zenon
by COPA-DATA

zenon manual

Runtime

v.8.20

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.

PROJECT SUPPORT

You can receive support for any real project you may have from our customer service team, which you can contact via email at support@copadata.com.

LICENSES AND MODULES

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

2 Runtime

Runtime is software that includes and supports graphic display possibilities for HMI/SCADA projects.

- Process display
- Archiving of data such as messages and process values
- Integrated alarm message list
- Recipes
- Multi-Touch

> **Attention**
>
> Note when configuring the project:
>
> - For the optimal display of zenon in the Runtime, the standard setting (corresponds to *100%*) is recommended for the Windows **display**. Higher values can lead to graphic elements, symbols, texts, etc. not being displayed correctly.
> - Windows themes can overlay elements in Runtime. Ensure, when configuring a project, that there is an appropriate distance from the elements to the screen edge.

Runtime is available as a 32-bit version and a 64-bit version starting with zenon 7.10. Projects can run on both versions. Details on compatibility with versions before zenon 7.10 can be read in the Project conversion manual, in the 64-bit version (on page 6) chapter. The Runtime is available as zenon Supervisor and as zenon Operator for Embedded operating systems. The versions differ in their functionalities. For details see Restrictions for zenon Operator.

SEVERAL INSTANCES OF RUNTIME

Only one instance of zenon Runtime can be started on a computer at any time. This applies regardless of whether Runtime is started as an EXE file, a zenon Web Client or as Runtime Control (OCX).

Exception: On the terminal server or terminal client, one instance of Runtime per user can be started as an EXE file, as a zenon Web Client or as Runtime Control (OCX). Only 1 instance can run at any time within a user context.

3 32-bit and 64-bit version

zenon is now available for Editor and Runtime as a 32-bit and 64-bit version. A 32-bit and a 64-bit zenon Editor or a 32-bit and a 64-bit zenon Runtime are installed on 64-bit operating systems. The file names of the executable files are identical for 32-bit and 64-bit.

On 64bit systems, all services present in 64-bit are registered and used in the 64-bit version. Editor and Runtime can be started alternately. Projects can be executed in both Editors and in both Runtimes.

Components such as, for example, zenon Logic Runtime, zenon Logic Workbench, drivers, Process Gateway and other tools are always only used in the 32-bit version.

GENERAL LIMITATIONS

The basic limitation of only 64-bit DLLs being able to be loaded in 64-bit processes also means that there are certain limitations when operating zenon. This mostly affects external components that are loaded in the Editor or Runtime. DLLs that are loaded using VBA/VSTA code and ActiveX controls are directly affected. These DLLs must be present as a 64-bit version for use in the 64-bit Editor or Runtime. ActiveX controls supplied by COPA-DATA are always available in 32-bit and 64-bit versions.

CONVERSION OF ZENON 5.50 PROJECTS

zenon 5.50 projects cannot be converted with the 64-bit Editor. These must be converted beforehand with the 32-bit Editor.

RGM LIMITATION

The Access database is no longer supported in the RGM. In order to be able to use MS Access data from previous versions under 64-bit, the project must first be converted in the 32-bit Editor. The **DataSource** property is no longer available as of version 7.10. For details, see the Converting Recipegroup Manager database chapter.

VBA

VBA was converted to VBA version 7.1. Therefore VBA is also available in zenon 64-bit. If, in the VBA code, Windows API or other imported DLL functions are accessed, these calls must be adapted to 64-bit. In general, the following applies: A VBA file created for a 32-bit version cannot be used without changes in a 64--bit version.

There are some defines/functions available in VBA in order to write 32-bit and 64-bit compatible code. For example:

```
#if Win64 then
Declare PtrSafe Function MyMathFunc Lib "User32" (ByVal N As LongLong) As LongLong
#else
Declare Function MyMathFunc Lib "User32" (ByVal N As Long) As Long
#end if
#if VBA7 then
Declare PtrSafe Sub MessageBeep Lib "User32" (ByVal N AS Long)
#else
Declare Sub MessageBeep Lib "User32" (ByVal N AS Long)
#end if
```

You can also obtain some useful notes on the porting of VBA 32-bit code to VBA 64-bit from Microsoft:

- Microsoft Office 2010, notes on porting:
  http://msdn.microsoft.com/en-us/library/ee691831.aspx
  (http://msdn.microsoft.com/en-us/library/ee691831.aspx)

COMPONENTS IN 32-BIT ONLY

The following components are also only available as 32-bit versions on 64-bit computers:

- Some programs, such as *SIC.exe* and *DiagViewer.exe*
- Licensing
- **Process Gateway**
- **Startup Tool**
- Windows CE
- Drivers
- zenon Logic Runtime and Workbench

4 Putting Runtime into operation

The procedure for putting Runtime into operation for the first time depends on whether the device you are operating is a turn-key device with Runtime installed or a device on which Runtime must be installed. If Runtime is already installed on the device, only the desired project needs to be copied to the Runtime device.

To do this:

- Save your project files on an external data medium.
- Use Drag&Drop or Copy&Paste to transfer these to your Runtime device.

or

- Get the project from the server on the Runtime device.
- Create a Client using the network topology.

Note: This is also possible by means of remote transport.

If Runtime is not yet installed on the device, it must first be installed. You can read how this works in the zenon standard installation chapter.

A license is required after installation in order to put it into operation. The license is transferred with the License Manager.

Note: The hard drive of the device you are operating may stop working after many years of use in a harsh industrial environment. To avoid loss of data, it may be a good idea to ensure that there is a redundant environment, such as working with both a Server and a Standby Server.

4.1 Installation

For the installation of zenon Runtime, start the installation process from your zenon installation medium.

This process automatically starts when the zenon installation medium is connected and guides you through the whole installation. Alternatively, it is possible to start the installation by executing **START.exe** in the root folder of your zenon installation medium.

Information

You can find further information in the installation and updates manual in the zenon standard installation chapter.

4.2 System requirements and operating systems

A detailed overview of supported operating systems and the required software and hardware can be found in the installation and updates manual.

4.2.1 System requirements when using DirectX

The following minimum requirements must be met when using *DirectX hardware* or *DirectX software*:

Note: For extensive projects or several projects loaded at the same time you will need accordingly faster/stronger hardware. The minimum requirements can increase as a result of this.

Parameter	Minimum requirements	Recommended
CPU:	Single core with SSE2 support.	Quad Core or more cores
Graphics card: (DirectX hardware only)	DirectX 11 mainstream graphics card. **Note:** When an integrated graphics chip is used in particular, it is possible, depending on the driver used, that there are impairments to the display quality.	Dedicated DirectX 11 AMD or nVidia high-end graphics card

Parameter	Minimum requirements	Recommended
Graphics memory: (DirectX hardware only)	1 GB VRAM **Note:** The size that is actually needed depends on the number of screens called up and the elements displayed.	2 GB VRAM
Driver graphics card: (DirectX hardware only)	The graphics card manufacturer's most recent driver.	
Operating system:	*DirectX hardware* and *DirectX software* only works on operating systems that support *DirectX11.1*. If the system does not support *DirectX 11.1*, it automatically switches to *Windows Enhanced*. The current *DirectX*- Runtime must be installed. For zenon it is installed together with the setup. For the zenon Web Client it must be installed manually.	

You can check the DirectX hardware compatibility of the graphics card and the driver with the Windows operating system tool **dxdiag.exe**.
Up to Windows 7: Check the **DDI version** value under **Display**. The value *11* for example means DirectX 11.
From Windows 8: All supported versions of DirectX are displayed in the **Display** tab under **Feature Levels**. For example, DirectX 11 is displayed as *11.0*.

4.2.2 File Structure

The special file structure is created or extended during the installation.

The zenon program files are copied to a folder which can be defined during the installation.

Additionally the installation asks for a folder for the SQL databases of the projects. The storage medium for project archiving (SQL, screens etc.) must have enough free space, because all current and future project data is stored there.

Folder	Path
Program folder	32-bit system: *%Program Files%\COPA-DATA\zenon7xxx* 64-bit system:

Folder	Path
	%Program Files%\COPA-DATA\zenon7xxx *%Program Files (x86)%\COPA-DATA\zenon7xxx*
Program data folder, e.g. global symbols, print templates, log files etc.	*%ProgramData%\COPA-DATA\zenon7xxx*
Database folder (SQL)	*%ProgramData%\COPA-DATA\SQL*
System folder	*%ProgramData%\COPA-DATA\System*
Settings Editor and profiles	*%Users%\UserName\AppData\Local\COPA-DATA\zenon\Editor*
Setting for Diagnosis Viewer.	*%Users%\UserName\AppData\Local\COPA-DATA\zenon\DiagView*

DEFINITION RUNTIME FOLDER AND DATA FOLDER

RUNTIME FOLDER

The Editor creates the Runtime files in the Runtime folder, or they are transferred to this folder by means of Remote Transport. The Runtime folder is created or updated when compiling a project in the Editor. The parameters for this folder can be set in the zenon Editor with the **Runtime folder** project property. With remote transfer, the Runtime folder is defined in Remote Transport settings.

DATA FOLDER

The Runtime saves all data files that were created in Runtime such as alarm files, archive files etc. in the data folder. The data folder is created as a subfolder of the Runtime folder by default. The folder is automatically assigned the name of the computer the Runtime is running on. You can change this save location in the project properties (**General**/**Data folder**).

Hint: Never set the data folder to a removable device such as an USB stick or a network device. It is recommended that the data is recorded locally and backed up externally.

> **Attention**
>
> If the defined path does not exist or is not available, no more data is written from the Runtime. This means a complete loss of data. The Runtime can still be operated but must be restarted as soon as the path is available again. The availability of the folder can be checked via system driver variable **Runtime folder not available** .

4.2.3 Free ports

zenon and zenon Logic need certain communication ports for the communication in the network. If these ports are occupied by other programs like e.g. an already installed SQL server, communication from zenon can be disturbed. Many ports in zenon can be changed using the **Startup Tool** or properties in the Editor.

This is how you check the port assignments:

1. Enter *netstat -a -n -o* in the command line.

   You can reach the command line in Windows:

   - by pressing the Windows-key and R
   - Enter *cmd* and confirm with **OK**.
   - A DOS-window pops up
   - enter the command *netstat*

2. A list of all currently used TCP and UDP ports will pop up.
3. Check the listening ports (status: *LISTEN*) if the process-ID (PID) of the ports needed by zenon and zenon Logic corresponds with the processes of zenon and zenon Logic.

   These PIDs can be read in the **Windows Task Manager**. To do this, open the **Windows Task Manager** and switch to the **Services** tab.

4. If another software uses these ports, reconfigure this software.
   You can see the ports that zenon and zenon Logic use in the **Port assignment by zenon and zenon Logic** table. Here you can also see if these ports can be amended in these programs.

PORT SETTINGS BY ZENON AND ZENON LOGIC

Application	Description	Ports	Transport log
stratonrt[k].exe	zenon Logic Runtime polling communication and zenon Logic Workbench.	*1200-1210*	**TCP**
stratonrt[k].exe		*4500-4510*	**TCP**
stratonrt[k].exe	zenon Logic redundancy.	*7000-7010*	**TCP**
stratonrt[k].exe	zenon Logic Runtime spontaneous	*9000-9010*	**TCP**

Application	Description	Ports	Transport log
	communication.		
zennetsrv.exe	zenon network service.	▸ *1100-1100*	**TCP**
zensyssrv.exe	zenon transport service.	▸ *1101*	**TCP**
zendbsrv.exe	zenon database service.	▸ *1103*	**TCP**
zenAdminsrv.exe	zenon administration service.	▸ *50777*	**TCP**
zenLogSrv.exe	zenon logging service.	▸ *50780*	**TCP**
CodeMeter.exe	Code Meter dongle service.	▸ *22350* (changeable but must not be changed)	**TCP**
WkSvW32.exe	WibuKey Network service	▸ *22347* (fixed)	**TCP**
Zenrt32.exe	**Message Control** with Voice over IP.	▸ *5060*: SIP ▸ *4000*: RTP ▸ *4001*: RTCP (fixed) SIP and RTP can be configured using the Editor. RTCP is automatically set by the system.	**UDP**

4.2.4 Exceptions for anti-virus

However zenon needs a range of services and operations that could categorize anti-virus programs as potentially dangerous.

Examples of this:

- Communication with CodeMeter --> USB dongle
- Loading VBA or VSTA code
- Embedding ActiveX elements in screens
- Frequent file access, for example for archiving, CEL, AML...

Note: There are compatibility problems with *Ahnlabs V3* virus scanner. If zenon refuses to start with an error message, put V3 into *Game-Mode* or uninstall V3.

4.2.5 firewall setup

zenon uses a number of ports. These must be activated in security measures, such as a firewall, and accessible during operation.

PORTS USED

For communication within zenon, only TCP ports are used; no UDP ports are used. zenon requires the following ports in a network:

Service	File	Exercise	TCP-port
Network service	**zenNetSrv.exe**	Runtime communication.	*1100*
Transport service	**zenSysSrv.exe**	Data transfer by means of Remote Transport (Editor).	*1101*
zenon Web Server	**zenWebSrv.exe**	On-site logging machine between zenon Web client and Runtime	*1102*

Port numbers can be amended individually via the **Listening ports** tab in the **Startup Tool**. In this case, the measuring range must be adapted manually!

Furthermore, zenon and zenon Analyzer services use a range of default ports:

STANDARD PORTS

ZENON

Application	Standard port
Network Service	*1100*
Transport Service	*1101*
WEB Service Classic	*1102*
DB Service	*1103*
SQL Browser Service, (for distributed engineering in the Editor)	*1434*
zenAdminSrv.exe	*50777*
zenLicTransfer (License Transfer Service)	*50784*

Application	Standard port
Logging Service	*50780*
SNMP Trap Service	*50782*
WEB Service Tunneling	*8080*

ZENON LOGIC

Application	Standard port
Assigned port for zenon Logic or straton depends on the project and service. E.g.: First zenon Logic project used *1200* and *9000*, second project *1201* and *9001* etc.	*1200 - 1210* *4500 - 4510* *7000 - 7010* *9000 - 9010*

ZENON ANALYZER

Application	Standard port
Administration Service	*50777*
Analyzer Connector Service	*50778*
Analyzer License Service	*50779*
ZAMS	*50781*

DRIVERS

Application	Standard port
Driver Simulation	*6000 - 6020*
Process Gateway OPC Server	*135*
Process Gateway SNMP	*161*
Process Gateway Modbus	*502*
Process Gateway IEC60870-5 104 slave	*2402*
Process Gateway DEC	*5555*
Process Gateway DNP3 Slave	*20000*

SERVICE GRID

Application	Standard port
Service Grid API	*9400*
Hub Controller	*9410*
Data Hub	*9411*
Hub Controller: Dedicated port for connection to **Data Hub**	*9412*
Configuration Backend	*9420*
Identity Service	*9430*
Policy Service	*9440*

Note: zenon drivers that communicate by means of Ethernet use TCP and thus might need authorizations in the firewall depending on the port used.

4.3 Hardware requirements

One of the requirements for putting Runtime into operation is that setup has been completed successfully. It is also important to have a valid license.

The minimum requirements are listed in the following table. These are based on a complete installation of Runtime. For extensive projects or several projects loaded at the same time you will need accordingly faster/stronger hardware. The minimum requirements can increase as a result of this.

Hardware	Minimum requirements	Recommended
CPU	Single core with SSE2 support.	Quad Core
RAM memory	Windows 7/8: from 512 MB. **Note:** Projects with big amounts of data, Network projects, multiple projects simultaneously and projects in redundance mode need more memory.	Windows 7/8: 4096 MB
Harddisk	2 GB free space for the Runtime-installation plus additional space for the projects. **Attention:** If you log historical data (e.g. Archive data or Alarm-/CEL-Data), you need sufficient harddrive space or you have to make sure during engineering that the	

Hardware	Minimum requirements	Recommended
	historical data is evacuated or deleted.	
Monitor resolution	VGA with 640 x 480 pixels.	
Graphics adapter	64 MB dedicated memory. Cards with shared memory can lead to performance loss. Note the System requirements when using DirectX (on page 9) chapter in relation to this.	
Input devices	Keyboard and/or mouse. Operation via touchscreen is also possible. Many individual, customizable soft keyboards for the touchscreen are available for you.	
USB interface **(optional)**	▸ For installation. Installation also possible via network or other storage media. ▸ For dongle. Network dongle also available.	
Network connection (optional)	64 kBits/s for standard Client/Server projects. 100 Mbit/s full duplex for redundant operation.	100 Mbits/s full duplex for standard Client/Server projects.
Remote connection (optional)	Minimum requirements: Dial-up modem with 9600 Bit/s.	1 Mbit/s full duplex.
WAN connection (optional)	Any desired connection via router, e.g. per ISDN or DSL Data transfer is slower in a WAN than in a local network for technical reasons. Be sure to check the possible data transfer rates of your WAN technology already at the time when you create the project.	
Message Control (optional):	Please refer to chapter Message Control for the requirements.	
Interfaces (optional)	The necessary interfaces depend on the requirements of the PLC and/or the bus connection, for example serial RS232 or RS422/485 interfaces, ISA/PCI slots, etc.	

4.3.1 Paths for setup and operation

Paths for zenon:

- Setup
- Runtime

> **Info**
>
> -
>
> You can display many default paths with the help of the **set** command:
>
> - start the command line (enter **cmd** in the Windows start area)
> - enter command **set**
> - Press the **Enter key** to display standard folders for Windows and zenon

Note: As absolute paths differ in different operating system, the paths are displayed as Windows environment variable in this chapter. For example *%ProgramData%* instead of *C:\ProgramData*.

SETUP

During setup, paths are set for required software from third-party providers, as well as the zenon Runtime folder.

The setup needs administrator rights. This is also true for changing the installation paths.

REQUIREMENTS

The installation paths of the required third-party software match the standard paths of the respective manufacturer and cannot be changed during setup.

The additional software packages that need to be installed depend on the type of installation:

- zenon Editor
- zenon Runtime
- zenon Web Client
- zenon Logic Runtime

Requirements	Editor	Runtime	Web Server	Web client	zenon Logic Runtime
Microsoft .NET Framework 4.6.2	+	+	-	+	-

Requirements	Editor	Runtime	Web Server	Web client	zenon Logic Runtime
Microsoft SQL Server 2017 Express (x64)	+	-	-	-	-
Microsoft Visual C++ 2010 Redistributable	+	+	-	+	+
Microsoft Visual C++ 2013 Redistributable	+	+	-	-	-
Microsoft Visual C++ 2017 Redistributable	+	+	+	+	+
Microsoft .NET Core Hosting Bundle 2.1.5	-	-	+	-	-
CodeMeter Runtime Kit 6.80	+	+	+	-	+
Microsoft Visual Studio Tools for Applications | VSTA 2.0	+	+	-	+	-
Microsoft Visual Studio 2008 Remote Debugger	+	+	-	-	-
Microsoft Web Deploy 3.6	-	-	+	-	-
Visual Basic for Applications | VBA 7.1	+	+	-	+	-
Visual Basic for Applications Language Pack | VBA 7.1	+	+	-	+	-
COPA-DATA Multiple Network Protocol Driver	+	+	-	-	+

ZENON RUNTIME

zenon Runtime uses the path that is set in zenon6.ini for projects. This path can be set using the Startup Tool as well as using Remote Transport from a zenon editor.

Object	Path
Project	*[user-defined path]\[project]\RT*
External files	*[User-defined path]\[project]\RT\FILES\...*
Exported archives, Chronological Event List and Alarm Message List	*[User-defined path]\[project]\Export* **Note:** Is created at the first export.
System files	Windows system folder.

ZENON LOGIC

Paths for zenon Logic are created analogous to the zenon paths.

4.3.2 Demo licenses and Demo mode

Each installation contains:

- Demo licenses: Allows you to test a product for a certain time period.
  Demo licenses follow the rules for licenses with a time quota.
- Demo mode: Allows you to start a product and use it for a certain period of time. It is closed again afterwards.
  Runtime Editor: 10 Minutes
  Runtime: 30 minutes

Each installation contains at least one demo license. This has a pre-defined duration or number of permitted starts. If these are used up, the product can continue to be used in demo mode, but it is ended after running for 10 minutes however. You can find out the usage period that is available in the **License Manager** in the **License usage** tab in the details of the license.

COPA-DATA products:

- zenon Editor with zenon Logic Workbench as a module
- zenon Runtime:
  Runtime for Demo licenses:
  - 43200 minutes (30 days) on hardware computers
  - 1440 minutes (1 Tag) for virtual machines
- zenon Logic Workbench and Runtime
- Process Gateway and OPC DA Server
- zenon Web Server and Web Client
- HTML Web Engine
- zenon Analyzer
- Hub Controller
- Service Grid API
- Identity Service

Demo licenses have their own software dongles. These cannot be used for other licenses. The last existing demo license cannot be deactivated using the **License Manager**.

The remaining duration and the number of the starts that are still possible for demo licenses and demo mode are shown in the start screen and in the version and license information.

MISSING DEMO LICENSE

If no demo license is available on your computer, then:

1. Open the **License Manager**.
2. Open the **License usage** tab.
3. Select the desired demo license.
4. Select the desired product.
5. Click on the button at the far right of the list.
   The button turns green in color and the license is now being used.

Note: If there is also no demo license available in **License Manager**, contact your COPA-DATA sales partner.

Hint

You can use demo licenses to test products and editions. To do this, place the demo license at the first location of the license list in the **License Manager**. This license is used the next time the product is started.

4.3.3 Runtime under Windows Embedded Standard

The minimum requirements relate to an installation of Runtime adapted to the Windows Embedded Standard 7 SP1 operating system with platform update. The hardware must be accordingly more powerful for extensive projects.

This table only states the figures that are different to the standard installation. The other parameters correspond to the figures described in the Hardware requirements for Runtime chapter.

Hardware	Minimum requirement	Recommended
Memory	▸ 512 MB. **Note:** Projects with big amounts of data, Network projects, multiple projects simultaneously and projects in redundance mode need more memory.	▸ 2048 MB
Storage medium	▸ 2 GB of free memory on C:\ drive before the installation of .NET Framework. 800 MB of free memory after the installation of the .NET Framework ▸ Plus memory space for the projects, archives, etc.	▸ 80 GB

Attention: If you log historical data (e.g. Archive data or Alarm-/CEL-Data), you need sufficient harddrive space or you have to make sure during engineering that the historical data is evacuated or deleted.

4.3.4 Runtime for Windows CE

The minimum requirements are based on a complete installation of the Runtime for Windows CE. The hardware must be accordingly more powerful for extensive projects.

Hardware	Minimum requirement	Recommended
CPU	At least 400 MHz	1 GHz
RAM memory	64 MB	1024 MB for Windows CE 6.0.
Storage medium	64 MB free harddrive space. Permanent recordable remanent storage medium for project data	256 MB free harddrive space or more.
Network connection	For standard Client/Server projects: 10 Mbit/s full duplex.	

Attention

Windows CE is no longer supported from version 7.50. zenon CE version 7.20 is installed. To use this, the Runtime files for version 7.20 must be created.

5 Uninstalling Runtime

To uninstall Runtime:

1. Open the **Control Panel**.
2. Click on **Programs**.
3. Click on **Programs and Features**.
4. Select the zenon 8.20 programs.
5. Click on **Uninstall**.
6. Follow the uninstall wizard.

6 Runtime for Windows CE

The Runtime installation for Windows CE consists of the following files:

File name	Description
zenonrCE.exe	The Runtime application
Cd_tooCE.dll	Necessary for the Runtime
ZennetsrvCE.dll	The control system netservice for Windows CE. Necessary for network projects.
SysSrvCE.exe	Transport service
LogCliLibCE.dll	The logserver client
zenon6.ini	Text file with settings for the Runtime like e.g. start project, language of the Runtime, etc.
UpdateCE.exe	Application, necessary for the CE Update tool.
RgermaCE.dll	German language file
RengliCE.dll	English language file
RfrancCE.dll	French language file
RitaliCE.dll	Italian language file
RrussiCE.dll	Russian language file
RspaniCE.dll	Spanish language file

Install the Runtime to the CE device with the entry CE Runtime Update-Programs (under menu **Option**).

Attention

Windows CE is no longer supported from version 7.50. zenon CE version 7.20 is installed. To use this, the Runtime files for version 7.20 must be created.

Info

The following is true for a connection to Windows CE 6.0 with **CX1000**, **Profibus** or **SYCONuni**: Make sure that the file **CDMemDrv.dll** has been transferred to the device. **CDMemDrv.dll** is a Windows CE device driver. The DLL is available for x86 and for ARMV4I.

Additionally there are a number of other drivers for hardware communication. If the processor type of the CE device is recognized by the editor, the Remote Transport automatically transports the drivers used in the project to the CE device. The manner or the number of transferred files is displayed in the Output Window of the Editor.

These files always have to be compatible with the according CE version and with the according processor type. Mixing files from different CE versions or even service packs can lead to failures and unwanted side effects, and is thus is not permitted.

Info

When starting **zenonrce.exe**, the file **syssrvce.exe** is also executed. Thus it is guaranteed that a TCP connection can also be established. Long delays are avoided and only one file is necessary for the autostart functionality.

6.1 System files

The Windows CE Runtime requires the existence of certain system files. In case one of these files is missing, the operating system sends an error message during Runtime start, that one or various components have not been found. The following system files are required:

File	Description
mfc90u.dll	Necessary for the Runtime. On startup, an error message pops up if this file does not exist.
msvcr90.dll	Necessary for the Runtime. On startup, an error message pops up if this file does not exist.
atl90.dll	Not necessary for starting the Runtime but for the use of drivers with network connections or the use of zenon in a network. If this file does not exist, the device will not work as a client or TCP/IP driver connections will not work.
IMGDECMP.dll	Not necessary for starting the Runtime but necessary for displaying Transparency if Alpha Blending is not integrated in the operating system. Animation of GIF files is not possible with Windows CE.
VBSCRIPT.dll + JSCRIPT.dll	Not necessary for starting the Runtime. This file is needed for the PCE (Process Control Engine).

Some of these system files are installed together with the installation of zenon for CE and can be transferred to the CE device using the **UpdateCE** Tool. All these system files should be integrated in the operating system image of the CE device by the manufacturer.

Attention

For manufactures of Windows CE OS-images:
CE versions older than 6.0 need the file **toolhelp.dll**. Activate the following option in **Platform Manager** in order for the file to be available on the CE device and for Toolhelp to also be available in SDK:
Core OS -> Display Based Device ->Core OS Services -> Debugging Tools -> *Toolhelp API*. Thus the *Toolhelp.dll* is part of the image.
Hint: Always use the most up-to-date Servicepack of the **Platform Builder**.

Note: The **toolhelp.dll** is not required for Windows CE 6 and should not be used with CE 6.

6.2 Update of the Windows CE Runtime

To perform an update of Windows CE Runtime:

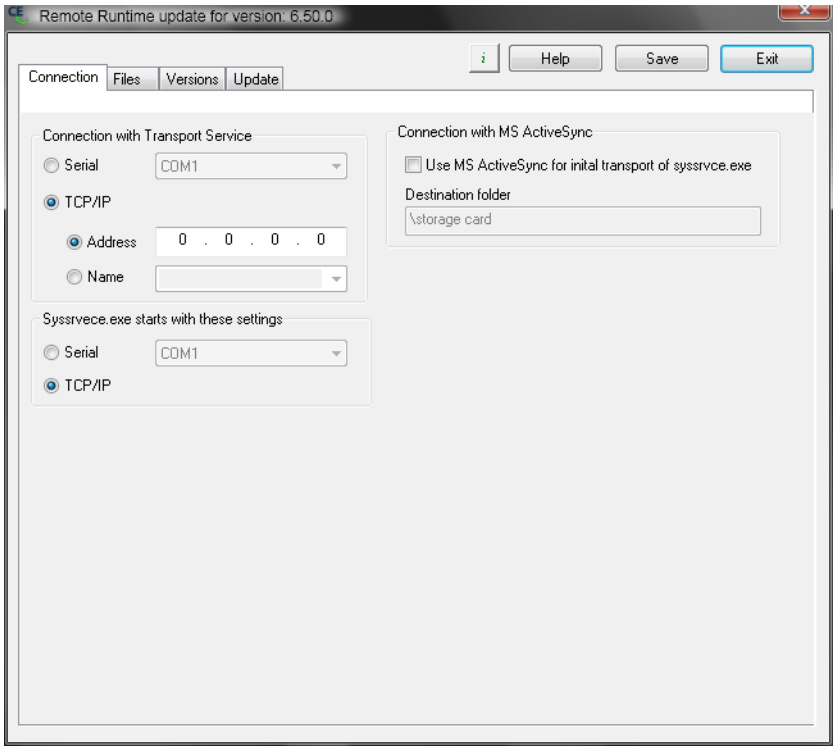
- Make sure that the zenon Transport Service (**SysSrvCE.exe**) runs in the CE device
- Make sure that you do not have established a remote connection via the zenon Editor to the device
- In the zenon **Extras** menu, select the **Update Windows CE Runtime** command.

  The dialog for transfer of Runtime files opens.
- Configure the connection.
- Define the data you want to transfer.
- Select the appropriate version.
- Start the update.

Note: If you are transferring/installing Runtime for the first time, note the information in the **Manual installation and Runtime update** (on page 32) chapter.

CONFIGURE CONNECTION

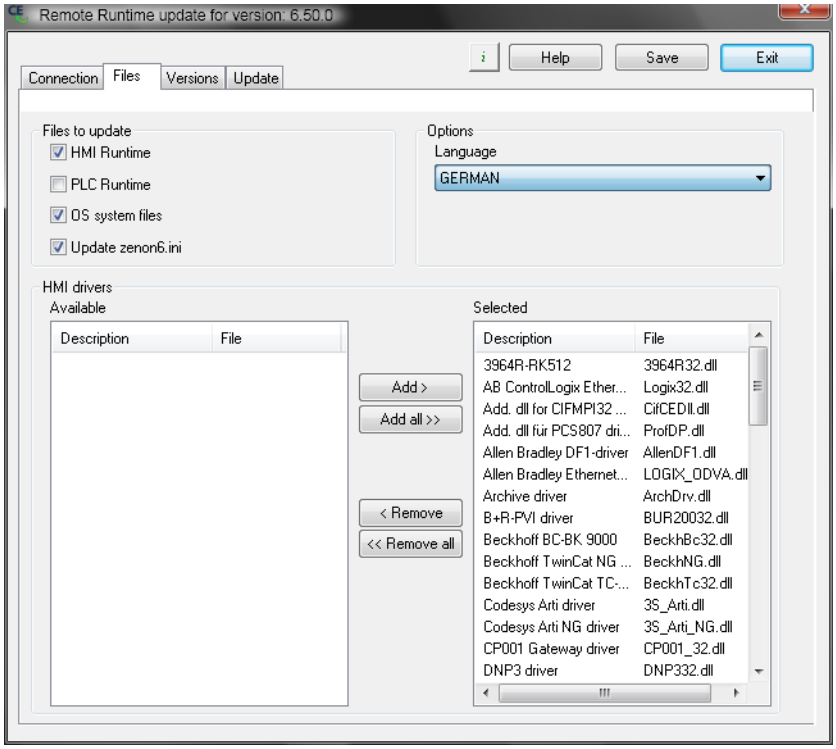
You can configure the connections to the Windows CE device in the tab **Connection**.



Parameter	Description
Serial	Settings for serial connection with Windows CE device, you have to select a port.
TCP/IP	Settings for TCP/IP-connection to the Windows CE device.
Adress	IP address.
Name	Computer name
Syssrvce.exe starts with these settings	Settings for starting **syssrvce.exe**.
Serial	*Active*:: serial connection selected, port must be selected.
TCP/IP	*Active*: TCP/IP-connection selected.
Connection with MS ActiveSync	Settings for connection via MS ActiveSync
Use MS ActiveSync for initial transport of syssrvce.exe	Active: syssrvce.exe is transferred during the first transport via MS ActiveSync.
Destination folder	Target folder.

Parameter	Description
Help	Opens online-help
Save	Saves all changes.
Exit	Closes the update CE-tool and reminds you before to save unsaved changes.

DEFINE FILES YOU WANT TO TRANSFER

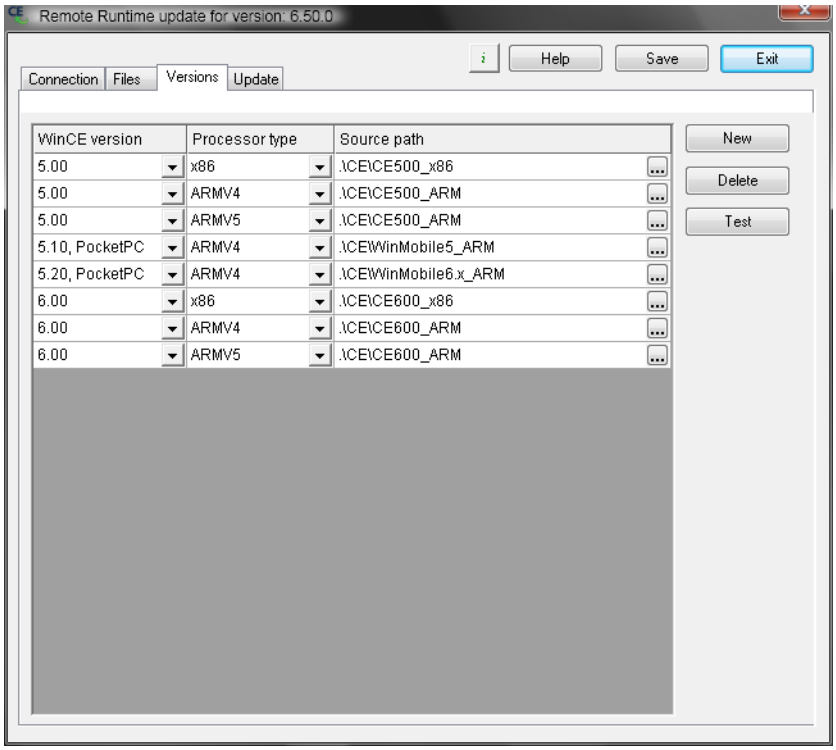


Parameter	Description
Files to update	Files to be transferred.
HMI Runtime	*Active*: Transfers zenon files to the target device. Default: *active*
PLC Runtime	*Active*: Transfers zenon Logic files to the target device. Default: *inactive*
OS system files	*Active*: Transfers necessary files for the OS. Default: *active*

Parameter	Description
Update zenon6.ini	Transfers zenon6.ini to the target device. This way, the license information of the target device is also changed.
Options	
Language	Desired target system language. Default: *English*
HMI drivers	Selection of HMI drivers for transfer.
Available	List of available drivers.
Selected	List of selected drivers.
Add	Adds chosen drivers to the list of selected drivers.
Add all	Adds all drivers to the list of selected drivers.
Remove	Removes chosen drivers from the list of selected drivers.
Remove all	Removes all drivers from the list of selected drivers.
Help	Opens online-help
Save	Saves all changes.
Exit	Closes the update CE-tool and reminds you before to save unsaved changes.

SELECT VERSION

Select the correct version in the tab **Versions** if it wasn´t automatically recognized.

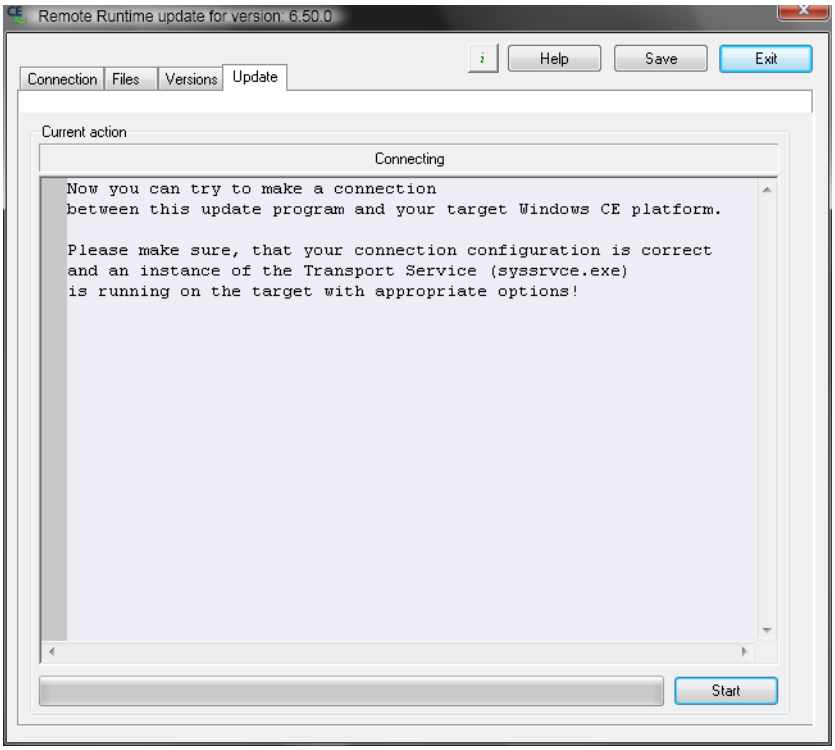


Parameter	Description
WinCE version	Version of the target device Windows CE OS. Click the button to open a drop-down list for selection.
Processor type	Processor of the device.
Source path	Path to the folder that contains the files. Click the button … and a dialog opens to select a folder.
New	Inserts a new, empty entry in the list.
Delete	Deletes the selected entry from the list
Test	Verifies settings.
Help	Opens online-help
Save	Saves all changes.
Exit	Closes the update CE-tool and reminds you before to save unsaved changes.

START UPDATE

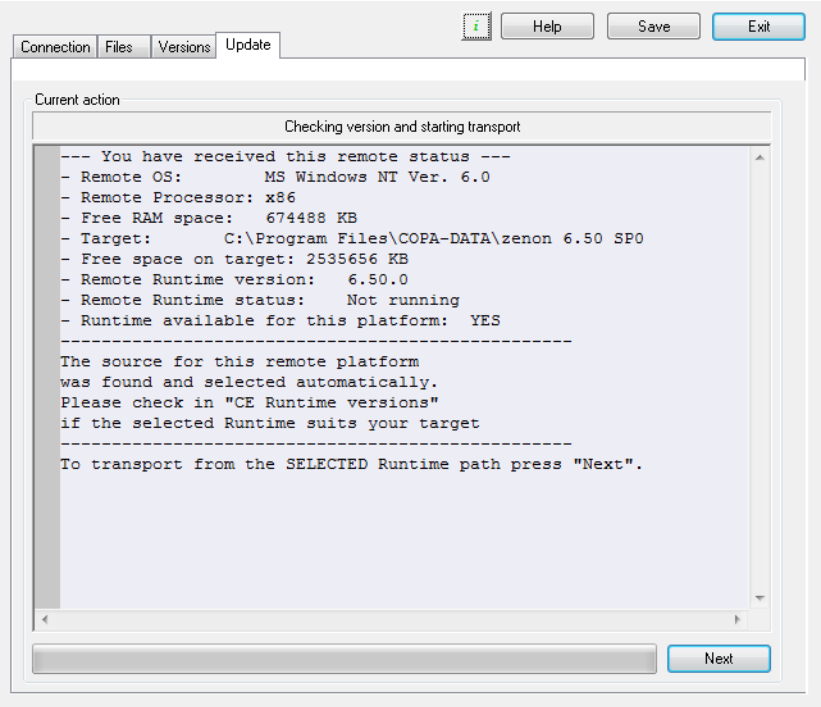
To establish a connection:

1. Click on the button **Start** on the tab **Update**.

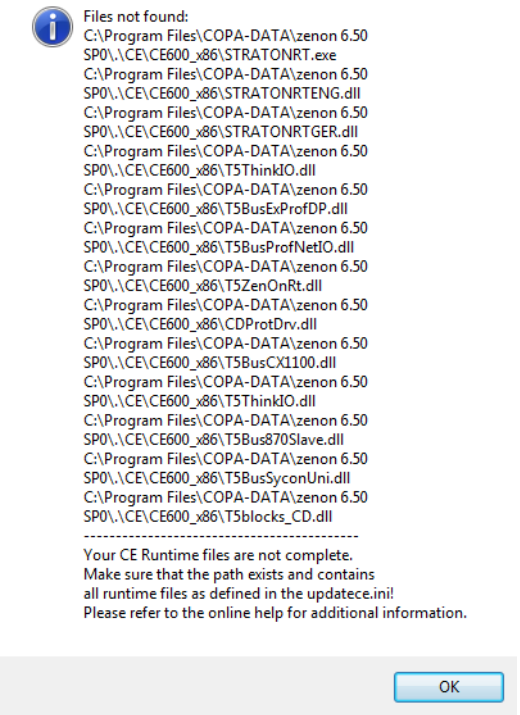


The data that shall be transferred is verified and displayed in a window.

2. Start the transfer to the target device by clicking on the button **Next**.



If the transfer cannot be initiated because files are missing, an error message with a list of missing files pops up:



If you get the error message **The current update was not completed**, the update was interrupted or not executed properly.

6.3 Manual installation and Runtime-update

Installation and update are also possible without remote transport and ActiveSync. You have to copy the needed files manually on a storage card for the CE device. You have to know CE version and processor type.

It is mandatory to copy the following files from the according folder for the correct platform to a storage card for the CE device:

- **zenonRCE.exe** (Runtime)
- **LogCliLibCE.dll** (Diagnose-DLL)
- **zenon6.ini** (Configuration file)
- **Cd_tooCE.dll** (Help DLL)
- **ZenNetSrvCE.dll** (Network)
- **CDHelper.dll** (Help DLL)
- One of the following language DLLs: **RChineCE.dll, RCzechCE.dll, REngliCE.dll, RFrancCE.dll, RGermaCE.dll, RItaliCE.dll, RRussiCE.dll, RSpaniCE.dll**(the selected language is stipulated in the **zenon6.ini** file.)
- **syssrvce.exe** (Transport service and Diagnosis Server)
- **atl90.dll** (System file, possibly part of the operating system)
- **mfc90u.dll** (System file, possibly part of the operating system)
- **msvcr90.dll** (System file, possibly part of the operating system)

Further files, such as drivers, are optional.

7 Starting the Runtime

There are several possibilities for starting zenon Logic Runtime:

- Using the Editor
  - by pressing the **F5** key
  - via symbol **Start Runtime** in the toolbarRuntime files
- via the **Startup-Tool**
- via the Windows start menu
- using Windows Autostart.

- via the service **zenStartupMgr** (on page 40)

DIENST ZENADMINSRV

The **zenAdminSrv** service must be started to start the Runtime. If the service is not available, you will receive an error message accordingly.

When starting via Windows Autostart, zenon attempts to start Runtime until the service is available or you end the error message by clicking on the **Cancel** button.

SIZE OF THE MAIN WINDOW

The size of the Runtime window when Runtime is started is configured for the local computer in the zenon Editor.
Carry out the following steps for this configuration:

1. Select the **Workspace** in the zenon Editor.
2. Click on the **Local Runtime size** property group.
3. Enter the values for **Runtime window width [pixel]**.
4. Enter the values for **Runtime window height [pixel]**.

FUNCTION "WINDOW TO FOREGROUND"

The Runtime can be moved behind other running programs with the help of function **Window to the background**. With function **Window to foreground** it is moved to the foreground.

Information

When switching Runtime to the foreground with the **Window to foreground** function, the Runtime window is defined as the upper window.

Note the following special cases:

- The **DIALOGKBD** keyboard cannot be called up in this situation. Other keyboards can be used.
- The alarm status line is covered by the Runtime window.

You have the following options to get the alarm status line back to the foreground:

- Activate the symbol keys (deactivate **Lock system keys** project property) and get the alarm status line back to the foreground with **Alt+Tab**.
- Activate the Windows task bar and click on the **Status** window.
- Move Runtime to the background again.
- Restart the Runtime.

Note: The **Window to foreground** function is not identical to the **Move frame to foreground** function!

SET START PROJECT

If you have created a project, you can start this in different ways.

Note: It is a requirement that the project generated is first transferred to the target system if the configuration computer is not also the target system.

There are several possibilities for starting a project in Runtime:

Using the Editor:

1. Start the zenon Editor.
2. Select the project that you want to set as the start project.
3. Click on *Set project as start project* in the context menu

Via the Startup Tool:

1. Open the **Startup Tool**.
2. Click on **Item--> Properties...**
3. Select, in the **RT project** field, the project that you want to set as the start project.

Via Remote Transport:

1. Activate the project in in the toolbar.
2. Select the desired project as the start project.

Via the network topology:

1. Add a computer.
2. Select this computer.
3. Select the desired project as the start project.

7.1 Set up Runtime autostart

To set up Runtime autostart, proceed as follows:

1. Select the *zenrt32.exe* application from the *C:\Program Files (x86)\COPA-DATA\zenon 8.20 SP0* directory.
2. Create a local link for this.
3. Copy this into the *Autostart folder*.

Runtime thus starts automatically the next time you log on.

7.2 Starting Runtime via the command line

You can also start Runtime from the command line and define a project as a startup project at the same time.

If the zenon Runtime is configured as a Terminal Server, the Runtime can be started several times with different start projects.

The general syntax is as follows:

C:\Program Files (x86)\COPA-DATA\zenon 8.20 SP0\zenrt32.exe /Project="[project path];[project name]"

Example: *C:\Program Files (x86)\COPA-DATA\zenon 8.20 SP0\zenrt32.exe /Project="C:\Temp\Project1;Project1"*

Parameter	Description
Project path	The project path points at any project that should be performed in Runtime. **Note:** The path must point to a valid zenon project folder containing a folder with the name **RT**. Otherwise, an error message is displayed when Runtime starts.

Parameter	Description
Project name	The desired project name under which the project should be started. The name can be assigned freely. This is necessary because the project name is not part of the project data.

Note: The use of the argument */Project* has a higher priority than the entries **VBF30** and **DEFANWENDUNG30** in the **zenon6.ini**file.

Note: If **zenrt32.exe** is called with the argument */Project=""*, the **Runtime Server** dialog appears when starting Runtime to select a startup project.

7.3 Keyblock Runtime Start

Keyblock Runtime Start is a program with which zenon Runtime runs as a **Shell**. In doing so, zenon Runtime is started, but all **Windows** system tasks are blocked. Keyboard shortcuts such as the **Windows key + L** or **Ctrl+Alt+Del** no longer have an effect.

Note: **Ctrl**+F4 cannot be blocked. To lock all key combinations with the Windows key: Deactivate the **Windows key** in the **Startup Tool** under **Application -> Options -> General**.

You can find details in the **Use** (on page 36) chapter. User can no longer access the operating system but only work on the zenon user interface. Limitations (on page 58) from the **Lock system keys** project property are thus enhanced.

The precondition for this is that the project properties are set **Runtime title** to *No title (full screen)*. Then zenon runs in full screen mode and the Runtime cannot be minimized.

This locking cannot be bypassed during Runtime. When the Runtime is closed normally, the system restrictions are canceled. If the Runtime is to be operable without these limitations, Runtime must be started without the **Keyblock Runtime Start**.

Note also the information provided in the **Protect Runtime files** (on page 38) chapter**.**

7.3.1 Use

To use **Keyblock Runtime Start**:

1. In the Windows start folder, under COPA-DATA, open the zenon **Tools**.
2. Select **Keyblock Runtime Start**.
3. The program is opened and automatically starts Runtime.
4. The program blocks all access to the operating system.

BLOCKED FUNCTIONS

Locked keyboard shortcuts:

- **Ctrl+Alt+Del**
- **Ctrl+Esc**
- **Alt+Tab**
- **Alt+Esc**
- **Alt+F4**
- **Windows+L** and the functionality of the Windows key itself
  To block all key combinations with the Windows key: Deactivate the **Windows key** in the **Startup Tool** under **Application -> Options -> General**.

Note: **Ctrl**+F4 cannot be blocked. Windows keys

System locks:

- Hiding of the control panel in the start menu.
- Locking of the toolbar for operation.
- Prevents:
  - Changing passwords
  - Closing Windows
  - Logout
  - Locking the computer
  - User change
- Hiding of all elements in the task manager.

Notes:

- When locking the system keys, normal operation of the scroll bars with the mouse in the Runtime is also blocked. You can work around the blocking with the help of the context menu.
- To block all Windows keys: Deactivate the **Windows key** in the **Startup Tool** under **Application -> Options -> General**.

> **Attention**
>
> Take care that you engineer a possibility to close the Runtime in your project. There is no possibility to end the Runtime regularly.
>
> - It can only be ended by shutting the computer down using the hardware
> - All system keys also remain blocked after restarting
>
> In order to make system keys accessible again after not being shut down properly (in the event of a power cut for example):
>
> - start the Runtime again with the help of **Keyblock Runtime Start**
> - end the Runtime regularly via a close button

NOTES ON AUTOSTART

If **Keyblock Runtime Start** is started using the operating system's startup process, then note the following:

- The Autostart folder is user specific:
  If another user logs in, the program is not executed.
- Execution of the Autostart programs can be prevented by pressing the **Shift** key when the operating system is booting.

7.3.2 Protect Runtime files

The access to the Runtime files can be strongly restricted and therefore well protected. At this only a single Windows user has read and write rights for the Runtime folder. All other Windows user do not have any rights in the Runtime folder including read rights. Operators in the Runtime log on as zenon users.

In order to limit access to the file system:

1. Only create a single Windows user (for instance: **zenon_ADMIN)** who is authorized to start zenon as well as to read and write in the zenon Runtime folder.
2. Disable access to the zenon Runtime folder for all other Windows users – including read authorizations!
3. Disable any remote access to the user **zenon_ADMIN**.
4. Block any software for remote maintenance or remote access.
5. Make sure that zenon can only be started if this user (**zenon_ADMIN)** is logged in.
   Since other Windows users do not have read authorization Runtime will only start in the context of this user (**zenon_ADMIN)**.
6. Make sure that zenon runs as shell:

a) For this purpose, create a zenon autostart with **Keyblock Runtime Start** (on page 36)

b) Activate the property **Lock system keys** in the group **Runtime settings** of project properties.

c) Start zenon in full-screen mode: Set property **Runtime title** to *No title*.

d) Ensure that you also take multi-monitor systems into account during configuration.

e) Disable Explorer start

f) Do not offer file selection dialogs.
Note: In this case, no functions that require the user to select files in the Runtime can be configured.

Access to the zenon file system is thus restricted.

7.4 Configure start delay

When using dongles for licensing, the Codemeter service must be ready before zenon Runtime starts. To amend the start time, you can configure a start delay for zenon Runtime.

Note: In doing so, note that this start delay can only be configured in the zenon6.ini file. There is no possibility to set the parameters of this in a graphical user interface.

ZENON RUNTIME START DELAY

Stipulate a start delay of at least 30 seconds in the **zenon6.ini** file for zenon.

To do this:

1. Open the **zenon6.ini** file with a text editor.
2. Navigate to the **[DEFAULT]** area
3. Create a new **STARTDELAY=** entry
4. Enter the desired delay in milliseconds.

Example:

[DEFAULT]

STARTDELAY= 30000

Information

You can find**zenon6.ini** in the following path:
%ProgramData%\COPA-DATA\System

8 Starting Runtime as a service

The zenon Runtime can automatically be started as a service by the **zenStartupMgr** service.

Attention: If Runtime is started using the **zenStartupMgr**, it can no longer be stopped or restarted by users.

To start Runtime as a service:

1. Register the file **zenStartupMgr.exe** as a service.
2. Configure the properties for sign-in.
3. Start, if necessary, Remote Transport (on page 44) with **zenStartupMgr.**
4. Define the Runtime to be started in the **Startup Tool** (on page 42).
5. Configure a start delay (on page 39) for zenon Runtime if you are using a dongle license.

REGISTER SERVICE

To register **zenStartupMgr.exe** as a service:

1. Open the command line.
2. You can navigate to the location of the **zenStartupMgr.exe** file.
   - For native 32-bit OS: *%ProgramFiles%\Common Files\COPA-DATA\zenStartupMgr*
   - For 64-bit OS:

     64-bit folder: *%ProgramFiles%\Common Files\COPA-DATA\zenStartupMgr*

     32-bit folder: *%ProgramFiles(x86)%\Common Files\COPA-DATA\zenStartupMgr*
3. Register the file as a service with the **zenStartupMgr.exe –service** command.
   Note: The service, if it is already running, is first stopped and then registered. If **zenStartupMgr** is running, it is closed.

CONFIGURE SERVICE FOR THE USE OF MANY DRIVERS

Windows as an operating system limits the number of windows that can be created due to its fixed, reserved desktop memory, depending on the version and possible interaction with the desktop.

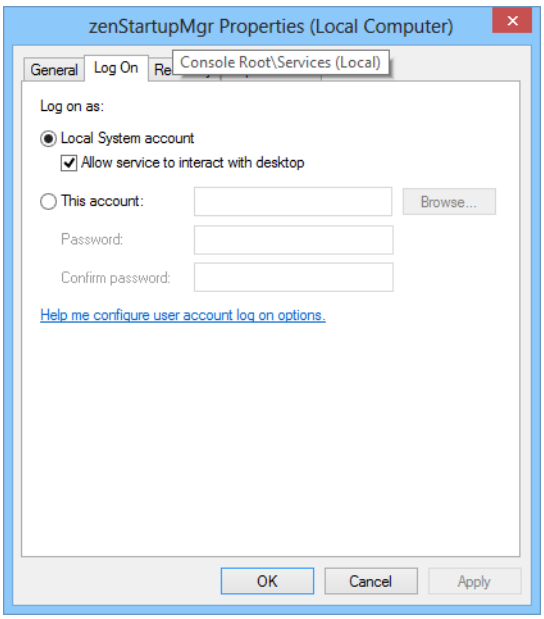
Version/action	Interactive Desktop	Non-Interactive Desktop
Windows 7 32-Bit	12 MB	512 KB
Windows 7/Windows Server 2008 R2 64-bit	20 MB	768 KB

zenon drivers each need several windows. The number of drivers that can be used can be influenced using the **Allow data exchange between service and desktop** option in the properties of the service.

- *Inactive*:
  A maximum of 20 drivers can be started.
- *Active*:
  As many drivers as there are in a Runtime started as a user process can be started.

To activate the **Allow data exchange between service and desktop** option:

1. Open the Windows Service Manager.
2. Open the properties of the **zenStartupMgr** service.
3. Go to the **Login** tab.



4. Activate the **Allow data exchange between service and desktop**.
   This service must be configured to *automatic* start type. With the service active, the user logged on to the computer is notified if the zenon Runtime as a service opens an additional window, for example in the event of a new alarm and active status line.

   Note Windows 8/Server 2012: In order for the service to be able to be started, the entry must be set correctly in the Windows registry:

   a) Go to the entry *HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Windows*.

   b) Open or create the DWORD value **NoInteractiveServices**.

   c) Set the decimal value of *1* to *0*.

5. To stop Runtime messages being displayed on the desktop:
   Deactivate the **Detection of inactive services** service.

INTERACTIVE ACCESS

If it is possible to interactively access the process, then:

1. Activate the **Allow data exchange between service and desktop** option in the service**.**
2. Activate the **Detection of interactive services** service.

Note: This type of access is not suitable for use for the actual operation of Runtime. Services are intended as program instances that run in the background that are not operated by means of a user interface. If operations on the Runtime Server are to be carried out, Runtime must be started as an application. If Runtime is to run on the Server as a service, operation of Runtime can be carried out from a zenon Client.

LIMITATIONS TO INTERACTIVE ACCESS

If zenon Runtime is started as a service and this is nevertheless to have a graphic user interface (GUI), this is implemented by the operating system by means of the *Secure Desktop* (detection of interactive services) system service.

Access to the Runtime user interface via the Windows system service has been subject to several restrictions since the Vista/Server 2008 version for security reasons. Above all:

- The interactive desktop is automatically ended after one minute of inactivity and the windows login screen is displayed.
- Display of DirectX effects, such as glow or shadowing for elements is not possible.
- Multi-monitor operation is not possible.

8.1 Configuration of zenon Logic Runtime

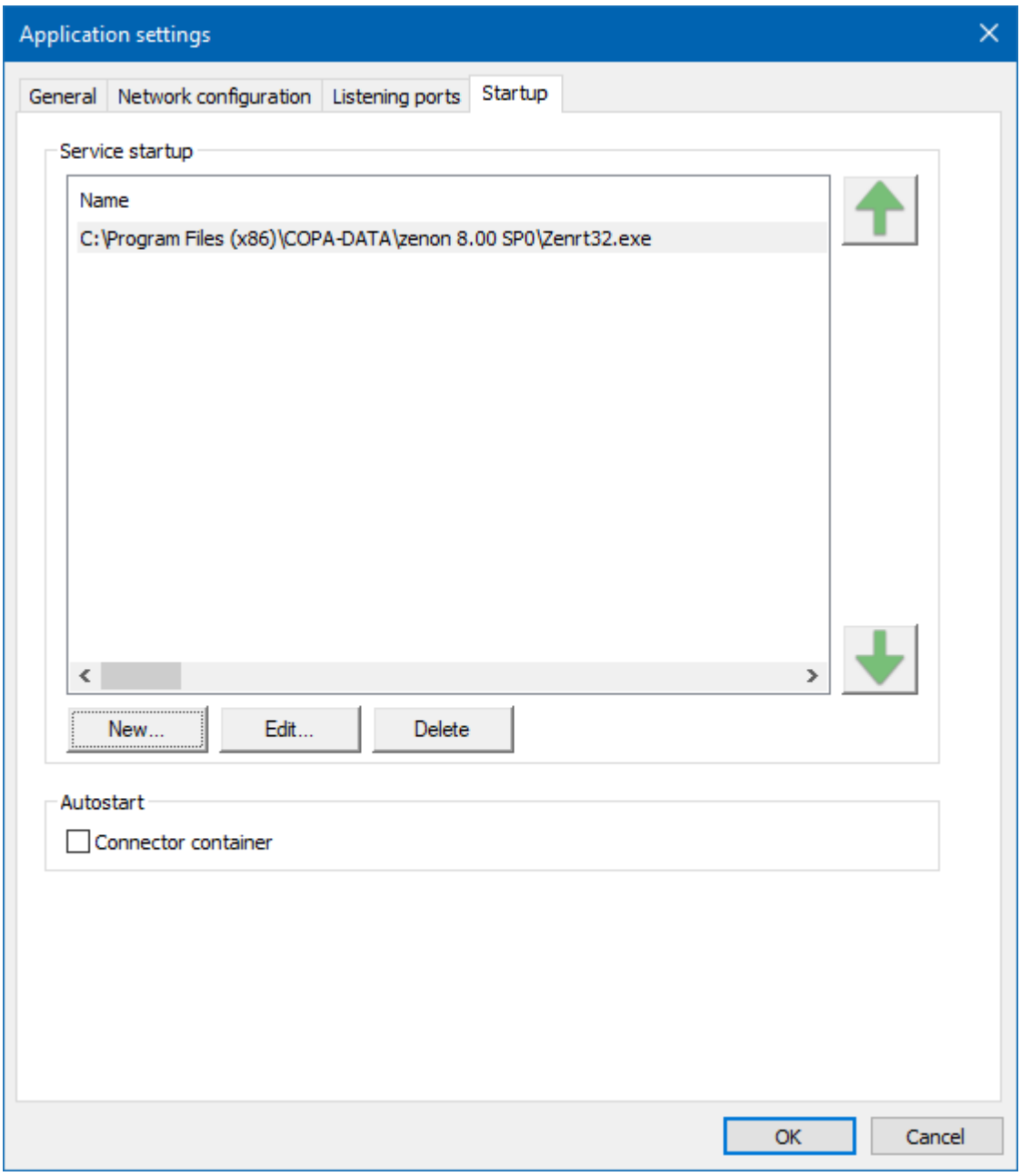
If zenon is running as a service, the user interface of the integrated zenon Logic Runtime is not available. To make changes to the configuration of zenon Logic Runtime, Runtime must be started normally.

8.2 Configuration in the Startup Tool

You can define how several programs are to be started by the service in the **Startup Tool**. To start an existing zenon Runtime as a service:

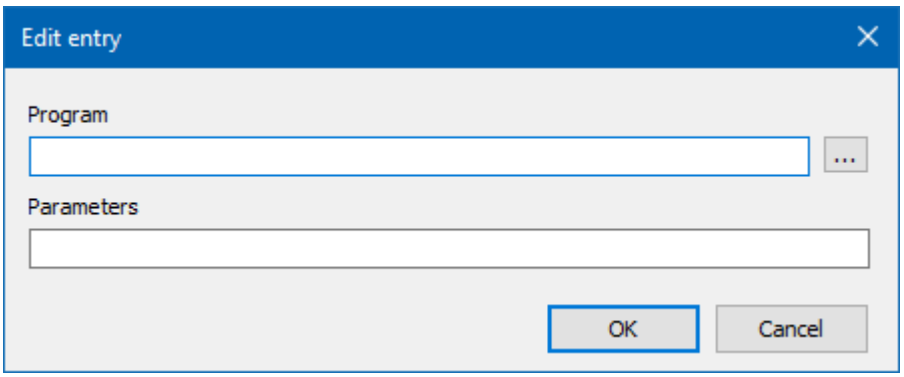
1. Open the zenon **Startup Tool**.
2. Click **Application**.
3. Select **Options**.
4. The dialog with the settings is opened.

5. Go to the **Service startup** tab.



6. To define a Runtime, click on the **New** button

7. The dialog for selecting a program is opened.



8. Click on the **...** button to open the file selection dialog
9. Go to the corresponding zenon folder
10. Select **Zenrt32.exe**.
11. Close the selection dialog and the **Startup Tool**.

8.3 Remote Transport

In its standard settings, zenon Remote Transport runs in the context of a logged-in user. In this configuration, it can neither start Runtime as a service nor reload or stop it.

Remote Transport must therefore be started with **zenStartupMgr**. To do this:

1. Create an entry for the start of **zenSysSrv** in the **Startup Tool -> Service startup**.
2. Place the entry before the entry for **zenrt32.exe**.
3. Deactivate **zenSysSrv** for logged-in users.

The Remote Transport can:

- Transfer data to the server
- Reload data

If Runtime runs as a service, it cannot be stopped or started by means of Remote Transport.

DEACTIVATE REMOTE TRANSPORT FOR LOGGED-IN USERS

If **zenSysSrv** is started with **zenStartupMgr**, it can no longer be started by logged-in users. To avoid this, remove it from the registry manually.

zenSysSrv Up to Windows 8:

1. Open the Windows dialog to execute commands (**Windows key + R**).
2. Enter **msconfig**.

3. The **system configuration** dialog is opened.
4. Go to the **Start** tab.
5. Deactivate the automatic start of zenon by deactivating the checkbox in front of **zenSysSrv.exe**.
6. Click on **Accept**.
7. Close the dialog by clicking on **OK**.
8. Restart the computer

zenSysSrv from Windows 8:

1. Open the Task Manager (**Ctrl + Shift + Esc** keys).
2. Go to the **Autostart** tab.
3. Deactivate the automatic start of zenon by deactivating the checkbox in front of **zenSysSrv - Transport service**.
4. Click on **Deactivate**.
5. Restart the computer

Attention: If a zenon version is re-registered via the **Startup Tool**, the entry is recreated and must be removed manually again.

8.4 Procedure

If the **zenStartupMrg** service is started, then it starts all processes entered in the Startup Tool consecutively as child processes. If the service is ended, it ends all processes it has sent in reverse order.

RIGHTS

All processes started by **zenStartupMrg** run under the user under which the service is started by the system. This is generally the local system account. Only resources that can access the local system account can be used.
The following are thus generally not reachable:

- Network drives
- Network printers
- Databases with the activated option **Use Windows NT integrated security**

If **zenStartupMgr** is started under a dedicated Windows user, no interaction with the desktop is possible.
Attention: The password of a dedicated user must not neither be changed nor expire. Otherwise the service will no longer start.

SECURE DESKTOP

Secure Desktop can be used to interactively access a process that has a user interface and is executed in a system context. The following must be the case for this:

- In the **zenStartupMgr** service, the option **Allow data exchange between service and desktop** must be activated
- The system service **Detection of interactive services** must be started

Note: This type of access is not suitable for use for the actual operation of Runtime. Services are intended as program instances that run in the background that are not operated by means of a user interface. If operations on the Runtime Server are to be carried out, Runtime must be started as an application. If Runtime is to run on the Server as a service, operation of Runtime can be carried out from a zenon Client.

Message boxes and modal dialogs are automatically suppressed if:

- zenon Runtime is started via **zenStartupMgr** in the system context
  and
- **Secure Desktop** is not displayed

> **Attention**
>
> Modal dialogs must never be called up via VBA or VSTA, because there is no possibility to close these again through user interaction.

9 Reloading Runtime

If the project configuration in amended in the Editor, Runtime can reload the amended files in ongoing operation. System-related changes are excluded from this. These generally require Runtime to be restarted.

To reload data:

1. Select the symbol for **Create amended Runtime files** or use Remote Transport.
2. Click on the button with the **Reload project online** function in Runtime.
   This must already be configured and present.

   Runtime loads the amended data.

RELOADING DELAYED BY THE SYSTEM

The reloading of Runtime is moved back to a later time by the system if:

- The user opens a context menu or a dialog
- A message box is shown

The reloading is only carried out in this case if these elements are closed again.

10 Runtime as ActiveX control

For KUKA HMIs, there is an ActiveX control available that allows the complete Runtime to run in a control. Runtime can thus be inserted into another application as a sub-program. The control acts, with a few exceptions, in the same way as the zenon Web Client and can also run in a browser.

Name Control: **zenRuntimeCtrl.ocx**

Differences to zenon Web Server:

- Can only run within a 32-bit zenon Runtime installation
- Server and standalone is also possible, not just client
  Note: The **Command Sequencer** and **Batch Control** module are not supported in this ActiveX control.
- Also starts the driver.
- Does not use zenon Web Server licensing.
  A zenon Runtime license is required
- The start project is read from the **zenon6.ini** file.
- OCX Properties server, zenon Web Server and project do not exist.
- Network communication is carried out using **zenNetSrv.exe**e, not **zenNetSrv.dll**.

11 Provision of zenon in the web browser without installation on the client

The **Open Source Apache Guacamole** project offers a possibility to use zenon Runtime to the full extent without installation on the client. Access is possible via a web browser.

You can find information on the download and installation of **Guacamole** at *https://guacamole.apache.org/*.

Attention

There may be performance impairments due to the necessary conversion of zenon screens for display. This affects screen switching most of all. This can take approximately 30 - 100% longer depending on screen content. These values also depend on the browser used.

Note: Artifacts can occur with the graphical display. For example, "levels" may appear in gradients. There may also be quality impairments when playing back videos.

12 Operation in the Runtime

TYPES OF OPERATION

Runtime can be operated with a mouse as well as with a keyboard or touch screen. For mouse or touch operation, you can primarily use buttons, status buttons, switches and invisible buttons in your project. The possibility of mouse operation of a screen object is signaled by a change in the cursor symbol. The arrow as a mouse pointer becomes a hand symbol.

Information

The switching time of screens in the Runtime depends on the number of linked variables. Only once all variables have been successfully signed in can the screen be operated.

If many variables are operated, this can slow the switching time. In this case, a progress bar is shown, which displays the loading progress.

12.1 Font type for dialogs in the Runtime

Font types for dialogs can be defined independently of the standard font type.
If the font size is increased, dialogs are also displayed larger.

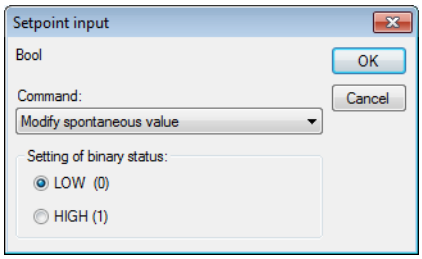
A separate font type for dialogs must be configured in the Editor:

1. Open the **Graphical design** group in the project properties.
2. Activate, in the **Runtime general** subgroup, the **Adjustable dialog font** property
3. Configure the desired font type using the **Dialog font** property.

Dialogs are then displayed in the selected font type and font size in the Runtime.

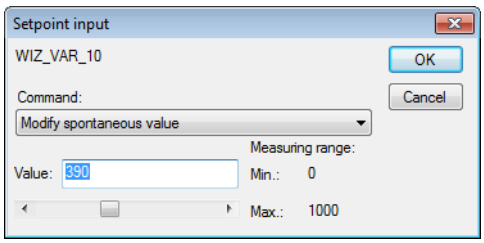
12.2 Mouse

Clicking on a dynamic element opens (depending on the setting of the dynamic element) the standard dialog for setpoint input. Depending on the data type of the variable the following dialogs are opened:



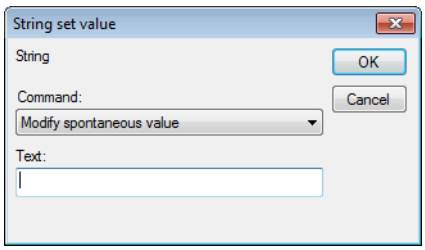
Possible operations are:

Parameter	Description
Switch to spontaneous value	Display spontaneous value (value from PLC) if substitute value was displayed before or the value was turned off.
Modify on-change value (default)	Write new spontaneous value to the PLC
Switching and modes. Spontaneous value	Combination of the two commands above
Switch to alternate value	Display substitute value (value from variable definition), if spontaneous value was displayed before
Modify alternate value	Set new substitute value for variable
Switching and modes. Alternate value	Combination of the two commands above
Switch off spontaneous value	Turn off connection to the PLC for this value by setting status bit *OFF* (bit *20*).
Switch on spontaneous value	Resume connection to the PLC for this value by resetting bit *OFF*.
Setting of binary status	- LOW (0) - HIGH (1)



Possible operations are:

Parameter	Description
Switch to spontaneous value	Display spontaneous value (value from PLC) if substitute value was displayed before or the value was turned off.
Modify on-change value (default)	Write new spontaneous value to the PLC.
Switching and modes. Spontaneous value	Combination of the two commands above.
Switch to alternate value	Display substitute value (value from variable definition),if spontaneous value was displayed before.
Modify alternate value	Set new substitute value for variable
Switching and modes. Alternate value	Combination of the two commands above.
Switch off spontaneous value	Turn off connection to the PLC for this value by setting status bit *OFF* (bit *20*).
Switch on spontaneous value	Resume connection to the PLC for this value by resetting bit *OFF*.
Value	New numerical value (within defined measuring range).



Possible operations are:

Parameter	Description
Switch to spontaneous value	Display spontaneous value (value from PLC) if substitute value was displayed before or the value was turned off.
Modify on-change value (default)	Write new spontaneous value to the PLC
Switching and modes. Spontaneous value	Combination of the two commands above
Switch to alternate value	Display substitute value (value from variable definition),

Parameter	Description
	if spontaneous value was displayed before
Modify alternate value	Set new substitute value for variable
Switching and modes. Alternate value	Combination of the two commands above
Switch off spontaneous value	Turn off connection to the PLC for this value by setting status bit *OFF* (bit *20*).
Switch on spontaneous value	Resume connection to the PLC for this value by resetting bit *OFF*.
Text	New text

LASSO

Dynamic elements that are linked to a variable or function can be pre-selected with the lasso in the Runtime and can therefore be used for VBA events.

SELECTION VIA LASSO

The following must be the case in order to select elements with the lasso in the Runtime:

- activate property **Interaction/Runtime lasso** in the project settings
- activate property **Runtime/selectable with lasso** in the property of the dynamic element

There are several methods for selecting elements available in the Runtime:

- Select elements: Left-click on a free area and move lasso over the screen elements while holding the mouse button pressed.
- Extend selection: **Ctrl+mouse click** on an element in order to select/deselect it in addition to the other elements already selected
- Add elements: While spanning the lasso, press and hold the **Control key** to add elements to the existing selection.
- Cancel selection: Spanning a lasso which does not contain elements.

12.3 Touch

With zenon, touch screens can also be operated with Multi-Touch gestures. You can obtain an example project from your zenon consultant.

The following is recommended for Multi-Touch:

- Use of a DirectX 11-compatible graphics card

- Use DirectX hardware in the project settings for the graphics quality.

Note: With Windows 8 gestures, interaction can be configured in many areas via Multi-Touch via properties with no additional programing necessary.

12.3.1 Navigation with Multi-Touch in the worldview

Multi-Touch gestures for zooming and scrolling are suitable for navigation on touch panels in the worldview. For this a screen of type Worldview overview is not necessary. The navigation can be implemented with:

- Windows 7 touch gestures (on page 53)
- Windows 8 touch gestures (on page 54)

RULES

- Move: If a screen in a container is not a worldview, it accepts the settings of the faceplate screen.

MOVING THE WORLDVIEW OR WORLDVIEW IN THE CONTAINER OF A FACEPLATE

- Screen is bigger than the frame: Content of the screen is moved.
- Screen is the same size or smaller than the frame: No reaction.

WORLDVIEW EMBEDDED IN FACEPLATE.

- Screen is bigger than the frame: Content of screen is moved.
- Screen is the same size or smaller than the frame: No reaction.

MOVE THE FRAME OR BORDER WITH THE MOUSE IF THE SCREEN IS A WORLDVIEW AND THE SAME SIZE OR SMALLER THAN THE FRAME:

- With the right mouse button: No reaction.
- With the left mouse button: Frame is moved.

CHANGE WORLDVIEW SIZE

The size of the worldview cannot be changed.
Exception: If the worldview is a faceplate, the size cannot be changed.

Click with right mouse button:

- Screen is bigger than the frame: Contents are moved.
- Screen is the same size or smaller than the frame: No reaction.

PROJECT CONVERSION

Values for **Move horizontally** and **Move vertically** when converting from an earlier version to zenon 7.20:

- Screen is bigger than the frame: *Move*.
- Screen is the same size or smaller than the frame: *No reaction*.

12.3.1.1 Navigation under Windows 7

To be able to use Multi-Touch gestures under Windows 7 to navigate in the worldview, you must:

- activate them via property **Multi-Touch for zoom and scroll**
- or implement them via VBA/VSTA

ZOOM AND SCROLL VIA PROPERTY MULTI-TOUCH FOR ZOOM AND SCROLL

To use Multi-Touch without VBA/VSTA:

1. In the project properties in the **Interaction** node for the **Recognition** property, activate *Windows 7*.
2. Deactivate property **Use screen size from frame** in node **Frame** at the properties of the screen
3. Activate property **Multi-Touch for zoom and scroll** in node **Interaction** at the properties of the screen

With this you can scroll and zoom in the screen with touch operation using Multi-Touch gestures. With this VBA/VSTA for zooming and scrolling is deactivated.

ZOOM AND SCROLL VIA VBA/VSTA

To implement zooming and scrolling via VBA/VSTA Events, property **Multi-Touch for zoom and scroll** must not be active.

The following is available in the `DynPicture`:

- Property

  `int ZoomLevel`: `Displays the current zoom level in the worldview (valid value only in the Runtime and for a worldview).`

- Style

```
SetZoomAndPos(float ZoomX, float ZoomY, int ZoomLevel, int CursorX, int CursorY, int PosX, int PosY, int PosMode);
ZoomX -> New zoom factor X direction; if not used, set to 0
ZoomY -> New zoom factor Y direction; if not used, set to 0
ZoomLevel -> Zoom level, if not used, set to -1
CursorX -> Cursorposition X
CursorY -> Cursorposition Y
PosX -> New position X (see PosMode)
PosY -> New position Y (see PosMode)
PosMode -> Coordinates in Pos
```

-1 = PosX, PosY are ignored

0 = center point , original coordinates

1 = center point, zoomed coordinates

2 = left top, original coordinates

3 = left top, zoomed coordinates

4 = zoomed coordinates of the cursor from the top left
The position of the window is changed in such a way that after the zooming, the mouse cursor is still over the same position of the screen

Attention: `ZoomX`, `ZoomY` and `ZoomLevel` can never be used simultaneously. Either you enter a `ZoomLevel` or a zoom factor for x and y axis.

12.3.1.2 Navigation under Windows 8

To navigate in a worldview with Multi-Touch under Windows 8:

1. In the project properties in the **Interaction** node for the **Recognition** property, activate *Windows 8*.
2. Deactivate, for the screen in the **Frame** group, the **Use screen size from frame** property and define the screen size as larger than the frame.
3. navigate to group **Interaction** in the screen properties.
4. Configure the properties for **Zoomen** and **Verschieben**.

For the move gesture, you can define the direction - horizontal, vertical or both. To do this, use the **Horizontal verschieben** and **Vertikal verschieben** properties.

12.4 Keyboard operation

You can also use the keyboard for all navigation and operation in the Runtime. In doing so, the focus is first set to a frame; within the frame, you can navigate with the arrow keys and instigate an action with the **Enter key**. The following functions (on page 55) are used for control:

- Set focus to frame
- Take focus away from frame
- Move focus
- Activate input to the element with the focus

The order in which the elements are selected is defined via the allocation of index numbers (on page 59) to the elements of a screen. This definition of the navigation order also works with symbols.

The keyboard operation can also be carried out with freely allocated shortcuts such as **Ctrl**+**M**. The shortcuts can be assigned to each operation element at the property **Key combination**.

12.4.1 Functions

For engineering the keyboard operation use the following functions:

SET FOCUS TO FRAME

This function sets the focus to a defined frame when operating the keys in the Runtime.

To configure the function:

1. Select, in the list of functions, in the **Screens** node, the **Set focus to frame** function.

   The dialog for selecting a frame is opened.
2. Select the frame you wish to assign
3. For multi-monitor projects, select the virtual monitor for opening the frame.
4. Close the dialog by clicking on OK.
5. Define the screen element that is to get the focus.
   To do this:
   a) Open the dialog for **Move sequence for focus** (on page 59) with **Sequence for left/right**.
   b) Define the element that should be triggered first.

Behavior in the Runtime:

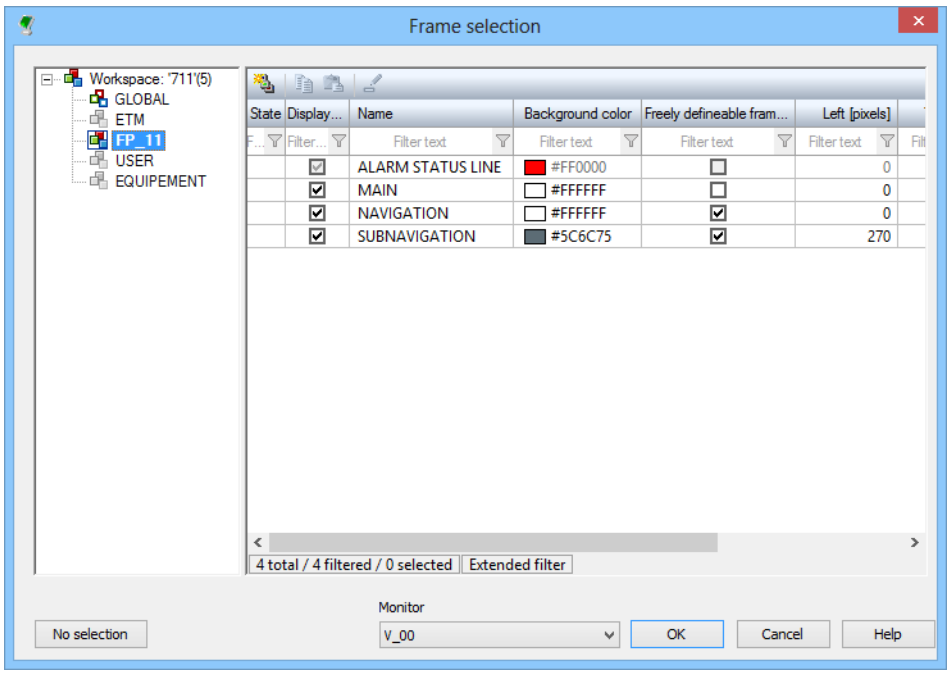
- The frame with a focus is displayed with a border. The line width and color of the frame are defined in the **Graphical design**/**Screens** node in project properties.

- In the screen, the element gets the focus that has been defined as the first element in the sequence for **left/right**. The sequence for **top/bottom** is ignored.

FRAME SELECTION DIALOG

In the **frame selection** dialog, frames can be selected for the execution of functions, from:

- Current project
- Subprojects
- All projects in the workspace with the **Keep project in memory** option active



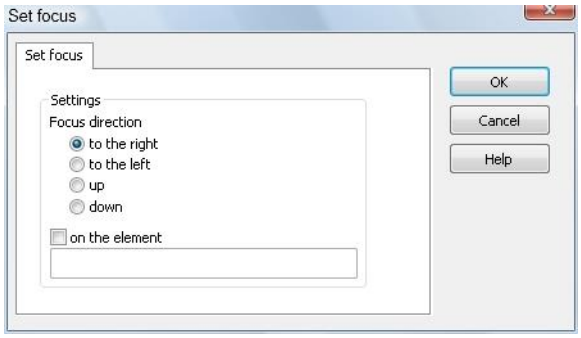
Parameter	Description
Project tree window	Displays all projects in the workspace. Frames can be selected from the current project and from all projects with the **Keep project in memory** option active.
Frames window	Selection of a frame. If several frames are selected, the frame at the top of the list is used to execute the function.
No selection	Removes selection and closes dialog.

NOTE: THE SIZE OF THIS DIALOG CAN BE ADJUSTED. THE POSITION AND SIZE OF THE DIALOG WINDOW ARE SAVED IN THE EDITOR, RUNTIME AND ZENON WEB CLIENT INDEPENDENTLY OF THE PROJECT.SET FOCUS ON FRAME WHEN CALLING UP FRAMES SEVERAL TIMES

If a frame is displayed on a monitor several times, then the focus is switched in the reverse of the switching sequence each time the functions of the focus are executed. If, for example, with a frame that is called up several times, the screens **Screen1**, **Screen2** and **Screen3** are open and then the focus is set to this frame, the focus goes to the last screen opened, **Screen3**. When the **Set focus to frame** function is called up again, the focus is set to **Screen2** etc.

MOVE FOCUS

This function set the focus on a particular element in the Runtime with keyboard operation and can therefore be used to navigate within a frame.



Property	Action
Direction	Define the direction in which the focus should be moved. The following defined sequence applies in the editor in the context menu or under **Edit/change focus sequence...**
on the element	Definition of the element for the focus. Enter the object name of the element.

ACTIVATE INPUT TO THE ELEMENT WITH THE FOCUS

This function activates the element that is being focused on in the frame selected.

TAKE FOCUS AWAY FROM FRAME

This function takes the focus from the current frame in the Runtime. To continue operating the keyboard, the focus must be set to a frame again.

12.5 Lock system keys

Windows keyboard shortcuts can be blocked for online operation. The block is configured in the Editor. It is also possible to block all system keys for Runtime.

> **Attention**
>
> This functionality is not available for CE projects.

LOCK WINDOWS SHORTCUTS

To block Windows keyboard shortcuts:
Activate the **Lock system keys** property in the **Interaction** area of the project properties.

LOCKED KEYBOARD SHORTCUTS

The following are locked:

Key combination	effect
Alt+Shift+Tab	Switches between open applications. Direction: backwards.
Ctrl+Tab	Switches to the next window within the application. Can be assigned application-specifically.
Alt+F4	Closes application.
Ctrl+Shift+Esc	Starts the Windows Task Manager.

All corresponding combinations with mouse actions are also blocked.

> **Hint**
>
> To lock all key combinations with the Windows key: Deactivate the **Windows key** in the **Startup Tool** under **Application -> Options -> General**.

KEYBOARD SHORTCUTS THAT ARE NOT BLOCKED

The following are not locked:

Key combination	effect	Workaround for a lock
Windows key + L	Locks the computer. **Attention:** All Windows keyboard shortcuts are available again after	Deactivate the **Windows key** in the **Startup Tool** under **Application -> Options -> General**.

Key combination	effect	Workaround for a lock
	unlocking.	
Alt+Esc	Switches to the next application.	Lock via **Keyblock Runtime Start** (on page 36).
Alt+Tab	Switches between open applications. Direction forwards.	Lock via **Keyblock Runtime Start** (on page 36).
Ctrl+Esc	Calls up Start menu.	Lock via **Keyblock Runtime Start** (on page 36).
Ctrl+F4	Closes the active window.	There is no workaround.

LOCK SYSTEM KEYS

To lock system keys in the Runtime:

- Activate the **Lock system keys** property.
- Deactivate the **Windows key** in the **Startup Tool**.
- Note the information in the **Protect Runtime files** (on page 38) chapter.

12.6 Define sort order inside a frame

You define the sequence for the actuation of the elements in a screen with the project configuration in the zenon Editor:

1. Open the screen.
2. Right-click an empty area.
3. Select **Shift order for focus** from the context menu.
4. Select from the drop-down list **Order for left/right** or **Order for up/down**.
5. Each element in the screen is displayed with a number in the top left corner.
6. Click on the element which should be selected first.
7. the number of the element changes to **1**.
8. click on all other elements in the order you want them to be selected:
   - Elements which have already been defined change the background color of the number.
   - Elements that cannot be selected using the keyboard in the Runtime (**Focus is set to the element** property *inactive*) are displayed with red figures.
   - If you click on an element while holding the **Control** key, the numbering starts with the index number of this element.

Information

In the properties of the elements the position is displayed via the properties **Focus position left/right** and **Focus position top/bottom**. These properties are for information only. The index values of the position cannot be changed there.

SYMBOLS

An index can also be assigned to **symbols**. During compiling the Runtime files, the symbol is disjointed and its elements are inserted. Thereby the order for selecting the elements which was defined in the symbol is taken into consideration.

Example

There are three elements in this screen: the numerical value **Z**, the button **B** and the symbol **VS**. The symbol from the project library receives the three numerical values: **N1**, **N2** and **N3**.

If you define the order in the screen as: **B** - **VS** - **Z**,
then the order in the Runtime is: **B** - **N1** - **N2** - **N3** - **N**.

12.7 Positioning and operation of frames

Frames can be switched to fixed or relatively defined positions in the Runtime. In this way, keyboard screens or pop-up screens can be positioned exactly where the mouse pointer or the element to be switched is located. Frames can be switched in the Runtime:

- at an absolute position
- relative to the mouse position
- relative to element

Absolute and relative positions are defined in the properties of the frame in the zenon Editor.

KEEP POSITION WHEN DISPLAYING AGAIN

Windows always take their screen position from the template. If pop up screens are planned and moved whilst the program is running, the pop up screen is moved back to the planned position the next time the screen switching function is activated. Using the **Keep position and size on reopening** property, the template can be set to keep its position as long as it is open, regardless of screen switches. If the property is active:

- If a screen is opened in Runtime in a manually-moved frame, its position and size are also retained if the screen is switched.
- If another screen is opened in the in the same template, its position and size are retained. This also applies for a substituted call.

With this, a different position position for a template can be used on each client, because the local position is always saved.

ABSOLUTE POSITIONING

The frame is always switched at fixed, pre-defined coordinates. A fixed screen position is always used for all screens that are loaded automatically by the system, e.g. by the time control or a limit value violation.

RELATIVE POSITIONING

Relative positioning is especially suited to loading pop-ups, keyboard screens or command processing screens. This is possible for all screen switches, set value inputs with freely defined keyboard screens, and command processing screens that are started manually by the user. It does not matter whether the action was triggered by a screen switch function, a set value function or directly via an element. Relative positioning is available for:

- dynamic elements to which you can allocate a function
- dynamic elements that can be used to send a set value (buttons, combined elements)
- Main and Context Menus
- execute the screen-specific function in the Alarm Message List screen

Relative positioning also works for nested function calls, i.e. when screen switching or the setting of values are triggered by a script. If the screen were to go beyond the edge of the monitor, it is automatically positioned at the edge of the monitor. This also applies to multi-monitor management, so that it cannot protrude into the adjacent monitor.

Attention

Relative positioning only works when called up directly by the user, not with automated functions or via VBA.

RELATIVE TO THE ELEMENT

Positions the reference point of the frame relative to the element from which the screen switch was initiated (e.g. a button or a combined element). The frame's reference point is then always placed on the element's calculated reference point in the Runtime. If the frame cannot be switched on in this position because the screen would then be outside the visible screen area, the alternative reference points are used.

First define a reference point for your frame and then the preferred position in the element. The following settings are available:

Property	Value
Reference point element vertical	▸ bottom ▸ top
Reference point element horizontal	▸ left ▸ right
Vertical movement in pixels	Enter the desired movement
Horizontal movement in pixels	Enter the desired movement

ALTERNATIVE POSITION

Define an alternative position for the reference point of the frame and the reference point at the element. Proceed with the definition of the alternative positions in the same way as with the definition of the favored positions. The alternative position is used when the frame cannot be placed at the favored position.

RELATIVE TO THE MOUSE POINTER

The screen is displayed depending on the position of the mouse pointer. If the reference point is at the top left side, the frame will be loaded to the bottom right side of the reference point. If the screen cannot be displayed completely at the configured position, the position will be moved until the complete screen is inside the displayed area of the monitor.

Define the frame's reference point above the two properties, vertical reference point and horizontal reference point. The reference point is marked on the frame with a red dot. The following settings are available:

Property	Value
Reference point frame vertical	▸ top ▸ bottom ▸ centered
Reference point frame horizontal	▸ Left ▸ right ▸ centered

Example

you have a button at the right border of the screen and try to position a popup window on its right side:

- Reference point element: right/centered
- Frame reference point: left/centered

In the Runtime, the pop-up window will appear to the right of the button, centered in the middle of the button. The popup window will not cover the button.

You use the same frame also for a button on the left screen border. There may not be enough room on the right side to completely display the popup screen. The alternative configured position will therefore be used. For this, you place the

- alternative reference point for the element on: left/centered
- the frame's alternative reference point on right/centered

CHANGE SIZE IN THE RUNTIME

The size with which a screen is called up in the Runtime and the permitted changes in size are configured with the properties of the **Position** group. In the zenon Editor, navigate to the **Screens** node. Click on the **Frames** subnode there and select a configured template in the *preview window*.

Property	Description
Width (maximum) [pixels]:	Defines the maximum width.
Height (maximum) [pixels]:	Defines the maximum height.
Limitation Minimum:	Defines limits for minimum. Possible values: - *Without*. No limitation If a different value is selected, the selected limit in the Runtime is displayed with a dotted line in the frame window. - *Width*: Width limitation. - *Height*. Height limitation. - *Relative*: Limitation to a percentage value of the set screen size. Only has an effect on Multi-Touch gestures. The corresponding values are defined with the **Value (minimum)** property.

Property	Description
	Default: *without*
Opening size:	Defines the size with which a screen based on this frame is called up in the Runtime: ▸ *Frame size*. Size as defined in **Width (maximum) [pixels]** and **Height (maximum) [pixels]**. If a different value is selected, the size in the Runtime is displayed with a dotted line in the frame window. ▸ *Width [px]*. Width as defined in **Value (Opening size)**, height is amended accordingly. ▸ *Height [px]*. Size as defined in **Value (Opening size)**, width is amended accordingly. ▸ *Relative [%]*. The size defined in **Width (maximum) [pixels]** and **Height (maximum) [pixels]** is amended to the percentage value defined in **Value (Opening size)**. The corresponding values are defined with the **Value (Opening size)** property. Default: *Frame size*

MOVING AND ZOOMING

Frames can be moved and zoomed in the Runtime.

MOVE

Moving always relates to the monitors defined in the monitor administration. These settings are not supported under Windows CE.

The moving of frames in the Runtime can be limited to:

- *Frame border*:
  The frame cannot be moved beyond the monitor limit.
- *Frame border area*:
  The frame can be moved beyond the monitor limit. However there must be an area on the monitor (**Minimum frame margin**) that can be accessed for further actions.

The limit is only effective if the frame is already within the defined limit when it is opened. If the moving of a frame is not limited, it can - with Multi-Touch gestures for example - be moved beyond the visible area.

Information

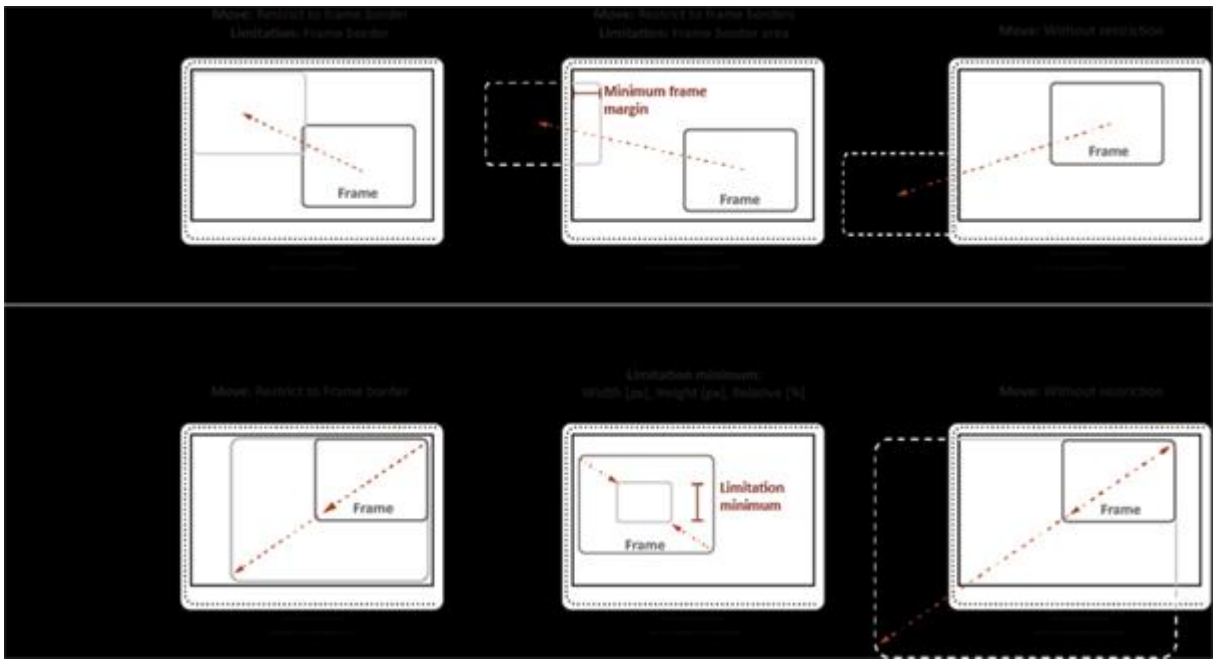
The moving of frames is most of all envisaged for full-screen operation. The Runtime application itself should not be moved to a different screen. If Runtime is moved, this can lead to limitations when moving.

ZOOM

The limits also apply for zooming. If a limit is reached when zooming, an attempt is made to continue zooming in the free space. In doing so, the side ratio is retained.

MOVE AND ZOOM OVERVIEW

When moving and zooming, the frames act as follows, depending on the settings made:



12.8 Configurable lists

A range of lists can be individually configured in the Runtime:

- Active Directory:
  - List in the Active Directory window (but not: tree)
- AML
- AML filter/CEL filter/time filter:
  - Lots: Archive selection

- Lots: Lot selection
- Batch:
  - List of master recipes
  - Control Recipes List
  - Tag List 1
  - Tag List 2
- Command Processing
  - List of all interlockings of the action
- User list:
  - User List
- CEL
- Extended Trend:
  - Expanded curve list
- Message Control:
  - Message queue
- Process Recorder
- RGM:
  - Recipe list and recipe value table (but not: CE recipe value table)
- Shift Management
- Text lists

The following are possible actions for these lists:

- Moving the columns
- Configuration of the columns
- Coloring of columns
- Highlighting of the position with the focus
- Use of response variables

For general information, see also the **Design lists** chapter.

Information

Touch boxes can be configured like lists:

- AML/CEL/time filter
- Edit User
- User Groups List

Note: Not all elements can be configured with touch boxes. For example, there is no header and there is only one column.

CONFIGURATION IN THE EDITOR

The appearance and behavior of the *configurable lists* in the Runtime can be preconfigured in the Editor using the properties of the **Representation** group.

Exception: Command input - the list of **all interlockings** is pre-configured in the **list configuration** property of the **command input**.

PROPERTIES OF THE HEADERS

The behavior of headers in the 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 the Runtime.
- **Show filter row**:
  Controls whether the filter line is displayed in the Runtime.
- **Disable sorting**:
  Controls the possibility of sorting table columns in the Runtime with a click on the header.
- **Freeze column location**:
  Controls the possibility to amend or move the width of table columns in the Runtime with mouse actions.
- **Deactivate context menu**:
  Activates or deactivates the context menu for the header.

BEHAVIOR IN THE RUNTIME

EDIT CELLS

You have the following possibilities for editing cells in the *configurable list* type lists:

- Double click on the entry

- Click in the cell, immediately followed by a second click (slow double click)

Depending on the screen type, there may also be a button available to activate editing.

COLOR COLUMNS

In configurable lists, the text color and background color of each column can be defined individually. To do this, configure the **User-defined colors** area in the *screen switching* function of the respective screen, in the **Format columns** tab.

SHOW FOCUS

In configurable lists, the respective focus can be signalized by means of different text and background colors. The cell, column or line that is in focus is emphasized in bold. These colors are configured in the **Selection colors for the object lists** project properties in the **Graphical design** node.

PRIORITIES WHEN COLORING

If several colors are applied on a list, the following priorities apply:

1. General: **Selection colors for the object lists**
2. RGM: **Recipe value validation**
3. RGM: Online validation
4. RGM: **Interlocking**
5. List: Column color

RESPONSE VARIABLES

Configurable lists can be linked to a *BOOL* response variable. These signalize if something has been selected in the list in the Runtime.

To link a response variable:

1. Click on **Configurable lists**.
2. Open the **Response variables** group in the properties.
3. In the **Selection active** property, click on the **...** button.

   The dialog to select a *BOOL* variable is opened.
4. Please select a variable.
5. Link the variable to an element that displays the status of the variable in the Runtime.
   Meaning of the values:
   - *0*: No selection in the list.
     Note: If a screen is closed, the list is automatically set to *0*.

- *1*: Something has been selected in the list.

12.9 Runtime messages

In Runtime, messages in relation to data export and user actions can be displayed for certain events. You configure these in the Editor.
To show messages:

1. Open the project properties for **Runtime settings**.
2. Configure the desired properties of the **Runtime messages for** group.

MESSAGES

Property	Description
Data export	Shows the result of a data export that has been triggered using the **Export data** function.
Replace files	Shows whether a file will be overwritten during data export.
Incorrect input	Allows Runtime messages for user actions to be switched on or off. This is applicable for the following user actions: - User locked - Entry of an invalid user name - Entry of an invalid password - System is locked - System is being unlocked - Your account has been locked - Write set value
Insufficient authorization	Makes it possible, if there are not sufficient authorizations, to display messages with different detailing. Select from drop-down list: - *No message*: No message is shown in the event of access being refused due to insufficient authorizations. - *Simple*: A message with notification of insufficient authorization is shown.

Property	Description
	▸ *With level number*: A message with notification of insufficient authorization is shown. The number of the required authorization level is shown. ▸ *With level name*: A message with notification of insufficient authorization is shown. The name of the required authorization level is shown. ▸ *With level number and level name*: A message with notification of insufficient authorization is shown. Number and name of the required authorization level are shown. Default: *Simple* Only available if the **Incorrect input** property is active.
Interlocked elements	There is notification in the Runtime of elements that could not be operated due to interlocking. The detailing of the declaration can be configured. Select from drop-down list: ▸ *No message*: No information is displayed. ▸ *Simple*: There is a notification that an element is interlocked. ▸ *With interlocking text*: There is a notification that an element is interlocked. In addition, the value for the **Interlocking text** property is displayed for the first locking interlocking found. ▸ *With linked interlocking text*: There is a notification that an element is interlocked. In addition, the values for the **Interlocking text** property are displayed for all locking interlockings. Default: *No message* Only available if the **Incorrect input** property is active.

VALIDATION

If access is blocked, corresponding messages are displayed. Depending on the cause, the system will be locked. A temporary login may be possible.

Reason	System	Temporary login
User not authorized	Locked	Not possible
Missing authorization for element.	Not locked	Possible
Element is interlocked	Not locked	Not possible
Missing authorization for element plus element is interlocked	Not locked	Not possible

13 Runtime files

DEFINITION OF RUNTIME FILES, RUNTIME DATA, RUNTIME FOLDER AND DATA FOLDER

RUNTIME FILES

Runtime files are files that are created by the editor that are read and interpreted by Runtime. Some Runtime files are modified by Runtime, for example:

- Standard Recipes,
- Recipegroup Manager etc.

The Runtime files are stored in the Runtime folder. This is defined in the Editor under *General*, in the properties window.

RUNTIME DATA

All data that is created and administered by Runtime, such as data from:

- The alarm message list,
- The Chronological Event List
- The Historian etc.

This also includes data that is created by the Editor (Runtime files) and that can be amended by Runtime, such as data from:

- Standard Recipes,
- Recipegroup Manager,
- User administration, etc.

The Runtime files are stored in the Runtime folder. This Runtime folder is configured in the Editor in the **General** project properties group in the **Runtime folder** property.

RUNTIME FOLDER

The Runtime files of the project are saved in this folder. The path is entered into *zenon6.ini* when the start project is set. The Runtime folder can be amended in the project properties. In the Runtime folder, there is the subfolder *RT* and a subfolder with the name of the computer. In the *RT* folder, there are all Runtime files that are created by the Editor. In the *Computer name* folder, there are all data files that are created in Runtime.

Note: Ensure that there is sufficient memory where the Runtime data is, because there can be much data, due to archiving, AML, CEL etc.

DATA FOLDER

The Runtime saves all data files that were created in the Runtime, such as alarm files, archive files etc. in the data folder. The data folder is created as a subfolder of the Runtime folder by default. The folder is automatically assigned the name of the computer the Runtime is running on. This save location can be amended in the project properties (**General**/**Data folder**).

Hint: Never set the data folder to a removable device such as an USB stick or a network device. It is recommended that the data is recorded locally and backed up externally.

13.1 Compatibility Runtime files

zenon Runtime is backwards compatible in the network and as a standalone.
That means:

- The Runtime can always load projects from older version and interpret and display these projects in accordance with their version.
- Even if Runtime, the server and standby all have a higher version number, they can load projects from older versions and interpret and display this version accordingly.
- Mixed operation is possible.
  With multi-project administration, projects from different versions can be loaded and run at the same time.

Note: Projects from version 6.20 SP4 on can be started directly without being converted first. Projects with a lower version number must be converted beforehand.

ONLINE COMPATIBILITY

The Runtime online compatibility enables Runtime systems to work together in the zenon network, as well as zenon web clients.

In doing so, the following applies: The version of the client Runtime must be the same or higher than the version of the server Runtime.
For example:

- A 8.20 client can work together with a 8.10 server.
- A 8.00 client can no longer work together with a 8.10 server. In this case, the client Runtime must be updated to version 8.10 or higher.

The current Runtime can load projects of the following versions:

- *6.20 SP4*
- *6.21 SP0*
- *6.21 SP1*
- *6.22 SP0*
- *6.22 SP1*
- *6.50 SP0*
- *6.51 SP0*
- *7.00 SP0*
- *7.10 SP0*
- *7.11 SP0*
- *7.20 SP0*
- *7.20 SP0[current Build-No.]*
- *7.50 SP0*
- *7.60 SP0*
- *8.00 SP0*
- *8.10 SP0*
- *8.20 SP0*

Due to the multi-project administration, projects from different versions can be loaded. For example, the integration project can have version 8.20, a sub project version 8.10 and another sub project version 7.60.Mixed operation also works in the network. With this different versions can also be started with the zenon Web Client.

Note: Starting with version *7.00*, if a graphics quality other than *Windows Basic* is set for a project in the editor (including the corresponding setting for the **Create Runtime files for** option), Runtime up

to zenon version *6.51* loads the *Windows extended* graphics quality. If the *Windows basic* graphics quality is set, this is retained.

Attention: Server and Standby Server must have the same version.

Note for Batch Control module: Runtime files for Batch Control up to and including version 7.00 SP0 are not compatible with subsequent versions. Compatible from version 7.10 onward.

VERSION 7.20 - LATEST BUILD

For compatibility with version 7.20, there is an additional possible selection available: "*Most recent version*". This can be selected by clicking the *7.20 SP0 + [most recent build no.]* entry in the drop-down list.

Selecting this option makes the Runtime files available for the current build of version 7.20. 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. In order to use it, you must have the most recent build of 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.Ensure that you use the most up-to-date build for both zenon 7.50 and 7.20.

COMPATIBILITY BETWEEN THE EDITOR AND RUNTIME:

With the zenon Editor, Runtime files can be created for different versions of Runtime. The Runtime version therefore does not need to correspond to the Editor version. This backward compatibility is particularly suited for use of mixed systems. For example: A project that has been configured with the Editor for version 8.00 can also be started with Runtime 6.20 SP4.

Attention

If, in a project with a later version of the Editor, properties are shown that are not available in the earlier version, these are not available. This can lead to unwanted results in the Runtime. When creating Runtime files for older versions, check for changed project configurations, in particular the driver configurations.

CREATING RUNTIME FILES

To create Runtime files in the editor for earlier versions:

1. Select the project in the project tree.
2. Navigate to the **General** section in project properties.
3. Open the **Create Runtime files for** property drop-down list.
4. Select the desired version from the drop-down list

Attention: In order to ensure consistency of Runtime files, all Runtime files must be newly created each time this property is changed. The configurations for all drivers are converted. Settings that do not exist in the respective version are set to the default setting.

ERROR CREATING RUNTIME FILES AND MICROSOFT OFFICE 365

In a certain configuration, an error may occur when generating Runtime files:

- You can create Runtime files on a computer with:
  - Windows 10
    operating system and
  - Office 365.
- Creating Runtime files has failed and is terminated with an error message.

This is caused when an incorrect version of a program library by VBA is loaded.

Solution:

1. Navigate to the folder: *%AppData%\Microsoft\FORMS*.
2. Delete the file **zenone32.box**.
3. This file is created new automatically by the zenon.

Creating Runtime files is possible again.

14 Runtime profiles

A Runtime profile is a reproducible optical snapshot of the screen in the Runtime including information about:

- Screen pattern (succession and position)
- Monitor assignment
- Filter

Runtime profiles are project- and user-related.

The following screen types support Runtime profiles:

- Alarm Message List
- Archive revision
- Chronological Event List
- Extended Trend

- Industrial Maintenance Manager
- Industrial Performance Analyzer
- Report Generator
- Message Control
- Variable Diagnosis

Every zenon user has an own folder in which he can administrate his profiles. Administrators can see and edit all profiles of all users and copy them to user *System*. From user account *System* all users can load Runtime profiles. It serves as an exchange account for Runtime profiles.

Note: Only administrators can copy to and manage profiles in the *System* user account. All other users can only manage their own profiles and load the profiles of the *System* user.

PREDEFINED PROFILES

Besides the individual profiles there are two pre-defined Runtime profiles:

- *DEFAULT*: created by the user and selected as the default profile
- *LAST*: is automatically saved to the user's directory when they log out
  Exception: The *LAST* profile is not saved:
  - for user **System**
  - a temporary login without writing permission

To each zenon user a start profile (on page 79) can be allocated during log in.

Attention

Profiles can only be saved correctly at the Server if all projects are available at the Server:

- from which screens are called up at the Client
- which call up screens with a screen switch function

The saving of Runtime profiles that do not correspond to these conditions can lead to errors in the Runtime.

14.1 Load and create profile in the Runtime

To create a profile in the Runtime, a corresponding function (on page 77) must be configured:

- Save profile: saves the current profile as it was defined in the function

- Profile administration (on page 79): opens the administration of the profiles for saving, loading, allocating and administrating profiles

In addition, pre-existing profiles can be activated in the Runtime:

- Load profile

When loading a profile, the same state as when the profile was saved is restored. The screen structure including all filters as it was when the profile was saved is restored.

Attention: If there are screen switch functions defined at the properties of a screen for functions **Start function** or **End function** - such as (**Screen switch**, **Index screen** or **Screen back**), they are not carried out! All other function types are carried out again.

If a script is called up in the two functions, the complete script and all screen switching functions therein are executed. Therefore it can happen that the screen structure after loading the profile does not match the screen structure when saving the profile.

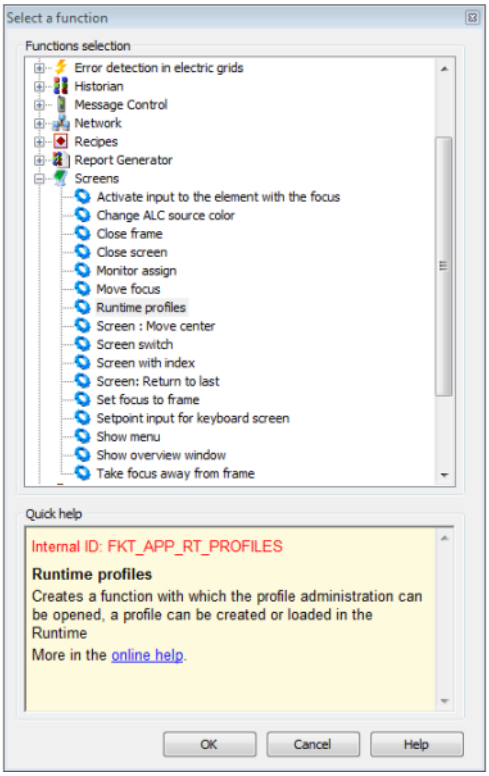
A similar behavior displays variable **'Screen active' variable**: If a limit value which contains a screen switch function is carried out in the variable, the function is carried out and may affect the display in an undesired way.

14.2 Engineering of a function

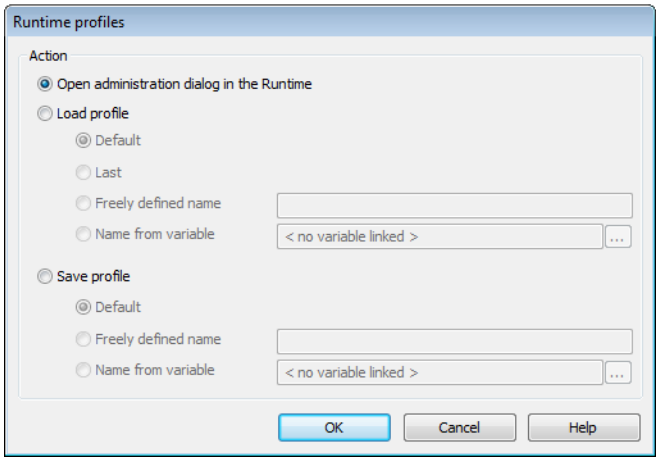
To load, save and administrate profiles in the Runtime, you configure a function in the Editor and assign it to a button:

- Select **New function**
- navigate to node **Screens**

- Select **Runtime profiles**



- the dialog for engineering the function is opened



Property	Action
Open administration dialog in the Runtime	Opens the dialog for administering (on page 79), creating, saving and loading profiles in the Runtime.
Load profile	Loads the profile defined in the dialog in the Runtime:
DEFAULT	Loads the *DEFAULT* profile. It was selected as the standard profile by the user from all existing profiles.

Property	Action
LAST	Loads the *LAST* profile. It was automatically written to the folder of the user at log out.
Freely defined name	Loads the profile with the name which was defined in the dialog. If the profile does not exist, the current profile is kept.
Name from variable	Loads the profile whose name was generated from the defined variable.
Save profile	Saves the profile which was defined in the dialog.
DEFAULT	Saves the current profile as *DEFAULT*.
Freely defined name	Saves the profile with the name which was defined in the dialog.
Name from variable	Saves the profile under the name which was created from the variable defined in the dialog.

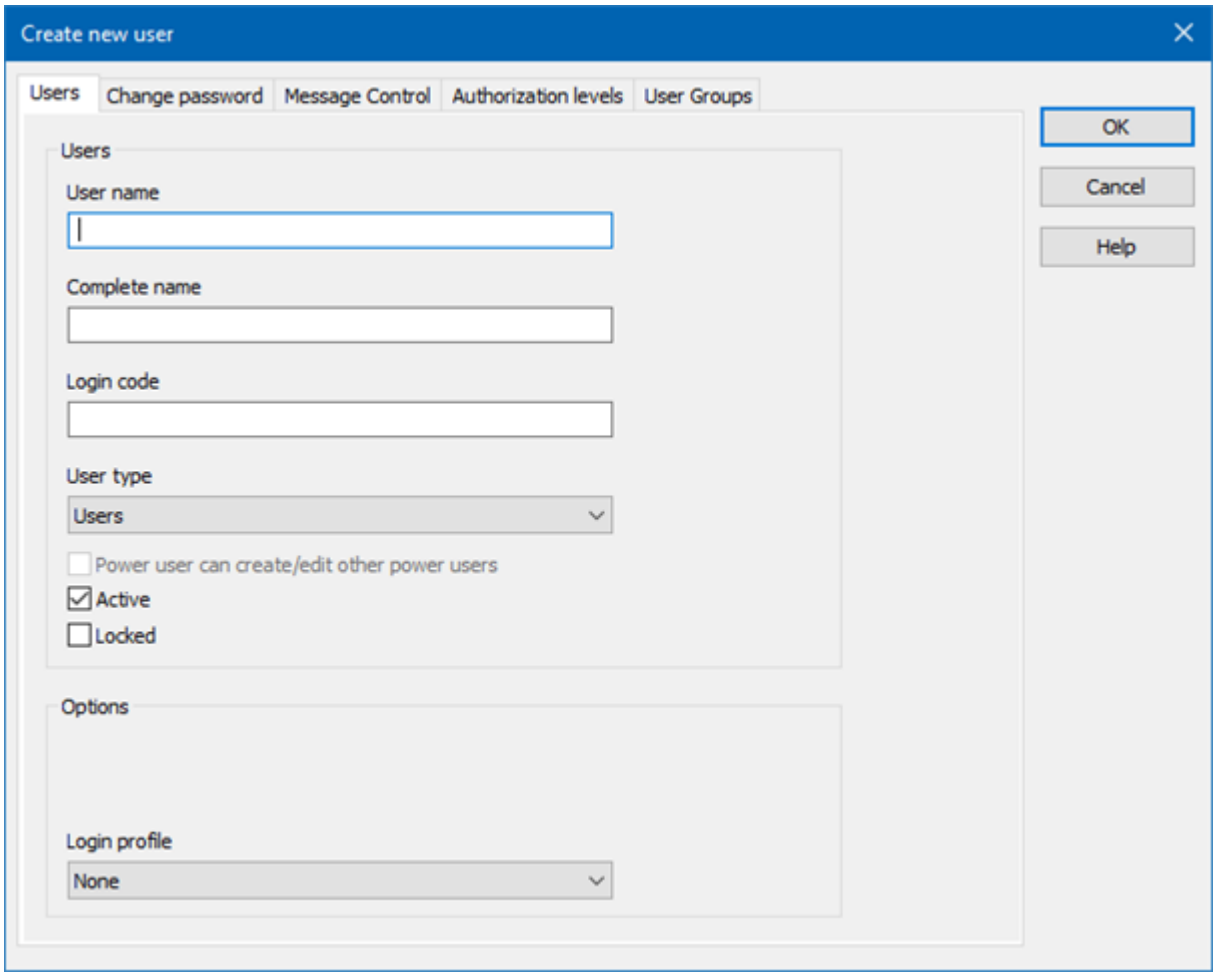
14.3 Allocate and administrate profile

Profiles can be allocated in the Editor and in the Runtime. They are created and saved in the Runtime.

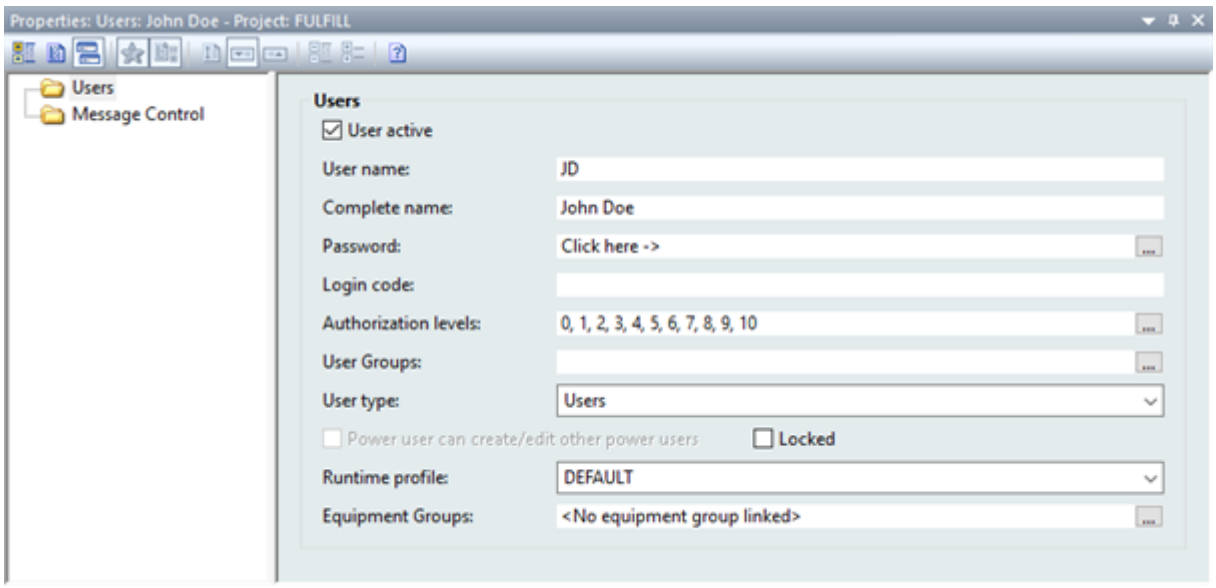
IN THE EDITOR

Users are assigned Runtime profiles in the Editor:

- In the dialog when creating a new user with the **login profile** property:



- Via the property **Runtime profile** of a user:



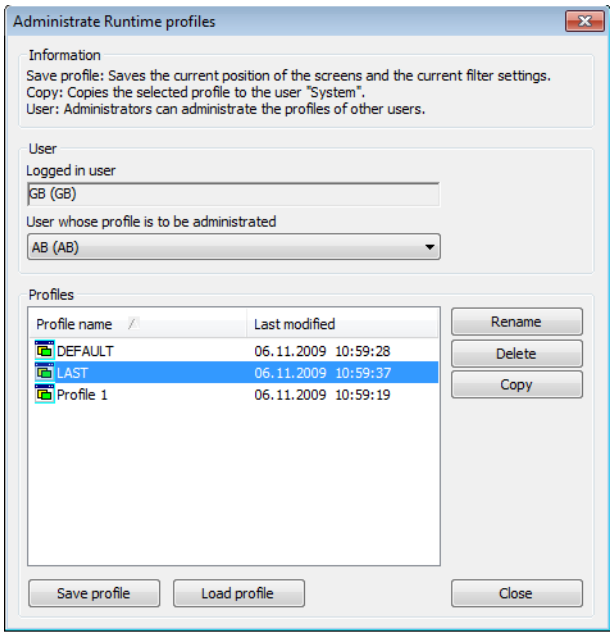
- In the drop-down list the following settings are available:

- *None:*
  No profile is allocated (default setting)
- *DEFAULT:*
  The *DEFAULT* profile is assigned
- *LAST:*
  The *LAST* profile is assigned

IN THE RUNTIME

In the Runtime you can save, load, allocate and administrate profiles with the help of the profile administration. For this you must engineer function (on page 77) **Runtime profiles** with property **Open administration dialog in the Runtime**.

Note: Administrators can administrate the profiles of other users.



Property	Action
Logged in user	User who is currently logged in to the system.
User whose profiles are administrated	User whose profiles are displayed and can be administrated.
Profiles	List of the available profile.
Profile name	Name of the profile.
Last modified	Date and time of the last modification.
Rename	Opens the selected profile name in order to

Property	Action
	rename it.
Delete	Deletes the selected profile after a confirmation message.
Copy	Copies the selected profile to the user *System*. At this the name can be changed. **Hint:** Profiles at user *System* are available for all users which are not logged in.
Save profile	Opens the dialog for assigning a profile name and saving the profile under this name. The following characters are forbidden: *Space* and the special characters */ | \ : * ? ! " < > $*
Load profile	Loads the selected profile.
Close	Closes the dialog.

KEYBOARD SHORTCUTS

Key	Action
F5	Updates the profiles' list. At unexpected events in the zenon network such as the loss of server or standby the contents of the list are automatically updated.
Esc	Closes the dialog.
Enter	Loads the selected profile.
Ins	Saves the current state of the new profile.
F2	Makes it possible to change the name of the selected profile.
Del	Deletes the selected profile after a confirmation message.

14.4 Storage directories of the profiles

The profile data are stored differently depending on their use in the Runtime:

Type of Runtime	Storage directory
Server or standalone:	Data are stored locally.

Type of Runtime	Storage directory
Client	Data are stored on the server and are requested and changed interactively.
Standby	Data are stored on the server and are requested and changed interactively and are synchronized locally.

15 Filtering in Runtime

Filtered data can be displayed in Runtime. To do this, configure the desired **filter** in the screen switch in Editor or in Runtime. Filters can also be saved in Runtime using **Filter profiles** (on page 83).

15.1 Filter profiles

Filter profiles are filter settings that the user can save and call up in Runtime in relation to a certain screen.

To be able to use filter profiles, the following control elements must be configured:

Control element	Description
Filter profiles	Profile administration in the Runtime
Profile selection	Selection of a saved profile from a drop-down list.
Save	Clicking on the button in the Runtime saves the filter settings as a profile. **Note:** The name can be a maximum of 31 characters long and must only contain valid characters. Prohibited are: **! \ / : * ? < > | ""**
Delete	Clicking on button in Runtime deletes the selected profile.

You can thus do the following in the Runtime:

- save filters
- use saved filters
- delete filter profiles
- Filter profiles can also be exported and imported (on page 84) with further control elements.

STORE FILTER PROFILE

To create a filter profile:

1. Define filter conditions
2. Enter a name in the **Filter profile** input field
3. Click the **Save** button.
   The filter profile is saved and can be selected in the drop-down list.

USE FILTER PROFILE

To use a filter profile:

1. Select a filter from the drop-down list **filter profiles**.
2. The filter is immediately applied.

DELETE FILTER PROFILE

To delete a filter profile:

1. Select a filter from the drop down list **Filter profiles.**
2. Click on the **Delete** button.
   The filter profile is deleted.
3. The deleted filter is still applied as long as a new filter is defined or selected.

15.1.1 Export and import filter profiles

You can transfer filter profiles to other projects and other computers with the control elements for import and export.

To export profiles:

1. Create the control elements for import and export in the screen:
   Control elements -> Filter profiles -> Import or Export
2. Start the Runtime.
3. Open the screen.
4. Create the desired profiles.
5. Save the profiles.
6. Export the profiles.
   These are saved in an XML file and can be imported with the same screen type in another project

To import profiles:

1. Save (on page 82) the XML file with the desired profiles on the Runtime computer, if the computer differs from the export computer.
2. Start the Runtime.
3. Open the screen.
4. Import the profiles

> **Information**
>
> At the import all existing profiles are deleted. The profile active at the time is not changed.
>
> XML files can only be imported in the screen type in which they were created. If you try to import profiles of other screen types, the import is canceled and an error message is displayed.

16 Handling of date and time

Date and time in zenon are either local time, UTC time or time periods. The basic principle is that:

- all times "from - to" in filter such as screen switch to AML or CEL are saved in UTC
- Times in modules such as PFS and Load Management are made in Local Time
- Time periods are in seconds

Exceptions:

- the IPA writes historic data in local time as "*datetime*" in the database
- The zenon **Read time from variable** and **Write time to variable** functions do not save times but take over the local time formatted as string from the control or writes it to the control

UTC

UTC means Coordinated Universal Time. The time unit is second. UTC is the uniform basis for the international time determination and is made available to the public via time senders and other time services. Dependent on the time zone certain time periods must be added or subtracted to or from UTC. This time period can vary one hour because of the day light saving time.
Examples:

Country	Local time
Alaska	*UTC -9*
Australia, Queensland	*UTC +10*
Bulgaria	*UTC +2*
United Kingdom	*UTC*
Korea	*UTC +9*
Central Europe (CET)	*UTC +1*
Central Europe (CET) Daylight Saving Time	*UTC +2*
Saudi Arabia	*UTC +3*
USA East coast	*UTC -5*
United Arabic Emirates	*UTC +4*

CONSEQUENCES TO THE CONFIGURATION

Engineered date and time mean different date and time depending on the execution location of the Runtime.

For example: In the Editor you engineer in the PFS for the execution of a function in time zone **UTC +1** the local time *14:00 o' clock*. After transferring the files to a Runtime in tiem zone **UTC +10**, the function is carried out at *23:00 o' clock*.

SWITCHING OF TIME ZONE

If the time zone is switched in the operating system, this change is automatically applied in zenon in the Runtime.

Hint

The time zone of the computer can be evaluated with the **[system information] summer/winter time** system driver variable.

The time difference of the time on the local computer to UTC can be evaluated with the **[system information] time offset to UTC** system driver variable.

16.1 Switch to daylight saving time

The following is applicable for the switch to daylight saving time or standard time:

- You must use the automatic Windows time amendment of the computer.
- Do not switch the time manually!

  Manual changes can cause problems with relevant times indicated for zenon, for example for save times of archives, time control, etc.

You configure the action of the **Scheduler** and **Production & Facility Scheduler** for time switching in the settings of this module:

- Scheduler time switching
- Production & Facility Scheduler time switching