zenon
by COPA-DATA

zenon manual

Process Recorder

v.8.10

© 2019 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. Technical data is only used for product description and are not guaranteed qualities in the legal sense. Subject to change, technical or otherwise.

Contents

1 Welcome to COPA-DATA help ... 4

2 Process Recorder ... 5

3 Introduction ... 5

3.1 Recording ... 6

3.2 Replay ... 6

4 Process Recorder data - recording ... 8

4.1 Process Recorder file (.rec) ... 9

4.2 Process Recorder data - recording ... 10

5 Engineering in the Editor ... 11

5.1 Variables and data types ... 11

5.2 Process Recorder screen ... 13

5.2.1 Create Process Recorder screen ... 13

5.2.2 Control elements for Process Recorder screens ... 14

5.3 zenon functions ... 20

5.3.1 Activate/deactivate project simulation ... 20

5.3.2 Screen switch Process Recorder ... 25

5.3.3 Process Recorder: Activate/deactivate playback ... 34

5.4 Display of changes to a project configuration ... 36

5.4.1 Configuration in the Editor ... 36

6 Process Recorder in the Runtime ... 37

7 Process Recorder in the zenon network ... 38

8 LOG entries ... 39

1 Welcome to COPA-DATA help

ZENON VIDEO-TUTORIALS

You can find practical examples for project configuration with zenon in our YouTube channel (https://www.copadata.com/tutorial_menu). The tutorials are grouped according to topics and give an initial insight into working with different zenon modules. All tutorials are available in English.

GENERAL HELP

If you cannot find any information you require in this help chapter or can think of anything that you would like added, please send an email to documentation@copadata.com.

PROJECT SUPPORT

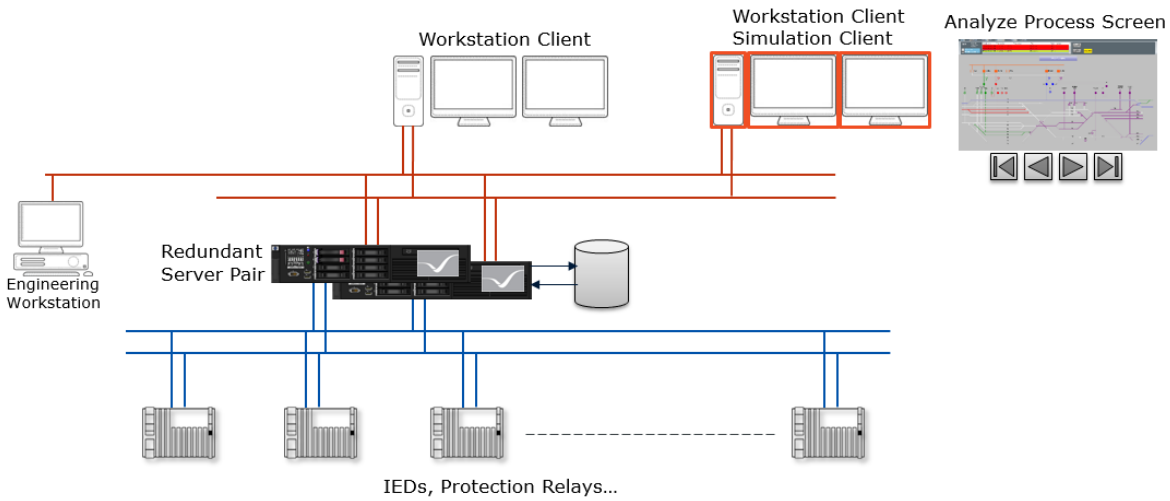
You can receive support for any real project you may have from our Support Team, who you can contact via email at support@copadata.com.

LICENSES AND MODULES

If you find that you need other modules or licenses, our staff will be happy to help you. Email sales@copadata.com.

2 Process Recorder

The **Process Recorder** module offers you the possibility to record process data in the productive Runtime. At a later point in time, this recorded data can be visualized in Runtime in a project simulation client.



Generally:

- Selected variables are recorded during the course of the productive process.
- Recording is carried out in the event of value changes.
- The recording contains limit value breaches and flashing status.
- These recordings can be played back in zenon Runtime in simulation mode.
- Time and time period of this playback can be freely selected.
- Existing zenon screens can be used.
- If playback is started in a Runtime with several projects, all recorded values from all projects are loaded, correctly sorted in chronological order and played back.
- The module is integrated into the zenon network and also supports redundancy.

3 Introduction

The module **Process Recorder** consists of two parts:

1. When configuring a project in zenon, variables for logging are activated.

   These variables are logged in Runtime during the course of the productive process.
2. The recorded data is played back again in Runtime by means of project simulation.

- This recorded data can be called up at any time and played back. This playback is carried out in simulation mode of Runtime outside of the productive process.
- With this playback, recorded processes can be visualized or analyzed.
- Existing zenon screens can be used for this playback. Because each desired zenon screen is used for playback, processes from the past can be traced in the working environment.
- Changes to the configuration of screens are displayed on playback.
  Note: For correct display in playback mode the general project property **Versioning active** must be activated when configuring.
- Screens that have only been configured by the person configuring the project for playback can also be used for the analysis.

3.1 Recording

Data from the ongoing process is continually recorded and saved.

The recording is activated for each individual configured variable or for a data type via the property **Activate recording** in the property group **Process Recorder**.

3.2 Replay

Recorded data from the productive environment can be played back at a later time:

- The recorded data can be played back for each recorded point in time.
- This playback is carried out in simulation mode of Runtime outside of the productive process.
- With this playback, recoded processes can be visualized or analyzed.
- Existing zenon screens can be used for this playback. Because each desired zenon screen is used for playback, processes from the past can be traced in the working environment.
- Playback is controlled with a zenon screen of the *Process Recorder* type.
- Screens which were projected for playback only by the project manager can also be used for analysis.

The following is applicable for playback:

- Display in playback is carried out solely on the basis of the recorded values. Current process data is not visualized during playback.
- Only screens are supplied with recorded values. All other zenon modules do not receive any data.

EXCEPTIONS

- **Internal driver**

  Internal variables that have not been activated for recording in the process recorder are (in addition to the recorded values) supplied with current values, even during playback.

- **System driver**

  System driver variables that have not been activated for recording in the Process Recorder are also supplied with current values during playback.
  This is applicable for variables of the following themes:

  - **Batch Control**
  - **Command Processing**
  - **Custom**
  - **User Administration**
  - **Printer**
  - **Folder**
  - **Project information**
  - **Command Sequencer**
  - **System information**

REQUIREMENT

When starting the replay in zenon Runtime, the appropriate file for the start project is selected at the given start time. A corresponding LOG message is created and can be displayed in the Diagnosis Viewer tool.

The appropriate file:

- Ends with ***.rec**
- is located in the simulation data folder on the executing computer **ProcessRecorder**
- is the file,
  - whose timestamp is closest to the selected start time of playback
  - and whose timestamp + 5 minutes still contains the selected point in time.

If no file meets these criteria, no file is loaded and a LOG message is also written.

Information

You can find more detailed information on the files used by the Process Recorder module in this manual in the chapter Process Recorder Data - Recording (on page 8).

You can find further information on the save location in the Runtime manual in the File structure chapter.
Note: You can open the local save location of the Runtime files in the zenon Editor with the **Ctrl+Alt+R** keyboard shortcut.

4 Process Recorder data - recording

To have process data available for subsequent playback or evaluation in the **Process Recorder** module, all variables are saved in files with the **Activate recording** property activated.

Recording is carried out in two files:

- Recorder File (**.rec**) (on page 9)
  - each time there is a value change
  - each time the status bit of the variable is changed
  - each time a time stamp is changed
- Additional Data file (**.lmt**) (on page 10)
  - In the event of a limit value breach
  - In the event of a change to the flashing status

Applicable to both files:

- The respective new value is written to the file. If there is still no value when Runtime is started, no entry is written.
- A new file is saved automatically every five minutes. This cycle is fixed and cannot be configured.
- The current recording file is only opened if required and closed again after writing.
  As a result, the file sync between server and standby is not blocked, because the file is only blocked for a short time.

The requirements for available memory increase. The longer a recording runs, the more memory space is needed.

Attention

Ensure that you have sufficient storage space.
Also note the quality of the storage location: It is explicitly not advised to use removable storage devices.

4.1 Process Recorder file (.rec)

Process Recorder files have a *.rec* file suffix.
The following are saved in this file:

- Current value of the variable
  Signal values of the variable are always saved, not the values calculated by the value amendment.
  This corresponds to the zenon Editor properties **Value range PLC** for the signal value and **Value adjustment linear** for the value amendment.
- Status bits of the variable
- Time stamp of the variable
  The time stamp of the variables are saved as UTC and have a precision of milliseconds.
  - External time stamp
  - Internal time stamp
  - Process Recorder time stamp
    This time stamp is only used for playback in the **Process Recorder** module. It is issued automatically and cannot be actively configured or modified.

The following is applicable for the REC file:

- Recording is carried out in an REC file in the data directory of the computer. The file is named automatically. The time stamp of the file creation plus a fixed, defined prefix, *PR*, is used to name the file: *PRYYMMDDHHMMSS.rec*.
  Example: *PR070203233651.lmt* for February 3, 2017, 23:36:51
- Save location of the Process Recorder files:
  The data for subsequent playback in the Process Recorder is saved in the data directory of the local computer.
  Save location of the logging files: ../[project name]/[computer name]/[project name]/ProcessRecorder/PRJJJJ.
  The *PRYYYY* folder is created automatically by zenon and contains all Recorder files from a year.
  You can find further information on the save location in the Runtime manual in the File structure chapter.

Note: You can open the local save location of the Runtime files in the zenon Editor with the **Ctrl+Alt+R** keyboard shortcut.

4.2 Process Recorder data - recording

Files with additional data have the file suffix *.lmt*.
The following are saved in this file:

- Variable ID
  Unique ID of the variable of the limit value breach
- Internal time stamp of the limit value breach
  The time stamp of the variables are saved as UTC and have a precision of milliseconds.
  - Process Recorder time stamp
    This time stamp is only used for playback in the **Process Recorder** module. It is issued automatically and cannot be actively configured or modified.
- Index breached
- Flashing status
  Status of the flashing (*Flashing yes*/*Flashing no*)

The following is applicable for the AdditionalData file:

- Recording is carried out in a LMT file in the data directory of the computer. The file is named automatically. The time stamp of the file creation plus a fixed, defined prefix, *PR*, is used to name the file: *PRYYMMDDHHMMSS.lmt*.
  Example: *PR070203233651.lmt* for February 3, 2017, 23:36:51
- Save location of the Process Recorder files:
  The data for subsequent playback in the Process Recorder is saved in the data directory of the local computer.
  Save location of the logging files: ../[project name]/[computer name]/[project name]/ProcessRecorder/PRJJJJ.
  The *PRYYYY* folder is created automatically by zenon and contains all Recorder files from a year.
  You can find further information on the save location in the Runtime manual in the File structure chapter.
  Note: You can open the local save location of the Runtime files in the zenon Editor with the **Ctrl+Alt+R** keyboard shortcut.

5 Engineering in the Editor

Configuration steps for the **Process Recorder** module:

1. Activate the module.
   To do this, activate the general project property **Activate Process Recorder** in the node **Process Recorder**.
2. Activate the property **Activate recording** for the variables that you want to record.
   To do this:
   a) Highlight the variable(s) that you want to record.
      Note: Multiple selection is possible.
   b) Activate, in the **Process Recorder** properties group, the **Activate recording**.
      Note: This property can also be configured for data types. This setting can be passed on to all variables of this data type.

   You can find further information on this in the Variables and data types (on page 11) chapter.
3. Configure a zenon *Process Recorder* screen.
   You can find further information on this in the Process Recorder screen (on page 13) chapter.
4. Configure a screen switch function (on page 25) or the **Activate/deactivate project simulation** functions.
   You can find further information on this in the **Activate/deactivate project simulation** (on page 20) chapter.
   You can find additional information in the project simulation manual.

5.1 Variables and data types

To record variables for subsequent playback, this must be activated with the property **Activate recording** for recording. You can find this property in the **Process Recorder** properties group of the variable or the data type.

When being created, variables inherit the configuration of the data type property.

The variables with parameters set as such are *advised* when Runtime starts. The variables are *unadvised* again when Runtime is ended.

INHERITANCE

Variables and data types take on an existing parameter setting from the higher-level data type.

Note:

- Data types take on the parameters of the property at the time the data type is newly created. Changes to the original data type are not applied.

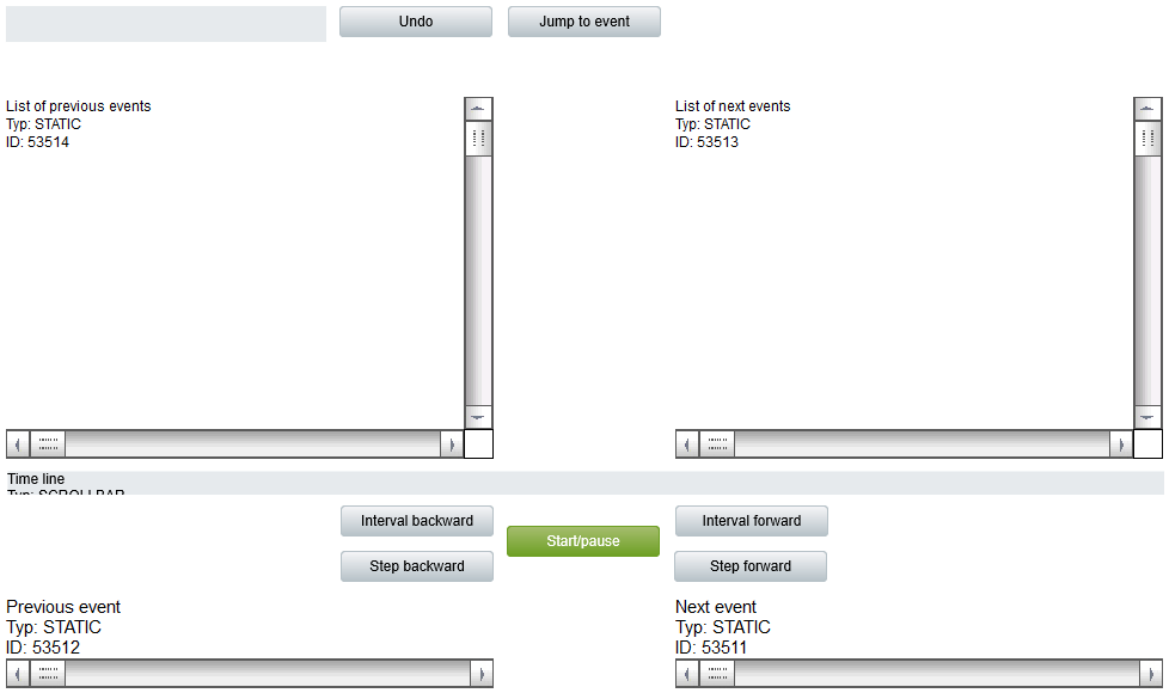
- Variables take on the parameters of the property from the assigned data type.
  Changes to the data type are applied for the variable:
  - If the **Activate recording** property has been applied by the data type and not amended, all changes to the data type are applied for the variable.
  - If the **Activate recording** property has been amended for the variable, changes to the data type are no longer applied for the corresponding variable.
    Attention: This is also applicable if the property corresponds to the parameter settings of the data type again after several changes. A change to the data type is not applied to the variable in this case.
- If a variable is assigned a new data type, the current parameter setting of the variables remains. This is then independent of the parameter settings of the (newly-assigned) data type.
- Arrays: The Arrays behavior corresponds to the behavior of the variables.
- Structure data types: These are supported. The behavior corresponds to behavior of the data types.

> **Hint**
>
> If the properties of a variable have been amended, the parameter setting can be transferred from the data type again with the *Link all properties with data type* context menu entry.

APPLY CHANGES FOR THE RECORDING

If the existing configuration of one or more variables is subsequently changed, the recording is amended accordingly after the project is reloaded in Runtime. Newly activated variables are only available for use in the process recorder from the time they are activated and after the project has been reloaded in Runtime. The same apples for amendments to a data type.

5.2 Process Recorder screen



The screen of the *Process Recorder* type is for controlling playback of recorded process data in Runtime.

The recorded data is played back in zenon in simulation mode.

> **Hint**
>
> Configure the Process Recorder screen in its own template.

5.2.1 Create Process Recorder screen

The *Process Recorder* screen is for control when playing back recorded process data in zenon Runtime.

ENGINEERING

Two procedures are available to create a screen:

- The use of the screen creation dialog
- The creation of a screen using the properties

Steps to create the screen using the properties if the screen creation dialog has been deactivated in the menu bar under **Tools**, **Settings** and **Use assistant**:

1. Create a new screen.

   To do this, select the **New screen** command in the tool bar or in the context menu of the **Screens** node.

2. Change the properties of the screen:

   a) Name the screen in the **Name** property.

   b) Select *Process Recorder* in the **Screen type** property.

   c) Select the desired frame in the **Frame** property.

3. Configure the content of the screen:

   a) Select the **Elements (screen type)** menu item from the menu bar.

   b) Select *Insert template* in the drop-down list.
   The dialog to select pre-defined layouts is opened. Certain control elements are inserted into the screen at predefined positions.

   c) Remove elements that are not required from the screen.

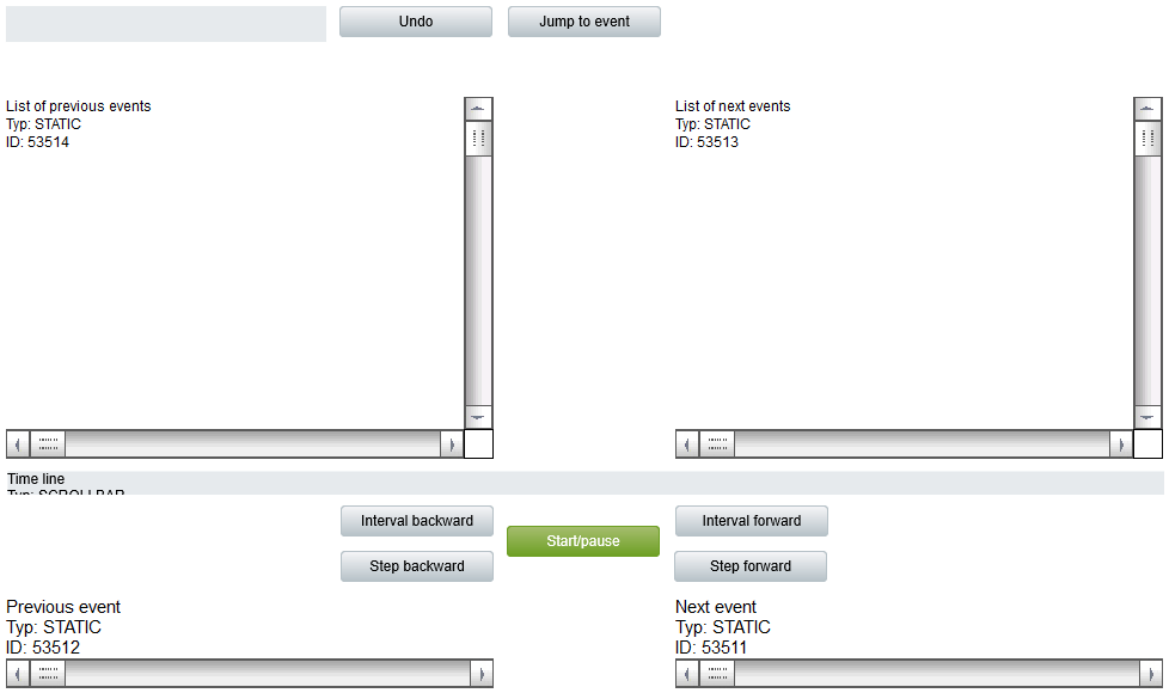
   d) If necessary, select additional elements in the **Elements** drop-down list. Place these at the desired position in the screen.

4. Create a screen switch function.
   Configure this screen switching function.
   You can find further information in relation to this in the Process Recorder screen switching (on page 25) chapter.

5.2.2 Control elements for Process Recorder screens

There are different control elements available in Process Recorder screens.

If you want to edit the list directly using the monitor, activate the Multi-Touch functionality. You can find detailed information in relation to this in the Configure interactions chapter.



Control element	Description
Insert template...	Opens the dialog for selecting a template 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 position 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 placed in the zenon screen. Elements can be moved on the screen and arranged individually.

REPLAY

Control element	Description
Playback position	Shows the point in time of playback in local time. Empty if Runtime is not in playback mode.

Control element	Description
	Format: *DD.MM.YYYY HH:MM:SS.MMM* **Note:** The value is N/A if no recorder file is loaded.
Undo	Cancels the reloading of recorded data. Playback is continued at the current point in time.

START/PAUSE

The button is named **Start/Pause**.

The display of this button in Runtime depends on whether playback is currently running or not.

- **Pause**:
  In playback mode, with playback running
- **Start**:
  Playback mode has not been started yet

Control element	Description
Start	Starts playback mode. Not visible in playback mode.
Pause	Pauses playback mode. Not visible with playback mode paused.

SINGLE STEP

Clicking on the button jumps to playback of the next or previous value change. The next or previous value change is listed in the event preview.

Control element	Description
Step forward	Switches to the next value change for playback. Not active in playback mode.
Step backward	Switches to the previous value change for playback. Not active in playback mode.

INTERVAL STEP

Clicking on the button jumps forwards or backwards by the configured interval.
Note: This interval step is always executed, regardless of whether these value changes are present or not after this interval jump.

Control element	Description
Interval forward	Jumps forward by the configured interval during playback. The interval is configured in the screen switching function (on page 26). Not active in playback mode.
Interval backward	Jumps back by the configured interval during playback. The interval is configured in the screen switching function (on page 26). Not active in playback mode.

LIST OF EVENTS

List of the next or previous value changes of the recording. The events within the current interval are displayed.
Playback jumps to the time of the selected event by selecting an event in the list and clicking on the **Jump to event** button.

Note: With playback active, these lists are blocked and are not updated.
List window in Runtime:

- The columns can be sorted and filtered in Runtime.
- Filtering and sorting can be saved in the filter profiles.
- The column width can be amended by holding down the right mouse button.
- Columns can be rearranged by means of drag&drop.

The following is applicable for the following control elements:

Longer texts can also be displayed in the Runtime over several lines using the **Automatic word wrap** property.

In the Editor, go to **Representation** in the properties of the respective list properties and activate the checkbox of the **Automatic word wrap** property.

The line height must be amended manually.

Control element	Description
Next event	Display of the next event for playback.
Previous event	Display of the previous event for playback.
List of next events	List of the next value changes in playback. The display of the list is configured in the screen switch function (on page 26). **Note:** The display of this list is empty in Runtime if the recording file is empty. The entries in this list are filtered for the duration of the interval.
List of previous events	List of the previous value changes of the playback. The display of the list is configured in the screen switching function (on page 26). **Note:** The display of this list is empty in Runtime if the recording file is empty. The entries in this list are filtered for the duration of the interval.

JUMP TO EVENT

Skips through playback to the time of an event selected in a **preview list**. If there are several results or no results from a preview list selected, the button is grayed out in Runtime.

TIME LINE

The time list visualizes the time period of playback. Positioning in the zenon screen is possible horizontally or vertically.

The position of playback can be moved forwards and backwards in Runtime with the mouse pointer. To do this, move the slide control in the scroll bar with the mouse button held down. This positioning is possible in each mode of the Process Recorder. Playback is then continued at the position of the time line. To stop moving, move the mouse out to the to the display area. The slide control is then placed in the initial position again.

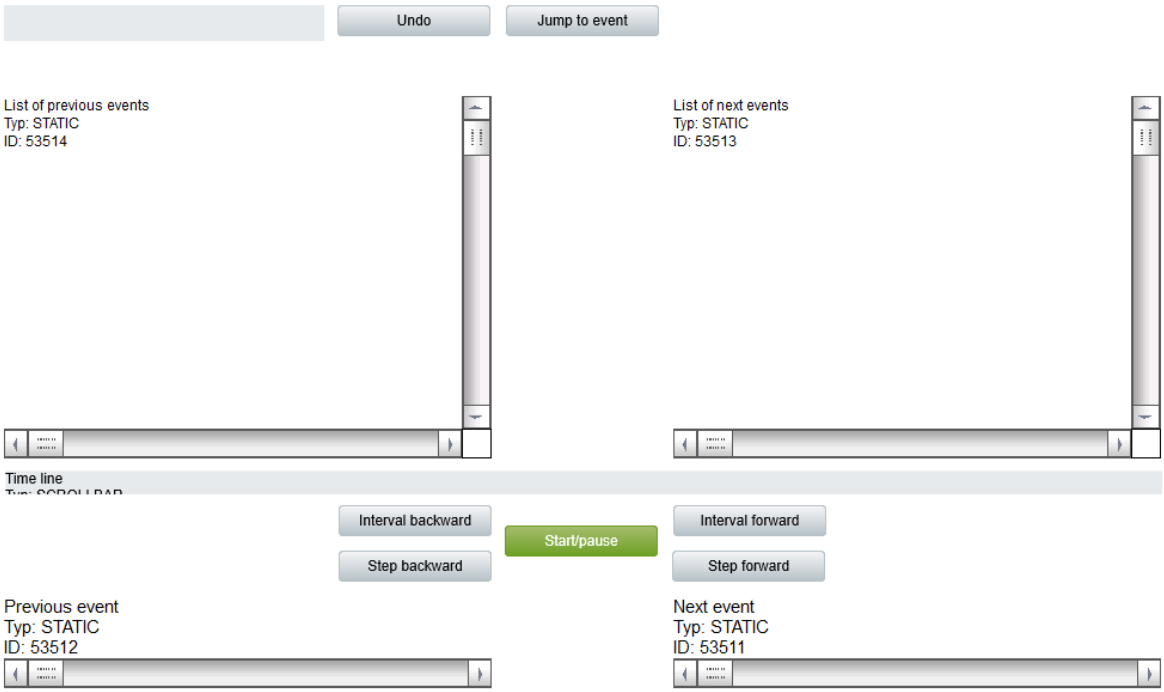
The following keyboard shortcuts are supported:

Key	Description
End	Jumps to the end of the playback time period.

Key	Description
Pos1	Jumps to the start of the playback time period.
Page Down	Jumps, during playback, forward to the configured time of an interval.
Page Up	Jumps, during playback, backward to the configured time of an interval.

Note: The start and end of a playback period can also be shown in Runtime with the **system driver** variables *[Process Recorder] start of playback period* and *[Process Recorder] end of playback period* in Runtime.

5.2.2.1 Operation in Runtime



The following is applicable for display in the Runtime:

- If zenon Runtime is not running in Process Recorder playback mode, all screen elements of the **Process Recorder** screen type are grayed out or empty.
- The lists of the **event preview** are updated dynamically.
- The recorded values of all loaded projects are taken into account for playback.
  These values are shown compiled in the lists.

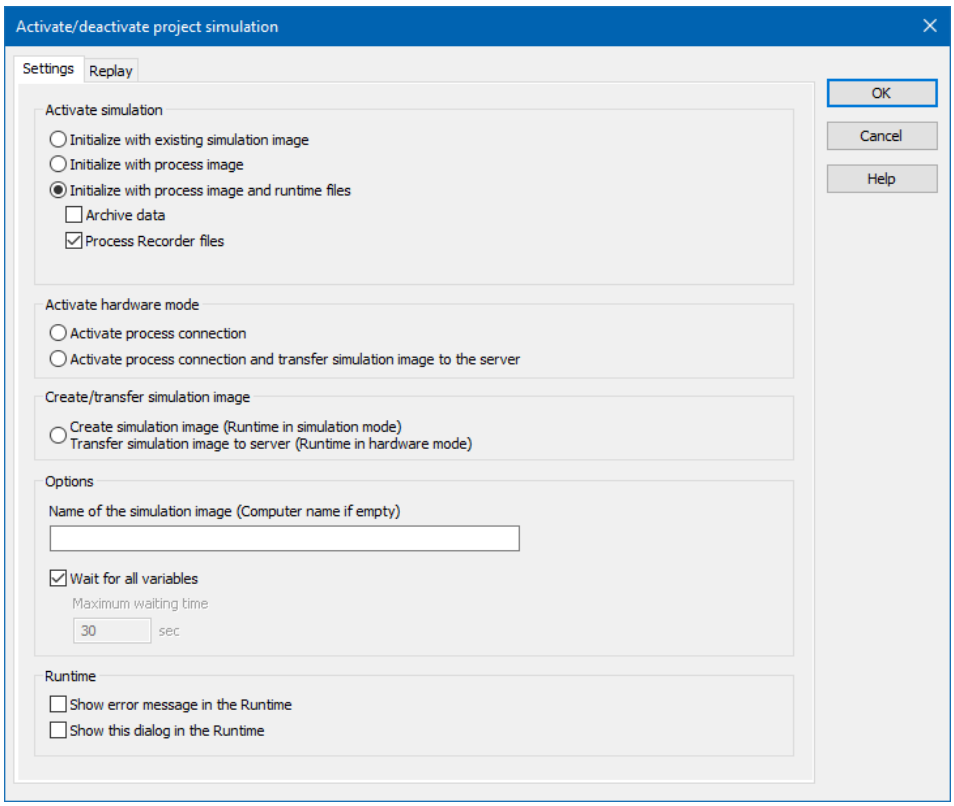
- When calling up a screen in playback mode, the recorded values are loaded from the Recorder files into the memory and buffered. The Process Recorder module detects if a further Recorder file needs to be loaded and loads this into the buffer in the background.
  - This ensures that in Runtime jumping forwards or backwards to events or at an interval (via buttons on the Process Recorder screen) is possible without delay.
  - However, if the loading of a Recorder file takes longer, this is visualized in Runtime by a dialog. The updating of the values is delayed briefly as a result.
- If, during ongoing playback, the **Activate/deactivate Process Recorder playback** function is called up with a new point in time, playback is paused.
- If a step is executed, a corresponding LOG entry (on page 39) is created.

5.3 zenon functions

5.3.1 Activate/deactivate project simulation

You start the project simulation with the **Activate/deactivate project simulation** function. In this chapter, you get information about this function that is relevant for the **Process Recorder** module.

You can find further information on this function in the project simulation manual, in the Activate/deactivate project simulation chapter.



INITIALIZE WITH PROCESS IMAGE AND RUNTIME FILES

Parameter	Description
Process Recorder files	When starting the project simulation, the pre-existing logging files are used for the simulation. **Note:** Not active if Initialize with process image and Runtime files is not active.
Start the Process Recorder in playback mode	The project simulation starts in the playback mode of the Process Recorder. This makes it possible for the existing recordings to be visualized in Runtime. **Note:** Only available if Process Recorder files is active. As an alternative to this property, the **Process Recorder: activate/deactivate playback** (on page 34) function can also be used at the start of playback.

Note: The current playback mode is also shown in the **system driver** variable *[Process Recorder] Recorder mode*.

FOR VISUALIZATION

ENGINEERING

Steps to create the function:

1. Create a new function:

   In the toolbar or in the context menu of the Functions node, select **New function**.
   The dialog to select a function is opened.
2. Navigate to the node **Application**.
3. Select the **Activate/deactivate project simulation** function.

   The dialog to set parameters for project simulation is opened.
4. Select:

   a) **Initialize with process image and Runtime files**

   b) Activate the **Process Recorder files**

   c) Optional: To start the simulation in Runtime directly after calling up the function in playback mode, configure the settings in the Playback tab.

5. In addition, you can configure the **Options** and **Runtime** properties.
6. Name the function in the **Name** property.

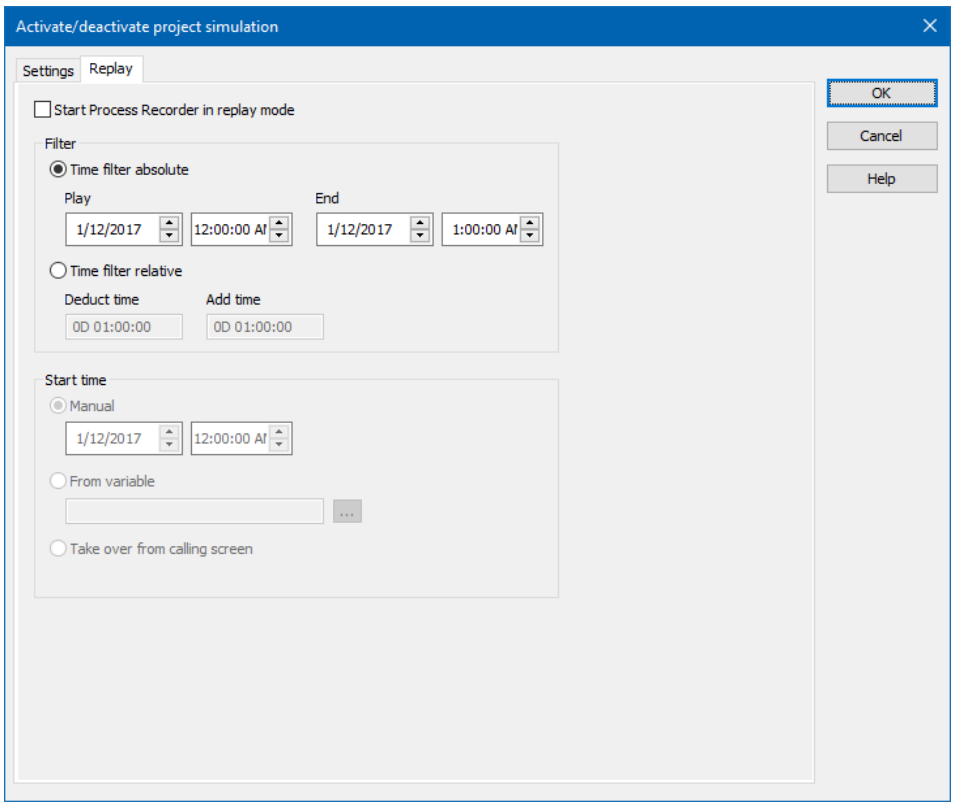
> **Information**
>
> In simulation, the **Process Recorder** does not record any further data.

5.3.1.1 Replay

In the **Playback** tab, you configure parameters for the playback mode of the **Process Recorder** module. If this dialog is shown in Runtime, only the time requirements can be amended.

The settings of this tab are optional.

They only make sense if a switch from live mode direct to playback is to be made.



Parameter	Description
Start Process Recorder in replay mode	Checkbox to select the Runtime behavior after the function has been executed.

Parameter	Description
	▸ *Active*: Runtime starts in playback mode. ▸ *Inactive*: Runtime is started in simulation mode. Default: *inactive*

FILTER

In this area you can configure the start time of playback in Runtime.

Parameter	Description
Time filter absolute	Start and end of the playback area are given manually. Entry of the start time of playback in input fields, separated according to start and end. Format: ▸ Date: *DD.MM.YYYY* Default: *Current date* ▸ Time: *hh:mm:ss* Default: *00:00:00* Default: *active* **Note:** not active if the filter type is **Time filter relative**.
Time filter relative	Start of playback range, relative to the start time. The start time is configured in the **Start time** area. Entry of the difference in input fields. ▸ Deduct time: *d hh:mm:ss*. ▸ Add time: *d hh:mm:ss*. Default: *inactive* **Note:** not active if the filter type is **Time filter absolute**.

START TIME

The options are then only available if **relative time filter** is activated as a filter. Otherwise the options are grayed out. Selection of the start time from an option field.

Parameter	Description
Manual	The start time of playback for the **Process Recorder** module is entered manually. Configuration of the start time with entry into both Default: *Date of day / 00:00:00*
From variable	The start time of playback for the **Process Recorder** module is then taken from the value of the configured variable. Click on button **...** in order to open the dialog for selecting variables. The DINT data type is recommended for the linked variable. If the variable does not have a valid value or has the value 0, the function is not executed! A corresponding LOG entry is created in the process.
Take over from calling screen	The playback start time in Runtime in the **Process Recorder** module is taken from the calling screen. Note: ▸ The **Activate/deactivate project simulation** function can be successfully called up from a zenon *Alarm Message List* or *Chronological Event List* screen. ▸ For calling up, only one entry (alarm or event) from the list can be selected. The incoming time can be used as a start time for playback in the Process Recorder module. If the function cannot be successfully started using **Take over from calling screen** a corresponding LOG entry is written.

CLOSE DIALOG

Option	Description
OK	**Applies all changes in all tabs and closes the dialog.**
Cancel	Discards all changes in all tabs and closes the dialog.
Help	Opens online help.

5.3.2 Screen switch Process Recorder

To open a screen from the Process Recorder version in Runtime:

1. Configure a screen of type *Process Recorder screen* (on page 13).
2. Create a function Screen switch for this screen.
3. Define the desired filter properties

CREATE A SCREEN SWITCH FUNCTION

A *screen switching* function is for calling up screens in Runtime. For screen switching to a Process Recorder screen, you can also configure the step size of the interval as well as the graphical appearance of the lists in the event preview.

ENGINEERING

Steps to create the function:

1. Create a new function:

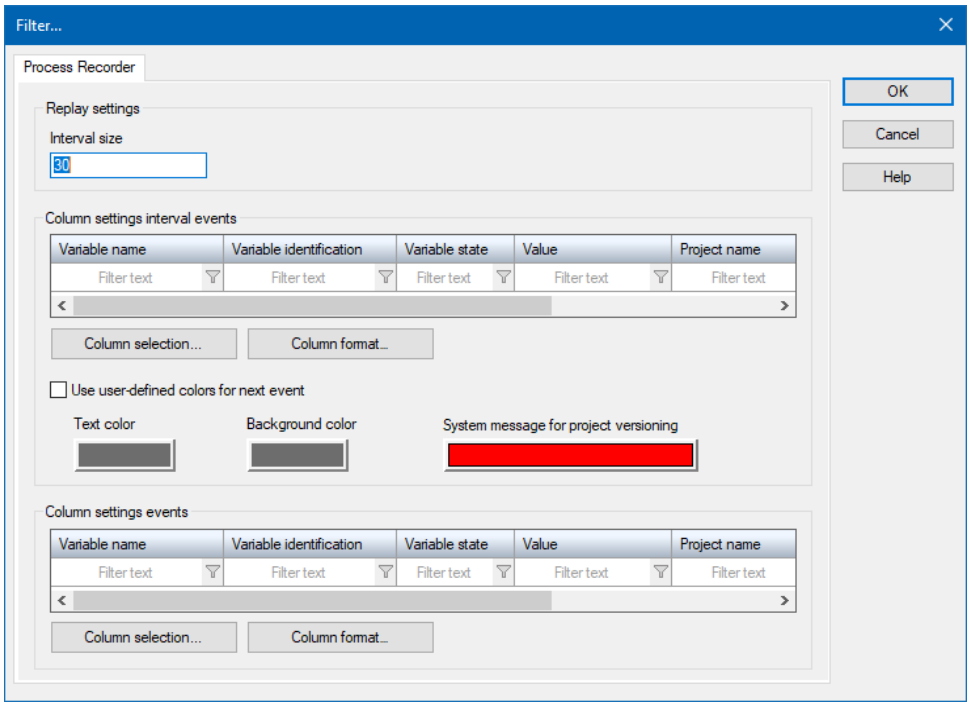
   In the toolbar or in the context menu of the Functions node, select **New function**.
   The dialog to select a function is opened.
2. Navigate to node **Screens**
3. Select the *Screen switching* function

   The dialog for selecting a screen is opened.
4. Select the desired screen.
   Note: If you select a screen from another project, ensure that the project is running in the Runtime.
5. Confirm your selection by clicking on the **OK** button.
   The **Filter** (on page 26) dialog to configure the playback settings and for the graphical appearance of the interval and event list is opened.
6. Name the function in the **Name** property.

5.3.2.1 Process Recorder screen switching filter

In this dialog, you configure:

- The step size (in seconds) of an interval step
- The graphical appearance of the list in the event preview



REPLAY SETTINGS

Parameter	Description
Interval size	Entry of the interval size. Entry in seconds. Default: *30*

COLUMN SETTINGS INTERVAL EVENTS

Configuration of the display in Runtime for **event list forwards** and **event list backwards**.

Parameter	Description
Preview	Preview of the configured display. **Note:** Configuration by means of mouse click, context menu or the **Column selection...** and

Parameter	Description
	Column format... buttons
Column selection...	Opens the dialog for selecting the columns which should be displayed (on page 29).
Column format...	Opens the dialog for configuring the column formats (on page 30).

USER-DEFINED COLORS

For the next and previous result in the event preview list and in the interval list, text and background color can - in contrast to configuration for the complete list - be configured separately.

The color for the entry in the event of a version change can be configured with an additional option.

Attention: In order for the configuration of the colors of the options in this area to be able to be visualized in playback mode, it is absolutely necessary to activate the checkbox for the **Use user-defined colors for next event** entry.

Parameter	Description
Use user-defined colors for next event	If this option is activated, only specially-configured appearances configured are used for the next or previous entry in the list. All other entries in the list use the settings as configured in Column format (on page 30). Activation by means of checkbox.
Text color	Color for the text display for the next event in the event preview list. Clicking on the color opens the color palette to select a color. **Note:** not active if **Use user-defined colors for next event** is inactive.
Background color	Color for the background color of the next event in the preview list. **Note:** not active if **Use user-defined colors for next event** is inactive.
System message for project versioning	Text color of the system message that is displayed in playback mode if a change to the project configuration is detected in the playback period. In doing so, the prescribed "New project version"

Parameter	Description
	text is added as an entry in the list.

COLUMN SETTINGS EVENTS

Configuration of the display in Runtime for **previous event** and **next event**.

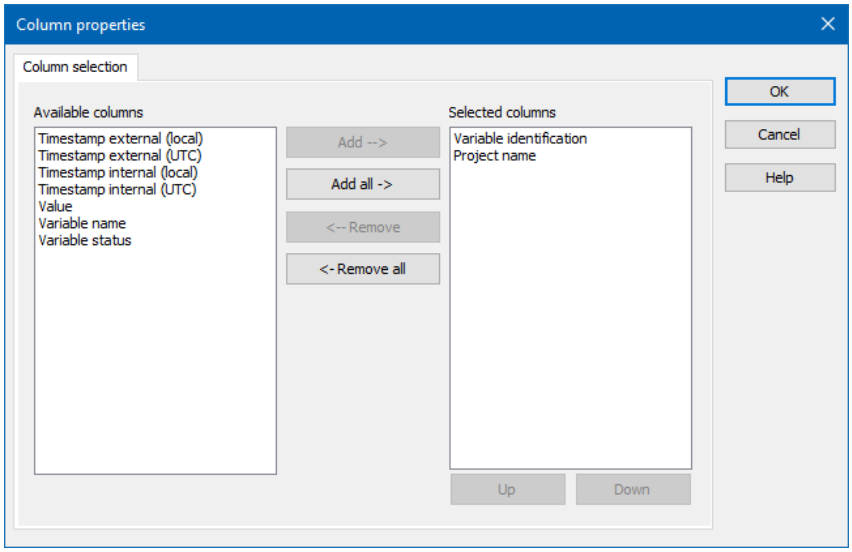
Parameter	Description
Preview	**Note:** initial amendments can already be configured in this preview by clicking the mouse or by means of the context menu.
Column selection...	Opens the dialog for selecting the columns which should be displayed.
Column format...	Opens the dialog for configuring the column display.

CLOSE DIALOG

Option	Description
OK	**Applies settings and closes the dialog.**
Cancel	Discards all changes and closes the dialog.
Help	Opens online help.

5.3.2.2 Column selection

You can configure the columns to be displayed in Runtime here



Option	Function
Available columns	List of columns that can be displayed in the table.
Selected columns	Columns that are displayed in the table.
Add ->	Moves the selected column from the available ones to the selected items. After you confirm the dialog with OK, they are shown in the detail view.
Add all ->	Moves all available columns to the selected columns.
<- Remove	Removes the marked columns from the selected items and shows them in the list of available columns. After you confirm the dialog with OK, they are removed from the detail view.
<- Remove all	All columns are removed from the list of the selected columns.
Up	Moves the selected entry upward. This function is only available for unique entries, multiple selection is not possible.
Down	Moves the selected entry downward. This function is only available for unique entries, multiple selection is not possible.

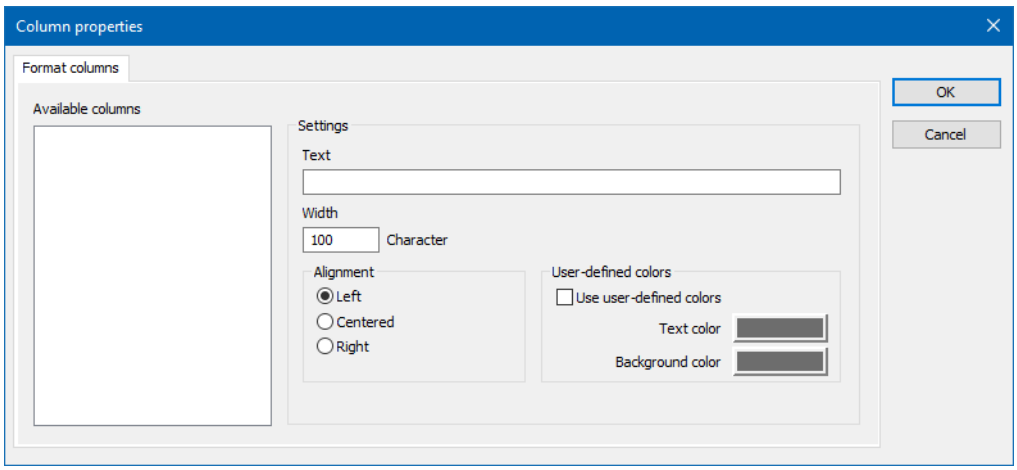
CLOSE DIALOG

Options	Description
OK	**Applies settings and closes the dialog.**
Cancel	Discards all changes and closes the dialog.
Help	Opens online help.

5.3.2.3 Column Format

Graphical appearance of the selected columns.

Configuration of the properties of the columns for configurable lists. The settings have an effect on the respective list in the Editor or - when configuring screen switching - in Runtime.



AVAILABLE COLUMNS

Option	Description
Available columns	List of the available columns via **Column selection**. The highlighted column is configured via the options in the **Settings** area.

SETTINGS

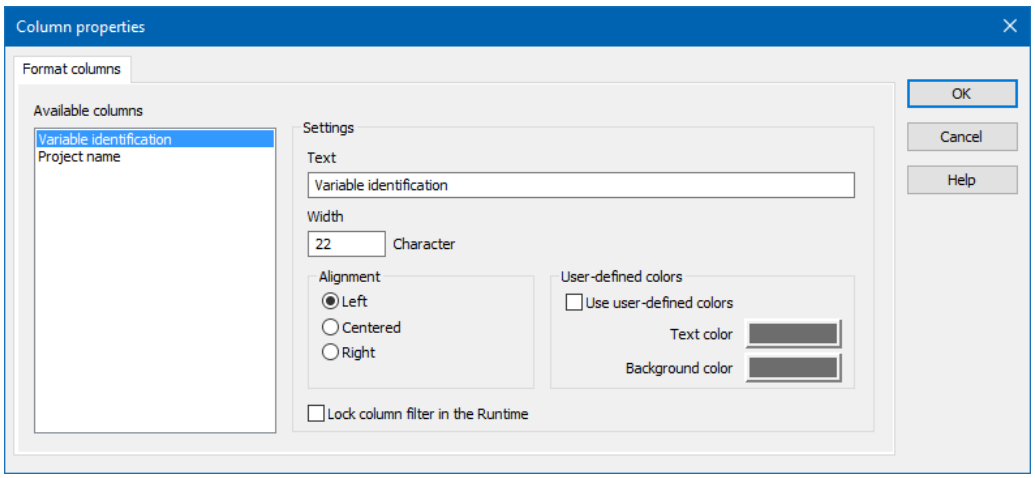
Option	Description
Settings	Settings for selected column.
Labeling	Name for column title.

Option	Description
	The column title is online language switchable. To do this, the **@** character must be entered in front of the name.
Width	Width of the column in characters. Calculation: Number time average character width of the selected font.
Alignment	Alignment. Selection by means of radio buttons. Possible settings: ▸ **Left**: Text is justified on the left edge of the column. ▸ **Centered**: Text is displayed centered in the column. ▸ **Right**: Text is justified on the right edge of the column.
User-defined colors	Properties in order to define user-defined colors for text and background. The settings have an effect on the Editor and Runtime. **Note:** ▸ These settings are only available for configurable lists. ▸ In addition, the respective focus in the list can be signalized in the Runtime by means of different text and background colors. These are configured using the project properties.
User defined colors	*Active*: User-defined colors are used.
Text color	Color for text display. Clicking on the color opens the color palette to select a color.
Background color	Color for the display of the cell background. Clicking on the color opens the color palette to select a color.
Lock column filter in the Runtime	▸ *Active*: The filter for this column cannot be changed in the Runtime. **Note:** Only available for: ▸ Batch Control ▸ Extended Trend

Option	Description
	▸ Filter screens ▸ Message Control ▸ Recipegroup Manager ▸ Shift Management ▸ Context List

CLOSE DIALOG

Option	Description
OK	**Applies all changes in all tabs and closes the dialog.**
Cancel	Discards all changes in all tabs and closes the dialog.
Help	Opens online help.



AVAILABLE COLUMNS

Option	Description
Available columns	List of the available columns via **Column selection**. The highlighted column is configured via the options in the **Settings** area.

SETTINGS

Option	Description
Settings	Settings for selected column.
Labeling	Name for column title.

Option	Description
	The column title is online language switchable. To do this, the **@** character must be entered in front of the name.
Width	Width of the column in characters. Calculation: Number time average character width of the selected font.
Alignment	Alignment. Selection by means of radio buttons. Possible settings: ▸ **Left**: Text is justified on the left edge of the column. ▸ **Centered**: Text is displayed centered in the column. ▸ **Right**: Text is justified on the right edge of the column.
User-defined colors	Properties in order to define user-defined colors for text and background. The settings have an effect on the Editor and Runtime. **Note:** ▸ These settings are only available for configurable lists. ▸ In addition, the respective focus in the list can be signalized in the Runtime by means of different text and background colors. These are configured using the project properties.
User defined colors	*Active*: User-defined colors are used.
Text color	Color for text display. Clicking on the color opens the color palette to select a color.
Background color	Color for the display of the cell background. Clicking on the color opens the color palette to select a color.
Lock column filter in the Runtime	▸ *Active*: The filter for this column cannot be changed in the Runtime. **Note:** Only available for: ▸ Batch Control ▸ Extended Trend

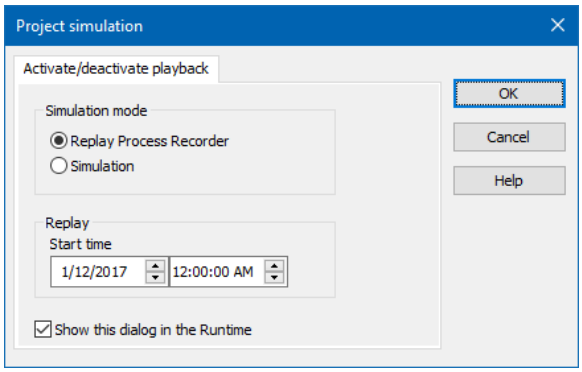
Option	Description
	▸ Filter screens ▸ Message Control ▸ Recipegroup Manager ▸ Shift Management ▸ Context List

CLOSE DIALOG

Option	Description
OK	**Applies all changes in all tabs and closes the dialog.**
Cancel	Discards all changes in all tabs and closes the dialog.
Help	Opens online help.

5.3.3 Process Recorder: Activate/deactivate playback

You start or stop the playback of recording in Runtime with the **Activate/deactivate project simulation** function.



Parameter	Description
Simulation mode	Type of simulation in zenon Runtime. Selection of the mode from an option field: ▸ *Process Recorder playback* Starts Runtime in playback mode of the project simulation for the **Process Recorder** module. ▸ *Simulation*

Parameter	Description
	Switches Runtime from playback mode back to project simulation. Default: *Simulation*
Playback start	Start time of playback. Entry of the point in time at which playback is started in Runtime. Selection of date and time from a combobox. Display format: *DD.MM.YYYY* or *HH:MM:SS* Click on the desired area to highlight this for the change. Change the area with an entry or by clicking on the arrow keys. Default: *Date of the call, 00:00:00* Note: Not active if **simulation mode** is *simulation*.
Show this dialog in the Runtime	Checkbox to select whether this dialog is shown in Runtime: ▸ *active*: This dialog is called up during operation in the Runtime on the current computer. In the network, this dialog is called up on the computer that executes the function. As a result, changes to existing parameter settings of an zenon Editor configuration are possible during execution in zenon Runtime. ▸ *Inactive*: This dialog is not shown in the Runtime during operation. The function or the command is immediately executed with the project configuration created in the Editor. Default: *inactive*

CLOSE DIALOG

Options	Description
OK	**Applies settings and closes the dialog.**
Cancel	Discards all changes and closes the dialog.
Help	Opens online help.

> **Hint**
>
> In order to have the greatest flexibility during playback in Runtime, configure the function with **Show this dialog in Runtime** activated.
>
> Link this function to a button in a template that can be reached by the screens used in playback.
>
> As a result, the start time can be selected flexibly during ongoing playback.

5.4 Display of changes to a project configuration

In playback mode, changes to the configuration for an existing element are visualized if the configuration is correct in the zenon Editor.

Example:

- The position of a screen element is amended in the configuration in the zenon Editor.
- This positioning is visualized when being played back in playback mode.

> **Attention**
>
> For correct display in playback mode the general project property **Versioning active** must be activated when configuring.

5.4.1 Configuration in the Editor

In order to correctly visualize project changes in the Process Recorder playback mode, the following configuration steps are necessary:

- Activate the **Versioning** for your project
  - Highlight the project in the tree view of the current **Workspace** in the Editor.
  - Activate, in the **General** project properties group, the **Versioning active** property.
  - Optional: Set the parameters for the additional options of the **Versioning** properties group.
- Configure changes or new content.
- Create a backup of your project.

- Go to the **Project backups** node in the tree view of the current project.
- Select the Create backup entry in the tool bar or in the context menu:
  The **Create project backup** entry is opened.
- Complete the fields of the dialog and confirm your input by clicking on the **OK** button.
  The project backup is created
- Successful backing up of the project is shown with a dialog in the Editor.
- The project backup is also shown as an entry in the main window after the dialog has been confirmed.

Changes to the project configuration are visualized in playback after a backup has been created.

Information

The memory required by the backup depends on the project configuration.

You can find more information in the project backup manual.

6 Process Recorder in the Runtime

The following is applicable for the **Process Recorder** in zenon Runtime:

- Loading files in playback mode:
  Data from the playback runs in its own thread. This allows the continuous playback of the recorded data without time-consuming reload pauses.
- Record files:
  The data recording of the **Process Recorder** module runs in its own thread. This allows continuous data recording without a loss of performance.
- Display of values:
  If a time range during which Runtime was not active is replayed in the playback, no values are shown during the playback. The elements in the process screens remain empty.
- Cross-project:
  Data that overlaps several projects in terms of time is compiled into a time bar and sorted chronologically.
- Variables that have not been recorded are marked with the *PR_NR* status bit.
  If the status bit is active, this status is shown in a yellow square in a screen element during playback.

PLAYBACK MODE

If Runtime is in playback mode, it continues to act as in project simulation.

Examples:

- No recording of **Process Recorder** data.
- Files are created or read in the simulation directory (for example: batch, command sequences).
- The **AUTOEND_Simul** script also works in playback mode when Runtime is closed.
- When closing Runtime, the **Variable image remanent** property is taken into account for each driver.
- Archives cannot be exported.
- The import of command sequences that have been created during simulation is not carried out.
- The Shift Management module does not access the database.

No data that is created by the playback is saved during playback. This is applicable for the following modules:

- CEL
- Alarm Message List
  The display of limit values, including flashing, is based on recorded data.
- Historian
- Process Recorder
- HD saving

> **Information**
>
> The teaching process for the **Command Sequencer** module is also available if Runtime is running in **Process Recorder**'s playback mode.

7 Process Recorder in the zenon network

The following rules are applicable for the behavior of the Process Recorder module in the network:

- Server and Standby create an initial file on startup.
- Behavior on the server:
  - A new entry is saved in the file on the server in the event of a value change.
  - The entry is also sent to the Standby Server.

- Behavior on the standby server:
  - The standby server gets the existing Process Recorder files from the server on starting.
  - After transferring the files:
    The standby writes the entry received from the server to the file on the standby.
  - If the file does not exist, the standby writes a corresponding LOG message.
  - To design the file sync as efficiently as possible, it is ensured that the Recorder files on the standby server have the file properties as on the server.
- For redundancy switching and reloading the project:
  - The current recorder file is closed.
  - A new Recorder file is created.

> **Attention**
>
> Note:
>
> - On the Standby Server, the values of the primary server are recorded in the **Process Recorder**. It is not the values that have been received by the standby server's own driver that are used.
> - Seamless redundancy is not supported for the **Process Recorder**. The variables are advised to the standby server but are not buffered. Recording is thus not guaranteed during redundancy switching.

8 LOG entries

The **Process Recorder** writes the following LOG entries.

RECORDING

Entry	Level	Description
Unable to Create File: [File name]	**ERROR**	The new file could not be created.
Unable to Create Filefolder: [Folder name]	**ERROR**	The target directory could not be created. Possible reasons: • Inadequate user rights • Insufficient free storage space

Entry	Level	Description
		▸ Target directory cannot be reached
Could not write to file!	**ERROR**	Access to the file failed or the file has not yet been created. This message can occur the first time a variable is *advised*, because the required file is not yet present and is only now created automatically.
Filehandler not initialized	**ERROR**	Functionalities for the writing of the Process Recorder file have not been started. This LOG entry can only be made on starting.
violated limit: <id>, blinking:	**ProcessRecorder DeepDebug**	A breach of a limit value has been recorded: ▸ *violated limit:* ID of the breached limit value
stop blinking	**ProcessRecorder DeepDebug**	An acknowledgment of the flashing has been recorded:

NETWORK

Entry	Level	Description
Send Data: Cls:CProcessRecorderMsgStartFile name:<filename> size:<filesize> time:<time>	**Debug**	LOG entry on the server. Sends a message to the standby that new file <*filename*> with the size <*size*> and time <*time*> has been written.
Receive Data: Cls:CProcessRecorderMsgStartFile name:<filename> size:<filesize> time:<time>	**Debug**	LOG entry on the standby. Received message that new file <*filename*> with size <*size*> and time <*time*> has been written on the server.
Unable to start new file correctly: <filename>	**ERROR**	LOG entry on the standby. New file <*filename*> could not be created.
Data is missing. File is smaller in size on SB<filename>	**ERROR**	LOG entry on the standby. In the event of a value change if the file

Entry	Level	Description
		<filename> is too small. This is the case if the file on the server has more entries than on the standby.
Receive Data: Cls:CProcessRecorderMsgValueChange file:<filename> startPos: :<start> modifyTime:<time> length:<length>	**DEBUG**	LOG entry on the standby. Notification from the server that new value changes have been recorded on the server. Value changes from the server to write to the file *<filename>* from position *<start>* with time stamp *<time>* received. Dump has a length of *<length>*
Send Data: Cls:CProcessRecorderMsgValueChange file:<filename> startPos: :<start> modifyTime:<time> length:<length>	**DEBUG**	LOG entry on the server. Send value changes to the SB for writing to the *<filename>* file from the *<start>* position with the time stamp *<time>*. Dump has a length of *<length>*
Unable to write value changes to file: <filename>	**ERROR**	LOG entry on the standby. Value changes could not be written to the file *<filename>*.
Unable to set modify time correctly: <filename>	**ERROR**	LOG entry on the standby. Value changes could be written but the time stamp of the *<filename>* file could not be amended. This is the case if the time stamp of the file is different to the to the time stamp of the file from the server. The file is synchronized during the next file sync.

RUNTIME

Entry	Level	Description
Reload PR with process recorder now enabled / disabled	**DEBUG**	Reloading in the event of a change of the activation of the Process Recorder: ▸ No change to the variable property. ▸ Process Recorder has been activated

Entry	Level	Description
		or deactivated.
Reload <n> variables with process recorder enabled	**DEBUG**	Reloading of the configuration for Process Recorder: The variable administration (with <n> new or amended values) was reloaded. The Process Recorder is now also used for this variable(s)
Runtime switch to Simulationmode	**DEBUG**	Runtime switches from playback mode to simulation mode.
Runtime switch to Playbackmode	**DEBUG**	Runtime switches from simulation mode to playback mode.
RT started in playback mode	**DEBUG**	Runtime was started in playback mode.
RT started in simulation mode	**DEBUG**	Runtime was started in simulation mode.
Start time for playback from Abs: | Rel: %source% is invalid	**ERROR**	No valid value or no variable for execution of the function found. *source*: ▸ *User input* ▸ *Variable name* (with start time from variable) ▸ *Screen name* (with start time from calling screen)
Missing project backup for version <version>	**ERROR**	When loading the recorder file, it is established that a project backup has been made. However, the attendant project backup file is not present. In the event that a project backup file is missing, the original project configuration (before the project backup was made) is visualized.
Could not extract <zipfile> to <destdir>: <error>	**WARNING**	The compressed project backup file <zipfile> could not be extracted in the target directory for playback.

Entry	Level	Description
		▸ *<error>* Detailed error description, depending on error that occurred.

REPLAY

Entry	Level	Description
No file loaded	**DEBUG**	When playback was started, no file that contains recorded data for the selected time point could be found.
Load file for replay <filename>	**DEBUG**	Notification of success on starting playback: The file <filename> was loaded and contains valid recording data for the selected time point.
No replay buffer filled	**DEBUG**	No values were found in the file.
Filled buffer for replay with <count> variables and <number> entries	**DEBUG**	Values for variables were found in the file. ▸ *<count>:* Number of variables with entries ▸ *<number>:* Number of value changes
Intervallstep forward	**DEBUG**	Successful jump forwards by an interval step during playback.
Intervallstep back	**DEBUG**	Successful jump backwards by an interval step during playback.
Cancel action	**DEBUG**	Successful cancellation of a jump command
singlestep forward	**DEBUG**	Successful switch to the next value change.
singlestep back	**DEBUG**	Successful switch to the previous value change.
playback status change	**DEBUG**	Playback was started or paused: Status: ▸ *started* ▸ *paused*
replay timer waiting nn ms	**DEEPDEBUG**	Search for value changes during playback.

Entry	Level	Description
Unload File: <filename>	**DEBUG**	The file *<filename>* is removed from the memory in playback mode.
Missing corresponding .lmt file to <filename>	**ERROR**	A corresponding recording file is missing for playback. From zenon 8.00, both the .rec file and the .lmt file are necessary for playback. Visualization in playback mode in Runtime is not possible if a file is missing.