dialog shows a preview of the selected XAML file. All elements named in the XAML file are listed in the tree:

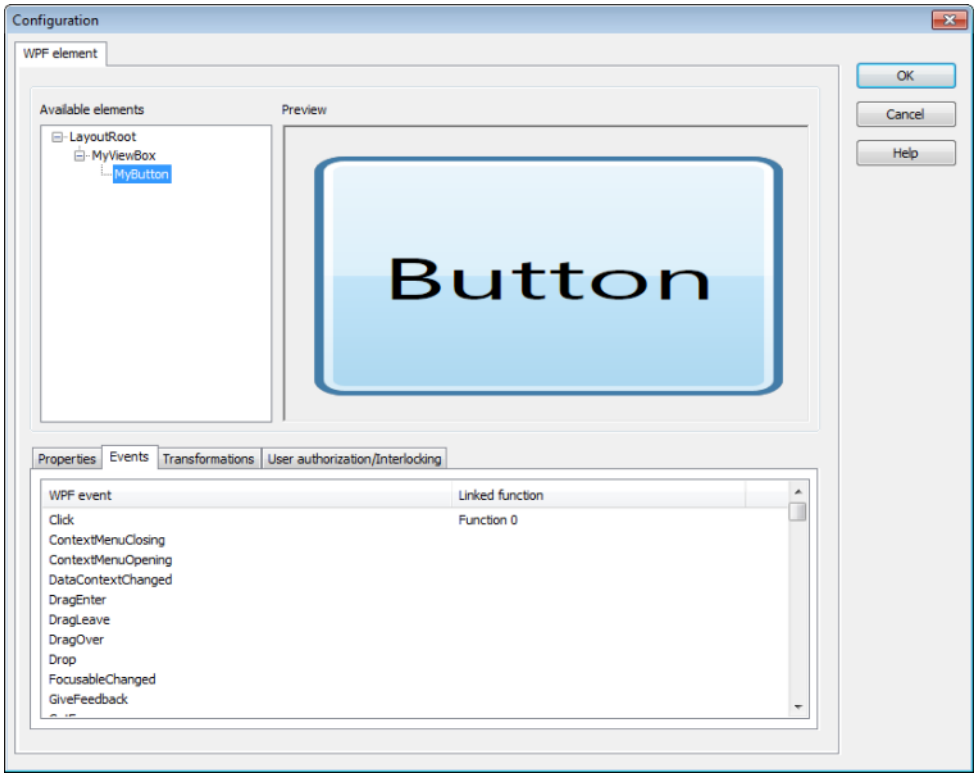


1. select the WPF button, which is in *LayoutRoot->MyViewBox->MyButton*
2. Look in the **`Properties`** Entry**`Content`** tab; this contains the button's text
3. Click the **`Link type`** column
4. Select **`Variable`** from the drop down list
5. Click in the **`Link`** column
6. the variable selection dialog is opened
7. select the `int` variable to link this variable with the **`Content`** property

EVENTS

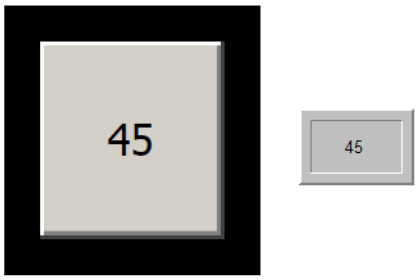
To also assign events:

1. select the events tab



2. look for the 'Click' entry, this event is triggered by the WPF element, as soon as the button is clicked
3. Click in the **`Link type`** column
4. Select **`Function`** from the drop down list
5. Click in the **`Link`** column
6. the **`function selection dialog`** is opened
7. select **`Function_0`**
8. Confirm the changes with **`OK`**
9. Insert a `numerical value element` into the screen
10. Link this `numerical value element` to the `int` variables too.
11. Compile the Runtime files and start Runtime.

The WPF element is displayed in Runtime, the button text is 0. As soon as you click on the button, the **click** event is triggered and the set value function is carried out. The value 45 is sent directly to the hardware and both numerical value and button display the value 45 .

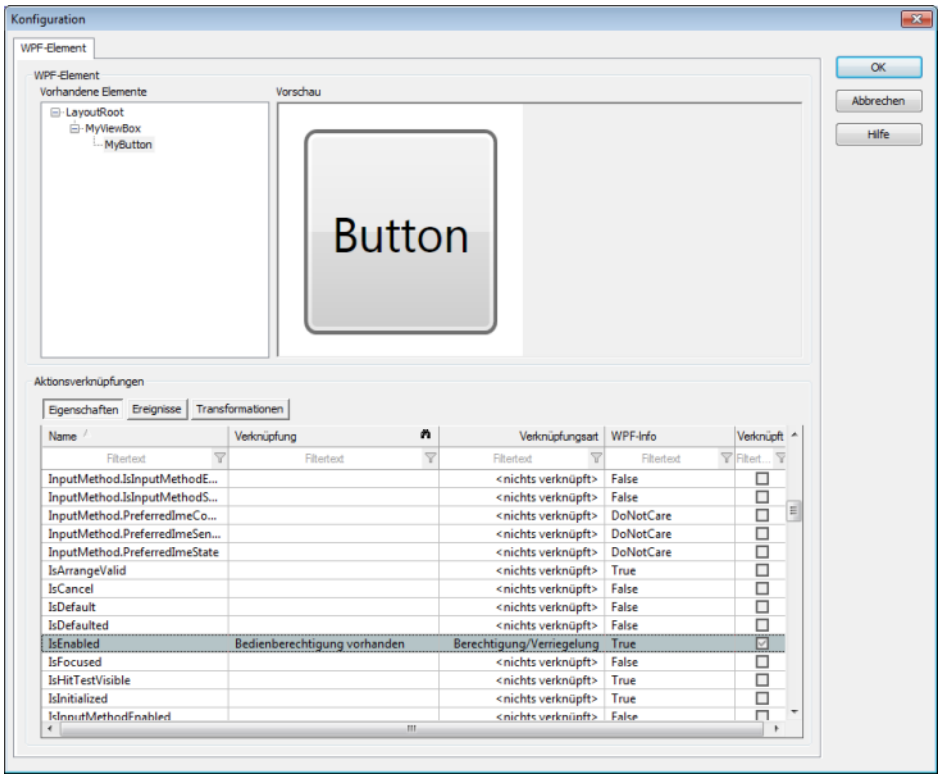


Define a set value of 30 via the numerical value element; this value is then also assumed by the WPF element.

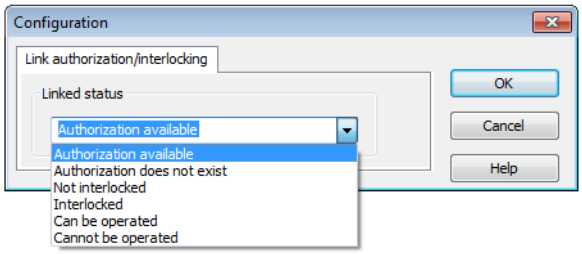
AUTHORIZATION

Similar to a numerical value, a WPF element can be locked according to authorizations (lock symbol) or switched to be operable. Set the user authorization level to 1 for the WPF element and create a user called **Test** withauthorization level **1**. In addition, set up the functions Login with dialog and Logout . You link these two functions with 2 new text buttons on the screen.

In the `WPF element` configuration dialog, select the **`MyButton`** WPF button and select the **`Properties:`** tab

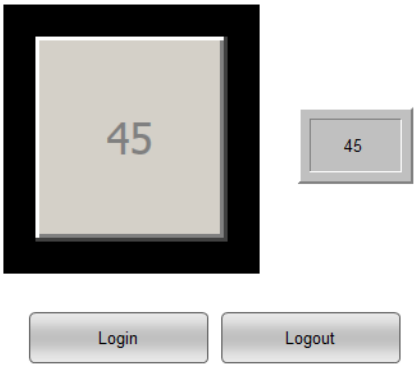


1. Select the **`IsEnabled`** element
2. Click in the **`Link type`** column
3. Select **`Authorizations/interlocking`** from the drop down list
4. Click in the **`Link`** column
5. In the drop-down list, select the `Authorized` option



6. Close the dialog with **`OK`**

Compile the Runtime file and note that Authorizations to be Transferred must also be selected. After Runtime has been started, the WPF button is displayed as deactivated on the screen and cannot be operated. If you now log in as the user **Test**, the button is activated and can be operated. The button is locked again as soon as you log out.



TRANSFORMATION

The XAML files must still be adapted to use transformations:

1. switch to the **Expression Blend** program
2. select **MyButton**, so that the properties of the element are visible in the events window

3. Under **Transform** at **RenderTransform** select the Apply relative transform option

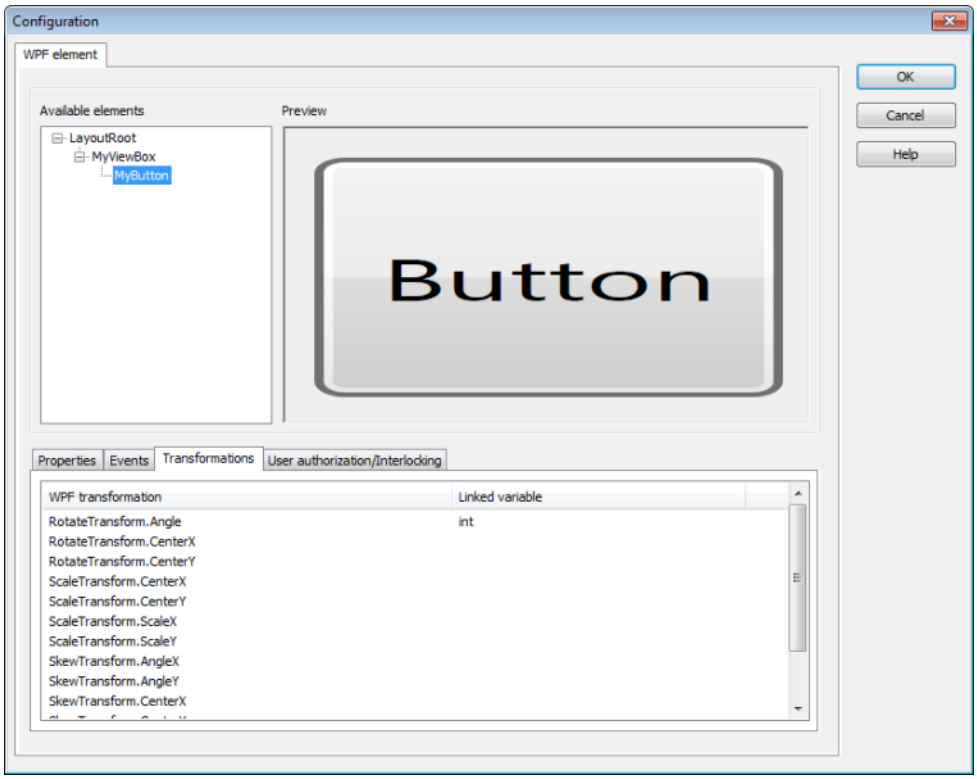
As a result of this, a block is inserted into the XAML file, which save the transformation settings in runtime.

```
<Button.RenderTransform>
    <TransformGroup>
        <ScaleTransform ScaleX="1" ScaleY="1"/>
        <SkewTransform AngleX="0" AngleY="0"/>
        <RotateTransform Angle="0"/>
        <TranslateTransform X="0" Y="0"/>
    </TransformGroup>
</Button.RenderTransform>
```

4. Save the file and replace the old version in zenon with this new file.
5. Open the WPF element configuration dialog again:

a) select the **`MyButton`** button

b) select the **`Transformations`** tab



c) select the **`RotateTransform.Angle`** element

d) Click in the **`Link type`** column

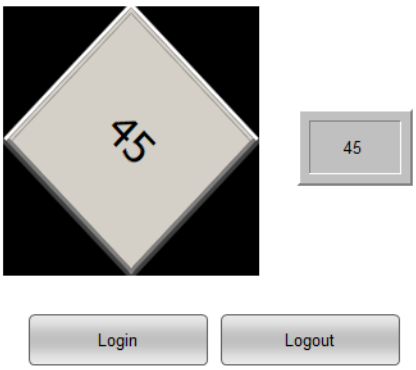
e) Select **`Transformations`** from the drop down list

f) Click in the **`Link`** column

g) the variable selection dialog is opened

h) select the **`int`** variable to link this variable with the **`RotateTransform.Angle`** property

Compile the Runtime files and start Runtime. Log in as the **Test** user and click on the button. The button has the value 45 and the WPF element rotates by 45°.



Integrate bar graph as WPF XAML in zenon

Example structure:

- Creating a bar graph (on page 138) in Adobe Illustrator and converting it to WPF
- Integrate into zenon
- Linking with variables
- Adapting the bar graph WPF element

CREATE BAR GRAPH

The first step is to generate a bar graph as described in the Workflow with Adobe Illustrator (on page 138) chapter. To be able to use the XAML file in zenon, insert this in the project tree in the **Files/graphics** folder.

INTEGRATE BAR GRAPH

Note: A zenon project with the following content is used for the following description:

- An empty screen as a start screen
- Four variables from the internal driver for
  - Scale 0

- Scale `central`
- Scale `high`
- Current value

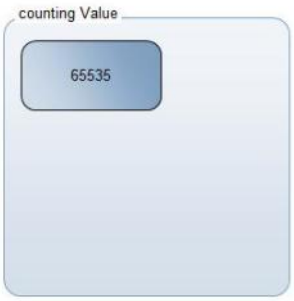
► A variable from the mathematics driver for displaying the current value (`255`)

To integrate the bar graph:

1. open the empty screen
2. place a `WPF element` in the screen
3. select `XAML file` in the properties window
4. Select the desired XAML file (for example **`bar graph_vertical.xaml`**) and close the dialog

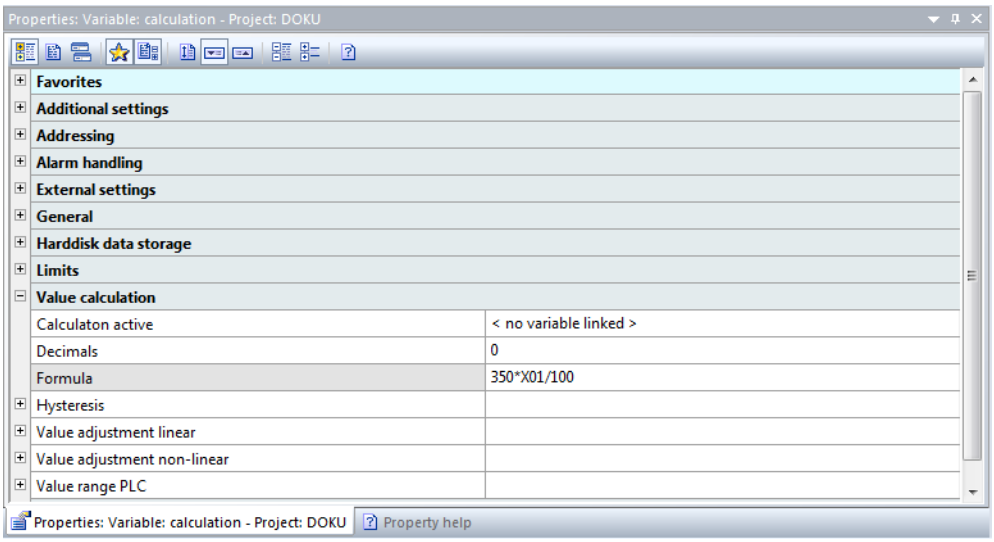
ADJUST BAR GRAPH

Before configuration, the scale of the XAML file is adapted if necessary:



To do this:

- Create a new mathematics variable that calculates the new value in relation to the scaling, for example:
- Variable: 0-1000
- Mathematic variable {value created in xaml file}*Variable/1000

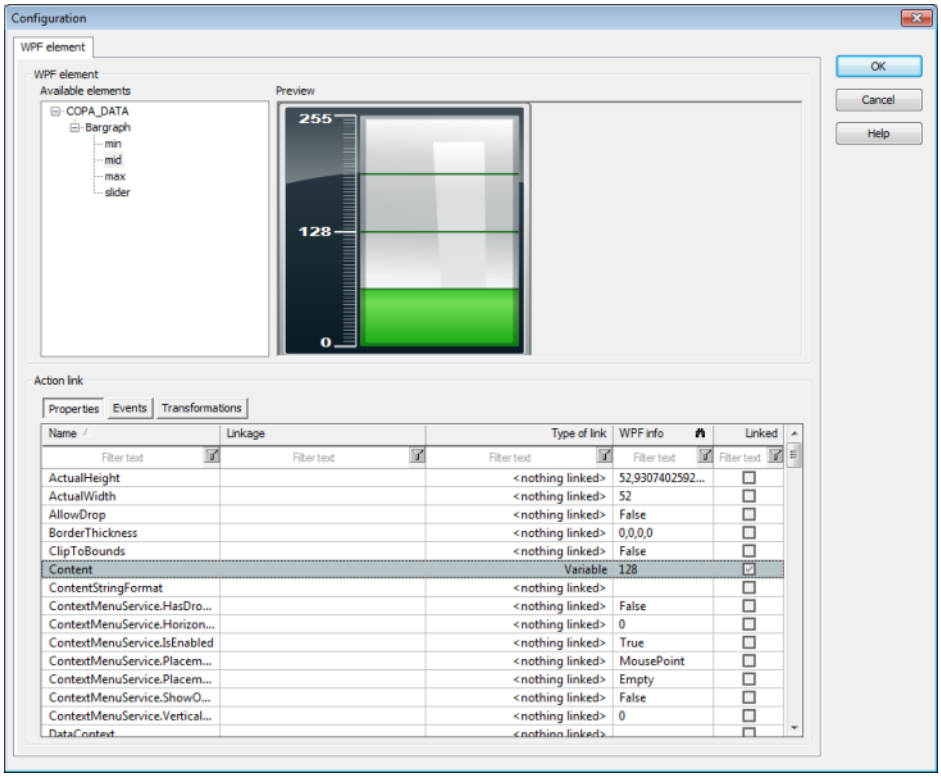


The XAML file is then configured.

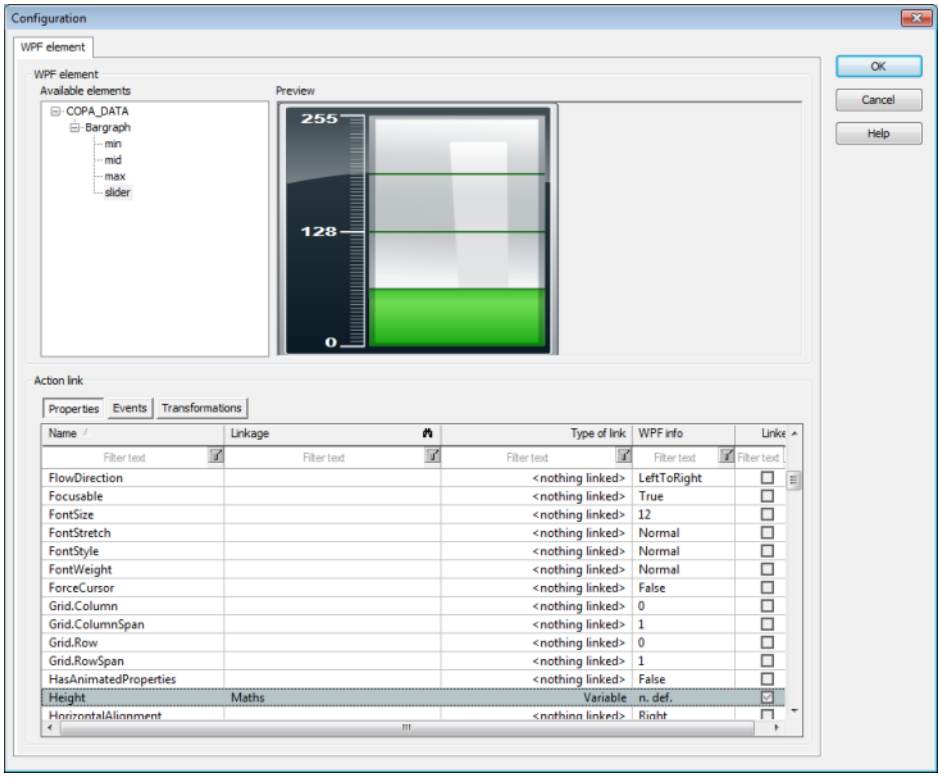
CONFIGURE BAR GRAPH

1. Click on the WPF element and select the `Configuration` property
2. The configuration dialog shows a preview of the selected XAML file.

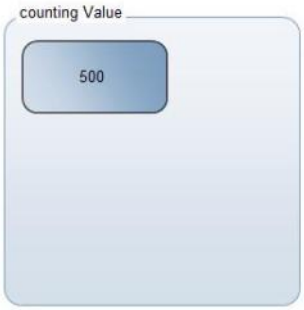
3. Select the minimum value, the average value and the maximum value and link each of these to the corresponding variable in the `Content` property



4. Select the **`Slider`** and link the **`Value`** property to the mathematics variables (in our example: calculation)



5. Check the project planning in Runtime:



Troubleshooting

ENTRIES IN LOG FILES

Entry	Level	Meaning
Xaml file found in %s with different name, using default!	Warning	The name of the collective file and the name of the XAML file contained therein do not correspond. To avoid internal conflicts, the file with the name of the collective file and the suffix **.xaml** is used.
no preview image found in %s	Warning	The collective file does not contain a valid preview graphic (**preview.png** or **[names of the XAML file].png**). Thus no preview can be displayed.
Xaml file in %s not found or not unique!	Error	The collective file does not contain an XAML file or several files with the suffix **.xaml**. It cannot be used.
Could not remove old assembly %s	Warning	There is an assembly that is to be replaced with a newer version, but cannot be deleted.
Could not remove old assembly %s	Error	A new version is available for an assembly in the work folder, but it cannot be copied there. Possible reason: The old example is still loaded, for example. The old version continues to be used, the new version cannot be used,
file exception in %s	Error	A file error occurred when accessing a collective file.
Generic exception in %s	Error	A general error occurred when accessing a collective file.

10.29 Numerical value

You display numerical values with the **numerical values** dynamic element.

To create a **numerical value** in a screen:

1. select the **numerical value** symbol in the **Elements** tool bar or in the **Elements** drop-down list
2. select the start point in the main window
3. pull open the element while pressing and holding the left mouse button

4. the dialog to assign a numerical value (on page 238) opens
5. select the desired variable and define the desired properties in the properties window

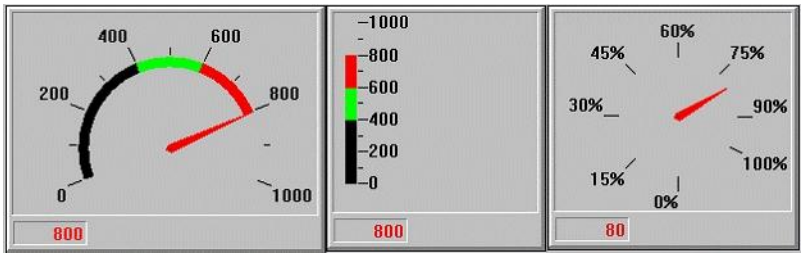
> **Info**
>
> *The following applies in runtime: A decimal value can be entered with a comma as well as with a period as a separator. It is always displayed as a period.*

10.30 Indicating instrument

You display the indicating instrument with the **indicating instrument** dynamic element.

To create an **indicating instrument** in a screen:

1. select the **Indicating Instrument** symbol in the **Elements** tool bar or in the **Elements** drop-down list
2. select the start point in the main window
3. pull open the element while pressing and holding the left mouse button
4. the dialog to assign a numerical value (on page 238) opens
5. select the desired variable and define the desired properties in the properties window



COLOR-CODED DISPLAY OF THE LIMIT VIOLATION

At limit violations the color from the variable can color the whole area or only the area starting from the limit violation. This functionality is controlled via property `Explicit` in group `Representation`.

This property affects only the display of values with defined limits from variables. Limits from reaction matrices are not considered.

`Active`: If a limit of the linked variable is violated, only the part of the display which reaches into the limit violation is displayed in the color of the respective limit.
If the variable has several defined limits, the respective areas of the display are always displayed in the color of the corresponding limit.

`Inactive`: If a limit of the linked variable is violated, the complete display is displayed in the color of the violated limit.

Default: `inactive`

10.31 Status Element

You control a screen symbol with a **`Status Element`** dynamic element. In doing so, properties of a linked variable such as colors or flashing for example are transferred to a symbol. You can therefore also show statistical limit value texts.

To create the **`Status Element`** in a screen:

1. select the **`Status Element`** symbol in the **`Elements`** tool bar or in the **`Elements`** drop-down list
2. select the start point in the main window
3. pull open the element while pressing and holding the left mouse button
4. the dialog to assign a variable (on page 238) opens
5. select the desired variable and define the desired properties in the properties window

> **Info**
>
> *Instead of the dynamic element **`Binary element`** the dynamics properties of the according element can be used.*

11. Edit screen element

The are many possibilities available to you for editing screen elements:

- Graphic actions (on page 194)
- Automated replacement (on page 225)
- Background graphics (on page 236)
- Selection dialog functions (on page 237)
- Variables selection dialog (on page 238)

11.1 Graphic actions

Graphic actions are also possible in zenon. To do this, use:

- Element properties
- Commands in the Edit menu
- Actions with the mouse:
  - Holding down the ctrl key when moving the mouse copies the element.
  - Moving with the right mouse button pressed opens the content menu, with a choice of whether to copy or move the item
- the context menu

> **Info**
>
> *Graphic actions such as shading and blurring can have an effect on the performance of the system, both in the editor and in runtime.*

ENTERING VALUES

To enter values within a defined limit, there are also sliders for many properties. These can be accessed via the . . . button+. Changes in values made with the slider are displayed in the properties field immediately.

When selecting colors, the color activated in the palette is displayed directly in the object as a preview.

11.1.1 Truncations

The elements **Rectangle** (on page 113) and **Button** (on page 43) and Polygon (on page 110), Polyline (on page 112) and Tube (on page 114) can be displayed with truncations.

> **Info**
>
> *Truncations are available only with limitations if property Windows CE project is activated or if property Graphics quality has value Windows Base.*

RECTANGLE AND BUTTON

You define the degree of truncation with the properties in Rounded corners node. 2 values can be given for each corner point:

- X value: Percentage value for the rounding based on the width of the element
- Y value: is a percentage value for the rounding based on the height of the element

0% means no rounding.
100 % means a quadrant.

If the Proportional property is activated, only the X value can be entered.

The values can be defined separately for each angle. If the value of an angle is set at −1, it then takes its rounding value from the All x [%] or All y [%] property.

> **Info**
>
> *If the value for All x [%] or All y [%] is changed, all values are set to -1 on the corresponding axis. Therefore different values can be re-harmonized very easily.*

Hint: There is a small yellow triangle in the lower right quadrant of the element. Click on it with the mouse and move the triangle up to the desired rounding.

Example:

There is a GIF file on the button with a transparent background. A red color gradient was defined for the button. The rounding of the angles was defined with `All x [%]` and `Proportional`, afterwards this value was changed individually for the lower left angle.

> **Info**
>
> *If the `With brightness values` property is used for displaying the fill color, no rounded edges can be used. If this element is activated for an element, that uses rounded-off corners, the element is displayed with non-rounded corners.*

POLYGON, POLYLINE UND TUBE

You can define the degree of rounding for polygons, polylines and tubes with the `Rounding` property in the `Representation` node. Enter a value between `0` and `1` for this.

11.1.2 Effects for screen elements

Several graphical elements can be highlighted with the help of effects in the Runtime.

You can find additional effects in chapter effects for frames (on page 254).

Glow effect

Many zenon support the glow effect.

REQUIREMENTS

To use glow effect for a graphical element, the following requirements must be fulfilled:

- The project property Graphical design/Graphics quality must be set to DirectX Hardware or DirectX Software. For details see chapter Graphics quality (on page 12).
- The graphical element must be supported.
  Except for the following exceptions all graphical elements are supported: WMF, symbols, combi element, ActiveX, WPF, all Win32 control elements, windows and several special control elements.
- The element must be visible.
  If the element is invisible, the glow effect is invisible automatically as it is derived directly from the element.

CONFIGURATION

To configure the glow effect for a supported element:

1. make sure that property Graphics quality is set to DirectX Hardware or DirectX Software
2. create the element
3. go to property node Effects
4. activate the Activate property in the Glow (only DirectX) node
5. configure transparency, color, spread and visibility

If you configure several properties for the same effect, the stronger is executed in the Runtime. For details, see **Dependance of the configuration** section. (For example: An invisible glow effect is always invisible; the settings for flashing has no effect anymore.)

Default value for the glow effect:

- Activate: inactive
- Only active if clicked: inactive
- Transparency [%]: 20 %
- Spread [pixel]: 15 pixel
- Coloring: inactive
- Color: #FFFF90 (yellow)

- Variable for color: none
- Variable for visibility: none
- Take over visibility from limit: inactive
- Visible from: 0
- Visible to: 0
- Variable for flashing: none
- Take over flash color from limit: inactive
- Make 2. flash state invisible: inactive

DEPENDENCIES OF THE CONFIGURATION

VARIABLE FOR VISIBILITY

Visibility has especially strong effects. If the glow effect is set to invisible, flashing and color is ignored. Dependencies:

1. Variable for visibility is linked and Take over visibility from limit is active:
   - Invisible in the limit is activated: Glow effect is invisible.
   - Invisible in the limit is not activated:
     The glow effect configured in the properties is displayed. Dependent of the settings the glow effect can also flash.
2. Variable for visibility is linked and Take over visibility from limit is not active:
   - The variable value lies within the limits of properties of Visible from and Visible to:
     The glow effect configured in the properties is displayed. Dependent of the settings the glow effect can also flash.
   - The variable value lies outside the limits of properties of Visible from and Visible to:
     Glow effect is invisible

3. `Variable for visibility` is not linked:
   The glow effect configured in the properties is displayed. Dependent of the settings the glow effect can also flash.

VARIABLE FOR FLASHING

For all states for which the originally configured glow effect is displayed, you get the glow effect which is configured in the properties. At this for example the color for a limit can come from an own variable; for details see **`Color from variable`**.

1. `Variable for flashing` is linked, `Take over flash color from limit` is active and `Make 2. flash state invisible` is active:
   - Limit was violated:
     Glow effect flashes and alternates between colored glow effect (from limit violation) and the originally configured glow effect.
   - Limit was not violated:
     Glow effect does not flash and the glow effect is displayed as it was originally configured.
2. `Variable for flashing` is linked, `Take over flash color from limit` is active and `Make 2. flash state invisible` is not active:
   - Limit violation is active:
     Glow effect flashes and alternates between colored glow effect (from limit violation) and no glow effect.
   - Limit violation is not active:
     Glow effect does not flash and the glow effect is displayed as it was originally configured.
3. `Variable for flashing` is linked and `Take over flash color from limit` is not active:
   - Limit violation is active:
     Glow effect flashes and alternates between originally configured glow effect and no glow effect.
   - Limit violation is not active:
     Glow effect does not flash and the glow effect is displayed as it was originally configured.

4. `Variable for flashing` is not:
   Glow effect can be displayed dependent on other properties.

COLOR FROM VARIABLE:

Represents the possibility to define the color of the glow effect at normal view via the limit of a variable. This is also true for the normal view when the glow effect does not flash.

1. `Variable for color` linked:
   a) Limit violation is active:
      The color from the violated limit is used.
   b) Limit violation is not active:
      For coloring the glow effect the defined color from property `Color` is used.
2. `Variable for color` not linked:
   For coloring the glow effect the defined color from property `Color` is used.

COLORING:

Defines whether the glow effect is created as image of the actual element or whether it is colored with a single color.

1. `Coloring` active and `Variable for color` linked:
   a) Limit violation is active:
      For coloring the glow effect the color from the violated limit is used.
   b) Limit violation is not active:
      For coloring the glow effect the defined color from property `Color` is used.
2. `Coloring` active and `Variable for color` not linked:
   For coloring the glow effect the defined color from property `Color` is used.
3. `Coloring` not active: An image of the actual element with all corresponding colors of the element for the glow effect is used.

HINTS FOR CONFIGURATION

To receive good results:

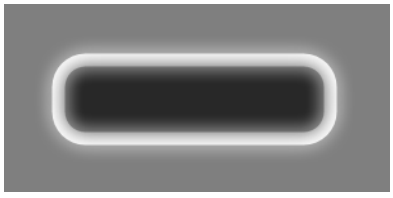
- Select a rather lower value for `Transparency [%]`. Without coloring the value may be a little bit higher than with coloring.
- Do not select a too small value for `Spread [pixel]`. It should be `>5` for the effect to be visible outside of the element.
  Attention: Too high values (`>20`) can lead to the glow effect being reduced very much. In this case the visibility can be increased by reducing the transparency. In general the glow effect needs a lot of performance. Thus you should use it consider it well when and where to use it. A higher value for `Spread [pixel]` costs more performance which especially affects elements which lie close together.
- As for DirectX the glow effect (just like the shadow) is derived from the original element, dynamic changes of the element affect the effect. In addition bitmap graphics can be equipped with this effect. Thus for example a PNG graphic with semi-transparent areas as well as a GIF animation obtain a correct glow effect which is created dynamically during the display. The glow effect just as the shadow also affects text by which correlating effects can be achieved.

Examples for glow effects

GLOW EFFECT

Element without glow effect:

Element with glow effect:



COLORS

Default color 0xFFFF90 (yellow):

Color 0xFFFFFF (white):

TRANSPARENCY

0% transparency:

20% transparency:

60% transparency:

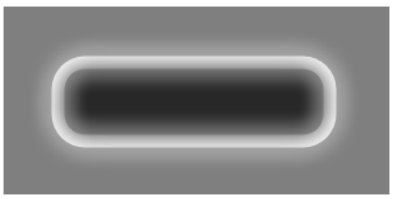
SPREAD

0 pixel spread and 0% transparency:

5 pixel spread and 0% transparency:

10 pixel spread and 0% transparency:

20 pixel spread and 0% transparency:



Color 0xFFFF90, 5 pixel spread and 20% transparency:

Color 0xFFFF90, 20 pixel spread and 20% transparency:

TEXTS

5 pixel spread and 0% transparency:

Text

Color 0xFF0000, 5 pixel spread and 0% transparency:

Text

Large font with 15 pixel spread and 0% transparency:

Text

Large font with color 0xFF0000, 35 pixel spread and 20% transparency:

Text

PART-TRANSPARENT GRAPHIC

Original:

20 pixel spread and 20% transparency:

Color 0xFF0000, 20 pixel spread and 20% transparency:

Without color, 20 pixel spread and 20% transparency plus shadow with 40 pixel distance, 60% transparency, black and 5 pixel soft focus.

ELEMENT CHANGES

3D button in contrast colors, 15 pixel spread and 0% transparency on 3D button:

3D button in contrast colors, pressed with 15 pixel spread and 0% transparency on 3D button:

Troubleshooting

OUTPUT

Entry	Level	Description
DirectX: Screen'Screen name' **- Element '** Element name' **uses an effect which cannot be displayed with the selected graphics setting.**	Warning	At an element the glow effect is activated. However DirectX Softwaree or DirectX Hardware is not selected or not available.

CHECK LIST FOR ERROR LOCALIZING

- Is DirectX turned on?
- Is DirectX available on the system? An according check can carried out by activating the debug messages in the Diagnosis Viewer. For successful support, an appropriate message is displayed.
- Does the element support the glow effect?
- Is the element visible?
- Cause static properties (e.g. transparency) the glow effect not to be displayed or displayed differently?
- Cause dynamic properties the glow effect not to be displayed or displayed differently?

- Is a property selected which causes the glow effect only under certain circumstances (e.g. button element with property "Only active if pressed")?

Frame and Shadows

Elements can be expanded with frames and shadows.

Exception: **`Status elements`** cannot obtain shadows.

> **Info**
>
> *If the transparency of an object (for example the background color) is changed, the display of the preconfigured shading also changes accordingly. The values for colors and transparencies of shading remain the same in the process.*

ALLOCATE FRAME AND SHADOWS

- To give an element a frame, select `Frame` in the property `Border type` node `Borders/Shadows`.
- To give an element a frame, select `Frame` in the property `Border type` node `Borders/Shadows`.

FOR SHADOWS AND FRAMES, YOU CAN DEFINE

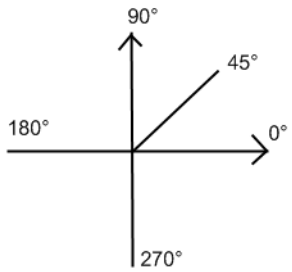
- Distance: distance of the frame or shadow to the element.
- Frame color: Color of frame or shadow.

FOR FRAMES, YOU CAN DEFINE

- Line width: The width of the frame.

FOR SHADOWS, YOU CAN DEFINE

- Angle: The angle in degrees with which the shadow falls, starting from the outside right counterclockwise.



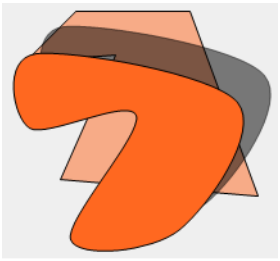
- Transparency: Transparency of the shadow between 0 and 100%, with 0% being opaque and 100% being transparent.
- Size of shadow: Size of shadow in pixels.
- Blurring: Blurs the surface color and makes the transition to shadow softer.
  Note: if a value is irregularly set to 0 here, then the calculation affects system performance.

> **Info**
>
> *Transparency for lines or filling colors of an element have an effect on the shadow. For example: If the filling color is set to 100% transparent, then no shade will be visible for the fill area.*

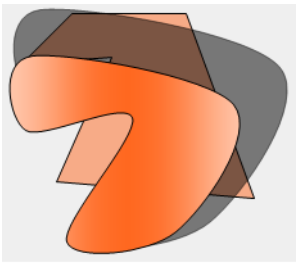
Examples

Shade with an angle of 45 degrees moved by 10 pixels

Shade with an angle of 45 degrees moved by 30 pixels

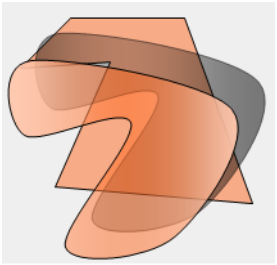


Shade with size of 30 pixels



Shade with size of -20 pixels

Shade with a size of -20 pixels and transparent color gradient in the element



Rounded rectangle with a blur of 10

11.1.3 Move element

To move elements, use:

FROM VERSION 6.50

the properties from the respective element:

- `Position`
- `Size and rotation dynamic`

You can also animate elements with properties

- `Colors dynamic` in the node `Color`
- `Flashing` in the node `Visibility/flashing`
- `Visibility` in the node `Visibility/flashing`

UP TO AND INCLUDING VERSION 6.22 AND RUNTIME COMPATIBILITY

- Alternatively, you can also use the Move Symbol (on page 118) dynamic element

11.1.4 Color gradient and transparency

Elements can be given color gradients or switched to transparent.

To do this, select the corresponding element properties in the `Color` node:

COLOR GRADIENT

Color gradients can be set in the following direction for the following elements:

Element	Linear	Defined point	Define point with beams
Rectangle	yes	yes	yes
Circle	yes	yes	no
Segment of a circle	yes	yes	no
Polygon	yes	yes	yes
Button	yes	yes	yes

To define a color gradient, select `Color Gradient` as a `Filling pattern`.

You can now define the color gradient via the properties of the group `Color gradient`. You can also define the color gradient with the mouse. To do this, click on the yellow circle in the element with (connecting line to the middle point) and move the circle until you reach the desired gradient.

Info

Right angles are positioned horizontally at angles between 45° and 135° and between 225° and 315°.

The `With brightness values` property is not available for polygons.

COLOR GRADIENT UNDER WINDOW CE OR FOR GRAPHICS QUALITY WINDOWS BASIS

Under Windows CE or for settings `Windows Basis` for property `Graphics quality`, color gradients for vector elements can only be accomplished via brightness values. The `Color gradient` property is not available. For the display of color gradients use property:

- `Offset [%]`: only works for element rectangle
- `Angle [°]`: only works for element rectangle
- `Brightness outside`
- `Brightness inside`

Note:

Property `With brightness values` need not extra be activated.

Property Angle [°] can only be changed in steps of 90° under Windows CE and at setting Windows Basis for property Graphics quality.

BUTTON

For element button under Windows CE and Windows Basis the following is also true:

- For property Filling the values Starting from a defined point and Starting from a defined point with beams are not available.
- The Angle [°] property is not available.

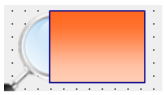
TRANSPARENCY

You can switch an element to be completely transparent with the Transparent property.

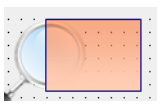
To switch elements to be partially transparent, use the properties for filling color transparency, gradient color and frame color. Here, you define the covering power of the colors with values between 0 and 100 percent. In doing so, 0 is invisible and 100 is opaque. All values between this appear transparent according to how they are set.

EXAMPLES

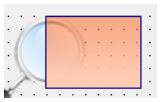
Transparency: filling color = 100, filling color 2 = 100



Transparency: filling color = 50, filling color 2 = 100



Transparency: filling color = 0, filling color 2 = 100



Info

Control elements to control list fields such as screen scroll bars do not support transparency.

11.1.5 Transfer format

To transfer the properties of a selected element to another one / multiple elements:

1. Transfer to an element:
   a) Click on element with original properties -
   b) Click on the **Transfer properties** symbol
      or the corresponding command in the context menu
   c) Click on target element: Properties are transferred
2. Transfer to multiple elements:
   a) Select source element -
   b) Select target element with Ctrl key held down (source and target elements are highlighted)
   c) Click on the **Transfer properties** symbol or the corresponding command in the context menu
   d) The source element´s properties are transferred to the target elements.

If several elements were chosen as source, the properties of the first selected element are transferred.

Info

Accept properties via properties window:

If several elements are highlighted, the properties of the last respective highlighted element in the properties input is displayed. If the properties window is displayed in the **Grouped view** mode or in the **All properties mode** mode, the respective property can be transferred to all highlighted elements by pressing the **Return** key. This is not possible in the **dialog view** mode. For details, see the Properties window chapter, **Multiselect** section.

RULES FOR TRANSFERRING FORMATS

TRANSFERRED ARE:

- Properties that directly define an element in form, color, and appearance and are not influenced by a variable.

NOT TRANSFERRED ARE:

- Deactivated properties
- Direct properties that it does not make sense to transfer, such as texts and graphics files.

  Exception:
  - Graphics files for a switch are copied as you normally always want to have the same switches.
  - Graphics files for buttons are not copied as different ones are normally used.
- Properties from groups:
  - Authorization
  - Automatic Line Coloring
  - General
  - Position
  - Size and rotation dynamic
  - Runtime
  - VBA
  - Write set value

- Everything that comes directly from variables
- Variables and linked functions

11.1.6 Copy, Insert and Move

To copy, move or delete elements on a screen,use

- Commands in the **Edit** menu
- Actions with the mouse:

  Holding down the **ctrl** key when moving the mouse copies the element.
  Moving with the right mouse button pressed opens the content menu, with a choice of whether to copy or move the item

- the context menu

MOVE

Objects can be moved with the mouse or the keyboard, with the exception of frames. You can use special effects in combination with function keys when moving:

Key	Action	Result
ctrl	move with mouse	object is copied.
Shift	move with mouse	object can only be moved horizontally and vertically.
Shift	move with arrows	Each key press moves the object by 10 pixels.
Right mouse button	move with mouse	The context menu lets you choose between Copy and Move.
Space bar		Mouse pointer becomes a hand. With this hand, the editing area can be moved with the left mouse button.

11.1.7 Add and delete dots

ADD AND DELETE DOTS FOR POLYLINES (ON PAGE 112), POLYGONS (ON PAGE 195) AND TUBES (ON PAGE 114):

ADDING CORNERS:

To add corners (corner points):

- move the mouse cursor to a line of the polygon:
- right-click
- select menu item **`Adding a node in the selected element`** from the context menu

or:

- press `Ctrl` and `Shift` simultaneously
- move the mouse cursor to a line of the polygon:
- the mouse cursor changes to an arrow with a plus-symbol
- left-click in order to add a corner

REMOVING CORNERS:

- move the mouse cursor to a corner of the polygon:
- right-click
- select menu item **`Deleting the node in the selected element`** from the context menu

or:

- press `Ctrl` and `Shift` simultaneously
- move the mouse cursor to a corner of the polygon:
- the mouse cursor changes to an arrow with a plus-symbol
- left-click to delete the corner

11.1.8 Arrows in vector elements

A symbol for the start and end of a line can be defined for all vector elements that display open figures. This applies for:

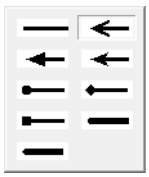
- Line
- Polyline
- Unfilled arc of a circle
- Unfilled segment of a circle

To add a symbol:

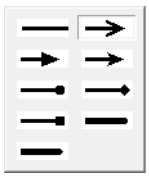
1. ensure that the `Graphics quality` property (`Screens` node) is activated
2. draw the element
3. In properties, in the `End of line` group, select the properties for:
   a) `Start type`: Symbol for start of line.
   b) `End type`: Symbol for end of line.
   c) `Size`: enlargement factor for line width. The factor relates to the line width of the vector element as defined in the `Line width [Pixel]` property. Default = 0

The following symbols are available:

from left or to the left:



from the right or to the right:



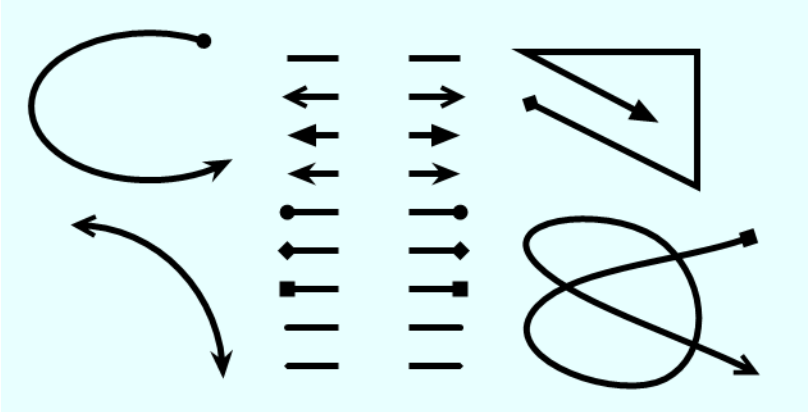
Info

These elements are not available:

- under Windows CE, because `Graphics quality` must be activated
- with a filled circle segment
- with a filled arc of a circle

Example for arrows

Vector elements with start and end symbols with an enlargement factor of 1.5:



11.1.9 Grid

The grid makes it possible to align elements precisely on the screen.

The basic settings are in the **`Options/Settings`** menu:

Grid settings	
`Display grid`	`Active`: Shows the grid that can be defined in the main window under `Grid type`. `Inactive`: Grid is not displayed.
`Position in grid`	`Active`: Screen elements are automatically aligned with the grid.

	This function is independent from the option `Display grid`.
`Grid color`	Defines the grid color. The windows color palette is used for selection.
`Grid type`	Defines the type of grid : solid, dotted or intersections.
`Grid pitch [in pixels]`	Defines the horizontal and vertical distance between unique marker points. Therefore, it also defines how finely scaled element sized can be displayed. Recommended grid distance: 10 pixels horizontally and vertically.

IN THE SCREEN CONTEXT MENU, YOU DECIDE

Menu entry	Description
`Display grid`	Switch grid to visible or invisible.
`Use grid`	Elements are aligned in the grid.
`Use magnetic points`	Touch points of objects move into place when neared and align themselves to each other.

Info

If grid points are 5 pixels or less, only the grid lines that are more than 5 pixels apart are displayed. The smaller grid lines are also used tor the positioning.

11.1.10 Font

A default font is automatically created when creating a project. This is provided as the standard font for new objects. The default font cannot be deleted. If you require other fonts:

- create the desired font in the Fonts node (subnode of Screens)
- select the font in the object properties (node `Representation`, property `Font`)

11.1.11 Selecting

Objects are selected by clicking the mouse. To select several objects:

- When clicking, hold down the `shift` or `ctrl` key
  or
- drag a connection rectangle with the mouse

To deselect an object when more than one object are selected at the same time, click it with the **`ctrl`** key held down.

CHANGE THE SEQUENCE OF SELECTION

If objects are to be aligned to each other, the first object selected serves as a reference. You recognize the reference object by the fully-filled touch points. To make another object the reference object:

1. hold down the `shift` key
2. click on the object that is to serve as the reference object with the mouse

ALIGNING A SELECTED OBJECT

You can find more information on aligning objects in the chapters on Editor/Tool Bar/Alignment and Editor/Menu Bar/Edit

11.1.12 Scale, center and zoom

SCALING

This you can scale elements in two ways:

1. proportional: Hold down the `shift` key when scaling
2. around the center: hold down the `shift` key to do this

Note: If an angel dissimilar to `0` via property `Rotation angle [°]` was defined for an element, scaling via arrow keys is not possible.

Info

Scaling of symbols:

If the symbol contains a ***static text*** *element and is embedded in this element as text, then this text:*

- Is scaled in the Editor
- Is not scaled in Runtime

This only applies to symbols. Embedded text is neither scaled in the Editor nor in Runtime in all arrangements.

CENTER

The currently selected element can be centered in the editing window with the H key.

ZOOMING

You can read information on zooming screens in the Screens toolbar chapter (manual: Editor)

11.1.13 Keyboard shortcuts

Shortcuts for the graphic editor:

Note for shortcuts: The plus sign (+) means that keys a pressed together.
For example:

Ctrl+A means: Hold key **Ctrl** and then press key **A**.

Ctrl++ means hold key **Ctrl** and press key **+**.

GENERAL

Command	Key combination
Main window: Scroll content with 'moving hand'	Press and hold `Space`
Close current screen	`Ctrl+F4`
Open properties	`Alt+Return`

SELECT

Command	Key combination
Select several objects	Press `Shift` or `Ctrl`
Deselect selected object during multi-select	`Ctrl+mouse click`
Selection: Change sort order. Defines the element on which all others realign	Press `Shift` during selection
Select hidden objects	1. `Press` *Alt* 2. *Click object and move it*
Select all elements of a screen.	`Ctrl+A`
Select next element according to the order of their creation	`Tab`
Select previous element according to the order of their creation	`Shift+Tab`

POSITIONING

Command	Key combination
Move selected object.	`Cursor keys`
Move by 10 pixels each time you press a cursor key	`Shift+arrow keys`
Move only horizontally or only vertically	Press `Shift` during moving
Centers the selected object in the working section	`H`

ACTIONS

Command	Key combination
Saves changes	Ctrl+S
Pastes element from the clipboard	Ctrl+V Shift+Ins
Inserts element from the clipboard at its original position; original and copy lie congruently on top of each user	Ctrl+Shift+V
Copies selected element.	Ctrl+C Ctrl+Ins
Copy instead of move	Press Ctrl during moving
Deletes selected element	Del
Cuts out the selected element	Shift+Del Ctrl+X
Undoes changes	Ctrl+Z Alt+Backspace
Add or delete node in the selected element. Add: Mouse cursor turns to plus symbol (+). Delete: Mouse cursor turns to minus symbol (-). Works for polylines, polygons and pipe elements.	Ctrl+Shift
Cancel drawing of polylines and polygons	S
Cancel drawing of polylines and polygons and delete the section which was drawn last	Esc
Move selected elements one level up	+
Move selected elements one level down	-
Move selected elements to the foreground	Ctrl++
Move selected elements to the background	Ctrl+-

SCALING

Command	Key combination

Change size	Move mouse cursor to the handle so that the mouse cursor changes to an arrow. After that you can position accurate to the last pixel with the help of the **Cursor keys** or in steps of 10 pixels with the help of the Cursor keys+Shift. **Note:** If an angel dissimilar to 0 via property Rotation angle [°] was defined for an element, scaling via arrow keys is not possible.
Scaling object around the center	Press Alt during scaling.
Proportional scaling	Press Shift during scaling

11.1.14 Assign key to control element

Operable elements in a screen can be linked using a key combination and operated in this way. This applies, for example, to:

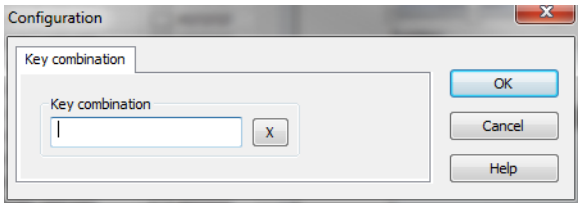
- Execution of control elements and assigned functions such as print, move screen, etc.
- Send value (such as with switch)
- Set focus of certain windows (such as curve list in Extended Trend or Comboboxes in the Recipe Group Manager)

The element control function can be carried out with the keyboard in Runtime.

To assign a key to a control element:

1. Highlight the control element
2. Navigate to the Key combination property (object properties -> Runtime -> Keyboard operation)
3. Click on the ... button or in the input field

4. The dialog for defining the key combination is opened



Parameters	Description
Key combination	Input of the key combination. Click in the field and press the desired key or key combination, for example: A. The key combination is displayed in the input field.
X	Deletes the character sequence in the input field.
OK	Accepts key combination and closes dialog.
Cancel	Discards input/change and closes dialog.
Help	Opens online help.

11.2 Replace link of variables and functions

If several variables or functions are to be replaced at the same time, it is best if you use automatic replacement for:

- Replacing linking in the Editor screen (on page 227)
- Replacing linking with screen switching (on page 229)
- Replace indices (on page 233)
- Linked symbols (on page 297)

You can also read about automated replacement for configuration in the **Efficient configuration with zenon** manual in the **Reusing elements** section.

11.2.1 Naming conventions

To be able to replace variables and other elements securely, the naming should be systematic and standardized if possible. You therefore support not only the reusability, but also maintenance and reverse engineering.

Different systems support you with systematic naming.

FOR EXAMPLE: ENERGY INDUSTRY

Germany

- KKS (Kraftwerk-Kennzeichen-System - Power Plant Classification System), for details (in German), see http://de.wikipedia.org/wiki/Kraftwerk-Kennzeichensystem (http://de.wikipedia.org/wiki/Kraftwerk-Kennzeichensystem)
- DIN 6779 (Kennzeichnungssystematik für technische Produkte und technische Produktdokumentation - Classification System for Technical Products and Technical Product Documentation), for details (in German), see http://de.wikipedia.org/wiki/DIN_6779 (http://de.wikipedia.org/wiki/DIN_6779)
- AKZ (Anlagenkennzeichnungssystem - Equipment Classification System), for details (in German), see http://de.wikipedia.org/wiki/Anlagenkennzeichnungssystem (http://de.wikipedia.org/wiki/Anlagenkennzeichnungssystem)

International

- KKS (Power Plant Classification System), for details, see http://en.wikipedia.org/wiki/KKS_Power_Plant_Classification_System (http://en.wikipedia.org/wiki/KKS_Power_Plant_Classification_System)

Such standards exist for all industries. It is recommended that their naming convention is used.

POWER PLANT CLASSIFICATION SYSTEM EXAMPLE:

Variables are to be named in accordance with the KKS in an energy project. A corresponding variable with the label `C01_MDY10-QA001_QA07` indicates:

- Wind energy equipment `C01` (row C, no. 1)
- Wind turbine control`MDY10`, Power part `QA001`,

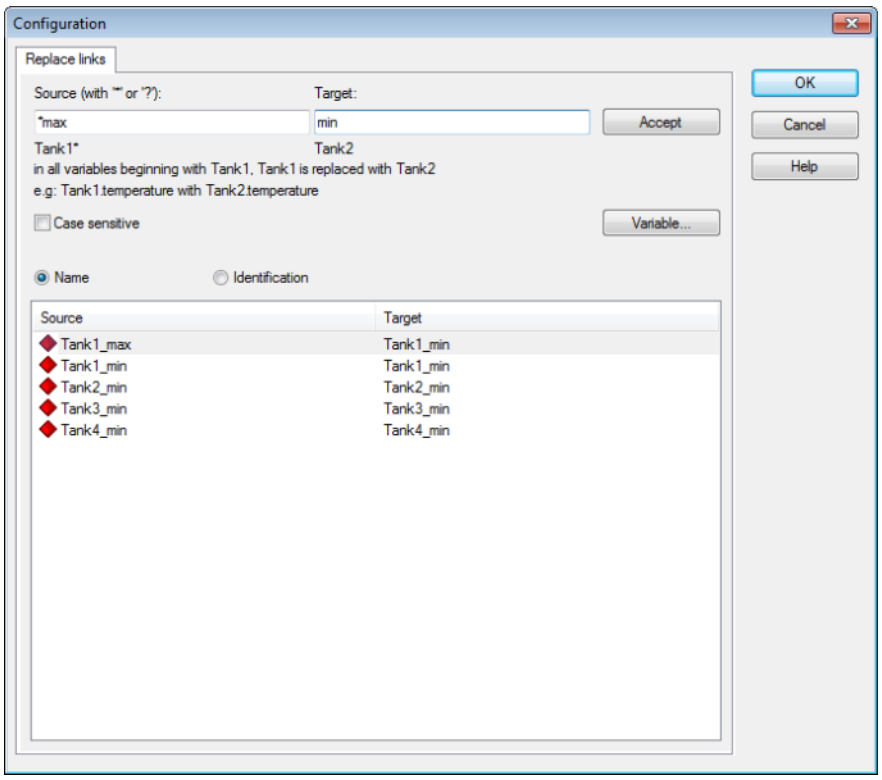
- Power protection **QA07**

11.2.2 Replace link in Editor screen

Linked variables or functions can be permanently replaced in the Editor in a rule-based manner. The replacement is not limited to the functions or variables defined in the properties of the `Variable / function` group. All other variables used the screen can, such as variables for `Visibility`, `Flashing`, `Size and rotation dynamic` and others. Rule-based replacement also works for several elements in a screen at the same time and for elements with embedded symbols.

To replace variables or functions in a rule-based manner in the element directly:

1. Highlight the desired element and open it by right clicking the element in the context menu
2. select **`Replace Links`**
3. The dialog for the replacement of links opens



Property	Description
Source	Enter the partial string to be searched for. Place holder ***** and **?** can be used. Placeholders are only permitted as prefix or suffix; e.g. *xxx or xxx*.
Target	Entry of the partial string
Name	Swaps information in process variable names.
Identification	Exchanges information in the identification
Note capitalization	When swapping, be sure that any capitalization is an exact match.
Accept	Swaps target strings from the `source` for those defined in the `target`.
Variable/function	Opens the selection list for variables/functions in relation to the selected line in the list. Clicking on the variable in the list defines new target variables. Alternative: Double-click on the source variable in question.

REPLACE

A) REPLACE BY MANUAL SELECTION

- select the element from the list that you would like as the source
- select a target element via the **`variable/function`** button
- the previous element is replaced by the new one

B) AUTOMATED REPLACEMENT WITH RULES

- In the source input field, define the parameters for the elements that you wish to replace
- define the parameters for the new variables/functions in the target input field
- specify what is to be replaced via Name/Identification
- Click on **`Accept`**

Several entries are separated by a semicolon (`;`).

Example:

```
Source:A;b

Target:C;d
```

> **Info**
>
> *The target variable or target function can also be in a different project as the source variable or source function.*

> **Attention**
>
> *When replacing variables, be aware of the type and signal resolution. If you replace a variable with one of an incompatible type, this can lead to errors during execution. You will be warned when making the substitution; the substitution will however be carried out.*

> **Info**
>
> *Replace via Drag&Drop: Elements that can only contain one variable or function can be swapped by dragging & dropping. Drag the new variable or function to the element using the mouse. It automatically replaces the previous one.*

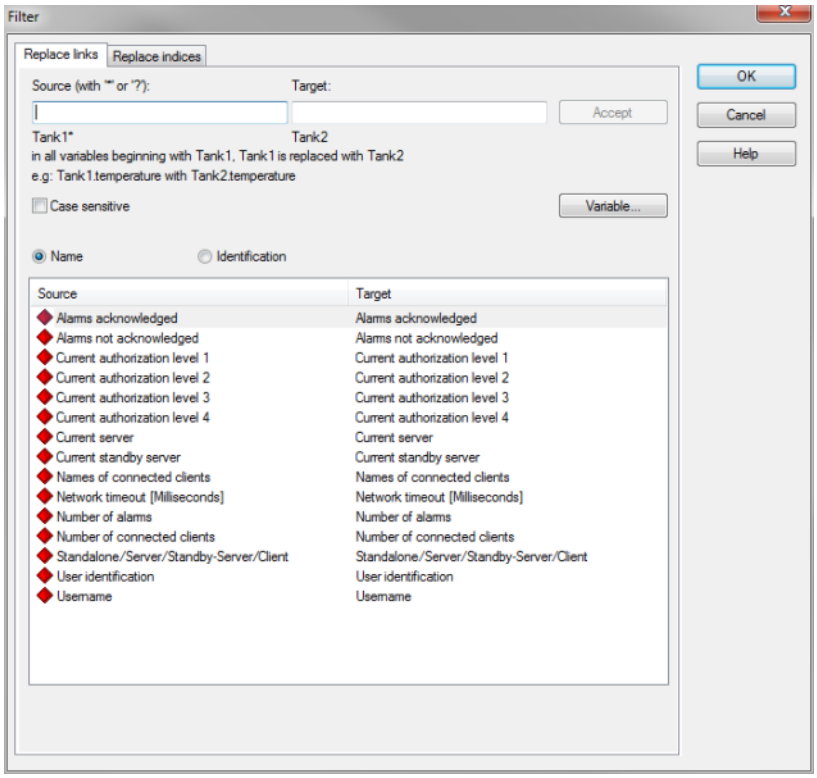
11.2.3 Replace link at screen switch

Linked variables or functions in one or more elements, as well as Alias in the ALC can be replaced with screen switching. In doing so, the elements in the screen are not replaced on a lasting basis, but the screen switching is linked in Runtime.

To replace linking when screen switching:

1. Configure a screen switching function
2. Contains elements that can be replaced in the screen, the dialog for replacement is opened

3. Assign the function to a button in order to be able to execute screen switching in Runtime



Property	Description
Source	Enter the partial string to be searched for. Place holder ***** and **?** can be used. Placeholders are only permitted as prefix or suffix; e.g. *xxx or xxx*.
Target	Entry of the partial string
Name	Swaps information in process variable names.
Identification	Exchanges information in the identification
Note capitalization	When swapping, be sure that any capitalization is an exact match.
Accept	Swaps target strings from the `source` for those defined in the `target`.
Variable/function	Opens the selection list for variables/functions in relation to the selected line in the list. Clicking on the variable in the list defines new target variables. Alternative: Double-click on the source variable in question.

REPLACE

A) REPLACE BY MANUAL SELECTION

- select the element from the list that you would like as the source
- select a target element via the **`variable/function`** button
- the previous element is replaced by the new one

B) AUTOMATED REPLACEMENT WITH RULES

- In the source input field, define the parameters for the elements that you wish to replace
- define the parameters for the new variables/functions in the target input field
- specify what is to be replaced via Name/Identification
- Click on **`Accept`**

Several entries are separated by a semicolon (`;`).

Example:

```
Source:A;b

Target:C;d
```

> **Info**
>
> *The target variable or target function can also be in a different project as the source variable or source function.*

> **Attention**
>
> *When replacing variables, be aware of the type and signal resolution. If you replace a variable with one of an incompatible type, this can lead to errors during execution. You will be warned when making the substitution; the substitution will however be carried out.*

> **Info**
>
> *Replace via Drag&Drop: Elements that can only contain one variable or function can be swapped by dragging & dropping. Drag the new variable or function to the element using the mouse. It automatically replaces the previous one.*

REPLACING INTERLOCKINGS

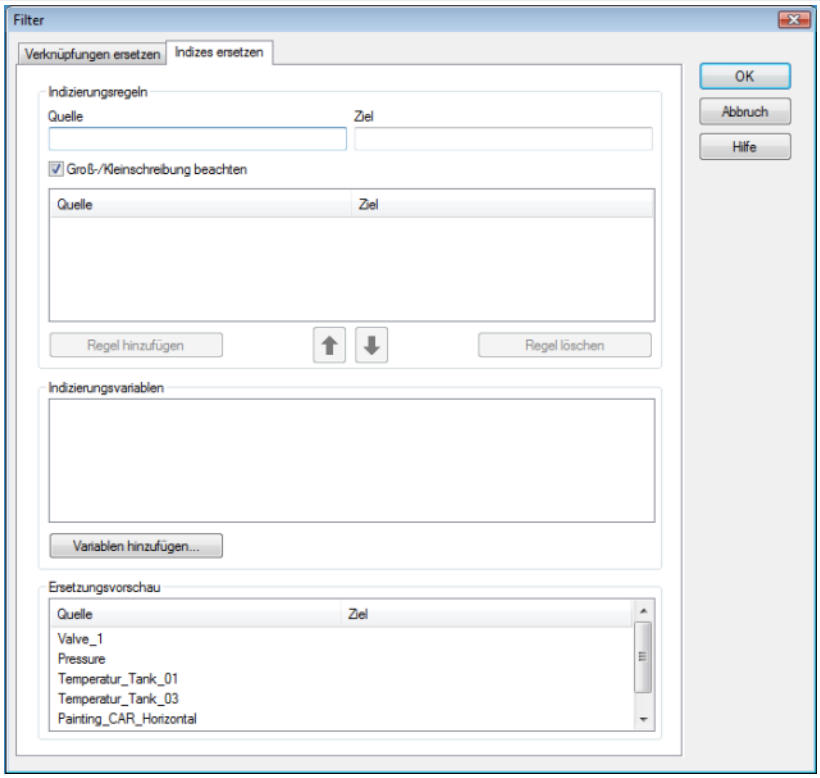
In the dialog, the variables of the interlockings used in this screen appear next to the variables directly linked to the screen. This also includes the result variable of an interlocking. If a variable was linked to both the elements of a screen and with interlockings of a screen, they only appear once in the dialog.

The variables to be replaced for all interlockings that use these variables are replaced in Runtime. In addition to the variables normally linked to interlocking, the result variables linked to the interlocking are replaced for the screen.

11.2.4 Replace indices

When switching screens in Runtime, variables can be replaced dynamically (virtually) using indexing rules. In doing so, variable values can be used as part of the replacement rule.

> **Attention**
>
> *Because the names are replaced, variables and functions must always be given different names. Variables and functions are replaced if the names are the same.*
>
> ***Example:***
>
> *If there is a variable "test" and a function "test" and the variable "test" is to be replaced, the function "test" would be replaced at the same time.*
>
> ***Tip:*** *Name variables and functions with a respective prefix (such as var_ and func_). You therefore always have a overview in lists too.*



To configure the replacement of indices:

1. All variables and functions used in the screen are displayed in the `Source` field of the **`Replacement preview`**.

2. Left-click the variable or function for which you want to create an indexing rule.
3. In the **Indexing rules** area, the variable/function is entered in the fields **Source** and **Target**. Alternatively, you can also enter the desired variable/function directly into the **Source** field in the **Indexing rules**.
4. Define the indexing rule.
   The dynamic part of the rule is inserted in curly brackets, for example {X01}.

Example:

The source variable **Motors1.speed** is replaced by the target **Motors{X01}.speed**, the placeholder **X01** is replaced with the respective value of the index variable in the Runtime.
The replacement rule would thus be:

Source: **Motor1**

Target: **Motor{X01}**

5. Carry this over to **Add Rule** by left clicking in the indexing rules.
6. All indexing rules that have been created are displayed in this list.
7. You can change the order of the entries using the button with the arrow symbol.

Attention: The substitutions are executed in the order in which they are displayed in the list!

8. Variables that are not present on the screen but are used in the replacement can be inserted via the **Add Variables...** button in the **Indexing Variables** window.
9. The variables which are in this list are always kept in the memory in order to minimize waiting time when the screen is loaded.

> **Info**
>
> *For the placeholder * the same rules as for Replace links (on page 227) are valid.*

Attention

If index variables are used in a network project, the Client must first fetch the value from the Server. Then the calculation is executed. On devices with outdated hardware this may cause delay at screen switching.
Therefore in a network project index variables should always be set to `DDE active`.

CROSS-PROJECT

If the target is in a different project to the source, this can be displayed using '#'.

Example

Source: VAR_1

Target: VAR_1 in Project_1

You thus enter VAR_1 as the source and Project_1#VAR_1 as the target.

Source: SUBPROJECT1#VAR_1

Target: {X01}VAR_1

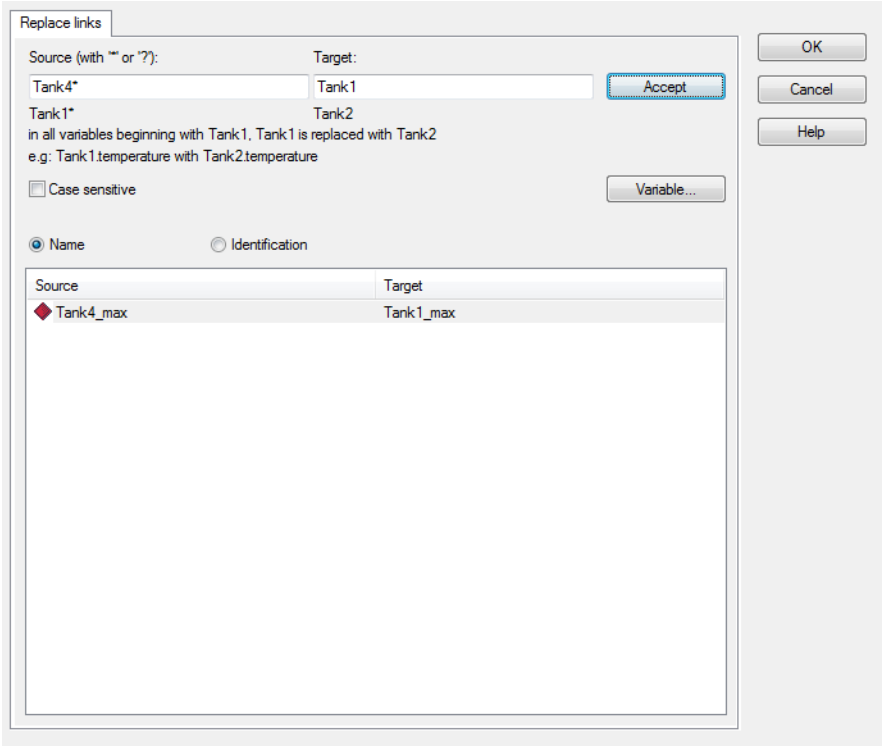
The index variable X01 is a string variable and has the project name including # as a value, for example "SUBPROJECT4#".

11.2.5 Example replace variable rule-based

All variables with "Tank1" in the name are to be replaced with "Tank4".

1. Enter "Tank1" into the `Source` text field. You can include all variables that start with "Tank1" with the `*` character.
2. Enter "Tank4" into the target text field.
3. Click on Accept.

4. The variables Tank1_max and Tank1_min are replaced by the variables "Tank4_max and Tank4_min.



11.3 Background graphics

A graphics file can be displayed in the background of every screen. These graphics are positioned in the visibility level furthest to the back and are covered by all other screen elements. It works along the lines of the background graphics of Windows Desktop.

Options such as centered or tiled are available for positioning. It is not possible to freely move the graphics.

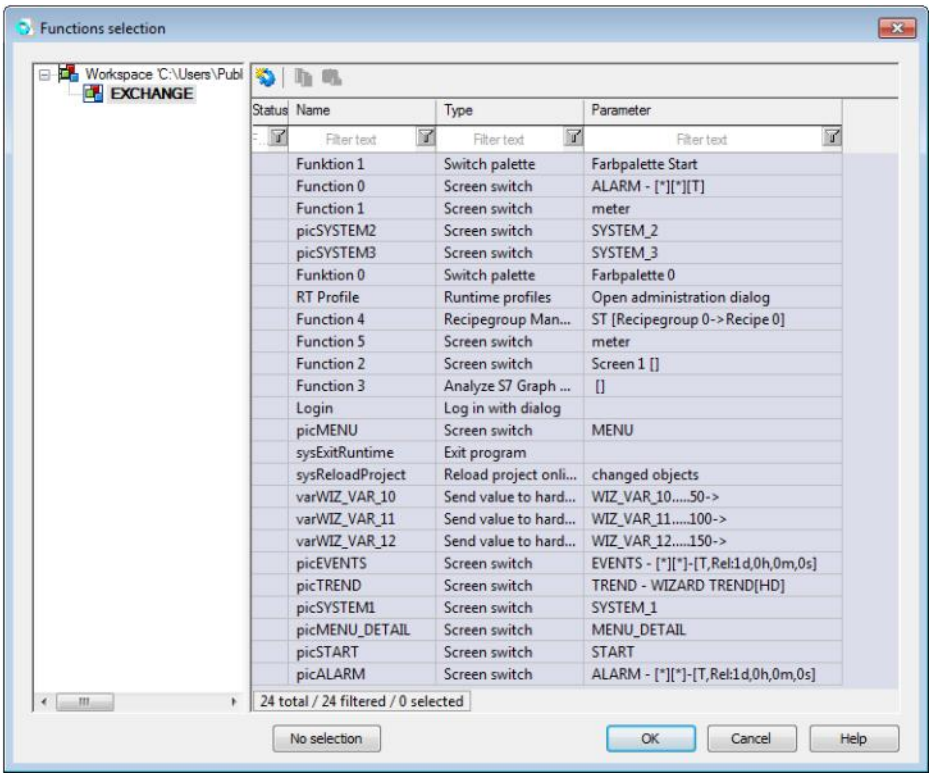
Only graphics data that is already available in the project in the Files/Graphics area can be used. See: Files.

Tip: If you wish to use the same background graphics for several screens, you can pre-define the background graphics in the frames. See: Background graphics for frames (on page 253)

Tip: Alternatively, a button with a graphics file can be used without a linked function to display a freely-positionable graphics file

11.4 Functions Selection dialog

In order to select functions, a filtered dialog is opened:



Element	Description
Project tree	Definition of the project from which the function should be selected.
Selection window	Selection of the function.
No selection	Depending on the element: ▶ the dialog is canceled ▶ already linked functions are deleted

Hint: The size of this dialog can be adjusted. The dialog size and position are saved.

CHANGE LINKED FUNCTION

1. manually

- in the Project Manager´s Detail view open the functions
- drag the desired function on the element while holding the left mouse button
- the previous function is substituted by the new one

2. automatic
   - Select the entry Replace links from the context menu. For more details see chapter Substitution of variables and functions in dynamic elements (on page 225).

11.5 Variable selection dialog

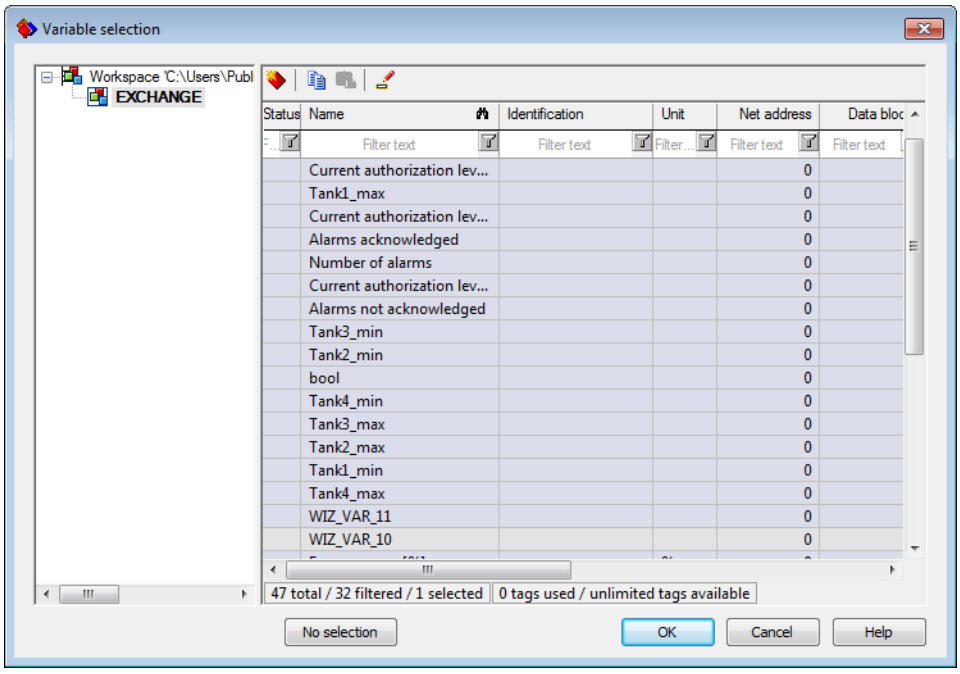
In order to select variables, a filtered dialog is displayed.

Hint: The size of this dialog can be adjusted. The dialog size and position are saved.

Variables can also be used throughout projects.

LINKING A VARIABLE

Elements that can only be linked with one unique variable open the following dialog:



Element	Description
Project tree	Definition of the project from which the variable shall be selected.
Selection window	Selection of the variable which shall be linked. It can be linked in two ways: ▶ By double-clicking on the desired variable: The dialog is closed automatically. ▶ By selecting the desired variable and subsequently clicking OK.
No selection	Depending on the element: ▶ the dialog is canceled ▶ Variables that are already linked are deleted (such as linked batch variables in the Historian)

Note: If the desired variable is not displayed, there can by several reasons for this:

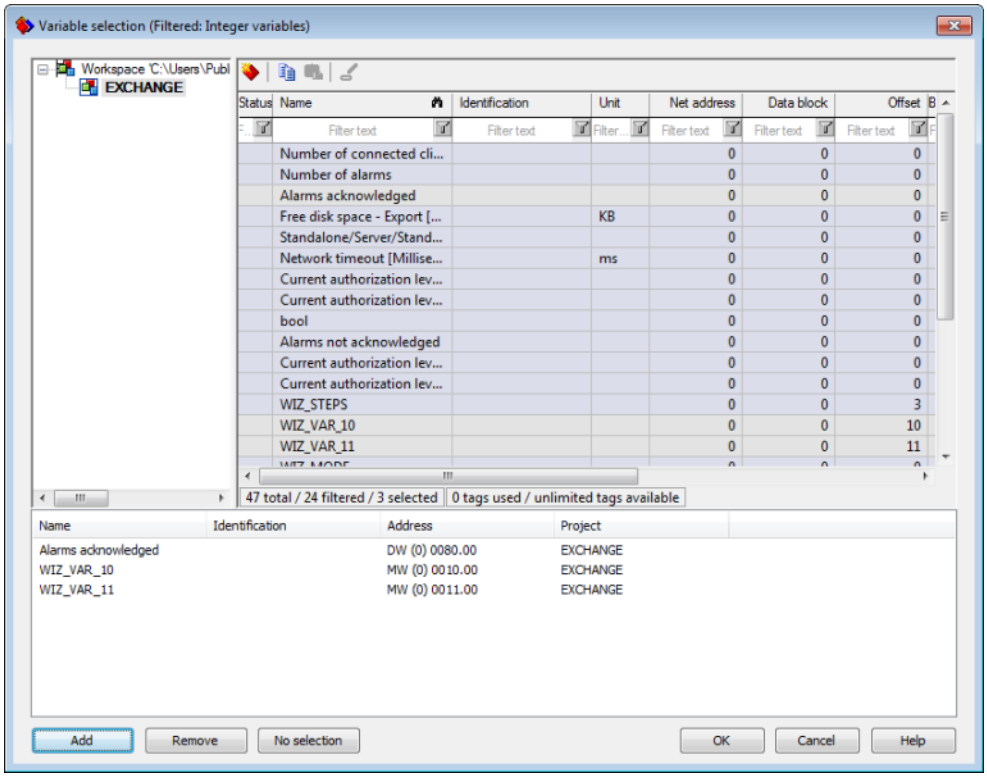
- You have selected a filter and the desired variable does not match the filter: Check whether there is a filter active in the filter line. The information about filtered elements which is displayed at the bottom of the selection window also helps when checking for an active filter.
- The dialog is filtered by the Editor and the variable does not match the filter. The filtered display takes place when only special data types are allowed; e.g. for the dynamic element Numerical value no string variables can be linked.
- You have selected the wring project in the project tree.

Attention

The dialog generally allows the choice of multiple variables via the usual Windows keyboard shortcuts, but in this step only one can be linked. For multiple selection the first chosen variable is assigned to the element.

LINKING MULTIPLE VARIABLES:

Elements that can be linked with multiple variables open the following dialog:



Element	Description
Project tree	Definition of the project from which the variable shall be selected.
Selection window	Selection of the variables: ▶ Double click the selected variable in order to add it to the variable list. ▶ You can move the selected variable to the variable list via Drag&Drop ▶ Select the desired variable. With the help of `Ctrl` and/or `Shift` multi-selection is possible. By clicking **Add** the selected variables are added to the variable list.
Variable list	Lists all selected variables.
Add	Adds the currently selected variable of the selection window to the variable list.
Remove	Removes the variables which are selected in the variable list from the list.
No selection	Depending on the element: ▶ the dialog is canceled ▶ certain links such as batch variables in archiving can be released

Note: If the desired variable is not displayed, there can by several reasons for this:

- You have selected a filter and the desired variable does not match the filter: Check whether there is a filter active in the filter line. The information about filtered elements which is displayed at the bottom of the selection window also helps when checking for an active filter.
- The dialog is filtered by the Editor and the variable does not match the filter. The filtered display takes place when only special data types are allowed; e.g. for the dynamic element Numerical value no string variables can be linked.
- You have selected the wring project in the project tree.
-

CHANGE LINKED VARIABLE

Select the entry Replace links from the element´s context menu. For more details see chapter Substitution of variables and functions in dynamic elements (on page 225).

12. Frames

Frames form the basis for the layout of the window and the screens displayed during runtime. You structure the display on the screen and determine the position. Each screen is assigned to a frame and displayed in the screen area defined by the frame.

Frames determine the general window properties such as position, size, appearance and manipulation options during runtime. Frames:

- Provide an overview of all windows defined in a project
- can be freely defined via the `Define free frame shape` property
- enable special functions to be allocated to all screens available in a frame, for instance:

  Return to last screen, set focus on frame, take focus from frame, close frame, acknowledge flashing alarms, print screenshot
- change all screens based on this frame
- Screens always appear in the same size at the defined location in runtime and cannot be moved as desired
- Frames can always be automatically closed during runtime if the focus is lost
- Screens within a frame can be changed at will in online operation.

> **Info**
>
> *If there is no frame when creating a screen, then zenon automatically creates a default frame that covers the whole screen.*

PROJECT MANAGER CONTEXT MENU

Menu item	Action
Frame new	Creates a new frame.
Open frame editor	Opens the frame editor in the main window and shows the list of frames in detail view.
Export XML all ...	Exports all entries as an XML file.
Import XML...	Imports entries from an XML file. Note: Existing frames are overwritten via the **Frames** node during XML import. In this case, all screens and their elements are adapted to the new frame.
Help	Opens online help.

CREATE FRAME

To create a palette:

1. In the project manager, select the **Frames** sub item in the **Screens** node
2. select **Create New Frame** in the context menu
3. a new frame is created
   The initial size is determined by the size of the display and can be set in the monitor administration(*Project Properties*-> Graphical design -> Monitor administration)
4. click in the new frame and drag the frame over the touch point to the desired size
5. position the frame at the desired location in the window
6. Define the desired properties in the properties window

> **Attention**
>
> *Frames in the global project and the subproject must be named differently.*

Example

three areas must be present in the whole project:

- A header appears at the top of the screen with the time and a logo.
  a "header" frame is defined for this area. This defines the size and position of the header.
- A menu appears at the bottom of the screen.
  A "menu" frame is created.
- Interim process screens.
  A "process screen" is created.

The frames are always displayed in the defined size at the defined location. All screens created then are linked to one of these frames and in the Runtime opened in it. If the screen design is changed, because, for instance, menu bars are to be displayed at the top and the header is to be a footer, only the position of the frames need be changed. The screens contained in the frames are automatically carried over.

12.1 Frame detail view toolbar and context menu

Menu item	Action
Frame new	Adds a new frame to the list and puts the focus onto this entry.
Open frame editor	Opens the frame editor in the main window.
Define free frame forms	Changes to the main window to define a free frame shape.
Jump back to starting element	If you entered the list via function **linked elements**, the symbol leads back to the start element. Only available in the context menu when all linked elements are opened.
Copy	Copies selected entries to the clipboard.
Paste	Pastes the contents of the clipboard. If an entry with the same name already exists, the content is pasted as "Copy of".
Delete	Deletes selected entries.
Export selected XML	Exports selected entries as an XML file.
Import XML	Imports XML files. **Note:** Existing frames are overwritten via the **Frames** node during XML import. In this case, all screens and their elements are adapted to the new frame.
Remove all filters	Removes all filter settings.
Edit selected cell	Opens the selected cell for editing. The binocular symbol in the header shows which cell has been selected in a highlighted line.
Replace text in selected column	Opens the dialog for searching and replacing texts.
Properties	Opens the **Properties** window for the selected entry.
Help	Opens online help.

12.2 Frame editor

You can define and position all frames in the frame editor.

> **Attention**
>
> *It is not possible to undo actions in the frame editor. The menu item* **Undo** (`ctrl+Z`) *is not available.*

12.2.1 Opening the frame editor

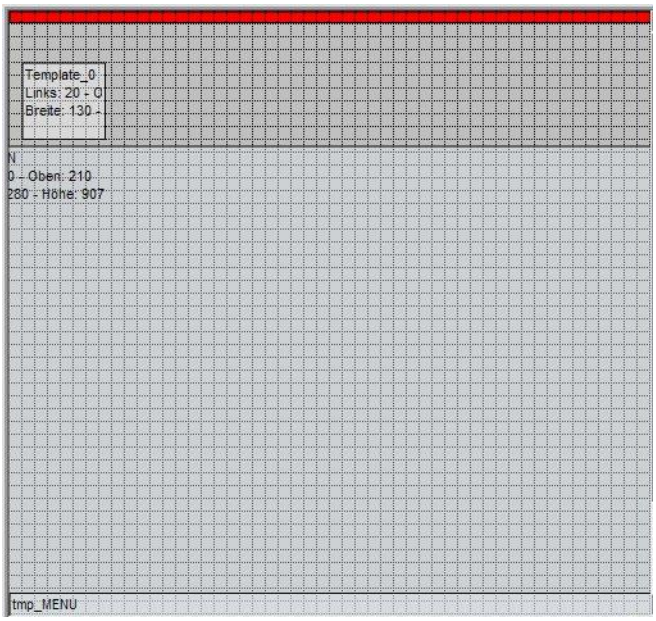
To open the frame editor:

- In the project manager, click on the desired frame twice in Frames Detail View

or

- select *Project Manager -> Frames -> Context Menu -> Open Frame Editor*

The size displayed in the frame editor is dependent on the monitor size set in monitor administration. See chapter on monitor configuration.) If the monitor settings are subsequently changed, the frame editor must be closed and reopened in order for it to take the new monitor settings into account.



The red frame for the alarm status line is automatically created at the upper edge of the screen by zenon for each project. Its position can be changed via the properties in the `Position` group. To do this, the `Use standard position` property must be deactivated.

12.2.2 Display frames in frame editor

Frames can be switched to visible or invisible by:

- Property`Display in Frame Editor`:
  activated: displays frame
- Drag&drop:
  invisible: Drag frame from the main window
  visible: drag frame from the detail view into the main window
- Detail view-> View of frame editor:
  invisible: Deactivate checkbox
  visible: Activate checkbox

You obtain information on the name and position of the frame in the upper left corner of the frame and via tooltip if you move the mouse pointer over the frame.

> **Info**
>
> *A system menu or an alarm status line is automatically taken into account in the frame editor and always displayed in the foreground. An incorrectly positioned frame can therefore not hide any important information during runtime.*

12.2.3 Create new frame

You have two possibilities to create a new frame:

- create a new frame:
  Project Manager -> Frames -> Context Menu-> Create New Frame or Frame *Detail View- > Context Menu - > New Frame*

  With this, a new standard frame is created in the size defined in monitor properties.
- to draw a new frame: *Main Window-> Context Menu -> Draw New Frame*

  The mouse pointer becomes a black square with a cross in the upper left left corner as the starting point. With this, you drag the frame in the size you have defined to the desired position.

POSITIONING FRAMES

The position and orientation of a frame or a group of selected frame can be selected via the **Arrange** context menu in the main window.

POSITIONING WITH CONNECTION POINTS

If the option 'Use connection points' was activated (see chapter Editor, Section Settings), the connection points of frames stick to those of other frames. In addition, frames can connect to the screen border and to a possibly present (if activated in the project) alarm line.

Frame with free frame shape

You can adapt the frame shape to your individual wishes using the `Freely defineable frame shape` property. After you have activated this property, you arrive at the frame editor in one of the following ways:

- Click on the button of the `Define free frame shape` property
- Click on the corresponding symbol frame in the toolbar (on page 244)
- Select the menu entry Define Free Frame in the context menu (on page 244) when selecting the frame
- Select the Define Free Frame menu entry in the Context menu in the frame editor
- Double click on a frame in the frame editor

Here, you can create the frame as you wish with the rectangle, circle, rounded rectangle and polygon elements.

Info

`Freely defineable frame shape` is not available:

- For alarm status line frame
- Under Windows CE

If the `Define free frame shape` property is deactivated an existing free frame form is not displayed in either the frame editor or the screen editor.

You decide how the respective element influences the appearance of the frame with the help of the `Area utilization` property. To do this, the following settings are available:

Parameters	Description
Cover	The surface of the element is used as the screen area.
Cut	The surface of the element is removed from the surfaces underneath. For instance, you can create "holes" in the frame, for example.
ORed	This is either cut out or covered according to whether a covering element is under the element or not.

Info

The way the surface is used always relates to the surfaces under the element. If you, for example, draw a covering element over a cutting element, the surface is displayed filled. If you reverse the sequence, the cutting element is removed from the covering element.

12.2.4 Positioning in Runtime

Frames can be switched to fixed or relatively defined positions in Runtime. In this way, keyboard screens or pop-up screens can be positioned exactly where the mouse pointer or the element to be switched is located. Frames can be switched in runtime:

- at an absolute position
- relative to the mouse position
- relative to the element

Absolute and relative positions are defined in properties.

RETAIN POSITION FOR NEXT TIME

Windows always take their screen position from the template. If pop up screens are planned and moved whilst the program is running, the pop up screen is moved back to the planned position the next time the screen switching function is activated. Using the Keep position when displaying again property, the template can be set to keep its position as long as it is open, regardless of screen switches. If the property is active:

- If a screen is opened in Runtime in a manually-moved template, its position is also retained if the screen is switched.
- If another screen is opened in the in the same template, its position is retained. This also applies for a substituted call.

With this, a different position position for a template can be used on each client, because the local position is always saved.

ABSOLUTE POSITIONING

The frame is always switched at fixed, pre-defined coordinates. A fixed screen position is always used for all screens that are loaded automatically by the system, e.g. by the time control or a limit violation.

RELATIVE POSITIONING

Relative positioning is especially suited to loading pop-ups, keyboard screens or command input screens. This is possible for all screen switches, set value inputs with freely defined keyboard screens, and command input screens that are started manually by the user. It does not matter whether the action was triggered by a screen switch function, a set value function or directly via an element. Relative positioning is available for:

- dynamic elements to which you can allocate a function
- dynamic elements that can be used to send a set value (buttons, combined elements)
- Main and Context Menus
- execute the screen-specific function in the Alarm Message List screen

Relative positioning also works for nested function calls, i.e. when screen switching or the setting of values are triggered by a script. If the screen would stand over the screen border, it is automatically positioned at the screen border. This also applies to multi-monitor management, where screen would rise into the neighboring monitor.

RELATIVE TO THE ELEMENT

Positions the reference point of the frame relative to the element from which the screen switch was initiated (e.g. a button or a combined element). The frame's reference point is then always placed on the element's calculated reference point. If the frame cannot be switched off in this position because the screen would then be outside the visible screen area, the alternative reference points are used.

First define a reference point for your frame and then the preferred position in the element. The following settings are available:

Property	Value
Reference point element vertical	below
	top
Reference point element horizontal	Left
	Right
Vertical movement in pixels	Enter the desired movement
Horizontal movement in pixels	Enter the desired movement

ALTERNATIVE POSITION

Define an alternative position for the reference point of the frame and the reference point at the element. Proceed with the definition of the alternative positions in the same way as with the definition of the favored positions. The alternative position is used when the frame can not be placed at the favored position.

RELATIVE TO THE MOUSE POINTER

The screen is displayed depending on the position of the mouse pointer. If the reference point is at the top left side, the frame will be loaded to the bottom right side of the reference point. If the screen cannot be displayed completely at the configured position, the position will be moved until the complete screen is inside the displayed area of the screen.

Define the frame's reference point above the two properties, vertical reference point and horizontal reference point. The reference point is marked on the frame with a red dot. The following settings are available:

Property	Value
Reference point frame vertical	top
	below
	centered
Reference point frame horizontal	Left
	Right

	centered

Example

you have a button at the right border of the screen and try to position a popup window on its right side:

- Element reference point: Right/centered
- Frame reference point: left/centered

In Runtime, the popup screen will appear on the right side of the button, centered to the middle of the button. The popup screen will not cover the button.

You use the same frame also for a button on the left screen border. There may not be enough room on the right side to completely display the popup screen. The alternative configured position will therefore be used. For this, you place the

- alternative reference point for the element on: left/centered
- the frame's alternative reference point on right/centered

12.2.5 Deleting frames

To delete frames, you can:

highlight one or more of the frames with the mouse and select **`delete`** in the context menu or press the `delete` key

highlight one or more of the frames in the detail view and select **`delete`** in the context menu or press the `delete` key

A request to confirm this action is made before it is deleted.

Attention

Deletion of frames cannot be undone.

12.3 Define background graphics for frames

A graphics file can be displayed in the background of every screen. These graphics are positioned in the visibility level furthest to the back and are covered by all other screen elements. They are positioned via properties; it is not possible to position them freely. Only graphics data that is already available in the project manager in the files/graphics area can be used.

If several screens use the same background color, this is defined with the screen frame:

1. In properties, click on the `Graphics file` property in the `Background graphic` node
2. select a file from the files/Graphics node in project manager
3. define the position with the `Alignment` property

this graphics file is then used for all screens that are created using this frame. However, individual graphics files can still be defined as a background screen for each screen.

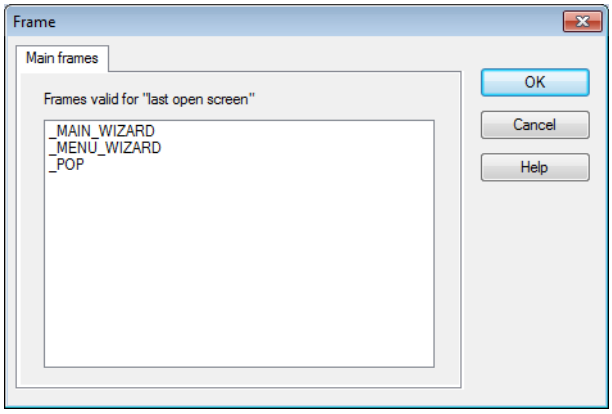
Attention! The graphics file is saved with the screens. Changing the graphics file or how they are set up in the frame is only effective for new screens. Existing screens must be changed manually.

12.4 Distributed engineering with frames

In multiuser operation, all usual mechanisms for allowing and accepting changes are implemented. In the frame editor however, in contrast to the screens, every frame is handled individually in the frame editor as opposed to the entire screen being released. For a better overview, frames where changes are not permitted are marked with a dotted border.

12.5 Main frame

Templates that were opened last can be displayed in the system driver using the variable **Last screen open**. Which templates are incorporated into the display is defined with the Main frames property in the project manager (Graphical design, Runtime general). Click on the ... button to open the dialog for selecting a template:



Several entries can be selected using the combination of Ctrl + a mouse click or Shift + a mouse click. Clicking on OK confirms the selection.

Info

*This property also has an effect on the **Screen: return to last** function.*

12.6 Effects for frames

Frames can also be provided with effects.

You can find additional effects for screen elements in chapter effects for screen elements (on page 196).

12.6.1 Highlight frame

With the help of **Highlight frame** you can highlight a screen in order to draw attention to a certain screen. For this the frame, on which the screen is based, is highlighted by coloring the remaining area of

the Runtime. For this a color and a transparency are defined for the surrounding of the frame of the screen which covers the screen lying below. The screen lying below cannot be operated.

Note: The screens must be based on different frames with different sizes. You must not call up more than one screen with this property at any time.

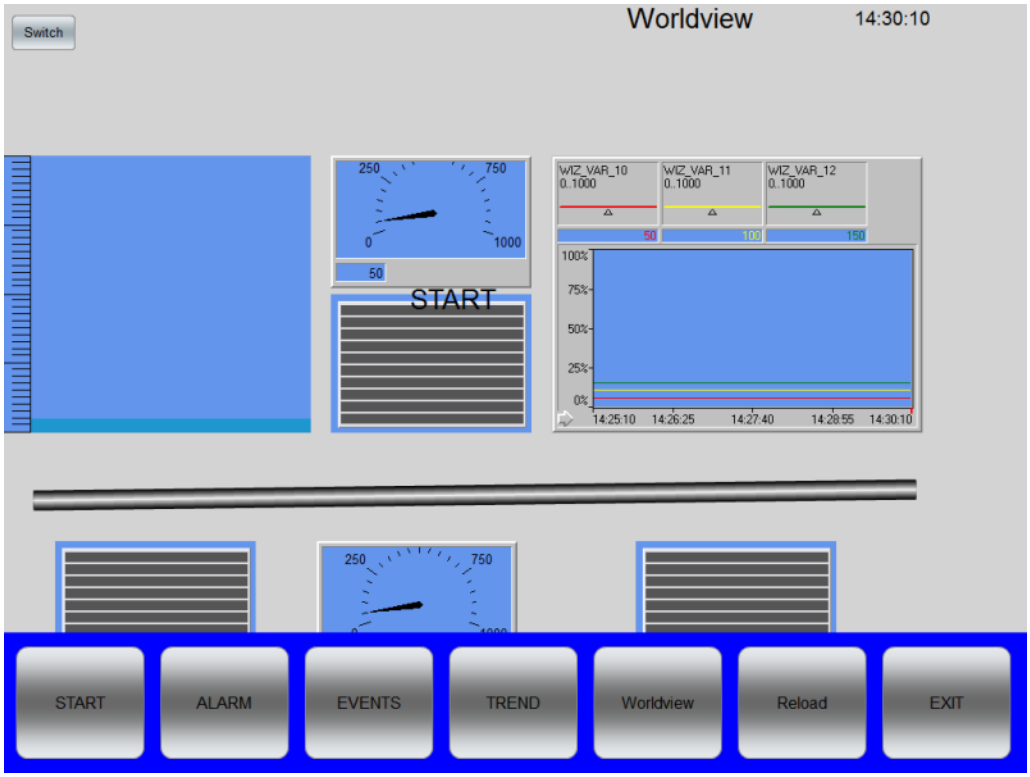
To use the effect, configure the screen which is to be on top of another, larger screen:

1. navigate to group `Window style` in the screen properties.
2. Activate the `Modal dialog` property.
3. Activate the `Highlight frame` property.
4. With property `Surrounding color` you configure the color with which the surrounding of the active screen should be colored.
   This property always affects the frame on which the screen is based.
5. With property `Surrounding transparency` configure the transparency of property `Surrounding color`.
6. Engineer a button for the screen in order to be able to close it in the Runtime.

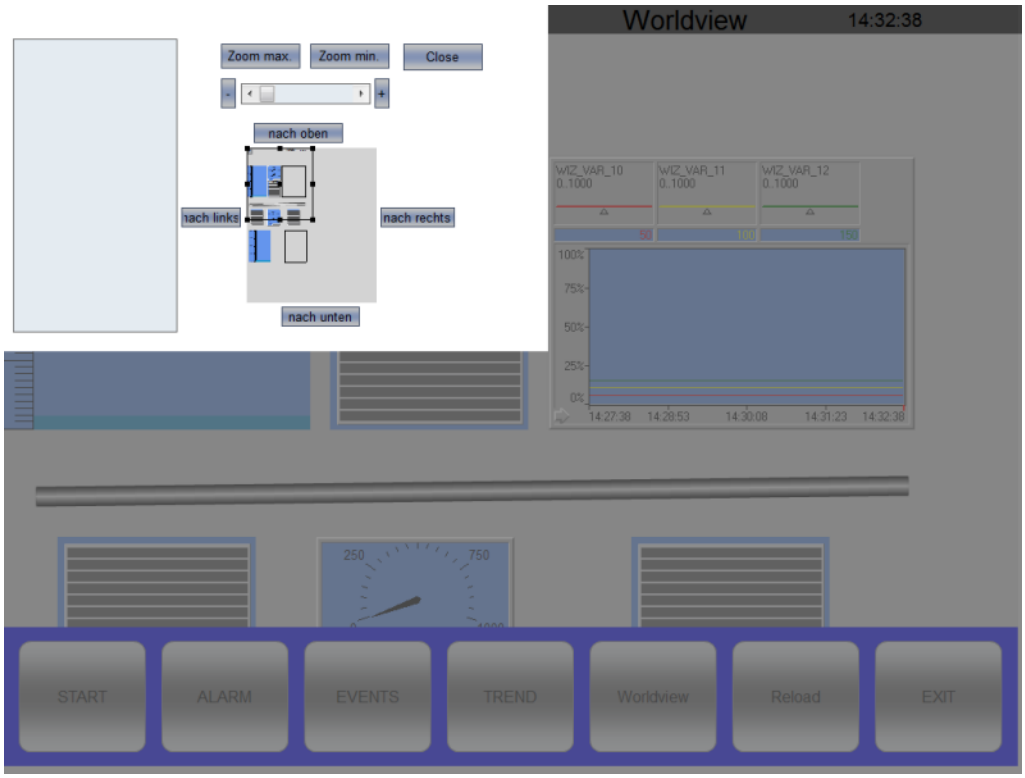
As soon as you open this screen above another, the area surrounding the frame is colored. Thus optically highlighting the screen. Only this screen can be operated. After closing the screen, other screens are displayed in normal color again and can be operated.

EXAMPLE

Screen is opened:



Second screen with active effect **`Highlight frame`** is also opened. Everything outside the additionally opened screen is highlighted with the color defined in property `Surrounding color` and the transparency defined in property `Surrounding transparency`.



13. Color palettes

Color palettes make it possible to compile individual colors into color palettes. You can define color set such as this, that can be easily edited, both in the editor and in Runtime. All colors that are defined with the help of palette colors can be easily changed over with this.

The uniform design of Company Designs (CD) can be completed very quickly in this way. If necessary, the design can be centrally changed in full (change palette) or only individual colors can be changed (change color in palette).

The zenon approach: Changes can be effortlessly made centrally or even globally with the help of a global project in this way.

CONFIGURATION

These palettes are created in the editor, can be exported and imported and there is a function to switch these in runtime. Color palettes can be created in both a global project and a local project. All palettes of a project must always contain the same amount of colors; the number of colors can therefore differ between a global project and a local project.

Attention: in the event of conflicts in runtime, the local project's color palettes take priority.

The colors of the color palette can be used anywhere where colors are used.

PROJECT MANAGER CONTEXT MENU

Menu item	Action
New color palette	Creates a new color palette with the standard names **Color palette** plus continuous **Index number**.
Export XML all ...	Exports all entries as an XML file.
Import XML...	Imports entries from an XML file.
Help	Opens online help.

13.1 Palettes toolbar and context menu detail view

TOOLBAR AND CONTEXT MENU

no.	Symbol	Action
01	**New color palette**	Creates a new color palette with the standard names **Color palette** plus continuous **Index number**.
02	**Create standard function**	opens the dialog to create a Palette Switching Function (on page 266) and automatically activates the selected palette in the drop-down list.
03	**Delete color palette**	Delete the palette of the highlighted color. Palettes can only be deleted individually
04	**New color**	Creates a new color with the standard color white and the standard name **Color** plus **Index number**. New colors are always created for for all palettes of the project. Hint: Give clearly distinct names for the global project and the local project.
05	**Delete color**	Deletes the selected color from all palettes of the project. Note: The command is only active if the last color in each palette is highlighted. Because: Only the last color of a palette can be deleted, so that the index numbers continue to work. Furthermore, the color must be deleted for all palettes, so that the number of colors remains identical in all palettes.
06	**Copy**	Copies selected colors or color palettes to the clipboard.
07	**Paste**	Pastes selected colors or color palettes from the clipboard.
08	**Jump back to starting element**	If you entered the list via function **linked elements**, the symbol leads back to the start element. Only available in the context menu when all linked elements are opened.
09	**Color palette in the editor**	opens a drop-down lists with all palettes that can be defined in the local project and in the global project.
10	**Save**	Saves all changes to the color palettes.
11	**Export selected XML ...**	Exports all selected palettes as an XML file.
12	**Import XML...**	Imports entries from an XML file.
13	**Properties**	Opens the property window.
14	**Help**	Opens online help.

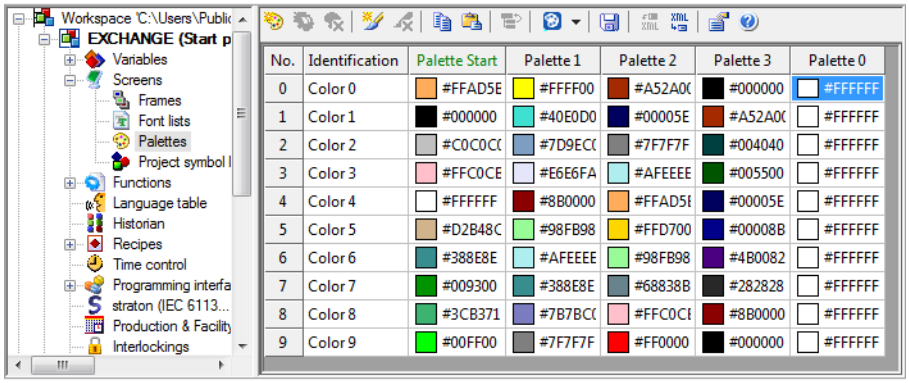
13.2 Create and edit colors

You can create color palettes globally or in a local project. Global color palettes are available in all projects, local color palettes are only available in the project in which they were created. If there are conflicts when setting these in runtime, the color projects of the local project take priority(see alsoColor palettes in Runtime (on page 265)).

CREATE COLOR PALETTE

To create a color palette:

- select the `Color Palettes` node (either below `Screens` in the local project of global project)
- select the **`New Color Palette`** command in the context menu
- a new palette is created with:
  - Standard name **`Color palette`** plus **`Index number`**, for example **`Color 0`**
  - the same number of colors as the pre-existing palettes, all colors are `white` as standard



RENAME COLOR PALETTE

To rename a palette individually:

- select the palette or one or more colors of the palette
- In properties, select the `Palette` property in the `Palette name` group

- enter the desired palette name
  Tip: Give them clear names in order to clearly distinguish between the global project and the local project

DELETE COLOR PALETTE

To delete a palette:

- highlight the palette's complete column
- In the context menu, select the **Delete Color Palette** command or press del
- the palette will be deleted without asking for confirmation.
  Note: Palettes can only be deleted individually

13.3 Create and edit colors

CREATE COLOR

In order to create a color:

- select a color or a palette
- In the context menu, select **New Color** or press the insert key
- at the lower end of the color table, a new color is inserted for all palettes with
  - Standard color white and
  - Standard name **Color** plus **Index number**, for example **Color 10**
- define the color and give it a name

RENAME COLOR

To rename a color:

- select the color
- In properties, select Color the propertyIdentification in the group

- enter the desired name

  Tip: You can already differentiate between a global project and a local project when giving it a name, so that you know where the palette comes from when choosing the color palettes later

DEFINE COLOR

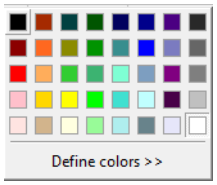
to define a color:

- highlight the desired color in the palette
- enter the color:
  - either directly into the table cell in the detail view, as a hexadecimal code or via the selection dialog by clicking on ...
  - or in the properties in the `Color` group in the Property `Color` field as a hexadecimal code or via the selection dialog by clicking on ...

DEFINE COLOR VIA DIALOG

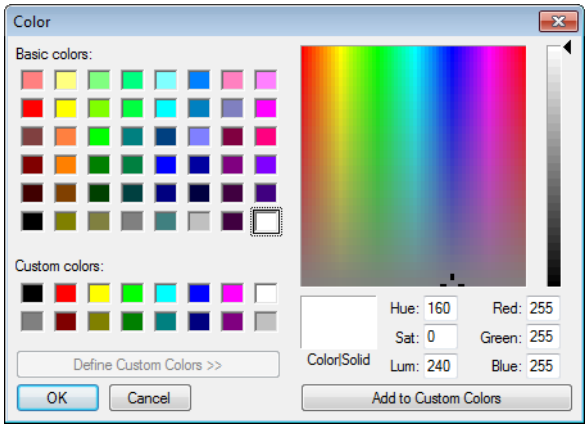
To select pre-defined base colors from a dialog or to create your own colors:

- in the detail view, click in the cell with the desired color or in Properties `Color` on ...
- the dialog for the selection of standard colors opens



- click on a standard color to allocate it

- to define your own color, click on the **Define Color** button



- define the color
- using the **Add Color** button, add it to **User-Defined Colors**
- select the desired color
- assign the color by clicking on **OK**

COPY COLOR

To copy several colors of the same color definition:

- highlight the hexadecimal code of the color in the input field of the desired color
- copy the hexadecimal code with `ctrl+C` or the **Copy** command in the context menu
- insert the hexadecimal code for all colors which should receive the same color definition with `ctrl+V` or the **Insert** command in the context menu

DELETE COLOR

Colors from palette are distinguished clearly by means of their index in the palettes (= row number in palette window). Therefore the colors can only be deleted at the end of the palette, because otherwise the indices of subsequent colors would change. Because all palettes in the project must have the same color, only complete rows can be deleted. To delete a color from all palettes:

- Highlight the rows to be deleted, starting with the lower end of the palette colors
- In the context menu, select **Delete Color** or press the `delete` key

13.4 Use palette in the Editor

SELECTING THE ACTIVE PALETTE IN THE EDITOR

In the editor, a palette can be used as the active palette for display and color selection from the local or global project.

To select a palette for display in the editor:

- Click on the table of color palettes with the right mouse button.
- select **Color palette in the editor** in the context menu
- select the desired palette from the drop-down list

Note: Palettes from the local project are arranged before those from the global project.

Attention: If there are palettes with the same name in the global and local projects, only the palettes from the local project are offered for selection.

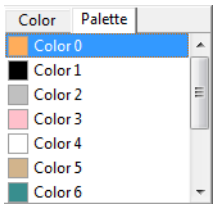
> **Attention**
>
> If there is a different number of colors configured per palette in the global and local project, there may be less colors available for the active palette selection!
>
> Palette definitions from the local project always have priority: If there is at least one color configured in the local project palettes, the amount of colors from the local palette are offered; otherwise the amount of a colors in the global project are offered.
>
> Colors missing from the active palette are displayed in white.

SELECTING THE PALETTE COLORS

Colors from the palette are assigned in the editor via:

- Dialog in properties
- Drag & drop the description of a color into the corresponding field in properties
- Direct input of the description of a color in the corresponding field in properties

If an element is assigned a color via properties, the color dialog opens with two tabs:



- Colors Selecting a particular color
- Color palettes: selecting a particular color palette from the active color palette

If there are not palettes with at least one color configured in the local or global project, or if the property does not allow a color to be assigned, only the tab colors are displayed.

COLORS FOR MULTI-USER PROJECTS

Changes to color palettes can only be enabled, transferred or discarded for all palettes of a project.

13.5 Create Runtime files

The palettes of the global project and local project are carried over in a joint list when creating Runtime files. There may be conflicts in doing so. Note:

- The local project always has priority.
- Each palette displays the unified amount and therefore contains as many colors as the largest palette.
- If the palette of the global project and the local project contain different amounts of colors, the shorter palettes are filled with `white`, so that all palettes have the same amounts of colors and a color is available in each palette for each color index. With palettes of the same name, the missing palettes of the larger palette are carried over

EXAMPLE:

- There are 3 palettes in the global project: **`Global 1`**, **`Global 2`** and **`Palette 1`** with 10 colors each.
- There are two colors in the local project: **`Palette 1`** and **`Palette 2`** with 5 colors each.

- In Runtime there is then:
  - **Palette 1:** comes from the local project and overwrites **Palette 1** from the global project; it contains the 5 colors from the local project plus the colors 6 - 10 from**Palette 1** from the global project
  - **Palette 2:** contains the 5 colors from the local project plus the colors 6 - 10 initialized as white
  - **Global 1: Contains the** 10 colors from the global project
  - **Global 2: Contains the** 10 colors from the global project

SWITCH COLORS IN RUNTIME

To switch colors in runtime, you need the Switch palette function (see chapter on switching color palettes (on page 266)).

You can find an example of switching in Runtime in the chapter on the example for editor und Runtime (on page 269).

13.6 Switch color palette function

When starting runtime, the first palette in the project automatically becomes active, so long as another palette is not activated via a start script.

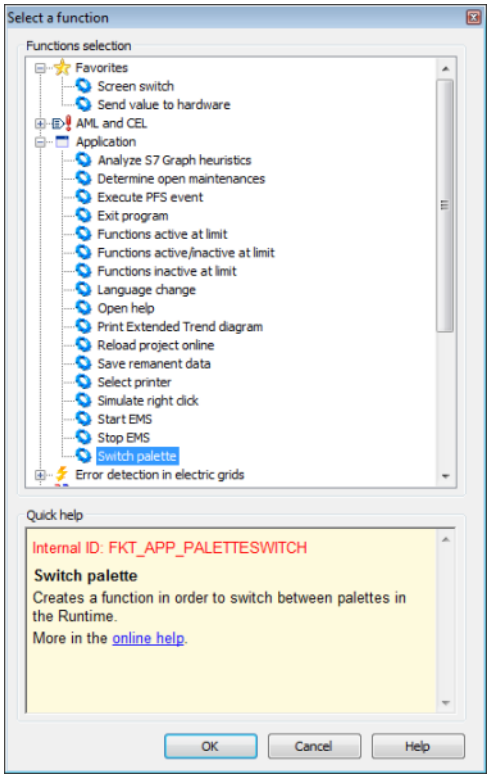
The active palette can be changed in runtime by means of the Switch palette function. All screens and elements are then relabeled with the colors of the new palette.

The respective active palette is defined via the name of the palette. If no palette exists with the given name, the first palette in the project is activated.

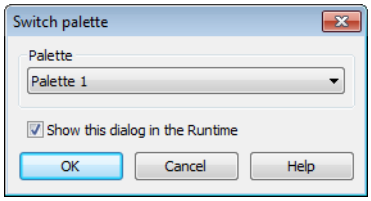
To create the Switch palette function:

- In the project manager, select the **New Function..** command in the **Functions** node.
- navigate to **Applications**

- select **Switch Color Palette**



- confirm with OK.
- The dialog to select the color palette is opened



Parameters	Description
Color palette	Select the desired color palette from the drop-down list.
Show this dialog in the Runtime	Enables a new color palette to be selected in runtime.

13.7 Export and import

Color palettes can be exported as XML files and imported from these.

EXPORT

To export a palette:

- select the desired palette
  to select several palettes use the `ctrl` or `shift` key or hold down the left mouse button and move the mouse over the title line of the table
  (**Note:** selected palettes can no longer be deselected individually)
- only complete selected palettes can be exported, not individual colors
- select **`Export selected XML...`** in the context menu or in the tool bar
- in the file manager that opens, select a location to save the file and a name for the XML file

IMPORT

To import a palette:

- select **`Import XML...`** in the context menu or in the tool bar
- select the file you wish to import in the file manager that opens
- The palettes saved in the XML file will be imported
- **Attention:** The sequence during exporting is taken into account:
  If, for example, a palette is exported to the third place in the table, this is imported at the same place or added at the end if the project has fewer palettes Palettes already present here are overwritten. That means:
  - If palettes are at the location that is imported to, the names and colors of the existing palettes are overwritten.
  - If a palette is imported to a free location, it is inserted at the end of the sequence and named with the consecutive standard names.
  - If the imported palette features more colors than those that already exist, the missing colors are supplemented with `white` in the existing palettes
  - If the imported palette has fewer colors than the existing one, the missing colors are accepted by the overwritten palette or supplemented with white, if no palette was overwritten.
  -

13.8 Error messages

When transferring elements, for example by importing XML files or copying & pasting, some palette colors may no longer be displayed correctly. Possible reasons:

- no color palettes have been created in the project yet
- Palettes have a different number of colors
- Colors were deleted

The corresponding error message is displayed in the properties input field, the element is displayed in `white`.

Error message	Reason
`No palette loaded!`	No palettes were configured in the local and global project.
`Palette color not available!`	The is no color in the project's palette with the corresponding index.

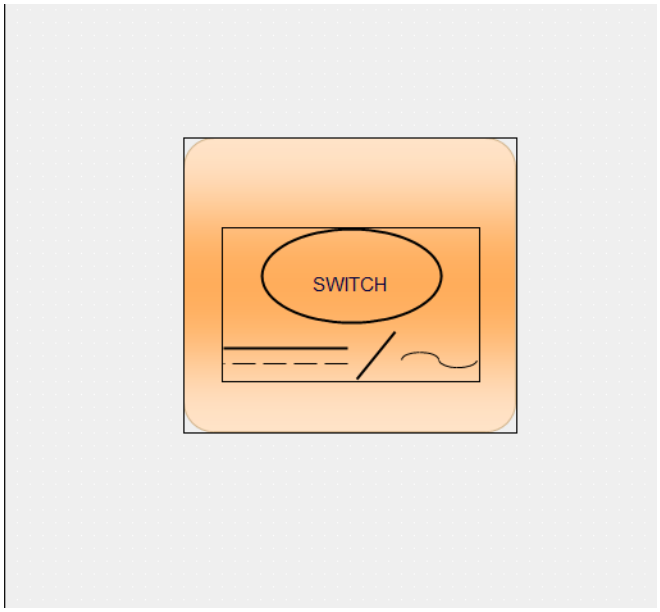
13.9 Example for Editor and Runtime

IN THE EDITOR

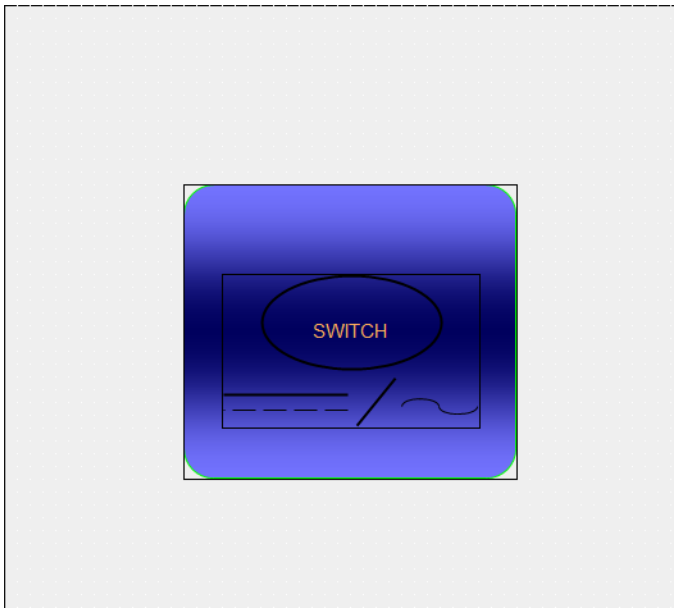
This is how you use color palettes in the editor for example:

- create a screen with a button with text and a graphic element
- choose `Background color` in properties
- click on ...
- the dialog for selecting colors opens
- select the **`Color palette`** tab
- the colors of the active color palette are displayed
- select the desired color

- repeat this step for the properties `Border color` and `Text color`
- `Screen with basic color palette`

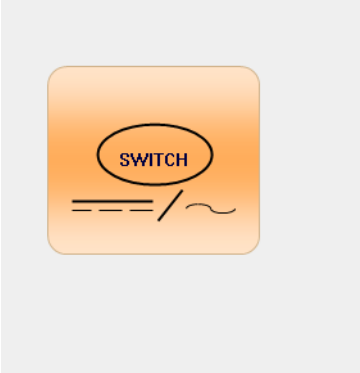


- In the color palette table, right-click in a cell
- select **`Color palette in the editor`** in the context menu
- in the drop down list, select a different palette, for example, `Palette 2`
- Screen in the editor:



IN RUNTIME

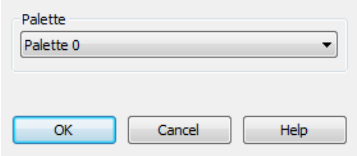
The screen with the color palette activated in the editor is displayed in Runtime.



If you have configured a switching function (see chapter on Color palette switching function (on page 266)), you can change the color palette in runtime.

To do this:

- Click on the element.
- If the **Offer this dialog in Runtime** option was activated, a new palette can be selected in Runtime; otherwise the palette given in the function is shown automatically



- the element is switched to the new colors

14. Fonts

In zenon all the fonts installed in the operating system can be used. The **Fonts** node is under the **Screens** node in the Project Manager.

PROJECT MANAGER CONTEXT MENU

Menu item	Action
Font list new	Creates a new font list in the detail view
Export XML all ...	Exports all entries as an XML file.
Import XML...	Imports entries from an XML file.
Help	Opens online help.

Info

The selected font must also be present on the target system with Runtime. If the font is not present on the target system, another font present on the system from the same family is used.

14.1 Font lists

By default, one font list exists. This is displayed in the detail view if the Screens / Fonts node is selected in the Project Manager.

TOOLBAR AND CONTEXT MENU

Parameters	Description
Font new	Creates a new font. The Windows dialog '**Font**' is opened, where the font type, style, size, etc. can be defined. See chapter Creating a new font (on page 275).
Delete font	Removes the selected font from the font list.
Font list new	Creates a new font list (same functionality as in the Project Manager). See chapter Creating a new font list (on page 274).
Delete font list	Deletes the selected font list.
Save	Saves the changes.
Font list in the Editor	Displays all existing font list, the active one is marked.
XML export selected	Fonts and font lists can be imported/exported. See chapter Import/Export / Screens / Fonts.

The name of the font list can be changed in properties.

> **Attention**
>
> *Name changes performed later on will not be integrated automatically. If you change the name of a font list after creating a function* ***Language Change*** *for this font list, you also have to change the name of the font list in that function.*

FONT LISTS: GLOBAL PROJECT VERSUS PROJECT

The Editor always has a font list actively switched. To use fonts from the standard project and the global project, the font lists in these two projects must be named the same.

Note when using font lists in global projects:

- If font lists are named the same in a global project and a standard project, the font list of the standard project takes priority in Runtime.
- If there are different names, two font lists are present in the standard project, either one of these is used in Runtime as desired.

To avoid this effect: Plan a language switching function by selecting the font list.

14.1.1 Creating a new font list

In order to link elements to fonts, which do not belong to Font list 1, another font list has to be created.

CREATING FURTHER FONT LISTS:

A new font list can be created in the Project Manager in the section Screens / Fonts with the entry **`Font list new`** from the context menu. The new font list is displayed in the detail view of the Project Manager.

The default font is also automatically created in the first position here.

ACTIVATING A FONT LIST:

To activate a font list, click in the header for the desired font list and select the entry **`Font list in the Editor`** from the context menu. This can be necessary for language switchable texts. So you can check already in the Editor, how the element with the font will look in the Runtime. (e.g. whether the font is too large for a button) See chapter Language change / Fonts: Language switch / Fonts

> **Info**
>
> *There is no default font list for Runtime. If several font lists are planned, the font list to be used in Runtime must be set using Language switching.*

14.2 Default font and system font

Every new element is linked to a default font. This is the font that is assigned to the element type. Changing the property `Font` of an element causes that every new element of that element type is connected to this new font. After the editor is restarted, the default font is used for all element types again. The default font is `Arial 10`, supports antialiasing and is on the second position of the list.

The following fonts are automatically created when a project is first created:

Font	Font style	Size	Info
Arial	Standard	8	
Arial	Standard	10	Default font
Arial	Standard	12	
Arial	Standard	14	
Arial	Standard	20	

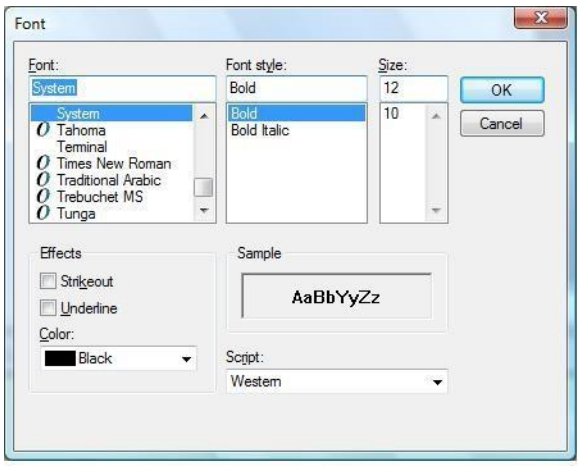
- The default font can be edited. The changes take effect everywhere where this font is linked.
- The system font is not scalable, which can lead to problems during zooming or when adjusting the resolution automatically.
- The default font can be changed via the `Font` property. The dialog is the same one that opens when **`New font`** is selected in the context menu.

14.2.1 Creating a new font

One font list exists by default, which is displayed in the detail view of the Project Manager and can be filled with different fonts.

CREATING A NEW FONT IN THE FONT LIST:

Click in the window of the detail view and open the context menu with a right mouse click. Select the entry '**Font new**' here.



The Windows dialog **Font** is opened. Here the following settings are available:

Parameters	Description
Font	All fonts of the operating system are sorted alphabetically
Font style	Bold, italic, etc.
Size	Font size in pixels
Effects	Underlined, strike-out
Color	Selection of the font color
Script	Western, Cyrillic, etc.

Info

The color settings are ignored. Text is displayed in black unless the color has been explicitly edited in the properties Text colors only change in the Runtime (e.g. because of limit violations).

For fonts in global projects refer to chapter: Project administration / global project

Attention

All fonts used must be installed on all computers on which the project will be started. The installation can also be carried out with Remote Transport. See Define Files (Copy&Register)

14.3 Naming and numbering fonts and font lists

When creating a zenon project, 5 default fonts (on page 274) are created. The name of the font list and the default font are dependent on the language in which the Editor was started. This applies for standard projects and global projects.

The fonts are given numbers. Number 1 and 2 are standard fonts. These two numbers cannot be changed, all other numbers can be adapted individually. When creating dynamic elements, this is automatically allocated to the font with number 2.

The Editor always has a font list actively switched. To use fonts from the standard project and the global project, the font lists in these two projects must be named the same.

Note: zenon projects that have been converted from version 5.50 into a later version have no default font list if the project has no fonts in 5.50.

GLOBAL AND LOCAL

Fonts from the local standard project have a higher priority than fonts from the global project.
For example: There is a font with the number 50 in the global project. This font can be used in the workspace in any standard project as long as the standard project does not have a font with the number 50. If there is also a number 50 in the standard project, this font is used.

The fonts with the numbers 1 and 2 cannot be deleted in the standard project. That means: The fonts with the number 1 and 2 can be changed in the global project. However, they cannot be used in the standard project, because these two numbers are always present in the standard project and have a higher priority. An offset can be used to differentiate fonts from the global project and the standard project. For example, font 2 from the standard project can become font 2002 in the global project.

Workaround to use fonts 1 and 2 from the global project in the standard project: Delete the font list in the standard project and name a new empty font list the same as the font list in the global project.

DELETE FONT LIST

If the fonts with the numbers 1 and 2 from the global project are to be used in the standard project, the font list must be deleted in the standard project. To do this:

1. This is how you create a project backup:
2. If adapted fonts from a standard project are to be used globally:
   a) Carry out an XML export of the font list from the standard project
   b) Carry out an XML import of the font list into the global project
3. Open the SQL folder for the standard project

   Hint: Highlight the project in zenon -> `press Ctrl+Alt+E` -> Explorer opens the project folder -> navigate on to `zenon/system`
4. close the workspace
5. Delete the file named **`fonts.bin`**
6. open the workspace
7. Now name an empty font list with the same name as the font list in the global project

Now the fonts with the numbers 1 and 2 from the global project are used in the standard project. New elements use the font with number 2 from the global project, if this is available, and the system font if the global project is not available.

DYNAMIC AND STATIC

Fonts for the **`Static text`** vector element and for dynamic elements are saved differently. The reason for this: **`Static text`** can also embed fonts.

`Embedded` means: The font is not linked to a font list, but is instead local and only configured for this element.

Attention: This does not mean that the font is embedded in the screen and is thus always displayed correctly. The font must continue to be present in the system that is displaying it.

IMPORT AND EXPORT VIA XML

When importing and exporting via XML, the `Name` property is used for identification with dynamic elements. If for example, in the global project there is a font with the name "`zenon`" and in the local

standard project there is another font that also has the name "zenon", then the font of the local project is used during import and export.

Hint: The font names must be clear in the global project and standard project. The Editor does not check this.

VBA

VBA uses the Number property of the font for dynamic elements. Here, the offset can be useful for fonts in the global project and in the standard project.

For example: Font 2 from the standard project is replaced by 2002 from the global project. For vector elements, the font number is saved in the second position of the font string. In the font string "0 7 0 0 700 255 0 0 0 3 2 5 34 Arial", the number of the font is7 and can easily be replaced by 2007 from the global project.

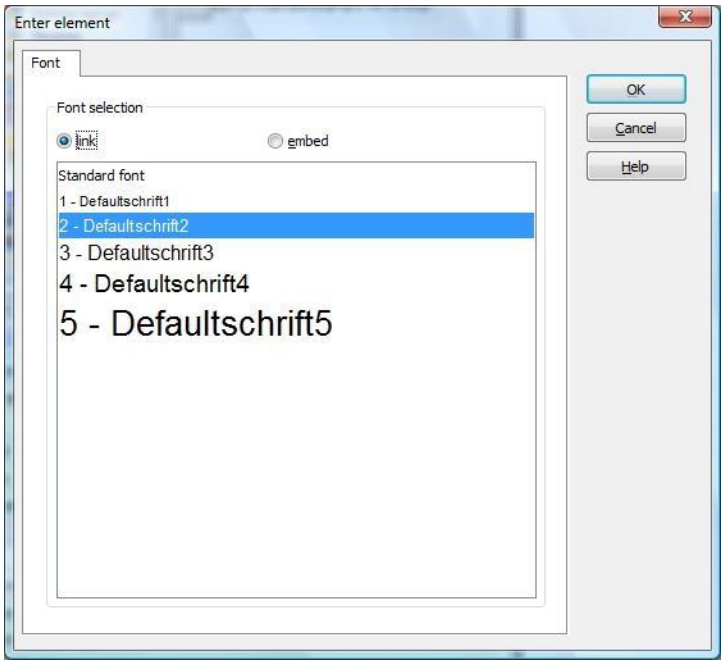
14.4 Linking fonts to elements

There are two possibilities to link a font to a dynamic or vector element (e.g: Text (on page 116)).

Parameters	Description
Drag&Drop	The table in the detail view has the following structure: Numbers, font lists and fonts After clicking the number of the desired font, it can be linked to the element with Drag&Drop.
Properties	It is also possible to link the font in the properties of the selected element. Under 'Representation/Font' the dialog '**Enter element**' is opened.

14.4.1 Embedded and linked fonts

Fonts can either be `embedded` in elements or `linked` to them.



Parameters	Description
`link`	Fonts are linked by default.
`embed`	For the **`static text`** vector element, there is the possibility of embedding the font. The font is then no longer linked to a font list, but is instead local and only configured for this element. **Attention:** This does not mean that the font is embedded in the screen and is thus always displayed correctly. The font must continue to be present in the system that is displaying it.

Info

Scaling of symbols:

If the symbol contains a ***static text*** *element and is embedded in this element as text, then this text:*

- Is scaled in the Editor
- Is not scaled in Runtime

This only applies to symbols. Embedded text is neither scaled in the Editor nor in Runtime in all arrangements.

14.5 ClearType

ClearType fonts can improve the legibility of fonts on LCD screens in native resolution. Windows standard fonts are often better suited to CRT monitors and LCDs with an interpolated resolution. You can define whether ClearType fonts are used in the project properties: Nodes `Graphical design` -> `Screens` -> `ClearType fonts`.

OPERATING SYSTEM DEPENDENT

The effects that the option has also depends on the operating system used:

- Windows Vista or Windows 7:
  - If ClearType is active in the operating system, ClearType is used regardless of the setting in zenon.
    Exception: DirectX can switch Cleartype on or off directly.
  - If ClearType is not active in the operating system, the setting in zenon is decisive.
- Windows XP:
  - The setting in zenon decides.

Setting	Settings of `Graphics quality` property	Description
`Active` (default)		ClearType fonts are used.

`Inactive`	`Windows basic`
	
	`Windows extended`
		Windows standard fonts are used.
		Clear Type must also be deactivated in the operating system with Windows Vista and Windows 7.

	`DirectX software`
	
	`DirectX hardware`
		Cleartype can be switched on or off regardless of the operating system

Attention: Changes in the setting only become effective after the Editor is restarted.

15. Symbols

Symbols are screen elements to which vector and dynamic elements are assigned as screen macros. The color attribute of a symbol can be changed by means of dynamic elements (binary element, multi-binary element and combined element) in Runtime.

zenon distinguishes between two symbol libraries:

Symbol library	Property
`Global system library`	Symbols are available in all projects. The **`Global symbol library`** node is in the project manager below the currently-loaded projects Global symbols are saved in the zenon program folder and only updated when the Editor starts. These symbols are not saved during project backup.
`Project Symbol Library`	Symbols are only available in the current project. The symbols are saved in the project folder. The project symbol library is in the current project in the **`Screens`** node and is backed up together with project backup.

In order to be able to work with both libraries at the same time, the previews of the global library and the project library are displayed in two separate windows. In doing so, the names of the libraries are displayed in the tabs at the lower edge of the screen.

Symbols can be moved or copied from one library into the other with Drag&Drop.

Info

Scaling of symbols:

If the symbol contains a ***static text*** *element and is embedded in this element as text, then this text:*

- Is scaled in the Editor
- Is not scaled in Runtime

This only applies to symbols. Embedded text is neither scaled in the Editor nor in Runtime in all arrangements.

SCREEN SYMBOLS AND SYMBOLS FROM THE GLOBAL SYMBOL LIBRARY

- Screen symbols are created in the screen and can only be edited there. They are linked in the library or embedded in screens.
- Symbols from a library are edited in the library and linked in the screen.

Attention

Symbols from the ***Global symbol library*** *are saved locally in their own file (*.sys) in the zenon program folder. These files must always be added manually to back-up the project with multi-user projects and when forwarding projects. Symbols from the local project symbol library are always contained automatically in project backups.*

Hint: *Always save symbols in the local project symbol library.*

PROJECT MANAGER CONTEXT MENU

Menu item	Action
Export XML all..	Exports all entries as an XML file.
Import XML	Imports from an XML file.
Open in new window	Opens a new window to view and edit the variable. (Default: at the bottom of the Editor.)
Help	Opens online help.

CHECKING LINKED SYMBOLS

Linked symbols are checked to see if they are present and correct:

- when the Runtime is compiled
- when the Editor is closed and new Runtime files are created

Missing symbols are marked red in the output window. The error message shows the name of the affected screens and symbols.

MENU SYMBOLS

As soon as a symbol is activated in the global symbol library or in the project symbol library, the Screens menu item is replaced by Symbol in the toolbar.

Possible actions are:

Menu item	Action
Save symbol	Saves the current symbol.
Save all symbols	Saves all symbols.
Change symbol size / background color	Opens the dialog for configuration (on page 288) of symbol size and background color. **Note:** Changes to the background color always affect all symbols in the global system library.
List of symbols	Opens a list of all symbols, used in the current screen. For each symbol its graphical components are listed.
Full screen mode	The full-screen hides all remaining windows and shows only the main window with all opened documents (screens, reports, ...) in full-screen mode. The full-screen mode can be closed by: ▶ the displayed button for closing the full-screen mode ▶ the short key Shift+F9 ▶ the key Esc
Redraw symbol	Refreshes the display.
Display grid	Switches the grid on or off.
Use grid	Switches usage of the grid on or off. On: All objects are automatically aligned with the grid.
Use magnetic points	Switches magnetic points on or off. Use: If you move an object near the handling point of another object, it snaps in place at the handling point.
Change order for focus	Opens drop-down list for definition of horizontal or vertical order for Keyboard operation. Order for left/right: Determines the horizontal order. Order for up/down: Determines the vertical order.

Zooming	Changes the resolution in fixed steps between 15% and 400%.
Screenshot print current symbol	The current symbol is printed on the default printer. Set printer properties via *File -> Standard configuration -> standard.*
Insert vector graphics	Opens the dialog for inserting an external vector graphic.

15.1 Symbol detail view toolbar and context menu

The symbol library can be operated from the context menu or the toolbar:

COMMANDS IN THE CONTEXT MENU AND TOOLBAR

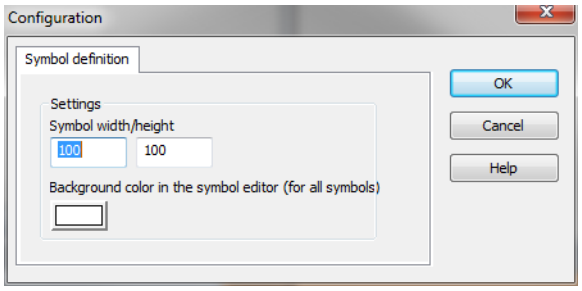
Toolbar	Context menu	Function
1	**Symbol new**	Creates a new symbol.
3	**Open symbol**	Opens the symbol in the symbol editor for editing purposes.
2	**Symbol group new**	A new symbol group is created and can be filled with selected symbols from the library or with self-defined symbols. (Copy and Paste). Only available in the global symbol library.
4	**Insert symbol as embedded symbol in the screen**	Embeds the symbol in a screen (for details, see subchapter Embedded symbols (on page 296)).
5	**Insert symbol as linked symbol in the screen**	Links the symbol in a screen (for details, see subchapter Linked symbols (on page 297)).

13	**Flat view**	Arranges all symbols in a flat view. Facilitates the search for certain symbols. **Hint:** In this view, you can´t create nor delete symbols and it is also impossible to copy or insert symbols from the clipboard.
14	**Hierarchical view**	Sorts all symbols hierarchical in subgroups and shows dependencies.
6	**Copy**	Copies the selected symbol to the clipboard (only hierarchical view).
7	**Paste**	Inserts the copied symbol in the active group (only hierarchical view).
8	**Delete**	Deletes the selected symbol or symbol group (only hierarchical view). **Attention:** Deletion cannot be undone!
9	**Edit selected cell**	Enables editing of the selected cell´s contents.
10	**Replace text in selected cell**	Opens the dialog for automatic replacement of texts (only available if cells that can contain text were selected).
11	**Properties**	Opens the window **Properties** for the selected symbol.
12	**Expand/reduce nodes**	Allows expansion/reduction of the selected node or all nodes in the hierarchical view.
15	**Export selected XML**	Opens the dialog for XML-export of the selected symbols. (For details, see chapter Import/Export of symbols.)
16	**Import XML**	Imports symbols from an XML-file (see chapter Import/Export of symbols).
18	**Remove all filters**	Deletes all filters in the column title.
19	**Help**	Opens the help file for the context menu.
17	-	Jumps back to the starting element (only toolbar).
	Symbol bar options	Clicking on the arrow opens the submenu: Active: Toolbar is displayed. If the toolbar is not displayed, it can be activated using the Menu *Options -> Toolbar*.

15.2 Configure size and background color

To configure the symbol size and background color:

1. Select the symbol in the context menu or in the, in the **Symbol** menu, the **Change symbol size/background color ...** command
2. The dialog for configuration is opened



Parameters	Description
Symbol width/height	Input of the height and width of the symbol in pixels.
Background color	Defines background color. Clicking on Color opens the palette. **Note:** Changes to the background color always affect all symbols in the global system library.
OK	Applies changes and closes dialog.
Cancel	Discards changes and closes dialog.

15.3 Release properties from link

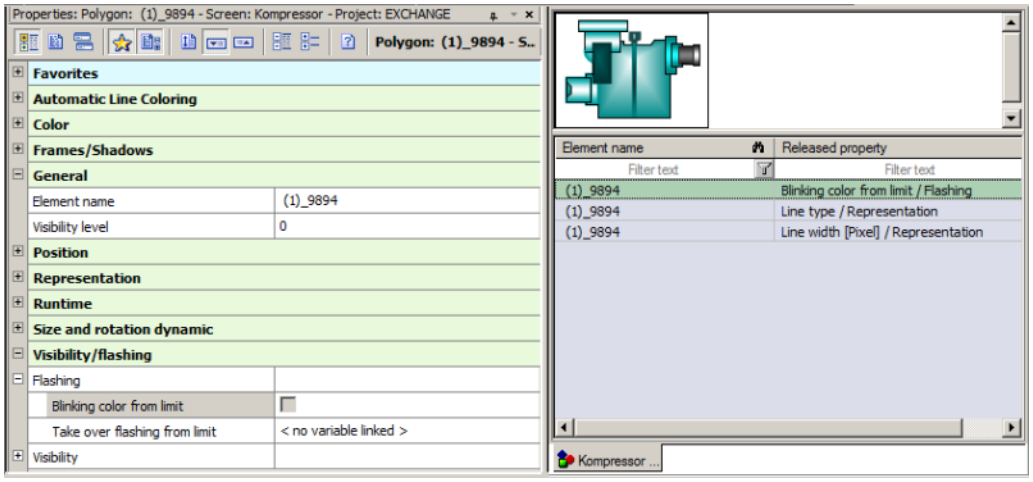
For optimum use of linked symbols, it is sometimes necessary to assign individual values to individual properties from system elements. For example, a motor symbol has a heading in which the name is contained.

You can release the Text property with this function. If the symbol is dragged onto the screen, the Text property can be changed individually for each motor.

This function is available for all properties.

CONFIGURATION

The symbol editor is opened when a symbol is opened. The symbol editor consists of a window divided into two parts to edit the selected symbol.



The symbol is edited in the upper area; in the lower area, the released properties of the element of this symbol are listed. You have different possibilities to release elements:

1. Drag&Drop
   a) Move the desired properties by dragging & dropping from the property window in the lower area of the symbol editor.
   b) **Note:** You must always click on the name of the property (not the value) and move it into the release area by dragging & dropping.
2. Property context menu
   a) Right-click on the name of the property.
   b) Select `Release [property]` in the context menu
   c) The property is entered in the list
3. Property toolbar
   a) Left-click on the name of the property.
   b) Click in the toolbar on the release symbol (green arrow upwards above text field)
   c) The property is entered in the list

Note: If the entry is not offered or the symbol is displayed as deactivated, the element cannot be released. Possible reasons:

- No screen element selected in the symbol editor

- Property was already released
- Element does not support release (ActiveX, for example)

The release area is a list that can be filtered. You can filter according to individual properties as needed. A context menu is available for all actions.

CONTEXT MENU FOR RELEASE LIST:

Parameters	Description
`Remove selected property`	Removes release of the element.
`Undo`	Revises last action.
`Help`	Opens online help.

EDITING RELEASED PROPERTIES

If the symbol is displayed in a screen, the additional node **`$_Element Name`** is available. All properties that have been released individually have been released. These are separated from the linking principle (released) and only concern this symbol if changed. If a property of one of the elements is selected in the **`Properties`** window, this element flashes in the screen.

The released properties can be changed:

- With the properties in the **`$_element_name`** branch
- with the Linked Symbol Wizard

LINKED SYMBOL WIZARD

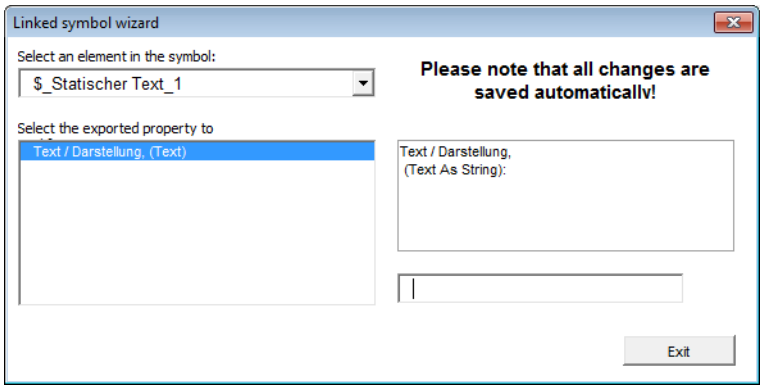
The linked symbol wizard is available to you in order to edit released properties easily and with a clear overview:

1. Double click on a symbol that:
   a) Is linked in the screen and
   b) has released properties
2. The Linked Symbol Wizard opens
3. Select the property to be changed

4. Enter the new value

   Hint: Each change is accepted immediately!

5. Close the wizard by clicking on **Exit**



Parameters	Description
Select an element in the symbol:	Selection of an element in the symbol from the drop-down list.
Select the exported property to	Selection of the released property from a list.
Display of property	Display the property to be changed-
Input field	Input of the new value of the property by means of text entry/number entry, drop-down list or selection dialog - depending on the property.
Exit	Closes wizard

Info

The wizard does not support all possible symbol properties.

15.4 Create and delete your own symbols

Create symbols either in the symbol editor or in a screen. Elements of a symbol can be:

- Vector Elements

- Dynamic elements
- other symbols: Symbols can be inserted into other embedded symbols

> **Info**
>
> *Elements contained in a symbol should only be placed within the symbol area and not jut out. If elements jut out over the symbol borders, artifacts may occur at the display in the Runtime. This is also true for elements using symbols (e.g. combi element).*

IN THE SYMBOL EDITOR:

1. select **`New symbol`** in the context menu or in the tool bar
2. Click in the **`Element Name`** cell and assign a name (periods are not permitted in symbol names)
3. Create the symbol in symbol editor
4. add any individual properties (see sub-chapter on releasing inherited properties (on page 288))
5. save the symbol

ON THE SCREEN:

1. select one or more element on the screen
2. In the context menu, select *Symbol -> Create Embedded Symbol* (or *-> Insert into Existing embedded Symbol*)
   or in the menu *Edit-> Symbol -> Create Embedded Symbol*
3. save the symbol

> **Attention**
>
> Fonts and graphics in symbols for use throughout the project:
>
> `Fonts`: fonts are defined through their ID. Ensure that the same ID is assigned to the desired font in all projects in which the symbol is used.
>
> `Graphics`: Graphics embedded into a symbol are not saved with the symbol. The corresponding graphics must be copied manually to the graphics library of the corresponding project.

EMBED SYMBOLS INTO SYMBOLS OR LINK SYMBOLS TO SYMBOLS

You can also embed or link existing symbols when creating symbols in a similar manner to the screens (see Insert symbol into screen (on page 295)). In doing so, you must heed the following rules:

- A symbol may not contain any recursion i.e. it may not link to itself!
- Project symbols may not be linked across projects!
- Symbols from the global library may not contain any links to project symbols.

> **Info**
>
> *You can only edit the contents of symbols in the symbol editor. If you insert an embedded symbol into a screen, you cannot edit the symbol in the screen editor.*

> **Attention**
>
> *You can only link symbols from the project library within the respective project. It is not possible to link symbols from the project library across projects. To use a symbol from another project library in another project, you can embed it.*

DELETE OR REMOVE

To delete a symbol:

1. highlight the symbol in the library

2. select **Delete** in the context menu or in the tool bar

To remove an embedded symbol in a new screen:

1. highlight the symbol in the library
2. In the context menu, select *Symbol -> Remove*
3. the symbol is removed in its elements
4. if the symbol for the symbol library has already been inserted, it remains there unchanged; it is deleted from symbol administration however

To delete a linked symbol:

1. highlight the symbol in the library
2. select *Delete* in the context menu.
3. the symbol remains unchanged in the symbol library, however it is deleted from symbol administration

> **Info**
>
> *To edit individual elements of a symbol without removing the symbol assignment, select:*
>
> - for linked symbols: **Edit in symbol editor**
> - for embedded symbols: **Single edit mode symbols / elements**

15.4.1 Transfer own symbols to the library

You can also transfer symbols that you have created on the screen into a symbol library:

1. In the context menu, select the symbol *Symbol -> Insert into symbol library*
2. the dialog for symbol definition opens

3. select your project library or the global symbol library and the appropriate folder
4. confirm the dialog with **OK**.
5. change to the symbol library selected
6. the symbol was entered with the element name **Embedded symbol_No**
7. name it as you wish
8. To continue editing the symbol or to allocate it individual properties(see Release Inherited Properties), (on page 288) open the symbol by double-clicking on it or via the **Open Symbol** context command in the symbol editor

> **Info**
>
> *A symbol that was created from an element and a linked symbol cannot be accepted into the symbol library. To be able to insert the symbol:*
>
> - Convert the linked symbol into an embedded symbol
> - Create a new symbol from the element and the embedded symbol
>
> This symbol can now be inserted into the symbol library.

15.5 Add symbols to screen

Symbols can be inserted into a screen in two ways, as

- Embedded Symbols (on page 296):These are no longer connected to the symbol in the library, changes to the original symbol have no effect on the embedded symbol.
- Linked Symbols (on page 297): These remain linked to the original symbol. Changes to the symbol also change the symbol in the screen.

> **Attention**
>
> *You can only link symbols from the project library within the respective project. It is not possible to link symbols from the project library across projects. To use a symbol from another project library in another project, you can embed it.*

Info

If you have activated the `Frame around dynamic elements in the Editor` *property, you can distinguish between embedded symbols and linked symbols at a glance. An arrow symbol is displayed in the upper right corner for linked symbols. You find this property under Options -> Settings in the* **`Settings`** *area.*

15.5.1 Embedded symbols

This is how you embed a symbol into a screen:

- hold down the `ctrl` key and move it by dragging & dropping

or

- select the **`Insert symbol as embedded symbol`** symbol in the context menu to embed the symbol into the current screen

Embedded symbols can then be resolved to their individual elements or their elements can be edited. To do this, select *Symbol -> Change to individual editing mode* in the context menu or *Edit-> Symbol -> Symbol/element individual editing mode* in the menu.

These symbols do not change, if the original symbol in the library is changed, as they are no longer linked to it.

REPLACE LINKED VARIABLES

The dialog to replace variables and functions opens when embedding a symbol. Here, you can automatically replace the elements present with those you want for the linked symbol. You can find details on automatic replacement of variables and functions in the chapter on replacing variables and functions in dynamic elements (on page 225) .

To replace variables in symbols that have already been inserted, select the **`Replace Links`** command in the context menu.

Hint: Ensure that the variables are named (on page 226) clearly and sensibly from the start.

15.5.2 Linked symbols

Linked symbols do not save variables that are physically present in the symbol library, only variable names. When inserting variables into a screen as a link, the variable names saved in the symbol can be assigned to variables from the project. If the project is given a variable that corresponds to a variable name saved in the symbol, this is linked automatically - without an extra rule.

INSERTING A SYMBOL WITH LINKING

This is how you link a symbol to a screen:

- move the symbol by dragging & dropping it into the screen

or

- select the Insert symbol as linked symbol symbol in the context menu to link the symbol into the current screen

Linked symbols are only saved in the screen as links. Each change of the symbol in the library has a direct effect on the symbol on the screen. For example, if a dynamic element is added to a symbol, it is available in all screens in which the symbol is used.
Exception: Changes to the symbol size only have an effect on the screen of the symbol Original Size is selected in symbol properties `Graphic size`.

With embedded symbols, you can:

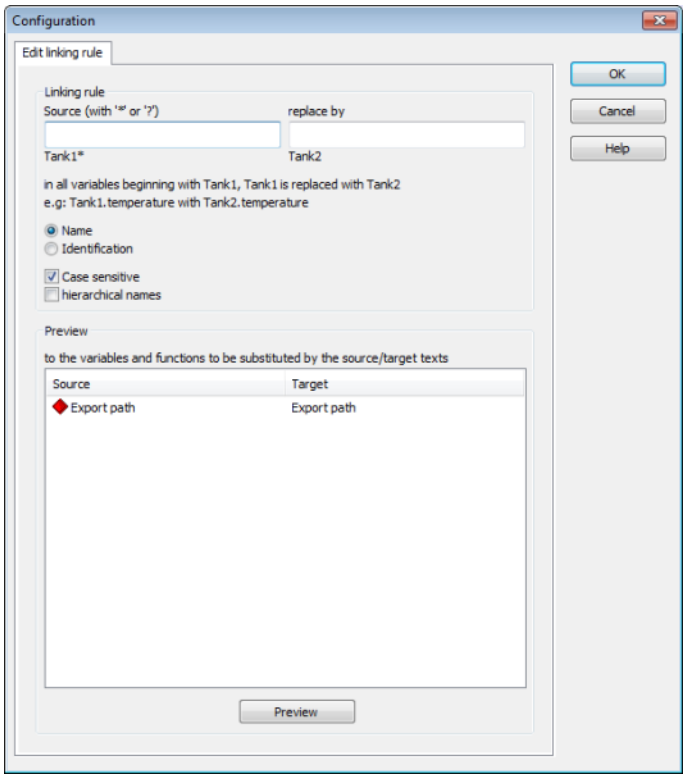
- turn them into embedded symbols: Context menu Symbol -> Convert Linked Symbol to Embedded Symbol
- Edit them in symbol editor: Symbol -> Edit in Symbol Editor
- insert them into an embedded symbol: Symbol -> Insert into Existing Embedded Symbol

Linked symbols cannot be resolved in their elements and their elements cannot be changed in `Edit elements` (`context menu ->` *Symbol -> Change to individual editing mode* mode or the *Edit -> Symbol -> Symbol/element individual editing mode* menu).

> **Attention**
>
> *With linked symbols, use only embedded fonts, because linked fonts are not scaled with the symbol. The position of the text would move in relation to the graphics in the symbol.*

REPLACE LINKED VARIABLES

When embedding a symbol that contains variables, the dialog to replace variables and functions opens. Here, you can automatically replace the elements present with those you want for the linked symbol. ALC aliases that are die used in elements in the symbol can also be replaced. Several entries are separated by a semicolon (;).



Property	Description
Source	Entry of the partial string to be searched; wildcards * and ? are possible. (Wildcards are only allowed as prefix or suffix; e.g. *xxx or xxx*.)
Target	Entry of the partial string
Name	Swaps information in process variable names.
Identification	Exchanges information in the identification
Note capitalization	When swapping, be sure that any capitalization is an exact match.
Combined names	Permitted combined names. Is not available for ALC aliases. These are always used without a symbol prefix.

You can find further details on automatic replacement of variables and functions in the chapter on replacing linking of variables and functions (on page 225) .

To replace variables in previously-inserted symbols, select the Preview property in the Linking rule node.

Hint: Ensure that the variables are named (on page 226) clearly and sensibly from the start.

EXAMPLE:

Source:A;b

Target:C;d

COMBINED NAMES

If the Hierarchical names property in the Linking rule node is active, the names of the linked variables from the object names of the symbol and the variables defined in the symbol are combined. The two names are separated by a dot: "**Symbolname.Variable**"

Example

Variable name: ***Motor.Speed***
*The symbol object name is "****Motor****". The dynamic element that is used in the symbol has the placeholder "****Speed****" as a linked variable.*
*The placeholder "****Speed****" can be easily typed in for the symbol property when creating the variable.*

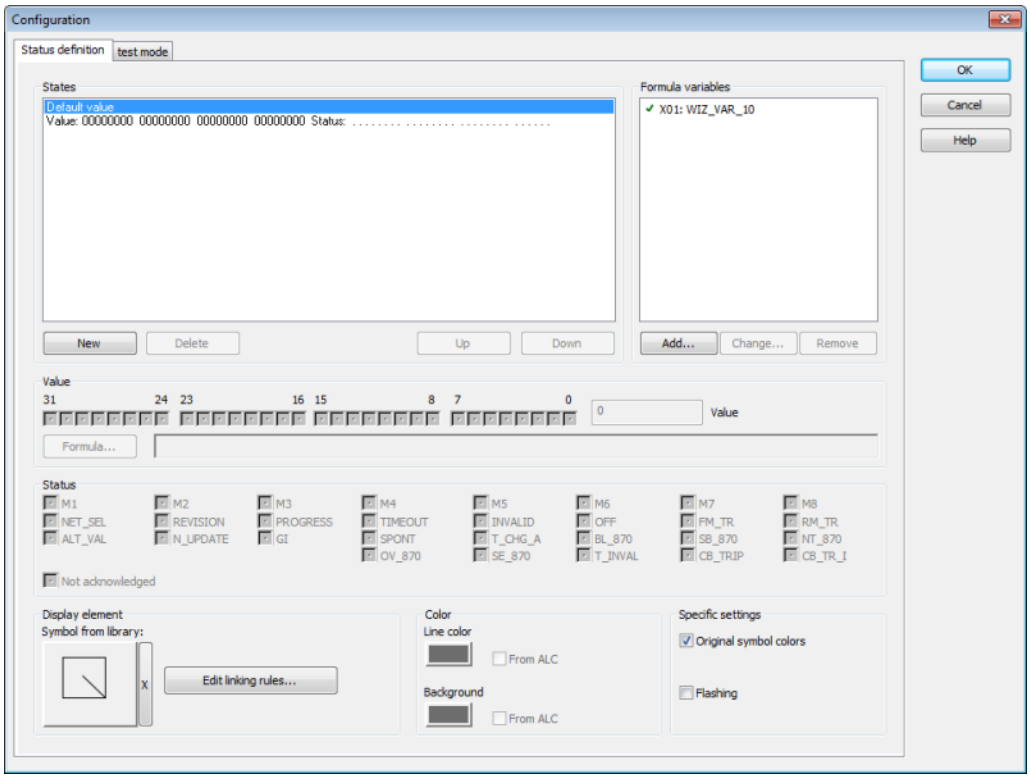
Combined names also work with nested symbols:
*If you have a "****Pump****" symbol, which in turn contains a "****Motor****" symbol and a "****Convertor****" symbol with the variable placeholder "****Output****", then the linking works if there is a* ***Pump.Motor.Output*** *variable and a* ***Pump.Convertor.Output*** *variable.*

15.5.3 Symbol selection in the Combined element

To link combined elements with a symbol:

1. select **Symbol from library** in the combined element properties in the Representation node in the Type of display property
2. click onConfiguration and test
3. the configuration dialog opens:

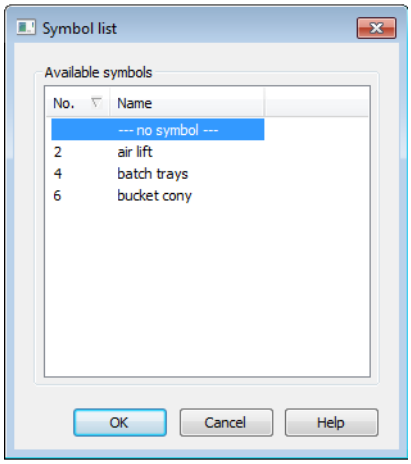
The general settings for all display types can be found in the chapter on statuses (on page 63).



Parameters	Description
Display element	
Symbol from Library	Clicking on the symbol opens the dialog to select a symbol. Clicking on the X next to the symbol deletes the selected symbol.
Edit linking rule	Variables and functions contained in the symbol can be exchanged with the help of linking rules. **Attention:** If a variable or function does not fit into the scheme, that link cannot be replaced.
Colors	Only active if the Original Symbol Colors property in Specific Settings is turned off.
Color	Choice of color.
Filling color	Choice of filling color.
From ALC	The color of the linked source is used.
Specific settings	
Original Symbol Colors	Active: Transfers original color from symbol. Deactivates Take color of main variable property.
Flashing	Active: Symbol flashes in runtime if status has been achieved.

15.6 Symbol list

You open a list that displays all symbols of the active screen via *Edit-> Symbol -> Toolbar...* :



Title	Contents
no.	Symbol number
Name	Name of the symbol

You sort the symbols according to number or name by clicking on the respective column title. A symbol selected in the list of symbols is blinking in the screen.

15.7 XML export and XML import

Import/export of symbols is possible from the project and the global symbol library. Single or selected symbols or the entire library can be imported or exported. The path for the export can be defined freely.

EXPORT SYMBOLS

To export symbols:

1. highlight the desired symbols
2. select **Export selected XML...** in the context menu of the detail view
   Alternatively: select, in the context menu of the **Project symbol library** node, the **XML export all...** command

3. the selection dialog for the saving location is opened
4. select the desired saving location
5. assign a name
6. pay attention to file type XML
7. confirm by clicking on the Save button

The selected symbols are exported to an XML file and can be imported in this or in other projects at any time.

IMPORT SYMBOLS

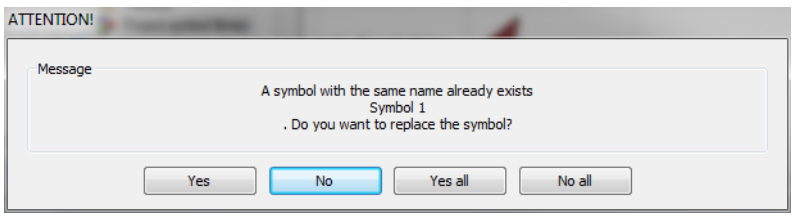
To import symbols:

1. in the context menu of the node or the detail view of the **Project symbol library** or in the detail view of the **Global symbol library** select **Import XML...** command
2. the selection dialog for the saving location is opened
3. navigate to the saving location of the desired XML file
4. Select the desired file
5. confirm the import by clicking on the **Open** button

The symbols are imported.

NAME CONFLICTS DURING THE IMPORT

At the import of a symbols the name (attribute **ShortName**) which is stored in the XML file is used for the name of the symbol which is created. If a symbol with the same name already exist, the import is halted and an error message is displayed:



Parameters	Description
Yes	The symbol described in the error message is replaced by the symbol from the XML file at the import. If a new conflict occurs, the error message is displayed again.
No	The symbol described in the error message is not imported. The existing symbol is maintained. If a new conflict occurs, the error message is displayed again.
Yes, all	The symbol described in the error message is replaced by the symbol from the XML file at the import. This setting is used automatically when errors due to using the same name occur. All affected symbols are replaced.
No, all	The symbol described in the error message is not imported. The existing symbol is maintained. This setting is used automatically when errors due to using the same name occur. All affected symbols are maintained.

Note: Only the **ShortName** is used. The file name does not matter for the name of the symbol which should be imported. Thus it cannot be used to solve the conflict.

16. Visibility levels

Similar to CAD programs, you can define visibility levels in the Editor and allocate them with unique elements. That way you can easily hide unused elements in complex images and show them again anytime. 15 visibility levels are available. Levels are allocated using the properties `Visibility level` in the node`General`. Unique levels can be activated and deactivated using the buttons of the toolbar **Visibility levels**:

Symbol (from left to right)	Function
`Show all visibility levels`	Shows all levels.
`Hide all visibility levels`	Hides all levels.
`Visibility level<No.>`	Shows / hides the relevant visibility level.
`Symbol bar options`	Clicking on the arrow opens the submenu: `Active`: Toolbar is displayed. If the toolbar is not displayed, it can be activated using the Menu *Options -> Toolbar*.

Info

Visibility levels only apply to the Editor and have no effect in Runtime.

Note on control elements:*Control elements are always displayed at the uppermost position. That also applies if they are covered by other elements in the Editor. This ensures that they can always be reached.*

17. Screen types

Screens are created in the basic setting with the `Standard` screen type. A drop-down list can be used to select special screen types for the properties in the screen properties.

Attention

If a type is changed from a standard type to a special type, the new screen type is fixed with the screen and can no longer be changed.

SPECIAL SCREEN TYPES

zenon offers a range of special screen types (on page 307) for special tasks. After selecting a special screen type, all user elements for the respective screen type are available in the **Control elements** menu. Menu item **add template** (on page 310) opens a selection dialog for adding pre-defined layouts with certain control elements at pre-defined locations.

> **Info**
>
> *Special screen types need only be created once, even when used multiple times. They can be opened with different data by means of the screen switching function.*

SPECIAL SCREEN TYPES AS A START SCREEN

Each screen type can also be used as a start screen in theory.

Attention: Special screen types are opened with an empty filter and thus no data are opened.

17.1 Special screen types

Special screen types in zenon:

Screen type	Description
Alarm Message List	Collects alarm messages and displays them in list form.
Alarm Message List Filter	filter for the Alarm Message List.
Archive revision	Display of saved archive values in a table and editing of these **ATTENTION:** Only available if the Historian is licensed.
Batch	Creating defaults for master recipe in module Batch Control. **ATTENTION:** Only available if Batch Control is licensed.
Command input	Display of command inputs. **ATTENTION:** Only available if the Energy Edition is licensed.
Chronological Event List	Collects process events, system events and user defaults and displays them in list form.
Chronological Event List Filter	Filter for the Chronological Event List
Energy Management System	planning and display of energy use management (electricity, gas). **ATTENTION:** Only available if the Energy Management System is licensed. **Note:** Not available under Windows CE.
Extended Trend	Displays online and archive values as diagrams.
Peak Load Projection	Screen for the display of Peak Load Projection **ATTENTION:** only available if SICAM 230 is licensed.
HTML (on page 349)	Makes it possible to display a screen in HTML, customized for Microsoft Internet Explorer.
Industrial Maintenance Manager	Display of maintenance information. **ATTENTION:** Only available if the Industrial Maintenance Manager is licensed. **Note:** Not available under Windows CE.
Industrial Performance Analyzer	Analyzes alarms in order to localize weak points (downtime) of equipment. **Note:** Not available under Windows CE.
Keyboard	Creates a freely definable virtual keyboard.

Login	Individual design of a login window.
Message Control	Screen for managing the message queu in Message Control.
Notebook (on page 352)	Displays protocols and text files which were created from database requests or existing text files.
Production & Facility Scheduler	Administrates schedules and profiles. **ATTENTION:** Only available if the Production & Facility Scheduler (PFS) is licensed.
Report Generator	For the output of reports that are produced by means of the extended Report Generator. **ATTENTION:** Only available if the Report Generator is licensed. If you only want to display reports but not edit them, you do not need a license for Runtime.
Report Viewer	Serves to display RDL reports for Archive, AML, CEL and online values. **ATTENTION:** only available if reporting is licensed.
Recipegroup Manager	Administrates recipes and their use within a recipe group. **ATTENTION:** Only available if the Recipegroup Manager (RGM) is licensed.
S7 graph	is used for sequential chains with alternative or parallel step sequences (SFC). Displays the current status of chain control in the PLC in runtime.
Standard recipes	Administrates recipes
Variable diagnosis	Displays variables in Runtime and sends set values to the hardware **Note:** Not available under Windows CE.
Video (on page 355)	Displays online and saved videos (*.avi). **Note:** Not available under Windows CE.
Worldview Overview	Makes it possible to scroll standard screens which are larger than the screen resolution.
Time filter	Enables general filter settings to be set for lists. The lists that are controlled by this are defined in the screen switching function.

17.1.1 Create special screen types

To create a special screen type:

1. create a new screen or select an existing screen of the type `Standard`
2. click on the predefined `Standard` type in the properties in the `Screen type` at `General` node

   Alternatively: In the **`Screen`** detail view, click on the screen type of the elements and select the desired screen type from the drop-down list
3. Select the desired screen type from the drop-down list.
4. select the desired control elements in menu **`Control elements`**
   - **`Add template`** (on page 310) opens a selection dialog for adding pre-defined layouts with certain control elements at pre-defined locations.
   - Elements that are already present on the screen are displayed in gray and can no longer be selected

> **Info**
>
> *Limitation for automatic insertion of the template:*
>
> *If there is already a static element, such as a control element, in the screen, no frame or frame text is inserted when inserting the default control elements.*
>
> ***Recommendation:*** *Insert, after creating a special screen type in the first stage, the default control elements from the template and then further elements.*

17.1.2 Template for standard screens and special screen types

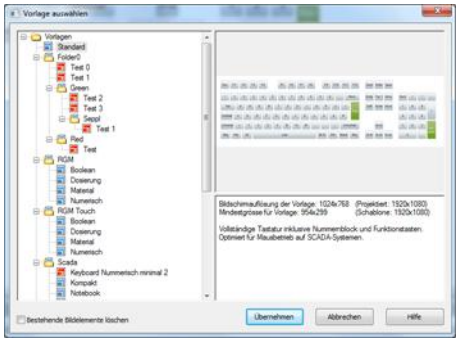
For special screen types and also for standard screen you can use templates for automatic inserting and positioning elements. Templates can be created from every screen by yourself. For special screen types pre-defined templates are shipped with zenon.

Templates are stored as compressed zip files which include three files:

- Meta data: XML file with the description of the template (name, description, size, etc.).

- Screen elements: XML file with the exported elements of the screen.
- Preview: Bitmap file with a preview of the template.

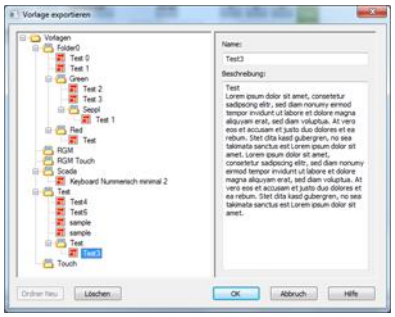
SHIPPED TEMPLATE



The shipped templates are available in different resolutions and also optimized for touch screens. These templates cannot be changed or deleted. Pre-defined templates by COPA-DATA are saved in the program folder under: `\Templates\ScreenTypes\<name of the screen type>\<file name>.zip`.

USER-DEFINED TEMPLATE

Users can create and save templates for all screen types. They can also be edited and deleted.



User-defined templates are saved in the zenon data folder: `Templates\ScreenTypes\<name of screen type>\<file name>.zip`.
e.g.: `%ProgramData%\COPA-DATA\zenon700\Templates\ScreenTypes\Alarm Message List\3e9afbcf-f93d-46d9-a199-e00effe4811c.zip`.

Note: The path depends on whether it is registered at the installation or via the Startup Tool. If the latter is the case, the current program folder is used as data folder.

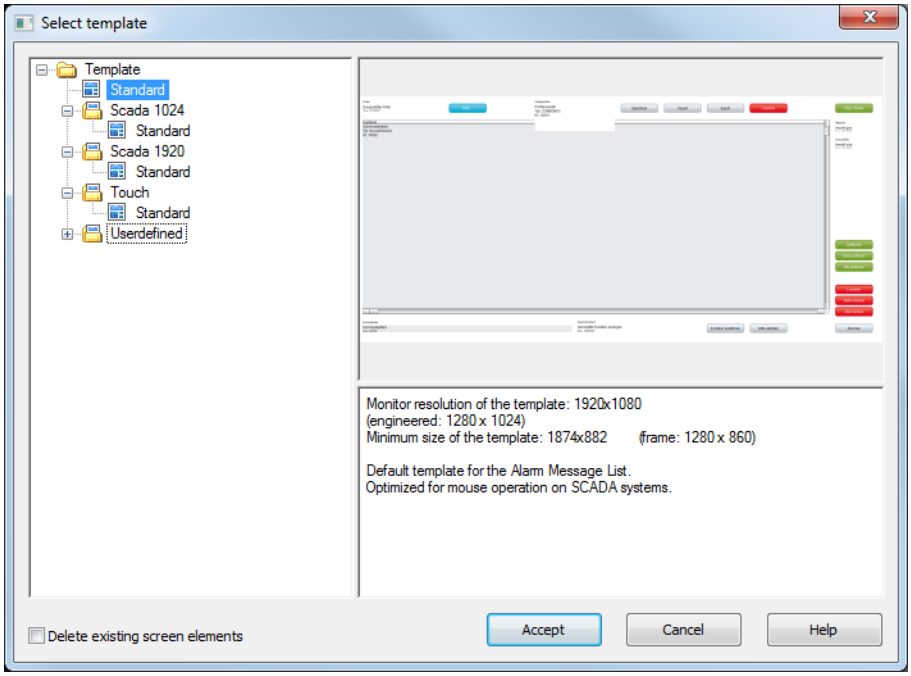
Use templates

To use templates for a screen:

1. open the screen
2. select **`Add template`** in the control element menu
3. the dialog fir selecting pre-defined and user-defined templates is opened

Note: for screen of type Standard only user-defined template are available

4. select the desired templates
5. Click on **`Accept`**
6. the elements defined in the templates are inserted at the pre-defined location



Parameters	Description
List field templates (left)	Zeigt alle vordefinierten und benutzerdefinierten Vorlagen an.
Preview to template (top right)	Preview of the screen at taking over the selected template.
Description template (bottom right)	Description of the template.
Delete existing screen elements	Active: Already existing elements in the screen are deleted when taking over the template.
Accept	Inserts the elements of the selected template in the screen and closes the dialog.
Cancel	Closes dialog without inserting elements.
Help	Opens online help.

Create templates

Folders and template can be newly created, deleted, renamed and moved for user-defined templates. Shipped, pre-defined templates cannot be changed or deleted.

CREATE FOLDER OR TEMPLATE

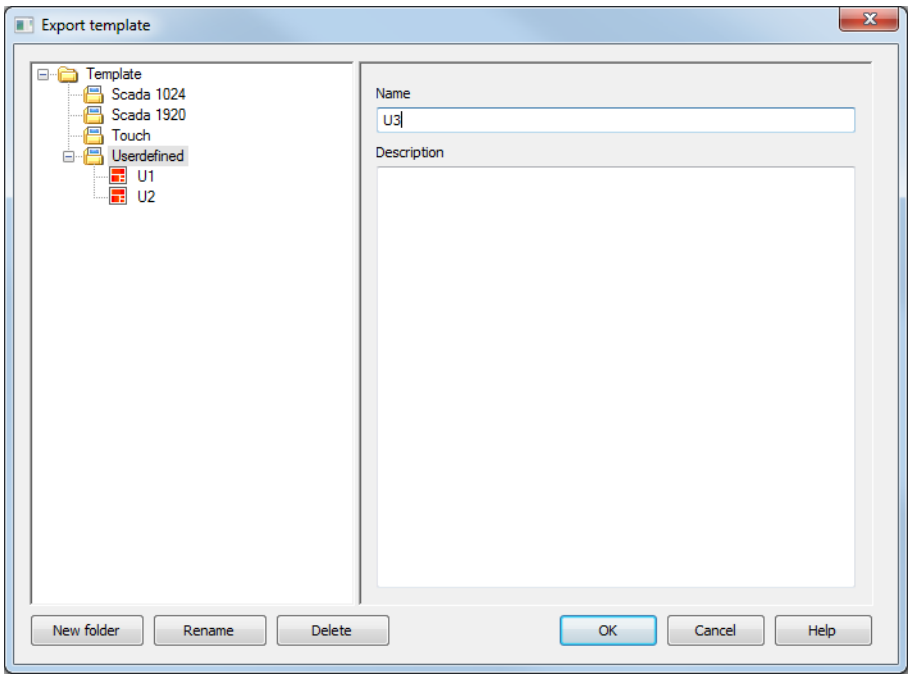
To create a folder, select:

- button **New folder** in the dialog
- or command **New folder** in the context menu
- or use the **Ins** key

To create a template:

1. highlight the desired screen in the detail view
2. select command **Create template for screen type** in the context menu or in the tool bar
3. the dialog for creating own templates is opened
4. select an existing folder or create a new one

5. enter a name for the new template
6. you can add a description as an option
7. confirm this with ok.
8. the template is added to the highlighted folder



Parameters	Description
List field templates (left)	Displays all pre-defined and user-defined template. Makes it possible to delete or rename folder and templates and to add folders via the context menu.
Preview to template (top right)	Preview of the screen at taking over the selected template.
Description template (bottom right)	Description of the template.
New folder	Creates a new folder. You can create several levels.
Delete	Deletes the selected folder with all subfolders after a confirmation request.
OK	Create template and closes dialog.
Cancel	Closes the dialog without creating the template.
Help	Opens online help.

DELETE FOLDER

To delete a folder or a template:

1. highlight the folder or template
2. select
   a) button **Delete** in the dialog
   b) or command **Delete** in the context menu
   c) or use the Del key

The folder or template is deleted after a confirmation request. For folders possible subfolders including all their templates are also deleted.

RENAME FOLDER OR TEMPLATE

To rename a folder or a template:

- Select the **Rename** command in the context menu
- or highlight the element and open the label for renaming by pressing F2.
- or at templates change the text in filed **Name**

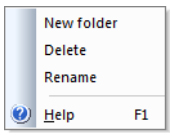
MOVE FOLDERS OR TEMPLATES

Folders and template can be moved via drag & drop with the help of the mouse to other folders.

Context menu create template for screen type

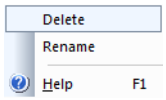
In the dialog for creating new template context menus are also available.

CONTEXT MENU FOR FOLDER



Parameters	Description
New folder	Creates a new folder below the highlighted level.
Delete	Deletes the highlighted folder after a confirmation message.
Rename	Opens the folder caption for renaming.
Help	Opens online help.

CONTEXT MENU FOR TEMPLATES

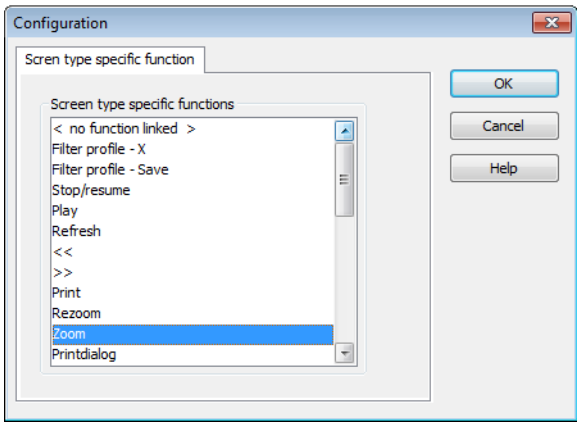


Parameters	Description
Delete	Deletes the highlighted template after a confirmation message.
Rename	Opens the template caption for renaming.
Help	Opens online help.

17.2 Screen-specific functions

If a dynamic element of the **Button** type is created in a screen with a special screen type, `Variable / function`activates additional property`Screen type specific function` is available in properties in the nodes. Therefore the **button** can be assigned special functions, appropriate to the screen type.

For example, in extended trend:



17.3 Screen filter

The screen filter dialog is displayed using the screen switch function on an Alarm Information List, Chronological Event List or Time Filter screen, .

With the screen filters, you have the possibilty to comfortably create individual filter pages. You do not have to use the pre-defined zenon dialogs for this; you can compose your own filter dialogs instead. You can do this according to aspects such as touch operation, filtering of special areas only, ability to switch languages, etc.

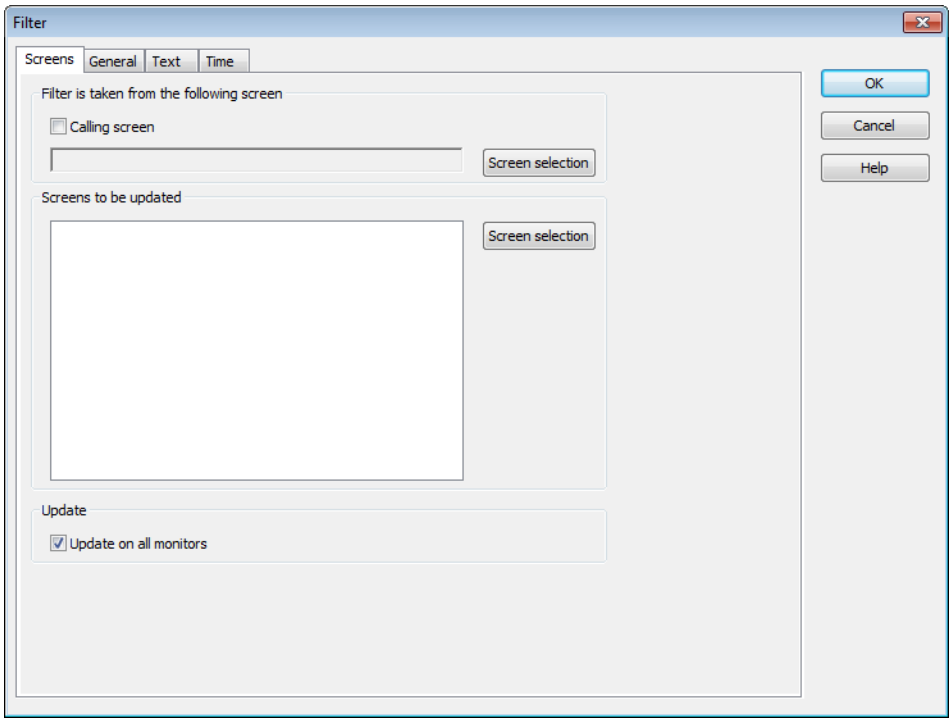
17.3.1 Screen filter dialog

You can make basic adjustments in this dialog. These settings influence the values that are displayed when the attendant filter screens are switched on during runtime.

> **Info**
>
> *Not all tabs are available for all screen filter types.*

Screens

On this tab, you can define the screens which are to be updated by the picture filter.



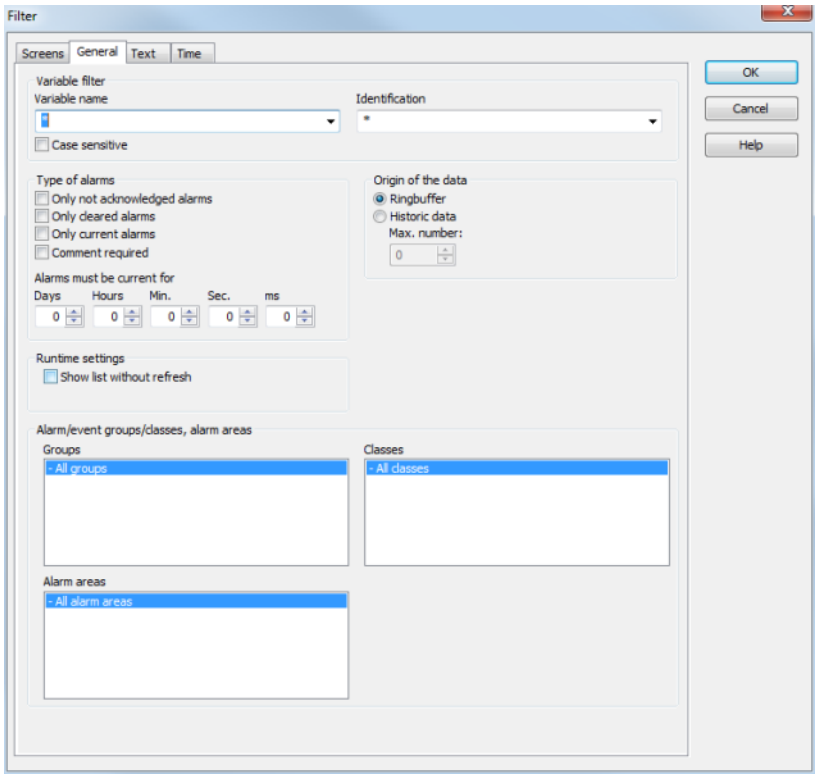
The following settings are available:

Parameters	Description
The filter is taken from the following screen	
`Calling screen`	Active: The filter settings are take over from the screen from which the filter screen is called up. The screen button is grayed out. You cannot explicitly select a screen, because the filter is always updated from the calling screen with this setting. **Note:** Settings in the **`General`**, **`Text`** and **`Time`** tabs are locked.
`Screen selection`	Click on button opens the Screen selection dialog. Select the screen from which the filter - when clicking button **`Update`** during Runtime - should be read. **Note:** It therefore only makes sense to select a screen which can adopt or fill the screen filter. The screen selected is entered into the list of screens to be updated. If you delete it from the list, the next screen on the list is automatically entered here. **Note:** Not available if you have activated the `Calling screen` checkbox.
Screens to be updated	
`Screen selection`	Click the button to open dialog Screen selection of the filter screens. Select the desired screen.
Updating	
`Update on all monitors`	`Active`: The screens from the list of the screens which must be updated are updated on all accessible monitors.

General

This tab is only available for the Alarm Message List and Chronologic Event List and shows the standard filter for the respective list. See Alarm Message List filter and Chronological Event List filter.

AML

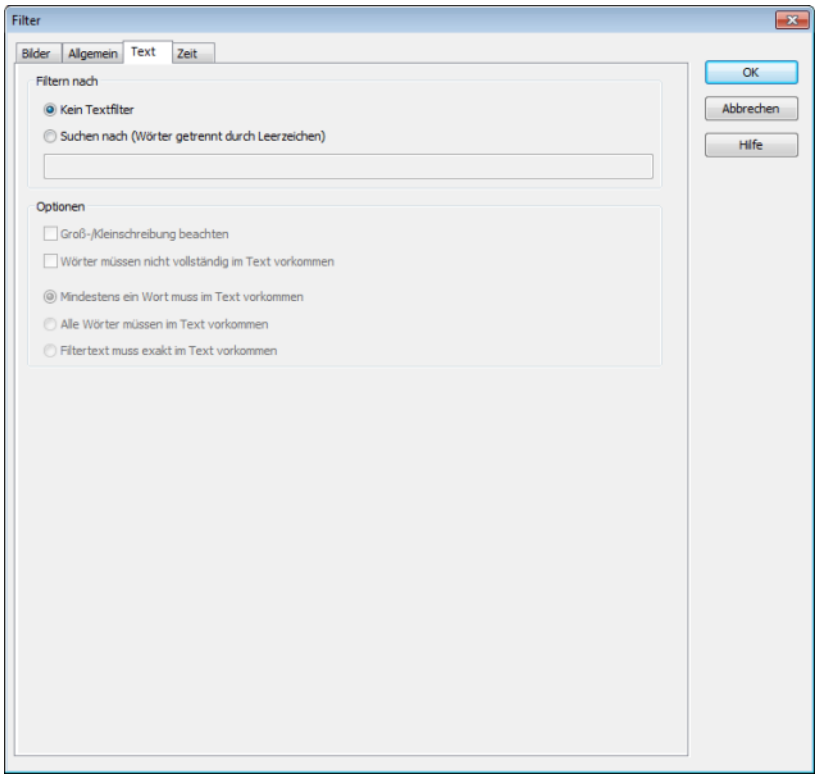

Filter
Screens
General
Text
Time
OK
Cancel
Help
Variable filter
Variable name
Identification
Case sensitive
Type of alarms
Only not acknowledged alarms
Only cleared alarms
Only current alarms
Comment required
Origin of the data
Ringbuffer
Historic data
Max. number:
Alarms must be current for
Days
Hours
Min.
Sec.
ms
Runtime settings
Show list without refresh
Alarm/event groups/classes, alarm areas
Groups
- All groups
Classes
- All classes
Alarm areas
- All alarm areas

Parameters	Description
Variable filter	Limitation to alarms of certain variables
Variable name	Enter the name or part of the name of the variable you want to filter. Use of a wildcard * is possible. **Hint:** Wildcards are only allowed as prefix or suffix; e.g. *xxx or xxx*.
Identification	Enter the identification or partial identification of variables according to which you would like to filter. Use of a wildcard * is possible. **Hint:** Wildcards are only allowed as prefix or suffix; e.g. *xxx or xxx*.
Note capitalization	Active: Upper and lower case is only taken into account when filtering for variable name and/or identification.
Alarm type	Type of alarm that is displayed.
Only unacknowledged alarms	Active: Only alarms that have not yet been acknowledged by the user are displayed.
Only historical alarms	Active: Only alarms that have already passed, i.e. whose values no longer in the critical range, are displayed.
Only current alarms	Active: Only alarms that are still active, i.e. whose values are still in the critical range, are displayed.
Alarms must at least be active	Use the spin control to define the minimum time that an alarm should be active in order for it to be displayed. Possible settings: ▸ Days ▸ Hours (hr.) ▸ Minutes (min.) ▸ Seconds (sec.) ▸ Milliseconds (ms)
Origin of the data	Display of current or current and historical alarms.
Ring buffer	Active: Only data from the ring buffer is displayed.
Historic data	Active: Data from the ring buffer and historical data

Maximum number	from the AML are displayed. The maximum amount of data to be displayed includes the data from the ring buffer.
Runtime settings	Behavior of the AML in Runtime
Show list without refresh	Active: No new entries are added as long as the list is displayed.
Alarm/event groups/classes, alarm areas	Selection of groups, classes and alarm areas
Alarm/event groups	Select, from the alarm/event groups present, the one from which alarms are to be displayed.
Alarm/event classes	Select, from the alarm/event classes present, the one from which alarms are to be displayed.
Alarm areas	Select, from the alarm areas present, the one from which alarms are to be displayed.

Text

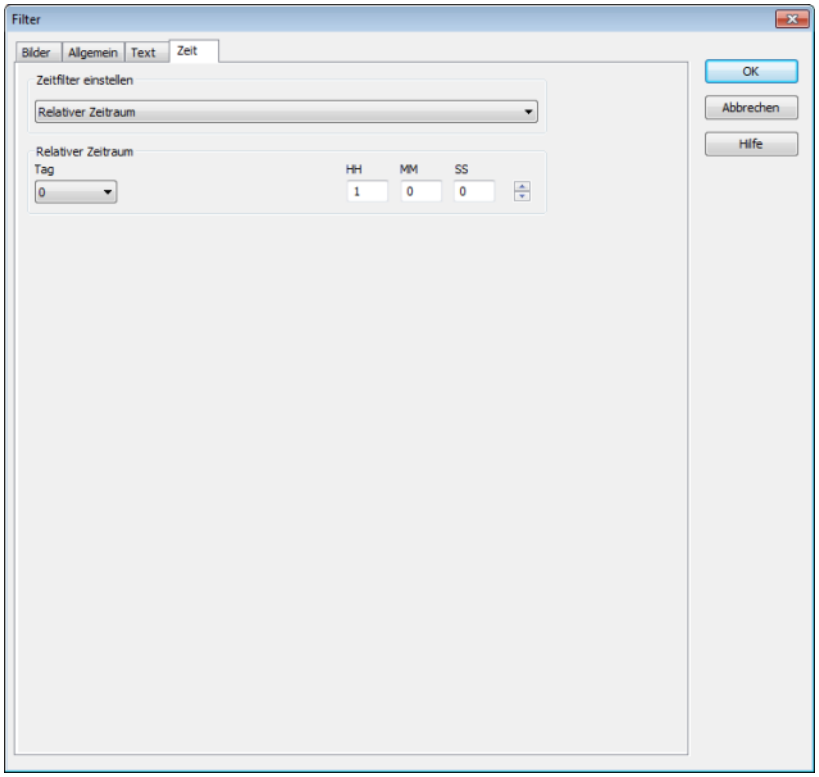
You can define the standard values for text filtering on this tab. Only available for Alarm Message List and Chonological Event List.



Parameters	Description
Filter according to	
No text filter	The text filter is not used.
Search for (words separated by spaces)	The text filter filter is used. Further options are activated.
Input field	Enter the corresponding words or character strings.
Options	
Note capitalization	Active: A distinction is made between upper case and lower case when filtering.
Words do not have to appear in the text in full	Active: Parts of words can also be taken into account during filtering.
At least one word must be present in the text	Active: At least one word of the search string has to be in the text.
All words must be present in the text	Active: All words must be present in the search string. In doing so, the sequence plays no role.
Filter text must appear in the text exactly	Active: The text must be exactly as defined in the search string.

Time

On this tab, you can define the time area which is to be used when the filter screen is opened.



The following settings are available:

Parameters	Options	Description
Absolute period of time		A fixed period of time is entered in the editor. When the function is executed, the defined absolute time period is exactly used. Example: If you want to see all alarms from January 1, 2011 to December 31, 2011, then you must enter the corresponding data at From and To. **Note:** Time is saved in UTC. For details see chapter Handling of date and time in chapter Runtime.
	From	Defines the start time in day, month, year, hour (HH), minute (MM), second (SS)
	To	Defines the end time in day, month, year, hour (HH), minute (MM), second (SS)

Relative time period		A relative time period is entered. **Attention!** this filter is constantly updated. It is therefore carried over. For example: You set up a relative time of 10 minutes and switch to an Alarm Message List Screen with this time filter at 12:00. Then you are shown the alarms from 11:50 to 12:00 when switching. If the Alarm Message List screen stays open, the filter is automatically updated. At 12:01, you see the alarms from 11:51-12:01 etc.
	Time	Defines the relative period in days, hours (HH), minutes (MM) and seconds (SS)
from HH:MM:SS o' clock		A time from which the filter is effective is stated. If the time is not reached on the current day, filtering takes place from the corresponding time the previous day. Example: You enter 23:00:00. If it is then 23:30 when executing the function, then it is filtered from 23:00:00 up to the current point in time. If it is 22:30 however, then filtering takes place from 23:00:00 on the previous day to the current point in time. **Attention!** The start point of this filter is not updated automatically. Only the existing times are used when shown, even if the screen remains open and 23:00:00 is reached. The end time point is not defined with this filter, it is carried over.
	Time	Defines the start time period in hours (HH), minutes (MM) and seconds (SS)
From day - HH:MM:SS time		A day and time for the start of the filter are entered. If the time given has not been reached in the current month, the corresponding time from the previous month is used. Example: You enter day 5 - 23:00:00. If it is the 10th of the month at 23:30, then filtering takes place from the 5th of the month from 23:00:00 to the current time point. If, however, it is the 4th of the month, then filtering takes place from the 5th of the previous month to the current time point. **Attention!** The start time of this filter is not updated automatically. Only the existing times are used when shown, even if the screen remains open and 23:00:00 is reached. The end time point is not defined with this filter, it is carried over.
	Time	Defines the start point in hours (HH), minutes (MM), seconds (SS)

From day, month - HH:MM:SS time		A month, day and time for the start of the filter are entered. If the time stated has not been reached in the current year, the corresponding time from the previous year is used. Example: You enter Month October,Day5-23:00:00. If it is October 10th at 23:30, then filtering takes place from October 5th from 23:00:00 to the current time point. If, however, it is only October 4th, then filtering takes place from the 5th of the previous year to the current time point. **Attention!** The start point of this filter is not updated automatically. Only the existing times are used when shown, even if the screen remains open and 23:00 on October 5 is reached. The end time point is not defined with this filter, it is carried over.
	Time	Defines the start point in months, days, hours (HH), minutes (MM), and seconds (SS)
No time filter		No time filter is used.

17.3.2 Creating an Alarm Message List filter screen

It is possible to adjust filter settings for the Alarm Message List in Runtime with the help of the Alarm Message List Filter screen. Only the filter elements that are actually required are configured and provided to the user The appearance can also be freely defined and thus adapted to different end devices. All filter settings that are available in the filter for the function to switch the screen to the Chronological Event List screen can be configured.

Therefore:

- Only the filter elements that are actually required are configured and provided to the user
- The user only has these filters displayed and has an overview
- The appearance can be freely defined and can, for example, ensure ease of use by means of a touch screen.

For details of use in Runtime, see Using the Alarm Message List Filter chapter.

For the definition of filter criteria, see Filter Alarm Message List Filter chapter.

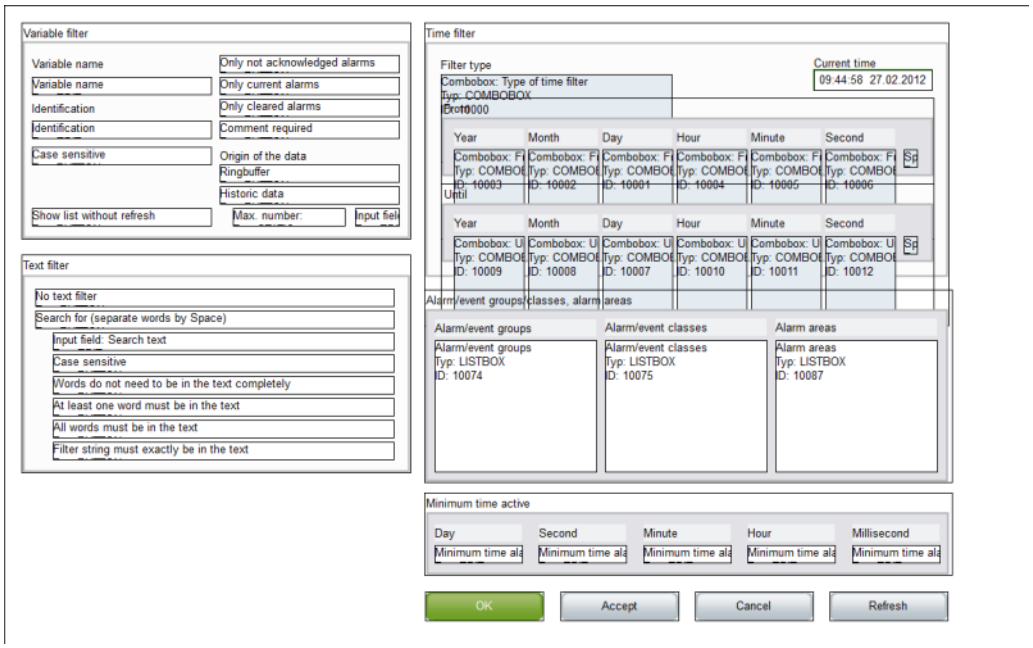
> **Attention**
>
> *Screens of type* ***Alarm Message List Filter****,* ***Chronological Event List Filter*** *and* ***Time Filter*** *must be engineered with an own frame. If they use the same frame as other screens, all screens based on this frame are closed when the screen is closed.*

CREATING AN ALARM MESSAGE LIST FILTER SCREEN

To create an `Alarm Message List filter` screen:

1. Select, in the toolbar or in the context menu of the **Screens** node, the **New Screen** command
2. An standard empty screen is opened
3. Change the screen type in the detail view; to do this:
   a) click on **Standard** in the **Screen type** column
   b) Select **Alarm Message List filter** from the drop down list
4. Select your own template
   (AML filter cannot be based on the same template as other screens)
5. Click in the screen.
6. Select the **Control elements** menu item in the menu bar
7. Click on **Add template** (on page 310) in the drop-down list
8. the dialog for selecting a template is opened
9. select the desired template
10. Select additional elements as required and insert them into the desired place on the screen

11. Create a screen switch function in order to be able to call up the screen in Runtime



ELEMENTS

The Alarm Message filter screen can contain the following control and display elements.

Element	Description
Add template	Opens the dialog for selecting a template (on page 310) for the screen type. Templates are shipped together with zenon and can also be created by the user. Templates add pre-defined control elements to pre-defined locations in the screen. Elements that are not necessary can also be removed individually once they have been created. Additional elements are selected from the drop-down list and dragged onto the screen. Elements can be moved on the screen and arranged individually.
General filter	Drop-down list of different general filters.
Insert all elements: General filter	Inserts all elements from the area of general filters into pre-defined places. Elements can be arranged individually.
Variable filter	Alarms of which variables are displayed:
▸ Variable name	Filter according to names of variables.
▸ Identification	Filter according to identification of variables.
▸ Note capitalization	Note capitalization when filtering the variables.
Alarm type	Which alarms are displayed:
▸ Only unacknowledged alarms	Only unacknowledged
▸ Only historical alarms	Only historical
▸ Only current alarms	Only current
▸ Comment required	Alarms that require a comment when acknowledged
▸ Minimum time for current alarms - days	Only alarms that have been current for at least the given number of days.
▸ Minimum time for current alarms - hours	Only alarms that have been current for at least the given number of hours.
▸ Minimum time for current	Only alarms that have been current for at least the given number of minutes.

alarms - minutes	
▸ Minimum time for current alarms - seconds	Only alarms that have been current for at least the given number of seconds.
▸ Minimum time for current alarms - milliseconds	Only alarms that have been current for at least the given number of milliseconds.
Origin of the data	Where does the data come from:
▸ Ring buffer	From the ring buffer.
▸ Historical data	From an archive.
▸ Labeling: Maximum number	Text for **Maximum number** input field
▸ Input field: Maximum number	Input of the maximum alarms to be displayed when historical alarms are displayed. 0: displays all
Runtime settings	
▸ Show list without refresh	Switches the AML in stopped state. New alarms are not added.
Alarm/event groups/alarm/event classes/alarm areas	List field for grouped display:
▸ Alarm/event groups	Alarm/event groups
▸ Alarm/event classes	Alarm/event classes
▸ Alarm areas	Alarm areas

Time filter	Drop-down list of different time filters (on page 325).
Insert all elements: Absolute time period - classic display	Inserts all elements for the definition of absolute time periods: Each element contains its own field.
Insert all elements: Absolute time period - small display	Inserts elements for the definition of absolute time periods in compact display.
Insert all elements: Relative time period	Inserts all elements for the definition of relative time periods.
Insert all elements: From HH:MM:SS o' clock	Inserts all elements for the definition of "from" time points in hours-minutes-seconds format.
Insert all elements: From day - HH:MM:SS time	Inserts all elements for the definition of "from" time points in day-hours-minutes-seconds format.
Insert all elements: From day, month - HH:MM:SS time	Inserts all elements for the definition of "from" time points in day-month-hours-minutes-seconds format.
Labeling: Time filter type	Labeling for time filter type.
Combobox: Time filter type	Combobox time filter type
Display: Time filter type	Field for time filter type display.
Time from	Fields and labeling for stating "from" time.
▸ Labeling: From year	
▸ Combobox: From year	
▸ Labeling: From month	
▸ Combobox: From month	
▸ Labeling: From day	
▸ Combobox: From day	
▸ Labeling: From hour	
▸ Combobox: From hour	
▸ Labeling: From minute	
▸ Combobox: From minute	
▸ Labeling: From second	

▸ Combobox: From second	
▸ Spin control: From	
Time to	Fields and labeling for stating "to" time.
▸ Labeling: To year	
▸ Combobox: To year	
▸ Labeling: To month	
▸ Combobox: To month	
▸ Labeling: To day	
▸ Combobox: To day	
▸ Labeling: To hour	
▸ Combobox: To hour	
▸ Labeling: To minute	
▸ Combobox: To minute	
▸ Labeling: To second	
▸ Combobox: To second	
▸ Spin control: To	
Absolute time filter	Fields and labeling for stating absolute time filter.
▸ Labeling: From	
▸ Calendar display: From date	
▸ Date display: From date	
▸ Time display: From time	
▸ Labeling: To	
▸ Calendar display: To date	
▸ Date display: To date	

▸ Time display: To time	
Text filter	Drop-down list of different text filters (on page 323).
Insert all elements: Text filter	Inserts all elements for text filters.
No text filter	Radio button to deselect text filter.
Search for (words separated by spaces)	Radio button to activate the search
Labeling: Search text	Labeling for search field.
Input field: Search text	Field for input of search term.
Options	Search options
▸ Note capitalization	Capitalization must be noted.
▸ Words do not have to appear in the text in full	Fragments can also be searched for.
▸ At least one word must be present in the text	At least one search term from several must be in the result.
▸ All words must be present in the text	All search terms must be included in the result.
▸ Filter text must appear in the text exactly	Exact text from the input field must be contained in the result.
OK	Accept inputs and close screen.
Cancel	Reject inputs and close screen.
Accept	Accept inputs and leave screen open for further inputs.
Refresh	Refresh screen.

FILTER SCREENS

Filter screens make it possible to transfer a preset filter from one screen to another. The filter of the source screen is set using the target screen. The screens can also be of a different screen type.

> **Attention**
>
> *If a different* `time period` *than* `Set filter for time filter type` *is selected for the Alarm Message List or the Chronological Alarm List, the time of the screen to be called up cannot be transferred over in Runtime.*

CALL DEFINITION

The following requirements must be met for the set filters to be used:

1. `Set filter for time filter type` is selected as a `time period` for the time filter.
2. The screen (**`Alarm Message List Filter`**, **`Chronological Event List filter`** or **`Time Filter screens`**) are activated using a button or a combined element. Only in this way can the relationship between filter screen and source screen be maintained.
3. The source screen and filter screen must be configured on different frames or monitors. The filter for the filter screen can only be updated if the source screen is open. This is only possible if both screens do not use the same frame or the same monitor.
4. The screen to be called up must be compatible with the filter screen to be called up (see table).

Source screen	AML filter	CEL filter	Time filter
Archive revision	T	T	T
Extended Trend	T	T	T
Time filter	T	T	X
Alarm Message List Filter	X	C	T
Chronological Event List Filter	C	X	T
Alarm Message List	X	C	T
Chronological Event List	C	X	T

Key:

- C: Common settings are updated.
- T: Time settings are updated.

- X: All settings are updated.

> **Info**
>
> **No filtering**
>
> *The filter screen is not filtered, but opened with the configured values, if:*
>
> - One of the conditions 1 to 3 is not met or
> - The `Screen to call up` setting is not activated for the **`Screen switching`** function or
> - The screen is not called up via a screen element
>
> *In this case, the **`Accept`**, **`Close`** and **`Update`** buttons are grayed out in Runtime and have no function.*

17.3.3 Create a screen of the type CEL Filter

It is possible to adjust filter settings for the Chronological Event List in Runtime with the help of the `Chronological Event List Filter` screen. Only the filter elements that are actually required are configured and provided to the user The appearance can also be freely defined and thus adapted to different end devices. All filter settings that are available in the filter for the function to switch the screen to the Chronological Event List screen can be configured.

Therefore:

- Only the filter elements that are actually required are configured and provided to the user
- The user only has these filters displayed and has an overview
- The appearance can be freely defined and can, for example, ensure ease of use by means of a touch screen.

For details of use in the Runtime, see Using the CEL Filter chapter.

For the definition of filter criteria, see Filter screen switch CEL Filter chapter.

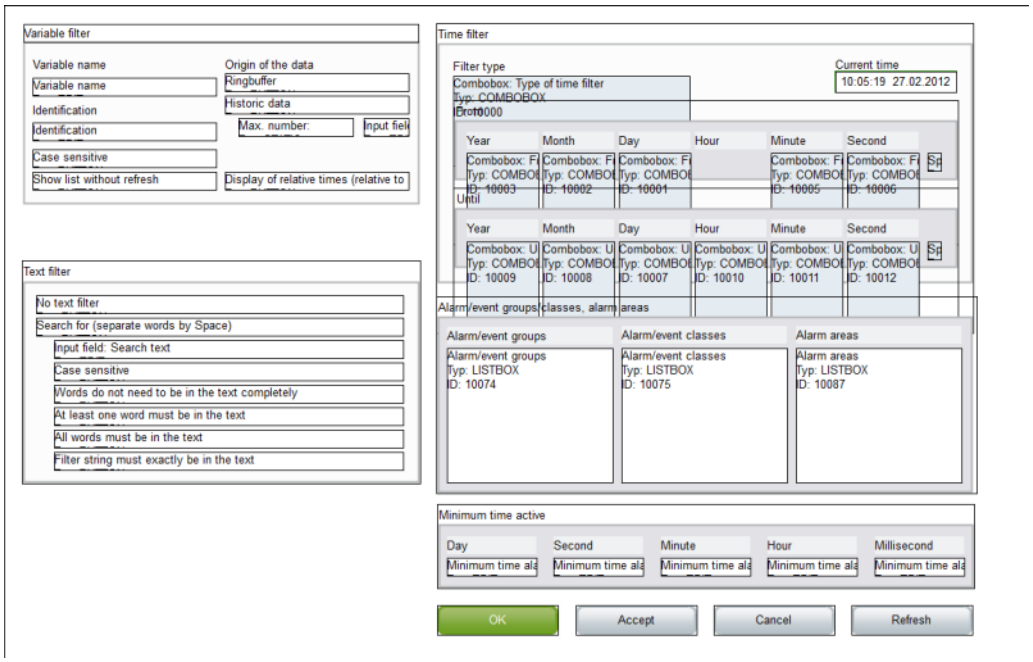
> **Attention**
>
> *Screens of type* ***Alarm Message List Filter****,* ***Chronological Event List Filter*** *and* ***Time Filter*** *must be engineered with an own frame. If they use the same frame as other screens, all screens based on this frame are closed when the screen is closed.*

CREATE A SCREEN OF TYPE CHRONOLOGICAL EVENT LIST FILTER

To create a screen of type `Chronological Event List Filter`:

1. Select, in the toolbar or in the context menu of the **Screens** node, the **New Screen** command
2. An standard empty screen is opened
3. Change the screen type in the detail view; to do this:
   a) click on **Standard** in the **Screen type** column
   b) select **Chronological Event List Filter** from the drop-down list
4. select your own frame
   (CEL filter cannot be based on the same frame as other screens)
5. Click in the screen.
6. Select the **Control elements** menu item in the menu bar
7. Click on **Add template** (on page 310) in the drop-down list
8. the dialog for selecting a template is opened
9. select the desired template
10. Select additional elements as required and insert them into the desired place on the screen

11. Create a screen switch function in order to be able to call up the screen in Runtime



ELEMENTS

The screen of type `Chronological Event List Filter` can contain the following control and display elements.

Element	Description
Add template	Opens the dialog for selecting a template (on page 310) for the screen type. Templates are shipped together with zenon and can also be created by the user. Templates add pre-defined control elements to pre-defined locations in the screen. Elements that are not necessary can also be removed individually once they have been created. Additional elements are selected from the drop-down list and dragged onto the screen. Elements can be moved on the screen and arranged individually.
General filter	Drop-down list of different general filters.
Insert all elements: General filter	Inserts all elements from the area of general filters into pre-defined places. Elements can be arranged individually.
Variable filter	Alarms of which variables are displayed:
▸ Variable name	Filter according to names of variables.
▸ Identification	Filter according to identification of variables.
▸ Note capitalization	Note capitalization when filtering the variables.
Origin of the data	Where does the data come from:
▸ Ring buffer	From the ring buffer.
▸ Historical data	From an archive.
▸ Labeling: Maximum number	Text for **Maximum number** input field
▸ Input field: Maximum number	Input of the maximum alarms to be displayed when historical alarms are displayed. 0: displays all
Runtime settings	
▸ Show list without refresh	Switches the AML in stopped state. New alarms are not added.
▸ Display relative time	Switches between the normal display and the relative-time display without the entries losing focus.

	Relative time: All entries are displayed in the time distance to the selected entry. The displayed time is the difference time passed since the selected entry. The selected entry automatically gets the time stamp 0. The other events have a: ▸ positive time difference to the selected entry if they occurred later ▸ negative time difference to the selected entry if they occurred earlier
Alarm/event groups/alarm/event classes/alarm areas	List field for grouped display:
▸ Alarm/event groups	Alarm/event groups
▸ Alarm/event classes	Alarm/event classes
▸ Alarm areas	Alarm areas

Time filter	Drop-down list of different time filters (on page 325).
Insert all elements: Absolute time period - classic display	Inserts all elements for the definition of absolute time periods: Each element contains its own field.
Insert all elements: Absolute time period - small display	Inserts elements for the definition of absolute time periods in compact display.
Insert all elements: Relative time period	Inserts all elements for the definition of relative time periods.
Insert all elements: From HH:MM:SS o' clock	Inserts all elements for the definition of "from" time points in hours-minutes-seconds format.
Insert all elements: From day - HH:MM:SS time	Inserts all elements for the definition of "from" time points in day-hours-minutes-seconds format.
Insert all elements: From day, month - HH:MM:SS time	Inserts all elements for the definition of "from" time points in day-month-hours-minutes-seconds format.
Labeling: Time filter type	Labeling for time filter type.
Combobox: Time filter type	Combobox time filter type
Display: Time filter type	Field for time filter type display.
Time from	Fields and labeling for stating "from" time.
▸ Labeling: From year	
▸ Combobox: From year	
▸ Labeling: From month	
▸ Combobox: From month	
▸ Labeling: From day	
▸ Combobox: From day	
▸ Labeling: From hour	
▸ Combobox: From hour	
▸ Labeling: From minute	
▸ Combobox: From minute	
▸ Labeling: From second	

▸ Combobox: From second	
▸ Spin control: From	
Time to	Fields and labeling for stating "to" time.
▸ Labeling: To year	
▸ Combobox: To year	
▸ Labeling: To month	
▸ Combobox: To month	
▸ Labeling: To day	
▸ Combobox: To day	
▸ Labeling: To hour	
▸ Combobox: To hour	
▸ Labeling: To minute	
▸ Combobox: To minute	
▸ Labeling: To second	
▸ Combobox: To second	
▸ Spin control: To	
Absolute time filter	Fields and labeling for stating absolute time filter.
▸ Labeling: From	
▸ Calendar display: From date	
▸ Date display: From date	
▸ Time display: From time	
▸ Labeling: To	
▸ Calendar display: To date	
▸ Date display: To date	

▸ Time display: To time	
Text filter	Drop-down list of different text filters (on page 323).
Insert all elements: Text filter	Inserts all elements for text filters.
No text filter	Radio button to deselect text filter.
Search for (words separated by spaces)	Radio button to activate the search
Labeling: Search text	Labeling for search field.
Input field: Search text	Field for input of search term.
Options	Search options
▸ Note capitalization	Capitalization must be noted.
▸ Words do not have to appear in the text in full	Fragments can also be searched for.
▸ At least one word must be present in the text	At least one search term from several must be in the result.
▸ All words must be present in the text	All search terms must be included in the result.
▸ Filter text must appear in the text exactly	Exact text from the input field must be contained in the result.
OK	Accept inputs and close screen.
Cancel	Reject inputs and close screen.
Accept	Accept inputs and leave screen open for further inputs.
Refresh	Refresh screen.

FILTER SCREENS

Filter screens make it possible to transfer a preset filter from one screen to another. The filter of the source screen is set using the target screen. The screens can also be of a different screen type.

> **Attention**
>
> *If a different* `time period` *than* `Set filter for time filter type` *is selected for the Alarm Message List or the Chronological Alarm List, the time of the screen to be called up cannot be transferred over in Runtime.*

CALL DEFINITION

The following requirements must be met for the set filters to be used:

1. `Set filter for time filter type` is selected as a `time period` for the time filter.
2. The screen (**`Alarm Message List Filter`**, **`Chronological Event List filter`** or **`Time Filter screens`**) are activated using a button or a combined element. Only in this way can the relationship between filter screen and source screen be maintained.
3. The source screen and filter screen must be configured on different frames or monitors. The filter for the filter screen can only be updated if the source screen is open. This is only possible if both screens do not use the same frame or the same monitor.
4. The screen to be called up must be compatible with the filter screen to be called up (see table).

Source screen	AML filter	CEL filter	Time filter
Archive revision	T	T	T
Extended Trend	T	T	T
Time filter	T	T	X
Alarm Message List Filter	X	C	T
Chronological Event List Filter	C	X	T
Alarm Message List	X	C	T
Chronological Event List	C	X	T

Key:

- C: Common settings are updated.
- T: Time settings are updated.

- X: All settings are updated.

> **Info**
>
> **No filtering**
>
> *The filter screen is not filtered, but opened with the configured values, if:*
>
> - One of the conditions 1 to 3 is not met or
> - The `Screen to call up` setting is not activated for the **`Screen switching`** function or
> - The screen is not called up via a screen element
>
> *In this case, the **`Accept`**, **`Close`** and **`Update`** buttons are grayed out in Runtime and have no function.*

17.3.4 Time filter screen

It is possible to make changes to time filter settings during runtime via the time filter screeen. The following screens can be influenced by the filter:

- Alarm Message List
- Archive revision
- Chronological Event List
- Extended Trend

The advantage of this type of screen is that only the filter elements that are actually necessary can be configured. Therefore the user does not need to get to grips with numerous filter settings. Only the filter settings that he needs are displayed. Furthermore, you are completely free to change the appearance and can, for example, ensure ease of use by means of a touch screen.

> **Attention**
>
> *Screens of type* ***Alarm Message List Filter****,* ***Chronological Event List Filter*** *and* ***Time Filter*** *must be engineered with an own frame. If they use the same frame as other screens, all screens based on this frame are closed when the screen is closed.*

In the time filter screen, all time filter settings that are also present in the filter for the function to switch screens to the appropriate screen can be projected.

FILTER SCREENS

Filter screens make it possible to transfer a preset filter from one screen to another. The filter of the source screen is set using the target screen. The screens can also be of a different screen type.

> **Attention**
>
> *If a different* `time period` *than* `Set filter for time filter type` *is selected for the Alarm Message List or the Chronological Alarm List, the time of the screen to be called up cannot be transferred over in Runtime.*

CALL DEFINITION

The following requirements must be met for the set filters to be used:

1. `Set filter for time filter type` is selected as a `time period` for the time filter.
2. The screen (**`Alarm Message List Filter`**, **`Chronological Event List filter`** or **`Time Filter screens`**) are activated using a button or a combined element. Only in this way can the relationship between filter screen and source screen be maintained.
3. The source screen and filter screen must be configured on different frames or monitors. The filter for the filter screen can only be updated if the source screen is open. This is only possible if both screens do not use the same frame or the same monitor.
4. The screen to be called up must be compatible with the filter screen to be called up (see table).

Source screen	AML filter	CEL filter	Time filter
Archive revision	T	T	T
Extended Trend	T	T	T
Time filter	T	T	X
Alarm Message List Filter	X	C	T
Chronological Event List Filter	C	X	T
Alarm Message List	X	C	T
Chronological Event List	C	X	T

Key:

- C: Common settings are updated.
- T: Time settings are updated.
- X: All settings are updated.

Info

No filtering

The filter screen is not filtered, but opened with the configured values, if:

- One of the conditions 1 to 3 is not met or
- The `Screen to call up` setting is not activated for the **`Screen switching`** function or
- The screen is not called up via a screen element

*In this case, the **`Accept`**, **`Close`** and **`Update`** buttons are grayed out in Runtime and have no function.*

UPDATING

When a filter screen is called up (**`Alarm Message List filter`**, **`Chronological Event List filter`**, **`time filter`**), the screens configured in the **`screen switching function`** are updated in two ways:

- If the filter screen is called up via a screen element, the target screens on the same monitor as the source screen are updated.
- If the filter is called up in a different way or if the `Update on all monitors` setting is activated, all target screens configured are updated.

They are updated as soon as you click the **`Accept`** button or as soon as you closes the filter screen with the close **`Close`** button. The **`Cancel`** button discards the changes and closes the filter screen.

UPDATE FILTER SETTINGS

You update the current filter settings for the source screen using the **`Update`** button. If the filter screen is not called up by a screen element or if the `Calling screen` has not been activated, all monitors are searched for screens that can be used for updating. The first screen that is found is then this is used for updating.

17.4 HTML

Makes it possible to display a screen in HTML. The currently installed Microsoft Internet Explorer is used as rendering engine.

17.4.1 Creating a screen of the type HTML

In order to create screen HTML:

1. create a new screen
2. select HTML as screen type
3. an empty screen is created
4. Click on **Add template** (on page 310) in the drop-down list
5. the dialog for selecting a template is opened
6. select the desired template

Addresses for the navigation and the search are defined when function Screen switch (on page 349) is created.

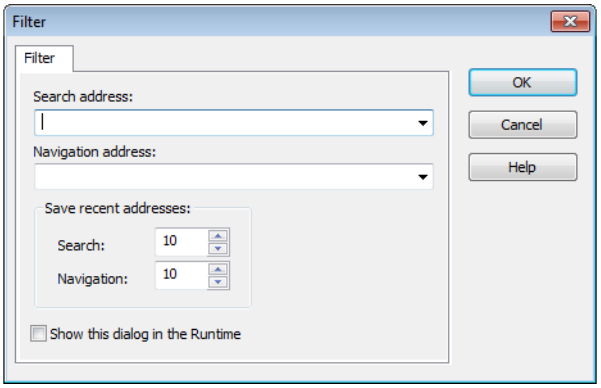
> **Info**
>
> *The menu item **Add template** adds pre-defined control elements to pre-defined locations in the screen. All control elements can be added, deleted and freely positioned. Other dynamic elements and vector elements can also be added.*

17.4.2 Function Screen switch - HTML

In order to create a function to switch to the screen HTML:

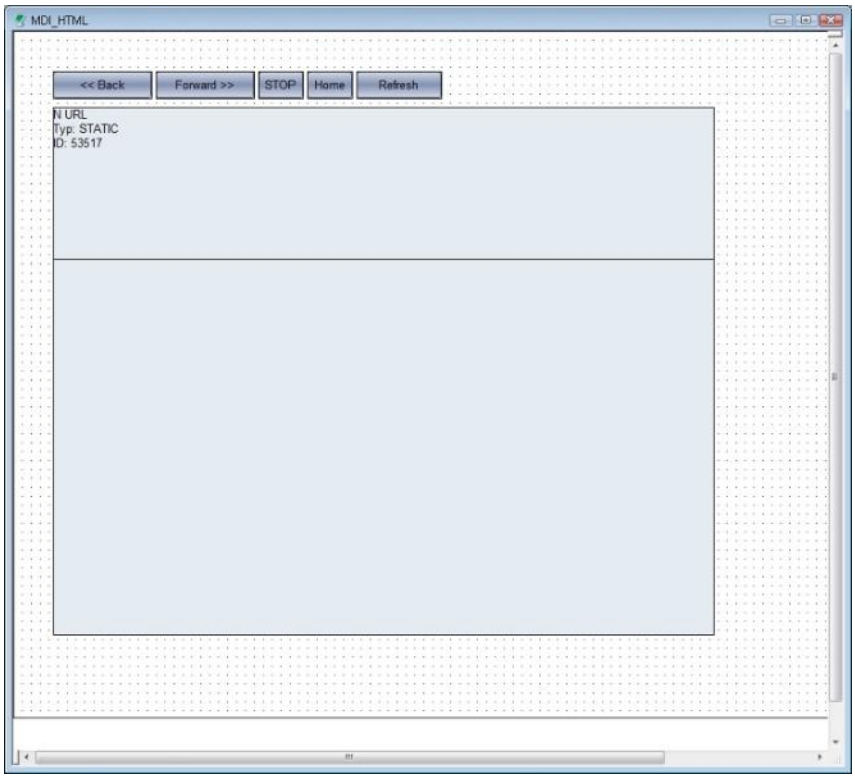
- select New function
- select screen switch

- select the screen HTML (on page 349)
- a dialog for defining the addresses for the search and navigation area is opened



Parameters	Description
Search address	Address of the page, that should be displayed in the search area.
Navigate address	Address of the page, that should be displayed in the navigate area.
Save recent addresses	
Find	Number of addresses, that should be stored by the system for the search area.
Navigation	Number of addresses, that should be stored by the system for the navigate area.
Show this dialog in the Runtime	Active: Opens the dialog when the screen is opened in the Runtime.

17.4.3 Operating during Runtime



Control element	Description
Find	Search area element
Navigate	Navigation area element When clicking a link in the field '**Search**', the according page will be displayed in the field '**Navigate**'. So e.g. in the field '**Search**' a navigation bar or the results of a search engine can be displayed without changing the contents, when a link is activated.
Search Back	Page up in the search area.
Search Forward	Page down in the search area.
Cancel search action	Cancel search action.
Search Home	Back to home in the search area.
Search Refresh	Refresh search area.

Search URL	Search for address or file.
Navigate Back	Page up in the navigation area.
Navigate Forward	Page down in the navigation area.
Cancel navigate action	Cancel navigate action.
Navigate Home	Back to home in the navigation area.
Navigate Refresh	Refresh navigation area.
Navigate URL	Navigate to address or file.
Filter	Open filterbox.

17.5 Notebook

With the notepad you can display text files and (optionally) edit them.

17.5.1 Creating screen Notepad

The display size of the screen Notepad is taken from the frame size (on page 242). The file which is opened must be determined during the definition of the Screen switch (on page 353).

In order to create a Notepad:

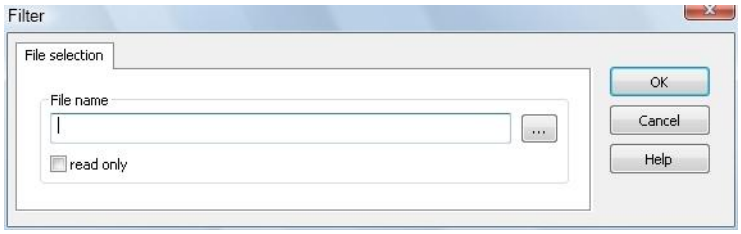
1. create a new screen
2. select Notepad as screen type
3. an empty screen is created
4. Click on **Add template** (on page 310) in the drop-down list
5. the dialog for selecting a template is opened
6. select the desired template

> **Info**
>
> **`Add template`** adds pre-defined control elements to pre-defined locations in the screen. All control elements can be added, deleted and freely positioned. Other dynamic elements and vector elements can also be added.

17.5.2 Function screen switch to Notepad

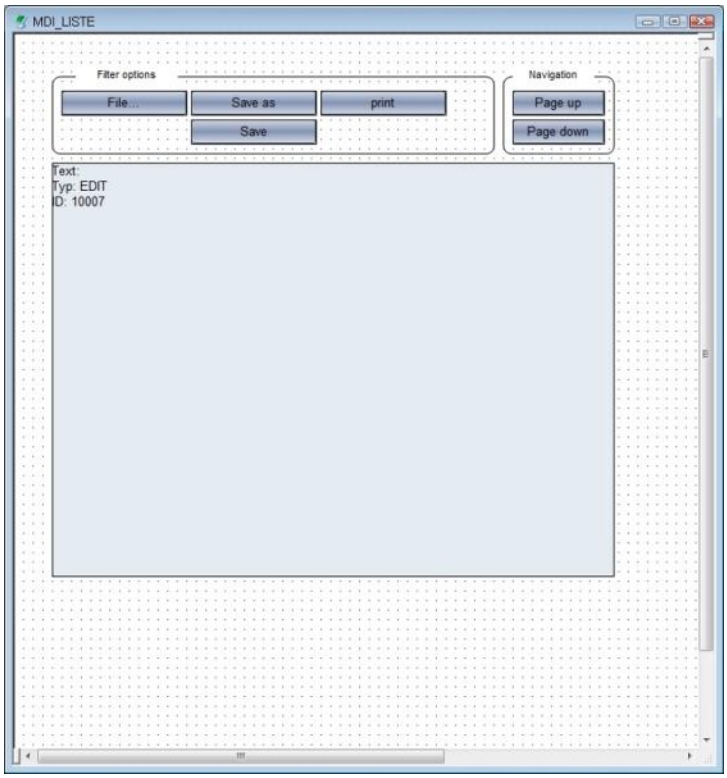
In order to create a function to switch to the screen Notepad:

- select New function
- select screen switch
- select the screen Notepad (on page 352)
- a dialog for defining the file which should be displayed is opened



Parameters	Description
`File name`	Selection of the file which is to be displayed Click on ... and a dialog is opened. The file must have already been created in node file/notepad. If no appropriate file has been created, you can select on in the dialog using symbol **`Add file`**. Allowed file types: TXT, HTM, HTML, FRM.
`Read only`	Active: In the Runtime the text can only be read but not edited.

17.5.3 Operating during Runtime



Control element	Action
Button file	File selection in online operation
Button refresh	Refresh display in online operation
Button save	Save changes in online operation to file
Button save as	Save changes to a file with a different name
Button print	Print report in online operation
Button print dialog	Printer settings
Button close	Close alarm information list
Button column left	Scroll text in list element to the right
Button column right	Scroll text in list element to the left
Button page left	Scroll text in list element to the right
Button page right	Scroll text in list element to the left
Button line up	Scroll text in list element up
Button line down	Scroll text in list element down
Button page up	Scroll text in list element up
Button page down	Scroll text in list element down
Text element	Display window of the text list

17.6 Video

Online and stored video recordings can be displayed in zenon by using a screen for the representation of videos. The format used is **.avi**.

In the Runtime videos are centered in the frame when displayed. The control is carried out using control element.

Note: The codecs needed to play back the video must be installed on the computer.

17.6.1 Creating a screen of the type Video

In order to create screen Video:

1. create a new screen
2. select Video as screen type
3. an empty screen is created
4. Click on **Add template** (on page 310) in the drop-down list
5. the dialog for selecting a template is opened
6. select the desired template

> **Info**
>
> ***Add template*** *adds pre-defined control elements to pre-defined locations in the screen. All control elements can be added, deleted and freely positioned. Other dynamic elements and vector elements can also be added.*

17.6.2 Screen switch - video

With screen `Video` you can display two types of videos in the Runtime:

1. saved videos (*.avi)
2. Videos of a surveillance camera

Which video is shown depends on which tab was shown when you saved the dialog.

> **Info**
>
> *If video files and online video should be displayed, create to screen switch functions.*

In order to create a function to switch to the screen Video:

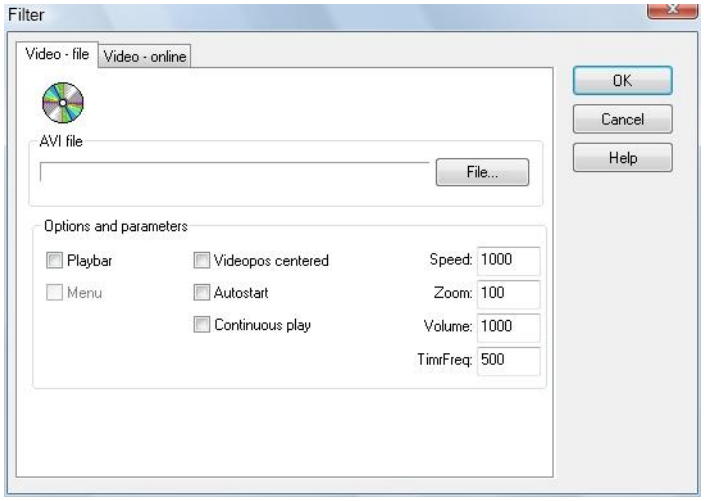
- select New Function

- select screen switch
- select the screen Video (on page 356)
- a dialog for the definition of the video source opens.
  - Video file (on page 357)
  - Video online (on page 358)

Video file

The parameters needed for the display of screen **video** for showing a surveillance camera are the following:

- Set the video which should be shown
- Settings for position, speed, volume and zoom factor of the display



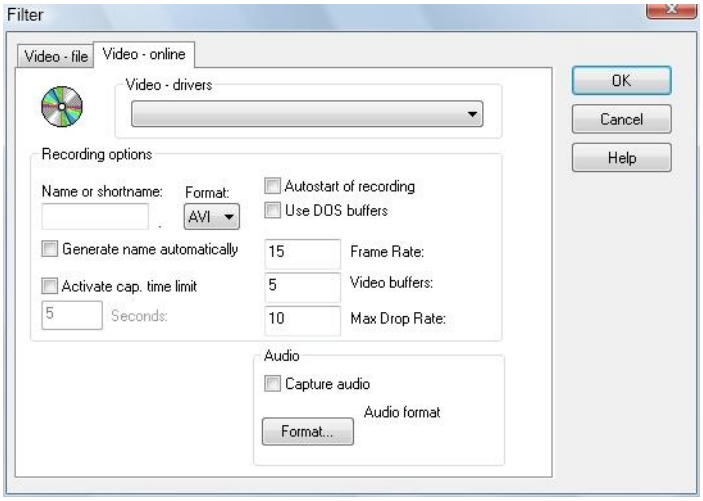
Parameters	Description
AVI-file	Saved video file to be displayed
Options and parameters	
Playbar	Display of a slider in the video screen.
Menu	Display of a menu in the playbar for operating the video (only possible with activated playbar)
Videopos center	Reference point for the position of the video display in zooming actions. Active: The zoomed video is placed in the center of the frame. Inactive: The video display is zoomed to bottom right.
Auto start	Video file is automatically started when opening the screen
Continuous play	Video file is restarted after each end.
Speed	Speed in thousandth of the original speed. Default: 1000 (original speed)
Zoom	Zoom factor for enlarging and reducing the original size in percent while obtaining the correct aspect ratio. Default: 100 (original size)
Volume	Speed in thousandth of the original volume. Default: 1000 (original volume)
TimrFreq	**Note:** Is currently not used. Timer frequency. Default: 500

Video online

The parameters needed for the display of screen Video for showing a surveillance camera are the following:

- Setting of the video driver

- Setting of the recording options if the video should be recorded



Parameters	Description
Video - drivers	Selection of the video driver used. **Note:** the driver should support Microsoft Capturing.
Record options	Options for the recording and saving of the video as an AVI-file or Bitmaps. If only online video should be displayed, there are no settings needed.
Name or abbreviation	Name of the file to be saved.
Automatic assignment	Name is automatically generated from the system time during recording.
Activate cap. time limit	Time limit for the recording in seconds.
Format	AVI or BMP.
Auto start of the recording	Recording is automatically started when the screen is displayed.
Use DOS buffers	Only relevant for 16-bit systems.
Frame Rate	Number of pictures per second for recording
Video buffers	Reserved buffer for video recording in MB
Max Drop Rate	Maximum amount of "drooped frames" in percent during recording. Value: 0 to 100 Default: 10
Video	Driver specific settings.
Views ...	Setting depends on the selected driver.
Format ...	Setting depends on the selected driver.
Compress ...	Setting depends on the selected driver.
Source ...	Setting depends on the selected driver.
Audio	Settings for the sound recording
Capture audio	Activates the sound recording.
Audio format	Driver specific settings.

17.6.3 Operating during Runtime

When displayed, the videos are centered in the frame.

Control element	Description
Video online	Elements for controlling an online video.
Format	Output format
Source	Data source
Display	Video display
Compression	Data compression.
Video file	Elements for controlling a video file.
Speed	Speed.
Zoom	Zoom.
Filters	Filter settings.
Start	Play video.
Replay	Play video again.
Record	Record video.
Pause	Stop video during playback.
Pause/Resume	Pause playback/resume playback.
Eject	Eject.
>>	Fast forward.
<<	Fast reverse.
End	Forward to the end of the video.
Pos1	Rewind to the beginning of the video.

18. Screen Functions

When the standard function is created from the context menu in the detail view, <CD_PRODUCTNANE> creates a function 'screen switch' for the selected screen.

18.1 Screen switch

The **screen switch** function makes it possible to switch between process screens in runtime

> **Example**
>
> *In a project, there is a **Start** screen and a screen of type **Alarm Message List**. During runtime, it should be possible to switch between these two screens. So, for example, a button is created in the **Start** screen and a function of type **screen switching** in the **Alarm Information list** screen. After this, the function is linked to the button, for example by dragging & dropping. Now during runtime, the **Start** screen can be left by pressing the button and the **Alarm Message List** can be switched on.*

CONFIGURE SCREEN SWITCHING

To configure the function:

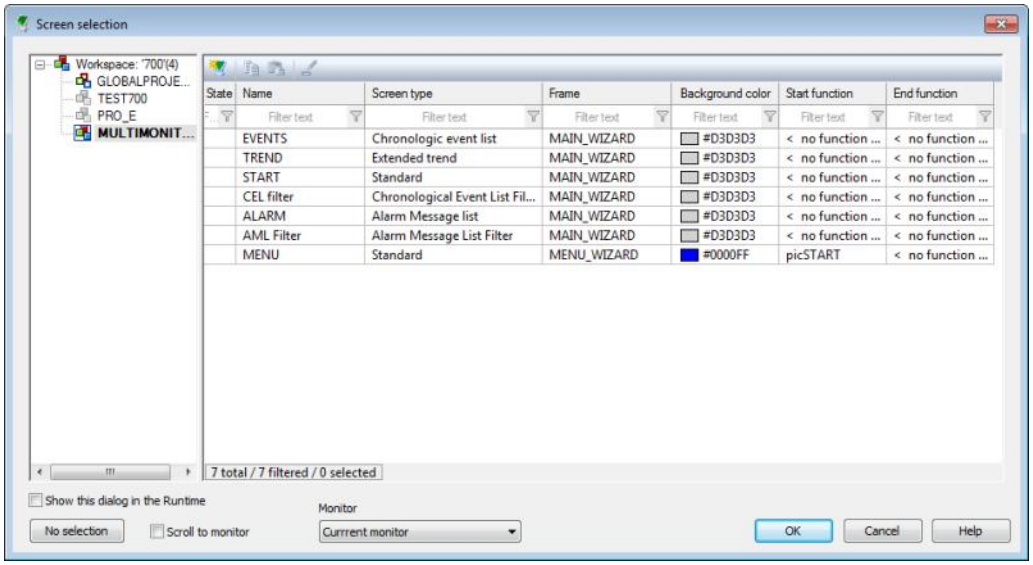
1. Create a new function
2. navigate to node `Screens`
3. Select the `Screen switch` function
4. The dialog for selecting a screen is opened
5. Select a screen
6. The dialog to replace variables and functions (on page 225) is opened
7. Close the dialog by clicking on OK
8. Allocate a button to the function

SCREEN SELECTION DIALOG

In the screens selection dialog, screens can be selected for the execution of functions from:

- Current project
- Subprojects

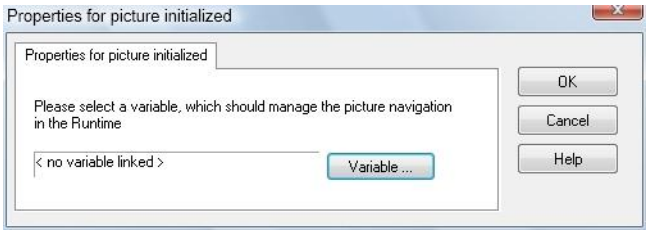
- All projects in the workspace with the **`Keep project in memory`** option active



Parameters	Description
Project tree window	Displays all projects in the workspace. Screens can be selected from the current project and from all projects with the **Keep project in memory** option active
Screens window	Selection of the screen. If several screens are selected, the first selected screen is used for the execution of the function.
Show this dialog in the Runtime	Switches to this screen selection dialog when this function is executed in Runtime.
No selection	Has no function for this screen selection.
Scroll to monitor	Active: Scroll bars are shown in Runtime, which make it possible to scroll to the monitor. Only available with multi-monitor systems.
Monitor	Selection of the monitor from the drop-down list. This contains: ▸ **Current monitor** ▸ All virtual monitors defined in the monitor administration ▸ optional: **All monitors** Only available with multi-monitor systems. Only available with multi-monitor systems.
OK	Applies settings and closes dialog.
Cancel	Discards all changes and closes dialog.
Help	Opens online help.

18.2 Index screen

This function is used to switch to the screen with the name stated in the linked string variable.



A string variable is to be selected from the list of variables as a transfer parameter.

Note for multiple-monitor systems: It is not possible to select monitors. The function always relates to the first monitor.

18.3 Close screen

A defined screen in Runtime is closed with this screen. If, with a multiple-monitor projects, a screen is switched to several monitors, it is possible to select which screen is closed on which monitor.

To configure the function:

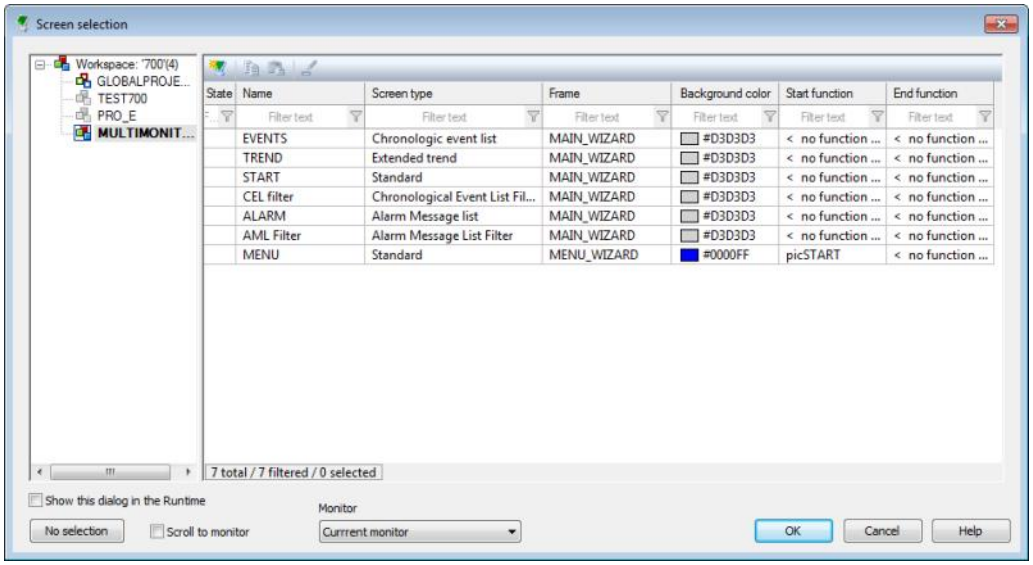
1. Select, in the list of functions, in the `Screens` node, the `Close screen` function
2. The dialog for selecting a screen is opened
3. select the desired picture
4. Select, with multiple-monitor projects, the virtual monitor on which the screen is to be closed from the drop-down list:
   - all objects
   - Current monitor
   - Designated virtual monitor

SCREEN SELECTION DIALOG

In the screens selection dialog, screens can be selected for the execution of functions from:

- Current project

- Subprojects
- All projects in the workspace with the **`Keep project in memory`** option active



Parameters	Description
Project tree window	Displays all projects in the workspace. Screens can be selected from the current project and from all projects with the **Keep project in memory** option active
Screens window	Selection of the screen. If several screens are selected, the first selected screen is used for the execution of the function.
Show this dialog in the Runtime	Switches to this screen selection dialog when this function is executed in Runtime.
No selection	Has no function for this screen selection.
Scroll to monitor	Active: Scroll bars are shown in Runtime, which make it possible to scroll to the monitor. Only available with multi-monitor systems.
Monitor	Selection of the monitor from the drop-down list. This contains: ▸ **Current monitor** ▸ All virtual monitors defined in the monitor administration ▸ optional: **All monitors** Only available with multi-monitor systems. Only available with multi-monitor systems.
OK	Applies settings and closes dialog.
Cancel	Discards all changes and closes dialog.
Help	Opens online help.

18.4 Screen back

This function reverts to the screen previously open in Runtime during online operation. In doing so, the same function that is used to to open the screen is executed once again. This means that screen filters are not saved but have to be recreated.

The screens that were previously open are stored in a buffer. By executing the `Screen switching` function, the screens of the corresponding frames (on page 242) are stored in the memory. The number of screens in the memory is configured in the `project.ini` file.

[FUNKTIONEN]

ZURUECK_ANZAHL=10

The frames have to be given as a transfer parameter.

The amount of memory must be given with the ZURUECK_ANZAHL= entry.

Note for multiple-monitor systems: It is not possible to select monitors. The function always relates to all monitors.

18.5 Focus: Activate input to the element with the focus

This function activates the element that is being focused on in the frame selected.

18.6 Set focus to template

This function sets the focus to a defined frame when operating the keys in runtime.

To configure the function:

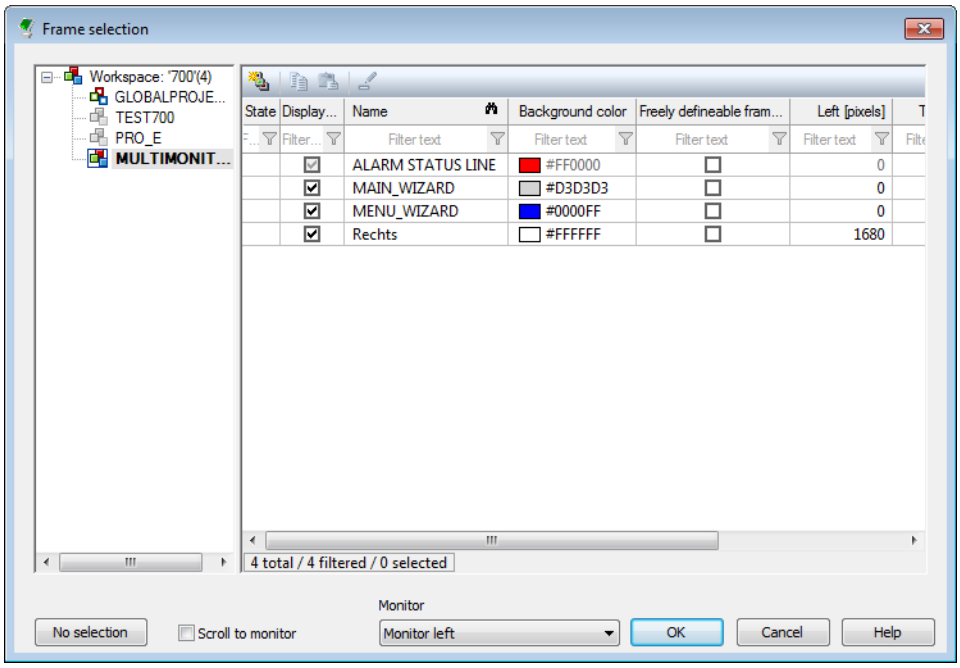
1. Select, in the list of functions, in the `Screens` node, the `Set focus to frame` function
2. The dialog for selecting a screen is opened
3. select the frame you wish to assign
4. For multi-monitor projects, select the virtual monitor for opening the frame

The frame with a focus is displayed with a frame in runtime. The line width and color of the frame are defined in the `Graphical design`/`Screens` node in project properties.

FRAME SELECTION DIALOG

In the **`Frame`** selection dialog, frames can be selected for the execution of functions, from:

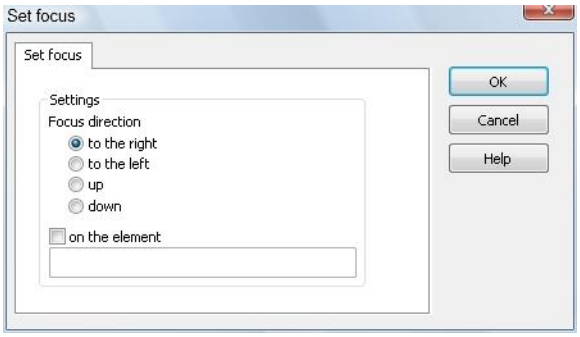
- Current project
- Subprojects
- All projects in the workspace with the **`Keep project in memory`** option active



Parameters	Description
Project tree window	Displays all projects in the workspace. Frames can be selected from the current project and from all projects with the **Keep project in memory** option active
Frames window	Selection of frame. If several frames are selected, the first selected frame is used for the execution of the function.
Show this dialog in the Runtime	Switches to this frame selection dialog when this function is executed in Runtime.
No selection	Has no function for this selection.
Scroll to monitor	Active: Scroll bars are shown in Runtime, which make it possible to scroll to the monitor. Only available with multi-monitor systems.
Monitor	Selection of the monitor from the drop-down list. This contains: ▸ **Current monitor** ▸ All virtual monitors defined in the monitor administration Optional: **All monitors** Only available with multi-monitor systems.
OK	Applies settings and closes dialog.
Cancel	Discards all changes and closes dialog.
Help	Opens online help.

18.7 Move focus

This function set the focus on a particular element in runtime with keyboard operation and can therefore be used to navigate within a frame.



Property	Action
`Direction`	Define the direction in which the focus should be moved. The following defined sequence applies in the editor in the context menu or under *Edit/change focus sequence* **...**
`on the element`	Definition of the element for the focus. Enter the object name of the element.

18.8 Focus: Delete from frame

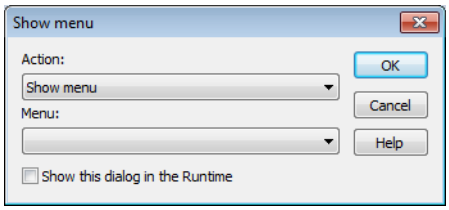
This function takes the focus from the current frame in runtime. To continue operating the keyboard, the focus must be set to a frame again.

18.9 Show menu

This function controls the display of main menus.

1. Create a new function with **`New function`**
2. In the **`Screens`** node, select the **`Display menu`** function

3. Define the action to be carried out and the main menu



Parameters	Description
Action	Selection of the action:
Show menu	The selected menu is displayed.
Hide menu	The selected menu is hidden.
Change menu	The current menu is hidden and the selected menu is displayed instead.
If no menu exists	The selected menu is displayed, if no menu is displayed at the moment.
Menu	Selection of menu.
Show this dialog in the Runtime	Active: This dialog is opened when the function is carried out in Runtime.

18.10 Monitor assign

In a multi-monitor-environment you can assign a certain virtual monitor to a single real monitor.

18.11 Close template

With this function, the selected frame (on page 242) with all screens that are based on it are closed. In contrast to the Close screen (on page 366) function, the precise screen name need not to be given in this case. If, with a multiple-monitor projects, a frame is switched to several monitors, it is possible to select which frame is closed on which monitor.

To configure the function:

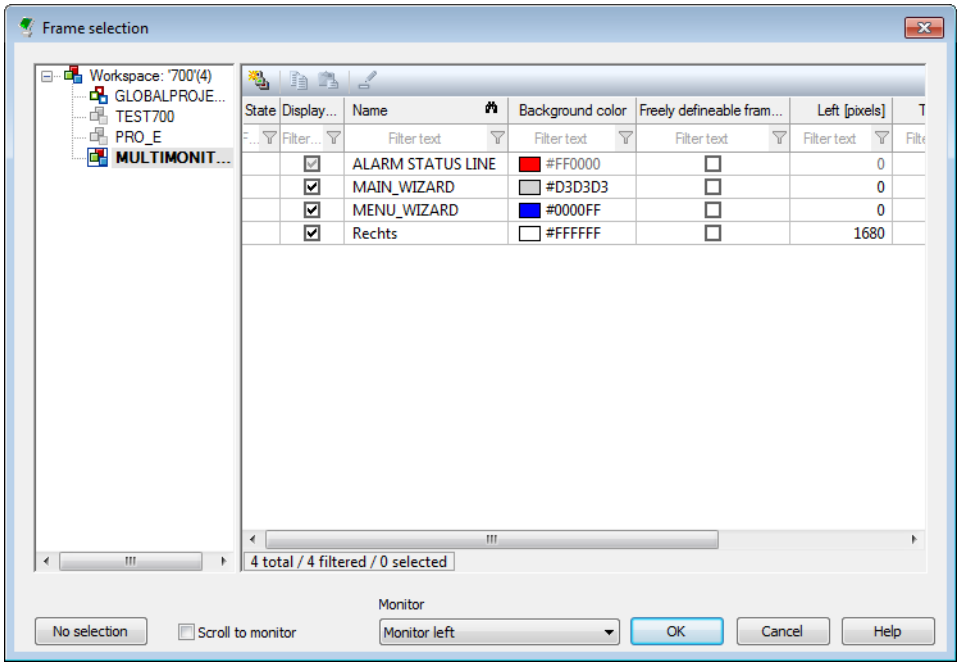
1. Select, in the list of functions, in the `Screens` node, the `Close frame` function
2. The dialog for selecting a screen is opened
3. select the frame you wish to assign
4. Select, with multiple-monitor projects, the virtual monitor on which the frame is to be closed, from the drop-down list:
   - all objects
   - Current monitor
   - Designated virtual monitor

FRAME SELECTION DIALOG

In the **`Frame`** selection dialog, frames can be selected for the execution of functions, from:

- Current project
- Subprojects

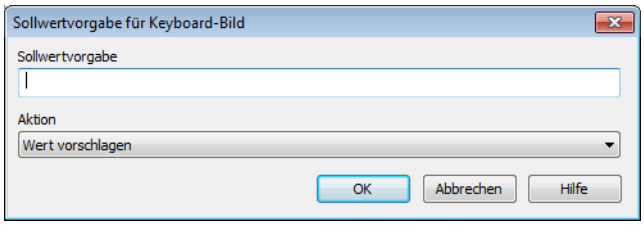
- All projects in the workspace with the **`Keep project in memory`** option active



Parameters	Description
Project tree window	Displays all projects in the workspace. Frames can be selected from the current project and from all projects with the **Keep project in memory** option active
Frames window	Selection of frame. If several frames are selected, the first selected frame is used for the execution of the function.
Show this dialog in the Runtime	Switches to this frame selection dialog when this function is executed in Runtime.
No selection	Has no function for this selection.
Scroll to monitor	Active: Scroll bars are shown in Runtime, which make it possible to scroll to the monitor. Only available with multi-monitor systems.
Monitor	Selection of the monitor from the drop-down list. This contains: ▸ **Current monitor** ▸ All virtual monitors defined in the monitor administration Optional: **All monitors** Only available with multi-monitor systems.
OK	Applies settings and closes dialog.
Cancel	Discards all changes and closes dialog.
Help	Opens online help.

18.12 Set point input for keyboard screen

The `Setpoint input for keyboard screen` function is not available for the keyboard screen. In addition the text field `Set value` must have been created. It makes it possible to set or set and send a previously defined value.



Parameters	Description
`Set point default`	Input of the target value.
`Action`	Selection of action from drop-down list. (for possible actions, see the following table.)
`OK`	Accepts inputs, closes dialog and creates functions with value and action.
`Cancel`	Discards inputs, closes dialog and creates functions without setpoint and action.
`Help`	Opens online help.

Actions in the drop-down list:

Action	Description
propose value	The setpoint default is written to the keyboard screen.
Accept value	The following happens: happens: ▸ the set value default is written to the keyboard screen ▸ The keyboard screen is closed with OK
RGM: Accept and send value	If the keyboard screen is called up from the Recipe Group Manger recipe table, then: ▸ The setpoint default is written to the keyboard screen ▸ The keyboard screen is closed with OK ▸ The recipe value is written to the variable
RGM: Accept value and save recipe	If the keyboard screen is called up from the Recipe Group Manger recipe table, then: ▸ The setpoint default is written to the keyboard screen ▸ The keyboard screen is closed with OK ▸ The recipe is saved
RGM: Write and send value and save recipe	If the keyboard screen is called up from the Recipe Group Manger recipe table, then: ▸ The setpoint default is written to the keyboard screen ▸ The keyboard screen is closed with OK ▸ The recipe value is written to the variable ▸ The recipe is saved

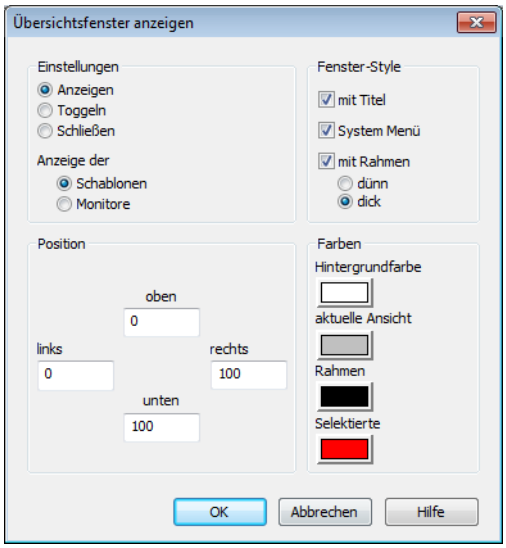
18.13 Display overview window

The `Show overview window` function displays the overview window in Runtime, which shows the real monitors or frames in a multi-monitor system. A monitor and/or a frame can be activated by means of a mouse click.

To configure the function:

1. Create a new function
2. Navigate to the node
3. Select the `Show overview window` function
4. the dialog for configuring the overview window is opened
5. Select your configuration
6. Close the dialog by clicking on OK
7. Allocate a button to the function

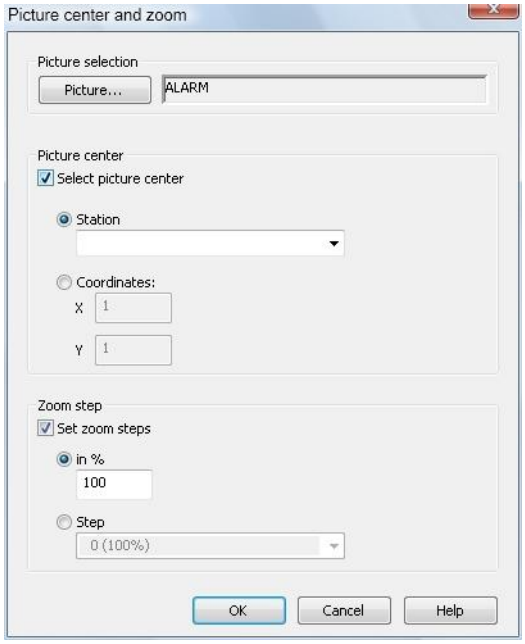
DISPLAYING THE OVERVIEW WINDOW



Parameters	Description
Settings	The function can either open, close or toggle the overview window. Selection is carried out by means of radio buttons:
Display	The overview window is opened.
Toggle	The display is switched between open and closed.
Close	The overview window is closed.
Display of	Selection of display in the overview window. Either frames or monitors are displayed.
Frames	The overview window divides the screen into frames.
Monitors	The overview window divides the screen into monitors.
Position	Position of the overview window on the screen, calculated in pixels from the upper left edge.
Window style	
With title	Active: The overview window has a Windows title bar.
System menu	Active: A system menu is displayed if the title bar is activated.
With frame	Active: Overview window is displayed with a border. The window size can be adjusted in Runtime by dragging the border. Selection of the border width by means of radio buttons:
Fine	Bold border.
Bold	Fine border.
Colors	Definition of the colors in the overview screen. Clicking on Color opens the palette.
Background color	Color of window background
Current view	Currently displayed frames/monitors.
Frame	Color of border.
Selected	Selected frames/monitors.
OK	Accepts settings, closes dialog and creates functions with assignment.
Cancel	Discards settings, closes dialog and creates functions with standard settings.
Help	Opens online help.

18.14 Screen: Move center

With this function, the screen center can be moved or the zoom factor can be changed in worldview function. As a requirement a screen has to be created, which is larger than the screen resolution. In the screen settings the option size From frame (on page 242) has to be deactivated and the size has to be defined directly.



Possible settings that can be configured are:

Screen	Selection of the screen for the new center
Select center of screen	Settings for the center
Object name	Set desired center via an object from the object list
Coordinates	Set desired center via coordinates (x,y)
Set zoom steps	Settings for the zoom factor
in %	Set desired zoom factor in percent
Level	Set desired zoom factor via the defined zoom steps