zenon manual

Release notes zenon 7.60

v.7.60

Contents

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 (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. General

2.1 Converting projects

Before you convert a project, please read back all Runtime changeable files (User Administration, Standard Recipes, Recipegroup Manager, Scheduler/PFS) into the Editor. This ensures a complete data conversion and makes sure that none of the changes made in the Runtime are lost. After converting to the new version, create all Runtime files once including Runtime changeable data.

Note: You can find important information for the conversion of certain versions in the zenon help in the Project conversion manual.

CONVERTING MULTI-USER PROJECTS

Multi-user projects can only be converted if no elements are checked out. This means that all people configuring projects have to **accept** their changes first.

If this is not possible for some reason, you have to create a project backup of the project on the project database server and then immediately restore it. This resets all the **Under construction** information. Attention: All changes in the local project versions are lost!

The conversion can only be done on the PC, on which the central project database resides. If there is no Editor on the PC (standalone database server – no longer supported), you must install the Editor first. Only after that can the conversion be done on this PC.

CONVERSION FROM VERSION 6.01 AND 6.20

zenon projects in version 6.01 and 6.20 can no longer be directly read back in zenon 7.10 or higher.

Background: Versions that are based on the MSDE (SQL Server 2000) are not compatible with the SQL Server 2012 used in zenon.

Solution: First convert in zenon 7.0 and then in 7.10 or higher.

CONVERSION OF PROJECTS FOR 7.20

For compatibility with version 7.20, there is an additional possible selection - "`Most recent version`" - available for the **Create Runtime files for** property. This can be selected with the `7.20 SP0 + [most recent build no.]` in the drop-down list.

With this selection, the Runtime files for the current build of version 7.20 are provided. Functionality that has since been incorporated into version 7.20 after the official release of 7.20 is thus supported. This is applicable most of all for enhancements to drivers that are now supported with this option. Please note that, to use the most recent build, you must have zenon 7.20 installed on your computer.

Note: The `7.20 SP0` selection compiles the Runtime files - as before - to the default settings of 7.20 SP0.

2.2   zenon Logic Workbench

zenon Logic Workbench 9.1 is integrated into zenon.

2.3 Supported operating systems

Supported operating systems and required service packs:

Operating system	zenon Editor	zenon Runtime	zenon Web Server	zenon Web Client	zenon HTML Web Engine	zenon Logic Runtime	zenon Analyzer Server
Windows 7 (Professional, Enterprise and Ultimate version, x86 and x64 versions).	SP 1	SP 1	SP 1	SP 1	SP 1	SP 1	Cannot run
Windows Embedded Standard 7 (if all necessary operating system components exist).	Cannot run	SP 1	SP 1	SP 1	SP 1	SP 1	Cannot run
Windows 8 and 8.1 (Standard, Professional, Enterprise version, x86 and x64 versions)	SP 0	SP 0	SP 0	SP 0	SP 0	SP 0	Only x64 with SP 0
Windows Embedded 8 Standard (if all necessary operating system components exist).	Cannot run	SP 0	SP 0	SP 0	SP 0	SP 0	Cannot run
Windows 10 (Home, Pro, Enterprise, Education, IoT Enterprise, version, x86 and x64 versions)	SP 0	SP 0	SP 0	SP 0	SP 0	SP 0	Only x64 version of Home, Pro and Enterprise with SP 0.
Windows Server 2008 R2 (All editions with the exception of Core)	SP 1	SP 1	SP 1	SP 1	SP 1	SP 1	Cannot run
Windows Server 2012 and 2012 R2 (All editions with the exception of Core)	SP 0	SP 0	SP 0	SP 0	SP 0	SP 0	Only x64 with SP 0
Windows Server 2016 (All editions with the exception of Core)	SP 0	SP 0	SP 0	SP 0	SP 0	SP 0	Only x64 with SP 0

2.4 Time to react to graphical changes (D 31687)

The graphical visualization is now checked for changes more often in Runtime.
This can speed up the reaction times for graphical visualizations.

Note: In exceptional cases, there can be a higher load placed on the system.

2.5 Import/export (F 26709)

The import and export of functions and variables has been optimized. Comprehensive XML and CSV file content is imported or exported more quickly.

In addition, the import dialog for pre-existing variables has been amended.

3. Internal driver - licensing

The licensing for the variables of the **internal driver** was amended with build 38837:

- ▶ Does not require a license:
  - No support of the external time stamp and no additional visualization of the status (`spontaneous` and `time internal` only) in the Runtime.
  - **Internal driver** variables are not calculated as **TAG** for licensing.

- ▶ Requires a license:
  - Complete support of all time stamps (internal and external) and all status bits in the Runtime
  - Each **internal driver** variable is counted as **TAG** for licensing.

AMENDMENTS IN PROJECT.INI

*In order to be able to visualize the external time stamp and all states of an **Internal driver** in the Runtime, the following entry must be entered manually in the `project.ini` in the Runtime [RT] section:*

INTERNALDRIVERWITHSTATUS=1

DISPLAY IN RUNTIME

The following is applicable for the display of **internal driver** variables in zenon Runtime:

▶ **Internal driver** variables **with** tag counting:
Display of all available variable information.

▶ **Internal driver** variables **without** tag counting::

- Time stamp
Only the current time of the value change of the internal time stamp.

- States:
status bits `Spontan` and `T_Intern` only. Also `T_STD` if standard time is set.

- Other statuses and time stamps are neither displayed nor processed (for example via zenon API, reaction matrices, combined element, assignment, zenon Logic, ...).

> 💡 **Information**
>
> *In modules that also support the external time stamp (such as AML, CEL, variable diagnosis, ...), for use of the **internal driver** that does not require a license, no value is displayed for the external time stamp.*

4. zenon Logic

4.1 Current zenon Logic Workbench (F 23583)

zenon Logic was, in the current version 9.1 integrated into the Logic workbench in the zenon Editor.

4.2 Display of versions and license information (S 40861)

The dialog to display version and license information can now be opened using the **Help** -> **Info...** menu item in zenon Logic Workbench.

4.3 IEC850 Driver (S 39551)

▶ Support for **INT64(U)** as `LINT` and `ULINT` data types.

> 💡 **Information**
>
> These enhancements are also applicable for the zenon IEC850 driver.

4.4   zenon Logic Runtime (FS 37547)

zenon Logic Runtime has been ported to the current version 9.1. This is included in the standard version of zenon.

Function blocks for communication with MySQL and SQLite databases are available.

4.5   zenon Logic Real Time extension for Windows 8.1 and 10 (S 22096)

The **zenon Logic Real Time Extension** is available for the 32-bit version of the following operating systems:

- ▶ Windows 8.1
- ▶ Windows 10

4.6   IO Driver

Enhancements and changes to the zenon Logic IO driver:

- ▶ PROFINET IO Controller:
  - Supports the `Octetstring` data type in the configurator.
- ▶ EtherCAT Master:
  Based on Beckhoff Stack.
- ▶ Brodersen IOTOOLS configurator has been removed.

5. Editor

5.1 Changes to menus

The structure of some menus has been changed in version 7.60.

Menu	Action	Description
View	Newly-created	Now contains access to the tool bars. These were previously in the **Options** menu (now **Extras**).
Tools	Renamed from **Options** in **Tools**.	The tool bars were removed from this menu and entries for programming interfaces and wizards were added.
File	Entries for programming interfaces and wizards were moved to the **Tools** menu.	
Help	New entry: **Update documentation...**	Updates the documentation using the **Documentation Download Tool**.
Help	New entry: **zenon Video Tutorials**	Calls up the **COPA-DATA YouTube channel** with examples of project configuration in a web browser.

5.2 Performance optimization - driver initialization

The performance when initializing drivers in the zenon Editor and Runtime has been increased considerably.

Drivers are now - depending on the number and availability of the cores on the computer doing the execution - registered and started at the same time.

Note: With network projects, the computer that is the primary server is the one doing the processing.

The shutting down of drivers is also carried out at the same time.
When closing or shutting down zenon Runtime, it is ensured that all commands are closed first, before Runtime is closed.

5.3 DPI awareness (F 36879)

The user interface of the zenon Editor has been amended so that it can now also be operated with different DPI settings without restriction.

The DPI settings can be amended using the **display settings** of Windows (**size of text, apps and other elements**); the default value is `100%`.

5.4 Notice texts in the detail view (S 44610)

For all lists in the detail view (variables, functions, screens, styles, etc.), a notice text is now shown if, during filtering, no entry corresponds to the filter criteria and the list is thus empty. This text is shown in the middle of the detail view and states: `No entry matches the filter criteria`

for styles, there is also the following info text if no style group has been created: `No style group has been created. Select "Create style group" in the context menu of a screen element`

5.5 zenon Video Tutorials (FS 37622)

Calls up the **COPA-DATA YouTube channel** with examples of project configuration in a web browser.

These tutorials can be called up in the zenon Editor.

▶ Using the **Help** entry in the menu bar

▶ By means of a link in the properties help

6. Energy Edition

6.1 Automatic Line Coloring (S 38951, S 39658)

FAULT LOCATION BASED ON IMPEDANCE

The placing of an error marker an an ALC project configuration in the topology was previously only possible by means of programming using the zenon API interface.

From zenon version 7.60, this project configuration can also be carried out with properties in the user interface of the Editor.

To do this, both the **Fault location based on impedance** property (**Automatic Line Coloring** project properties group) and **Marker color** property (**Automatic Line Coloring** combined element properties group) have been introduced.

Error markers can be configured for **ALC Function type** `switch` combined elements.

> 💡 **Information**
>
> *The ALC topology package must be licensed for impedance-based error detection.*

MARKER VIA API ON CURVE, POLYLINE AND PIPE

The marker element for the normal 2-point line element has been enhanced.

Supported ALC elements

- ▶ Line;
  also non-horizontal lines
- ▶ Polyline
- ▶ Pipe

6.2 New "All interlockings" control element in a command screen (S 27814)

For `command` screens, there is a new "`All interlockings`" control element available. This interlocking list can be configured and supplements or replaces the existing and unchangeable "`All interlockings`" control element. The existing control element has been moved to the **Compatible elements** control elements group.

This control element can be configured with the **List configuration** property in the **Command Processing screen** properties group of the **Command** module.

7. Runtime

7.1 Performance optimization - driver initialization

The performance when initializing drivers in the zenon Editor and Runtime has been increased considerably.

Drivers are now - depending on the number and availability of the cores on the computer doing the execution - registered and started at the same time.

Note: With network projects, the computer that is the primary server is the one doing the processing.

The shutting down of drivers is also carried out at the same time.
When closing or shutting down zenon Runtime, it is ensured that all commands are closed first, before Runtime is closed.

7.2 Stability improvements

The behavior of zenon Runtime on reloading has been revised.

As a result, there is, especially for complex project configurations, a significantly reduced loading time and increased stability.

These improvements also include comprehensive project configurations, such as with integration projects.

8. Optimization of the communication to the driver (F 25422)

The drivers are registered in the operating system at the same time. On the calling computer, the corresponding drivers, divided according to the available threads, are registered in parallel.

The new behavior was incorporated into the existing documentation. In addition, in the existing documentation for the **Variables** manual, the **Activate variables in zenon** chapter has been revised and enhanced accordingly.

9. Screens

9.1 Hierarchical filter added to the equipment model screen type (S 44621)

The `equipment model` screen was enhanced with the **hierarchical filter** screen.

If this screen element is activated, the variables that are linked to an equipment group of the selected equipment model are shown.

9.2 Variable diagnosis screen type enhanced (S 29404)

In the **variable diagnosis** screen type, the list of variables has been enhanced with the following column entries:

- ▶ **Actual value (raw data)**: in the value range of the controller
- ▶ **Status (binary)**: Status bits of the variable
- ▶ **Time stamp**:
  This new time stamp is shown in microseconds [µs].
  - `internal (UTC)`
  - `internal (local)`
  - `external (UTC)`
  - `external (local)`
- ▶ **Symbolic address**

9.3 New properties for configurable lists (F 32114)

The properties for `configurable lists` have been enhanced.

The behavior of headers in Runtime is now defined in the Editor by means of the properties of the **Header** group:

▶ **Show header**: Controls whether the header is displayed in Runtime.

▶ **Show filter row**: Controls whether the filter line is displayed in Runtime.

▶ **Disable sorting**: Controls the possibility of sorting table columns in Runtime with a click on the header.

▶ **Freeze column location**: Controls the possibility to amend or move the width of table columns in Runtime with mouse actions.

▶ **Deactivate context menu**: Activates or deactivates the context menu for the header.

ACTION IN THE EVENT OF PROJECT CONVERSION

The following is applicable for project conversion:

▶ If a project is compiled for a version before zenon 7.60, the **Header static** property is activated if the **Disable sorting** property is not activated in the current project.

▶ If a project is converted from a version before zenon 7.60 into the current version, then the **Freeze column location** and **Disable sorting** properties are activated if **Header static** is inactive in the original project.

9.4 Symbol in the dialog: Remove all filters (F 23670)

There is now the symbol for **Remove all filters** in the tool bar for the following dialogs. All filters set by the user are deleted when this symbol is activated.

▶ Screen selection dialog

Note: If a filter has already been selected in the selection list, the pre-selected screen is shown in the screen selection dialog, even if it does not suit the filter criteria.

▶ Function selection and script selection dialog

▶ Variable selection dialog

9.5 Line configuration for trend element amended (F 12432)

The property to configure the line type of the trend element has been amended.
Selection is made using the new **Line type** property: It offers a screen list for selection.
The previous property continues to be available for XML import and via the API.

9.6 Full-screen mode (F 4357)

The **Control elements** menu can be called up in full-screen mode.

In addition, the position of the **full screen** tool bar is saved and called up again the next time on restarting.

9.7 Win32 control elements replaced (F 23610)

Most static Win32 control elements for entries have been replaced by `dynamic text` elements; checkboxes and radio groups have been replaced by `switch` elements. These elements are assigned the respective function using the **Screen type specific action** property. However, if these elements have been deactivated or there are not the necessary user rights for operation, the project setting from the **Locked/Interlocked elements** group is used.

REPLACED ELEMENTS

The following elements have been replaced in the following screens.

INPUT FIELD WITH DYNAMIC TEXT

Replacement of the previous static Win32 control element with a `dynamic text` element for:

Active Directory user administration

- ▶ **Domain name**
- ▶ **User name**
- ▶ **Password**

Alarm Message List Filter

- ▶ **Set filter type (Display)**
- ▶ **Archive list status**
- ▶ **Lot list status**
- ▶ **Lot name filter (Input field)**

- ▶ **Search text**
- ▶ **Variable name**
- ▶ **Identification**
- ▶ **Maximum number**

Equipment Model

- ▶ **Description of the equipment model**

Archive revision

- ▶ **Set filter (Display)**
- ▶ **Archive status**
- ▶ **Short name**
- ▶ **Total number**
- ▶ **Number of INVALID**

Command Processing

- ▶ Control elements, **information** group
  - • **Action variable (name)**
  - • **Action variable (identification)**
  - • **Action variable (measuring unit)**
  - • **Action variable (status)**
  - • **Response variable (name)**
  - • **Response variable (identification)**
  - • **Response variable (value)**
  - • **Response variable (measuring unit)**
  - • **Response variable (status)**
  - • **Minimum set value**
  - • **Maximum set value**
  - • **Interlocking text**
  - • **Active action**
  - • **Switching direction**
  - • **Trigger source**
  - • **Action name**
- ▶ Control elements, **Lock** group
  - • **User**
  - • **Lock code**

- **Comment**

Chronological Event List Filter

- ▶ **Set filter type (Display)**
- ▶ **Archive list status**
- ▶ **Lot list status**
- ▶ **Lot name filter (Input field)**
- ▶ **Search text**
- ▶ **Variable name**
- ▶ **Identification**
- ▶ **Maximum number**

Extended Trend

- ▶ **Diagram name**
- ▶ **Set filter**

Industrial Performance Analyzer

- ▶ **Set filter**

Load Management

- ▶ **Supply area name**

Report Generator

- ▶ **Set filter**

Report Viewer

- ▶ **Set filter**

Worldview Overview

- ▶ **Show zoom steps**

Time filter

- ▶ **Set filter type (Display)**
- ▶ **Archive list status**
- ▶ **Lot list status**
- ▶ **Lot name filter (Input field)**

CHECKBOX WITH SWITCH

Replacement of the previous static Win32 control element with a `switch` element for:

Alarm Message List filter, Chronological Event List filter.

- ▶ **Case sensitivity**

- **Words do not need to appear in full within the text**
- **Show list without refresh**
- **Case sensitivity**
- **Only unacknowledged alarms (AML filter only)**
- **Only active alarms (AML filter only)**
- **Only cleared alarms (AML filter only)**
- **Comment required (AML filter only)**
- **Exclude system messages from filter (CEL filter only)**
- **Show relative times (relative to selected entry) (CEL filter only)**

Batch Control

- **Dynamically update control recipe list**
- **Prepared control recipes**
- **Currently executed control recipes**
- **Finished control recipes**
- **Outdated control recipe**

Chronological Event List

- **Show relative times**

RADIO BUTTON WITH SWITCH

Replacement of the previous static Win32 control element with a `switch` element for:

Alarm Message List filter, Chronological Event List filter.

- **Ring buffer**
- **Historic data**
- **No text filter**
- **Search for (separate words by ...)**
- **At least one word should be in the text**
- **All words should exist in the text**
- **Filter string has to appear exactly in the text**

Alarm Message List filter, Chronological Event List filter, Time filter

- **All elements in the time filter group - time filter type**

Batch Control

- **All elements in the recipe control group - switching of execution mode**

Already replaced in version 7.50

Alarm Message List and Chronological Event List

- ▶ **Comment field**

- ▶ **Set filter**

- ▶ **Status**

- ▶ **Total number**

- ▶ **Number of unacknowledged**

- ▶ **Linked function (display)**

Chronological Event List

- ▶ **Set filter**

- ▶ **Total number**

- ▶ **Status of Chronological Event List**

User Administration

- ▶ **Current user (display)**

- ▶ **Old password**

- ▶ **User name**

- ▶ **Email**

- ▶ **Substitute person**

- ▶ **Group name**

- ▶ **Cell phone**

- ▶ **NA code**

- ▶ **New password**

- ▶ **Password**

- ▶ **Confirm password**

- ▶ **Signature**

- ▶ **Lock code for Command Processing**

- ▶ **Telephone**

- ▶ **PIN code**

- ▶ **Complete name**

    Note for Web Client: The following **dynamic text** elements in the "User Administration" screen can also be used for **zenon Web Server Pro** on the **zenon Web Client**: **User name, password, signature**.

Recipegroup Manager

- ▶ **User name**

▶ **Last modify time**

▶ **% (Input field)**

▶ **Formula (input field)**

▶ **Comment 1 - 8**

▶ **Recipe number**

Note: If the element is used in Runtime to replace a static Win32 control element and it is deactivated, the background color and text color is set to gray values.

CHECKBOX WITH SWITCH

Replacement of the previous static Win32 control element with a `switch` element for:

User Administration

▶ **Administrator**

▶ **Active**

▶ **Locked**

▶ **Message Control user**

9.8   WPF (F 5071)

The following changes were carried out for the WPF element:

▶ Transparency made possible

▶ Replacement of DLLs in ongoing operation made possible

TRANSPARENCY

The WPF elements now support full transparency. In order for WPF controls in which a transparent background has been defined to also be displayed as transparent, the following must be the case on the computer for both Editor and Runtime:

▶ The operating system must be at least Windows 8.1

▶ The .NET framework version 4.6 or higher must be installed

WPFs are not shown as transparent in Windows 7 or 8. Instead, the transparent areas are filled with the background color set on the zenon screen.

REPLACEMENT OF DLLS

DLLs that are part of a WPF element can now also be removed or replaced during ongoing operation.
To replace a DLL:

1. Close all zenon screens in which the WPF element is used.

2. Close all symbols that use a desired WPF element.

3. Replace the DLL in the `\wpfcache` folder of the Runtime files:
   You can find this folder at `...\PROJECTNAME\RT\FILES\zenon\custom\wpfcache.`

This only applies for the Editor.

9.8.1   COMTRADE-Viewer (S 45961, 45964, Def. 3725)

Changes were made in the processing of *.cfg- and *dat files in the **COMTRADE-Viewer** WPF element. File content that does not comply with standards is treated as standards-compliant.

The information from the *.cfg file allows the evaluation of the *.dat file. It contains the data from various analog and digital series of measurements of currents and voltages. The data is broken down into individual data sets and shown in hex format.

New features for the **COMTRADE-Viewer**:

*.cfg files

▶ The last file entry is a time multiplier. This entry is multiplied by the time stamp of one of each entry from the *.dat file when a disturbance (error message) is read in. If there is no time multiplier, a factor with the value of 1 is used internally. The *.cfg file is not changed in the process.

▶ Certain standards apply for the entries of the digital measured values. If there is no zero at the end of the entry, the **COMTRADE-Viewer** adds one. The *.cfg file is not changed in the process.

*.dat files

▶ The viewer can now also read *dat files that start with the index 0 or >1. In doing so, a check is carried out to see whether these data sets are numbered continually in steps from 1. If there are data sets that are not correctly numbered, the file cannot be read in.

9.8.2 Waterfall diagram

Display of days for bar values (Def. 37356)

The display of days has been added to the waterfall chart. For this, the following applies:

- ▶ If the value is less than 24:00 hours, the time is shown in hours, minutes and seconds.
- ▶ If the diagram contains a bar that contains more than 24 hours, the bar is shown in the format D HH:MM:SS for the complete waterfall diagram.

Link bars to zenon function

The bar of the display in zenon Runtime can be linked to a zenon function.

The project configuration steps necessary for this are documented in the **zenon WPF** manual, in the **Pre-built elements** area of the Waterfall diagrams chapter.

9.9 Screen elements

9.9.1 Fill pattern for buttons (F 9067)

The properties of the **Fill** group of buttons have been enhanced with the **Fill pattern** property.

If, before conversion, `single-color` was configured under **Color gradient**, the fill pattern is now set to `single-color`. If, under **Color gradient**, one of the other options was configured, the fill pattern is now set to `gradient` and the type of gradient is set according to the original configuration under **Color gradient**. This is also applicable to XML import from older versions.

If Runtime files are created for older versions, a conversion to the other direction is undertaken. Access via the API is effected by means of the previous **Filling** property.

9.9.2 Radiant as a color gradient property for static elements (F 23622)

The `radiant` option is available for the **Color gradient** property with the following static elements:

- ▶ Rectangle
- ▶ Polygon
- ▶ Ellipse/Circle

▸ Segment of a circle

Previously, this option was only available for the dynamic element button.

9.9.3 Compatible display of elements in Runtime with Windows basis (F23622)

For a compatible display of buttons and vector elements (for projects that were created before version 7.60), an additional entry has been created in the **project.ini** configuration file:

USEGDILEGACYDRAWING

Mode for compatible display of elements in Runtime with `Windows Basic` set.

▸ `0`: Buttons and vector elements are shown using the new character routines (corresponds to the display in the Editor).

▸ `1`: Buttons and vector elements are shown as in versions < 7.60.

Default: `0`

For project backups that are read back from versions < 7.60, the value is set to `1` when reading back.

9.9.4 Display of the variable value using placeholders in the combined element (S 39503)

The following placeholders can be used display the value of a variable in the combined element, along the lines of pre-existing placeholders.

Placeholder	Description
%v	Shows the current variable value.
%v,1,2	Shows the current variable value starting at position 1 for 2 characters.
%V	Shows the current variable value.
%V,3,2	Shows the time stamp minus the first 3 and the last 2 characters.

9.10   TIF graphics format is supported

The TIF/TIFF graphics format (Tagged Image File Format) is now supported for:

▸ Backgrounds for screens

- ▶ Graphics files for buttons
- ▶ Graphics files for switches

TIF graphics are not supported for the combined element and in the Report Generator.

Note: TIF is only supported under DirectX, not under Windows.

10. Functions and scripts

10.1 Selection of a virtual monitor with screen switching indexed (S 26338)

When screen switching with the **Screen with index** function, a virtual monitor can be selected from a drop-down list as a target of the screen switching.

If only one monitor has been configured, this drop-down list is not shown.

10.2 New function for the Process Recorder module (F 7962)

The functions of the Application group have been enhanced with the **Activate/deactivate Process Recorder playback** for the Process Recorder module.

10.3 New function - show license information (S 39318)

A new **Show license information** function was added to the **Application** group, which can be used to show the version and license information.

10.4 Activate/deactivate project simulation (S 29588)

The **Activate/deactivate project simulation** function was enhanced with the option for the **Process Recorder** module.

New entries:

- ▶ **Activate simulation**

- **Initialize with process image and Runtime files**

- New tab:
  Playback

10.5 Operating authorizations function in the network amended (S 39741)

The **Authorization in the network** function has been enhanced with the option for operating authorizations via equipment model.

In addition, this dialog can now also be shown in Runtime.

11. Modules

11.1 Alarms administration

11.1.1 Linking of an alarm cause for acknowledgment (F 23653)

The same as for the mandatory entry of a comment for alarms that require acknowledgment, these can now be configured so that an alarm cause must be linked before acknowledgment.

With this property, it is possible to use the screen switch function to filter for alarm message list screens and alarm message list filters, as well as in the alarm message list screen.

11.1.2 Multi-line display in the AML (S 40798, S 40800, S 40847, S 40801)

It is now possible to display texts in multiple lines in the AML and CEL. If the column width is not sufficient for the text to be displayed, a line break is carried out and the line height is increased accordingly.

This option can be activated using the **Automatic word wrap** property of the control element.

11.2   Equipment Modeling (F 39735)

11.2.1   Hierarchic filter (S 39736)

Hierarchical filtering is supported when filtering for equipment models. A variable now need only be assigned to one equipment group level of the equipment model and it is then automatically incorporated into a filter for higher levels.

To do this, the new **hierarchical filter** option was introduced in the filter dialog of the **Screen switch** function.

11.2.2   Operating authorization (S 39741)

The equipment model has been added to the operating authorizations in zenon.

▸   The **Bedienberechtigung aktiv** project property has been replaced with the new **Operating authorization in the network** property.

▸   Selection of the type of operating authorization in a drop-down list:

- `Inactive`
  No operating authorization required for operation in Runtime.

- `Global operating authorization`
  Applicable for the complete project and all elements contained therein.

- `Operating authorization via equipment model`
  Applicable for parts of the project, based on the equipment model.

COMPATIBILITY

Existing project configurations are also transferred to the **Operating authorization in the network** property during conversion.

11.3 Historian

11.3.1 INSERT für CRATE.IO (F 26708)

During SQL evacuation of archive data and during SQL export of archive data to CRATE.IO, with **INSERT**, the values are now used for the Crate-specific SQL command **UNNEST**. The values of the archive file are transferred block by block as strings in the process.

11.3.2 Status variable for archive status and evacuation status (S 40156)

The project configuration dialog for archive configuration has been enhanced with two settings:

▶ In the **Runtime** tab, a BOOL variable can be linked for the status of the archive (value TRUE, even if the archive is running).

▶ In the **Save** tab, a BOOL variable for the for the status of the evacuation can be linked (value TRUE in the event of problems). This variable can be evaluated using a reaction matrix, for example.

11.4 User Administration

11.4.1 Login with alternative domain

AD domain users can now, for signing into zenon, even use a different AD domain than that which is used for sign-in in Windows.

Configuration is carried out using the new **Acive Directory domain** property. For versions from zenon 7.60, only configuration by means of the Editor is recommended. For versions 7.20 and 7.50, configuration can be by means of **project.ini**. The setting is entered into **project.ini** in the **[PASSWORD]** section as **ADDOMAIN**.

11.4.2 Login with login code (F 26260)

Users can also be logged in without a password in Runtime by means of a separate login code. This code is linked to the user and is transferred for login by means of a variable. Configuration is carried out using the **Login without password** function. The login code is linked to a user by means of a dialog for user configuration or the **Login code** property.

In Runtime, the code is read by means of registration by with a login code from the variable and the user linked to this is searched for. If a user is found, they are logged on. If automatic login for subprojects has been configured, the login is also carried out for the subprojects.

In Runtime, the login code can only be amended by a user with administrator rights.

> ⚠ **Attention**
>
> Login with a login code only works for zenon users and is not available for AD users.

11.4.3 Login with cached credentials (F 26422)

For AD domain users, login with cached sign-in information is possible. The sign-in can also take place if there is no connection to the AD domain controller.

To allow a login with cached login information:

- ▶ The user must have logged in at least once previously with their access data

- ▶ A user group must be selected for the new **User group for Active Directory login with cached credentials** property

- ▶ The Windows functionality for logging in with cached credentials must not be deactivated (by means of group guidelines, for example)

The last 10 items of login information are cached by default.

If an AD user attempts to log on in Runtime, if there is no connection to the AD Domain Controller, a check is made to see whether, in the **project.ini**file, there is a GUID for the **USRGROUP_AD_CACHED** entry with cached login information:

- ▶ The sign-in is rejected if there is no GUID.
  No user group has been configured or a configured user group has been removed by clicking on **No selection**.

- ▶ If there is the GUID of a valid user group, the user is signed in with permissions from this group.
  A valid user group has been configured.

- ▶ If there is the GUID of an invalid user group, the user is signed in without rights.
  A user group was entered but the user group has been deleted.

Each attempt to sign in with cached login information is entered in the CEL.

> ⚠ **Attention**
>
> *If AD users are used, the users must not be deleted in the domain controller. Users who are not longer valid should only be deactivated.*

11.4.4　New function authorizations for shift management (F 1502)

Two function authorizations have been added to the Shift Management module. These provide authorization levels for the creation, editing and deleting of shifts and shift models in Runtime.

11.5　Chronological Event List

11.5.1　Multi-line display in the CEL (S 40798, S 40800, S 40847, S 40801)

It is now possible to display texts in multiple lines in the AML and CEL. If the column width is not sufficient for the text to be displayed, a line break is carried out and the line height is increased accordingly.

This option can be activated using the **Automatic word wrap** property of the control element.

11.6　Context List (F 23630, F 23631, F 23653)

The **Context List** module allows central administration of hierarchically-structured texts in Runtime. It can be used for the central administration of alarm causes. To do this, a Context List entry is linked to one or several entries in the Alarm Message List. Possible alarm causes can thus be pre-defined centrally. The Context List also simplifies evaluation, for example in reports. Alarm causes that are also required (alarm causes) can be easily added in runtime using the zenon `Context List` screen.

The following is applicable for the **Context List** module:

▶ Several nodes can be arranged in parallel or hierarchically in a Context List. Each node can contain several entries.
  Context Lists are limited to a maximum hierarchy level of 5 levels and the language cannot be switched.

▶ Context Lists are persistent. They cannot therefor be deleted, only hidden from the user interface. Gaps in reports are thus avoided.

▶ If context lists are used in a network project, they are saved on the server. Clients are automatically synchronized.

Context lists are configured in Runtime using a `Context List` screen.

`Context List` screens can also be used to assign alarm causes to alarms in Runtime.

11.7 Process Gateway

11.7.1 MS Azure

HTTP proxy server freely configurable (FS 36510)

For the **MS Azure - Process Gateway**, the login can be freely configured on an HTTP proxy server, regardless of the user logged in.

Corresponding input fields were added to the configuration dialog for the **MS Azure - Process Gateway**.

Configurable connection to the IoT hub (FS 37486)

In the MS Azure Process Gateway, a further MS Azure Connection to an IoT Hub can be configured. The payload of the connection is transferred in JSON format.
In doing so, the following is applicable:

▶ Actual values are sent to the `IoT Hub` as a message.

▶ Messages received by the `IoT Hub` are decoded and written to the contained value as a set value.

11.7.2 ICCP-TASE.2 (FS 36382 - S 31441, S 31442, S 32431)

Enhancements for the **ICCP-TASE.2 Process Gateway**:

▶ When starting the Process Gateway's ICCP clients, the automatic establishment of a connection to a configurable remote ICCP server can be activated.

▶ For the connection status between the **Process Gateway** and remote ICCP communication partner, variables for display of the status can be configured for both server and client in zenon Runtime.

▶ When the Process Gateway client establishes a connection, a check of the variables available at the remote server can be activated with the variables configured in the client:

- Names of the missing ICCP data points can be configured in a string variable for display in zenon Runtime.

- Missing data points are removed from the dataset.

- For zenon variables that form the missing data points, a user status bit (`Merker1 – Merker8`) can be set.
  As a result, unsupplied variables can be evaluated, for example by means of a reaction matrix.

The configuration dialog of the Process Gateway has been enhanced in the **General** tab accordingly.

11.7.3   OPC UA - amendments to OPC UA specification, version 1.03 (FS 37354, S 39662, 39663, 39664)

The maximum number of subscriptions has been limited to 10 per session by default.

AMENDMENTS TO THE CURRENT OPC UA SPECIFICATIONS

The **OPC UA** module of the **Process Gateway** has been enhanced with the following function - in line with version 1.03 of the `OPC UA Specification`:

▶ Sequence for `PublishRequests` (first in, first out):
When sending, the incoming sequence of the `PublishRequest` is taken into account. If a `PublishResponse` is processed and sent, this is always assigned to the oldest `PublishRequest`. It is thus ensured that the mandatory sequence of value changes by the OPC UA server is adhered to. This is also applicable if a client creates or deletes several `Subscription`.

▶ `BadNoSubscription` for all `PublishRequest` if the last `Subscription` has been deleted or the `Session`   has been closed:
If the last `Subscription` has been logged off from the server, for `PublishRequests`s that are still logged onto the server, `PublishResponses` with the value `OpcUa_BadNoSubscription` is created and sent to the client. The open `PublishResponse`s are then deleted on the server.

▶ Timeout for `PublishRequests`:
Before the `PublishResponse` for the `Subscriptions` are sent, it is ensured that only valid - i.e. not expired `PublishRequests` are present. The validity is checked using the time stamp of the `PublishRequest` as well as the `timeoutHint` stated in the `PublishRequest`. The resulting end time must be less than the current time of the server. If this is not the case, a

`Response` with `OpcUa_BadTimeout` is sent back to the client. The `Request` is deleted in this case.

11.7.4   DNP3 - Command Processing, Command Routing (S 42405, 42406)

The **DNP3 Process Gateway (accessDNP3)** supports, from zenon 7.60 `Select before Operate` and Command Routing.

The following functionality has been implemented:

▶ `Select before Operate` for Binary Outputs and `Analog Outputs`

- Configurable for each data point

- Configurable timeout

▶ Command routing option for `Binary Outputs`

- A correspondingly-configured command group with auto-remote action is required

- Feedback to the master for a successful `Select` or error during `Select` to a downstream IED or controller.

- Feedback to the master for a successful `Operate` or error during `Operate` to a saved IED or controller.

- Feedback to the master for active command interlockings with auto-remote action

▶ Automatic amendment to value `0` or `1 when interlocking variables of USINT data type for binary outputs` of:

- `trip / close`

- `latch on / latch off`

- `pulse on / pulse off`

11.8   Process Recorder (F 7962)

The **Process Recorder** module, which must be paid for, offers you the possibility to record process data in the productive Runtime. At a later point in time, this recorded data can be played back again in Runtime in a project simulation client.

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

1. Recording of processes

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. Playback of the recording

The recorded data is visualized again in Runtime by means of project simulation.

The playback of the recorded values is visualized in the existing zenon screens.

Playback is controlled with a zenon `Process Recorder` screen.

11.9   Project simulation

11.9.1   Licensing

From zenon 7.60, the **project simulation** is included with the payable modules **Command Sequencer** and **Process Recorder**.

The **Project Simulation** module can no longer be licensed as an individual module.

11.9.2   Function - amendments for Process Recorder (S 29588)

The **Activate/deactivate project simulation** function was enhanced with the option for the **Process Recorder** module.

New entries:

  ▶ **Activate simulation**

    • **Initialize with process image and Runtime files**

  ▶ New tab:
    Playback

11.10   Reporting

11.10.1   Additional columns in the data set for AML for alarm causes (S 32657)

For the Report Viewer AML data set, five new columns were added, for the display and evaluation of alarm causes:

- ▶   ALARMCAUSELEVEL1
- ▶   ALARMCAUSELEVEL2
- ▶   ALARMCAUSELEVEL3
- ▶   ALARMCAUSELEVEL4
- ▶   ALARMCAUSELEVEL5

11.11   Recipegroup Manager (S 19885)

VALIDATION OF RECIPES

Recipes of the recipe group manager are now validated at all points before writing. In doing so, a check is made to see whether all values in the recipe are in a valid range of values.

Validation is active by default and can be deactivated using the **Invalid recipes writeable via function** property in the Recipegroup Manger group.

11.12   Shift management (F 1502, 8887, 10673, 18731)

Shift management is a module in zenon for the modeling and organization of shifts. This module is not available in global projects and CE projects. A shift calendar is shown in Runtime. It scales depending on the time interval to be displayed.

You can do the following locally and in the network with the help of shift management:

- ▶   Create shifts
- ▶   Enter shift times

- ▶ Administer shift times

- ▶ Enter and administer break times

- ▶ Link equipment groups to shifts

- ▶ Evaluate shifts with the zenon Analyzer

- ▶ Create and apply shift modules

- ▶ Copy and paste shifts and shift models

The creation, editing and deletion of shifts and shift models can be limited to certain users by means of function authorizations.

When creating and editing shifts and shift models in the network, the following is applicable:

- ▶ Editing is not carried out on the server directly, but with a local copy.
  All changes are sent to the **Server 1** and to the **Server 2** and updated there.
  This concerns the creation, changing and deletion of shifts and shift models, as well as the insertion of shift models into the shift calendar.

- ▶ If several users are editing the same shift or the same shift model, the changes made by the user who has saved last are applied.

- ▶ When the model is saved, only the content is saved, not the name and description.

11.12.1 Shift model (F 8887)

Shift models provide templates for shifts.
They are shown as an alternative to shifts and configured there.

Shift models can be created, edited, applied and deleted locally and in the network.

Configuration of the list of shift models (F 1502)

The columns of the list for the shift model can be configured in the screen switching. The List of Shift Models tab is now available for this. Columns can be selected and their graphical appearance can be configured.

11.12.2 Filter shifts using an equipment model screen (F 18731)

Shifts can also be filtered in Runtime using an `equipment model` screen. To do this, in screen switching to an `equipment model` screen in the **Screens to be updated** option, the desired `shift management` screen is selected.

The dialog can be called up in Runtime using the **Filter** button.

If there is also filtering in screen switching for the `shift management` screen, the filter continues to limit the display in the shift management screen in the equipment model screen.

> ### Example
>
> *Initial situation:*
>
> ▸ In the screen switching to a `shift management` screen, there is filtering for the equipment groups `1` and `2`.
>
> ▸ In the screen switching to the `equipment model` screen, only Group `1` is offered.
>
> ▸ In the **Screens to be updated** option, the `shift management` screen has been linked.
>
> Result: In the `shift management` screen, there is only filtering for equipment group `1`.

11.12.3 Link shifts to users and notify (F 8882, F 8974)

Shifts can be linked to users. These users can be assigned to a user group and thus be selected as the recipient of messages by the **Message Control** module. The **Send a Message** function synchronizes the user of the user group with the shift and notifies the users available for **Message Control**.

11.12.4 SQL Export (F 11220)

Shifts and breaks can be exported to SQL.

11.13 Variables

11.13.1 Placeholder for additional variable information in dynamic limit value text (S 39503)

To display additional variable information (value, identification, unit of measurement, etc.) in dynamic limit value texts, all placeholders that are also supported for the combined element are now supported.

In doing so, the display of the limit value text is supported via the `dynamic text` element.

12. zenon Web Server

12.1 HTML Web Engine

Changes and new functionalities for the HTML Web Engine:

12.1.1 AML and CEL (F 25996, F 26731, F 32953)

AML AND CEL: DYNAMIC DISPLAY IN THE HTML WEB CLIENT

The following is now applicable for AML and CEL:

▸ AML and CEL are dynamically amended on the HTML web client.

▸ All supported columns are updated. No new columns are added.

▸ No more static loading.

AML AND CEL: COLUMN DISPLAY

The **Identification** and **Resources label** columns are shown. The headers are labeled with **Identification** and **Resource Label**. The language of the texts used in these columns can be switched.

ACKNOWLEDGE ALARMS

Alarms can be selected and acknowledged individually. Acknowledgment is carried out by means of a button with a screen-specific **Acknowledge** function. Acknowledgment of all or the acknowledgment of a site is not possible.

ALARMS: ALARM CAUSE REQUIRED

Alarms for limit values can now be created with the **Alarm cause required** property activated. The acknowledgment of these alarms is however not supported by the HTML Web Engine.

USERS

With events triggered by the web engine, the users authenticated in the HTML Web Engine are shown for:

▶ Write set value

▶ Successful login

SORTING

Lists in AML and CEL can now be shown in sorted form. Text is sorted according to natural sorting.
Sorting is carried out by clicking on the column header.
Possible sorting:

▶ Ascending

▶ Descending

▶ Standard (as supplied in zenon Runtime)

TIME FORMATS

Date and time in AML and CEL are now shown in the localized display of the respective client. In doing
so, the **UTC-DateTime** is transfered and reformatted on the basis of the settings of the local computer.

12.1.2   Deployment Tool

An existing HTML Web Engine instance can be updated using its own option. This is how new builds and
new versions can be loaded on the fly.

12.1.3   Extended Trend (F 29756)

DATA AGGREGATION

Data is aggregated under certain conditions for the display of data in Extended Trend screens. The speed
of the display in the **HTML Web Engine** is thus sped up so that it corresponds to that of zenon Runtime.
Aggregation is carried out by coming up with the average value and filtering out redundant data points.

TIME FILTER

The following time filters are supported in the Extended Trend:

▶ **Absolute period of time**

▶ **Relative period of time**

All other time filters are not supported. Screen switching with invalid time configurations are not available in the HTML Web Engine. Buttons with corresponding calls are deactivated.

12.1.4   Language switching in Runtime (F 332037)

Language switching is now available in Runtime for the following columns of the AML and CEL:

▶ **Text**

▶ **Measuring unit**

12.1.5   Supported elements (F 32954)

ELEMENTS

The `Spot` option has been removed from the **Color gradient** property in zenon and implemented as its own **Spot** property (on page 30).
This change was undertaken for the HTML Web Engine.

STATIC ELEMENTS

Circle, **circle segment**, **polygon** and **rectangle** now support the **Invert color gradient** property.
When selecting `color gradient` for the **Fill pattern** property, the brightness gradient of the fill pattern is shown in inverted form.

DYNAMIC ELEMENTS

Element	Description
Button	Now also supports: ▶ **Fill pattern** ▶ **Color gradient**

12.1.6 Encrypted communication (F 8017)

The TCP connection between **SCADA Runtime Connector** (**zrsConnector.exe**) and **SCADA Runtime Connector Client** (**zrsConnCli.dll**) can now be configured with strong encryption (AES). To use the encrypted communication, issue an encryption password for Runtime and Client.

13. Network

13.1 Operating authorization (S 39741)

The equipment model has been added to the operating authorizations in zenon.

- ▸ The **Bedienberechtigung aktiv** project property has been replaced with the new **Operating authorization in the network** property.

- ▸ Selection of the type of operating authorization in a drop-down list:

  - `Inactive`
    No operating authorization required for operation in Runtime.

  - `Global operating authorization`
    Applicable for the complete project and all elements contained therein.

  - `Operating authorization via equipment model`
    Applicable for parts of the project, based on the equipment model.

COMPATIBILITY

Existing project configurations are also transferred to the **Operating authorization in the network** property during conversion.

13.2 Encrypted communication for remote Runtime driver and SCADA Runtime connector (S 7597)

Encrypted communication is supported by:

- ▸ HTML Web Engine
  Configuration in the zenon HTML Web Engine Deployment Tool

- ▸ Remote Runtime driver
  Configuration in the driver configuration

▸ SCADA Runtime Connector
for zenon and zenon Analyzer

The settings for this are configured in the **Startup Tool** in the **Network configuration** tab with the **Encrypt Runtime Connector communication** property.

14. Mobile applications

14.1 Everywhere Essentials Alarm Management

Everywhere Essentials - Alarm Management visualizes alarms of the productive Runtime on mobile devices. Filtering of the alarms according to equipment groups can be configured in the connection. This app is available for devices with the Apple iOS operating system.

14.2 Everywhere App by zenon allows self-signed certificates (F 32092)

From Android version 4.4, it is now possible to use signed certificates with the Everywhere Server by zenon. Official certificates continue to be the only ones that can be used for versions before this. Version 4.03 or higher must be used for the use of certificates.

For each profile, a certificate installed on the device can be selected and linked. This certificate is used when establishing a connection.

15. Programming interface

15.1 Add-Ins (F 34882, F 349883, F 35660, 35661, F 36606, F 37779)

The new COPA-DATA Add-In interface provides the possibility to enhance the functionality of zenon and to provide help with project configuration.

Add-Ins call up:

- ▶ In the Editor: Entries in the **Tools** menu.
  Add-Ins are imported and installed and can then be used in the Editor.

- ▶ In the Runtime: Via the detail view of the **Programming interfaces** node, **Add-Ins** subnode.
  Add-Ins are imported and saved in the Runtime folder. They are installed when Runtime is started and can then be used.

Add-ins are available as:

- ▶ Wizard: Is started by the user, is run through and ended again after solving the task. Other windows are possibly not accessible during execution. A wizard can run with a user interface or in the background.
  For details on the wizards supplied with zenon, read the Wizards manual.

- ▶ Service: Is started with the Editor or Runtime and runs in the background.

The following is currently supported in programming environments:

- ▶ SharpDevelop (included in the installation package)

- ▶ Microsoft Visual Studio (available via the Visual Studio Marketplace)

When using zenon COM objects with self-created user controls or external applications, they must be enabled using the **Marshal.ReleaseComObject** method. Enabling by means of the **Marshal.FinalReleaseComObject** method must not be used, because this leads to a malfunction of zenon add-ins.

15.2   API - ALC markers for line, polyline and pipe (S 39658)

The marker element for the normal 2-point line element has been revised. Previously, the marker element was positioned incorrectly if the points were not on a horizontal line.

If the percentage value exceeds 100%, it is shown with a maximum of 100%. If the percentage value is negative, it is shown at 0%. Previously the marker was incorrectly shown outside the element.

These marker elements are available for the following elements:

- ▶ Line

- ▶ Polyline

- ▶ Pipe

15.3   Read configuration of the visibility/flashing properties group via API (FS 36442)

The **IElement** class has been enhanced with new properties in order to be able to read additional information from the **Visibility/flashing** properties group in runtime:

▶ `tpVisibilityType VisibilityType;`
Reading the project configuration of an element for the **Visibility** property:
Enhancement of `IVariable* VisibilityVariable;`

▶ `IInterlocking* VisibilityInterlocking;`
Reading of the **Interlocking** if `From interlocking` has been configured for the **Visibility**.
Enhancement of `VARIANT_BOOL VisibilityFromLimit;`

▶ `VARIANT_BOOL VisibilityWhenInterlocked;`
Reading the variable if `Von Variable` has been configured for the **Visibility** property.
Additional reading of the configuration of the **from** and **to** properties.
Enhancement of `double VisibilityFrom;` and `double VisibilityTo;`

16. Styles (F 21624, F 2204)

The styles, already known from zenon 7.50, have been consistently enhanced in version 7.60:

▶ Styles are now also available in the local project and can be edited externally using the zenon API or XML. Furthermore, they can be easily duplicated with copy and paste.

▶ The zenon screen elements have been consistently made "style-compatible" and there is thus nothing preventing styles being used with the dynamic screen elements.

Furthermore:

▶ Additional entries for multi-users have been added to the **Styles** menu.

▶ Line styles can be linked to a **trend element**.

16.1 Trend Element (F 12432)

The trend element now has a limited number of permitted curves. The maximum number is 32.

17. Drivers

17.1 All drivers

17.1.1 All drivers - driver selection dialog revised (S 30729)

The dialog to select a driver when being created in zenon has been redesigned:

▶ The available information of the dialog has been rearranged.

▶ The size and position of the dialog can be changed.
   This is saved remanently.

17.1.2 All drivers - driver configuration via API

The **General** tab in the driver configuration dialog can be exported or configured via API for all zenon drivers.

💡 **Information**

This possibility applies to all zenon drivers.

17.1.3 API enhancement for various drivers (S 42564)

The complete driver configuration can be exported or configured by means of API for the following drivers:

▶ **Fraport**

▶ **IEC850**

▶ **IEC870**

▶ **Logix32**

▶ **S7TCP32**

▶ **Sprecon870**

▶ **Steriflow**

▶ **Straton32**

▶ **Straton32_Micon**

17.1.4 Performance optimization - driver initialization

The performance when initializing drivers in the zenon Editor and Runtime has been increased considerably.

Drivers are now - depending on the number and availability of the cores on the computer doing the execution - registered and started at the same time.

Note: With network projects, the computer that is the primary server is the one doing the processing.

The shutting down of drivers is also carried out at the same time.
When closing or shutting down zenon Runtime, it is ensured that all commands are closed first, before Runtime is closed.

17.1.5 New ConnectionStates driver variable (S 48549)

The **Communication details** driver object type has been enhanced with information about the connection status of the driver:

▶ `ConnectionStates` (Offset 61)

17.1.6 New entries in the driver data sheet

The following entries have been added to the driver data sheets:

▶ alternative IP address

▶ extended API

▶ Hysteresis

▶ Supports status bit **WR-SUC**

17.1.7 Renaming of driver variable driver object type to communication details (Def. 37610)

The existing **driver variable** driver object type has been renamed to **communication details**.

17.2 3S_V3

17.2.1 3S_V3 - XML file import supports ENUM data types (S 26374)

XML import from a symbol file supports the correct data type with ENUM data types.

When importing variables as an XML from an XML symbol file, the zenon variable now has the same data type as defined in the controller.

17.2.2 3S_V3 Driver - Redundancy (S 29104)

The establishment of a connection to a redundant controller through configuration of a secondary node address.

As an option, the status of the connection can be queried by means of a variable. To do this, the new **connection status** driver object type was implemented for the 3S_V3 driver.

17.3 Selection of 256 serial communication interfaces - all drivers with serial interface (FS 36542)

256 serial interfaces are supported for drivers with a serial interface.

To do this, the drop-down list to select the communication interface has been extended to 256 in the respective driver configuration dialog. With drivers that support fewer interfaces by default, the drop-down list is reduced in size accordingly.

17.4 BACnetNG

17.4.1 BACnetNG - read trendlogs (S 28963)

The BACnetNG driver now also supports Trendlogs.

There are new driver object types available for this:

- ▶ **Trendlog**
- ▶ **Trendlog - internal total record count**

17.4.2 BACnetNG - Foreign Device (S 38213)

In order to implement communication to BACnet devices in a different subnetwork, communication via BACnet Broadcast Management Device (BBMD) is envisaged in the BACnet standard.

The driver configuration dialog of the **BACnetNG driver** has been enhanced in the **Settings** tab with configuration fields for signing in as a foreign device on a BBMD.

17.5 BeckhNG - new data types LINT and ULINT (S36800)

The **BeckhNG driver** now also supports LINT and ULINT data types.

The new data types are also supported during import from a TPY file.

17.6 COMLI (F 23879)

The **COMLI driver** communicates with the ABB COMmunication LInk protocol via a serial interface. zenon is always the master with this communication.

- ▶ Supported COMLI telegrams: 2, 3, <, =, J, K, M, O, P, Ä, Å
- ▶ **Time marked events** are represented for the driver with their own driver object type. These events can then be linked to the corresponding variables to a reaction matrix in zenon. As a result, the **Time marked events** can be written to the chronological event list or the alarm message list.

17.7 EIB-KNX bus driver

17.7.1 Offline variable import from .esf file (S 38058)

The variable import of the EIBV-32 driver has been converted to offline import for zenon version 7.60:

▶ Variables can be imported from an ETS project configuration in zenon by means of offline import.

▶ The variables are imported from an ETS export file with the .esf file format. An ETS export file from version 4 is required for this.

Please keep in mind: Online import for the **EIBV2_32 driver** will no longer be supported by this driver from zenon 7.60.

17.7.2 Main group address increased to 32 (FS 37548)

The address of the main group has been increased to 32 (addresses `0` to `31`).

17.8 IEC850 (F 39549)

17.8.1 Support for several variables with the same addressing (S 39550)

Value changes are distributed to all variables with the same addressing.

This is applicable for addressing via **Symbolic address**, **Name** or **Identification**.

17.8.2 Support for INT64(U) bType as LINT, ULINT or LREAL (S 39550)

▶ Enhancement for bType INT64 in the driver
The **IEC850 driver** now also supports the `LINT` and `ULINT` data types.
These data types are used by default, when importing variables from the driver (online from the PLC or offline from a file) for 64-bit integers (`signed`/`unsigned`).
These can be amended manually to `LREAL`.

> ### 💡 Information
>
> These enhancements are also applicable for zenon Logic.

17.9   IEC-62056-21

17.9.1   Initial Baud rate can be configured (FS 37678)

The initial Baud rate can now be configured under **Settings**.

17.9.2   Timeout and repetitions can be configured (FS 36778)

New configuration options for connections:
   ▶   **Timeout**
   ▶   **Connection repetitions (Retries)**

In addition, the documentation has been revised.

17.10   OPCU-UA server - current stack (F 8938, S 23759)

The **OPC UA Ansi C Stack** was updated to version 1.02-336.1 for zenon drivers, zenon Process Gateway and zenon Logic OPCUA Server.

17.11   INTEGRA

17.11.1   INTEGRA - REAL data type for driver object types (S 44140)

The following driver object types now support the REAL data type:

- ▶ **Data Point (Data Block)**
- ▶ **Data Point (Inst. Data Block)**
- ▶ **Data Point (Marker)**

17.11.2 INTEGRA - alarm handling (FS 37133, S 44141)

The alarm handling of the **INTEGRA driver** has been revised:

- ▶ Initialization of the alarms for each establishment of a connection by means of the **Alarm with time stamp** driver object type.

- ▶ Reading of the program structure from **DB2** by means of the new **DB structure** option in the driver configuration dialog.

- ▶ Reading of the initial values from **DB3**.

- ▶ Removal of the old **Alarm bit** driver object type.

REDUNDANCY

In addition, the configuration fields for redundancy have been compiled into a configuration field for **Server 1** and **Server 2**. The same respective configuration is used for both servers.

17.12 Jetter - LREAL data type (FS 36290)

The driver configuration for the **JETTER.exe** driver has been updated to the most recent version, 1.27.0.3627.

As a result, the `LREAL` data type is supported for the new **Register - Double** driver object type.

17.13 LON32 - compatibility with LonMaker 3.24 (36821)

The **LON32 driver** has been amended for communication with the current LonMaker version 3.24.

17.14   Modbus RTU

17.14.1   Modbus RTU and Open Modbus TCP (S 25021)

The default setting of the net address has been amended from `0` to `1`.

17.14.2   Modbus RTU - WSTRING for holding register (FS 36145)

The Modbus RTU driver MODRTU32.exe now also supports the `WSTRING` data type for the **Holding Register** driver object type (channel type 8).

17.15   Modbus Energy

17.15.1   Direct redundancy switching to secondary IP address in the event of a failure of the primary connection (FS 37567)

In the event of a failure of the TCP connection, there is a direct switch to the secondary IP address. In doing so, there is no longer a wait until the next successful establishment of a connection to the primary IP address.

17.16   NMEA 2000 - support of several instances for PGN parameters (FS 36451)

The **NMEA 2000 driver** supports instances for PGN.

The number of the instance is coded in the **Net address** of the variable:

Calculation: *Instance number * maximum net addresses + net address of the PLC*

17.17 OPC UA (FS 36017)

17.17.1 Enhancements (FS 35862)

DATACHANGEFILTER CAN BE CONFIGURED:

▶ The **Data-Change trigger** property supports the "`STATUS_VALUE_1`" and
"`STATUS_VALUE_TIMESTAMP_2`" status.
If `STATUS_VALUE_TIMESTAMP_2` is configured, the driver includes a `DataChangeFilter` in
the `CreateMonitoredItemRequest` and requests the server to send a
`DataChangeNotification` accordingly if the time stamp has also changed. The driver sends
an updated value to Runtime if the time stamp changes a monitored element.

▶ The new **Use positive hysteresis of variable as AbsoluteDeadband** property causes the driver to
include a `DataChangeFilter` im `CreateMonitoredItemRequest` with a correspondingly-
configured `AbsoluteDeadband`. In the project configuration in the Editor, for the variable in
Positive for signal property (**Value calculation** variable property group), there must be a value
defined that does not equal 0.

CONFIGURABLE ADDRESSING

- `via Persistent NodeID`
- `Namespace and BrowsePath`

> **Information**
>
> The "**Persistent Node ID**" option can only be activated for projects that have the corresponding information entered into the configuration in the **Symbolic address** property.

17.18   PRISMIC (F 31924)

The **PRISMIC** driver receives Event Records from Prismic PMS via Ethernet, decoded, and formats these and forwards them to zenon.

Nine PRISMIC event types are represented for the driver with their own driver object type. These events can then be linked to the corresponding variables to a reaction matrix in zenon. As a result, the events can be written to the chronological event list or the alarm message list:

For decoding, the driver uses .CSV files for certain event types. As a result, content such as register number or status information can be converted to readable texts. This content can be configured accordingly.

17.19   RemoteRT - encrypted connection and blockwise reading (S 24980)

Enhancements to the **remote Runtime driver**:

▶   Block Read:
The **remote Runtime driver** (**RemoteRT.exe**) now supports blockwise reading of values. If blockwise reading is not possible, the values are read individually.

▶   Spontaneous communication:
The **remote Runtime driver** can read values from Runtime spontaneously. This can be activated in the driver dialog with the new **Use spontaneous mode** property.

▶   User name and password:
The entry of a user name and password is supported for encrypted connection to a net address.

17.20   S7TCP_32

17.20.1   New data types for variable import

When importing variables form a Step7 or TIA project by means of **Import S7 project...**, the following new data types are supported:

- ▶ **LREAL**
- ▶ **LINT**
- ▶ **ULINT**
- ▶ **WSTRING**

17.20.2   Addressing by means of host name or IP address (S 29101)

The connection to an S7 controller via TCP/IP can be configured by means of host name or IP address. This is possible for both the primary connection as well as the secondary connection.

17.21   S7-TIA - S7 driver for S7-1500/1200S7TIA

17.21.1   Performance optimization through precompiling of TIA project files (FS 36026)

With the external **TIAtoAGL** command tool, comprehensive TIA project files can be pre-compiled and optimized for further processing (such as online variable import).

The activation of this optimized file is triggered in the **Options** tab of the driver dialog.

17.21.2   Variable import from PLC directly (FS 37563)

With the new **Symbols from PLC** option field in the **Options** tab of the driver configuration dialog, variables can be taken directly from a PLC in the project configuration in the <CD_PRODCTNAME> Editor. No TIA project file needs to be linked for this option.

17.22   SAIA2ND32 - support for PG 5.2.x (F 23886)

From zenon 7.60, it is now possible to communicate with the controller using version 5.2 SAIA PG Software or higher.

The support for older software versions has been discontinued by the manufacturer.

17.23   System driver

17.23.1   Name of system driver variables can be changed (S 268750)

The name of system driver variables can be amended from zenon 7.60. The string length can also be changed for system driver variables of the string data type.

⚠ **Attention**

Please note that, with a change ot an existing system driver variable with the **Typ** property in the **System driver variable** property group, the name is not changed automatically.

In this case, it is urgently recommended that the **Name** and **Identification** properties are amened manually.

17.23.2   [System Information] theme (S 27615, S 24979)

The **[system information] Runtime status (simulation)** theme has been enhanced with the "`Process Recorder playback (5)`" status.

New system driver variables:

▶   **[system information] operating system user**
    Display of the user who has started Runtime on the current computer.

▶   **[system information] dongle status**
    Display of the current dongle status.

▶   **[system information] Runtime version**
    Display of the current version of Runtime.

▶   **[system information] time offset to UTC**
    Display of the time difference to UTC in seconds.

▶ **[system information] daylight saving time / standard time**
Display of the type of time on the computer.

17.23.3 Theme [user administration] (S 42281)

New system driver variables:

▶ **[user administration] system lock**
Current status of the system lock

17.23.4 Theme [printer] (S 42282)

New system driver variables:

▶ **[Printer] report printer - Number of print jobs**
Number of report printer print jobs still to be processed.

▶ **[Printer] report printer - name**
Current name of the report printer.

▶ **[Printer] report printer - status**
Current operating status of the report printer.

18. Connectors

18.1 Connector enhanced (F 24006)

The connector (container, plug-in, client) has been enhanced:

▶ Alarm causes can also be issued by the connector.
The zenon project must be at least version 7.60 for this.

▶ Alarms can obtained after being filtered according to alarm causes.

The context list of a project can be obtained.
The zenon project must be at least version 7.60 for this. An empty list is shown otherwise.

19. Tools

19.1   3D Integration (F 25999)

The **3D integration package** from zenon offers an easy and simple possibility to link 3D files from a CAD program to project configurations in zenon. The structure of a 3D model is transferred and visualized in a preview.

▶ In the **3D Configurator**, a zenon application, 3D CAD files can be linked to zenon project configurations in a graphical user interface.

The selected assembly group or an individual object can be:

- Assigned to one or more variables.
  If a variable is linked, the visibility, flashing and color settings are taken from the variable.

- Assigned to one or more function(s).
  If a function is linked, the function is triggered by clicking on the object in Runtime. This is visualized in Runtime with a different mouse pointer.

- Be assigned to a variable in addition to a camera position.
  If a camera position is linked, the position is called up if the limit value of the linked variable is breached.

▶ The project configuration is displayed in zenon Runtime with a **zenon WPF element**.
  Display of the 3D project configuration in Runtime in a zenon screen.

- Free navigation in the 3D model:
  The display can be moved, rotated, enlarged or reduced.

- Execution of functions in the 3D model:
  A configured function can be executed by clicking on an object or an assembly group.

- Calling up the 3D model in a defined perspective:
  The 3D model with views of a configured position can be visualized by setting a value of a "camera variable".

- Visualization of a limit value breach:
  When a limit value is breached, an object or an assembly group can be shown in color or flashing in the 3D model.

- Objects or assembly groups can be switched to visible or invisible.

19.2   Diagnosis Viewer (S 34519)

SEPARATE QUEUE PER THREAD

Instead of a joint logging queue for all threads, a separate internal queue is used for each thread. More LOG entries can be buffered as a result. Runtime is thus placed under less load with complex loggings with many LOG entries.
Example: configured **Messagelevel** in the Diagnosis Viewer: **Deep Debug**.

Exception: If the setting for the number of the buffered LOG entries is amended in Runtime, the queues for all threads are reassigned. Logging is blocked whilst this new assignment is running.

INCREASED PRECISION OF THE TIME STAMP (FROM WINDOWS 8)

The time stamp of the LOG entries has been reimplemented. As a result, it is guaranteed that, for Windows operating systems from version 8, each entry gets a unique time stamp.

For Windows 7 (or lower), the behavior remains unchanged:
The time stamp has a precision between 1 ms and 16 ms. Messages created quickly after one another get an identical time stamp.

19.3   GIS Integration (F 28850)

The **GIS Configuration Tool** allows objects to be designed with a geographic reference and to link these objects with zenon ALC information, variables and functions.
Display in the zenon Runtime visualizes ALC engineering with selectable Map providers.

19.4   Startup Tool

19.4.1   OCX registration (S 45531)

The checkboxes for OCX registration and deregistration in the Startup Tool under **Application -> Options -> General** have been removed. The corresponding files are now automatically registered and deregistered correctly.

19.4.2   Updating of start dialog (S 45530)

The **Update Help** button, through which the **Documentation-Download-Tool** can be started, has been added.

In addition, the **Documentation-Download-Tool** can also be started from the **Help -> Update Help...** menu.

20. Variables

20.1   Variable diagnosis screen type enhanced (S 29404)

In the **variable diagnosis** screen type, the list of variables has been enhanced with the following column entries:

▸ **Actual value (raw data)**: in the value range of the controller

▸ **Status (binary)**: Status bits of the variable

▸ **Time stamp**:
This new time stamp is shown in microseconds [μs].

- `internal (UTC)`

- `internal (local)`

- `external (UTC)`

- `external (local)`

▸ **Symbolic address**

20.2   Export - Import (D 36309)

When exporting variables and subsequently importing them, it was possible in some cases that the linking to properties of the data type get lost. If, for a data type, the parameters of one of the following properties has been set and a variable derived from this, then the linking was overwritten when importing previously exported variables:

▸ **Limit Values** group:

- **Variable** for **Dynamic limit value active**

- **Alarm/Event Group**
- **Alarm/Event Class**
- **Function**
▶ **Alarm handling** group:
- **Acknowledgement variable**
- **Visual acknowledgement variable**
- **Alarm area 1**
- **Alarm area 2**
- **Alarm area 3**
- **Alarm area 4**

The export process has been amended to prevent this. Variables are exported and imported correctly again.

Procedure in version 7.60:

▶ If the variables have not been imported yet, they need only be exported again.

▶ If variables have also been exported and have lost their linking, they must be amended individually and linked to the property again.

▶ If the XML export files continue to be used in external applications, these applications must be amended. The information must be read from the data type instead of from the variable.

20.3 Name of system driver variables can be changed (S 268750)

The name of system driver variables can be amended from zenon 7.60. The string length can also be changed for system driver variables of the string data type.

> ⚠ **Attention**
>
> Please note that, with a change ot an existing system driver variable with the **Typ** property in the **System driver variable** property group, the name is not changed automatically.
>
> In this case, it is urgently recommended that the **Name** and **Identification** properties are amened manually.

20.4   New status bit PR_NR (F 7962, S 29380)

For the **Process Recorder** module, the status bit **PR_NR** (no value present in the Process Recorder's recording) has been introduced.

This status bit shows in Runtime that the variable has not been recorded during playback in the Process Recorder. If the status bit is active, this status is shown in a yellow square in a screen element during playback.

20.5   Two time stamps for actual values (S 27995):

Actual values of the variables contain two time stamps from zenon 7.60:

▶   `Internal time stamp:`
     Time point at which the driver has received the value from the controller.

▶   `External time stamp:`
     Time point at which the controller supplies the driver (optional, not possible in all protocols).

The following rule is applicable for display in Runtime:

▶   External time stamp:
     This is shown if a valid external time stamp is available.
     The value has **T_EXTERN** as an active status bit and no **T_INVAL**.

▶   Internal time stamp:
     If no external time stamp is available or the external time stamp is invalid (**T_INVAL**), the internal time stamp is used for the display of values.
     The value has no active status bit **T_EXTERN** or the active status bit **T_EXTERN** also has the status bit **T_INVAL** active.

The **T_STD** status bit - standard time (winter time) - has been divided into two status bits:

▶   **T_STD**:
     The status bit accompanies an internal time stamp

▶   **T_STD_E**:
     The status bit accompanies an external time stamp

Only one of these two status bits is active. Activation follows the above-described rule.

With external time stamps, in order to be able to better distinguish invalid time stamps received and imprecise time stamps received (such as in the IEC 61850 protocol), the new status bit **T_UNSYNC** has been introduced.

The **variable diagnosis** screen type has been amended and enhanced accordingly.

20.6   Minimum string length for string variables changed from 0 to 1 (S 30718)

the minimum **String length** of a variable has been amended from 0 to 1.

When converting from zenon projects from a version before 7.60, a project configuration of this property with the value 0 is automatically replaced with the value 5.

20.7   Optimization of the entry of REAL values (S 33296, S 40854, S 42855)

VALIDATION WHEN ENTERING VALUES WITH DECIMAL POINTS

The entry of values in Runtime for variables with a numerical data type has been optimized. When entering, only as many decimal points as are configured for the variable in the Editor are permitted.

The following input methods have been taken into account in the process

- ▶   Value entry via **Keyboards**
- ▶   Value entry via the **Setpoint input** dialog
- ▶   Direct entry of values via the **Numeric value** and **Dynamic text**

If there is an invalid entry, a corresponding message is shown.

FURTHER VALIDATIONS

The entry of values depending on the **Base for display** set and the **Data Type** used has been optimized. Only valid characters can be entered.

21. Wizards

21.1 Metering Point Administration

21.1.1 Filtering of measured values by means of variable (Def. 37304)

In the **metering point administration**, there is now the possibility to filter entries in Runtime using a string variable.

21.1.2 Semi-automatic correction of invalid relative values (F 25000)

In the event that recorded relative and absolute values are invalid, the possibility of a quick, semi-automatic correction has been implemented in the **metering point administration**.

To do this, the new **Correct invalid values** button has been added to the administration of the metering points. The check for invalid values and the display and correction of these is carried out in a separate dialog.

21.1.3 New "Validation" metering point type (F 30979)

The **validation** metering point type was added to the **metering point administration**. Archived measured values can thus be confirmed or amended. However, the original measured values are always retained.

21.2 Project Configuration Wizard (S 42645, 42646, 42833)

The **Project Configuration Wizard** has been ported into the new COPA-DATA add-in interface in zenon 7.60.

In addition, the wizard has been enhanced with the optional possibility to configure diagnosis screens (Dashboards) with the wizard automatically. These diagnosis screens can also be created for existing projects when the wizard is executed.

The wizard replaces the **Project Wizard** from the previous version.

22. Important information

22.1 ActiveX Controls

If special ActiveX controls are developed, the following has to be considered:

If the DISPATCH – which is passed in the **zenonInit** event of zenon – is saved in the ActiveX control, an **AddRef** has to be carried out because this DISPATCH is only valid within the **zenonInit** event. If "**AddRef**" is not called, a crash of the entire Runtime will be the result. Additionally, a release has to be performed in the "**zenonExit**" event.

22.2 Alarm status line and Windows application switching under Windows 7

If, on a computer with the operating system Windows 7/Server 2008, applications that are running are switched through several times using the `Windows key + Tab key`, the following behavior can occur when selecting zenon Runtime:

▶ The alarm status line is switched to the background

▶ This can only be brought back to the foreground by the user intervening.

This behavior results from the operating system. Microsoft provides the hotfix **2587473** to rectify this. This can be requested from Microsoft directly: http://support.microsoft.com/kb/2587473/en-us (http://support.microsoft.com/kb/2587473/en-us).

22.3 Screen-type specific functions (Def. 31123)

If screen-type specific functions are invalid, for example as a result of copying a button to a screen of a different type, then the invalid linking is pointed out and this can be replaced. Invalid functions are removed during compiling.

When converting projects, these can contain screen-type specific functions that are still invalid in Runtime.

22.4 Screen elements with the same ID

Each element in a screen must have a unique ID. If several elements with the same ID are used in a screen, all duplicates are removed during compilation.

Example: If a button is copied and inserted into the same screen, it has the same ID. The copy is removed during compilation.

Exception: Several containers can be created in a `Faceplate` screen.

22.5 Integration of VBA wizards and VSTA wizards

All VBA wizards are saved in the file "zenWorkspace.vba" by the zenon Editor. All VSTA wizards are saved in workspace AddIn.

When performing a new installation, these files will only be copied to your computer if they do not already exist in the installation folder. Existing VBA/VSTA files are not overwritten, because all your changes would be deleted in this case. If you want to use our new wizards or modified ones, you can import them manually via the menu "**File – Update Wizards**" in the Editor. At this you can decide yourself which wizards you want to overwrite.

22.6 Erroneous line display if extended graphics mode deactivated

In the extended graphics mode, dashed lines with a line width >1 can be drawn. If you deactivate the extended graphics mode and zoom onto the line, the line will be displayed as solid.

22.7 Notes on the enabling of zenon COM objects in conjunction with add-ins

When using zenon COM objects with self-created user controls or external applications, they must be enabled using the **Marshal.ReleaseComObject** method. Enabling by means of the **Marshal.FinalReleaseComObject** method must not be used, because this leads to a malfunction of zenon add-ins.

22.8 Complex vector graphics

Please note when configuring process screens. When using many or complex vector graphics, loading screens in the Runtime can take a long time.

22.9 Converting existing data

If a project is started in Runtime version 7.x for the first time, the Runtime files of the concerned modules are converted. This guarantees that data changed in online operation is not lost. To do this, please read about conversion of projects (on page 74) in the General (on page 11) chapter.

Attention: All files have to be created in the Editor for the new version; otherwise the project cannot be started!

22.10 Converting projects

Before you convert a project, please read back all Runtime changeable files (User Administration, Standard Recipes, Recipegroup Manager, Scheduler/PFS and Message Control) into the Editor. This ensures a complete data conversion and makes sure that none of the changes made in the Runtime are lost. After converting to the new version, create all Runtime files once including RT changeable data.

Note: You can find important information for the conversion of certain versions in the zenon help in the Project conversion manual.

CONVERTING MULTI-USER PROJECTS

Multi-user projects can only be converted if no elements are checked out. This means that all people configuring projects have to **accept** their changes first.

If this is not possible for some reason, you have to create a project backup of the project on the project database server and then immediately restore it. This resets all the **Under construction** information. Attention: All changes in the local project versions are lost!

The conversion can only be done on the PC, on which the central project database resides. If there is no Editor on the PC (standalone database server – no longer supported), you must install the Editor first. Only after that can the conversion be done on this PC.

CONVERTING PROJECTS WITH EMBEDDED FONTS

From zenon 7.50 onwards, the "**embedded font**" property is no longer available in the "**static text**" element.

In order to continue to be able to use project configurations that use this property in earlier versions, carry out the necessary amendments in zenon 7.20 with the **Static Text Font Conversion Wizard**.

> ⚠ **Attention**
>
> Please ensure that your existing zenon projects are first prepared in version 7.20 with the **Static Text Font Conversion Wizard** accordingly, before you convert your project to a higher zenon version.
>
> The wizard is only included in current builds of version 7.20.

CONVERSION FROM VERSION 6.01 TO 6.20

zenon projects in version 6.01 or 6.20 can no longer be directly read back in zenon 7.10 or higher.

Background: Versions that are based on the MSDE (SQL Server 2000) are not compatible with the SQL Server 2012 used in zenon.

Solution: First convert in zenon 7.0 and then in 7.10 or higher.

22.11   MS-ActiveX element DBGrid32.ocx does not work

There a several problems known in context with the use of Microsoft ActiveX element **DBGrid32.ocx** in the Runtime. Therefore please use other ActiveX elements such as **MSDATGRD.ocx**.

22.12   Reload of projects with Simulator driver variables

Simulator driver variables, not projected as HD variables, are reset to the value 0 with the function "**Reload**". Only HD simulator driver variables keep their value after reloading.

22.13   Network access - Firewalls

Different components of zenon try to access the network and can cause an alarm by firewalls or personal firewalls. If you want to use the network or the zenon Remote Transport, you have to unlock the according TCP/IP ports.

The following zenon components result in network access:

- ▶   Administration service (`zenAdminSrv.exe`)
- ▶   Editor (`zenone32.exe`)
- ▶   Database server (`zendbsrv.exe`)

- ▶ Diagnosis Server (`zenLogSrv.exe`)

- ▶ OPC Server (`zenOPCsrv.exe`)

- ▶ Process Gateway (`zenProcGateway.exe`)

- ▶ Remote Desktop (`zenVncSrv.exe` and `zenVncCli.exe`)

- ▶ Network server (`zennetsrv.exe`)

- ▶ Transport service (`zensyssrv.exe`)

- ▶ Drivers with TCP/IP connections

- ▶ zenon Web Server (`zenWEBsrv.exe`)

- ▶ zenon Logic Workbench

- ▶ zenon Logic Runtime

22.14 Process Desk – killing tasks

The Process Desk of zenon now allows you to kill tasks that got stuck.

Attention: Some drivers need a certain follow-up time, because they write a process image on closing. Premature closing can result in data loss! Use this option only in case of emergency, when you are really sure, that the task will not close on its own.

22.15 Page preview and printout for report generator

In order to use the page preview and the printing of the Report Generator, a printer must be configured.

22.16 Saving reports of the Report Generator in the Runtime

Please be aware that on saving reports in the Runtime, all functions are replaced by the current contents of the cells (numbers). The functions in these reports (**.xrs** files) are no longer available. Additionally, these reports can no longer be edited in the Editor. So please use the MDI function "**Save as**" so that the original reports from the Editor are not overwritten. Moreover, we recommend to define the original reports as `read-only`.

22.17 The database server service must be entered correctly in the Startup Tool

Beside the versions you can also change the data base server with the **Startup Tool**. If you use this function, please note:

Between version 6.21 SP0 and 6.22 SP0 the SQL Service was entered incorrectly in the zendb.ini by the setup. This was no problem because the **zenDBSrv** did not consider the value. As of 6.22 SP1 this is the case again.
If you read the values using function **Read from zenDB.ini**, the values are stored wrongly in the **Startup Tool**. You must check existing entries and change them if necessary.

22.18 zenon Logic Intellisense is slow

For large programs the Intellisense function of the zenon Logic Workbench can cause the project to open very slowly. In this case you should deactivate the Intellisense function in the straton Workbench.

22.19 String arrays with straton32 driver

Several string arrays with the same sice can be read out correctly with the **straton32** driver only as of version 6.22 SP1 and zenon Logic Workbench SR7-3. If projects of older versions are converted, the string length must be changed for every string array in order for the communication to work.

22.20 Transport service Autostart

The transport service (**zensyssrv.exe**) is normally started automatically by the operating system when a user logs in. If the transport service is not started, the computer cannot be reached via the Remote Transport.
At a new installation it is restarted after the computer has rebooted.
If you accidentally delete the entry for the automatic start from the registry, you can restore it with the help of command `Register` in the Startup Tool. At this the transport service is also automatically restarted.

22.21 Overwriting Runtime files

When creating Runtime files in the Editor it can happen, that files changed in online operation are overwritten. This occurs with the following modules:

- ▶ Recipegroup Manager

- ▶ Production & Facility Scheduler or Scheduler

- ▶ User administration

- ▶ Standard Recipes

In order to guarantee that data created in runtime (recipes, schedules etc.) is not lost when creating Runtime files, there is a new tab in the dialog for project configuration: **Runtime changeable data** For the modules mentioned above you can define here whether the concerned files should be overwritten when Runtime files are created. If the checkboxes are not active, the files are overwritten!

This behavior is also true for the Remote Transport, when the Runtime files are to be transferred to another computer. So these checkboxes also apply here. If you want to transport all files to the remote system, deactivate all checkboxes. Otherwise the corresponding data will not be transported.

When creating Runtime files and when using Remote Transport, a message appears in the output window indicating that the concerned files were not overwritten.

The standard setting is: Runtime Files are not overwritten!

22.22   Wibu Key error message „WK1128"

If you get the error message **WK1128** when starting the Editor or Runtime, an obsolete version of **WibuKey** is being run. Install the current version of the **WibuKey** software from the installation medium.

22.23   WibuKey software installation removed from setup (Def. 34653)

The additional **WibuKey** administration software is no longer automatically installed with the setup from zenon version 7.50. This software is however supplied with zenon. If necessary, install the current version of the **WibuKey** software from your zenon installation medium:

- ▶   `...\AdditionalSoftware\WIBU-SYSTEMS WibuKey`

22.24   zenon in the Startup folder with dongle licensing

If zenon is started from the Startup folder, it may happen that it starts before the Wibu Key or Codemeter driver. Consequently, no dongle will be found and zenon will start in demo mode.

You can change this behavior by configuring a delayed start of the Runtime. For this, you need to make the following entry in the **zenon6.ini** file:

[DEFAULT]

```
STARTDELAY=[delay of the Runtime start in ms]
```

22.25   zenon Web Client: No support for Google Chrome from version 42

From version 42, Google Chrome no longer supports NPAPI plugins. Chrome can thus no longer be used as a zenon web client from version 42.

23. Encrypted communication (S 7597)

Encrypted communication is supported by:

- ▶ HTML Web Engine
  Configuration in the zenon HTML Web Engine Deployment Tool

- ▶ Remote Runtime driver
  Configuration in the driver configuration

- ▶ SCADA Runtime Connector
  for zenon and zenon Analyzer

The settings for this are configured in the **Startup Tool** in the **Network configuration** tab with the **Encrypt Runtime Connector communication** property.