zenon manual

Network

v.8.10

© 2019 Ing. Punzenberger COPA-DATA GmbH

All rights reserved.

Distribution and/or reproduction of this document or parts thereof in any form are permitted solely with the written permission of the company COPA-DATA. Technical data is only used for product description and are not guaranteed qualities in the legal sense. Subject to change, technical or otherwise.

Contents

1 Welcome to COPA-DATA help

ZENON VIDEO-TUTORIALS

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

GENERAL HELP

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

PROJECT SUPPORT

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

LICENSES AND MODULES

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

2 Network

zenon networks can be set up quickly and securely.

zenon in the network allows you to, among other things:

- full access to the Runtime of different computers
  This way, actions such as the acknowledgment of alarms at a workspace on all other computers in the network thus become visible.
- Centralized logging and archiving

- Creation of redundant systems (see redundancy (on page 77), circular redundancy (on page 102))
- Redundancy switching with integrated rating methods
- Creation of distributed systems (see multi-project administration (on page 30))
- Use of zenon Web Server and zenon Web Client for access using the web browser.
- Use of zenon in a terminal server environment (on page 61)
- Use of strong encryption (on page 47)
- Concurrent work on a project from several computers (see distributed engineering)

SIMPLE ADMINISTRATION OF THE ZENON NETWORK

The network functionality of zenon makes it possible to deploy projects in a distributed manner on different computers. You can thus create very efficient, complex network setups (on page 11) with it. In doing so, setups can also be configured in such a way that certain project content is only visible at a certain location (a certain computer) for an activity. The zenon Editor supports users in creating and configuring such configurations.

The integrated topology administration (on page 67) creates interrelationships for the individual projects in the process, with the attendant computers in graphical form. A testing routine checks the configured structure to see that it is complete and that there are no configuration errors.

With the network nodes function, zenon also checks to see if the selected network topology can work.

Information

With network projects, note the roles (on page 10) (Primary Server, Standby Server or Client) in which modules and functions (on page 148) can be administered and executed.

ZENON WEB SERVER

The zenon Web Server allows access to Runtime via a web browser. No adaptations to the project are necessary. Access is via the zenon Web Client. This offers the same functionality and display as an installed zenon Runtime. The zenon Web Server is available as:

- zenon Web Server: Pure monitoring functionality.
- zenon Web Server Pro: Complete operation and monitoring functionality. It is possible to directly engage in processes over the web.
- **zenon Web Server Pro Light**: Limited functionality of zenon Web Server Pro for use with zenon Operator.

Information

You can find further information in the zenon Web Server manual.

ACTION WITH LICENSED DATA CONCENTRATOR

With a Data Concentrator license, the connection of a client to a Primary Server or Standby Server is not possible! This licensing is primarily for zenon Analyzer use. A connection query from a client to a server is always rejected by the server. The values are not updated on the client. The configured standby server can connect to the server. This guarantees that redundancy also works with a Data Concentrator license.

RUNTIME COMPATIBILITY:

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 makes interoperability of Runtime systems (also via zenon Web Clients) in the zenon network possible even if the version of the client Runtime is higher than the version of the server Runtime.

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*
- *8.10 SP0*
- *7.11 SP0*
- *7.20 SP0*
- *7.20 SP0[latest build no.]*
- *7.50 SP0*
- *7.60 SP0*
- *8.00 SP0*
- *8.10 SP0*

Due to the multi-project administration, projects from different versions can be loaded. For example, the integration project can have version 8.00, a subproject version 7.60, and another subproject version 6.51.

3 Roles of computers

With zenon, it is possible to create diverse network topologies. Starting with the simple client-server model through to comprehensive multi-hierarchical models.

IT-specific terms such as Server and Client are also used in zenon. However, in order to achieve unique identification of the individual components for complex multi-hierarchical structures with various computers and projects involved, we always speak of roles in zenon. Roles are always to be considered from the point of view of a project.

zenon Runtime can, on one computer, start one or more projects, depending on the project configuration (see also multi-project administration (on page 30)). In doing so, the computer on which Runtime is started assumes one of the following roles for the respective projects:

- Primary Server
- Standby Server
- Client

These roles are shown below using examples in different topologies.

Info

If, in the course of this documentation, we speak of Primary Server, Standby Server or Client, the role that the computer takes for the project is always meant.

3.1 Terminology

The following terms are used for the role description in the zenon network:

Parameter	Description
Server:	Computer with connection to the PLC. The server takes on the administration of process and project data exclusively. Communication is checked by means of a watchdog (on page 18). In the event of a server failure, the Standby Server undertakes its tasks, provided a standby was defined. As soon as the server is ready again, it automatically takes on its tasks and synchronizes all data.
Standby Server:	Takes on, in redundant systems, the role of the server, if this fails. It acts like a client in the network, but al saves all data like the server. In the event of hardware redundancy, the standby communicates with the redundant PLC both ways. The standby works with an internal buffer. Data loss during the downtime between server failure and the standby taking on the server role is thus avoided.
Clients:	Each computer on which Runtime is started is a client. Clients connect to the server to receive process data or to send this.

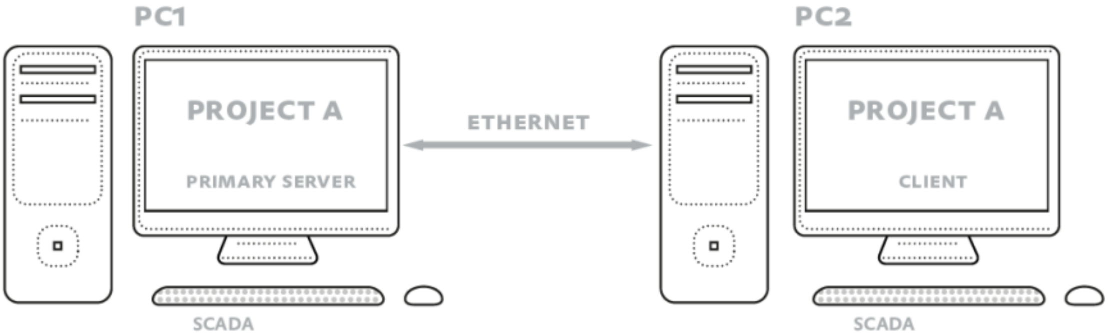
Information

Server and client are not defined in relation to a computer, but in relation to a project.

If the names of the Server or Standby Server are changed, these cannot be reloaded. They are only updated by restarting Runtime.

3.2 Client-server model

In the Client-Server model, one computer is the Primary Server; all other computers are Clients.



- Computer 1 is the Primary Server for Project A.
- Computer 2 is the Client for Project A.

> ### Information
>
> You can find detailed information on this in the Multi-Project Administration (on page 30) chapter.

3.3 Redundant model

In the redundant model, one computer is the Primary Server and one computer is the Standby Server. All other computers are Clients.
If the Primary Server fails, the Standby Server takes on this role. All Clients connect to the new Primary Server.

REDUNDANCY WITHOUT CLIENTS

- Computer 1 is the Primary Server for Project A.
- Computer 2 is the Standby Server for Project A.
- If Computer 1 fails, Computer 2 is the new Primary Server for Project A.

REDUNDANCY WITH CLIENTS

- Computer 1 is the Primary Server for Project A.
- Computer 2 is the Standby Server for Project A.

- If Computer 1 fails, Computer 2 is the new Primary Server for Project A. All Clients automatically connect to Computer 2.

> **Information**
>
> You can find detailed information on this in the Redundancy (on page 24) chapter.

3.4 Multi-hierarchical models

It is possible to create different multi-hierarchical topologies with the help of multi-project administration. In doing so, it is possible to start several projects on one computer. The computer assumes a certain role for the project in the process.

> **Information**
>
> Multi-hierarchical projects can also be executed without a network on standalone computers.

EXAMPLES

MULTI-CLIENT MODEL

- Computer 1 is the Primary Server for Project A.
- Computer 2 is the Primary Server for Project B.
- Project I runs on Computer 3 (integration project) as a standalone project and starts Projects A and B.
    - Computer 3 is a Client for both of these projects.

MULTI-SERVER MODEL

- Project I runs on Computer 1 (integration project) as a standalone project and starts Projects A and B.
    - Computer 1 is the Primary Server for both of these projects.
- Project I runs on Computer 2 (integration project) as a standalone project and starts Projects A and B.
    - Computer 2 is a Client for both of these projects.
- Computer 3 is the Client for Project A.
- Computer 4 is the Client for Project A.

MULTI-CLIENT-MULTI-SERVER MODEL

- Project I runs on Computer 1 (integration project) as a standalone project and starts Projects A, B, C and D.
  - Computer 1 is the Primary Server for projects A and B.
  - Computer 1 is the Client for projects C and D.
- Project I runs on Computer 2 (integration project) as a standalone project and starts Projects A, B, C and D.
  - Computer 2 is the Client for projects A and B.
  - Computer 2 is the Primary Server for projects C and D.
- Project I runs on Computer 3 (integration project) as a standalone project and starts Projects A, B, C and D.
  - Computer 3 is the Client for projects A, B, C and D.

> **Information**
>
> You can find detailed information on this in the Multi-Project Administration (on page 30) chapter.

4 Requirements

A basic requirement for using zenon is a functional Windows network.

GENERAL

The following prerequisites are necessary:

- TCP/IP as the network protocol
- Functional naming, can be chosen as **DNS**, **WINS** or local **HOST files**.
- Free TCP Port *1100*:
  If a network project is loaded, zenon Runtime automatically starts the **zenNetSrv** network service. This service opens port *1100*. This must therefore be contactable remotely and must not be blocked by a firewall.

> **Info**
>
> zenon networks work with all supported operating systems.

IPV4 AND IPV6

The zenon network allows the choice of using IPv4 or IPv6. Dual operation is not possible. The setting is made via:

- **Network configuration** in the **Startup Tool**
  or
- zenon6.ini

If this setting is changed, all ongoing zenon processes must be restarted.
The services affected are:

- **zenAdminSrv**
- **zenSysSrv**
- **zenLogSrv**
- **zenDBSrv**

The following components are not affected by the setting; they always use IPv4:

- Driver communication with the PLCs
- Protocol communication via process gateways
- Workbench and Runtime communication in zenon Logic

> ⚠ **Attention**
>
> IPv6 only works with zenon version 7.00 onwards. If IPv6 is used, no versions prior to zenon version 7.00 can be started.

PORTS USED

For communication within zenon, only TCP ports are used; no UDP ports are used.

The following ports are required for zenon in the network:

Service	File	Goal	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 by means of the **Listening ports** tab in the **Startup Tool**. In this case, the measuring range must be adapted manually!

Information

You can find further information in the Tools manual, in the **Startup Tool** chapter.

STANDARD PORTS

Application	Standard port
zenon	
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*
Logging Service	*50780*
zenVNC.exe	*5600 - 5610*
SNMP Trap Service	*50782*
WEB Service Tunneling	*8080*
zenon Logic	
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	
Administration Service	*50777*
Analyzer Connector Service	*50778*

Application	Standard port
Analyzer License Service	*50779*
ZAMS	*50781*
Drivers	
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	
Service Grid API	*9400*
Hub Controller	*9410*
Data Hub	*9411*
Configuration Backend	*9420*
Identity Service	*9430*
Policy Service	*9440*

CHECK THE REQUIREMENTS

NAME RESOLUTION

To check the name resolution:

1. Start the windows command line
   cmd.exe
2. Enter the following command:
   ping [COMPUTER NAME]
3. If the name resolution is correct, you receive the IP address of the computer with Runtime as the answer; otherwise you receive an error message

TCP PORTS

To check the contactability of the TCP port 1100:

1. Start the zenon Runtime with a network project on a Remote computer:
   This starts the program **zenNetSrv.exe** and the TCP port *1100* is opened.
2. Start the windows command line
   cmd.exe
3. Enter the following command:
   telnet RECHNERNAME 1100
4. As soon as a connection is established, the content of the command line window disappears. Otherwise an error message will be displayed.

> **Attention**
>
> The Telnet command is not part of the Windows operating system and must be installed separately. You can find instructions for this in the operating system help pages (search for: *Telnet*).

4.1 Time synchronization in the network

With network projects, all computers in the network must be time-synchronized. zenon automatically carries out the synchronization necessary for this.

In a topology with several Primary Servers (such as circular redundancy (on page 102)), it is recommended that time synchronization is implemented by means of an external time service (such as DCF77) or Windows resources. In this case, the automatic time synchronization in zenon must be deactivated.

> **Attention**
>
> If the time difference between the server and the client is more than 5 seconds, no more files are synchronized.

AUTOMATIC TIME SYNCHRONIZATION IN ZENON

If the time synchronization is to be turned on or off manually, the following entry in **zenon6.ini** must be amended:

[Netz]

```
TIMESYNCH=1 -> automatic time synchronization active (default)
TIMESYNCH=0 -> automatic time synchronization inactive
```

EXTERNAL TIME SYNCHRONIZATION USING THE OPERATING SYSTEM

If the automatic time synchronization in zenon is deactivated, synchronization can be carried out via the operating system. To do this, a time server must be specified for this (with or without an external time service such as DCF77), which takes on the time synchronization with the other computers.

In the conventional Client-Server/Standby topology (without multi-project administration), the Primary Server is the active time master. This should keep its own time itself by means of an external time service if possible. The respective clients get the current time from this (depending on the timeout that has been set) and update their own times accordingly. Communication is via SNTP (*Simple Network Time Protokoll*). The delay in transfer is taken into account in the process.

Information

Watchdog

Time synchronization is carried out periodically at the set timeout time.

When using the default setting of 30 seconds for the **Network communication timeout** property in the **Startup Tool**, the network service (**zenNetSrv.exe**) of each client sends a Watchdog to the network service (**zenNetSrv.exe**) of the Primary Server every 10 seconds during online operation. If the Primary Server responds to at least one of the three watchdogs within the 30 seconds, the client assumes that the network connection is working.

Configuration in the Startup Tool:

Application -> Options-> Network configuration tab **-> Option Networkcommunication timeout**.

Configuration in zenon6.ini:

Alternatively, the setting can be made directly in zenon6.ini:

[Netz]

NET_TIMEOUT_MSEC=*30000*

(timeout in milliseconds, Default: *30000*.)

Note the additional configuration necessary in WAN (on page 21).

Attention: The minimum timeout time is 5 seconds. If a lower value is defined, this is interpreted as 5 seconds.

COMMANDS UNDER WINDOWS

For external synchronization using Windows, enter the following command with the respective necessary arguments in the console for command processing: **NET TIME [\\computer name | /DOMAIN[:domain name] : /RTSDOMAIN[:domain name]] [/SET] [/YES]**

Argument	Description
NET TIME	▸ Synchronizes the time of the computer with that of another computer or another domain or ▸ Displays the time for a computer or a domain If this command is executed without further arguments, then the current date and the current time of the computer that was defined as the time server for the domains is displayed.
Computer name	The name of the computer that checks or is to be synchronized.
DOMAIN[:Domain name]	The time is synchronized with the primary domain controller of the **Domain name** domain.
RTSDOMAIN[:Domain name]	The time of the computer is synchronized with a reliable time server from the **Domain name** domain.
/SET	Synchronizes the clock of the computer with the stated computer or the stated domain. After the command has been set, the server time is displayed and a request is made to see if this time is to be set.
/YES	Displays the current server time and synchronizes this with the local computer without a further request or confirmation.

Example

NET TIME \\Server /SET /YES

4.1.1 Time synchronization in the WAN

In a WAN and for dial-up connection, the standard defined value of 30 seconds for the timeout means that the connection would probably be maintained permanently.

Select a timeout time in the WAN that only initiates the establishment of a connection at the desired interval.

However, note: The longer the timeout, the later server failures are detected. For example, if you select *64800* as the time for the timeout, the timeout time is 18 hours. A connection is made every 6 hours and a watchdog is sent. A server failure would thus only be noticed after around 18 hours.

> **Information**
>
> If no entry for the timeout is defined in **zenon6.ini**, the standard timeout of 30 seconds is used when Runtime is started.

FUNCTION SCREEN SWITCH

Active data is requested when a screen is switched. Procedure:

- A check is made to see if a watchdog was sent to the Primary Server in the last 30 seconds.
- If this is not the case, a watchdog is sent to the Primary Server immediately. the waiting time for a response is 40 seconds.
- If a Server break down is recognized, the zenon network service automatically tries to reconnect every 30 seconds.

This would lead to a permanent connection establishment in the WAN network. This behavior can lead to entries in **zenon6.ini** being amended:

1. Open zenon6.ini.
2. Go to the section
   [NETZ]
3. Create or edit the entry

   NET_CONNECTWAIT_MSEC=30000
   This defines the value for a reconnect in milliseconds.

   Maximum value: Timeout time
4. Create or edit the entry

   NET_CONNECTCOUNT=

   This defines the number of repetitions for a reconnect per cycle.

   The default is *0* repetitions, this means 1 attempt at reconnection.

5 zenon network - Setting up

TOPOLOGIES

zenon supports several network topologies:

- Client server network (on page 23):
  The same project runs on the server and all clients.
- Multi-server network:
  A client can access different servers and thus display the data of different projects at the same time.
- Multi-client-multi-server model:
  All clients and servers communicate with each other. Other projects can be accessed from each project.

CONFIGURING THE NETWORK

To make a network network-compatible:

1. navigate to the **Network** node in properties
2. Activate the property **Network active**
3. Use the **Server 1** property to define the computer that takes on the server role in the project
   Note: The IP address is not sufficient; the name of the computer must be entered.

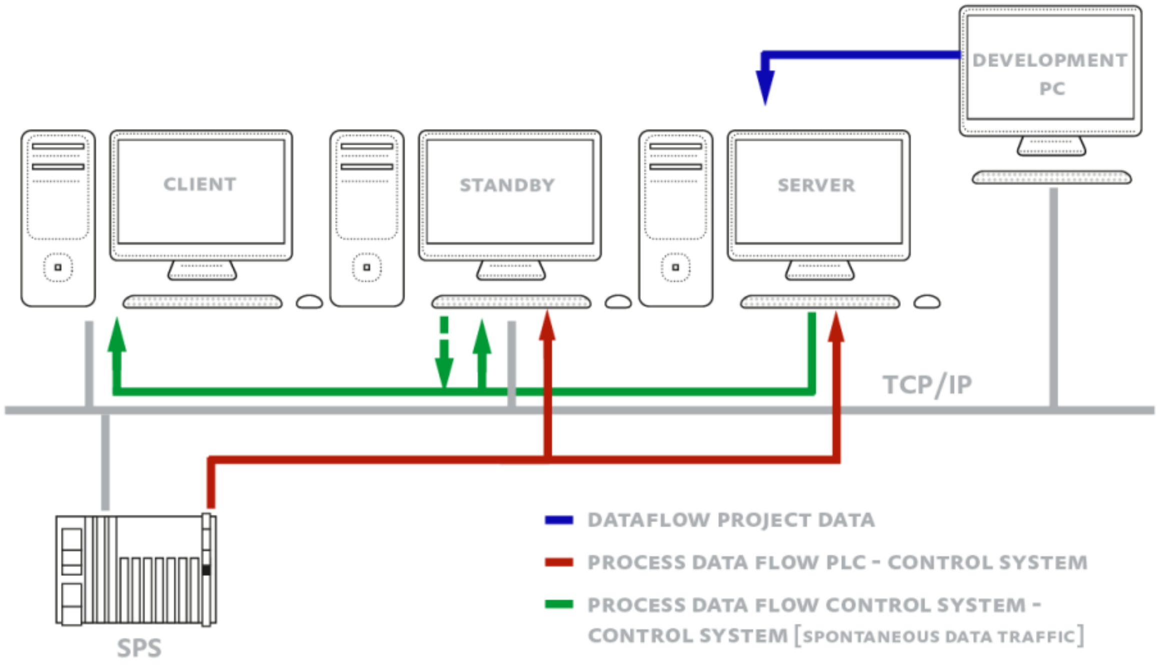
If necessary, you still configure the following in this section:

- Standby Server (on page 98): **Server 2** property:
- Redundancy (on page 77): **Redundancy type** property:
- **Termination message**: Defines if, when Runtime is ended on a server, the clients are informed 70 seconds in advance.
- Operating authorization (on page 133): If active operations are to be executed at the same time on just one station.

> ### Attention
>
> Issue of a name for **Server 1** and **Server 2**:
>
> - The computer name must be entered.
>   The entry of an IP address is not permitted.
> - *localhost* must not be used.

5.1 Client-server model

In the conventional Client-Server model, only one project is used, which is started on all computers involved. In doing so, a certain computer is the Primary Server for this project. All other connected computers are Clients.


DEVELOPMENT PC
CLIENT
STANDBY
SERVER
TCP/IP
SPS
DATAFLOW PROJECT DATA
PROCESS DATA FLOW PLC - CONTROL SYSTEM
PROCESS DATA FLOW CONTROL SYSTEM - CONTROL SYSTEM [SPONTANEOUS DATA TRAFFIC]

To set up the Client-Server model, the following must be the case in the project:

- The **Network active** property must be activated
- The name of the computer that is to be the Primary Server is to be entered in the **Server 1** property

Recommendation: Select the most powerful computer in the network as the Primary Server.

In the zenon Client-Server model:

- Only the Primary Server has a direct connection to the PLC
- The Primary Server administers all process data (such as online data, archive data, alarms, recipes, etc.)
- The Primary Server administers all project data (such as screens, functions, defined variables, etc.)
- Each other computer starts as a Client in the network
- Each client establishes the connection to the Primary Server when Runtime is started,

- synchronizes the project data and
- displays the current process data

> **Information**
>
> The Client-Server model is fully supported under Windows CE. Windows CE devices can be used as Primary Servers or as a Client.

5.1.1 Redundancy

Redundant SCADA servers are used, if 100% process control and data safety is demanded, even when the server fails.

You achieve this fail safety by defining a second server, a so-called Standby Server, along with the project server. This standby automatically recognized a server failure If the Primary Server fails, the Standby Server takes on the complete functionality of the server.

In order to avoid data loss in the time between the server failure and the detection of the failure, the standby always buffers all data. This data buffering also takes place if the Standby Server is not the Primary Server. After a server failure this buffer is merged with the last data from the server and the new incoming data, so no data can be lost. The control system thus guarantees seamless redundancy.

Depending on the configuration of the **Redundancy mode** property, the original server is started either as a Server or as a Standby after restarting.

Redundancy mode	Description
Non-dominant	The original Server (**Server 1**) starts as a Standby. **Server 2** retains the
Dominant	The original Server (**Server 1**) takes on the role of Server again after a been synced. **Server 2** is downgraded to the Standby.
Rated	Depending on the configured rating (on page 87), the original server either as a Server or as a Standby after restarting.

You can find detailed information on this in the Redundancy modes (on page 84) chapter in this manual.

> **Info**
>
> If the standby is running, when the server is started, the server copies all Runtime data from the standby. If you made any changes in the project data, while the server was offline and you have only updated them on the (not running) server, these changes will be overwritten, when the server copies the data from the standby.
>
> In this case you have to update the standby before starting the server; or you close the standby when starting the server. As soon as it is restarted the standby then will get the new data from the (running) server.

CONFIGURING THE STANDBY

- In the network properties, enter, in addition to the **Server** (property: **Server 1**) the name of he **standby server** too (property: **Server 2**).
- To transfer the Runtime files, the process is the same as with the client.

The control system supports two different kinds of redundancy:

Software redundancy	The PLC is not redundant. Only the control system is redundant. (standard case)
Hardware redundancy	Both the PLC and the control system are redundant.

You can find detailed information on this in the Redundancy (on page 77) chapter in this manual.

5.1.2 Configuring the server

The Primary Server makes the connection to the PLC and administers all data, both online data and configuration data. The Clients synchronize their data with the Primary Server.

To set up the Primary Server

1. Activate the **Network active** property.
2. Define the computer that is to be the Primary Server for the project using the **Server 1** property.
   Note: The IP address must not be used; the name of the computer must be entered.
3. Note the correct configuration of the internal variable drivers (on page 166)
4. Optional: Create AUTOSTART and AUTOEND scripts (on page 165) for the clients if necessary

> **Attention**
>
> Issue of a name for **Server 1** and **Server 2**:
>
> - The computer name must be entered.
>   The entry of an IP address is not permitted.
> - *localhost* must not be used.

- If the development computer on which you created the project is also the Primary Server, configuration is now complete.

If the development computer is not the desired server, transfer your project configuration to the desired computer. The data can be transferred by means of Remote Transport or the network topology.

> **Information**
>
> You can find further information on this in the administer and check network topology (on page 67) chapter or in the Remote Transport manual.

5.1.3 Configuring the clients

Clients can be set up via Remote Transport, via the network topology or manually. Setup via Remote Transport or the network topology is recommended.

For this, the following applies:

- If the development computer is also is a client, start Runtime locally on that computer.
- You set up all other clients either by means of Remote Transport (on page 27), network topology (on page 67) or manually (on page 27).
- If special processes are to be executed on the clients, a respective script in the project must be created, which defines the behavior on startup (AUTOSTART_CLIENT script (on page 165)) and when being ended (AUTOEND_CLIENT script (on page 165)).

> **Information**
>
> You can find further information on this in the administer and check network topology (on page 67) chapter or in the Remote Transport manual.

5.1.3.1 Set up client with Remote Transport

By default, Remote Transport always transports the Runtime files to the computer that is defined as the server in the **Network** project properties group. To set up clients from the development computer by Remote Transport, the Remote Transport connection must be set up before the Client is set up.

To set up clients using Remote Transport:

1. Open the **General** node in Project Properties.
2. Click on the **Remote Transport** property.
   The dialog **Remote Transport** is opened.
3. Enter the name of the client in the network in **Connection** under **Name**.
4. Confirm the configuration by clicking on the **OK** button.
5. Establish a connection to Remote Transport.
   To continue to use Remote Transport, it is best to use the toolbar symbols.
6. Transport all Runtime files to the client with Remote Transport.
7. Set the start project for the client with Remote Transport.
8. Start the Runtime on the client with Remote Transport.
9. Close the Remote Transport connection

> **Information**
>
> Find out more information in the chapter Remote Transport.

5.1.3.2 Setting up a Client by means of network topology

Network topology is suitable for setting up several Clients at the same time. In doing so, Runtime files can be transferred to several computers at the same time by means of multiple selection.

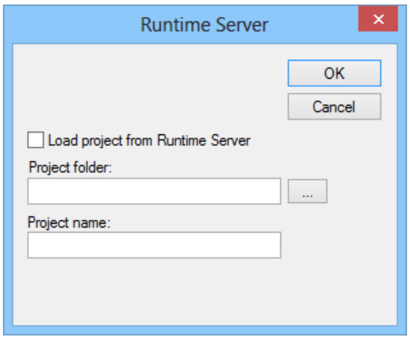
> **Information**
>
> You can find detailed information on this in the Network topology (on page 67) chapter.

5.1.3.3 Setting up the client manually

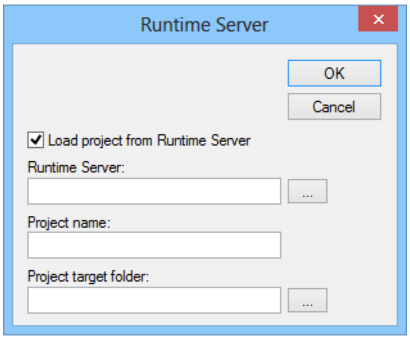
To configure clients for the start of Runtime:

1. Close the zenon Editor and the zenon Runtime.
2. Open the **zenon6.ini** file with a text editor.
   You can find the file in the *%ProgramData%\COPA-DATA\System\* folder
3. Delete or comment out the **VBF30 =....** line.

   (**Note:** This entry defines which project is to be loaded when the Runtime is started.)
4. Leave the Editor closed and start Runtime.
5. A request is made in a dialog, requesting which project is to be loaded.



6. Activate the check box for the **Load project from Runtime server** option.



7. Configure the input field:
   a) **Runtime server**:
   Computer that is set up as the Primary Server (on page 11). The name can be entered directly or selected from a list using the **...** button.
   You can find further information on configuration for a redundant network after these step-by-step instructions

   b) **Project name**:
   Name of the project that runs on the Primary Server.
   Note: Ensure that the project names are correct. If the folder name does not correspond to the project name, the project name must be changed here!

   c) **Project target folder**:
   Folder for Runtime on the client's local hard drive. You can either select an existing folder using the **...** button or enter a path manually. If a folder that does not exist is entered, this is created automatically.

d) Confirm the input by clicking **OK**.

8. These project configurations have the following effect in zenon Runtime.
Runtime:

a) Creates a connection to the Primary Server

b) Copies its Runtime files to a project target folder

c) Starts Runtime

d) Requires - if necessary - Runtime to be restarted
Note: The check box of the **Load project from Runtime server** property must be deactivated before restarting the Runtime. Take the project path previously selected under the **Project target folder** and enter it under the **Project folder**. Enter the previously defined project name under **Project name**.

9. In the **VBF30=...** entry in the *zenon6.ini* file, you set the project target folder.

Runtime then starts the network project automatically on the client each time it is started.

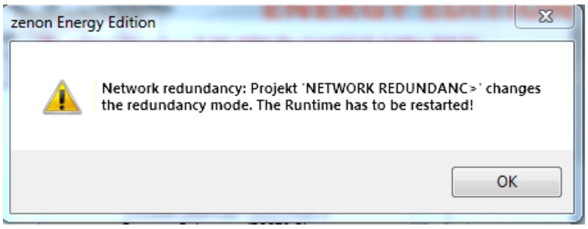
> **Attention**
>
> Repeat this process for each client.

CONFIGURATION OF THE RUNTIME SERVER IN THE REDUNDANT NETWORK

In the event of a redundant zenon network configuration, the issuing of roles for the primary server and the standby server depends on the **Redundancy mode** set. In doing so, the role of the configured **Server 1** and **Server 2** computers can change over time in the Runtime, depending on the configuration of the **Redundancy mode** and the current evaluation (for **Redundancy mode** *evaluated*).

You should therefore always enter both servers for the configuration in the **Runtime Server** dialog. The server names are separated from one another by a semicolon (;). Empty spaces for the server names are permitted.

The sequence of the servers entered corresponds to the sequence of configuration. The Runtime client does not start if the computer names do not correspond.



Network redundancy: Projekt 'NETWORK REDUNDANC>' changes the redundancy mode. The Runtime has to be restarted!

If the project configuration of **Server 1** or **Server 2** is changed, this change is discovered when reloading on the client. In Runtime, a dialog is displayed that informs you of the amended server configuration and forces the client to be restarted.

Ensure that the new Runtime files are transferred to the client again if necessary.

5.1.3.4 Behavior in Runtime

Network projects can be operated in the same way in the network by all computers in the Runtime and are visualized in the same way. If there is no valid project defined when Runtime is started, the dialog to define the Runtime projects is opened. For details, see the Set up client manually (on page 27) section.

Differences between the Primary Server and the Client:

- Only the Primary Server of the project has a connection to the hardware and administers the process data.
- The other computers (Clients) receive, from this:
  - Current values of the variables
  - Chronological Event List system messages
  - Alarm information
  - Recipes
  - Archive data
  - etc.

The transfer from the Primary Server to the Clients (such as a value change from a driver) is spontaneous and event-triggered (such as calling up a zenon *Extended Trend* screen, which needs archive data from a Primary Server).

MONITORING THE CONNECTION

When using the default setting of 30 seconds for the **Network communication timeout** property in the **Startup Tool**, the network service (**zenNetSrv.exe**) of each client sends a Watchdog to the network service (**zenNetSrv.exe**) of the Primary Server every 10 seconds during online operation. If the Primary Server responds to at least one of the three watchdogs within the 30 seconds, the client assumes that the network connection is working.

5.2 Multi-Project Administration

Multi-project administration makes decentralized solutions possible. Subprojects can be distributed to different computers. The individual computers in turn can be the Primary Server, Standby Server or Client for the respective subprojects.

The following is possible with the help of multi-project administration:

- Several projects in one workspace can be edited in the Editor at the same time

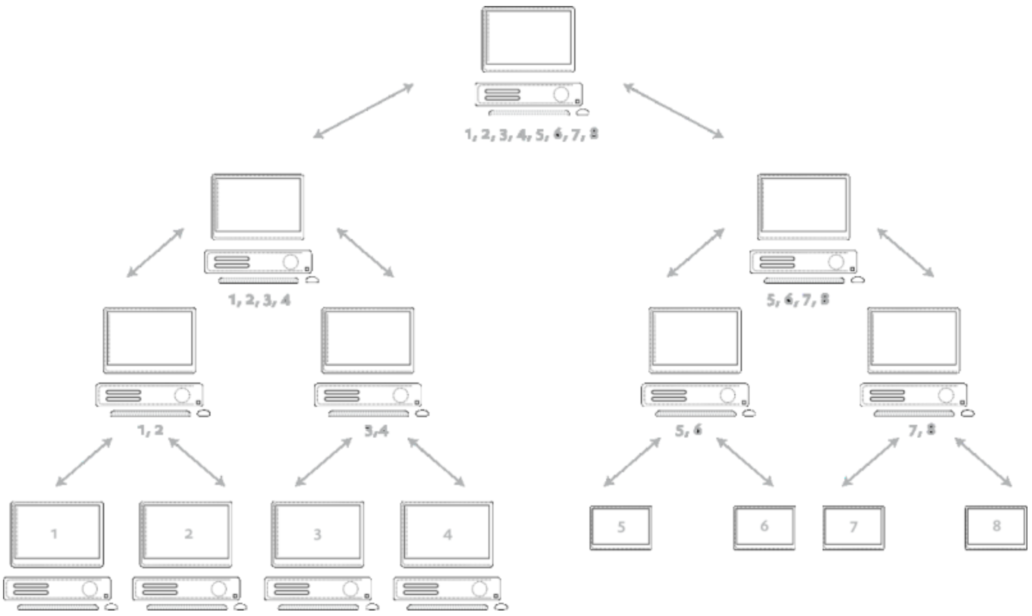
- Several projects can be started in Runtime at the same time and thus variables, functions, archives etc. from other projects can be accessed directly throughout the projects

> **Information**
>
> Multi-project administration is not available under zenon Operator. Here, only one project and a global project per workspace can be created and administered in the Editor. Runtime can only start one project.

STRUCTURE

An integration project that is loaded in Runtime as a start project is required.

zenon creates a multi-hierarchical project tree, at the top of which is the integration project. Multi-project administration makes it possible to place the projects in a logical connection to one another.



> **Information**
>
> Configure and check the topology with the zenon network topology (on page 67).

WORK EFFICIENTLY WITH MULTI-PROJECT ADMINISTRATION AND THE PROJECT HIERARCHY

zenon enables you to reuse data and screens from existing projects consistently. zenon multi-project administration makes a logical connection between the individual projects and places these in a hierarchical connection to one another. The user can display this project hierarchy graphically in the zenon Editor by dragging the projects to the desired position with the mouse and thus creating a multi-hierarchical project tree.

The project that is highest in the hierarchy is the integration project. All other projects are subordinate to this project. The data from individual projects is available throughout all projects in the project structure.

The zenon multi-project structure is comparable to a file folder:

- Additional sheets – zenon projects – can be added at any time. The folder always automatically covers all information of the sheets stored in there. It is possible to browse through the pages at any time and look at the information without taking the individual pages out. In the zenon multi-project structure, users can change between the individual screens or projects without having to take these out.
- The integration project can be compared to the contents of the file folder. It serves as a central navigation project and makes it possible to display screens or data from the subordinate projects. The individual projects are autonomous and can continue to be operated autonomously. Access from a project to the data or screens of another project is enabled via the zenon standard interfaces. Expansions or amendments to projects are made directly in the individual projects. Any maintenance work that may be carried out only has an influence on the respective project; the overall system remains unaffected by this.

Note: Please pay attention to the recommendations for monitor configuration during configuration.

MULTI-PROJECT ADMINISTRATION MEANS

- Small-sized, clear structures.
- Easy, quick and clear maintenance of the individual projects.
  It is possible, for example, to deactivate individual projects without influencing the others. In the same way, projects can be distributed to different processors.
- Load distribution.
- Cross-project operation, as all projects on a processor are simultaneously active.
- Multiple-hierarchy network structure allows the centralization of data (measured values, alarms, plant information, archive data, etc.) in a higher-ranking project.
- No limit on the number of projects per processor.
- Summary of projects to large control rooms.
- Node structure – physical network separation.

5.2.1 Definition of the structure in the Editor

The structure is created by simply dragging & dropping in the Editor. You also need, in addition to productive projects, an integration project that administers all other projects. Because standalone projects do not send data to other computers, a Primary Server must be defined in each (sub-)project. The integration project itself can also be a network-capable productive project.

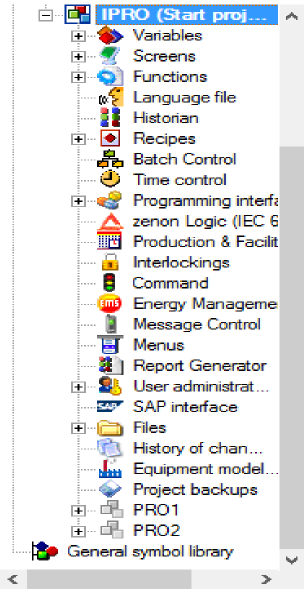
EXAMPLE

Three projects are used in this example:

- Integration project *IPRO*
- Productive project *PRO1*
- Productive project *PRO2*

To create the structure:

1. Create three projects:
   - *IPRO*
   - *PRO1*
   - *PRO2*
2. Define a Primary Server for each computer
   (The integration project can also be implemented as a standalone project)
3. In the Project Manager, drag *PRO1* by holding the left mouse button to *IPRO*
4. Do the same for *PRO2*
5. *PRO1* and *PRO2* are now displayed in the Project Manager as branches of the *IPRO*



The hierarchical structure of the network has thus been created.

> **Information**
>
> In order for elements of sub projects, such as screens, variables or functions to be able to be selected, the **Keep project in the memory** function (in the project context menu) must be activated.

5.2.2 Transferring and starting projects

With the help of the network topology (on page 67), the integration project can automatically be transferred to the respective target computer with all sub projects. All sub projects are also automatically transferred if the integration project is also the start project.

TRANSFERRING AND STARTING PROJECTS MANUALLY

Remote Transport can be used to transfer and start projects manually.
Each project is individually transferred to the corresponding computer.

- With Remote Transport, move all Runtime files of PRO1 to its Primary Server.
- Set the start project with Remote Transport.
- Start the Runtime with Remote Transport.
- Stop the online connection.
- Do the same for *PRO2*.

> **Info**
>
> In order to transfer the integration project and both sub projects to a Client via Remote Transport, 3 Remote Transport processes are necessary (a process for each project).
>
> Network topology (on page 67) is preferred here, because all projects can also be transferred to several computers at the same time.

ADDITIONAL INFORMATION

- For details of network topology, read the Administering and checking network topology (on page 67) chapter.
- You can find the configuration of the computer with an example for automatic transfer of sub projects in the Configuration of computers in the network (on page 73) section.
- You can find details for data transfer in the Remote Transport manual.

5.2.3 Administering projects

You have the following possibilities, among others, for accessing the data from sub projects:

- Integration project (on page 35)
- Navigation between projects (on page 36)
- Using variables or functions from another project (on page 37)
- Sending recipes to different variables in different projects (on page 41)
- Create archives for use throughout projects (on page 39)
- Creating a joint AML or CEL for different projects (on page 42)

> ### Attention
>
> During configuration, note the roles in which the modules and functions are executed (Primary Server, Standby Server, Client). You can find a list of the possible configurations here: Behavior of modules in the network (on page 148).

5.2.3.1 The integration project

The integration project administers subprojects that can be accessed in Runtime. The integration project can be used as a pure administration project in multi-project administration (for example just for the navigation to the sub projects) or also as a complete productive project (with its own PLC connection, archiving, etc.). If the integration project is used as a start project, all sub projects are automatically started in Runtime.

In an integration project, you can for example create central Alarm Message Lists or Chronological Event lists for all subordinate projects with a few mouse clicks. This allows the alarms of all sub projects in the Alarm Message List of the integration project to be displayed in chronological order.

> ### Attention
>
> When designing the multi-project administration, ensure that the navigation (on page 36) works.

> ### Information
>
> These must be editable in order to be able to delete subprojects. For example, a subproject that has been created with an earlier editor version than the integration project can only be deleted after conversion.

5.2.3.1.1 Navigation between projects

When administering more than one subproject in an integration project, it is absolutely necessary to ensure that it is possible to switch in the Runtime from one subproject to another or to the integration project.
Hint: Create a frame that is always in the foreground. Create a screen with navigation buttons based on this.

SCREEN SWITCH TO SUBPROJECTS

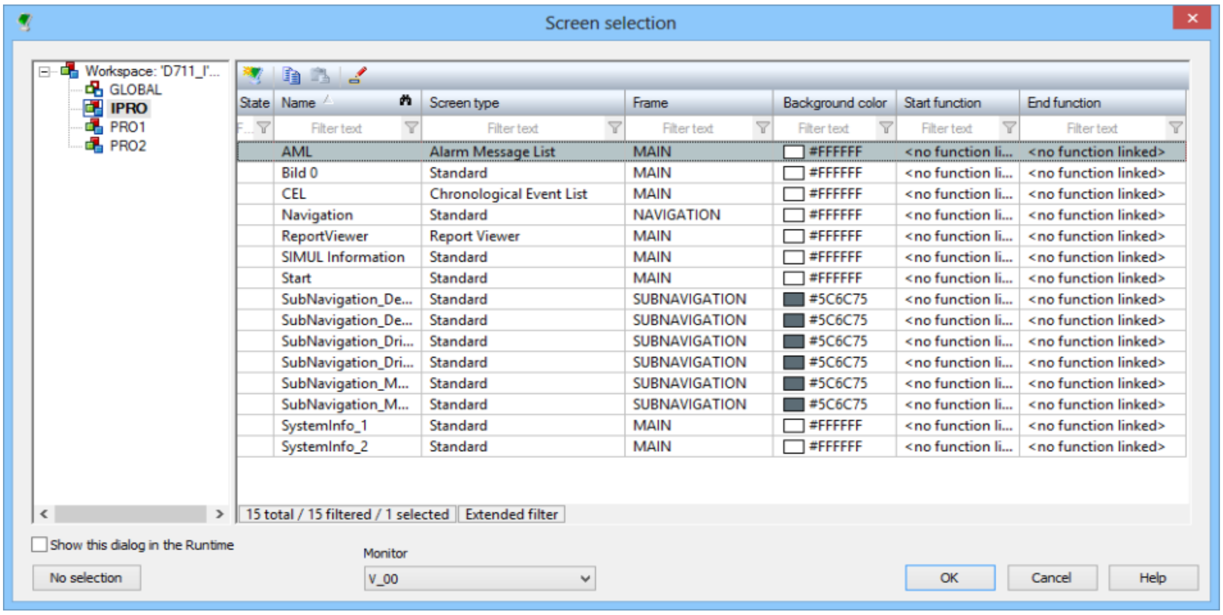
To switch between screens of individual projects, use the zenon screen switching function. In order for the navigation to be available at all times, first create a frame that is always in the foreground:

1. Create a new template that offers space for the navigation controls.
2. Assign it the **Always in the foreground** property.
3. Activate the **Border type** and **Title** properties (this means that the template can be moved in the Runtime).
   Of course, a template can also be used without a frame and title being in a fixed position.
4. Create a screen with navigation buttons on the basis of this template

EXAMPLE OF SWITCHING BETWEEN PRO1 AND PRO2

1. Create a new screen switch function.

2. If there is more than one project available in the current workspace, the dialog to select a screen for the selection of a project is expanded.



Screen selection

Workspace: 'D711_I'...
GLOBAL
IPRO
PRO1
PRO2

State	Name	Screen type	Frame	Background color	Start function	End function
F...	Filter text	Filter text	Filter text	Filter text	Filter text	Filter text
	AML	Alarm Message List	MAIN	□ #FFFFFF	<no function li...	<no function linked>
	Bild 0	Standard	MAIN	□ #FFFFFF	<no function li...	<no function linked>
	CEL	Chronological Event List	MAIN	□ #FFFFFF	<no function li...	<no function linked>
	Navigation	Standard	NAVIGATION	□ #FFFFFF	<no function li...	<no function linked>
	ReportViewer	Report Viewer	MAIN	□ #FFFFFF	<no function li...	<no function linked>
	SIMUL Information	Standard	MAIN	□ #FFFFFF	<no function li...	<no function linked>
	Start	Standard	MAIN	□ #FFFFFF	<no function li...	<no function linked>
	SubNavigation_De...	Standard	SUBNAVIGATION	■ #5C6C75	<no function li...	<no function linked>
	SubNavigation_De...	Standard	SUBNAVIGATION	■ #5C6C75	<no function li...	<no function linked>
	SubNavigation_Dri...	Standard	SUBNAVIGATION	■ #5C6C75	<no function li...	<no function linked>
	SubNavigation_Dri...	Standard	SUBNAVIGATION	■ #5C6C75	<no function li...	<no function linked>
	SubNavigation_M...	Standard	SUBNAVIGATION	■ #5C6C75	<no function li...	<no function linked>
	SubNavigation_M...	Standard	SUBNAVIGATION	■ #5C6C75	<no function li...	<no function linked>
	SystemInfo_1	Standard	MAIN	□ #FFFFFF	<no function li...	<no function linked>
	SystemInfo_2	Standard	MAIN	□ #FFFFFF	<no function li...	<no function linked>

15 total / 15 filtered / 1 selected | Extended filter

□ Show this dialog in the Runtime

Monitor: V_00

3. Select *PRO1*.
4. Select the start screen of *PRO1* and close the dialog with **OK**.
5. Repeat points 1 to 4 for *PRO2*.
6. Add two text buttons with the text *PRO1* and *PRO2* to the navigation screen.
7. Link the two text buttons to each of the functions that have been created.

Attention

zenon does not check in the Editor to see if the network structure in Runtime actually allows access to the selected project/screen.

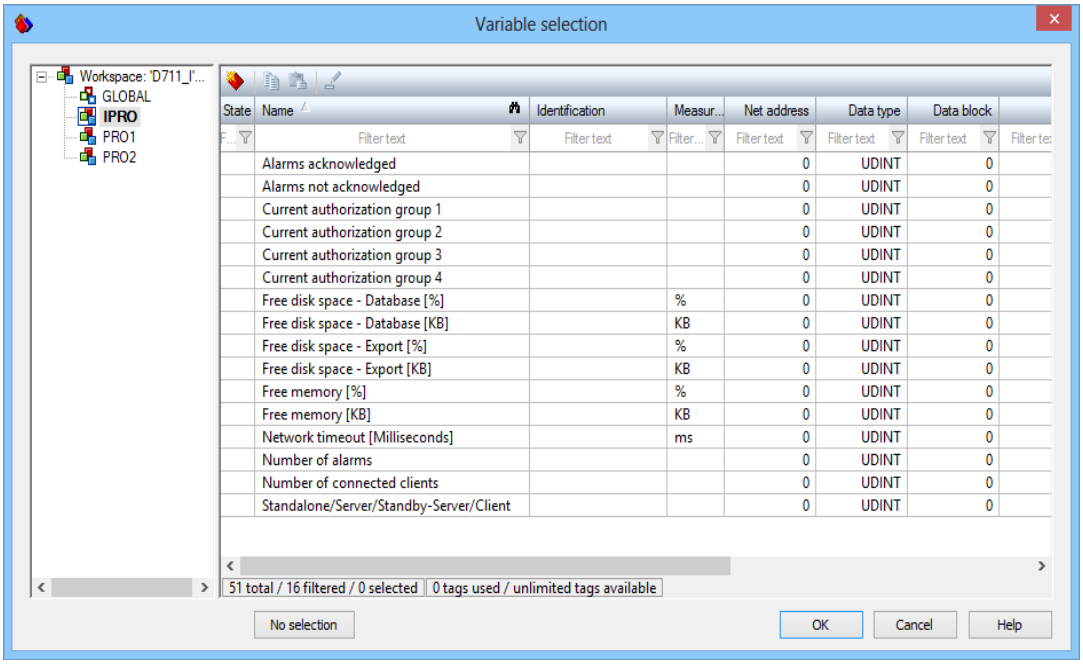
For example, in the Editor, screen switching to a screen in the integration project can be created in the *PRO1* project. This switching only works in Runtime if the integration project has also been started. This screen switching will not work on a computer on which the *PRO1* project (start project) has been started.

5.2.3.1.2 Variables and functions

You can access variables and functions from other projects in the same workspace directly by means of **dynamic elements**.

VARIABLE EXAMPLE

1. Open the start screen of the *IPRO*.
2. Add a new **counter value** dynamic element.
3. The variable selection dialog now opens.



State	Name	Identification	Measur...	Net address	Data type	Data block
F...	Filter text	Filter text	Filter...	Filter text	Filter text	Filter text
	Alarms acknowledged			0	UDINT	0
	Alarms not acknowledged			0	UDINT	0
	Current authorization group 1			0	UDINT	0
	Current authorization group 2			0	UDINT	0
	Current authorization group 3			0	UDINT	0
	Current authorization group 4			0	UDINT	0
	Free disk space - Database [%]		%	0	UDINT	0
	Free disk space - Database [KB]		KB	0	UDINT	0
	Free disk space - Export [%]		%	0	UDINT	0
	Free disk space - Export [KB]		KB	0	UDINT	0
	Free memory [%]		%	0	UDINT	0
	Free memory [KB]		KB	0	UDINT	0
	Network timeout [Milliseconds]		ms	0	UDINT	0
	Number of alarms			0	UDINT	0
	Number of connected clients			0	UDINT	0
	Standalone/Server/Standby-Server/Client			0	UDINT	0

51 total / 16 filtered / 0 selected | 0 tags used / unlimited tags available

4. Here, you can select not just variables from the *IPRO*. To select a variable from a different project:

   a) In the left list area of the project, select a project from the tree view of the workspace. The variables of the selected project are shown in the main area.

   b) Select the desired variable with a mouse click.

5. Select a variable from *PRO1* or *PRO2*.

Proceed in the same way for functions.

Attention

zenon does not check in the Editor to see if the network structure in Runtime actually allows access to the selected project and its variables/functions.

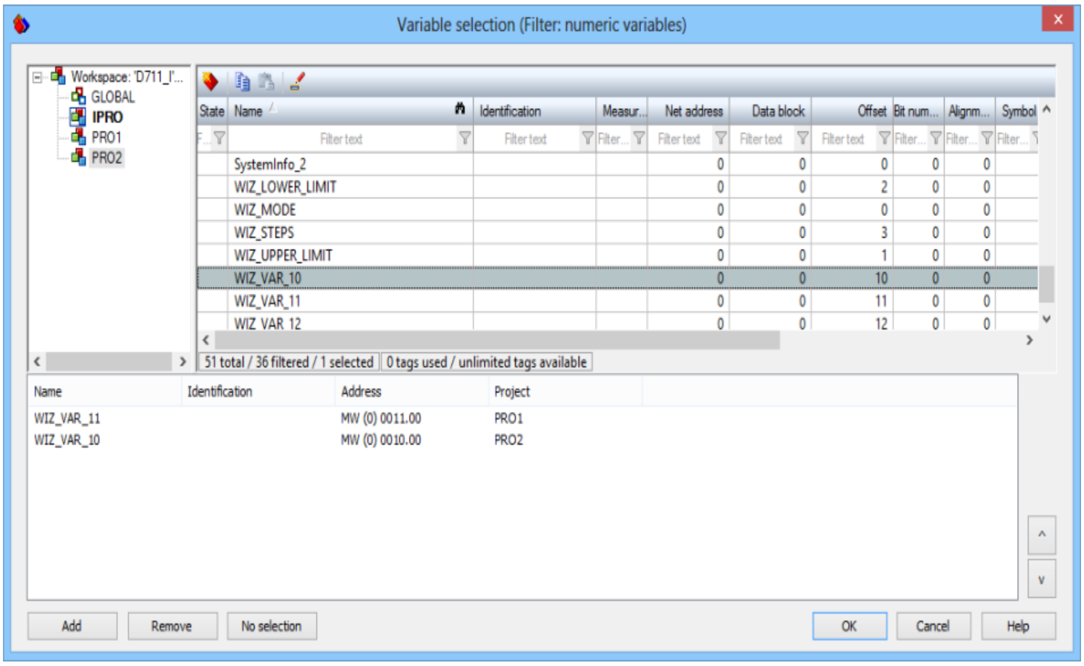
For example, in the Editor, in project *PRO1*, a variable from the integration project can be selected. This connection only works in Runtime if the integration project has also been started. This connection will not work on a computer on which only project *PRO1* has been started (start project).

5.2.3.1.3 Archives

Values of variables of different projects of the workspace can be recorded in an archive. The values recorded in this way can be filtered, displayed in list form or trend form, and they can be printed or exported just like data from normal archives.

EXAMPLE OF ARCHIVE

1. In the project *IPRO* open the node **Historian**.
2. Create a new archive named *BA - BASIS*.
3. Open the context menu of RECIPE1 and select **Add variable**.
4. The dialog for selecting variables is opened



Variable selection (Filter: numeric variables)

Workspace: 'D711_I'...
GLOBAL
IPRO
PRO1
PRO2

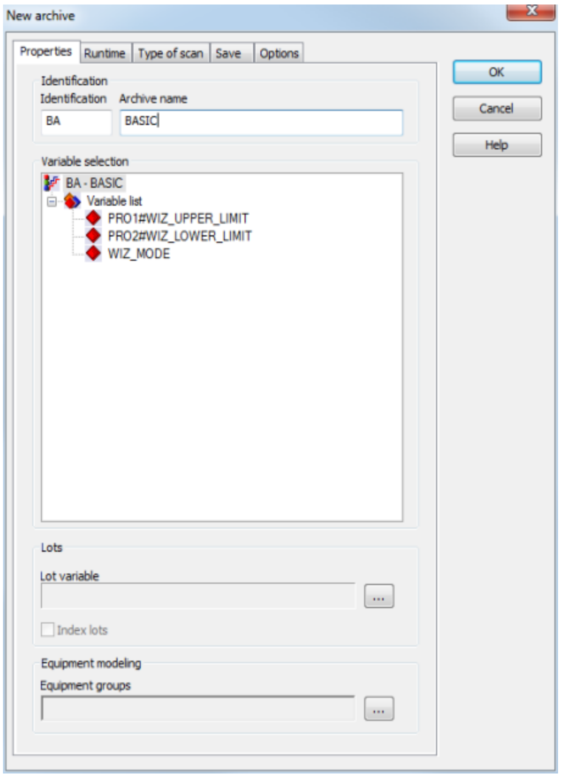
State	Name	Identification	Measur...	Net address	Data block	Offset	Bit num...	Alignm...	Symbol
	SystemInfo_2			0	0	0	0	0	
	WIZ_LOWER_LIMIT			0	0	2	0	0	
	WIZ_MODE			0	0	0	0	0	
	WIZ_STEPS			0	0	3	0	0	
	WIZ_UPPER_LIMIT			0	0	1	0	0	
	WIZ_VAR_10			0	0	10	0	0	
	WIZ_VAR_11			0	0	11	0	0	
	WIZ_VAR_12			0	0	12	0	0	

51 total / 36 filtered / 1 selected | 0 tags used / unlimited tags available

Name	Identification	Address	Project
WIZ_VAR_11		MW (0) 0011.00	PRO1
WIZ_VAR_10		MW (0) 0010.00	PRO2

5. Here, you can select not just variables from the *IPRO*. To select variables from other projects:
   a) In the left list area of the project, select a project from the tree view of the workspace. The variables of the selected project are shown in the main area.
   b) Select the desired variable with a mouse click.
6. Select variables from *PRO1* and *PRO2*.

7. The project name is written in front of the variable name in the archive variable list.



New archive

Identification: BA
Archive name: BASIC

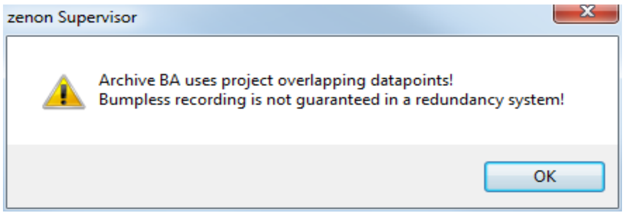
Variable selection:
- BA - BASIC
  - Variable list
    - PRO1#WIZ_UPPER_LIMIT
    - PRO2#WIZ_LOWER_LIMIT
    - WIZ_MODE

Attention

zenon does not check in the Editor to see if the network structure in Runtime actually allows access to the selected project and its variables.

For example, in the Editor, in project **PRO1**, a variable from the integration project can be selected. This connection only works in Runtime if the integration project has also been started. This connection will not work on a computer on which only project **PRO1** has been started (start project).

After the selection of variables has been concluded, a warning dialog indicates that seamless recording is guaranteed under all circumstances.



zenon Supervisor

Archive BA uses project overlapping datapoints!
Bumpless recording is not guaranteed in a redundancy system!

EXAMPLE

- The project *PRO1* is executed redundantly; one computer is the Primary Server, a second computer is the Standby Server.
- The same applies for the *PRO2* project.

- The integration project with the subordinate projects *PRO1* and *PRO2* is started on a third computer. This is the client for all sub projects.

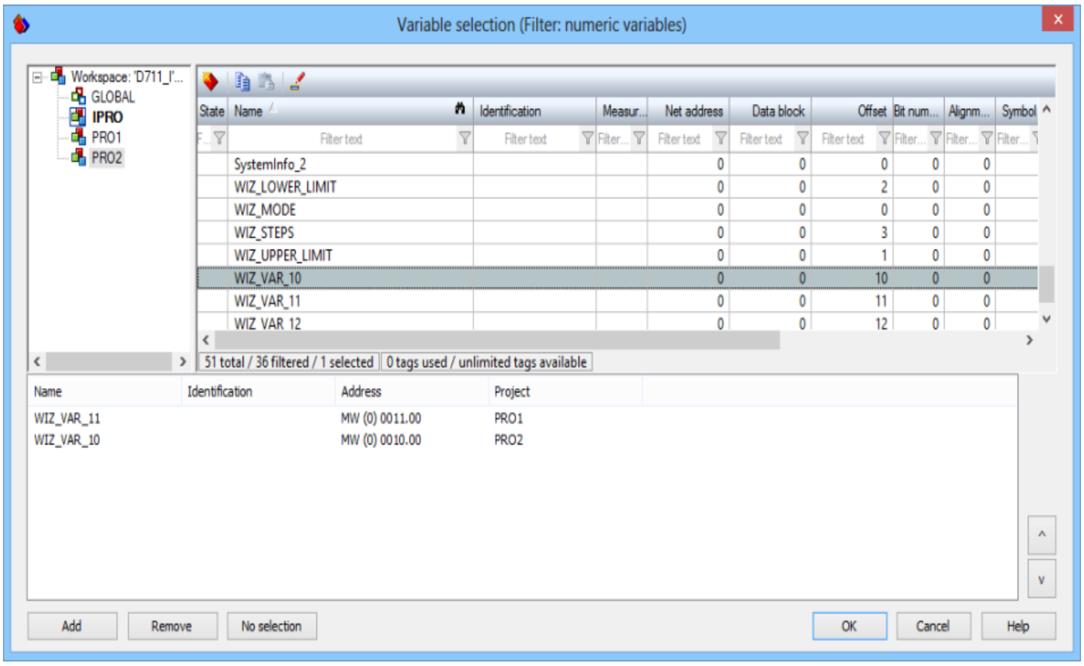
If variables of the projects *PRO1* and *PRO2* are now archived in the integration project, then the computer receives the data in relation to the network from the respective Primary Server of *PRO1* and *PRO2*.
If, for example, the Primary Server of *PRO1* fails, for the time period until the Standby Server of *PRO1* has taken over the Server role, there would be alternative values in the archive for variables from *PRO1*.
Note: The Standby buffer of seamless redundancy only saves variables of the project for which the computer has been configured as one of the two servers.
Solution: In order to ensure recording without interruptions, the archiving must be local in a redundantly-executed subproject.

5.2.3.1.4Recipes

You can write values to variables from different projects of the workspace in a recipe.

RECIPE EXAMPLE

1. In the project *IPRO* open the node **Recipes**.
2. Create, under **Standard recipe recipes** a new recipe with the name **Recipe 1**.
3. Open the context menu of **Recipe 1** and select **Add variable**.
4. The dialog for selecting variables is opened



State	Name	Identification	Measur...	Net address	Data block	Offset	Bit num...	Alignm...	Symbol
	SystemInfo_2			0	0	0	0	0	
	WIZ_LOWER_LIMIT			0	0	2	0	0	
	WIZ_MODE			0	0	0	0	0	
	WIZ_STEPS			0	0	3	0	0	
	WIZ_UPPER_LIMIT			0	0	1	0	0	
	WIZ_VAR_10			0	0	10	0	0	
	WIZ_VAR_11			0	0	11	0	0	
	WIZ_VAR_12			0	0	12	0	0	

51 total / 36 filtered / 1 selected | 0 tags used / unlimited tags available

Name	Identification	Address	Project
WIZ_VAR_11		MW (0) 0011.00	PRO1
WIZ_VAR_10		MW (0) 0010.00	PRO2

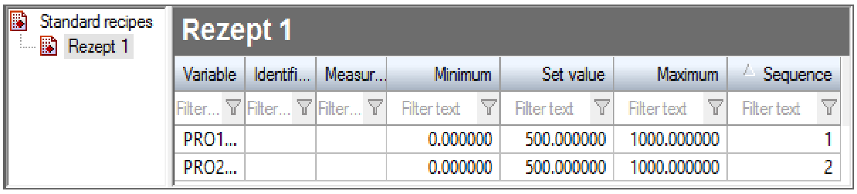
5. Here, you can select not just variables from the *IPRO*. To select variables from other projects:

a) In the left list area of the project, select a project from the tree view of the workspace. The variables of the selected project are shown in the main area.

b) Select the desired variable with a mouse click.

c) Select variables from *PRO1* and *PRO2*.

d) The name of the respective project is also placed in front of the variable name in variable list of the recipe.



Standard recipes
Rezept 1

Rezept 1

Variable	Identifi...	Measur...	Minimum	Set value	Maximum	Sequence
Filter...	Filter...	Filter...	Filter text	Filter text	Filter text	Filter text
PRO1...			0.000000	500.000000	1000.000000	1
PRO2...			0.000000	500.000000	1000.000000	2

Proceed in the same way for the Recipegroup Manager.

Attention

zenon does not check in the Editor to see if the network structure in Runtime actually allows access to the selected project and its variables.

For example, in the Editor, in project **PRO1**, a variable from the integration project can be selected. This connection only works in Runtime if the integration project has also been started. This connection will not work on a computer on which only project **PRO1** has been started (start project).

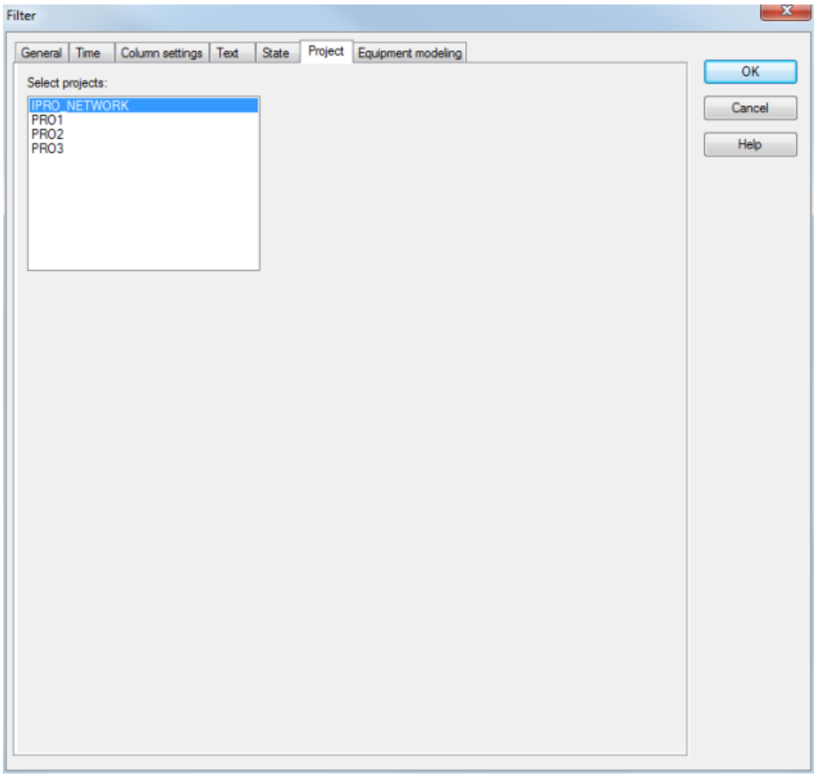
5.2.3.1.5 Alarms and CEL

In zenon, system messages and alarms from different projects of a workspace can be displayed together in a list. These entries can be filtered, displayed, printed or exported like the data from normal Alarm Message Lists or Chronological Event Lists.

AML example

1. Create an *AML* screen.
2. Add control elements to the screen via **Control elements -> Add templates**.
3. Create a function **Screen switch** for this screen.
4. The filter dialog for alarm lists is opened.
5. Open the **Project** tab.

6. Select the projects that are to be displayed in the AML of the *IPRO*. (Multiple selection button **Ctrl** key plus a mouse click.)

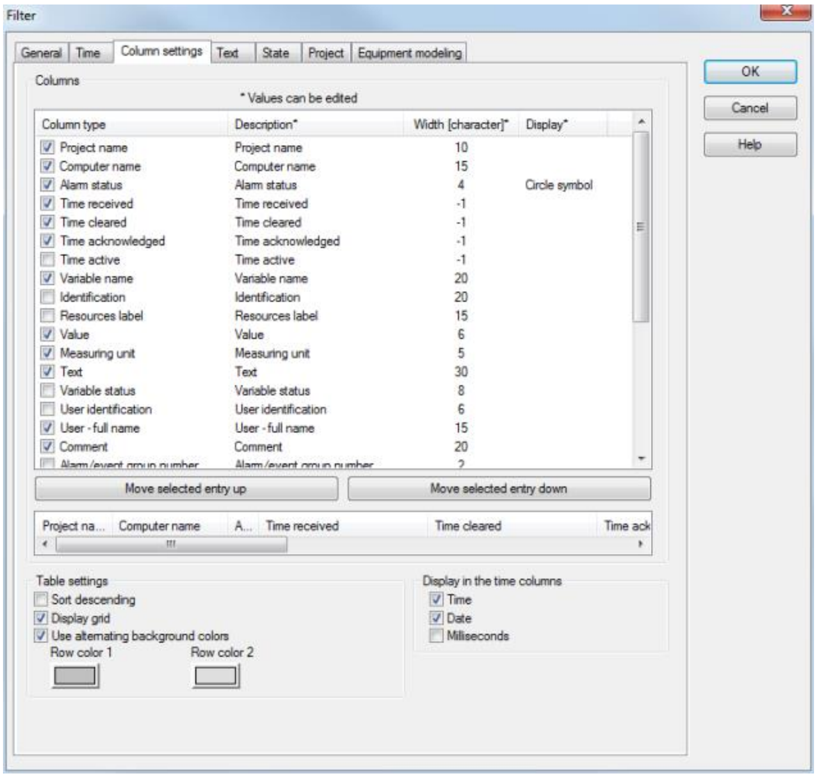


Select projects:

IPRO_NETWORK
PRO1
PRO2
PRO3

7. Open the **Column settings** tab.

8. Select the **Project name** property for display in Runtime.
   You thus gain an overview of the project from which an alarm comes in Runtime.



Columns

* Values can be edited

Column type	Description*	Width [character]*	Display*
☑ Project name	Project name	10	
☑ Computer name	Computer name	15	
☑ Alarm status	Alarm status	4	Circle symbol
☑ Time received	Time received	-1	
☑ Time cleared	Time cleared	-1	
☑ Time acknowledged	Time acknowledged	-1	
□ Time active	Time active	-1	
☑ Variable name	Variable name	20	
□ Identification	Identification	20	
□ Resources label	Resources label	15	
☑ Value	Value	6	
☑ Measuring unit	Measuring unit	5	
☑ Text	Text	30	
□ Variable status	Variable status	8	
□ User identification	User identification	6	
☑ User - full name	User - full name	15	
☑ Comment	Comment	20	
□ Alarm/event group number	Alarm/event group number	2	

Project na...	Computer name	A...	Time received	Time cleared	Time ack

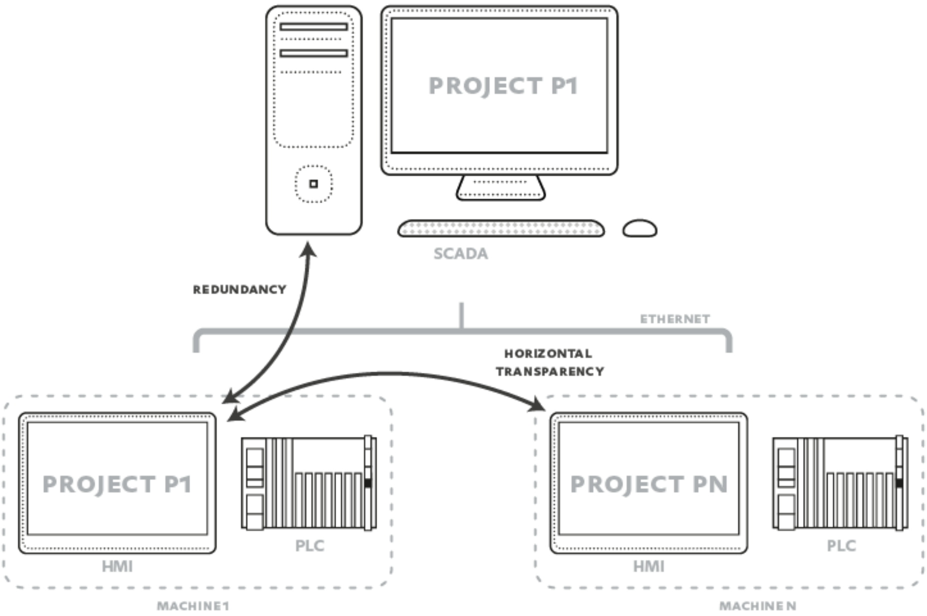
Table settings
□ Sort descending
☑ Display grid
☑ Use alternating background colors
Row color 1
Row color 2

Display in the time columns
☑ Time
☑ Date
□ Milliseconds

Proceed in the same way for the Chronological Event List.

5.3 Horizontal transparency

Multi-project administration is also the basis for **horizontal transparency** in zenon.
This enables all projects located on the same level to be shown and operated on each terminal.



The following conditions must be met in order to be able to use **horizontal transparency** in the Runtime:

- The integration project (on page 35) must be defined as the start project.
- The integration project must have the necessary navigation.
- Licensing: All variables and drivers of all running projects must be licensed.
  Pay particular attention to the total quantity and special drivers.

EXAMPLE

Several terminals belong to one machine. Each has its own visualization project. With the help of **horizontal transparency**, it is possible to show and operate, on each terminal, its own project and all other projects. This way the entire machine can be monitored and operated from each terminal.

5.4 Optimization for projects with a large number of Clients

Large network projects can, under certain circumstances with standard settings, place the Primary Server under full load for a period of time by reloading many Clients. The extent of the load depends on several factors (Primary Server resources, available bandwidth, etc.).
Guideline values:

- Runtime files of 10 MB or larger
- More than 50 clients

In this case, the reloading process can be optimized in order to prevent all Clients being reloaded at the same time. You undertake the project configuration of this in project.ini.

Possible INI entries for the optimization of the reloading process of clients in the zenon network:

- **RELOADDELAY_SEC**
- **CLIENT**

RANDOM RELOAD DELAY (RELOADDELAY_SEC)

With the **RELOADDELAY_SEC** ININ entry, the reloading is delayed by a random value.
To do this:

1. Open **project.ini** in the \Projekt_SQL_Ordner\FILES\\zenon\system\\. folder
   Hint: Highlight the project in the project manager and press the keys **Ctrl+Alt+E**; **t**he Windows Explorer opens the \Project_SQL_folder\FILES\\ folder
2. Go to the **[NETZ]**. section
3. Create the entry **RELOADDELAY_SEC=***[Value]*.
4. Select a *value* value for the delay.

When reloading, a random delay in seconds is calculated for each client, which is between *0* and the selected value. *0* means no delay (standard action). The selected value has no influence on standalone projects, the Primary Server or the Standby Server.

Note: This entry should only be set in very large projects with a noticeable delay when reloading. The standard settings provide better performance in normal projects.

DEFINED RELOAD DELAY (CLIENTX)

With the **CLIENTx** INI entry, the reload times for clients can be defined differently.

1. Open **project.ini** in the \Projekt_SQL_Ordner\FILES\\zenon\system\\. folder
   Hint: Highlight the project in the project manager and press the keys **Ctrl+Alt+E**; **t**he Windows Explorer opens the \Project_SQL_folder\FILES\\ folder
2. Go to the **[NETZ]**. section
3. Create a **CLIENTx=***[value]* entry for each client.
   x = the serial numbering for clients.
   CLIENT[serial number]=client name,[reload delay in seconds]
4. Select a *value* value for the delay.
   Example:
   - *CLIENT0=VM-CDSBG104,5*
   - *CLIENT1=WKS001,10*

Attention: the consecutive numbering of the clients must be continuously consecutive. If there are, for example, entries for CLIENT0, CLIENT1, CLIENT2, CLIENT3 and CLIENT5, only the entries for the following clients are taken into account: Client0, Client1, Client2 and Client3. The reload delay for Client5 is not taken into account.

Information

For clients that do not have a **CLIENTx** entry, the random reload delay, as given in the **RELOADDELAY_SEC**entry, is applicable.

If this entry is also empty, reloading takes place immediately.

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.

6 Strong encryption of network communication

zenon enables strong encryption of communication in the zenon network. Strong encryption works from zenon Version 7.0 for all supported operating systems and for the zenon Web Client.

If encryption is active, communication between the Primary Server, Standby Server, Clients and zenon Web Clients is in encrypted form; the zenon Web Server only forwards data packets and is not affected by encryption.

Information

Network communication was also encrypted in earlier versions of zenon. The method has changed with version 7. The term "encryption" in conjunction with zenon 7 or later always means strong encryption.

6.1 Basics

Encryption for zenon Runtime is available from version 7.00. It is not possible to communicate with earlier versions of zenon if encryption is switched on. Encryption does not impair any zenon functionality.

BASIC ENCRYPTION FROM ZENON 7.00

To use the strong encryption of the zenon network, note:

- The password is encrypted individually on each computer and stored in **zenon6.ini**. That means:
  - The password cannot be transferred by copying **zenon6.ini** to another computer.
  - If hardware components are changed, in the network adapter area in particular, the password may be invalid and need to be re-entered.
- Encryption must always be activated or deactivated for all components involved in the zenon network. Communication between encrypted and unencrypted systems is not possible. **zenon Web Server**s only act as a proxy computer and are not affected by encryption.
- If encryption is activated on a computer, it always applies for the projects of this computer with the **Network active** property active.

> ### Information
>
> AES 192 from Microsoft (https://msdn.microsoft.com/en-us/magazine/cc164055.aspx) is used as the encryption algorithm for network communication.
>
> SHA 256 from Microsoft (https://msdn.microsoft.com/en-us/library/system.security.cryptography.sha256%28v=vs.110%29.aspx) is used in order to generate the key from the entered password.

COMPATIBILITY

Encryption is not compatible with versions prior to zenon 7.00 SP0. That means:

System 1	System 2	Communication
zenon 7 encrypted	zenon 7 encrypted	Yes
zenon 7 unencrypted	zenon 7 unencrypted or zenon prior to version 7 unencrypted	Yes
zenon 7 encrypted	zenon 7 unencrypted or zenon prior to version 7 unencrypted	No

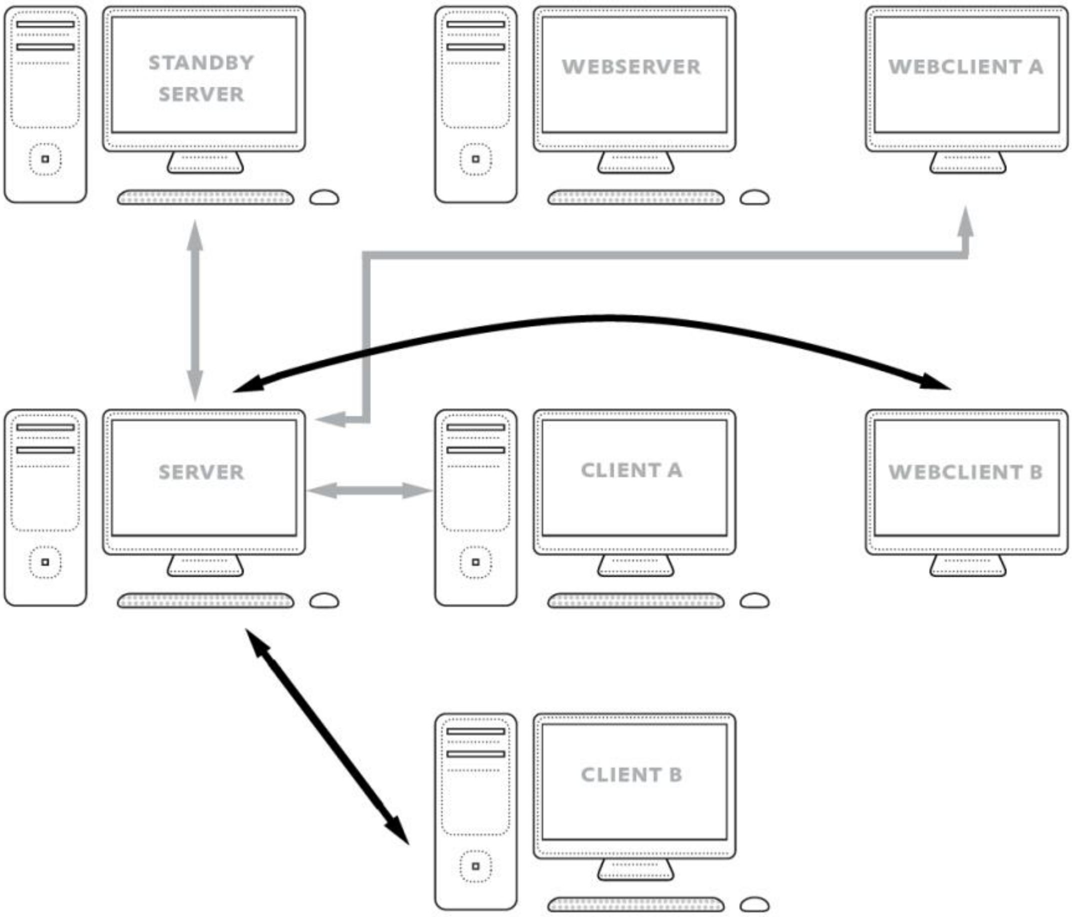
Errors (on page 54) are logged in the Diagnosis Viewer's LOG file.

EXAMPLE

The following illustration shows an example of a network with Primary Server, Standby Server, two clients, one zenon Web Server and two Web Clients. All devices are running zenon 7.00 SP0. The devices are configured as follows:

- Encryption is activated on the Primary Server using the Startup Tool (on page 50).
- Encryption is also activated on the Standby Server and client A via Remote Transport (on page 51) when Runtime files are transferred.
- Client B and Web Client B still communicate without encryption.
- On Web Client A, encryption is activated on the server using the Startup Tool (on page 50).
- Because the zenon Web Server does not evaluate the data packets, but instead forwards these on immediately, it does not require encryption. In theory, it can also have an older version, and the Web Clients can nevertheless create encrypted connections.

This configuration leads to the following result:



- The Standby Server communicates successfully with the Primary Server.
- Client A can log in to the Primary Server and exchange data.
- Because client B sends unencrypted messages and these are rejected by the Primary Server because encryption is active, client B cannot communicate with the Primary Server and is therefore offline.
- Web Client A logs on to the server via the zenon Web Server and can exchange data.
- The unencrypted messages from Web Client B are forwarded from the zenon Web Server to the Primary Server, but the server rejects these. Web Client B cannot communicate with the Primary Serverr and is therefore offline.

As soon as encryption via Remote Transport or the Startup Tool configuration on client B and via **Encrypt network communication** on Web Client B is activated, these connections can also make connections to the Primary Server.

6.2 Activate encryption

Encryption can be activated in different ways:

- By means of the **Startup Tool** (on page 50) for the local computer and the zenon web client
- via Remote Transport (on page 51)

> **Hint**
>
> For quick, easy activation of the encryption, it is recommended that the configuration is carried our on a computer using Remote Transport (on page 51).

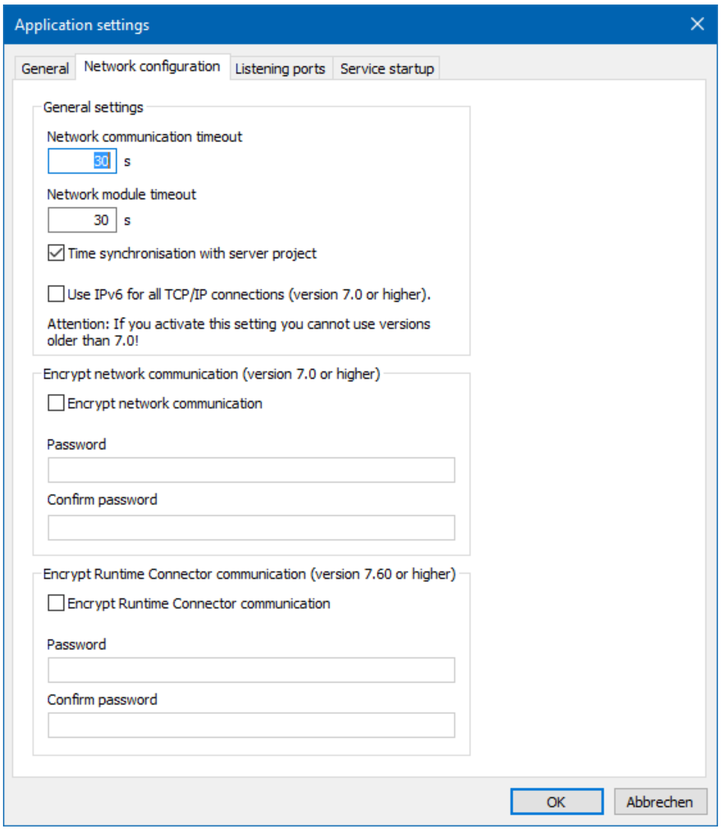
6.2.1 Locally via the Startup Tool

To activate encryption on the local computer or for the <CD_PRODUCTNAME Web> Client:

1. Open the zenon **Startup Tool**.
2. Click **Application -> Options**.

   The dialog for the settings is opened.

3. Select the **Network configuration** tab.



Application settings

General | Network configuration | Listening ports | Service startup

General settings

Network communication timeout: 30 s

Network module timeout: 30 s

☑ Time synchronisation with server project

□ Use IPv6 for all TCP/IP connections (version 7.0 or higher).

Attention: If you activate this setting you cannot use versions older than 7.0!

Encrypt network communication (version 7.0 or higher)

□ Encrypt network communication

Password

Confirm password

Encrypt Runtime Connector communication (version 7.60 or higher)

□ Encrypt Runtime Connector communication

Password

Confirm password

4. Activate the checkbox **Encrypt network communication.**
5. Enter the password and verify it.
6. Confirm the dialog by clicking on **OK**.

CONNECTOR ENCRYPTION

In order to activate the encryption for the SCADA Runtime Connector zenon or zenon Analyzer, the HTML web engine or for the Runtime remote driver, configure the Encrypt Runtime connector communication group of properties.

6.2.2 Via Remote Transport

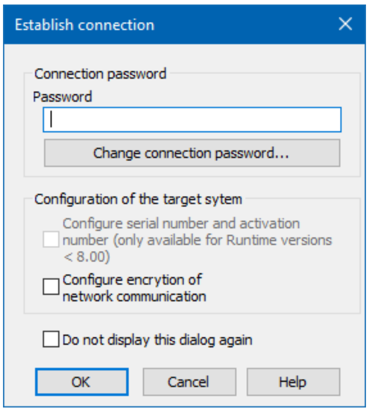
Encryption can be activated on remote computers using Remote Transport. However, this is only possible if the Remote Transport connection is protected with a password.

To activate encryption using Remote Transport:

1. Click on the corresponding button in the Remote Transport toolbar

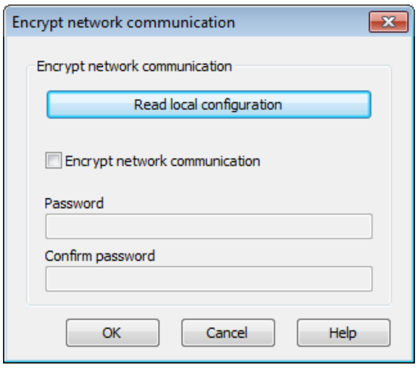
   or select, in the project's context menu: **Set up Remote Transport> connection.**

The dialog for setting up the connection is opened


Establish connection
Connection password
Password
Change connection password...
Configuration of the target sytem
□ Configure serial number and activation number (only available for Runtime versions < 8.00)
□ Configure encrytion of network communication
□ Do not display this dialog again
OK
Cancel
Help

2. Enter the connection password or create one, if none has been set
3. Activate the **Configure encryption of network communication** checkbox
4. Click on **OK**.

   The dialog for encryption of network communication is opened


Encrypt network communication
Encrypt network communication
Read local configuration
□ Encrypt network communication
Password
Confirm password
OK
Cancel
Help

5. Activate the **Encrypt network communication** checkbox
6. Issue a password (for criteria, see the **network password encryption** (on page 52) section.)
   To quickly transfer the local configuration to other computers, the local password can first be read via **Read local configuration**.
7. Confirm the dialog by clicking on the **OK** button.

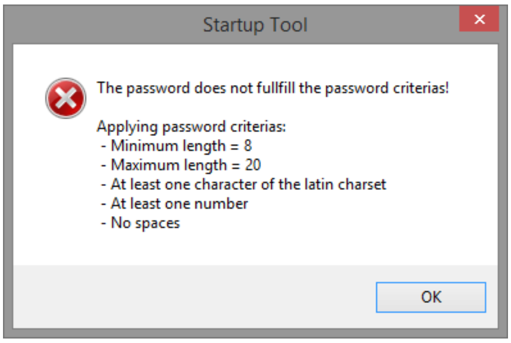
6.3 Network encryption password

The following is applicable for encryption of the communication in the network:

- Minimum length *8 characters*
- Maximum length: *20 characters*
  The displayed length is always set at *20 characters*, in order to hide the actual length.
- Permitted characters:

  - Letters: *A - Z; a - z*
  - Digits: *0 - 9*
  - Special characters
- Non-permitted characters:
  - Space
  - **Enter key** (Return key)
- Summary: a password must contain at least 1 figure and 1 letter

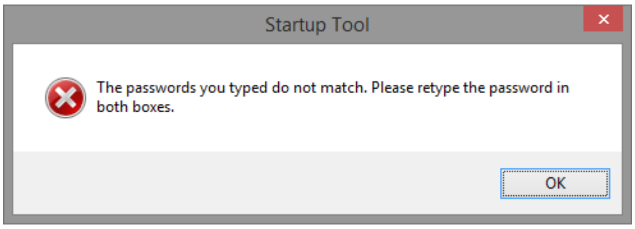
If the password entered does not correspond to these requirements, an error message is shown:



The password does not fullfill the password criterias!

Applying password criterias:
- Minimum length = 8
- Maximum length = 20
- At least one character of the latin charset
- At least one number
- No spaces

If, when configuring, the confirmation password is entered incorrectly, this is shown with an error message:



The passwords you typed do not match. Please retype the password in both boxes.

6.4 Checklist for errors

In the event of errors, check:

- Do all computers have access to the network and does the naming resolution work between the computers?
- Was the **Network active** property activated for the project in the Editor?
- Is zenon Runtime version 7.00 SP0 or higher being used? (relevant if encryption is being used)
- Is - for projects with encryption - the configuration correct on all computers? (**USE_ENCRYPTION** setting in **zenon6.ini**: The same for all computers, either *0*, *1* or not present.)
- Was the password set correctly?

- Was the hardware changed on one of the computers involved after the encryption has been configured?
- Does a ping work on a computer?
  - Yes: Network connection present, fault is with the communication.
  - No: Check the network.
- Is it possible to connect to Telnet?
  - The connection is made: Both computers communicate at the same level. Check the password.
  - The connection is made and lost again: One computer communicates with encryption, one without encryption.
  - Faulty connection: zenon Runtime does not run on the target computer.

  Note: Telnet must be installed as an extra on more recent Windows operating systems.
  Connection is generally made via port 1100.
  The Telnet command is then: **open [IPAdresse] 1100**
- Are the required (operating system) functions available (primarily relevant for CE terminals)?
  - Non-existent functions lead to Runtime not being able to start.
- If the service provider or one of the algorithms is not available, an error message (on page 54) is written to the log file when Runtime is started.

Errors (on page 54) are logged in the Diagnosis Viewer's log file.

6.5 Error message

Errors are either displayed in the output window of the zenon Editors, in pop-ups or in the LOG files of the Diagnosis Viewer.

NO CONNECTION

If a client has been configured with an incorrect encryption password (not the same as the password on the Primary Server) then this is evident from the following events:

- The Client is offline, although the Primary Server can be reached by pinging.
- The Primary Server writes error messages to the LOG file:

  SysMod Error: Serialize in Object Project: [project name] Modul: [module number]

  or:

  NET Error During Decryption [Error number]

POP-UPS AND ERROR MESSAGES

Encryptions errors are notified by means of pop-ups (on page 55) and entries in LOG files (on page 57) or in the zenon output window (on page 56).

6.5.1 Error messages in pop-ups

STARTUP TOOL AND WEB CLIENT

The following error messages are output by the zenon **Startup Tool** as a pop-up for local encryption or by the **Encrypt Network Communication Tool** for the configuration of the zenon Web Client.

These messages are always in English. Messages from Runtime itself are in the respective language that has been set.

Error message	Description
The password has to be entered in both text boxes!	When configuring the encryption, the user has left one of the two input fields (**Password** or **Password confirmation**) empty.
The passwords you typed do not match. Please retype the password in both boxes.	The content of the input field for password confirmation is different to the content of the input field for the password.
The network password does not fullfill the password criterias! *Password criterias:* *- Minimum length = 8* *- Maximum length = 20* *- At least one character of the latin charset* *- At least one number* *- No spaces*	The password entered does not meet the password criteria. The password criteria are displayed in the error message.
The network password could not be encrypted!	An error occurred when encrypting the network password.
The network encryption configuration in the file zenon6.ini is invalid. *Please enter a new configuration.*	When opening the **Network configuration** tab, it was established that the **zenon6.ini** file does not contain a valid configuration for the network encryption. A new configuration must be entered.
The network encryption password in zenon6.ini is invalid.	The password read in from the **zenon6.ini** file is invalid and must be reentered.
The password for network encryption is	Message when Runtime starts if the password cannot be verified.

Error message	Description
invalid and must be entered again!	

REMOTE TRANSPORT

The following error messages are given by **Remote Transport** as a pop-up when the remote computer is encrypted.

Error message	Description
For configuring the network encryption, the Remote Transport connection must be protected with a password!	An attempt was made to configure remote encryption without securing the Remote Transport connection by means of a password.
You must enter the password in both input fields!	When configuring the encryption, the user has left one of the two input fields (**Password** or **Password confirmation**) empty.
The password confirmation does not match the password!	The content of the input field for password confirmation is different to the content of the input field for the password.
The password entered does not correspond to the password criteria. *Password criteria:* *At least 8 characters* *Maximum 20 characters* *At least one letter* *At least one number* *No spaces*	The password entered does not fulfill the password criteria. The password criteria are displayed in the error message.
An error occurred when encrypting the password!	An error occurred when encrypting the password. If this error occurs during configuration via Remote Transport, a more detailed error message is written to the log.
An error occurred when decrypting the network password from zeon6.ini.	The password stored in **zenon6.ini** cannot be decrypted. If this error occurs during configuration via Remote Transport, a more detailed error message is written to the log.
The encryption configuration in zenon6.ini is not valid and must be reentered.	The password read off from **zenon6.ini** is invalid. The password must be entered again.

6.5.2 Error messages in the output window

Errors are displayed in the output window as messages:

Message	Level	Description
The server reports an error when compiling the data for the encryption configuration.	**ERROR**	The remote **zenSysSrv** reports an error when compiling the information for the encryption of the password for network encryption. The adapter information cannot be read off.
**** Configure the encryption of the network communication at the target system.*		This message is at the start of the conclusion message after encryption has been configured on a remote device via Remote Transport. After this, there is a message in relation to the success of the remote configuration.
The server reports an error when the encryption configuration is changed.	**ERROR**	The remote **zenSysSrv** reports an error when the encryption configuration is saved to the remote device. The configuration was not saved.
The configuration was successfully saved on the server.	**MESSAGE**	The remote **zenSysSrv** reports that the encryption configuration was successfully saved.
The version of the remote zenSysSrv is too low. The encryption cannot be configured.	**ERROR**	An attempt was made to configure the encryption on a remote device, which has a **zenSysSrv** from a version prior to version 7.00 SP0. Encryption is only available from zenon version 7.00 SP0; an earlier version of **zenSysSrv** cannot therefore configure this.

6.5.3 Error messages in LOG files

Errors in encrypted network traffic are documented in LOG entries. The **Error IDs** of the error messages in the following summary are system or COM error codes. You can find more information in the MSDN library.

LOG entry	Level	Description
NET Error During Acquiring Cryptography Context [Error-ID]	**ERRORS**	The creation of a service provider for encryption was unsuccessful.
NET Error During Creating Hash [Error-ID]	**ERRORS**	The creation of a hash value was unsuccessful.
NET Error During Using Hash	**ERRORS**	The processing of a hash value was unsuccessful.

LOG entry	Level	Description
[Error-ID]		
NET Error During Destroying Hash [Error-ID]	**ERRORS**	The release of a hash value that is no longer required was unsuccessful.
NET Error During Deriving Key [Error-ID]	**ERRORS**	The creation of a key for symmetrical encryption was unsuccessful.
NET Error During Configuring Key [Error-ID]	**ERRORS**	The setting of parameters for symmetrical encryption was unsuccessful.
NET Error Cryptography Not Initialized!	**ERRORS**	An encryption or decryption function was called up but initialization of the required parameters (service provider, key) was unsuccessful.
NET Error Invalid Pointer passed!	**ERRORS**	An encryption or decryption function was given invalid parameters.
NET Error Message Length Must Not Be 0!	**ERRORS**	The encryption function was called up with an empty message.
NET Error During Buffer Length Calaulation [Error-ID]	**ERRORS**	The calculation of required buffer size for encryption was unsuccessful.
NET Error Buffer Length Must Not Be 0!	**ERRORS**	The buffer for encryption or decryption has not been created.
NET Error During Decryption 0x%x	**ERRORS**	An error occurred during decryption.
NET Error During Encryption 0x%x	**ERRORS**	An error occurred during encryption.
NET Error: Encryption Is Required And Project [Projekt] Received Plaintext Network Message	**ERRORS**	Encryption is active and a decrypted message was received. The message is discarded in this case.
NET Error: Encryption Is Not Supported And Project [Projekt] Received Encrypted Network Message	**ERRORS**	Encryption is not active and an encrypted message was received. The message is discarded in this case.
NET Cryptography Successfully Initialized	**DEBUG**	The parameters required for encryption and decryption were initialized successfully. The parameters are initialized when Runtime is started.

LOG entry	Level	Description
NET Uninitializing Cryptography	**DEBUG**	The parameters required for encryption and decryption were released. This happens when Runtime is ended. If the log connection is separated before release, the message does not appear in the Diagnosis Viewer
NET Error During Buffer Size Calculation [Error ID]	**ERRORS**	An error occurred when the necessary buffer size for compiling information for encrypting or decrypting the network password was calculated.
NET Error During Buffer Size Calculation: No Adapters	**ERRORS**	The computer does not have a network adapter. For this reason, the network password cannot be encrypted or decrypted.
NET Error During Adapter Info Query [Error ID]	**ERRORS**	An error occurred when the adapter information for encrypting or decrypting the network password was read off.
NET Error Passwort Not Properly Formatted	**ERRORS**	The hex dump of the encrypted password is in an invalid format.
NET Error During Decrypting Password [Error ID]	**ERRORS**	An error occurred when decrypting the network password.
NET Error During Encrypting Password [Error ID]	**ERRORS**	An error occurred when encrypting the network password.
NET Cryptography Is Disabled	**DEBUG**	Encryption of the network traffic is deactivated.
NET Error No Password	**ERRORS**	Encryption is active, but no password is entered.
NET Error Password Could Not Be Decrypted	**ERRORS**	The password for network encryption could not be decrypted.
NET Password successfully loaded	**DEBUG**	The password for network encryption has been loaded successfully.
Network Cryptography Disabled By Remote Configuration	**DEBUG**	**zenSysSrv** reports that encryption of network traffic on the computer was deactivated by the Remote Transport configuration.
Network Cryptography Enabled By Remote Configuration	**DEBUG**	**zenSysSrv** reports that encryption of network traffic on the computer was activated by the Remote Transport configuration.
Network Cryptography Remote Configuration Error	**ERRORS**	A configuration sent by Remote Transport for network encryption is erroneous.

LOG entry	Level	Description
Error During Buffer Size Calculation [Error ID]	**ERRORS**	An error occurred when the necessary buffer size for compiling information for encrypting or decrypting the network password for the configuration of Remote Transport was calculated.
Error During Buffer Size Calculation: No Adapters	**ERRORS**	The computer does not have a network adapter. For this reason, the network password cannot be encrypted or decrypted and thus not set via Remote Transport (it must therefore be connected via COM). The use of network encryption on a computer without a network adapter makes no sense however.
Error During Adapter Info Query [Error ID]	**ERRORS**	An error occurred when the adapter information for encrypting or decrypting the network password for configuration via Remote Transport was read off.
NET Error During Decrypting Password: The Password is Invalid!	**ERRORS**	The password is no longer valid, because the initial data for computer-dependent encryption has changed. This error can be rectified by configuring the password again. The decryption process is usually canceled before the validity of the password is checked, because the old password cannot be decrypted with the new encryption data. This leads to the „*NET Error During Decrypting Password 0x80090005*" error, which is usually displayed instead of „*NET Error During Decrypting Password: The Password is Invalid!*". Another consequence is that a password that is now invalid on the computer in question can lead to error messages when network packages are sent or received. The „*NET Error Cryptography Not Initialized!*" error message is written to the LOG file.
Connector-Container: Decryption failed. Perhaps wrong encryption	**ERRORS**	The encryption of the data packet received by the connector client was not successful.

LOG entry	Level	Description
key.		

7 zenon on the Terminal Server

The zenon Runtime can also be used together with a terminal server solution.

Limitations:

- The Editor cannot run on a terminal server.
- Project simulation is not available for clients at the terminal server.
- zenon Logic is not available for clients at the terminal server.

> **Information**
>
> Keep in mind that the name of the terminal client is resolved. That means: The name of the device that initiates the terminal connection is the name of the Clients for the zenon network.
>
> Attention: Ensure that all corresponding ports have been unlocked when using a firewall.

Terminal server solutions are offered by several manufacturers. All tests with zenon were carried out using the Windows terminal server (Windows Remote Desktop Services).

> **Attention**
>
> When using zenon with a terminal server, it must be licensed with a Network dongle.

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.

7.1 How terminal servers work

With terminal servers, it is possible to provide data and applications centrally, regardless of the end device. Terminal servers make it possible to start several shell instances (desktops) that are separate from one another on the terminal server. If a client connects to the terminal server, a new shell instance is created and the client is assigned its own graphic user interface. The applications are executed on the terminal server itself and the data is saved on the terminal server. Input (via the keyboard, mouse etc.) and output (display, audio, etc.) is on the Client.

> **Info**
>
> Not all software is compatible with terminal servers.

7.2 Advantages and disadvantages

The use of zenon offers the following advantages and disadvantages:

ADVANTAGES

- Only one computer (the terminal server) has to be maintained.
- Clients do not have to be very powerful (Thin Clients).
- Clients can have different operating systems (Windows, Windows CE, Linux, Unix, MacOS, iOS, Android etc.).
- High degree of data security, because no data is saved on the Client.

DISADVANTAGES

- All started programs of all instances run on one computer (the terminal server).
  This:
  - must have sufficient computing power for all started programs.
  - must have sufficient RAM for all started programs.
- All interfaces have to be shared. E.g. network adapters, COM ports, parallel ports.
- The network load can be high with an appropriate number of Clients (such as transfer of graphic data).
- The screen resolution is defined by the client started first. If screens in different resolutions are to be used, this can be implemented on the terminal server by means of an entry in the **zenon6.ini** (**SERIALIZE=***1*). All screens are then calculated again for the client, which further increases the necessary performance for the terminal server.

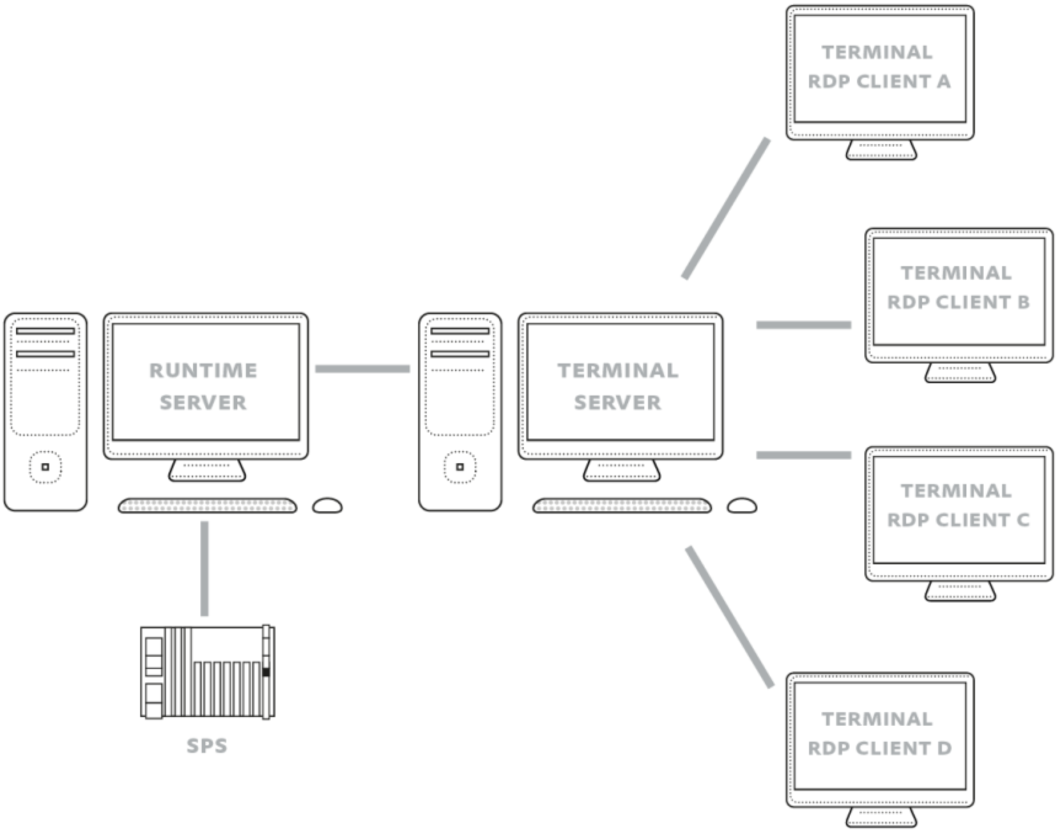
7.3 How zenon works on the terminal server

On a terminal server, operation is possible as:

- zenon Client possible without limitations
- zenon standalone systems only worthwhile and possible:
  - as a superordinate integration project to start several Client subprojects
  - without drivers, database connections etc.
- zenon Primary Server not possible

SCHEMATIC DISPLAY

Example topology of a terminal server network with zenon:



Computers	Description
Runtime Server	zenon Primary Runtime Server
Terminal Server	Terminal Server and n-fold Runtime Client
Terminal RDP Client A **- D**	Terminal Clients (input and output only)

7.4 Required settings

In order for zenon Runtime to be able to be started more than once on the terminal server, several settings must be made. The settings for registration, screen resolution and transfer service can be undertaken with the zenon Startup Tool.

Information

The following parameters are automatically set with registration via the Startup Tool:

- INI entries:
  - **[TERMINAL]**
    CLIENT=*1*
  - **[DEFAULT]**
    SERIALIZE=*1*
- Registration of **ZenSysSrv.exe** as a service
- Deregistration **ZenDBSrv.exe**

GENERAL SETTINGS

1. **Registration**

   Register the use on the terminal server via the zenon Startup Tool.
   Alternatively, you can configure the corresponding INI entries manually.

   zenon6.ini entry:
   The following entry must be added in **zenon6.ini** on the terminal server. On the Primary Runtime server no settings are needed.

   [TERMINAL]
   CLIENT=*1*

   1: The Runtime can be started several times, and all settings for the terminal server operation are automatically set by the Runtime.

0: Runtime can only be started once per terminal server session. Operation on the terminal server is not possible. (standard setting)

2. **Automatic adjustment of the screen resolution**

   Per default the first client at the terminal server defines the screen resolution. This setting can be amended on the terminal server with the following entry in zenon6.ini:

   [DEFAULT]
   SERIALIZE=*1*

   - *0*: Screen resolution individual, all screens are recalculated for each client
   - *1*: The first client started sets the screen resolution.

3. **Transfer**

   The transport service (**ZenSysSrv.exe**) must be registered and started as a Windows service, not as a standard EXE file. This setting is automatically set when registering via the Startup Tool, but can also be set manually.

   Manual setting:

   a) Start the program from the command line interface with the **-service** option.

      For example: **C:\Programs (x86)\COPA-DATA\zenon810\zenSysSrv.exe -service**

   b) Then start the Windows Service Manager. The service will be started automatically during every computer restart.

   Note: The setup always registers the transport service as a standard EXE. Therefore the transport service must be re-registered as a Windows service after every reinstallation.

4. **Runtime folder**

   All users must have write access to the Runtime folder. All Windows users (Windows users: **All**) in Windows Explorer must have complete access to the Runtime folder and all its subfolders.

   Attention

   If **TERMINAL=** *1* is set, the project simulation is no longer available.

SELECTIVE RELOADING OF SINGLE PROJECTS

Projects can also be synchronized selectively. In this case clients only reload projects if project changes exist. To activate the selective reloading:

1. Open the **zenon6.ini** file with a text editor.
2. Go to the **[TERMINAL]** section.
3. Edit or create the entry: **CLIENT_NO_FILE_ALIGN=**.

   Possible values:

- *0*: Projects are always reloaded by all clients
- *1*: Selective synchronization active. Only the zenon client which is started in the console session of the terminal server synchronizes the Runtime files with the zenon server.

After synchronizing the Runtime files, the console client writes the **reloadindicator.tmp** file to the directory that contains the **project.ini** file of the program . The session clients at the terminal server check every 10 seconds whether this file is available. If it exists and its file time stamp is newer than the date of the last reload, a session client reloads automatically.

ENTRY IN ZENON6.INI FOR SELECTIVE RELOAD

[TERMINAL]

CLIENT=*1*

CLIENT_NO_FILE_ALIGN=*1*

[DEFAULT]

SERIALIZE=*1*

> **Information**
>
> You can get further information in the configuration files manual in the Terminal server [TERMINAL] chapter.

7.5 zenon Remote Desktop versus Terminal Server

The terminal server solution is different from zenon Remote Desktop primarily due to the following points:

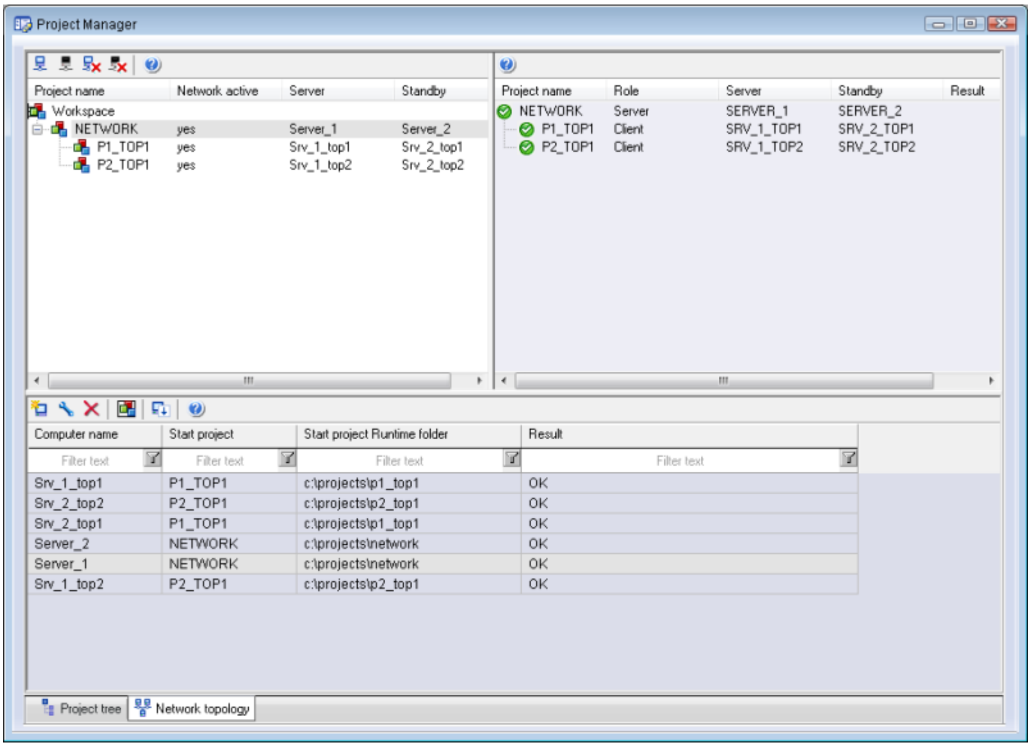
zenon Remote Desktop connection	Terminal server connection
All connected stations always see one and the same desktop. If e.g. one user starts a program, all see the same program, the same mouse cursor, the same keyboard input, etc.	Each connected station has its own desktop - an own instance. Only it sees what happens there. Mouse actions and keyboard inputs only affect this one instance.
	That also means: In each instance, a program can be started separately, e.g. a text editor. The program then runs on the terminal server several times and therefore needs more resources.

Information

You can also find further information in the zenon Remote Desktop manual

8 Administering and checking network topology

The network topology is displayed in a separate project manager tab.



Project name	Network active	Server	Standby
Workspace			
NETWORK	yes	Server_1	Server_2
P1_TOP1	yes	Srv_1_top1	Srv_2_top1
P2_TOP1	yes	Srv_1_top2	Srv_2_top2

Project name	Role	Server	Standby	Result
NETWORK	Server	SERVER_1	SERVER_2	
P1_TOP1	Client	SRV_1_TOP1	SRV_2_TOP1	
P2_TOP1	Client	SRV_1_TOP2	SRV_2_TOP2	

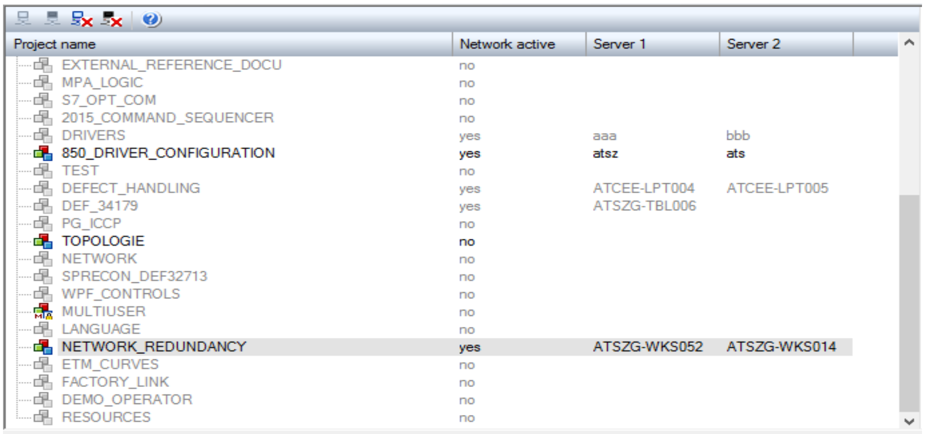
Computer name	Start project	Start project Runtime folder	Result
Filter text	Filter text	Filter text	Filter text
Srv_1_top1	P1_TOP1	c:\projects\p1_top1	OK
Srv_2_top2	P2_TOP1	c:\projects\p2_top1	OK
Srv_2_top1	P1_TOP1	c:\projects\p1_top1	OK
Server_2	NETWORK	c:\projects\network	OK
Server_1	NETWORK	c:\projects\network	OK
Srv_1_top2	P2_TOP1	c:\projects\p2_top1	OK

The display is divided into three areas:

- Topology tree (on page 68) (top left):
  shows active projects; the global project is not displayed
- Event tree (on page 69) (top right):
  only the result is displayed; represents the topology tree of a selected computer
- Computer list (on page 70)(bottom):
  List display and configuration of computers in the network

8.1 Topology tree

The topology tree displays active projects in hierarchical form.



Project name	Network active	Server 1	Server 2
EXTERNAL_REFERENCE_DOCU	no		
MPA_LOGIC	no		
S7_OPT_COM	no		
2015_COMMAND_SEQUENCER	no		
DRIVERS	yes	aaa	bbb
850_DRIVER_CONFIGURATION	yes	atsz	ats
TEST	no		
DEFECT_HANDLING	yes	ATCEE-LPT004	ATCEE-LPT005
DEF_34179	yes	ATSZG-TBL006	
PG_ICCP	no		
TOPOLOGIE	no		
NETWORK	no		
SPRECON_DEF32713	no		
WPF_CONTROLS	no		
MULTIUSER	no		
LANGUAGE	no		
NETWORK_REDUNDANCY	yes	ATSZG-WKS052	ATSZG-WKS014
ETM_CURVES	no		
FACTORY_LINK	no		
DEMO_OPERATOR	no		
RESOURCES	no		

Parameter	Description
Project name	Is defined in the project tree tab and cannot be changed here.
Network active	Displays whether the network option is active for this project. The setting can be changed via the **Network active** property. ▶ *Yes* project is a network project ▶ *No* project is not a network project
Server 1	Displays the Primary Server defined for this project. The setting can be changed via the context menu, the symbol in the toolbar or the **Server 1** property.
Server 2	Displays the Standby Server defined for this project. The setting can be changed via the context menu, the symbol in the toolbar or the **Server 2** property.

8.1.1 Toolbar and context menu

Property	Description
Set computer as Primary Server	Defines the computer highlighted in the computer list (on page 70) as the Primary Server for the project highlighted in

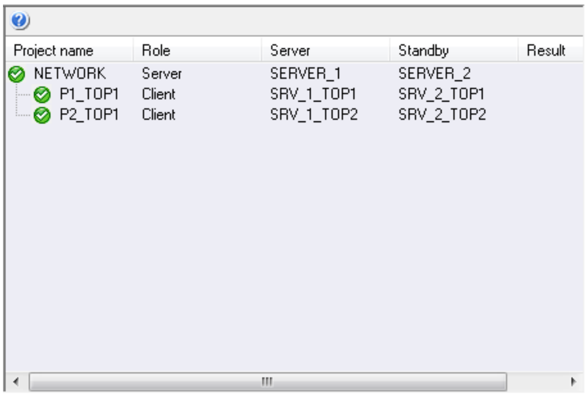
Property	Description
	the tree.
Set computer as Standby Server	Defines the computer highlighted in the computer list (on page 70) as the Standby Server for the project highlighted in the tree.
Delete Primary Server	Deletes the Primary Server defined for the highlighted project. **Note:** Configured computer is removed from the **Server 1** property.
Delete Standby Server	Deletes the Standby Server defined for the highlighted project. **Note:** Configured computer is removed from the **Server 2** property.
Help **(? symbol)**	Opens online help.

8.2 Result tree

The result tree represents the project tree of the computer selected in the computer list (on page 70) from the project, which is set as a start project for the selected computer and displays these project settings.

The result tree is empty if:

- The start project of the selected computer was not found
- More than one computer in the list was selected



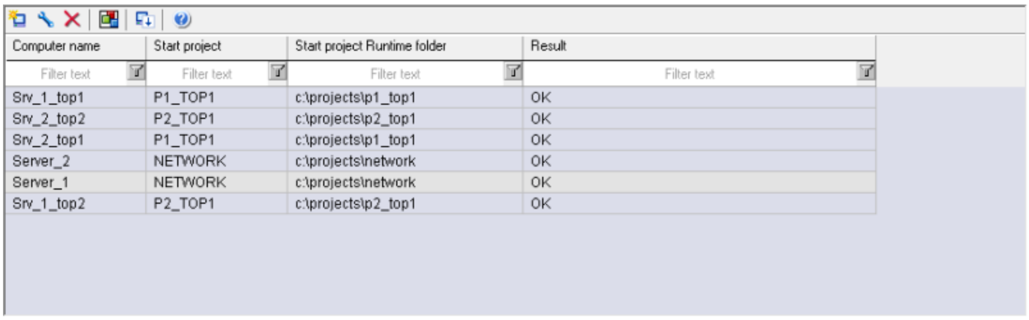
Project name	Role	Server	Standby	Result
✓ NETWORK	Server	SERVER_1	SERVER_2	
✓ P1_TOP1	Client	SRV_1_TOP1	SRV_2_TOP1	
✓ P2_TOP1	Client	SRV_1_TOP2	SRV_2_TOP2	

Parameter	Description
Project name	Projects that are assigned to the selected computer.
Role	Role of the computer: - Primary Server - Standby Server - Client
Primary Server	Name of the computer that acts as a Primary Server in the Runtime.
Standby Server	Name of the computer that acts as a Standby Server in the Runtime.
Result of test	Shows detailed error messages (on page 74) for topology test.

8.3 Computer list

The computer list shows computers that have been set up in the project and allows them to be configured. The list relates to the workspace and is saved in the workspace file (***.wsp6**).

The cells can be sorted and filtered. The column width can be amended with a right mouse click.



Computer name	Start project	Start project Runtime folder	Result
Filter text	Filter text	Filter text	Filter text
Srv_1_top1	P1_TOP1	c:\projects\p1_top1	OK
Srv_2_top2	P2_TOP1	c:\projects\p2_top1	OK
Srv_2_top1	P1_TOP1	c:\projects\p1_top1	OK
Server_2	NETWORK	c:\projects\network	OK
Server_1	NETWORK	c:\projects\network	OK
Srv_1_top2	P2_TOP1	c:\projects\p2_top1	OK

Parameter	Description
Computer name	Name of the computer. Can be changed by: - Click in the cell: Clicking on the **...** button opens a drop-down list of the computers currently available in the network. **Note:** This is only possible for pre-existing entries. - **Edit computer** entry in the context menu or toolbar. Opens the dialog to configure the computer (on page 73) in the network.

Parameter	Description
	- **Computer name** property.
Start project	The start project assigned to the computer Can be changed by: - Click in the cell: Select from drop-down list. **Note:** This is only possible for pre-existing entries. - **Set start project** entry in the context menu or the toolbar Sets the project selected in the topology tree (on page 68) as the start project. - **Start project** property.
Start project Runtime folder	Folder for project files on the target computer. The files of the start project are saved in this folder. All other projects relating to this correspond to the structure of the **Runtime folder** set up on the local computer. For example: **Start project Runtime folder**: *C:\Projecs\Top* = location where start project is stored. Sub projects are stored in *C:\Projects\*. **Hint:** Use the project name as folder name in order to create the same structure as on the engineering computer automatically. The **Start project Runtime folder** can be changed by: - double clicking on the computer: Opens the Configure computer dialog (on page 73) in the network. - Click in the cell: Manual entry possible. - **Start project Runtime folder** property.
Result of test	Shows the result of the topology test. - **OK:** All projects are free of errors. - **Error detected - For details see detail view!** One or more projects have an error. - **Critical error detected - For details see the detail view!** A project has a serious error. Serious errors halt further testing. - **Not tested, because there is a serious error in the structure**: The computer was not fully tested, because the test was

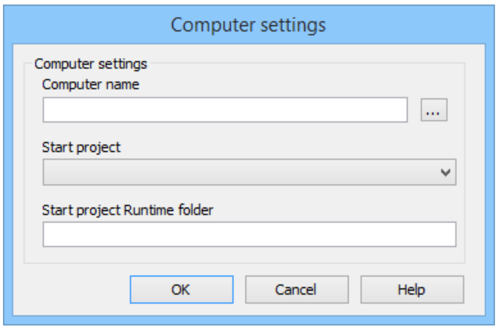
Parameter	Description
	ended due to a serious error. Detailed error messages (on page 74) are displayed in the result tree.

8.3.1 Toolbar and context menu

Entry	Description
Add computer...	Opens the Configure computer dialog (on page 73) in the network.
Edit computer...	Opens the dialog to configure the computer (on page 73) in the network for these computers.
Delete computer	Deletes computer from the topology after requesting confirmation. Multiple selection is possible. **Attention:** Deletion of the Primary Server or Standby Server leads to serious errors in the topology.
Set start project	Sets the project selected in the topology tree (on page 68) as the start project.
Copy Runtime files from all projects on the computer	Copies all projects valid for the computer to the target computer. The result is displayed in the output window.
Help **(? symbol)**	Opens online help.

8.3.2 Computer network configuration dialog

The following configurations are necessary to configure the computers:



Parameter	Description
Computer name	Clicking on the **...** button opens the dialog to select a computer from the list of computers that are currently available in the network. **Note:** If many computers are available in the network, it can take some time to call up the dialog.
Start project	Selection of the start project from a drop-down list. All projects that are currently available in the workspace are displayed.
Start project Runtime folder:	Folder for project files on the target computer. The files of the start project are saved in this folder. All other projects relating to this correspond to the structure of the **Runtime folder** set up on the local computer. **Hint:** Use the project name as folder name in order to create the same structure as on the engineering computer automatically. For example: Project name = **I-Project** at **Start project Runtime folder** enter: *C:\Projects\I_Project* The sub projects in relation to this path are stored at *C:\Projects\Projektname*, for example: The project name is **SubProject1**, then the Runtime folder for this is *C:\Projects\SubProject1*. Requirement: The Runtime folders are left at their default settings and the projects were created at one file level. If this is not the case, it may be the case that subprojects cannot be copied, because the relative folder cannot be created from the start project. **Example:**

Parameter	Description
	The integration project has the following set up as a Runtime folder: *C:\Workspace\Projects\I_Project*. The sub project has the following set up as a Runtime folder: *C:\Subproject*. The start project Runtime folder is set to *C:\Project*. The sub project cannot be transferred, because the relative folder would be..\..*Project*. This does not work, because the Runtime folder for the sub project would be below *C:\*. Solution: Set the **Runtime folder** project property correctly. It is best to do it so that the Runtime folder is at the same level for all projects.

CLOSE DIALOG

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

8.4 Error messages from topological testing

The topology test is always carried out if settings concerning the topology change. The effect of each change can be observed immediately this way. The topology is also tested if the topological view is changed.

TESTS CARRIED OUT

- Is the project defined in the project tree available in the project tree?
- Was a Primary Server defined?
- Were different computers defined for Primary Server and Standby Server?
- Can the client achieve its Primary Server/Standby Server?
- Can the Primary Server reach its clients?
- Can the Standby Server reach its clients?
- Is the Primary Server available for a project in the topology?
- Is the Standby Server available for a project in the topology?
- Is a computer included more than once in the path from Client to the Primary Server?

NOT TESTED:

- Is a client only updated on one path by the Primary Server or do several paths exist?

CLIENT TO SERVER

- Does the client reach its Primary Server via the Primary Server's chain?
- Was a computer that routes switched to its Standby Server?

  Info: The server must also be able to be reached by the client via the project's Standby Server that routes.

ERROR MESSAGE

Errors that are recognized during the topology test are displayed in the result tree (on page 69) in the test result column.

Error	Reason	Solution
The start project is unknown!	Start project cannot be found.	Correct project configuration or include missing project in the workspace.
The computer is entered as a server and standby!	Server and Standby Server must be different computers.	Define different computers as Server and Standby Server.
No computer is entered as server!	The project is a network project but no server was configured.	Define a computer as a server.
The project is not started on computer [name]! However it is necessary because higher hierarchic levels need access to it.	The project is not loaded on the computer stated. The project is however routed via this.	Adapt topology or start project for the computer or deactivate the **Routing active** property.
Circular access to the server: The computer [name] redirects to the client [name]!	The routing path from the client to server goes around in a circle. The computer that acts as a node redirects to the client.	Adapt topology or start project for the computer or deactivate the **Routing active** property.
Circular access to the standby: The computer [name] redirects to the client [name]!	The routing path from the client to the standby goes around in a circle. The computer that acts as a node redirects to the client.	Adapt topology or start project for the computer or deactivate the **Routing active** property.

Error	Reason	Solution
The computer [name] is not included in the list of computers	Computer is missing in the list of computers for the topology.	Add computer to topology.
Not checked because there is critical error in the topology	No check could be made out for this computer because there was a critical error.	Rectify other errors so that the check can also check this computer.
Circular access path: [computer names]	The routing path from the client its the server/standby goes around in a circle. The "computer names" field contains the names of the computer that are affected. Structure: The first computer is always the client. The separator between the computer names indicates whether the following computer is a server or standby. - > identifies the following computer as a server. - "+" labels the following computer as a standby. For example: Computer1+Computer2 > Computer3	Adapt topology or start project for the computer or deactivate the **Routing active** property.
Circular access path exists from server to client!	The server is found according to the node when searching for the client computer.	Adapt topology or start project for the computer or deactivate the **Routing active** property.
The server (name) cannot reach this client!	The path is not closed from the stated server to the client. The client is included in the client list on the server but is not updated. (The blue dots do not disappear on the	Adapt topology or start project for the computer or deactivate the **Routing active** property.

Error	Reason	Solution
	client.)	

9 Redundancy

Redundancy in zenon ensures that processes are not interrupted even in the event of a failure of the Primary Server and that no data is lost. The term used in zenon is **seamless redundancy**.

Seamless redundancy means that the time period between the failure of the Primary Server and detection of the failure is protected from data loss. This is implemented as follows:

1. The Standby Server buffers all data.
2. The Standby Server detects the failure of the Primary Server and automatically takes on the complete functionality thereof.
3. The standby server adds the data from the buffer to the corresponding modules (AML, CEL, archives).

This redundancy also means the online data snychronization of Runtime files. The project status of the current Primary Server applies automatically for all connected Clients and the Standby Server.

The standby server and the connected Clients automatically synchronize the online data. This ensures that:

- all computers have the same project status;
- the Standby Server always saves a copy of a functioning project status.

  In the event of a sudden loss of power to the Primary Server, the files on its drive may be damaged because they were only partially saved. The Standby Server automatically takes on the role of the Primary Server with saved, undamaged files. If the failed server is restarted, it always begins in the role of the Standby Server and thus the online data synchronization of its files is restored.

Information

Project changes must be made on the current Primary Server and not on the shutdown Standby Server.
The server that is started second in the network takes on the project status of the Primary Server. This also applies for the dominant network when starting **Server 1** while **Server 2** is already performing the role of the Primary Server.

Which additional files will be synchronized between the Primary Server and the Standby Server is also configured in **remote transport**.

Example: A controller has two **IP addresses** and the driver is primarily to use one of the IP addresses - which differs depending on whether the driver was started on **Server 1** or **Server 2**.

This is possible if the driver is used which stores these addresses in a configuration file. After the initial transfers of different configurations to **Server 1** and **Server 2**, you can deactivate the *\zenon\custom\drivers* entry in the project properties under **general**/**remote transport**. This prevents the configuration file of the driver of the Primary Server from overwriting the configuration file of the Standby Server during online data synchronization.

SOFTWARE REDUNDANCY AND HARDWARE REDUNDANCY

With redundant systems in zenon, a distinction is made between software redundancy (on page 79) and hardware redundancy (on page 79).

CIRCULAR REDUNDANCY

zenon circular redundancy (on page 102) is a special form of redundancy that allows seamless redundancy for several projects with a low amount of hardware being used and very simple eingineering.

ZENON REDUNDANCY AND ZENON LOGIC REDUNDANCY

You can find details on the combination of zenon redundancy and zenon Logic redundancy in the zenon Logic Runtime manual).

> **Info**
>
> If only one controller is available which offers only one communication channel, the **Stop at the Standby Server** option in the general settings of the driver configuration can be activated. The driver is thus stopped at the Standby Server and only started again at the upgrade.

REDUNDANCY WITH ZENON OPERATOR DEPENDING ON ITS ROLE

For **zenon Operator**, depending on the role, the following is applicable for connections to one server with Supervisor license:

As client

- Connection to **Server 1** and **Server 2** is possible.
- The client connects to the Standby Server if the Primary Server fails.

- Several instances of a driver are supported (because it is started on the Primary Server and/or the Standby Server.
- All variables of all drivers are displayed and can be written.
- Archives can be read.
- Redundancy switching can be executed.

As Standby Server

- Redundancy with zenon Operator (licensed for **Server 1** and **Server 2**) is not supported.
- If Runtime is started with an Operator license as a Standby Server, there is an undefined status.

As a data server

- More than one instance of a driver can be started. However, only one of the instances can send values to Runtime. zenon Operator thus behaves along the lines of a CE panel on which only one driver can be started for technical reasons.

9.1 Types of redundancy

For redundant systems with zenon, a distinction is made between:

- Software redundancy (on page 79)
  The primary server communicates with the controller on a two-way basis; the standby server is read only.
- Hardware redundancy (on page 82)
  Both servers communicate on a two-way basis with the respective connected controller. Is usually applied in conjunction with controllers connected in series.

9.1.1 Software redundancy

The system usually consists of a controller and two redundant computers (primary server and standby server). Both computers must be connected to the PLC.

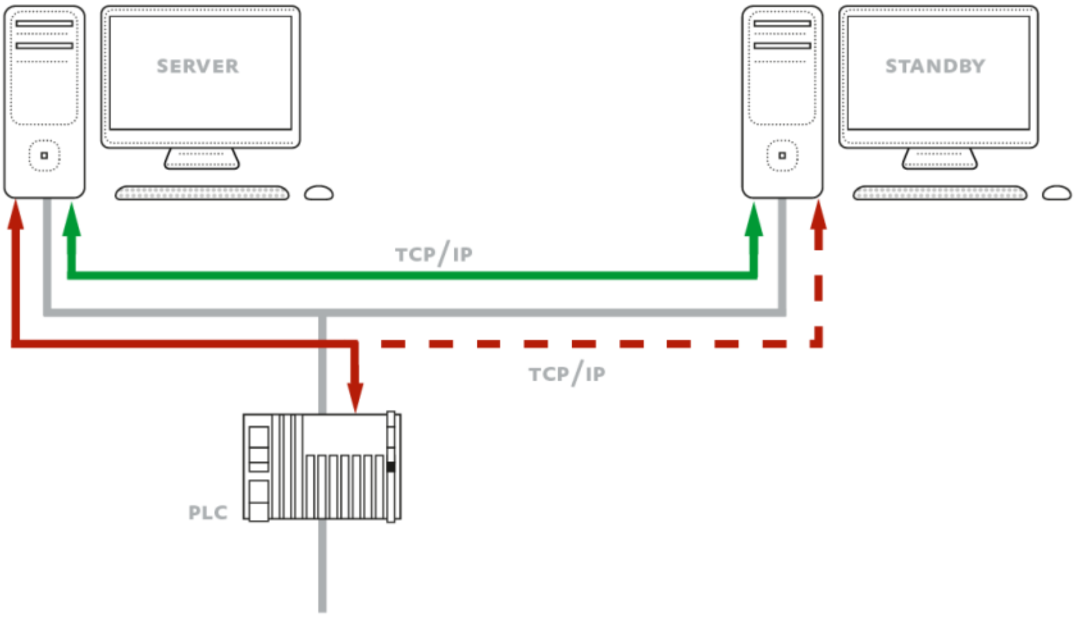
SOFTWARE REDUNDANCY WITH A CONTROLLER

BEHAVIOR IN OPERATION

In normal operation, both computers communicate with the controller; in doing so:

- The Primary Server communicates with the controller in two directions (read and write)
- The Standby communicates with the controller as read only

- Both computers keep the controller's data current and synchronous



ACTION IN THE EVENT OF A FAILURE OF THE PRIMARY SERVER:

In the event of a failure of the Primary Server:

- The Standby Server becomes the new Primary Server
- The Seamless redundancy (on page 97) ensures that all data is complete without omissions, including data from the time between the failure and the switch
- The new Primary Server communicates with the PLC both ways

SOFTWARE REDUNDANCY WITH TWO CONTROLLERS VIA TCP-IP

It is also possible to operate two controllers with software redundancy.

The requirements for this are:

- The driver supports this.
- There is a configuration file that can be manually edited (using a text editor).

To configure, carry out the following steps:

1. Configure the driver for communication with the primary controller with its IP address in the Editor.
2. Transfer the Runtime files to **Server 2** initially.

3. Enter the IP address of the secondary controller into the driver configuration at **Server 2** instead of the the IP address of the primary controller.
4. Deactivate the transfer of the driver files in the project settings for **remote transport**.

   It is thus ensured that the configuration at **Server 2** is not overwritten by the configuration of **Server 1**.

BEHAVIOR IN OPERATION

In normal operation:

- The Primary Server communicates both ways with the first controller
- The Standby Server communicates read only with the second controller
- The integrity of the data is ensured by means of a data sync in zenon

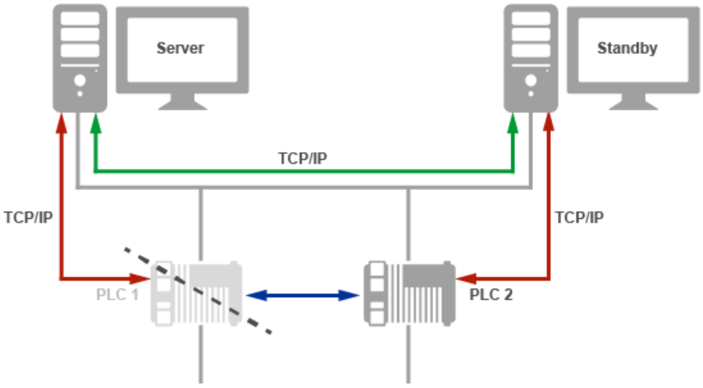
ACTION IN THE EVENT OF A FAILURE OF THE PRIMARY SERVER

If the Primary Server fails, the Standby Server takes over its role and communicates on a two-way basis with the secondary controller.

If the server that has failed is started again, it takes over, depending on the selected **redundancy mode (on page 84),** either the role of the Primary Server again or that of the Standby Server.

ACTION IN THE EVENT OF A FAILURE OF THE FIRST CONTROLLER

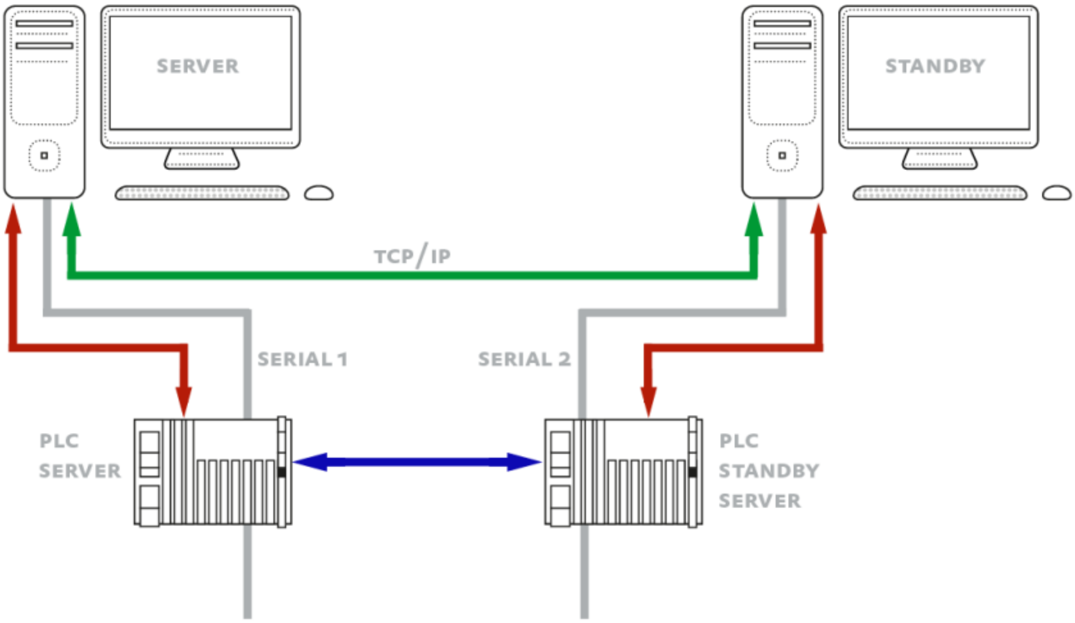
If the first controller fails, redundancy switching is not carried out automatically.



You can implement automated redundancy switching by means of corresponding configuration of **Redundancy mode** - *Evaluated* (on page 85).

9.1.2 Hardware redundancy

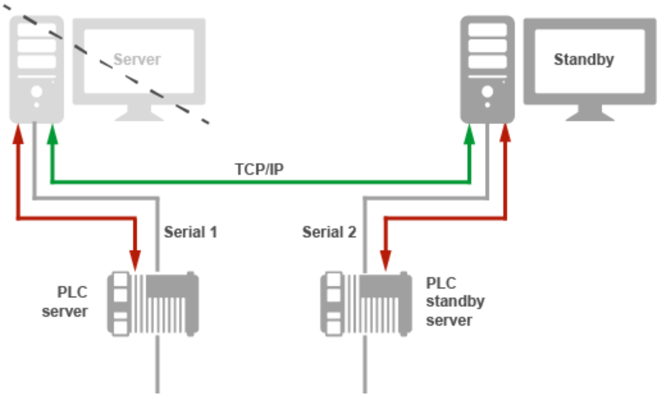
Hardware redundancy must have two controllers and two computers, in contrast to software redundancy. Hardware redundancy is primarily used in conjunction with controllers connected in serial.



In normal operation:

- The Primary Server communicates both ways with the first controller
- The Standby Server communicates both ways with the second controller

ACTION IN THE EVENT OF A FAILURE OF THE PRIMARY SERVER

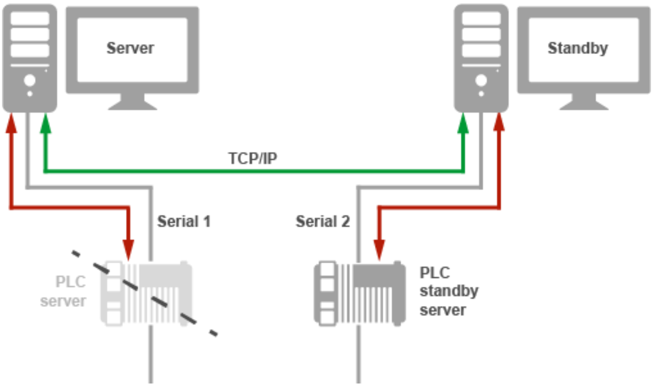


In the event of a failure of the Primary Server:

- The Standby Server becomes the new Primary Server

- Seamless redundancy (on page 97) ensures that all data is complete without omissions, even during the time between the failure of the Primary Server and the switch to the Standby Server

ACTION IN THE EVENT OF A FAILURE OF THE FIRST CONTROLLER



If the controller connected to the Primary Server fails (initially **Server 1**), redundancy switching is not carried out automatically.

To get automated redundancy switching:

1. Create two variables:

   a) Variable A, which is read by the Primary Server's controller

   a) Variable B, which is read by the Standby Server's controller

   Attention: To do this, activate the option for Variable B: **Read from Standby Server only**.

2. Create two functions:

   a) A function for redundancy switching (on page 145) at **Server 2**

   b) A function for redundancy switching on **Server 1**

3. Create two binary reaction matrices, which each check the **INVALID** status for the value *1*.

   a) For the first reaction matrix, link the function for redundancy switching to **Server 2**

   b) For the second reaction matrix, link the function for redundancy switching to **Server 1**

   Info: You can find more information on the creation and configuration of reaction matrices in the following chapter: Reaction Matrices

4. Link:

   a) The first reaction matrix to Variable A

   b) The second reaction matrix to Variable B

If, during ongoing operation, the controller of the Primary Server fails (Variable A gets the status INVALID), redundancy switching to **Server 2** is carried out.

If the PLC on the original Standby Server (**Server 2**) fails, redundancy switching to **Server 1** is carried out.

To get redundancy switching after reaching the controller again:

1. Add an entry to the reaction matrix in which the **INVALID** status is checked for the value *0*.
2. Link the corresponding function to redundancy switching.

9.2 Redundancy modes

If *software redundancy* (on page 79) is selected as a redundancy type, there are three different redundancy modes available. The distribution of roles (Primary Server or Standby Server) during operation and after the re-availability of a server that failed beforehand depends on the selected redundancy mode.

A distinction is made between

- *Dominant*
- *Non-dominant*
- *Rated*

THE BEHAVIOR IN DETAIL

Dominant

1. The Primary Server fails.
2. The original Primary Server goes online again.
   In doing so:
   a) It connects to the current Primary Server
   b) It synchronizes all data
   c) It becomes the Primary Server itself again
3. The original Standby Server becomes the Standby Server again.
4. All Clients connect to the new Primary Server.

Non-dominant

1. The Primary Server fails.
2. The original Primary Server goes online again.
   In doing so:
   a) It connects to the current Primary Server
   b) It synchronizes all data
   c) It becomes the Standby Server itself

3. The current Primary Server remains the Primary Server.
4. All Clients retain the connection to this.

Rated

With rated redundancy, the roles of Primary Server and Standby Server are given on the basis of an evaluation matrix.
In doing so:

1. both computers each carry out a rating calculation on the basis of configured rating criteria.
2. The computer that has the higher rating becomes the Primary Server
3. There is no exchange of roles if the evaluation calculation for both servers results in the same value.
4. Alarms and CEL entries are written by the Primary Server
5. The clients connect to the Primary Server

9.2.1 Redundancy in a rated network

With hardware redundancy in a rated network, rating criteria (on page 87) decide which computer takes on the role of the Primary Server and which takes on the role of the Standby Server.

This rating can be freely configured and can include different criteria. Each criterion is assigned rating points - the weighting. The sum of the weighting points decides on the respective server role.

Information

If one of the two servers, regardless of which role they are currently in, loses the connection to a different server (due to a hardware or network failure, for example), it automatically upgrades itself to the Primary Server.

REASONS FOR DELAYING THE REGRADING

The following causes can delay a redundancy switching:

- The internal modules have not yet been fully initialized. This is possible if a driver delays the regrading process.
- The regrading is already active but not yet completed.
- The file sync is still active.
- Reloading is taking place.

> ⚠ **Attention**
>
> If, in a rated network, server redundancy switching is triggered due to a rating, there is no guarantee that pending functions in the queue of the old Primary Server have also actually been processed successfully before switching.
>
> **Reason**
>
> During redundancy switching in rated networks, both servers take on the role of the Primary Server for a short time. In this time period, **Server 1** and **Server 2** synchronize their data. The time period of this status depends on the network load and can be between 200 and 500 milliseconds.

TIME SYNCHRONIZATION

Recommendation: With a rated network, always deactivate the time synchronization of **Server 1** and **Server 2** through the zenon network.

To do this:

- Deactivate the automatic time synchronization in zenon.
- Activate external time synchronization on **Server 1** and **Server 2** by means of the operating system.

> **Information**
>
> You can find detailed information on this in the Time synchronization in the network (on page 18) chapter.

9.2.1.1 Redundancy in a rated network

The following properties of the **Network** properties group must be configured for the configuration of a rated network:

1. **Network active** must be activated.
2. **Server 1** and **Server 2**
3. **Redundancy type** is *Software redundancy*
4. **Redundancy mode** is *Rated*
5. Redundancy rating in the **Valuations** property field.
   You can find details on rating in the chapter "Configuration of redundancy rating (on page 87)".
6. **Switching delay [s]**

7. **Dead time after switching [s]**
8. **Hysteresis** (the rating points)

9.2.1.2 Configuration of redundancy rating

The dialog to configure the redundancy rating is opened by clicking on "**...**" in the **Valuations** properties field in the **Network** properties group.
The result of this rating is evaluated with the system variables **[Network] Rating result for Server 1** and **[Network] Rating result for Server 2**.
Note: You can find further information on the network system variables in the System driver manual in the **Theme [Network]** chapter.

VARIABLES FOR EVALUATION

The following are particularly suitable for evaluation:

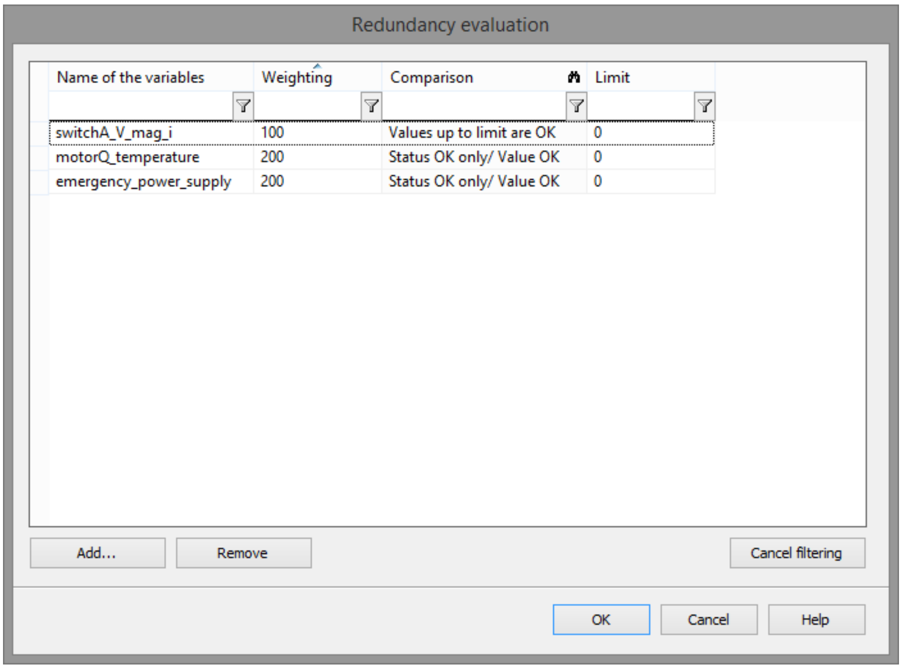
- Process variables: To detect the loss of connections to a PLC.
- Local variables: For information on resources of the PC. For example, with system driver variables from the **[HW resources] theme** (**sysdrv.chm::/25958.htm**).
- Internal variables with the *Local* setting for the **Calculation** property.

Attention: Evaluations that are based on the variables of the math driver, or on variables whose drivers have been stopped on the standby server, always only have an effect on the current primary server.

Math variables and process variables whose drivers have been stopped on the standby server never update the evaluation on the standby server. For these variables, the weighting calculated in the evaluation contains the last value from the time when this server still acted as the primary server.

REDUNDANCY RATING DIALOG

The fields in this dialog can be filtered and sorted.



Name of the variables	Weighting	Comparison	Limit
switchA_V_mag_i	100	Values up to limit are OK	0
motorQ_temperature	200	Status OK only/ Value OK	0
emergency_power_supply	200	Status OK only/ Value OK	0

Option	Description
Name of the variable	List of the variables that can be used for the rating. **Note:** A maximum of 200 variables can be used for the redundancy rating.
Weighting	Value of the test criterion. Provides the value of the individual test criterion in the entirety of the test results. The weightings of all variables that are applicable (see **Comparison**) are counted up. - Input range: *0 - 1000* Default: *100*
Comparison	States the type of comparison, from which the value of the weighting is taken into account in the overall assessment. Drop-down list: - *Not used* No comparison to the weighting is carried out. - *Only status OK /value OK*

Option	Description
	A check to see if there is a valid value. As soon as there is a value without the status *INVALID*, the value of the weighting is used for the overall evaluation. **Note:** It allows a check to see whether there is a connection to the controller - using any desired variable that the PLC provides. ▶ *Values from limit are OK* Values that are greater than or equal to the limit that has been entered use the value of the rating in the overall evaluation. ▶ *Values up to limit are OK* Values that are less than or equal to the limit that has been entered use the value of the rating in the overall evaluation. Default: *not used*
Limit	Limit for comparison with *Values from limit are OK* or *Values up to limit are OK* Default: *0* **Note:** If the comparison is "*not used*" or "*Only status OK /value OK*", entries in this field have no influence on the rating.
Add...	Opens the dialog to select variables.
Delete	Removes the selected variables from the list **Note:** Multiple selection is possible.
Cancel filtering	Removes the filter criteria.

CLOSE DIALOG

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

Example

A variable is configured in the rating dialog with the following settings:

- Weighting: *50*
- Comparison: *Values from limit are OK*
- Limit: *50*

Set the value of the variable to 55. The condition becomes true as a result and the weighting of 50 is added to the result of the **[Network] Rating result for Server 1** system variable.

A further variable is configured with the following settings:

- Weighting: *30*
- Comparison: *Values up to limit are OK*
- Limit: *10*

As long as the value of this second variable = < *10*, the weighting is added to the system variable. If the value of the second variable is *5* for example, the **[network] rating result for Server 1** is *80*. This is also applicable if the previously described variable is true.

9.2.1.3 Switching delay, down time and hysteresis

SWITCHING DELAY

The **Switching delay [s]** is the time (tolerance time) in which value changes of the rating result do not trigger a regrading of the server. In doing so, short peaks or a brief failure do not lead to an immediate server change.

Note: A further improvement of the rating result during the configured switching delay does not reset the timer.

Example

If a **Switching delay [s]** of *30* seconds is configured, the rating on the current Standby Server must be at least 30 seconds better than that of the Primary Server in order to trigger regrading.

After successful regrading, the timers of **Dead time after switching [s]** and **Switching delay [s]** switching delay can run at the same time; both must have expired in order to trigger a regrading.

DOWN TIME

This setting prevents resetting on the basis of the Primary server during the configured time.

Note: The **Redundancy switch** function can be executed.

Example

If the rating of the current Standby Server - after the last regrading within the set down time - is higher than that of the current server, there is not another regrading until this time expires. The start of the time period is the time in which the role swap of the server is completed.

If the rating increases within the down time of the Standby to more than that of the Server, but it goes down again to below or the level as the rating of the server before it expires, there is no regrading.

HYSTERESIS (IN RATING POINTS)

If a **Hysteresis** is configured, a check is made before redundancy switching to see whether the rating results between the two computers (**Server 1** and **Server 2**) correspond to the configured rating points. The switching is triggered once the value has been reached.

Example

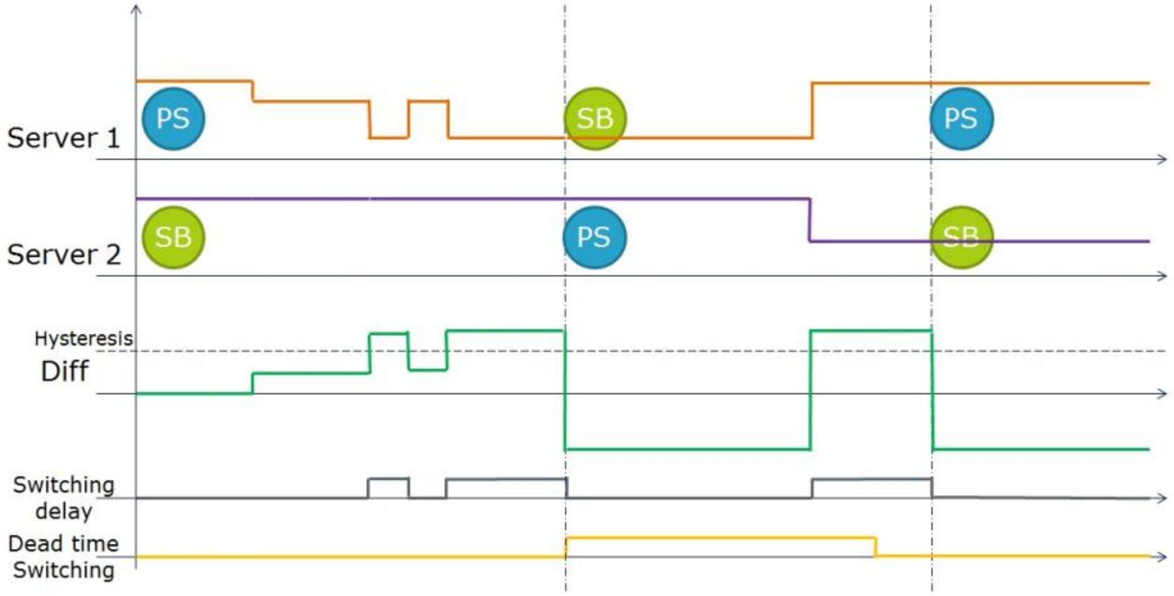
Server 1 is Primary Server.

Server 1 and **Server 2** each have rating result *0*.

The **Hysteresis** is configured with value *100*.

If the rating changes from **Server 2** to *99*, nothing happens; if the value reaches *100*, a regrading is triggered.

9.2.1.3.1 Switching delay - example

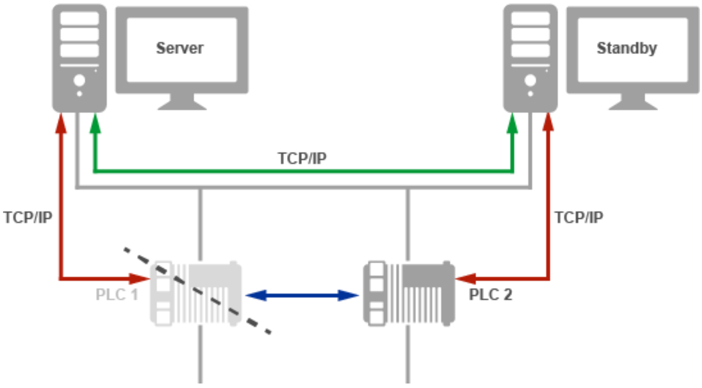


In this example, the switching of the two servers is triggered by a higher rating of the **Server 2**.

- Once the reason for switching has occurred, the delay time is waited.
  Server 2 then becomes the Primary Server.
- Although **Server 1** could take on the role of the Primary Server again, the delay time is waited for before **Server 1** takes on the role of the Primary Server.
- **Server 2** takes on the role of the Primary Server once the difference of the rating system variables is higher than the hysteresis of the delay time. After switching, the counting of the configured **Dead time after switching** is activated.
- **Server 2** takes on the role of the Standby Server at the time at which the difference is higher than the hysteresis again, once the switching delay time has expired.

Note: The role exchange of the two servers is visualized in the graphics by the respective vertical dashed line.

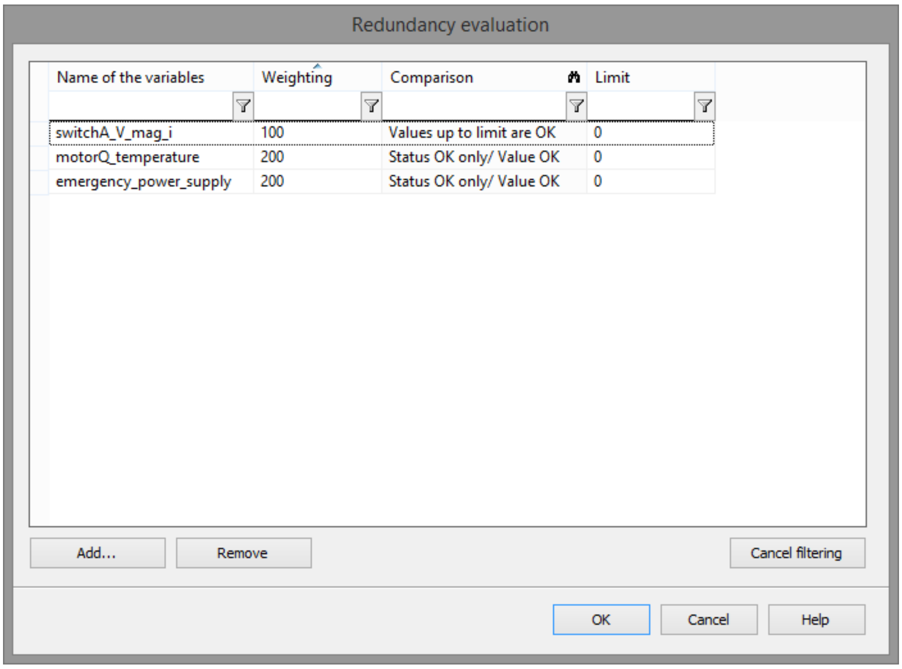
9.2.1.4 Example: Configuration for software redundancy with 2 controllers



CONFIGURATION IN THE EDITOR

To get automatic redundancy switching in the event of a failure of a controller:

1. Create a configuration as described in **software redundancy** (on page 79) - software redundancy with two controllers via TCP-IP.
2. Create a variable from the corresponding driver.
3. Add this variable in the dialog for the **redundancy rating** (on page 87).


Redundancy evaluation

Name of the variables	Weighting	Comparison	Limit
switchA_V_mag_i	100	Values up to limit are OK	0
motorQ_temperature	200	Status OK only/ Value OK	0
emergency_power_supply	200	Status OK only/ Value OK	0

4. For **Comparison**, configure *Status OK only / value OK* with a corresponding weighting.

BEHAVIOR IN THE RUNTIME

1. If the controller of the Primary Server fails or the connection to the controller is interrupted, the variable gets the **INVALID** status and the rating falls by the configured value.

2. On the Standby Server, at the time of the failure, the **INVALID** status is shown visually, but the value from the standby buffer is used internally for the evaluation. This value has a valid status because the Standby Server reads from the second controller.
3. Once the **switching delay** (on page 90) has expired, the Standby Server takes on the role of the Primary Server and applies the valid values from the Standby buffer.

Information

To calculate the redundancy rating, the status of values from the buffer of the Standby Server is used. This is an exceptional case because the status of the Primary Server is always used for other evaluations.

9.2.1.5 LOG entries in the rated network

The following log entries are written in the Diagnosis Viewer for the rated network:

LOG entry	Debug Level	Description
LOG SendData Project:RATED_NET To:CDSBG079.COPA-DATA.INTERNAL Modul:10 Prior:0 Class:CD_CSystemDaten	**Debug**	Data from the CD_CSystemDaten class is transported via connection 0 (watchdog) to computer *cdsbg079*. The module gives the zenon module to the one that has the data. *Module 10:* Synchronous execution network module
LOG SendData Project:RATED_NET To:W7X64 Modul:10 Prior:0 Class:CD_CSrv1SystemDaten	**Debug**	As above, the target is *Server2* and the source is *Server1*
LOG ReceiveData Project:RATED_NET To:S Modul:10 Class:CD_CSrv2SystemDaten	**Debug**	As above, the target is *Server* and the source is *Server1*
<Beschreibung> Prj:<ProjektName> client:<ClientRechner> Cmd:<CommandTxt>(<CommandNum>) Timeout:<Timeout>, ReqId:<ReqId> reload:<Reload> stat1:<SubCommand> stat2:<Stat2> string:<AddText> list:<ItemCnt>		Description of the function for which the message is created. e.g.: Server received system data.; Client send watchdog; etc. *ClientRechner:* Source or target depending on direction *CommandTxt:* Command as plain text. e.g.:

LOG entry	Debug Level	Description
		Server_REQ_DateiListe *CommandNum:* Numerical identification of the command: ▸ *0* = Server_REQ_LifeMsg, ▸ *1* = Server_REQ_DateiListe, ▸ *2* = Server_REQ_GetDatei, ▸ *3* = Client_REQ_UpdateProjekt, ▸ *4* = Server_REQ_RedundanzUmschaltung, ▸ *5* = NETSRV_ConnectionClosed, ▸ *Timeout*: Timeout in ms *ReqId:* Request ID *Reload: 1* if Server is in Reload, otherwise *0* *SubCommand:* Number of subcommand. Meaning depending on CommandNum For Command 0 ServerReqLifeMsg: ▸ *1* = STAT_CLIENT_ABGEWIESEN, ▸ *2* = STAT_CLIENT_ANGENOMMEN ▸ *3* = STAT_CLIENT_SERVERCLOSE ▸ *4* = STAT_CLIENT_SERVERSWITCH ▸ *5* = STAT_CLIENT_SB_ANGENOMMEN ▸ *6* = STAT_SB_SERVER_TO_SB ▸ *7* = STAT_PEER_ALIVE For Command 4 redundancy switching: ▸ *0* = IdleChangeOfDominance ▸ *1* = AdviseChangeOfDominance ▸ *2* = ConfirmChangeOfDominance *Stat2:* Additional command-specific

LOG entry	Debug Level	Description
		information. e.g.: With command Server_Req_LifeMsg, subcommand: STAT_CLIENT_ANGENOMMEN, response for a Client logging on to a Server is the project version number in HiWORD. AddText: Additional text. ItemCnt: Number of objects in the list.
Client send watchdog. Prj:RATED_NET client:ATSZG-WKS053.COPA-DATA.INTERNAL Cmd:Server_REQ_LifeMsg(0) Timeout:30000, ReqId:0 reload:0 stat1:2(STAT_CLIENT_ANGENOMMEN) stat2:1 string: list:0 quantifier:<Quant> role:<RoleTxt>(<RoleNum>)		Message from **Server 1** to Server2 *Quant:* Current rating. *RoleTxt:* Current role as text *RoleNum:* Current role as number
Server received system data. Prj:RATED_NET client:WKS053-W7X64 Cmd:Server_REQ_LifeMsg(0) Timeout:30000, ReqId:0 reload:0 stat1:2(STAT_CLIENT_ANGENOMMEN) stat2:1 string: list:0 expSrv1role:<ExpRoleTxt>(<ExpRoleNum>)		*ExpRoleTxt:* Role expected by Server2 for **Server 1** as text. *ExpRoleNum:* Expected role as number
Manual role switch for project %s denied.	**WARNINGS**	Manual redundancy switching cannot be carried out because a regrading is already active.
Current evaluated server1 role: %s(%i) Performing manual role switch.	**DEBUG**	A manual redundancy switch is carried out. **Server 1** must switch its role according to the message.
Unknown standby logon! Project:<PrjName> new client <Pc>!	**ERROR**	Message is logged if a Client logs on as a Standby Server, but the Client name does not correspond to that of **Server 1** or **Server 2**. The identification is carried out through the host name. *PrjName:* Project that logs on to the Standby Server. *PC:* Name with which the Standby Server logs on.

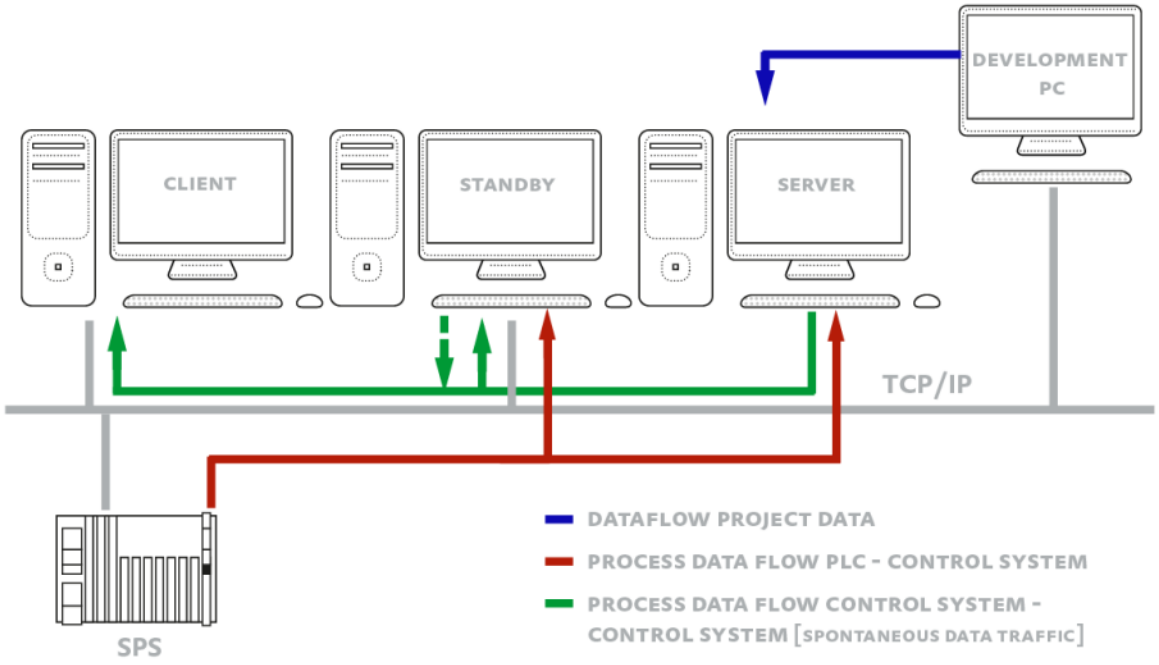
LOG entry	Debug Level	Description
Unknown standby logon! Project:<PrjName> new client <Pc>!	**ERROR**	Message is logged if a Client logs on as a Standby Server, but the Client name does not correspond to that of **Server 1** or **Server 2**. The identification is carried out through the host name. *PrjName:* Project that logs on to the Standby Server. *PC:* Name with which the Standby Server logs on.

9.3 Configuration of seamless redundancy

For the configuration of seamless redundancy for a zenon project, you need two (conventional) computers.

Define, in the project properties:

1. A computer as Primary Server (on page 25) for the project
2. A computer that is to be the Standby Server (on page 98) for the project.


DEVELOPMENT PC
CLIENT
STANDBY
SERVER
TCP/IP
SPS
DATAFLOW PROJECT DATA
PROCESS DATA FLOW PLC - CONTROL SYSTEM
PROCESS DATA FLOW CONTROL SYSTEM - CONTROL SYSTEM [SPONTANEOUS DATA TRAFFIC]

PROCEDURE FOR SEAMLESS REDUNDANCY

The process of seamless redundancy is implemented as follows:

- One computer is the Primary Server and one computer is the Standby Server for a project.
- As in a normal Client-Server model, the Primary Server has ownership of all data.
- The Standby Server acts outwardly to the user like any other computer on the network that has started the project.
- The Standby Server independently records all historical data such as alarms, events and archives and synchronizes other data (recipes, users, etc.) with the Primary Server.
- If the Primary Server fails, the Standby Server upgrades itself and takes over its tasks.

  In order to avoid data loss in the time period between the server failure and the detection of the failure, the standby always buffers all data.
  After a failure of the Primary Server, this buffer is merged with the last data from the server and the new incoming data, so no data can be lost.
- All Clients connect to the new Primary Server.
- If the original Primary Server goes online again, a distinction must be made between when redundancy mode is used.

  Find out more information in the chapter **Redundancy modes** (on page 84).

> **Info**
>
> Project changes need only be entered on the Primary Server; the standby server and the connected clients automatically synchronize online data. This ensures that the project status is the same on all computers.

ZENON REDUNDANCY AND ZENON LOGIC REDUNDANCY

You can find details on the combination of zenon redundancy and zenon Logic redundancy in the zenon Logic Runtime manual).

9.4 Setting up the Standby Server

To set up the Standby Server:

1. In the project properties in the zenon Editor, switch to the **Network** group.
2. Enter, in the **Server 2** property, the name of the computer that is to serve as the Standby Server for the project.
   Note: the computer must have a connection to the PLC.

   You can enter the computer name:

a) By selecting from the drop-down list by clicking on the **...** button
The **Select the desired computer** dialog is opened with all computers available in the network.
Note: This process can take a very long time, depending on the number of computers available in the network.

b) By entering the server name in the input field

3. Select, in the **Redundancy type** property, the desired type of redundancy (on page 79) from the drop-down list:

> **Information**
>
> Servers from different domains are permitted!
> In this case, configure the server name including the domain name.
>
> e.g.: *terminal_01.mydomain.net*

Size of the standby buffer

The size of the standby buffer depends on the project configuration. In doing so, four thirds of the configured network timeout is always buffered.

The project configuration of this network timeout is configured in the zenon6.ini file in the **[Netz]** area with the *NET_TIMEOUT_MSEC=* entry. In doing so, all data is saved in the buffer on the standby server.

In addition, this INI entry determines the time period that the standby server waits before it upgrades itself to the primary server.

9.5 Special setups in the communication between Primary Server and Standby Server

Note the rules in the following setup:

- The Primary Server has failed
- The Standby Server has taken on its role
- The original Primary Server is restarted
- The original Primary Server gets all Runtime files from the current Primary Server (originally the Standby Server)

In exceptional cases, there may be conflicts if:

1. Changes have only been made to the project of the original Primary Server that has been stopped
2. It is not clear, due to network problems, which computer is the Primary Server.

1. PROJECT CHANGES WITH THE PRIMARY SERVER STOPPED

If, during the time in which Runtime is stopped on the original Primary Server, you make changes to the project and introduce these before synchronizing on this (stopped) computer, these changes are overwritten again as soon as the original Primary Server gets the data from the current Primary Server (the original Standby Server).

To prevent this: Introduce the amended data onto the Primary Server (configured Standby Server) as well, before starting the original (configured) Primary Server.

2. PROJECT CHANGES IN REDUNDANCY MODE OR THE SERVERS USED

The following changes are only accepted after the Primary Server has restarted:

- Change of the **Redundancy mode**.
- Changes in the **Server 1** properties or **Server 2** in the **Network** properties group.

> **Attention**
>
> In this case, reloading the project in Runtime is not sufficient to apply the necessary changes! To accept all changes, restart the Primary Server.

3. SERVER ROLE NOT CLEAR DUE TO NETWORK PROBLEMS

In exceptional cases, it may happen that both computers are the Primary Server. The cause of this can be, for example, a loss of network connection due to a switch failure, a disconnected network cable etc. In this case, the communication between the Primary Server, Standby Server and Clients depends on the error screen.

If, with this setup, the error screen is resolved and both servers communicate with each other again (at this time), then the original (configured) server has data sovereignty. That means: The standby server's more recent data could be overwritten.

To prevent this:

1. Always check the role distribution with the **[Network] Current Primary Server** system variable: You will see the role that Runtime has and discover duplicate Primary Servers.
2. End zenon Runtime on the Primary Server that lost the network connection.
3. Set up the network connection again.
4. Restart zenon Runtime on this computer.
5. Runtime then starts the project with the computer as a Standby Server, updates its data and only then switches back to the Primary Server role.

Hint: Monitor the network connection with the Redundancy Management Tool (on page 103).

Attention

PNG graphics files cannot be overwritten if they are currently being displayed in Runtime.

Background: The Runtime protects opened **.png** files. This prevents these being overwritten.

Solution: Before Remote Transport is instigated, it must be ensured that screens with ***.png** files:

- Are not called up in Runtime
- Are not being used by another program

This also applies for the reloading of amended Runtime files. The Runtime sync in the network does not work for a ***.png** screen if this is switched on a zenon computer that is involved in the process (standby server, client).

9.6 Integrated rating methods for redundancy switching

In the dominant network, the **Server 2**, as the configured computer, regularly sends a telegram to the (dominant) **Server 1**, so that this - as soon as it receives a connection - can take on its role again. In a rated network, no telegram is sent any more in this case.

If there are two servers, the rating decides on the role.

USE OF DOMINANT BEHAVIOR IN ZENON LOGIC

- A driver can delay upgrading to a server on the dominant server. This function is no longer present in a rated network with severs that have equal rights.
- Archives also receive an extra byte per entry on the Standby Server as well as on the **Server 2**. This ensures that the archives are different and are included in a synchronization.
- Furthermore, the network topology for the drivers uses the terms Dominant and Standby as well as the roles as Server or Standby. However only the role is evaluated, so no change is therefore necessary.
- zenon Logic is supplied with information on the label and role via **CStratonVM::UpdatePrjSTates**. The only thing that is evaluated is whether a computer is configured as a Standby and not the Primary Server. in this case, zenon Logic is also not active.

9.7 zenon circular redundancy

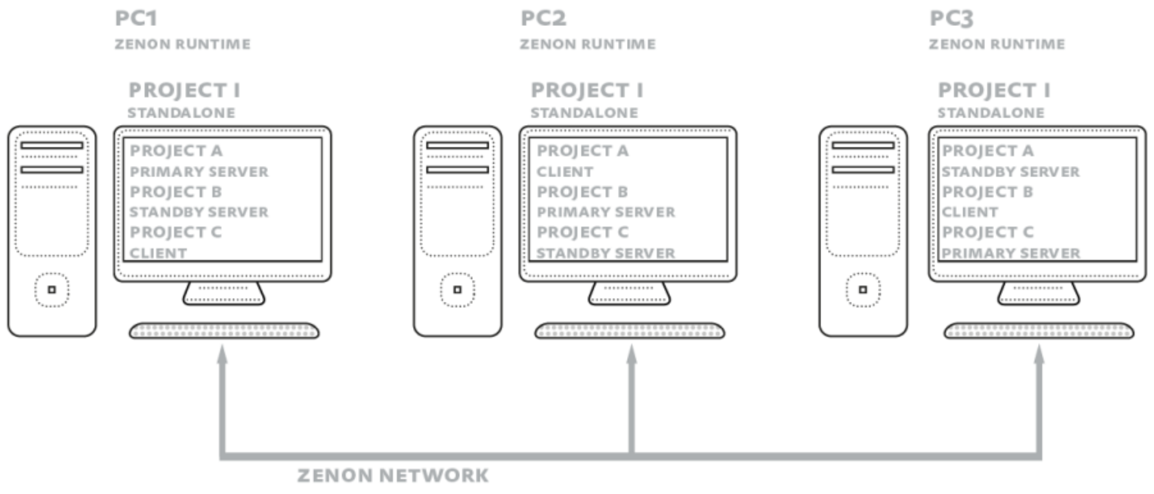
zenon circular redundancy allows seamless redundancy for several projects with a low amount of hardware being used.

Two computers are normally required for each redundant project: one computer as the Primary Server and one computer as the Standby Server. 3 projects would thus require 6 computers. Just three computers are sufficient for implementing three projects with zenon circular redundancy. Another computer is added for each further project. zenon combines multi-project administration (on page 30) with seamless redundancy (on page 77).

CONCEPT OF CIRCULAR REDUNDANCY

Circular redundancy uses the possibility of multi-project administration. Several projects can run simultaneously on one computer. Each computer is the Primary Server for a project and the Standby Server for a second "neighboring project" and can also be the Client for further projects. This results in a circle. Instead of four computers, for example, and licenses for two projects, six for three or eight for four, you only need half of that.

Topology with three projects



- Computer 1 is the Primary Server for Project A and Standby Server for Project B.
- Computer 2 is the Primary Server for Project B and Standby Server for Project C.
- Computer 3 is the Primary Server for Project C and Standby Server for Project A.

  The circle is closed!
- Each computer can be a Client for the other projects at the same time.
- Expense: 3 computers and 3 Runtime licenses

> **Info**
>
> An integration project (on page 35) is needed to start more than one project on a computer.

Normally you would need six computers and six Runtime licenses in this example. zenon circular redundancy is of course not limited to three projects, but can connect as many projects as desired in a circle. The fact that the computers can also be clients for other projects allows easy implementation of a low-cost, fail-safe, highly-available production line.

TIME SYNCHRONIZATION FOR ZENON CIRCULAR REDUNDANCY

If zenon time synchronization (on page 18) is active, the Standby Server and clients always receive the current time from the Primary Server. This makes no sense when using zenon circular redundancy, because the individual PCs are Primary Server and Standby Server at the same time. Computer 1 for example, would thus obtain the time from computer 2, computer 2 would obtain it from computer 3 etc.

Recommendation: In this case, deactivate the zenon time synchronization and carry out external time synchronization. You can find instructions for this in Time synchronization in the network (on page 18).

9.8 Redundancy Management Tool

The **Redundancy Management Tool** monitors the network adapter and its connection to the network. If the device loses the connection to the network, e.g. by removing the network cable, the **Redundancy Management Tool** stops the Runtime. This process can be canceled by the operator within a configurable period of time. If the connection to the network is reestablished, the **Redundancy Management Tool** restarts the Runtime.

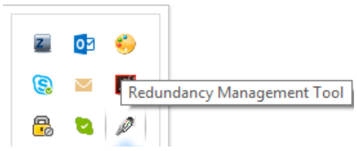
START AND CONFIGURATION

The **Redundancy Management Tool** can be configured as an application with a dialog or by means of the command line interface.

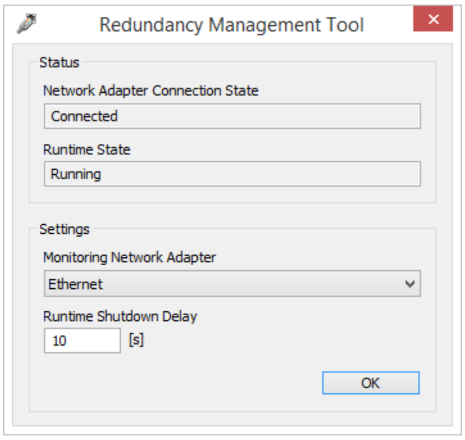
To open the dialog:

- Using the Windows start folder:
  Start -> All programs -> COPA-DATA -> Tools 8.10 -> **Redundancy Management Tool**
- Using the Startup Tool:
  Tools -> **Redundancy Management Tool**

- Direct start of the **zenon_redman.exe** file from the zenon program folder



After the start, the **Redundancy Management Tool** is also displayed as a symbol in the right area of the Windows task bar. A double click on the symbol opens the configuration dialog:



Network Adapter Connection State: Connected

Runtime State: Running

Monitoring Network Adapter: Ethernet

Runtime Shutdown Delay: 10 [s]

STATUS

Status information on the monitored network adapter.

Parameter	Description
Network Adapter Connection State	Information about the status: - *Connected* Connection to the network established. - *Disconnected* Connection to the network interrupted.
Runtime State	Status of the zenon Runtime - *Running* Runtime is running. - *Stopped by Redundancy Management Tool*: Runtime was closed by the tool. - *Stopped* Runtime is not running.

SETTINGS

Configuration of the network adapter to be monitored

Parameter	Description
Monitoring Network Adapter	Selection of the network adapter which should be monitored from the drop-down list. The list shows all network adapters available on the computer.
Runtime Shutdown Delay	Setting of the delay time before ending Runtime, in seconds. In this period of time, the operator can cancel the closing of the Runtime. Default:*10 s* Maximum value: *2147483647 s* Values above this are interpreted as *0*.
OK	Applies settings and closes the dialog. Settings are applied in the INI file.

INI FILE

During the configuration via the dialog, the RedMan.ini file is created in the %ProgramData%\COPA-DATA\System path.
It contains the following entries:

Parameter	Description
[DEFAULT]	Group in the INI file.
ADAPTER=	Selected LAN connection.
DELAY=	Value for delay time.

COMMAND LINE INTERFACE:

The **Redundancy Management Tool** can also be started via the command line interface.

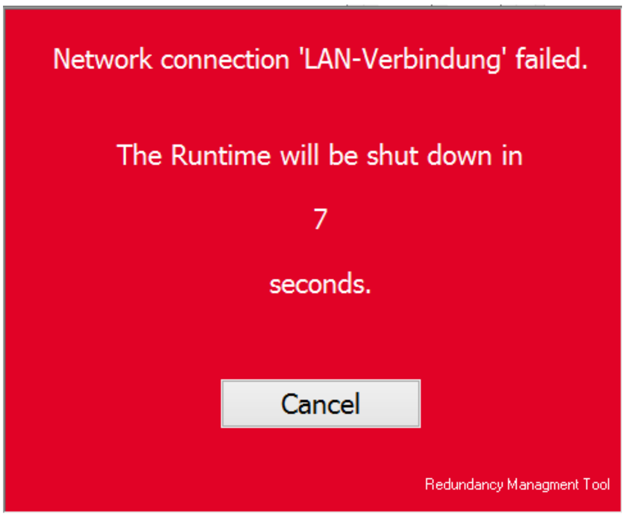
Possible parameters:

- **ADAPTER=***[Name]*
  Defines the network adapter which should be monitored.
- **DELAY=**[Seconds]
  Defines the waiting time after a connection loss.
  Maximum value: 2147483647. Values greater than this are interpreted as 0.
- **HELP,?**
  Displays help about the command line parameters.

> **Information**
>
> During the configuration via command line interface:
>
> - these settings are taken over directly
> - the configuration is deactivated in the dialog
>
> no INI file is written

IN THE RUNTIME

During the Runtime, the **Redundancy Management Tool** continuously monitors the network connection. If the connection is canceled, the **Redundancy Management Tool** shows a warning message. The Runtime is ended after the configured delay time.



Network connection 'LAN-Verbindung' failed.

The Runtime will be shut down in

7

seconds.

As soon as the connection is available again, the **Redundancy Management Tool** restarts the Runtime.

Click on the **Cancel** button to halt the countdown and prevent the closing of the Runtime. If the connection is reestablished, the dialog is displayed again when the connection fails again. The user can cancel again or let the tool close the Runtime.

> **Information**
>
> The current status of the connection and Runtime is also always displayed in the configuration dialog.

ERROR HANDLING

ERROR MESSAGE

Errors are displayed by pop-up messages.

Message	Meaning
GetAdapterAddresses not supported on this platform! Error code '%u'!	Operating system version is not supported
GetAdapterAddresses did not return information about network adapters. Error code '%u'!	No network adapter found.

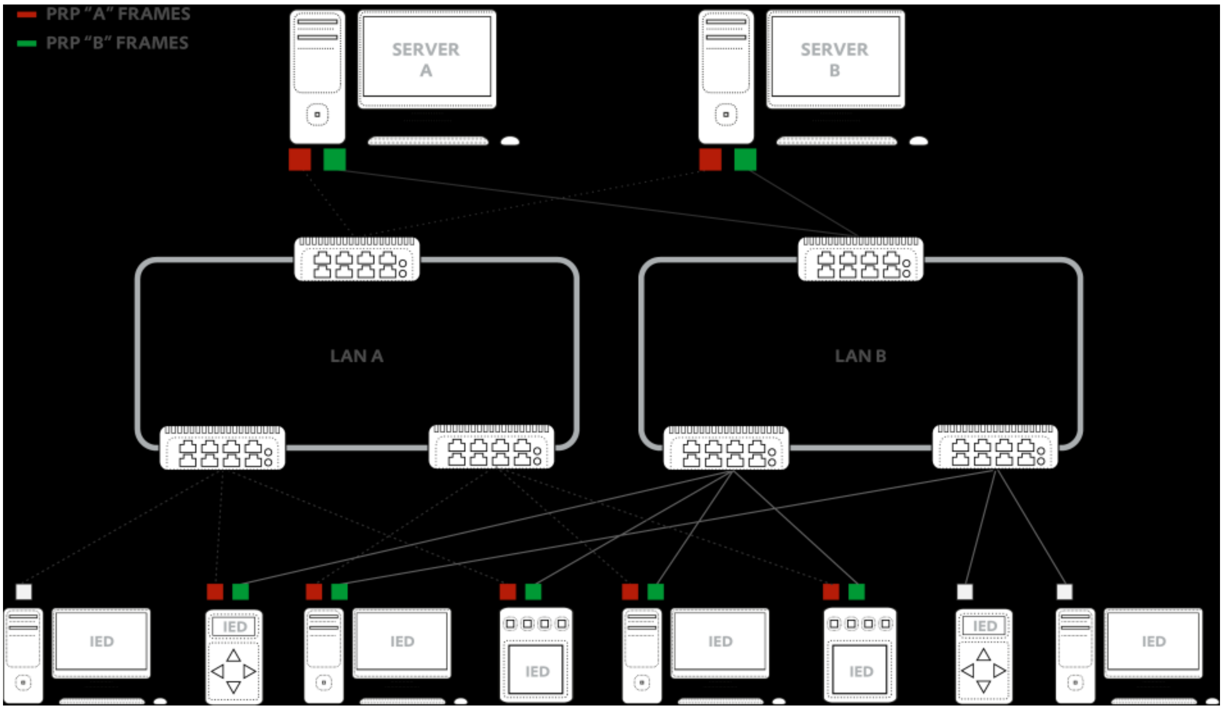
LOG FILES OF THE DIAGNOSIS VIEWER

The following is documented in the log file of the Diagnosis Viewer:

Entry	Debug Level	Meaning
Network link '%s' down for '%u' seconds. zenon Runtime will be terminated.	Error	Network connection failed: The Runtime is closed.
Network link '%s' is up. Restarting zenon Runtime now.	Information	Network connection available again: The Runtime is restarted

10 COPA-DATA PRP

zenon supports the Parallel Redundancy Protocol (PRP) for hardware-redundant communication in an Ethernet network. The protocol is standardized in IEC 62439-3.



Every PRP node, a Dual Attached Node (DAN), is bound to two networks, here *LAN A* and *LAN B*. As PRP is a Layer-2 protocol, the protocol of the two networks must be identical on the MAC level. Therefore, this basically means that there must be no direct connection between the LANs.

If one of the networks fails (e.g. if a cable is removed for one of the networks), this has no influence on the other one. The communication between PRP nodes remains seamless as long as one of the LANs is connected.

The PRP communication is established directly on the OSI Layer 2 level, thus making it applicable for TCP, UDP, Ethernet Multicast (e.g. GOOSE) etc. Its use is independent of the zenon Editor and the zenon Runtime. No special configurations are required in zenon, and the use of PRP is "transparent" for <CD_PRODUKTNAME> modules, drivers, process gateways etc. The settings are only made in the components of the operating system.

To use the protocol, the computer must have two network cards and be configured accordingly. You need the following for the use of PRP:

- *COPA-DATA PRP driver* network service (for the network cards of the computer).
- **PRP configuration and diagnosis tool** application.

You can find this on the installation medium. You can find a detailed description of the required configuration steps in this manual in the installation and configuration (on page 110) chapter.

10.1 System requirements

PRP communication is supported for 100 Mbit/s Ethernet in the following operating systems:

- Windows 7
- Windows 8
- Windows 10 from version 1607
  Attention: Earlier versions of Windows 10 are not supported.

Note: In the Parallel Redundancy Protocol (PRP), the original Ethernet frames are supplemented by additional information (on OSI Layer 2), which is different in *LAN A* and *LAN B* . The non-PRP nodes (devices that do not support PRP) can be connected to the PRP network via a Redundancy Box (RedBox).

Attention:

- A direct Ethernet connection - between a PRP and a non-PRP node of the communication network - will lead to errors in the Ethernet communication.
- Changing the assignment of network cards to *LAN A* and *LAN B* (or in the cabling) will lead the PRP communication to fail in all the PRP nodes.

The PRP solution of COPA-DATA integrates both the (*CDPrpFlt.sys*) kernel driver as well as a system component - the network bridge. The network bridge (Network Bridge) is an integral part of the Windows operating system and thus dependent on the operating system version and the installed Windows updates. This solution also leads to an increased dependency of the network drivers used by the operating system – i.e. the network adapters and the additionally installed protocol or filter drivers.

In the event of compatibility problems, you need to check whether additional 3rd Party network components have been installed on the system. These are to be uninstalled, if possible. In Windows, the list of installed third-party components and drivers can be shown using the properties of the network adapter.

You should perform a Windows update to ensure that the operating system is up to date.

Attention: Owing to potential dependencies, there is no guarantee that the PRP solution works for every combination of operating system and network card type and/or additional third-party components. In the worst case, there can be an incompatibility between the network card driver and the COPA-DATA PRP implementation, which can only be solved technically by switching to a different adapter type (a different network card).

10.2 Hardware requirements

The following hardware requirements are applicable for communication via PRP:

- The computer must be equipped with at least 2 network cards.

- Only 100 Mbit/s Ethernet speed is supported.
  Both the two network cards as well as the Ethernet network components (routers and switches) must be set to 100 Mbit/s Ethernet.
- Both network cards used for PRP must support *Jumboframes*.
- For both the network cards used, the *Offload* tasks must be deactivated.
- Configuration of the locally administered MAC address must be possible for both network cards for the purpose of changing the MAC address.

> **Attention**
>
> PRP communication is only supported within a redundant network. In doing so, two physical networks can be connected via PRP.
>
> A connection with additional Ethernet networks or to another PRP network (e.g. 2x2 PRP) is not supported.

The cards configured for PRP can only be used in a PRP communication network. All other network cards of the computer can still be used for normal communication via Ethernet.

10.3 Installation and configuration

To prepare the computer for PRP installation:

1. Switch the computer off and separate the computer from the power supply (physical reset).
2. Restart the computer

Carry out the following steps in the operating system:

1. Configure your two existing network adapters.
2. Create a network bridge (= **Bridge**) from the network adapters.
3. Install the *COPA-DATA PRP driver* for the network bridge.
4. Configure your PRP connection

You can find a detailed description in the further chapters.

NOTE:

Note:

- Administrator rights on the computer are required for installation.
- The system must be restarted for the installation.
- Note the instructions for the respective steps.
- The packet sync of the network service supports networks up to 100 Mbit.

- The PRP files can only be updated with a zenon main version or a service pack. Build versions are not in a position to do this.

> **Attention**
>
> Ensure that you carry out the configuration steps in the given sequence.

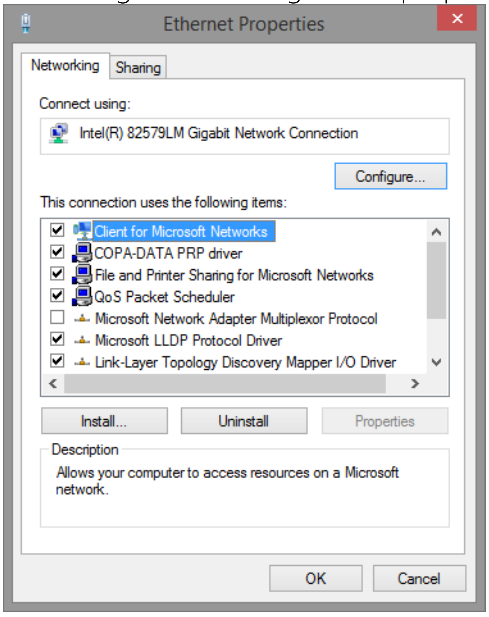
10.3.1 Installation and configuration

In the first step, amend the configuration of the operating system for both network adapters used. The configuration dialog and the naming of the enhanced properties depends on the network card used.

NETWORK ADAPTER 1

Configure the first network adapter in the operating system.

1. Open the **Change adapter settings** system setting.
   You can find these settings in the *Control Panel => Network and Internet => Network and Sharing Center*
2. Select the desired network adapter.
3. With the right mouse button, select the **Properties** entry in the context menu.
   The configuration dialog for the properties of the network adapter are opened.



Connect using:
Intel(R) 82579LM Gigabit Network Connection

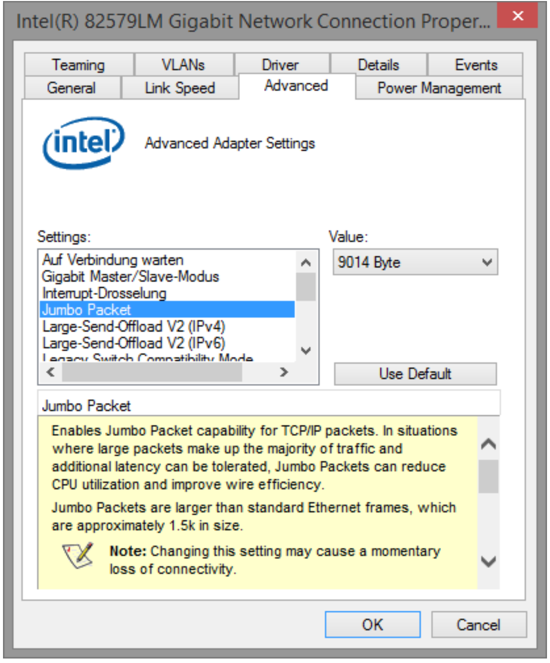
This connection uses the following items:

☑ Client for Microsoft Networks
☑ COPA-DATA PRP driver
☑ File and Printer Sharing for Microsoft Networks
☑ QoS Packet Scheduler
□ Microsoft Network Adapter Multiplexor Protocol
☑ Microsoft LLDP Protocol Driver
☑ Link-Layer Topology Discovery Mapper I/O Driver

Description
Allows your computer to access resources on a Microsoft network.

4. Click on the **Configure ...** button
   The properties window of the network adapter is opened.

5. Switch to the **Advanced** tab there.
6. In the list of **settings** there, select the **Jumbo Packet** entry
   Note: The name of this entry may be different for each network card.



Advanced Adapter Settings

Settings:

Auf Verbindung warten
Gigabit Master/Slave-Modus
Interrupt-Drosselung
Jumbo Packet
Large-Send-Offload V2 (IPv4)
Large-Send-Offload V2 (IPv6)
Legacy Switch Compatibility Mode

Value:

9014 Byte

Jumbo Packet

Enables Jumbo Packet capability for TCP/IP packets. In situations where large packets make up the majority of traffic and additional latency can be tolerated, Jumbo Packets can reduce CPU utilization and improve wire efficiency.

Jumbo Packets are larger than standard Ethernet frames, which are approximately 1.5k in size.

Note: Changing this setting may cause a momentary loss of connectivity.

7. Select a value in the **Value** drop-down list.
   Select the lowest available value that is greater than 1530 bytes.
   Attention: The **Disabled** setting must not be selected.
8. **Deactivate** all **Offload** applications of the network adapter, e.g.: **Checksum Offload**, **Large Send Offload** etc.
9. In the **Advanced** tab, select the **Locally administered address** setting.
10. Enter a unique MAC address that begins with *0A* in the *Value:* input field. You can change the address in the **Value** input field.

    The format of the MAC address depends on the hardware used.
    Examples:

    - 0A:80:41:ae:fd:7e
    - 0A-80-41-ae-fd-7e
    - 0A8041aefd7e
11. Ensure that the same MAC address is used for both connections used.
    - This MAC address must start with *0A*.
    - The MAC address in the local network must be unique.
12. Finish configuration of the network card by clicking on the **OK** button.

NETWORK ADAPTER 2

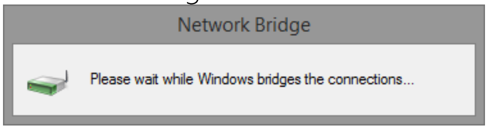
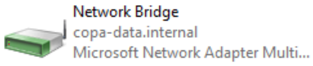
Repeat the steps for the second network adapter.
When entering the MAC address, ensure that the same MAC address as the one in the previous configuration is entered.

> ⚠ **Attention**
>
> Ensure that
>
> - the same MAC address is used on both network adapters;
> - this MAC address begins with *0A*;
> - And it is not used by any other computer in the local network.

10.3.2 Installation and configuration

In this step, you combine two network adapters with a network bridge. Amend the configuration for both network adapters used.

Create a network bridge in the system settings.

1. Open the **Change adapter settings** system setting.
   You can find these settings in the *Control Panel => Network and Internet => Network and Sharing Center*
2. Select the two network adapters that you want to use for PRP communication.
   Note: The necessary configuration has already been carried out for both network adapters. A subsequent amendment to the configuration of a network adapter only becomes effective if you then create a new bridge.
   Attention: Both network adapters selected must be configured with the same MAC address!
3. With the right mouse button, select the **Bridge connections** entry in the context menu.
   A network bridge is created for the selected network adapter. This is visualized in a dialog.
   
   Network Bridge
   Please wait while Windows bridges the connections...
4. The bridge created is displayed in the Control Panel:
   
   Network Bridge
   copa-data.internal
   Microsoft Network Adapter Multi...

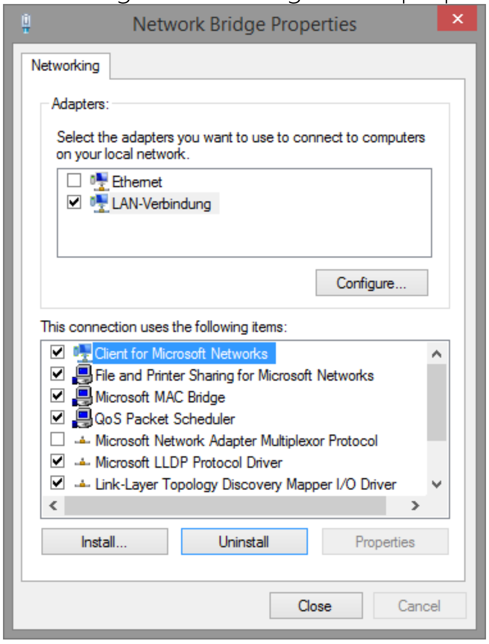
   Attention: The bridge must only contain two adapters.

10.3.3 Installation and configuration

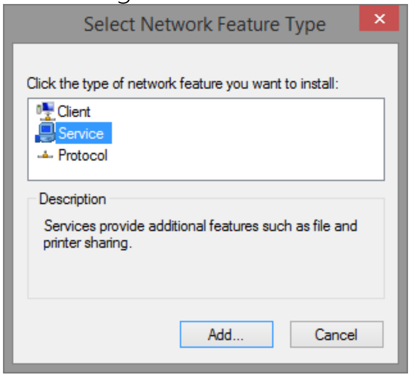
In this step, you install the service system required for PRP communication.

Install the *COPA-DATA PRP driver*

1. Select the Bridge created.
2. With the right mouse button, select the **Properties** entry in the context menu. The configuration dialog for the properties of the bridge is opened.

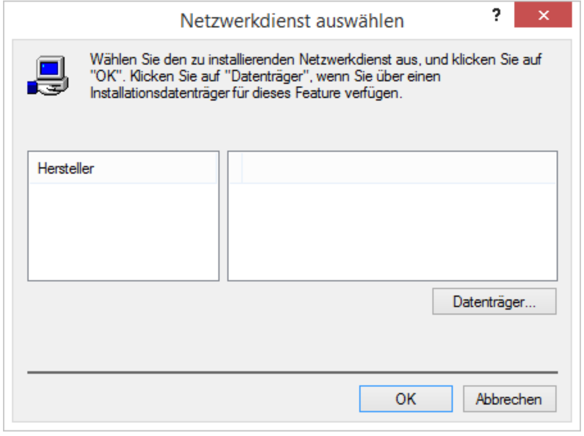

Network Bridge Properties
Networking
Adapters:
Select the adapters you want to use to connect to computers on your local network.
☐ Ethernet
☑ LAN-Verbindung
Configure...
This connection uses the following items:
☑ Client for Microsoft Networks
☑ File and Printer Sharing for Microsoft Networks
☑ Microsoft MAC Bridge
☑ QoS Packet Scheduler
☐ Microsoft Network Adapter Multiplexor Protocol
☑ Microsoft LLDP Protocol Driver
☑ Link-Layer Topology Discovery Mapper I/O Driver
Install...
Uninstall
Properties
Close
Cancel

3. Click on the **Install** button. The dialog to install a network feature is opened.


Select Network Feature Type
Click the type of network feature you want to install:
Client
Service
Protocol
Description
Services provide additional features such as file and printer sharing.
Add...
Cancel

4. Select **Service** as the network feature to be installed.

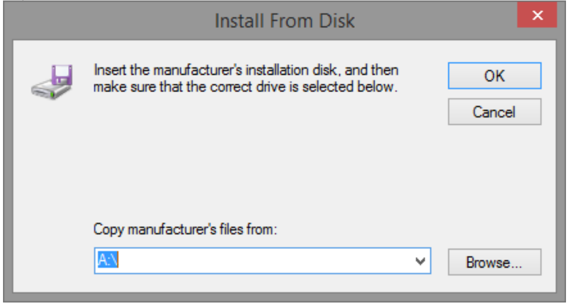
5. Click on the **Add...** button
   The dialog for the selection of the network service is opened.



Netzwerkdienst auswählen

Wählen Sie den zu installierenden Netzwerkdienst aus, und klicken Sie auf "OK". Klicken Sie auf "Datenträger", wenn Sie über einen Installationsdatenträger für dieses Feature verfügen.

6. Click on the **Data medium ...** button
   The dialog to select the save location of the installation program for the network service is opened.



Install From Disk

Insert the manufacturer's installation disk, and then make sure that the correct drive is selected below.

Copy manufacturer's files from: A:\

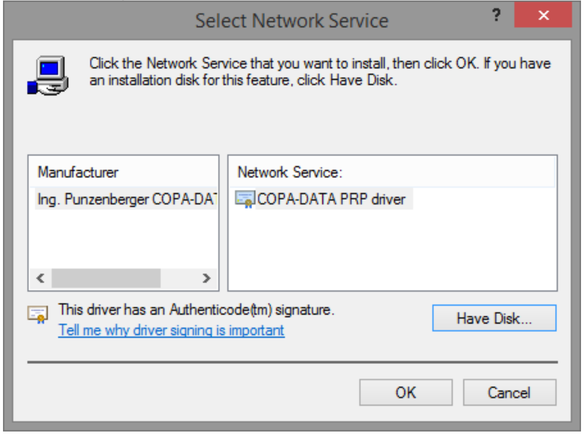
7. Click on the **Browse** button.

8. Go to the following folder on your local computer:

   - *\Programs (x86)\Common Files\COPA-DATA\CDPrpFlt\*
     for 32-bit operating systems.
   - *\Programs\Common Files\COPA-DATA\CDPrpFlt\*
     for 64-bit operating systems.

9. Select the *CDPrpFlt.inf* file.
   Attention: Ensure that you select the correct installer for your operating system (32-bit or 64-bit).

10. Confirm the selection by clicking on **OK**.
    The dialog to select the network service is opened.



Select Network Service

Click the Network Service that you want to install, then click OK. If you have an installation disk for this feature, click Have Disk.

Manufacturer: Ing. Punzenberger COPA-DAT

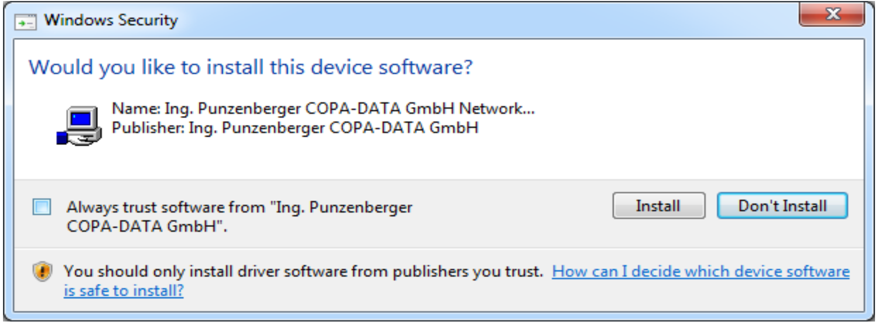
Network Service: COPA-DATA PRP driver

This driver has an Authenticode(tm) signature.
Tell me why driver signing is important

11. Select the *COPA-DATA PRP driver* network service.

12. Confirm your selection with **OK**.

- Confirm the Windows request for confirmation by clicking on the Install button.
  Attention: It may then be necessary to restart your computer.



Windows Security

Would you like to install this device software?

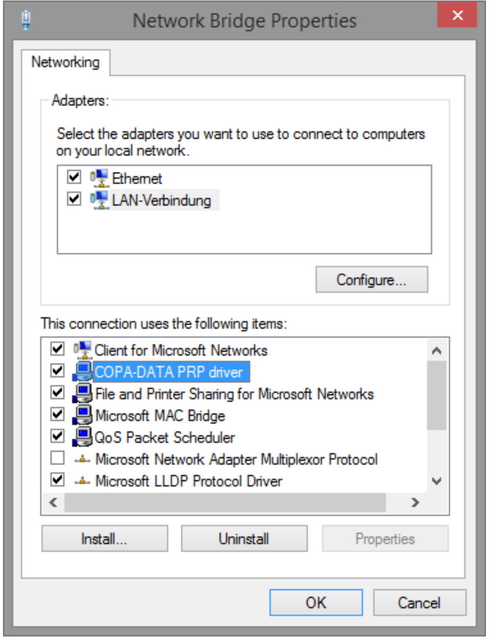
Name: Ing. Punzenberger COPA-DATA GmbH Network...
Publisher: Ing. Punzenberger COPA-DATA GmbH

□ Always trust software from "Ing. Punzenberger COPA-DATA GmbH".

You should only install driver software from publishers you trust. How can I decide which device software is safe to install?

Note: This request for confirmation is not shown if you have already activated the "... *always trust*" box when installing zenon program components earlier.

13. After successful installation (and restarting the computer) the service is visible in the properties window of the network adapter in the list of elements used.



Network Bridge Properties

Adapters:

Select the adapters you want to use to connect to computers on your local network.

☑ Ethernet
☑ LAN-Verbindung

This connection uses the following items:

☑ Client for Microsoft Networks
☑ COPA-DATA PRP driver
☑ File and Printer Sharing for Microsoft Networks
☑ Microsoft MAC Bridge
☑ QoS Packet Scheduler
□ Microsoft Network Adapter Multiplexor Protocol
☑ Microsoft LLDP Protocol Driver

14. Ensure that the LAN connection and the network service **COPA-DATA PRP driver** are activated using the checkbox.

Attention

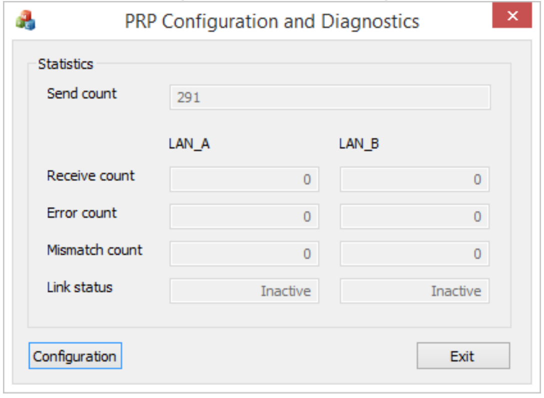
Ensure that use in the active system is not jeopardized by the required restart.

10.3.4Configuration of PRP connection (step 4 of 4)

Before configuration, ensure that the LAN connection and the **COPA-DATA PRP driver** network service are activated.

PRP CONFIGURATION

1. Start the program called *PRPCfgDiag.exe*.
   You can find this software on your computer in the following folder: *C:\Program Files (x86)\Common Files\COPA-DATA\STARTUP*.
   The **PRP Configuration and Diagnostics** dialog is opened.



PRP Configuration and Diagnostics

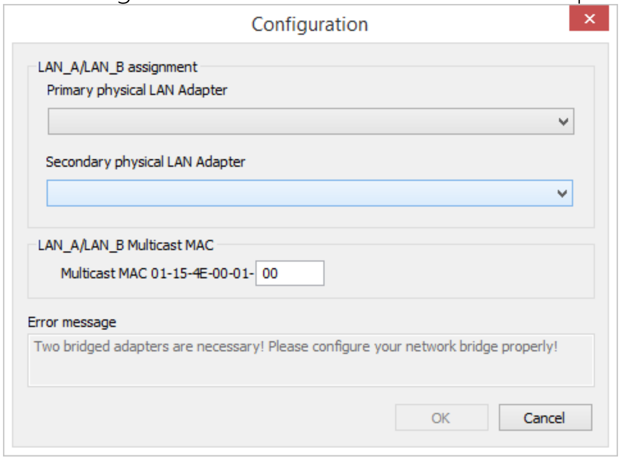
Statistics

Send count: 291

	LAN_A	LAN_B
Receive count	0	0
Error count	0	0
Mismatch count	0	0
Link status	Inactive	Inactive

   Note: The **PRP Configuration and Diagnostics Tool** is only available in English. You can find a detailed description of the **PRP Configuration and Diagnostics Tools** in the PRP configuration and diagnosis tool (on page 119).

2. Click on the **Configuration** button.
   The dialog for the selection of the network adapter is opened.



Configuration

LAN_A/LAN_B assignment

Primary physical LAN Adapter

Secondary physical LAN Adapter

LAN_A/LAN_B Multicast MAC

Multicast MAC 01-15-4E-00-01- 00

Error message

Two bridged adapters are necessary! Please configure your network bridge properly!

   Note: The content of the drop-down list is based on the system settings.

3. Select the network adapter for **LAN_A** and **LAN_B** from the drop-down list.
   Note: Ensure that for all PRP-compatible devices in the Ethernet network, the references between the physical network and **LAN_A** or **LAN_B** are configured the same. The plugs of the cables for **LAN A** and **LAN B** must not be mixed up.

4. Confirm the assignment with **OK**.
5. End the configuration by clicking on the **Exit** button.

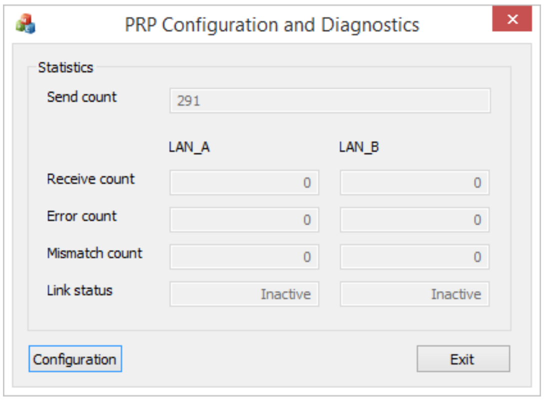
After these steps, the entire communication of this network adapter is carried out using the PRP protocol. This means that all applications on the computer that access the Ethernet via this interface (TCP, UDP etc.) communicate with the support of PRP. No additional engineering is required in the applications; the use of PRP is "transparent" for them.

> **Attention**
>
> If the PRP communication does not function, make sure that all the System requirements (on page 109) and the Hardware requirement (on page 109) are met.
>
> You can also delete the network bridge, restart the computer (don't forget Windows Updates) and carefully repeat all the configuration steps down to the last detail.

10.4 PRP configuration and diagnosis tool

The **PRP Configuration and Diagnostics Tool** performs two tasks:

- Visualization (on page 120)
  Display of the data traffic sent via PRP. The information is displayed separately for the two network adapters used.
- Configuration (on page 121)
  Assignment of the configured network adapter.



Send count: 291

	LAN_A	LAN_B
Receive count	0	0
Error count	0	0
Mismatch count	0	0
Link status	Inactive	Inactive

Note: This dialog is only available in English.

The **PRPCfgDiag.exe** program is supplied with zenon.
You can find this software on your computer in the C:\*Program Files (x86)**Common Files**COPA-DATA**STARTUP* folder.

REQUIREMENTS

The **PRP Configuration and Diagnostics Tool** needs the following for operation or configuration:

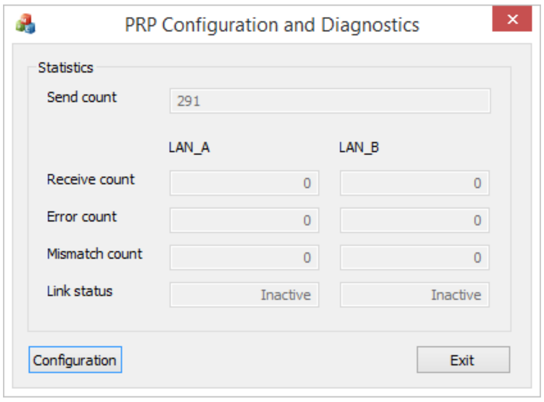
- Two network adapters that are combined into a bridge in the system settings.
  Note: In this bridge, only the two network adapters that are used for PRP communication can be configured. Other network adapters must not be included in this bridge.
- The CDPrpFlt driver must be installed in the operating system.

> **Information**
>
> You can find information on the installation and necessary preparations in the system settings in the installation and configuration (on page 110) chapter.

10.4.1 Statistics

The data flow is visualized in the **Statistics** dialog. The setting is displayed separately for both LAN adapters.

The flow of data is always recorded, even if the tool is not open.



	LAN_A	LAN_B
Send count	291	
Receive count	0	0
Error count	0	0
Mismatch count	0	0
Link status	Inactive	Inactive

Note: This dialog is only available in English.

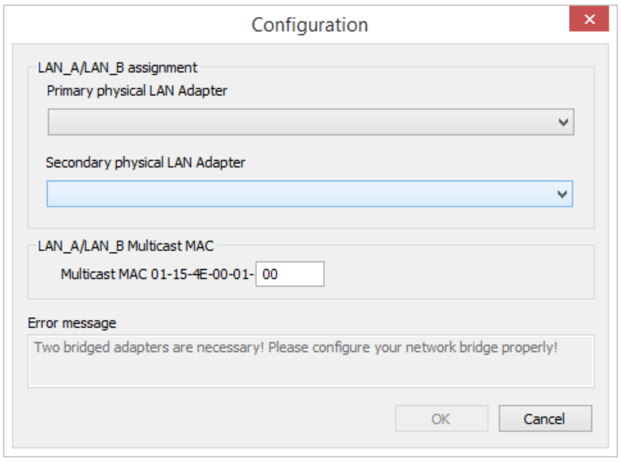
Parameter	Description
Send count	Display of the Ethernet frame sent.
Receive count	Display of the Ethernet frame received.
Error count	Display of corrupt PRP frames. Possible cause could be, for example, a mix-up of *LAN A* and *LAN B* (at any PRP nodes).
Mismatch count	Display of PRP frames received/sent differently if the network data traffic of the two LAN adapters

Parameter	Description
	differs from one another.
Link status	Status of the network card: - *Active* PRP-Supervision frames are received correctly for the respective LAN (*LAN_A* or *LAN_B*). - *Inactive* No PRP Supervision frames are received within the past two seconds. There is no PRP station in the network or there is an error.
Configuration	Opens the configuration dialog (on page 121).
Exit	Closes the program. **Note:** The data continues to be recorded.

10.4.2 Configuration

The following is carried out in the **Configuration** dialog:

- Network adapter is assigned by means of a drop-down list.
  The content of the drop-down list is based on the network settings.
  You can find further information in the installation and configuration (on page 110) chapter.
- The multicast MAC address is visualized
- Error messages from the network adapter configuration are visualized in an output window


Configuration
×
LAN_A/LAN_B assignment
Primary physical LAN Adapter
Secondary physical LAN Adapter
LAN_A/LAN_B Multicast MAC
Error message
OK
Cancel

Multicast MAC 01-15-4E-00-01- 00

Error message: Two bridged adapters are necessary! Please configure your network bridge properly!

⚠ **Attention**

The computer must be restarted after changes to the configuration have been made.

Note: This dialog is only available in English.

Parameter	Description
Primary physical LAN Adapter	Assignment of a network adapter to the physical connection to *LAN A* - for the primary LAN adapter. **Attention:** the assignment to *LAN A* and *LAN B* cannot be mixed in a PRP network. In the drop-down list, the adapters that are included on the configured bridge are listed. You can find information on this in the installation and configuration (on page 110) chapter.
Secondary physical LAN Adapter	Assignment of a network adapter to the physical connection to *LAN B* - for the secondary/redundant LAN adapter. In the drop-down list, the adapters that are included on the configured bridge are listed. You can find information on this in the installation and configuration (on page 110) chapter.
LAN_A/LAN_B Multicast MAC	Multicast MAC address for PRP Supervision frames. This address for the communication in the network is preset and cannot be changed. **Note:** Ensure that all PRP nodes use the same multicast MAC address in your network. No network adapter is to use this MAC address for another purpose. The last byte can be configured in the input field. The input format for this entry is hexadecimal (hex).
Error message	Output window with error messages.
OK	Accepts all changes and switches to statistics dialog (on page 120).

Parameter	Description
Cancel	Discards all changes and switches to statistics dialog (on page 120).

Attention: In the PRP communication network, the original Ethernet frames are supplemented by additional information (on OSI Layer 2), which is different in *LAN A* and *LAN B*. Mixing up the assignment of network cards (or cables) to *LAN A* and *LAN B* leads to errors in the Ethernet communication throughout the entire PRP network.

11 Settings in zenon Editor for Service Grid

The use of zenon in the Service Grid requires certain requirements and settings in the zenon Editor for the synchronization with the zenon Analyzer.

11.1 Connection to service hub for the zenon project

You can configure the connection to the **Service Hub** for the zenon project in the project properties in the zenon Editor. This connection is used to configure the synchronization of data between the zenon Editor and zenon Analyzer.

Attention

This configuration has no influence on the data transfer from the zenon Runtime to the Service Grid.

Runtime data is transferred using the Service Grid Runtime Add-in. You can configure this connection with the **Service Node Configuration Tool**.

To configure the connection from the zenon Editor to the **Service Hub**:

1. Highlight the project.
2. Go to the **Network** node in the properties.
3. Go to the **Service Grid** area.
4. In the **Service Hub** property, select the desired type of connection to the **Service Hub** from the drop-down list:
   - *<No service hub selected>*: No connection to a service hub is established. Existing connections are separated in the zenon Editor.
   - *<Apply from the global project>*: The service hub configured in the global project is used.
   - *<configured service hubs>*: List of all available connections.

The selected connection is entered in the **project.ini** when the Runtime files are created.

12 Metadata Synchronizer

The **Metadata Synchronizer** sends metadata from zenon to a zenon Analyzer metadata database.

Requirements: zenon Analyzer 3.30 or higher and zenon 8.10 or higher.

In contrast to the **Analyzer Export Wizard**, the **Metadata Synchronizer** is implemented in zenon and zenon Analyzer directly. This results in many benefits, most of all:

- The transfer runs much more quickly.
- Increased stability and tolerance of errors.
- Version independence starting from zenon 8.10 and zenon Analyzer 3.30.

DATA TRANSFER

The **Metadata Synchronizer** transfers from zenon to zenon Analyzer:

- Alarm/event classes and alarm groups
- Users
- Equipment models
- Network:
  If the **Network** property is active, configurations for **Server 1** and **Server 2**.
- Projects
- Project contents:
  - Variables
  - Archives
  - Shifts
  - Status texts
  - Efficiency class models
- Sankey models
- Waterfall models

The following is applicable for the target during transfer:

- Objects that no longer exist are deleted.
  Exception: Projects
  During deletion, insofar as possible, all dependent objects are also deleted.

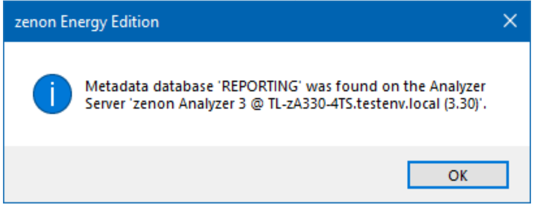
- Existing objects are updated.
- New objects are added.

Note: Objects that have been created by **ZAMS** or the **Metadata Editor** are not changed.

12.1 Configuration

To transfer data for zenon Analyzer in the Service Grid:

1. Ensure that a valid connection has been selected in the zenon Editor for the **Service Hub** project property in the **Network**/**Service Grid** node.
2. Navigate to the **Metadata Synchronizer** node in the project properties.
3. Select a **Analyzer instance**.
   click on the **...** button to open the dialog to select an Analyzer server (on page 125).
4. To do this, select a **Metadata database**.
   Click on the **...** button to open the dialog to select and configure a database (on page 126).
5. Optional: Test the configuration by clicking on the **...** button in the **Test connection** property.



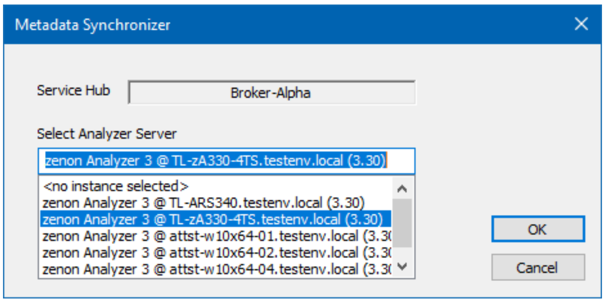
zenon Energy Edition

Metadata database 'REPORTING' was found on the Analyzer Server 'zenon Analyzer 3 @ TL-zA330-4TS.testenv.local (3.30)'.

The **Metadata Synchronizer** can now be executed in the zenon Editor.

12.1.1 Analyzer Server selection dialog

You select the Analyzer server in the service hub with this dialog.



Metadata Synchronizer

Service Hub: Broker-Alpha

Select Analyzer Server: zenon Analyzer 3 @ TL-zA330-4TS.testenv.local (3.30)

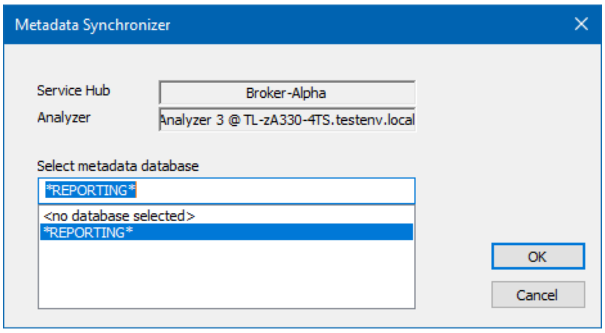
<no instance selected>
zenon Analyzer 3 @ TL-ARS340.testenv.local (3.30)
zenon Analyzer 3 @ TL-zA330-4TS.testenv.local (3.30)
zenon Analyzer 3 @ attst-w10x64-01.testenv.local (3.3
zenon Analyzer 3 @ attst-w10x64-02.testenv.local (3.3
zenon Analyzer 3 @ attst-w10x64-04.testenv.local (3.3

Option	Description
Service Hub	Display of the service hub configured in the

Option	Description
	Service Hub property.
Select Analyzer server	Direct entry of the Analyzer servers or selection from a drop-down list: - Selection of an Analyzer server: Applies selected instance. - *<No Analyzer server selected>*: Removes the configured Analyzer server. - *<Apply from the global project>*: Applies the configuration selected in the global project. **Hinweis:** In order for an Analyzer server to be able to be selected, a valid connection to the **Service Hub** must be configured. This is established with the **Service Node Configuration Tool**.
OK	Applies settings and closes the dialog.
Cancel	Discards all changes and closes the dialog.

12.1.2 Database selection dialog

You select the metadata database with this dialog.



Metadata Synchronizer

Service Hub: Broker-Alpha

Analyzer: Analyzer 3 @ TL-zA330-4TS.testenv.local

Select metadata database: *REPORTING*

<no database selected>
REPORTING

Option	Description
Service Hub	Display of the service hub configured in the **Service Hub** property.
Analyzer Server	Display of the analyzer server configured in the **Analyzer instance** property.
Select metadata database	Direct entry of the metadata database or selection

Option	Description
	from drop-down list: ▸ Selection of a metadata database: Applies selected metadata database. ▸ *<No database selected>*: Removes the configured database. ▸ *<Apply from the global project>*: Applies the configuration selected in the global project. **Note:** In order for a metadata database to be selected, a valid connection to the **Service Hub** and to the **Analyzer Server** must be configured.
OK	Applies settings and closes the dialog.
Cancel	Discards all changes and closes the dialog.

12.2 Execution

The **Metadata Synchronizer** can be executed and stopped.

To transfer metadata to a database:

1. Go to the **Extras** menu in the zenon Editor.
2. Select the **Execute** **Metadata Synchronizer** entry.

   The **Metadata Synchronizer** is started. Metadata is collated and transferred to the configured database.

   The actions and the result are displayed in the output window.

To stop the **Metadata Synchronizer**:

1. Go to the **Extras** menu in the zenon Editor.
2. Select the **Stop Metadata Synchronizer** entry.

   The **Metadata Synchronizer** is stopped.

SYNCHRONIZATION RULES

VISUAL NAME

Behavior when synchronizing the Metadata Editor and the **Metadata Synchronizer**:

1. A variable does not exist in the Metadata database:
   - **Visual name** in the zenon Editor is empty: The variable name is entered as the **Visual name** in the Metadata database.
   - **Visual name** in the zenon Editor is configured: The **Visual name** in the zenon Editor is entered as the **Visual name** in the Metadata database.
2. A variable already exists in the Metadata database and the visual name corresponds with the variable name:
   - **Visual name** in the zenon Editor is empty: The variable name is entered as the **Visual name** in the Metadata database.
   - **Visual name** in the zenon Editor is configured: The **Visual name** in the zenon Editor is entered as the **Visual name** in the Metadata database.
3. A variable already exists in the Metadata database and the visual name does not correspond with the variable name:
   - **Visual name** in the zenon Editor is empty: The visual name in the Metadata database remains unchanged.
   - **Visual name** in the zenon Editor is configured: The **Visual name** in the zenon Editor is entered as the Visual name in the Metadata database. Visual names changed in the Metadata Editor are overwritten.

The name in the zenon Editor is always used as the visual name for projects. When updating renamed projects (if the **Visual name** property remains empty), the zenon Analyzer overwrites none of the changes made with the **Metadata Editor**.

DESCRIPTIONS

If descriptions for objects applied from the **Metadata Synchronizer** from zenon are empty, the descriptions present in the database remain unchanged.
This applies for:

- Equipment groups
- Alarm/Event classes
- Alarm/Event groups
- User
- Projects
- Archives
- Variables (**Identification** is used)

NORMALISATION

Data for efficiency classes must be normalised for use in the zenon Analyzer. Data from the zenon Editor are never normalised. Normalisation can only be configured in the Metadata Editor.
During synchronization, the **Metadata Synchronizer** checks whether the efficiency class model already exists in the Metadata database:

- efficiency class model is not present: no normalisation is present. This must be configured in the Metadata Editor.
- Efficiency class model is present: The normalisation present in the Metadata database remains unchanged.

SANKEY DIAGRAMS AND WATERFALL MODELS

Sankey diagrams and waterfall models are validated after checking the variables and before sending the data.
In doing so, the following applies:

- Connections in Sankey diagrams may only use variables that are contained in project, archive, variable and compression during synchronization.
- Waterfall models may only use variables that are contained in project and variable, but not archives during synchronization. For waterfall charts, it is sufficient if the variable is contained in any archive.

VARIABLES

All variables are checked before synchronization to see if they have to be synchronized. A variable is only synchronized if it meets at least one of these conditions:

- The variable has an assigned reaction matrix:
  In addition to the default entry, this reaction matrix contains at least one other entry that generates the alarms (AML) or events (CEL).
- The variable does not have an assigned reaction matrix:
  It has at least one activated limit value that generates an alarm (AML) or an event (CEL).
- The variable is contained in at least one archive.

VALIDATION

Entries from zenon are largely validated before transfer. Errors are corrected. If correction is not possible, the respective object is excluded from synchronization. All validation errors are logged in the diagnosis viewer as a warning and displayed in the output window in the zenon Editor.

This applies for:

- Archives

- Reaction matrices
- Alarm classes
- Alarm groups
- Equipment models
- User
- Projects
- Sankey diagrams
- Shift models
- Variables
- References to equipment models
- Waterfall charts

13 Routing

It is strongly recommended that the **Routing active** project is no longer used for a current project configuration. This property is deactivated by default if the zenon network is reconfigured.

PROCEDURE

For routing, the packets of subordinate projects are sent to the first client project (**FCP**) in the branch. The computer acts as node computer and can route packets. Thereby all network packets from the outside use this computer. However, this setting can lead to bottlenecks and influences the possible network topology. It is only worthwhile using it in special network setups, e.g. for slow WAN networks or routed networks.

- Example:
  If, in a setup consisting of several computers, not all computers can reach the others, a computer can act as a router.
- Technical implementation:
  The **Server 1** and the **Server 2** of the subordinate projects are amended on that of the ECP; this is the **Server 1**/**Server 2** active in Runtime.

13.1 General notes on routing

Two basic rules must be noted when configuring network structures with routing. If one of these rules is not adhered to, communication problems or other undesired effects may occur depending on the respective structure.

- Rule 1: Server and levels
  A PC that acts as a server may only act as a server or Standby several times in one level (circular redundancy). It must not be defined as a server a level above or below.
- Rule 2: Standalone

  If the start project is a single-user project, only one single level below can be used for network projects.

CLIENT SENDS TO A SERVER

- The client sends the packet to the server active in the project in Runtime.
- If the project on this computer is not the server, the packet is sent until it arrives at the server.
- This functionality is not affected by an integration project.

SERVER SENDS TO A CLIENT WITH ROUTING

1. If the server has a direct connection to the client, the packet is sent there.
2. If there is no connection to the target computer, the server sends the packet to all computers on which the project is running for which it acts as a server.
3. If the node has a direct connection to the client, the packet is sent there.
4. If the computer works as a node, then the packet is sent to all computers which have connected to the node computer. It the target computer is also the source computer, the packet is not sent any further.
5. The procedure is continued at point 3.

Note: Points 2 and 4 are only carried out if routing is active on these computers.

Information

The Server and Standby need not correspond to what has been configured on the client computers, otherwise they may change themselves depending on the topology of the respective computer.

WHAT IS A CLIENT CONNECTION?

A network service connection is labeled as a client connection if it is made to the server or standby handling the process by a client. This is recognizable in that there is a connection to port 1100 on the target computer.

> ⚠ **Attention**
>
> It is not guaranteed that a pure client computer added to a functional, defined topology will work. It is possible that some projects cannot be reached by the server due to routing on client computers in particular.

CHECKING THE ROUTING

To check the routing settings, use the procedure from "Administering network topology (on page 67)".

13.2 Compatibility

RULES FOR ROUTING BEFORE ZENON VERSION 6.50:

1. The first client network project of a branch on a computer defines the server and standby for all subordinate projects in the branch. This also applies
   - If a subordinate project on this computer is a server or a standby server
   - for projects that do not really have a Standby Server
2. If the subordinate project is not a network project or is not a server, the branches of the subprojects of the start project are considered in parallel. Different computers can therefore be servers for the subprojects. The rules from item 1 apply for the branches.
3. Single user projects are not taken into account for the topology, with the exception of the start project.
4. If the start project is not a server (i.e. single user, client or standby not handling the process), routing is not activated in the network service. This only affects the direction from the server to the client.

RULES FOR ROUTING FROM ZENON VERSION 6.50:

The **Routing active** property is deactivated as standard from version 6.50 onwards.

WITHOUT ROUTING

If the **Routing active** property is not active for the start project on the computer, routing does not take place. Each project then connects directly to the corresponding computer, where it is the server. The computer is then not a node and packets are also not routed from here.

WITH ROUTING

The rules as they were prior to 6.50 remain valid.

Exception:

- A project that is a server or standby on the computer remains a server or standby, even if the superordinate project uses another server or standby.

14 Operating authorization in the network

A network project can be operated from all stations in the same way with the basic settings.

Operating here means actively intervening in the process, such as:

- Send values
- Write recipes
- Acknowledge alarms
- etc.

There is thus the danger that two users on two different stations want to set different values for the same variable at the same time.

In this case:

- Both actions are executed
- The value that is entered last overwrites all previous ones

BLOCK OF ACTIONS ON CLIENTS BY MEANS OF OPERATING AUTHORIZATION

In zenon you have the possibility to allow operation of the project only from one station at the time. In this case the operator has to get the authorization, before he can operate the project. Opening screens, as well as read access to lists such as AML, CEL, recipes, etc. is possible on each station as always.

Information

Operating authorizations for projects without a network can be implemented by evaluating a binary variable for the project property **Operation lock**. For details, see the Operating authorizations chapter in the Project administration and workspace manual.

SUPPORTED ZENON ELEMENTS

The following zenon elements support **Operating authorization in the network**, both for *global operating authorizations* as well as for *operating authorizations via equipment model*:

- Clock
- Universal slider
- Dynamic text
- Bar display
- Pointer instrument
- Numeric value
- Switch

In addition, the *global operating authorization* needs a corresponding operating authorization for each write access to Runtime.

PROCEDURE

The following is applicable if the **Operating authorization in the network** property is engineered:

- Authorization must be obtained if active operation takes place.
- If operation is locked by another computer, a dialog is opened on the computer that is locking it.
- The user who is locking it can release the authorization or keep it locked.
- If there is no response, the authorization is released after a pre-set timeout.
- If an interruption in the network connection is recognized, then the authorization for this computer is reset.

For details see chapter:

- Configuring operating authorization (on page 136)
- Operating authorization in the Runtime (on page 138)

Information

The processes of the operating authorizations create corresponding entries in the CEL.

SYSTEM VARIABLES FOR OPERATING AUTHORIZATION

The system variables inform you about operating authorization.

- **[Network] operating authorization: Computer that owns it**:
  Name of the computer that has the authorization (Data type: *String*)
- **[Network] Operating authorization available**:
  Computer has authorization (Data type: *Bool*)
- **[Network] Authorization denied**:
  Computer requests authorization, but does not receive it (Data type: *Bool*)

Information

The system variables are only applicable for the *global operating authorization*.
The *Operating authorization via equipment modeling* is used for the visualization in Runtime of variables linked to the equipment group directly.
You can find further information in the Equipment modeling manual in the Engineering in the Editor chapter.

14.1 Types of operating authorization

With operating authorizations in zenon, a distinction is made between:

- Global operating authorization
  - Can be activated or deactivated for the complete project.
  - Operating authorization is locked or unlocked for all elements.
- Operating authorization via equipment modeling
  - Can be activated or deactivated for parts of the project, based on the equipment model.
  - Operating authorization is locked or unlocked for equipment groups.

Attention

The *Operating authorization via equipment modeling* is not available for the **Batch Control** module.

14.2 Engineering in the Editor

To enable the operating authorization in the network,:

- Activate **Operating authorization in the network**.
- Set the properties for the operating authorization.
- Engineer one or more functions for getting and releasing operating authorizations in the Runtime.

OPERATING AUTHORIZATION VIA EQUIPMENT MODEL

If the operating authorization is controlled by means of the equipment model, variables must be linked to the corresponding equipment group.
You can find further information in the Equipment modeling manual in the Operating authorization via equipment model chapter.

14.2.1 Activating authorizations

Carry out the following steps to activate the operating authorization:

1. navigate to the **Network** group in Project Properties.
2. In the **Operating authorization in the network** drop-down list, select the type of operating authorization.
3. Define the **Timeout for request [s]**:
4. Defines the period of time in which a computer can respond to an approval request. The authorization is automatically approved after this time has expired.
5. Default: *60* seconds
6. Define the **Timeout for operating authorization [s]**:

   Defines the period of time in which a computer that has authorization must report to the Primary Server. The authorization is automatically approved after this time has expired. Connection interruptions in the network are therefore recognized. The authorization can therefore not be blocked by a computer that cannot be contacted.

   Default: *60* seconds.

   Attention: Select the time period as shorter than the network timeout.

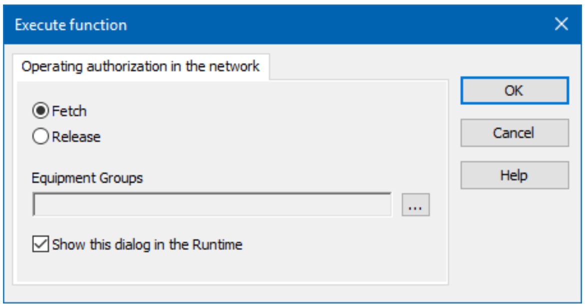
14.2.2 Engineering in the Editor

To obtain operating authorizations or to approve these, the corresponding functions must be available in Runtime. To do this, create two buttons that are designated for the corresponding functions:

- *Get authorization*:
  Obtains authorization from the user's own computer
- *Approve authorization*:
  Approve authorization or explicit request

GET AUTHORIZATION

1. Create a new function.
2. Select the **Network** function in the **Operating authorization in the network** group.
3. The selection dialog for authorizations in the network is opened.



Execute function

Operating authorization in the network

● Fetch

○ Release

Equipment Groups

☑ Show this dialog in the Runtime

4. Select **Get**.

If this function is executed in Runtime, the authorization can be obtained from the user's own station.

APPROVE AUTHORIZATION

1. Create a new function.
2. Select the **Network** function in the **Operating authorization in the network** group.
3. The selection dialog for authorizations in the network is opened.
4. Select **Approve**.

If this function is executed in Runtime, the authorization can be approved again.

Information

You can find further information in the Creating operating authorization functions in the network (on page 144) chapter.

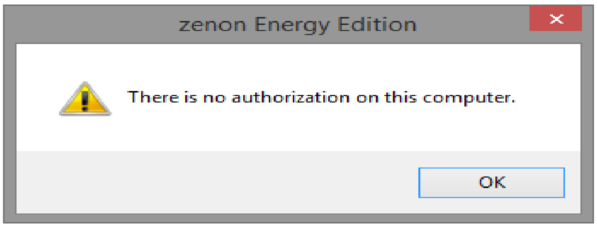
14.3 Operating authorization in the Runtime

If the **Operating authorization in the network** project property is engineered, active operations are only executed in the Runtime if there is operating authorization for the station.

If the operating authorization is not present, it can be requested by means of a function.

EXAMPLE

If there is no operating authorization, a set value should be written to a variable:



zenon Energy Edition

There is no authorization on this computer.

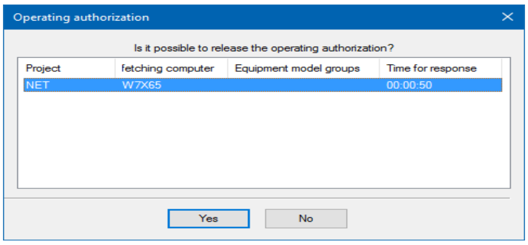
1. The set value is not sent to the hardware when the button is clicked on.
2. Instead, a dialog opens informing you that you do not have the authorization for this project.
3. Click on the **Obtain the operating authorization** button that has been configured.

DOES NOT BLOCK AUTHORIZATION FOR ANY OTHER COMPUTER:

If the operation is freely available:

- You receive the authorization
- You can write the set value
- You can, after the operation with the configured **Operating authorization in the network** function, with the *Unlock authorization* option selected, make this available to other users.

BLOCKS AUTHORIZATION FOR ANOTHER COMPUTER:



Operating authorization

Is it possible to release the operating authorization?

Project	fetching computer	Equipment model groups	Time for response
NET	W7X65		00:00:50

If authorization is blocked:

- A dialog to unlock the operating authorization is opened on the computer that is blocking.
- Possible unlocking options by the user of the blocking computer:

- **Yes**:
  The authorization is passed over to the other computer.
- **No**:
  Authorization remains blocked
- **No reaction**:
  Countdown corresponding to the timeout time defined in the **Timeout for operating authorization [s]** property runs out. The authorization is automatically released at 00:00.

14.4 Project configuration amendments - reloading of Runtime

If changes to parts related to operating authorizations (project properties, equipment model, variables) are made, the current operating authorization must be applied by reloading or restarting the Runtime.

In doing so, the following applies:

- The old project configuration is applicable until the client has reloaded.
- If tokens that have already been distributed in Runtime are no longer needed after reloading, these are unlocked again by the reloading. As a result, no messages about assigned tokens are sent to the server. The server stops monitoring the tokens on the client.
  A corresponding LOG entry is created for this.
- When reloading, the configured timeout times **Timeout for operating authorization [s]** and **Timeout for request [s]** are also reloaded automatically. Open requirements of an operating authorization are rejected by the reloading and no longer modified.
  Note: If the timeout is reduced for the operating authorization, clients that have not yet reloaded lose their tokens relatively quickly because they report to the server too late.

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.

14.5 LOG entries for Operating authorization

Text	Level	Description
Token no longer relevant for group <group>	**Project DEBUG**	The *<group>* is no longer current after reloading and has no meaning for the operating authorization.

Text	Level	Description
Token no longer exists for group guid <guid>	**Project DEBUG**	The group with the guid <*guid*> has been removed on reloading.
Token reserved for group <group> for host <host> with id <id>	**Project DEBUG**	Message on the Server or Standby Server by means of the issue of a token: The token for the <*group*> group for <*host*> computer under ID <*id*> has been reserved.
Token denied for host <host>	**Project DEBUG**	Message on the server that the token query for the <*host*> host has been rejected. **Background:** One of the queried groups is already reserved for a different computer.
LOG SendData Project:<project> To:<hostname> Modul:8 Prior:1 Class:CEqTokenReservationMsg	**NET DEBUG**	Reservation information for a reservation in the <*project*>. Sent by the Server to the Standby Server <*hostname*>
LOG ReceiveData Project:<project> From:<server> To:<hostname> Modul:8 Class:CEqTokenReservationMsg	**NET DEBUG**	The reservation information received from the Server on the Standby.
Token reservation lifted for <group> for id <id>	**Project DEBUG**	Reservation for the <*group*> with the <*id*> has been withdrawn. **Background:** Reservation no longer necessary, because the token can be issued directly.
LOG SendData Project:<project> To:<host> Modul:8 Prior:1 Class:CEqTokenRemoveReservationMsg	**NET DEBUG**	Network message from the server to the standby server for the synchronization of the reservation.
LOG ReceiveData Project:<project> From:<server>	**NET DEBUG**	Network message on the Standby Server for the synchronization of the reservation.

Text	Level	Description
To:<host> Modul:8 Prior:1 Class:CEqTokenRemoveReservationMsg		
Token reservation lifted due to timeout	**Project Debug**	The reservation has been withdrawn, because it has existed for too long.
Standby received 1 token reservations	**Project DEBUG**	Existing reservations have been transferred from the server to the starting Standby. Background: The reservation must be known for a possible server switch.
Token reservation <id=1> for host WKS086-W7X64 with 1 groups: <relevant.screen.batch >	**Project DEEPDEBUG**	Each reservation is logged in its own line: - Reservation ID - Name of the reserving host - Number of the group - Group Name
Token acquired for Group: <voller Gruppenname>	**Project DEBUG**	Message when getting an operating authorization for the <*full group name*> group. (Server/Client)
Token released for Group: <voller Gruppenname>	**Project DEBUG**	Message when releasing an operating authorization for the <*full group name*> group. (Server/Client)
SendData Project:<Projekt> To:C Modul:8 Prior:1 Class:CEqReleaseTokenMsg	**Net DEBUG**	States that the server has sent a ReleaseMessage to all clients
ReceiveData Project:<Projekt> From:<Client> To:S Modul:8 Class: CEqReleaseTokenMsg	**Net DEBUG**	A release token message <*Client*> has been received by client on the server.
ReceiveData Project:<Projekt> From:<Client> To:C	**Net DEBUG**	On the client, a release token message <*Client*> was received from the server.

Text	Level	Description
Modul:8 Class: CEqReleaseTokenMsg		
ReceiveData Project:<Projekt> From:<Server> To:<Client> Modul:8 Class:CEQTokenInfoAnswer	**Net DEBUG**	An info answer was received on the client so that the client knows the current token issue on the server
SendData Project:<Projekt> To:<Client> Modul:8 Prior:1 Class:CEQTokenInfoAnswer	**Net DEBUG**	Server sent token info <*Client*> to the client
ReceiveData Project:MANYMODULS From:<Client> To:S Modul:8 Class:CD_CNetTokenQuit	**Net DEBUG**	On the server, the confirmation <*Client*> that it does not exist yet has been received from the client.
SendData Project:MANYMODULS To:S Modul:8 Prior:1 Class:CD_CNetTokenQuit	**Net DEBUG**	On the client, the confirmation that it (the client) still exists has been sent to the server.

15 zenon functions in the network

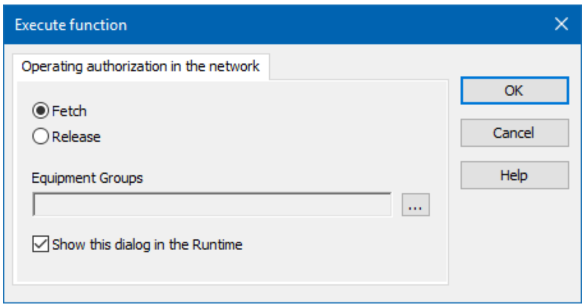
Special zenon functions for the network:

- **Authorization in network** (on page 143)
- Redundancy switch
  a) in redundancy mode: **Dominant** (on page 145)
  b) in redundancy mode: **Non-dominant**
  c) in redundancy mode: **Rated**

In general, the location of execution (on page 155) must be noted when using functions in the network. For some functions, the location of execution can be freely configured; this is fixed for others.

15.1 Authorization in network

The **Operating authorization in the network** function can be used to request, transfer or sort operating authorizations according to equipment groups.



Execute function

Operating authorization in the network

● Fetch

○ Release

Equipment Groups

☑ Show this dialog in the Runtime

Parameter	Description
Fetch	Fetches operating authorization for the calling computer.
Release	Transfers the operating authorization to the requesting computer. **Note:** Each client can only approve their tokens. There is no general releasing.
Equipment groups	The operating authorization is requested or unlocked for variables of the selected equipment group. Click on button **...** in order to open the dialog for selecting an equipment group. Multiple selection and hierarchical filtering is possible. If no equipment group has been selected, the request or release is for the complete project. **Note:** In the selection dialog, only the equipment groups for which the **Equipment model relevant for operating authorization** property has been activated are offered. This property must be activated in the higher-level equipment model. If an equipment group that is no longer available for the operating authorization has been configured in the function dialog, no operating

Parameter	Description
	authorizations are issued. A LOG message is created in this case.
Show this dialog in the Runtime	Selection of whether the dialog is shown in Runtime. ▸ *Active*: Dialog is displayed in Runtime Settings can be edited before execution. ▸ *Inactive*: Dialog is not offered in Runtime. The settings configured in the Editor are applied.

CLOSE DIALOG

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

15.1.1 Create operating authorization in the network function

An *operating authorization in the network* function is used for requesting or permitting an operating authorization for an action in Runtime.

ENGINEERING

Steps to create the function:

1. Create a new function:

   In the toolbar or in the context menu of the Functions node, select **New function**.
   The dialog to select a function is opened.

2. Navigate to the node **Network**.
3. Select the **Operating authorization in the network** function.

   The dialog for engineering a function is opened.

4. Select the desired option:

   a) *Get*
   Request operating authorization for blocked element.

   b) *Release*
   Transfers operating authorization

5. Optional - with operating authorization via equipment model:
   Link the corresponding equipment groups by clicking on the ... button.
   Note: Only the equipment models for which the **Equipment model relevant for operating authorization** property has been activated are offered for selection.
   You can find further information in the Equipment Modeling manual in the Engineering in the Editor chapter.
6. Name the function in the **Name** property.
7. Link the function to a button.

15.2 Redundancy switching in the dominant network

This function makes it possible to switch manually between the Primary Server and the Standby Server. The current Primary Server thus becomes the Standby Server and vice versa.

The function is available for all redundancy modes (zenon Editor **Redundancy mode** property):

- Non-dominant
- Dominant
- Rated

Note: The redundancy switching function is only available if the **Network active** property has been activated.

APPLICATION EXAMPLES

Possibilities for use in practice are:

- Planned maintenance work on the Primary Server
- Better hardware coupling of the Standby Server
- In a rated network:
  Deactivate rating temporarily

> ⚠ **Attention**
>
> This function is not suitable for testing redundancy, as the behavior differs from that of a server failure.

SWITCHING DELAY

Some zenon modules can delay redundancy switching. For example, a **command** that is running delays redundancy switching.

Please note the module descriptions in the Behavior of zenon modules in the network (on page 148) in this manual in relation to this.

15.2.1 Engineering in zenon

CREATE REDUNDANCY SWITCHING FUNCTION

A **Redundancy switch** function serves to control the roles of **Primary Server** and **Standby Server** in a zenon Network.

ENGINEERING

Steps to create the function:

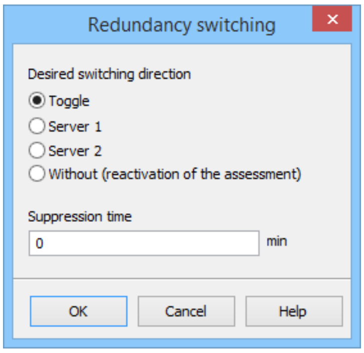
1. Create a new function:

   In the toolbar or in the context menu of the Functions node, select **New function**.
   The dialog to select a function is opened.
2. Navigate to the node **Network**.
3. Select the **Redundancy switch**.
   The dialog to configure redundancy switching (on page 147) is opened.
4. Configure the behavior of redundancy switching.
5. Name the function in the **Name** property.

> **Hint**
>
> Configure a separate redundancy switching function for each switching direction.

15.2.2 Configuration dialog for redundancy switching

You configure the action of a planned **Redundancy switch** in the **redundancy switching** dialog.



Desired switching direction

● Toggle
○ Server 1
○ Server 2
○ Without (reactivation of the assessment)

Suppression time

0 min

SWITCHING DIRECTION

Option	Description
Toggle	**Primary Server** and Standby Server switch roles.
Server 1	**Server 1** becomes (or remains) **Primary Server**.
Server 2	**Server 2** becomes (or remains) **Primary Server**.
Without (reactivation of the assessment)	No switching takes place. Instead, a possible switching for the **suppression time** configured in the properties field is prevented for the configured duration. If the **suppression time** is *0*, the switching is possible again immediately. **Note:** This option is only available in a rated network. If the evaluated network is switched to *dominant* or *non-dominant*, this selection must be replaced with a permitted switch the next time the function is opened.

SUPPRESSION TIME

Option	Description
Suppression time	The time in minutes within which no automatic redundancy switching takes place due to an amended rating. If the value is *0*, the automatic switching is carried out as configured in the rating principles.

Option	Description
	Range of values: *0* to *10080 minutes* (= one week) This means: The current Primary Server (**Server 1** or Server 2) remains the Primary Server for this time, regardless of the current result of the rating. If **Dead time after switching [s]** and the **suppression time** are configured differently in a rated network, the greater value prevails here. **Note:** This option is only available in a rated network.

NAVIGATION

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

> **Information**
>
> You can find further information in the Switching delay, down time and hysteresis (on page 90) chapter in this manual.

16 Behavior of zenon modules in the network

With network projects, the behavior of individual modules and functions in the network should be noted.

16.1 AML and CEL

ALARMING

The alarming is administered on the Primary Server. The Primary Server answers requests for alarming from the clients. Changes are synchronized between the Primary Server and Standby Server.

Information

If the Standby Server takes on the role of the Primary Server after the failure of the previous Primary Server, the missing data in the AML, CEL and archives is filled. The missing data comes from the internal buffer of the Standby Server. This buffer is supplied with values by drivers.

CHRONOLOGICAL EVENT LIST

The CEL is administered on the respective Primary Server. Changes are synchronized between the Primary Server and the Standby Standby Server.

In a rated network, no new CEL entries are created for regrading due to the rating.

Hint

In order to log the ratings in the CEL, create a system driver variable "**[Network] Rating Result of Server 1**" or "**[Network] Rating result of Server 2**":

- Select, in the **Workspace**, in the **Variables** node, the **New variable...** entry
- Select *SYSDRV - driver for system variables* as a **driver**.
- In the **Variables of the system driver** dialog, select the *Network* entry in the **Theme** entry.
- Select **[Network] Rating Result of Server 1** and **[Network] Rating Result of Server 2** and add these to the list of system driver variable to be created by clicking on the **Add** button.
- Create a numerical reaction matrix, activate the "**Treat each change of value as new limit violation**" and set the "**In Chronological Event List**" option
- Link this newly-created reaction matrix to the two variables.

16.2 Historian

Archiving is carried out on the Primary Server.

The Primary Server synchronizes the archive data with the Standby Server and responds to requests from the Clients (such as calling up an *Extended Trend*). screen).

Information

If the Standby Server takes on the role of the Primary Server after the failure of the previous Primary Server, the missing data in the AML, CEL and archives is filled. The missing data comes from the internal buffer of the Standby Server. This buffer is supplied with values by drivers.

REDUNDANCY SWITCHING

With redundancy switching, there may be a delay in switching by the zenon Historian module.

If variables from a different project are archived in an archive, the starting behavior of Runtime in the zenon network changes. In this case, the archives are only started once all projects have been loaded.

As a result, it is ensured that all variables to be archived are detected before the archiving starts and that the computer takes on the role of the Primary Server.

Example

zenon Runtime is started on a computer that has been configured as **Server 1** or **Server 2** during project setup.

- Runtime starts in the role of the Standby Server.
- All projects are loaded.
- The archiving is compared.

These steps are carried out regardless of the current role or evaluation of the computer.

Only once these steps have been completed is redundancy switching carried out - if necessary.

16.3 Batch Control

The module **Batch Control** is fully capable of using a network in terms of Client/Server technology. This means that Batch recipes can be created, duplicated, edited, deleted, etc. on a Client. The whole recipe management remains always on the server. Likewise the whole process control such as **start** recipe, **pause** recipe, **stop** recipe, etc. can be done from the Client. Also mode changes and manual operations such as **jump** are possible.

⚠ Attention

Module Batch Control does not support redundancy. There is no synchronization between Standby Server. When the Server breaks down, the executed Batch recipes are not continued on the Standby! Recipes can also be started whilst the configured Server 2 is the primary server.

For using Batch Control in a network the following is true:

ALLOCATION

The forcing of allocations can be carried out from the Server or Client.

FUNCTIONS

Functions are always carried out at the Server.

PHASES

- Editing phases in the master recipe:
  - Edit mode: Changes a done locally at the Client.
    If during the editing the recipe is saved on another computer in the network, the current configuration is lost. An appropriate message is displayed and the editing dialog is closed. The new data from the server are displayed.
  - Test mode: Changes a done at the Server.
- Control recipe: Changes a done at the Server.
- If a recipe is saved in the network, all Clients using this recipe are updated.
- If a recipe is opened on a client, the current version on the server is always displayed, even if it has not yet been saved there.
- If a recipe is deleted on a computer, a message is displayed on all computer on which the recipe is opened that the recipe has been deleted.

MODE

- The mode (automatic, semi-automatic, manual) can be switched by the server and the client.
- Jumps in the recipe and step-by-step progress of a recipe can be done from Server and Client.

RELOAD

Changes made to the recipes on the client that have not been saved can be overwritten when reloading.

RECIPES

- Recipes can be started and controlled by the zenon server and by zenon clients.
- If parameters in a recipe are changed whilst the recipe is saved on a different zenon client, the change to the parameters is refused and not carried out.
- A master recipe can be changed on the zenon client whilst it switches to test mode on the zenon server and is sent to the zenon client. The changes that were last saved are transferred. This means: If the zenon client saves last, the recipe is switched to editing mode again. If the zenon server saves last, the change to the zenon clients is discarded and the recipe is in test mode.
- If a communication error occurs when deleting a recipe or an operation template, the deletion is refused with an error message.

WEB CLIENT

With a standard web client:

- The settings for grid and color can be changed
- No recipes can be created or edited
- The size of the editing area cannot be changed
- In the toolbar, all symbols that are not permitted are deactivated; it is not possible to select the corresponding objects.

Web client PRO is not affected by these restrictions.

16.4 User Administration

User administration is administered on the Primary Server. Log-in procedure:

1. The login request is sent to the Primary Server.
2. It answers with the list of authorized users.
3. The client verifies the data.

If changes to user administration are made on a client in Runtime, the complete user list is sent from the client to the Primary Server.

Info

Active Directory, ADAM, ADLDS users

If Active Directory, ADAM or ADLDS users are used, all computers (regardless of whether they are the Primary Server, Standby Server or Client) communicate directly with the Active Directory, ADAM or ADLDS server.

This means that all computers must be in the corresponding infrastructure (such as Active Directory domains when Active Directory users are used); it is not sufficient that only the Primary Server is in the Active Directory domains with the corresponding users.

16.5 Files

Lists for the files of all modules are created when data is exchanged between the Primary Server and the Standby Server. The Primary Server monitors these lists for changes. Changes that are detected are transferred to the Standby Server.

⚠ Attention

The Primary Server does not react to watchdogs that are sent by the Standby Server when lists are created. Note this when stipulating the time for the network timeout.

With Remote Transport, all files required for the project are transferred to the target system.

In doing so, all files are always transferred to the folder:

Standard

- All files that are in the project's Runtime folder (**...\RT\FILES\zenon\system\**).
  These files determine the appearance and behavior of the project and are transferred as standard:

Info

Files with the following suffixes are not transferred by default:

- **.hot**
- **.ho**
- **.ret**
- **.re**

Optional

In addition, all files that are embedded into the project must be transferred. They are selected using the **Active** checkbox of the Remote Transport settings. These files are in the following subfolders of the project folder:

- \zenon\custom\graphics: for graphics
- \zenon\custom\lists: for language tables
- \zenon\custom\media: for all media files
- \zenon\custom\reports: for the reports of the Report Generator
- \zenon\custom\help: for help files
- \zenon\custom\additional: for additional files
- \zenon\custom\rdlc: for Report Viewer files
- \zenon\custom\drivers: for drivers
- \straton: for zenon Logic

  Recommendation: Project basis path, graphics, language tables, report tables and media files are always transferred.
  The following are transferred from the basis path by default: The files **project.ini**, **Projekt.vba**, **monitor.mon** and the **Projekt** folder.

  As a default zenon always uses relative paths and not absolute paths, so that the files can easily be found on the target system.

  For the files that can be transferred optionally, the original paths should be used (empty field under target), so that zenon can find them on the target system.

GLOBAL PROJECT

If there is a global project in the workspace, this is automatically transferred. No additional settings need to be made. Always all files necessary for the global project will be transported.

Attention

If the time difference between the server and the client is more than 5 seconds, no more files are synchronized.

16.6 Extended Trend

Extended Trend shows information from archives and online data. This data is saved in the Primary Server and requested by the Primary Server if required (if a trend screen is called up on the client).

16.7 Functions

For functions that are used in the network:

- The place of execution can be freely configured in some cases
- The place of execution is stipulated in some cases

Information

Scripts combine several functions. The place of execution then depends on the settings of the **Execute script** function. This setting overwrites the settings of the individual functions.

CONFIGURE PLACE OF EXECUTION

For functions where the place of execution can be freely configured, the corresponding parameters are available in the properties of the function.

To define the place of execution:

1. Navigate to the **Execution** group in the Properties.
2. Select the desired place of execution by checking the checkbox. Multiple selection is possible:
   - **Current computer**: Function will be executed on the current computer.
   - **Primary Server**: Function will be executed on the Primary Server.
   - **Standby Server**: Function will be executed on the Standby Server.
   - **Client**: Function will be executed on all clients.

OVERVIEW OF FUNCTIONS IN THE NETWORK

The following table shows which functions are executed and where they are executed.

Key:

- **Adjustable:** Behavior can be configured

  +: Yes

  -: No

  O: Default

- If not adjustable, O identifies the place of execution:
  - Active computer
  - Primary Server
  - Standby Server
  - Client

Function	Adjustable	Current computer	Primary Server	Standby Server	Client
AML and CEL					
Alarms: acknowledge flashing	-	O			
Delete alarms	-		O	O	
Acknowledge alarms	-		O	O	
Alarm/event group log in/log off	-	O			
Activate/deactivate Alarm Message List, alarm/event groups/classes	-		O	O	
Alarm Message List active	-		O		
Alarm Message List active/inactive	-		O		
Alarm Message List inactive	-		O		
Export AML	+	O			
Save AML and CEL ring buffer	-		O	O	
Export CEL	+	O			
Print AML or CEL	+	O			
Create/print IPA document	-		O		

Function	Adjustable	Current computer	Primary Server	Standby Server	Client
Switch online printing on/off	-		○	○	
Online printing start new page	+	○			
Switch online printer	-		○		
Application					
Select printer	+	○			
Start Load Management	-		○		
Stop Load Management	-		○		
Print Extended Trend diagram	+	○			
Switch palette	+	○			
Functions active at limit values	-		○	○	
Functions active/inactive at limit values	-		○	○	
Functions inactive at limit values	-		○	○	
Open help	+	○			
Reload project online	+	○			
Determine open maintenances	-		○		
PFS - execute user-defined event	+	○			
Activate/deactivate project simulation	-	○			
Simulate right click	+	○			
Save remanent data	+	○			
Exit Runtime	+	○			
Analyze S7 Graph heuristics	+	○			
Execute SAP function	+	○			
Language switch	+	○			

Function	Adjustabl e	Current computer	Primary Server	Standb y Server	Clien t
Topology - Search for ground fault	-		○		
Topology - Check connections	-		○		
Historian					
Archive: Stop	-		○	○	
Index archive	-		○		
Archive: Start	-		○	○	
Export archives	-	○			
Show open archives	-		○	○	
User Administration					
Change user	+	○			
Login with dialog	+	○			
Login without password	+	○			
Logout	+	○			
Change password	-	○			
Screens					
Change ALC source color	+	○			
Screen with index	-	○			
Close screen	+	○			
Screen: Return to last	-	○			
Screen: Move center	+	○			
Screen switch	+	○			
Activate input to the element with the focus	+	○			
Set focus to frame	+	○			

Function	Adjustable	Current computer	Primary Server	Standby Server	Client
Move focus	-	○			
Take focus away from frame	+	○			
Show menu	+	○			
Assign monitor	+	○			
Runtime profiles	+	○			
Close frame	+	○			
Setpoint input for keyboard screen	-	○			
Display overview window	+	○			
Fault locating in electric grids					
Acknowledge ground fault message	+	○			
Stop search for ground fault	+	○			
Start search for ground fault	+	○			
Acknowledge short-circuit message	+	○			
Message Control					
Save current queue	-		○		
Group/class/area/equipment suppressed	-		○		
Send a Message	-		○		
Send Message: activate	-		○		
Send Message: deactivate	-		○		
Network					
Operating authorization in the network	+	○			

Function	Adjustable	Current computer	Primary Server	Standby Server	Client
Redundancy switch	-			○	
Report Generator					
Print Report Generator	+				
Report Generator: execute	+				
Export Report Generator	+				
Recipes					
Recipe Group Manager	-	○			
Standard Recipe	-	○			
Standard Recipe single directly	+	○			
Standard Recipe single with dialog	-	○			
Standard Recipe single with online dialog	-	○			
Script					
Script: execute	+	○			
Script: select online	+	○			
Variable					
Export data	-		○		
Read dBase file	+	○			
Print current values	+	○			
Measuring unit conversion	+	○			
HD administration active	-		○	○	
HD administration inactive	-		○	○	
HD administration inactive/active	-		○	○	
Write set value	-		○		
Driver commands	-	○			

Function	Adjustabl e	Current computer	Primary Server	Standb y Server	Clien t
Transfer driver simulation image to the Standby Server	-			○	
Write time to variable	+	○			
Read time from variable	+	○			
VBA					
Open PCE editor	-		○		
Open VBA Editor	+	○			
Execute VBA macro	+	○			
Display VBA macro dialog	+	○			
VSTA					
Open VSTA editor	+	○			
Execute VSTA macro	+	○			
Display VSTA macro dialog	+	○			
Windows					
Play audio file	+	○			
File operations	+	○			
Start continuous tone	+	○			
Stop continuous tone	+	○			
Window to the background	-	○			
Window to foreground	-	○			
Print screenshot	+	○			
Start program	+	○			

16.8 Message Control

Message Control will be executed on the Primary Server.

The Primary Server synchronizes with the Standby Server.

16.9 Programming Interfaces

VBA AND VSTA

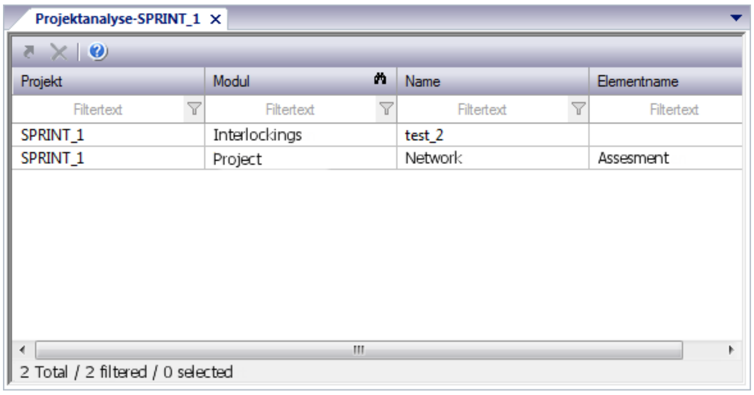
Code in VBA or VSTA is always, by default, executed locally on the system on which it is started or where events occur.

The place of execution can however be defined otherwise when this is called up via the function (on page 155).

PCE

PCE is always executed on the server in the network. On standalone computers in standalone projects.

16.10 Cross-reference list



Projekt	Modul	Name	Elementname
Filtertext	Filtertext	Filtertext	Filtertext
SPRINT_1	Interlockings	test_2	
SPRINT_1	Project	Network	Assesment

2 Total / 2 filtered / 0 selected

The use of variables in rated networks is also taken into account. The property name of the place of usage is stated as the element name. This is either "event variable" or "evaluations".

Information

You can find out further information in the cross-reference list manual.

16.11 Report Generator & Report Viewer

The ***.xrs** files of the Report Generator and the ***.rdl** files of the Report Viewer are synchronized on all systems in the network (Clients, Standby Server, Primary Server).

EDITOR

If the file is modified in the zenon Editor, transferred to the Primary Server and reloaded, it is automatically distributed to the other computers in the network.

RUNTIME

If the file is amended in Runtime, the changes are only saved on a temporary basis and replaced the next time a reload takes place or when Runtime is restarted.

16.12 Recipes

The execution of recipes is different for standard recipes and the RGM.

STANDARD RECIPES

Standard recipes are managed on the Primary Server and on the Standby Server.

If a standard recipe is changed by a user in Runtime, the client requests the full recipe list from the Primary Server. In the event of changes, the recipe list is sent back to the Primary Server.

> **Information**
>
> This list is not identical to the file rezepturen.cmp

If a recipe is changed and executed in Runtime on the client, it is executed with the new values. When closing the standard recipe, you are given the option to save the changes.

RECIPEGROUP MANAGER

When the Recipegroup Manager screen is loaded on the client, a list of all recipe names is requested by the Primary Server. As soon as a recipe is selected, it is loaded by the Primary Server.

16.13 Command Sequencer

The network concept of the **Command Sequencer module** works according to the following principle:

- The command sequences can be configured on the client, as well as the **Server** or on the **Standby Server**.

- The configured command sequences are administered on the **Primary Server** and distributed to the clients.
- The command sequences can be operated on both the client and the **Server**.
- The command sequences are always executed on the **Primary Server**.
- With redundancy switching, the command sequences are canceled.
  These can be restarted manually on the new **Primary Server**.

ROLE SWITCH BETWEEN SERVER 1 AND SERVER 2

- Redundancy switching is delayed until all active command sequences have been completed.
- The start of command sequences is blocked during a redundancy switching. The buttons on the client are grayed out in this time.
  - This redundancy switching can be planned in a **Rated network**.
  - In a **dominant network** or a **non-dominant network**, redundancy switching is carried out when the **Primary Server** fails.
- It is possible to start command sequences again once the switch has been carried out or if the switch has been completed.
  An entry is written to the CEL in this case.
- CEL messages are written for the following events:
  - Start of a command sequence on the server is blocked.
  - If a command sequence in the dominant network is to be started on the **Server 2** .
  - The command sequence cannot be started because there is currently a redundancy switch pending.
  - **Note:** No CEL message is generated if an incorrect command sequence is started.

LOG ENTRY

Entry	Level	Description
The sequence (mrid:<id1>, crid:<id2>)<name> could not be started, because a redundancy switch is pending.	**ERRORS**	The command sequence cannot be started because there is a redundancy switch pending.

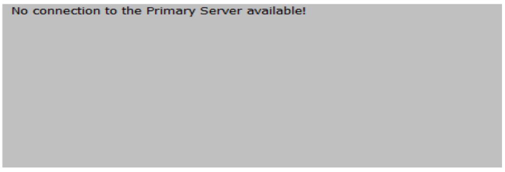
SPECIAL CASE: TWO SERVERS IN THE NETWORK

In the event that, when switching the **Primary Server**, there are still command sequences running on the "new" **Standby Server**, these are canceled on the Standby. This can only occur if both servers in the network were no longer connected (due to a network failure for example) and are now connected again. In this case, the change to the command sequence is not transferred to the Primary Server.

This means that, if there is a connection and command sequences that are now canceled on the Standby have already been opened on the Primary Server, these continue to be considered as running. It can only be restarted again once this command sequence has been closed on **Server 1** and **Server 2**.

NO CONNECTION TO SERVER AND STANDBY

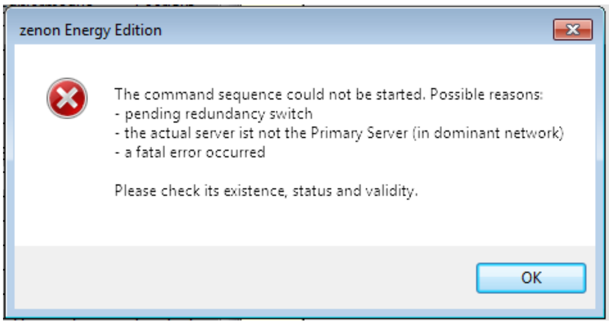
If the command sequence screen is opened on the Client when neither **Server 1** nor **Server 2** are contactable, the command sequences editor on the Client is not available. The command sequence editor remains empty. An error text is displayed in zenon Runtime instead of the command sequence image.



No connection to the Primary Server available!

ERROR DIALOG

If a command sequence cannot be started, the following error dialog is shown:



zenon Energy Edition

The command sequence could not be started. Possible reasons:
- pending redundancy switch
- the actual server ist not the Primary Server (in dominant network)
- a fatal error occurred

Please check its existence, status and validity.

16.14 Scripts

Scripts combine several functions. The place of execution depends on the settings of the **Script: execute** function. This setting overwrites the settings of the individual functions.

The execution of scripts in the network is controlled with predefined scripts:

Script	Description	Place of execution
AUTOSTART	The script is executed automatically when Runtime starts before the start screen is loaded if the project is the Runtime start project. It is not executed when subordinate projects are started.	▸ **Network project:** Primary Server ▸ **Single-user project**: Active computer

Script	Description	Place of execution
AUTOEND	The script is executed automatically when Runtime is ended if the project is the Runtime start project. It is not executed when subordinate projects are ended.	- **Network project:** Primary Server - **Single-user project**: Active computer
AUTOSTART_CLIENT	The script is executed automatically on a client when Runtime starts before the start screen is loaded if the project is the Runtime start project. It is not executed when subordinate projects are started.	**Single-user project**: Active computer
AUTOEND_CLIENT	The script is executed automatically on a client when Runtime is ended if the project is the Runtime start project. It is not executed when subordinate projects are ended.	**Single-user project**: Active computer
AUTOSTART_SRVPRJ	Script is executed automatically when Runtime is started for any desired project on the Primary Server before the start screen is loaded.	**Network project:** Primary Server
AUTOEND_SRVPRJ	Script is automatically executed when Runtime of a desired project is ended on the Primary Server.	**Network project:** Primary Server

16.15 Context lists

When using a **Context List** in a network project, saving is carried out on the server. Clients are synchronized automatically.
If a list is processed on several clients at the same time, the last-saved version is used by the server and distributed to all clients.

If the client loses the connection to the server, the **Context List** is emptied on the client and the screen elements are grayed out for editing. Linked entries in the Alarm Message List are shown with the text **<Alarm cause does not exist>**.
As soon as there is a connection to the server, the **Context List** is shown and the screen elements are released for editing.

16.16 Drivers in the zenon network

In the zenon network, the Primary Server normally communicates with the controller via the driver. Requests from the Client are routed via the Primary Server. This gets the information from the controller and forwards it to the Client. Limit values are monitored by the Primary Server.

DRIVER

Drivers are executed on the Primary Server and Standby Server.

Exception: The mathematics driver is only executed in the primary server.

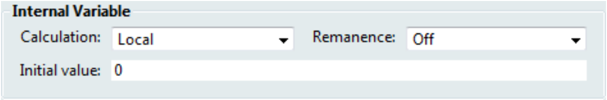
16.16.1 Driver for internal variable

With internal variables, it is possible in zenon to stipulate, for each variable, whether the variable is local on the computer or the same in the complete network.

> **Information**
>
> **Driver object type** *internal variables* are only available for the **driver for internal variables(INTERN)**.

To start calculating internal variables:

1. Navigate to the **Internal Variable** node in the properties of internal variables

   Internal Variable
   Calculation: Local
   Remanence: Off
   Initial value: 0

2. Stipulate the place of execution using the **Calculation** property:
   This property can be set individually for each variable. Arrays are also supported.

   a) *Local*: The internal variable is evaluated or managed locally for network projects, i.e. also on the Standby Server and the Clients. The values are not synchronized with other computers in the network.

   b) *Network*: With network projects, the internal variable is evaluated or administered on the project's Primary Server. It has the same value on the Primary Server and all Clients.

16.16.2 Drivers - Limit values and reaction matrices

In principle, limit values and reaction matrices are monitored on the Primary Server (where the AML is also administered).
The following applies for local internal variables in addition:

- Limit values and reaction matrices are monitored locally on all computers (Primary Server, Standby Server and Clients).
- Linked limit values are only executed where the limit value of the local internal variable was violated.

- Limit value violations of these variables on the Standby Server or Client are not an alarm.

16.16.3 Driver - regrading delay in the event of redundancy switching

When a configured server is started or restarted, the regrading to the Primary Server can be delayed by a driver.

This ensures that regrading only takes place if all process variables have a valid value.

> **Information**
>
> The regarding delay is only supported by very few drivers, such as the Siemens AK-Treiber for example
>
> You can find out whether a regrading delay is supported in the respective documentation of the driver.

LOG MESSAGES

Parameter	Level	Description
Redundancy Switch confirmation requested for project <Project name> module:VAR sequenceNo:<SequenceNo>	**Debug**	Confirmation that the Primary Server can be switched requested - *Project name* The project name - *SequenceNo* Serial number for the assignment of *request* and *response*.
Redundancy Switch confirmed for project <Project name> module:VAR sequenceNo: <SequenceNo>	**Debug**	Confirmation that the Primary Server can be switched.
Redundancy Switch delayed for project <Project name> module:VAR sequence id: <SequenceNo>	**Debug**	Delay activated for regrading.
Sample driver gave startup okay	**Debug**	Delay time on the driver expired.
me:<transport class> mod:<NetzwerkModuleId>(<network module name>) msg:<network module command> SeqId:<SequenceId>	**Debug**	LOG entry if cross-computer confirmation is necessary: - *Transport class*: Class name for the transport of

Parameter	Level	Description
dest:<Target computer> prj:<project name>		the command. Varies according to module. - *Network module ID*: ID of the network module. - *Network module name*: Name of the network module. - *SequenceId*: Serial number that is used to assign *request* and *response*. - *Target computer*: Computer on which the data is transferred. Optional. *S* denotes the active server - *Project name*: Name of the project for which the transport is executed.

16.17 zenon Web Client in the redundant network

In the event of a redundant zenon network configuration, the issuing of roles for the primary server and the standby server depends on the **Redundancy mode** set. In doing so, the role of the configured **Server 1** and **Server 2** computers can change over time in the Runtime, depending on the configuration of the **Redundancy mode** and the current evaluation (for **Redundancy mode** *evaluated*).

You should therefore configure both servers for the zenon Web Server. To do this, in the *global_vars.js* configuration file, change the line with the entry **RUNTIMESERVER=** and enter both computers there.

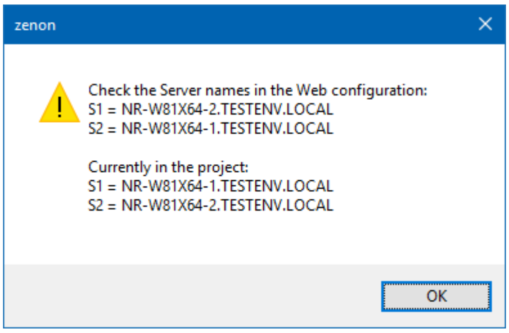
In doing so, the sequence should conform to the configuration in the zenon Editor.

You can find details on configuration in the configuration of global_vars.js chapter.

AMENDED SERVER CONFIGURATION

If the server names configured in the Editor do not correspond to the server names of *global_vars.js*, the zenon Web Client will not start.

If the configuration of the server is amended for a running system in zenon, the "*Runtime is busy* dialog will be shown in the zenon Web Client.



Check the Server names in the Web configuration:
S1 = NR-W81X64-2.TESTENV.LOCAL
S2 = NR-W81X64-1.TESTENV.LOCAL

Currently in the project:
S1 = NR-W81X64-1.TESTENV.LOCAL
S2 = NR-W81X64-2.TESTENV.LOCAL

After a project synchronization, the currently-running and the actual project configuration will be shown in a further dialog. In this case, the browser window must be closed by the user and the zenon Web Client must be restarted.

Attention

If the project configuration of **Server 1** and**Server 2** is changed in the zenon Editor, the *global_vars.js* file must also be amended accordingly.

You can find further information in the zenon web server manual in the configuration of global_vars.js chapter.

16.18 Time Control

Time control will be executed on the Primary Server. The function triggered is executed on the systems that were selected for execution of the function in the settings.

16.19 Allocations

Allocations are always executed on the Primary Server.

Attention

This is relevant to local internal variables most of all. These are not executed on the Standby Server or on Clients!

17 Network messages from the system driver

The following system driver variables are available for this subject area:

Name	Data Type	Comment
[Network] Current Primary Server	*STRING* *lokal*	Computer name of the current Primary Server If the name was acquired from the host file, it will be the name used there. With DNS, this is the Fully Qualified Domain Name. **Note:** If the network is deactivated, the variable sends the status INVALID. The **[Network] Current Standby Server** variable remains empty in contrast.
[Network] Current Standby Server	*STRING* *lokal*	Computer name of the server which is currently not handling processes. If the name was acquired from the host file, this is the name entered there. With DNS, this is the Fully Qualified Domain Name.
[Network] Number of connected clients	*UDINT*	Delivers the number of clients currently connected to the server. This number also includes the standby server, if there is one.
[Network] Authorization denied	*BOOL*	Shows whether a requested authorization is denied in the network. The value of this variable is changed only for a short time and then changed back to the initial state. - *0* = operating authorization request accepted - *1* = operating authorization request declined
[Network] Authorization available	*BOOL*	Shows whether there is an authorization for the current project on the local computer. - 0 = No - 1 = Yes
[Network] operating authorization: Computer that has it	*STRING*	Shows the name of the computer that has the authorization for the currently loaded project.
[Network] Evaluation result of Server 1	*UDINT*	In the event of changes to a variable from the evaluation matrix, this value is written to the corresponding system driver variable for **Server 1** and **Server 2** after calculation of the new result of the

Name	Data Type	Comment
		evaluation. The values are equal to one another (server <-> standby), so that the current value is always displayed on both sides. However, after the other side has a failure, this value remains for the attendant variable and only updates itself once it reconnects. **Note:** You can find further information on evaluation in the Network (on page 7) manual in the Configuration of redundancy evaluation (on page 87) chapter.
[Network] Evaluation result of Server 2	*UDINT*	In the event of changes to a variable from the evaluation matrix, this value is written to the corresponding system driver variable for **Server 1** and **Server 2** after calculation of the new result of the evaluation. The values are equal to one another (server <-> standby), so that the current value is always displayed on both sides. However, after the other side has a failure, this value remains for the attendant variable and only updates itself once it reconnects. **Note:** You can find further information on evaluation in the Network (on page 7) manual in the Configuration of redundancy evaluation (on page 87) chapter.
[Network] Names of connected clients	*STRING*	xxx Delivers the names of the clients currently connected to the server. The standby server, if there is one, is also included.
[Network] Primary Server <-> Standby Server in data sync	*BOOL*	A binary variable that takes the value *1* (for a short time) when the system performs a redundancy switch between server and standby server. ▸ *0 = no file sync* ▸ *1 = file sync active*
[Network] Primary Server broke down	*BOOL* *lokal*	Indicates that the connection to the process handling server was lost. Depending on the network position of the computer, this means:

Name	Data Type	Comment
		- Dominant Server: While it is not yet the process handling server, the value changes to TRUE if the connection to the process handling server is lost. Always FALSE after synchronization. - Non-dominant Server: Changes to TRUE if the connection to the dominant server, which was the process handling server, is lost. Changes back to FALSE if the Standby Server was promoted to be the Primary Server. EVALUATION: Preferably via a reaction matrix (REMA), as the Alarm Management is also swapped and taken over by the SB at that time. The Online Container is also not suitable because the variables are re-initialized during redundancy switching. Client: Changes to TRUE if the connection to the process handling server is lost. Changes back to FALSE if the client connects to the SB computer that is now the process handling server.
[Network] Primary Server shut down	*BOOL* *lokal*	Indicates the regular stop of the process handling server. The value changes to TRUE if the Primary Server was stopped properly. FALSE if there is a process handling server in the net. Depending on the network position of the computer, this means: - Dominant Server: While it is not yet the process handling server, the value changes to TRUE if the process handling server has stopped. - Non-dominant Server: Changes to TRUE if the dominant server, which was the process handling server, has stopped. Changes back to FALSE if the StandBy was promoted to be the process handling server. - EVALUATION: Preferably via a reaction matrix (REMA), as the Alarm Management is also swapped and taken over by the SB at that time. The Online Container is also not suitable

Name	Data Type	Comment
		because the variables are re-initialized during redundancy switching. - Client: Changes to TRUE if the dominant server has stopped. Changes back to FALSE if the client connects to the SB computer that is now the process handling server. Is also TRUE while the process handling non-dominant server changes back to be the non-process handling server.
[Network] Standalone/Primary Server/Standby Server/Client	*DINT*	Shows the type of the local computer in the network. - *-1* = Standalone - *0* = Client - *1* = Primary Server - *2* = Standby Server
[Network] Standby Server broke down	*BOOL*	Changes to TRUE if the connection to the currently non-process handling server is terminated unexpectedly. If there is a connection, the value is FALSE. Depending on the network position of the computer, this means: - Dominant Server: The variable only acts as described from the time when the standby became the server handling the process. - Non-dominant Server: If, during file synchronization, the connection to a server that is dominant but is not handling the process is interrupted, the value changes to TRUE. Always FALSE if not the server handling the process. - Client: As per server handling the process.
[Network] Standby Server shut down	*BOOL*	Is TRUE on the process handling server, if the non-process handling server was stopped properly and if there is no connection anymore. Changes to FALSE if the non-process handling server has registered at the process handling server.

Name	Data Type	Comment
		Depending on the network position of the computer, this means: - Dominant Server: Only from the time when the standby became the server handling the process does the variable act as described. - Non-dominant Server: If this is ended during file synchronization with a server that is dominant but is not handling the process, the value changes to TRUE. Always FALSE if not the server handling the process. - Client: As per server handling the process.
[Network] Standby Server started	*BOOL*	Is TRUE if the non-Primary Server has signed into the Primary Server, the file sync was carried out and the connection between the two computers is active. Depending on the network position of the computer, this means: - Dominant Server: Only from the time when the standby became the server handling the process does the variable act as described. - Non-dominant Server: Becomes TRUE if the dominant server not handling the process starts. Changes to FALSE if the computer is the server handling the process. - Client: As per server handling the process.
[Network] Timeout [ms]	*UDINT*	Shows the timeout in milliseconds for the zenon network as configured in the project configuration.
[Network] Switch from Primary Server to Standby Server	*BOOL*	A binary variable that takes on the value 1 if the server becomes the standby server during a redundancy switch. - *0*= registered server is available as server in the network. - *1* = registered server is available as standby server in the network.
[Network] Switch from Standby Server to Primary	*BOOL*	A binary variable that takes on the value 1 if the standby server becomes the server during a

Name	Data Type	Comment
Server		redundancy switch. - *0* = registered Standby Server is available as standby server in the network. - *1* = registered Standby Server is available as server in the network.

-

18 Visualization of the connection in zenon Runtime

The display of the connection status in Runtime is configured with the **Display status of variable** screen element property.

This property is activated for all zenon screen elements by default. Additional project configuration steps are thus not necessary. The network status is, if the connection is interrupted, visualized with a blue dot in zenon Runtime.

VISUALIZATION IN THE EVENT OF INTERRUPTION ("BLUE DOT")

- If a client loses the connection to the sever, this is visualized on the client in Runtime:
  All variables that are supplied with values via the network are marked with a blue dot in Runtime in the event of a breakdown in internal communication.
  - This is also applicable for internal driver variables that are supplied with data via the server:
    Examples:
    - Internal driver variable, with configured *Netzwerk* value in the variable **Calculation** property. This property is in the **Internal Variable** properties group.

    - System driver variable **[Network] Current Standby Server**
- If a variable on the server needs to be be requested on the client, it gets the blue dot immediately.
  - Values from the SB are sent by the standby server and visualized with a blue dot on the primary server in the event of a failure of the standby server.
  - This blue dot disappears as soon as the source of the data point is connected to the primary server. This results in the client with valid data.
    A source can be both the standby as well as the primary server.
- Blue dots are displayed if:

- The primary server has no communication to the standby server and the **Read from Standby Server only** variable property has been activated. You can find this property in the **Addressing** variable properties group.
- The client has no communication to both servers. In this case, communication to the configured **Server 1** **AND** **Server 2** is interrupted.

- With redundancy switching during the time of switching. This time period is very short.
  Note: Data does not get lost as a result.