



**COPADATA**  
do it your way

# zenon manual

## Allocations

v.8.00





©2018 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

<b>1. Welcome to COPA-DATA help .....</b>	<b>4</b>
<b>2. Allocations .....</b>	<b>4</b>
<b>3. Engineering in the Editor.....</b>	<b>6</b>
<b>4. Allocation detail view of toolbar and context menu.....</b>	<b>6</b>
<b>5. Trigger variables and trigger type.....</b>	<b>7</b>

# 1. Welcome to COPA-DATA help

## ZENON VIDEO-TUTORIALS

You can find practical examples for project configuration with zenon in our YouTube channel. 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](mailto: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](mailto: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](mailto:sales@copadata.com).

# 2. Allocations

Allocations make it possible to place the current value of a variable onto a second variable within or without the driver.

zenon can write values from the PLC directly into a SQL database or write value changes from one driver to another driver when they occur. You can also make cross-project allocations.

**Attention:** Cross-project allocations are executed only after all projects have been loaded.

## TRANSFER

An allocation transfers value, state and time stamp of the source variable to the target variable.



### Information

Status and time stamp are only processed by the following drivers:

- ▶ Internal driver
- ▶ SQL driver
- ▶ AK driver (SICAM 230)

A new write function is triggered by the driver during transfer. It writes the value, which is then read back. Time stamp and status are from the read time.

**Exception:** With an internal driver, the time stamp and the status are also transferred from the source variable.

States Real time external `T_EXTERN`, Real time internal `T_INTERN` und Daylight saving time/standard time announcement `T_CHG_A` are not transferred. They are defined by the time stamp.

The signal resolution should therefore have the same limits for both variables (overflow behavior).

The transfer of the value occurs in the event of a value change: Whenever the source variables value is changed, the value of the target variable is changed as well. If the value of the target variable is changed independently from that of the source variable, then the target variable has the different value as long as the value of the source variable does not change.



### Information

*Allocations to **Internal variables** are also only carried out on the server.*

## PROJECT MANAGER CONTEXT MENU

Menu item	Action
<b>New allocation</b>	Inserts a new allocation in the list.
<b>Export all as XML</b>	Exports all entries as an XML file.
<b>Import XML...</b>	Imports XML files.
<b>Help</b>	Opens online help.

### 3. Engineering in the Editor

To define allocations:

1. Select sub-node **Allocations** in node **Variables** in the **Project manager**.  
Already existing allocations are displayed in the detail view
2. Select the desired action from the context menu or the toolbar, e.g. **New allocation**
3. In the properties window the properties of the new allocation are displayed and can be edited

An allocation implicitly needs:

- ▶ an unique name
- ▶ a source variable
- ▶ a target variable

When selecting property **Source variable** or **Target variable** the corresponding variable list for selecting the desired variable is displayed. Each variable of one of each driver can be selected as a source. The target variable must be readable/writable.

Each value change to the source variable is transmitted to the target variable of the allocation.

The context menu of the detailview of the Project Manager allows to cut, copy, insert or delete existing allocations.

### 4. Allocation detail view of toolbar and context menu



## CONTEXT MENU

Menu item	Action
<b>New allocation</b>	Inserts a new allocation in the list.
<b>Jump back to starting element</b>	Makes it possible to jump back to the starting element.
<b>Copy</b>	Copies the selected entries to the clipboard.
<b>Paste</b>	Pastes the content from the clipboard. If an entry with the same name already exists, the content is pasted as " <b>Copy of...</b> ".
<b>Delete</b>	Deletes selected entries after a confirmation from list.
<b>Edit selected cell</b>	Opens the selected cell for editing. The binocular symbol in the header shows which cell has been selected in a highlighted line. Only cells that can be edited can be selected.
<b>Properties</b>	Opens the <b>Properties</b> window.
<b>Export selected as XML...</b>	Exports all selected entries as an XML file.
<b>Import XML</b>	Imports entries from an XML file.
<b>Remove all filters</b>	Removes all filter settings.
<b>Help</b>	Opens online help.

## 5. Trigger variables and trigger type

A trigger variable and a trigger type can be defined for assignments. Event-triggered assignments can thus be defined. Values are thus no longer assigned automatically, but only when the value of a trigger variable changes.

Trigger variable must be `binary variable`. Possible configurations are:

Trigger variable	Trigger type	Result
not defined		Each change in the source variable causes the source value to be assigned to the target variable.
defined	Edge	The assignment is only executed if the trigger variable undergoes a value change from 0 to 1. Only a positive edge of the bit variable leads to the assignment being released. The assignment is no longer instigated by a change to the value of the source variable.
defined	Gage	The assignment is always carried out as long as the trigger variable has the value 1. Each time the value of the source variable changes, the assignment is activated and the value is written to the target variable. This assignment remains active until the value of the trigger variable changes back to 0.