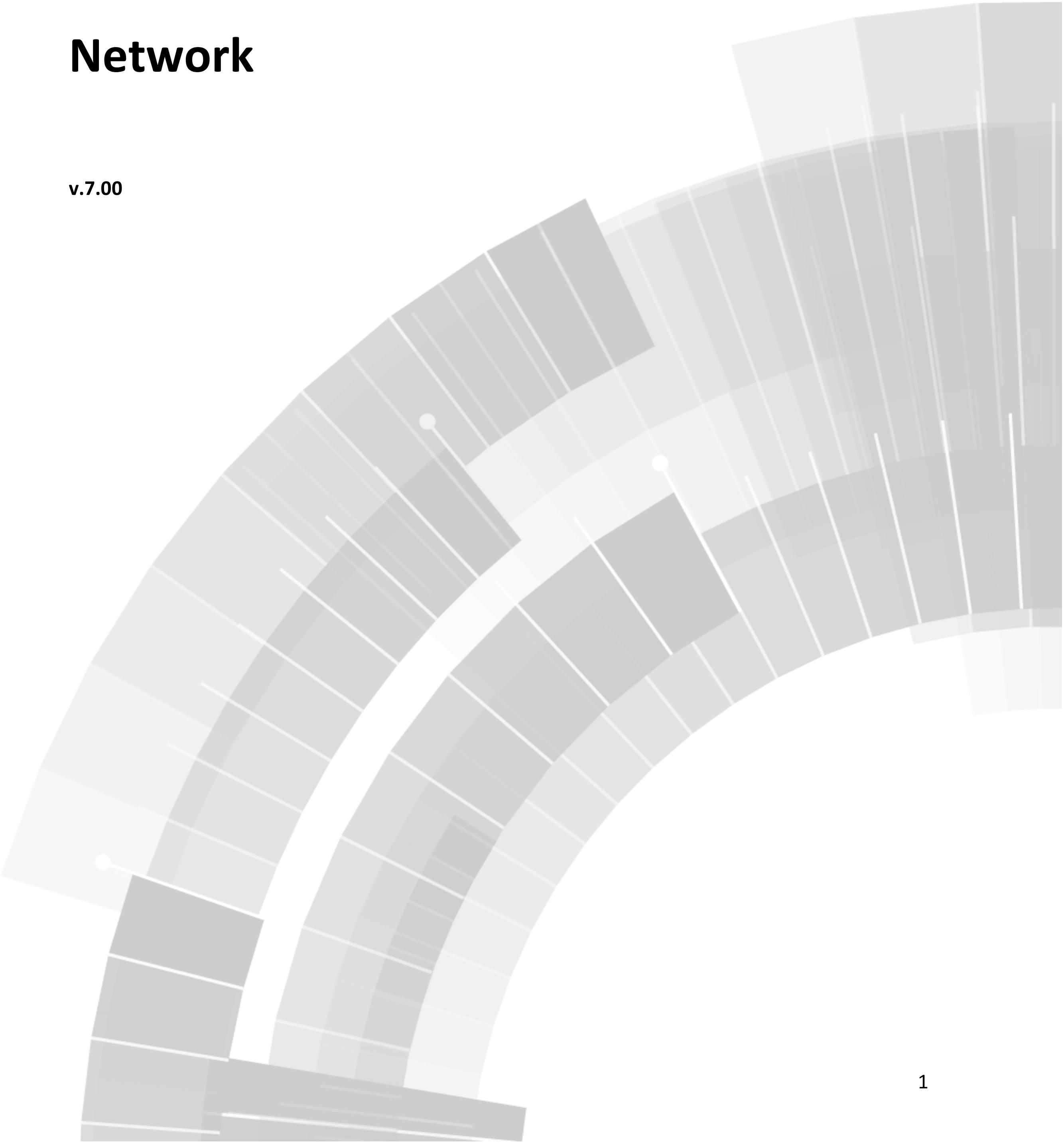
COPA-DATA
do it your way

zenon manual

Network

v.7.00



Contents

1. Welcome to COPA-DATA help

GENERAL HELP

If you miss any information in this help chapter or have any suggestions for additions, please feel free to contact us via e-mail: documentation@copadata.com (mailto:documentation@copadata.com).

PROJECT SUPPORT

If you have concrete questions relating to your project, please feel free to contact the support team via e-mail: support@copadata.com (mailto:support@copadata.com)

LICENSES AND MODULES

If you realize that you need additional licenses or modules, please feel free to contact the sales team via e-mail: sales@copadata.com (mailto:sales@copadata.com)

2. Network

zenon networks can be set up and configured very quickly and securely.

zenon in the network makes it possible for you to:

- Create distributed systems (decentralization)
- Work on several stations at the same time (see distributed engineering)
- To have full access to configuration and Runtime of different computers

- Use of web server and web client for mobile access
- Have an overview of all workspaces (actions such as acknowledgement of alarms at a workspace are visible at all others)
- Centralized logging and archiving

zenon supports client-Server, (on page 17)multi-Server (on page 23)und multi-client-multi-server. (on page 25) as network topologies (on page 14)

> **License information**
>
> *Part of the standard license of the Editor and Runtime.*

SIMPLE ADMINISTRATION OF THE ZENON NETWORK

The well-though-out network functionality of zenon makes it possible to implement projects on different servers and to create complex network constellations (on page 14) very quickly. The individual stations can be configured in such a way that only the project contents that are necessary for activities at the respective location are visible. The zenon Editor supports users when administering the network.

The integrated topology administration (on page 71) creates the interrelationships for the individual projects with the attendant servers in graphical form. A testing routine checks the configured structure to see that it is complete and that there are no configuration errors. Configuration errors are determined quickly. With the network nodes function, zenon also checks to see if the selected network topology can function.

> **Info**
>
> *With network projects, note the computers on which modules and functions (on page 106) can be administered and executed.*

WAN

Within a network, zenon transfers data spontaneously – and is thus already optimized for use with in a WAN. Depending on the configuration, it is also recommended that the watchdog traffic (on page 13) between the client and server is limited. The communication distance between client and server can be automatically closed. The routers then establish a new connection when data exchange is absolutely necessary.

WEB SERVER

zenon Webserver allows access to Runtime via the intranet or internet. No adaptations to the project are necessary. Access is gained via the web client. This offers the same look & feel as zenon Runtime. 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.

3. Requirements

Using zenon in a network requires a running Windows network. The display of the network computer in Windows Explorer is not sufficient to guarantee that the zenon network is functional.

GENERAL

The following requirements must be met:

- 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 program opens port `1100`. This must therefore be reachable remotely and may not be blocked by a firewall.

zenon networks function securely with these operating systems:

- Windows XP
- Windows Vista
- Windows 7
- Windows Server 2003/2008

IPV6

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

- Network configuration in the Startup Tool

  or

- in zenon6.ini

If this setting is changed, all ongoing zenon processes must be restarted. This concerns zenAdminSrv, zenSysSrv, zenLogSrv and zenDBSrv in particular.

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

- Driver communication with the PLCs
- Protocol communication in the Process Gateway plug-ins
- Workbench and Runtime communication in zenon Logic

> ⚠ **Attention**
>
> *IPv6 only works with version 7 onwards. No versions prior to version 7 can be started if this is active.*

PORTS USED

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

Service	File	Goal	TCP-port
Network service	**`zenNetSrv.exe`**	Runtime communication.	1100
Transport service	**`zenSysSrv.exe`**	Data transfer via Remote Transport (editor) and Diagnosis Server.	1101
zenon Webserver	**`zenWebsrv.exe`**	On-site logging machine between web client and Runtime	1102

CHECK THE REQUIREMENTS

NAME RESOLUTION

To check the name resolution:

1. Start the windows command line (cmd.exe)
2. Execute 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 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. Execute the following command: **`telnet COMPUTERNAME 1100`**
4. The command line window turns completely black as soon as a connection is established (to end the command line: `Close` command in the context menu or click on **`x`** on the program window), otherwise an error message is displayed

Note: *With Windows Vista/7, you must first activate the* **`Telnet-`** *command. You can find instructions for this in the operating system help pages (search for:* `Telnet)`.

3.1 Time synchronization in the network

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

In a topology with circular redundancy (on page 88) or with several servers, it is recommended that time synchronization is carried out using DCF (radio controlled clock) or the Windows commands. In this case, the automatic time synchronization in zenon must be deactivated.

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

DEACTIVATING TIME SYNCHRONIZATION IN ZENON

If the time synchronization is to be turned on or off manually, an entry in zenon6.ini is necessary:

```
[Netz]

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

> **Example**
>
> *[Netz]*
>
> *TIMESYNCH=0*

EXTERNAL TIME SYNCHRONIZATION USING THE OPERATING SYSTEM

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

The current server is the active time master in the classic client server topology. It should update itself with DCF (radio controlled clock) or via the Windows `Time` commands. The clients get the current time from the server (depending on the defined timeout) and update their times accordingly. Communication is carried out via SNTP (System Network Time Protocol), which takes the delay time into account.

Info

Watchdog

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

With the standard settings of 30 seconds for the `Timeout [s]` *property, each client's network service (**`zenNetSrv.exe`**) sends a watchdog to the server's network service (**`zenNetSrv.exe`**) every 10 seconds during online operation. If the 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 project properties:

`Network` *node,* `Timeout [s]` *property). The user, who is logged on the client needs authorization to change time on the computer.*

zenon6.ini

On clients or remote computers without zenon Editor, the setting is made via zenon6.ini:

```
[Netz]

NET_TIMEOUT_MSEC=30000
```

(time-out in milliseconds, default: 30000.)

Note the additional configuration necessary in WAN (on page 13).

COMMANDS UNDER WINDOWS

For external synchronization using Windows, enter the following command with the respective necessary arguments in the console for command input:

```
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 computers 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 and/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

3.1.1 Time synchronization in the WAN

In the WAN and for dial-up connection, the standard defined value of 30 seconds for the watchdog means that the connection is not maintained permanently.

Select a `Timeout [s]` time in the WAN, which only makes connections at the desired intervals. Note: The longer the time-out, the later server failures are detected. For example, if you select 64,8000 as the time for `Timeout [s]`, then the time-out time is 18 hours. A connection is made every 6 hours and a watchdog is sent. A server failure is thus only noticed after 18 hours.

Info

A fixed time-out time of 30 seconds is always used when a client 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 Server in the last 30 seconds.
- If this is not the case, a watchdog is sent to the 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. Navigate to the
   `[NETZ]` section.
3. Create or edit the entry

   ```
   NET_CONNECTWAIT_MSEC=30000
   ```

   This defines the value for a reconnect in milliseconds.

   Maximum value: Time-out time
4. Create or edit the entry

   ```
   NET_CONNECTCOUNT=0
   ```

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

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

4. Setting up the zenon network

In the zenon network, you work with:

Parameters	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 11). 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.
`Data server:`	The Data server is a computer with direct communication to the process but no server tasks. In normal operation, the data server sends all data to the Server, which manages data storage and data distribution. In the event of a network or server breakdown, the data server continues to work as a single station and therefore guarantees proper operation of the process. The data is transferred as soon as the server connection is re-established. **Area of application:** An on-site operating system without powerful hardware, such as an IPC or CE terminal. Complete data archiving and data distribution is carried out by the data server. On-site operation is still possible in the event of a server failure. This is not the case in a classic client server network.

Info

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 loaded subsequently. They are only updated by restarting Runtime.

TOPOLOGIES

zenon supports several network topologies:

- Client server network (on page 17): The same project runs on the server and all clients.

- Multi-server network (on page 23): A client can access different servers and thus display the data of different projects at the same time.
- Multi-client-multi-server model (on page 25): All clients and servers communicate with each other. Other projects can be accessed from each project. Configuration is carried out in zenon

CONFIGURING THE NETWORK

To make a network network-compatible:

1. navigate to the `Network` node in properties
2. Activate the `Network active` property
3. Use the `Server` 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:

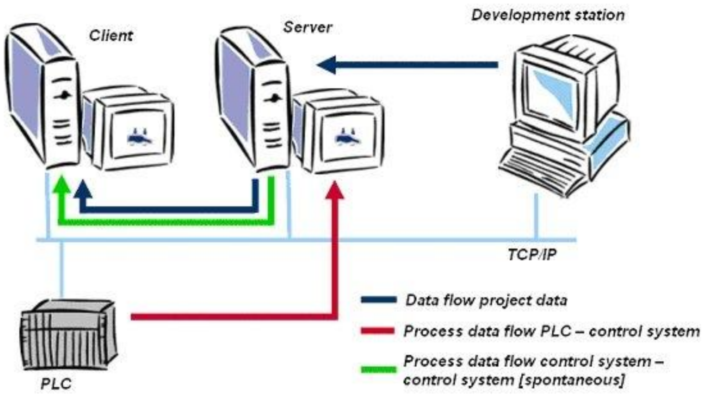
- Routing (on page 94): `Routing active` property
- Standby server (on page 87): `Standby` property
- Redundancy (on page 81): `Redundancy type` property
- `Termination message`: Defines if, when Runtime is ended on a server, the clients are informed 70 seconds in advance

With this, you have configured the basic properties. Repeat these steps for all clients in the zenon network. Remote computers can also be set up using remote transport (on page 58).

> ⚠ **Attention**
>
> *Naming for server and standby server:* **`"localhost"`** *must not be used.*

4.1 Client-server model

In the client-server-model, the client and server use the same project. The server project is started as the current project on the client.


Client
Server
Development station
TCP/IP
PLC
Data flow project data
Process data flow PLC – control system
Process data flow control system – control system [spontaneous]

To set up the client-server model, the following is set up on each computer:

- The `Network active` property is activated
- The name of the server is entered in the `Server` property

Recommendation: The server should be the most powerful computer in the network.

In the zenon client-server network:

- Only the server has a direct connection to the PLC
- The server administers all process data (such as online data, archive data, alarms, recipes, etc.)
- The server administers all project data (such as screens, functions, defined variables, etc.)
- Each other computer that starts the same project is automatically recognized and defined as a client.
- Each time Runtime starts, each client makes
  - the connection to the server,
  - synchronizes the project data and
  - displays the current process data

> **Info**
>
> *Windows CE is fully integrated. Systems under Windows CE can be used as a server or client.*

4.1.1 Configuring the server

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

To set up the server:

1. Activate the `Network active` property
2. Use the `Server` 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.
3. Note the correct configuration of the internal variables
4. Create AUTOSTART and AUTOEND scripts for the clients if necessary

> **Attention**
>
> *Naming for server and standby server:* **`"localhost"`** *must not be used.*

If the development computer on which you created the project is also the Runtime server, configuration of the server is now complete.

CONFIGURATION OF INTERNAL VARIABLES

If variables have already been created before the project was defined as a network project, internal variables are defined locally. They are executed locally on each client. If these variables are also to be synchronized with the server in the network:

1. Navigate to the `Internal Variable` group in the internal variable properties.
2. Select `Network` in the `Calculation` property drop-down list

- `Local`: The variables are administered on the client. The value of the variable can be different on each client. Limit values are also evaluated locally.
- `Network`: The variables are evaluated and administered on the project's server. It has the same value on the server and all clients.

SCRIPTS FOR CLIENTS

If special screens are started or functions are executed on the clients, the corresponding scripts must be set up accordingly:

- AUTOSTART_CLIENT: Is executed when the system starts and defines all processes to be executed when the client is restarted. When a special start screen is activated, for example.
- AUTOEND_CLIENT: Is executed when the system is stopped and defines all processes to be executed when the client is stopped. Resetting outputs, for example.

Both scripts run on the server.

4.1.2 Configuring the clients

Clients can be set up manually or by means of Remote Transport. Setup using Remote Transport is recommended.

For this, the following applies:

- If the development station also is a client, simply start the Runtime there.
- You set up all other clients either via Remote Transport (on page 20) or manually (on page 20).
- If special processes are to be executed on the clients, a respective script on the server (on page 18) must be created, which defines the behavior on startup (AUTOSTART_CLIENT script) and when being ended (AUTOEND_CLIENT script).

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 properties. To set up clients from the development computer by Remote Transport, the transfer direction 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.
3. The **`Remote Transport`** (on page 58) dialog is opened.
4. Enter, in **`Connection`** under **`Name`**, the name of the client in the network.
   Separate several clients with a semi-colon (`;`).
5. Close the project configuration with **`OK`**.
6. Establish an online connection to the client.
   To continue to use Remote Transport (on page 58), it is best to use the toolbar symbols.
7. Transport all Runtime files to the client with Remote Transport.
8. Set the start project for the client with Remote Transport.
9. Start the Runtime on the client with Remote Transport.
10. Stop the online connection.

Setting up the client manually

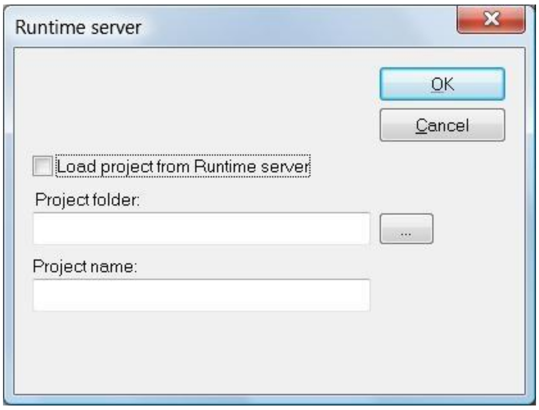
To configure clients for the start of Runtime:

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

This entry defines which project is to be loaded when Runtime is started.

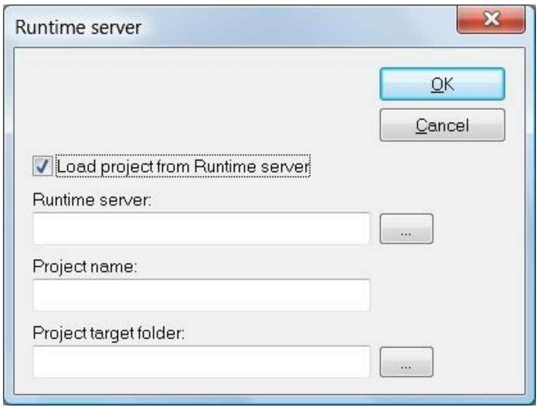
4. Leave the Editor closed and start Runtime.

5. A request is made in a dialog, requesting which project is to be loaded.



□ Load project from Runtime server

6. Activate the `Load project from Runtime server` checkbox.



☑ Load project from Runtime server

7. Enter:

   a) Runtime server: Computer that is set up as the server (on page 14). The name can be entered directly or selected from a list using the `...` button.

   b) Project name: Name of the project that runs on the server.

   c) Project target folder: Folder for Runtime on the client's local hard drive. You either can select an existing directory using the ... button or type it in by hand. If a folder that does not exist is entered by hand, this is created automatically.

   d) Confirm the settings with OK.

8. zenon Runtime:

   a) Now creates a connection to the Runtime server

   b) Copies its Runtime files to a project target folder

   c) Starts Runtime

d) Requests a restart of Runtime if necessary

9. The entry `VBF30=...` in the `zenon6.ini` file is set on the project target folder.

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

Repeat this process for each client.

Behavior in Runtime

Network projects can be operate in the same way in the network by the server and clients 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 20) section.

Differences between the server and client:

- Only the server of the project has a connection to the hardware and administers the process data.
- The clients receive, from the server:
  - Current values of the variables
  - Chronological Event List system messages
  - Alarm information
  - Recipes
  - Archived data

The transfer of data to the client is spontaneous and event-controlled.

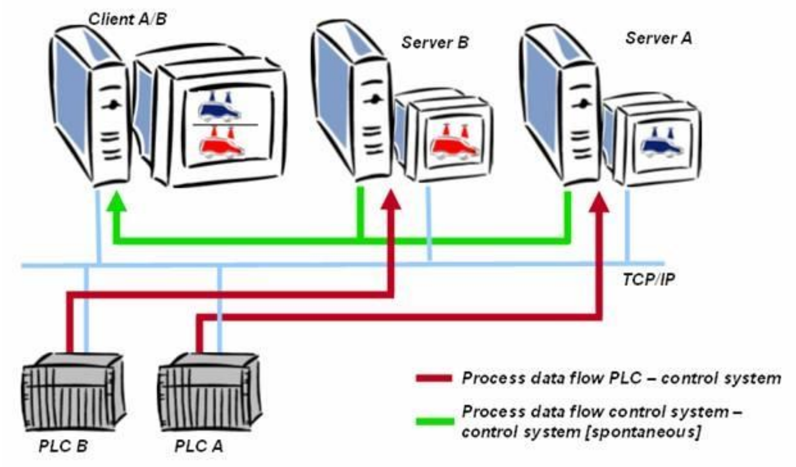
MONITORING THE CONNECTION

With the standard settings of 30 seconds for the `Timeout [s]` property, each client's network service (**`zenNetSrv.exe`**) sends a watchdog to the server's network service (**`zenNetSrv.exe`**) every 10 seconds during online operation. If the server responds to at least one of the three watchdogs within the 30 seconds, the client assumes that the network connection is working. This standard setting can be changed in the Editor in the network configuration: `Network` section of project properties, `Timeout [s]` property with default of 30 seconds.

4.2 Multi-server model

zenon defines and differentiates in a project between:

- Server
- Standby server
- Clients


Client A/B
Server B
Server A
TCP/IP
PLC B
PLC A
Process data flow PLC – control system
Process data flow control system – control system [spontaneous]

It is therefore possible to have different servers, clients and standby projects running at the same time and in parallel on one computer without problems.

In combination with the ability of the Project Manager to create hierarchical tree structures, this offers numerous new possibilities for network applications.

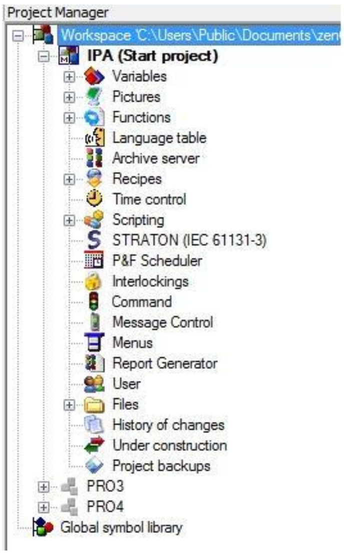
The multi-server model makes a distinction between:

- Substations
- Main stations
- Headquarters

The single computers are servers for their projects.

HIERARCHY

The hierarchy of the projects is defined in the Project Manager. To do this, drag & drop a project to a superordinate project.



```
Project Manager
Workspace 'C:\Users\Public\Documents\zen
  IPA (Start project)
    Variables
    Pictures
    Functions
    Language table
    Archive server
    Recipes
    Time control
    Scripting
    STRATON (IEC 61131-3)
    P&F Scheduler
    Interlockings
    Command
    Message Control
    Menus
    Report Generator
    User
    Files
    History of changes
    Under construction
    Project backups
  PRO3
  PRO4
  Global symbol library
```

Mayor projects have access to all the minor projects under them. Subordinate projects can access projects at the same level and superordinate projects. For problem-free access, note:

- All projects involved must have the `Network active` property active
- The target project must run on a server,
  this server computer must be able to be reached by the computer making the request
  (the server has started the integration project and all subprojects.)
- The project making the request must be started on the computer making the request

Example: If Project **`SUB1`** on **`Computer B`** wants to have access to the superordinate project **`Integration1`** on **`Server A`**, the following must happen:

- **`Server A`** must be reachable from **`Computer B`**
- For **`Integration 1`** and **`SUB1`** the respective `Network active` property must be set to `active`
- **`Integration 1`** must be started with all subprojects on **`Server A`**
- **`Integration 1`** and **`SUB1`** must be started on **`Computer B`**

⚠ **Attention**

Project names have to be unique in multi-server systems.

4.3 Multi-client-multi-server model

In this model, a PC can be server for one project and can at the same time be client for another project. A single PC can even be multi-server (server for several projects) and multi-client (client for several projects) simultaneously. This structure is implemented via the zenon multi-project administration (on page 25).

The multi-client model forms the basis of:

- zenon Horizontal Transparency (on page 40)
- zenon Circular Redundancy (on page 88)

4.4 Multi-project administration

Multi-project administration makes decentralized solutions possible. Subprojects can be operated as server projects on different computers.

The following is possible with this:

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

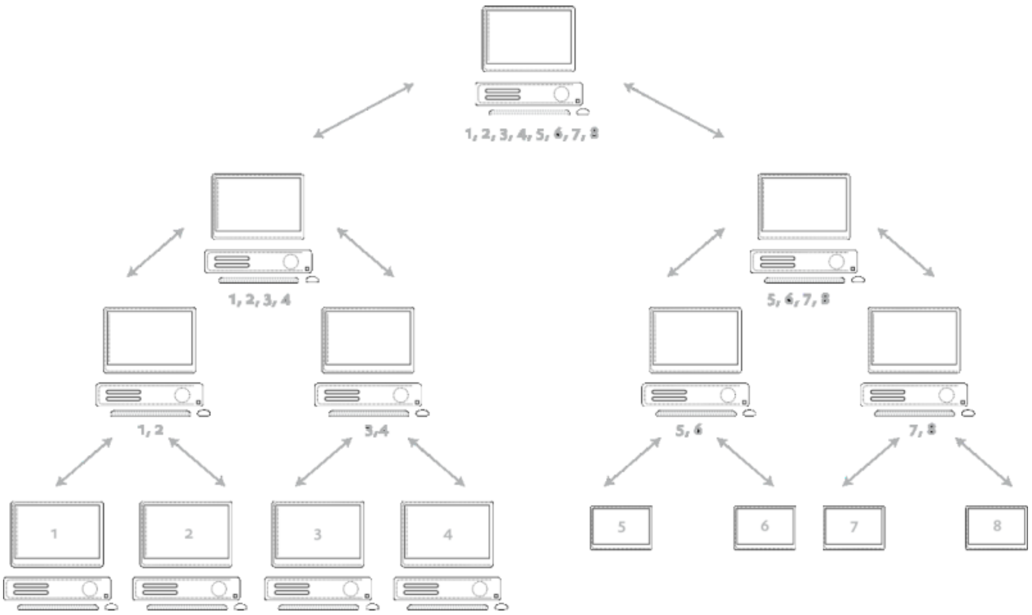
Info

Multi-project administration is not available under zenon Operator. Here, only one project and a global project can be created and administered.

DESIGN

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.



Info

Configure the topology and check it with the zenon network topology (on page 71).

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.

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.
- Sophisticated load sharing.
- Cross-project operation, as all projects on a processor are simultaneously activate.
- Multiple-hierarchy network structure allows the centralization of data (measured values, alarms, plant information, archive data, etc.) in a higher-ranking level.
- No limit on projects per processor.
- Centralization in large control rooms is possible.
- Node structure – physical network separation.

4.4.1 The integration project

The integration project administers subprojects that can be accessed in Runtime. The I-project can be used in multi-project administration as a pure administration project or alternatively as a fully-fledged project. If the integration project is used as a start project, all sub-projects are automatically started in Runtime.

In an integration project, you can create central Alarm Message Lists or Chronological Event lists for all integrated projects with a few mouse clicks. Thus, for example, all alarms of the projects in the Alarm Message List of the integration project are displayed and chronologically sorted.

> ⚠ **Attention**
>
> *When designing the multi-project administration, ensure that the navigation (on page 30) works.*

4.4.2 Definition of the structure in the Editor

In the Editor, the structure of the network is created by simply dragging & dropping. You also need an integration project (I-project), which administers all other projects, in addition to the productive projects. Because single-user projects do not send data to clients, a server must be defined in each project. The integration project can also be a productive project.

EXAMPLE

Three projects are used in this example:

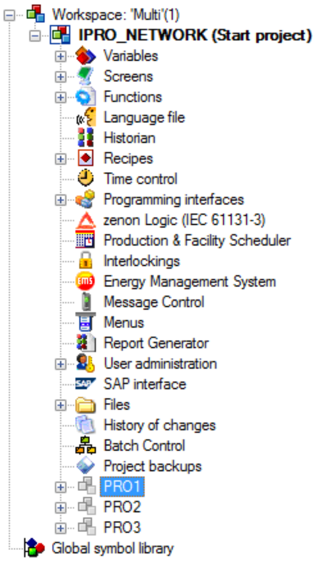
- Productive project `PRO1`
- Productive project `PRO2`
- Integration project `I-PRO`

To create the structure:

1. Create:
   - I-PRO
   - PRO1
   - PRO2
   - and define a server each
2. In the Project Manager, drag PRO1 to I-PRO by holding the left mouse button

3. Do the same for PRO2

4. PRO1 and PRO2 are now displayed in the Project Manager as branches of the I-PRO



With this, the hierarchical structure of the network has been created.

Info

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

4.4.3 Transferring and starting projects

In the topological structure, all subprojects of an integration project are automatically transferred to the respective targets. All subprojects are also started if the integration project is also the start project.

For details of network topology, read the Administering and checking network topology (on page 71) chapter. You can find the configuration of the computer with an example for automatic transfer of subprojects in the Configuration of computers in the network (on page 77) section.

TRANSFERRING AND STARTING PROJECTS MANUALLY

To transfer or start projects manually:

- Transport all Runtime files of PRO1 to the according server with Remote Transport.
- Set the start project for the server of PRO1 with Remote Transport.
- Start the Runtime on the server of PRO1 with Remote Transport.
- Stop the online connection.
- Do the same for PRO2.

4.4.4 Administering projects

You have several possibilities for accessing the data from subprojects, for example:

- Navigation between projects (on page 30)
- Using variables or functions from another project (on page 32)
- Sending recipes to different variables in different projects (on page 33)
- Create archives for use throughout projects (on page 35)
- Creating a joint AML or CEL for different projects (on page 38)

> ⚠ **Attention**
>
> *During configuration, note which modules and functions are executed on which devices - server, standby, or client. You can find a list of the possible configurations here: Behavior of modules in the network (on page 106).*

Navigation between projects

When administering more than one project in an integration project, it is absolutely necessary to ensure that it is possible to switch from a subproject to another project or to the integration project in Runtime. To do this, create a template that is always in the foreground and place navigation buttons there.

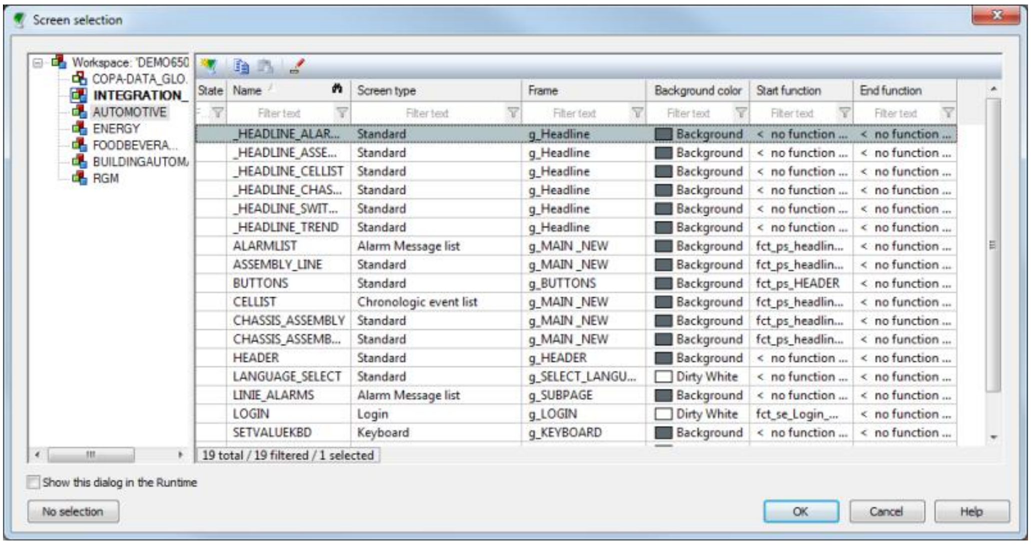
SCREEN SWITCH TO SUBPROJECTS

To switch between 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 frame that offers space for navigation
2. Assign it the `Always in the foreground` property
3. Activate the `Border type` and `Title` properties
4. (this enables the frame to be moved in Runtime)
5. 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: 'DEMO650
COPA-DATA_GLO.
INTEGRATION_
AUTOMOTIVE
ENERGY
FOODBEVERA...
BUILDINGAUTOM.
RGM

State	Name	Screen type	Frame	Background color	Start function	End function
	_HEADLINE_ALAR...	Standard	g_Headline	■ Background	< no function ...	< no function ...
	_HEADLINE_ASSE...	Standard	g_Headline	■ Background	< no function ...	< no function ...
	_HEADLINE_CELLIST	Standard	g_Headline	■ Background	< no function ...	< no function ...
	_HEADLINE_CHAS...	Standard	g_Headline	■ Background	< no function ...	< no function ...
	_HEADLINE_SWIT...	Standard	g_Headline	■ Background	< no function ...	< no function ...
	_HEADLINE_TREND	Standard	g_Headline	■ Background	< no function ...	< no function ...
	ALARMLIST	Alarm Message list	g_MAIN _NEW	■ Background	fct_ps_headlin...	< no function ...
	ASSEMBLY_LINE	Standard	g_MAIN _NEW	■ Background	fct_ps_headlin...	< no function ...
	BUTTONS	Standard	g_BUTTONS	■ Background	fct_ps_HEADER	< no function ...
	CELLIST	Chronologic event list	g_MAIN _NEW	■ Background	fct_ps_headlin...	< no function ...
	CHASSIS_ASSEMBLY	Standard	g_MAIN _NEW	■ Background	fct_ps_headlin...	< no function ...
	CHASSIS_ASSEMB...	Standard	g_MAIN _NEW	■ Background	fct_ps_headlin...	< no function ...
	HEADER	Standard	g_HEADER	■ Background	< no function ...	< no function ...
	LANGUAGE_SELECT	Standard	g_SELECT_LANGU...	□ Dirty White	< no function ...	< no function ...
	LINIE_ALARMS	Alarm Message list	g_SUBPAGE	■ Background	< no function ...	< no function ...
	LOGIN	Login	g_LOGIN	□ Dirty White	fct_se_Login_...	< no function ...
	SETVALUEKBD	Keyboard	g_KEYBOARD	■ Background	< no function ...	< no function ...

19 total / 19 filtered / 1 selected

□ Show this dialog in the Runtime

3. Select **PRO1**.
4. Select the start screen of **PRO1** and close the dialog with **OK**.
5. Repeat the process for **PRO2**.
6. Add two text buttons with the text **PRO1** and **PRO2** to the navigation screen.

7. Link the the two text buttons to 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 project PRO1. However this switching will not work in Runtime!*

Variables and functions

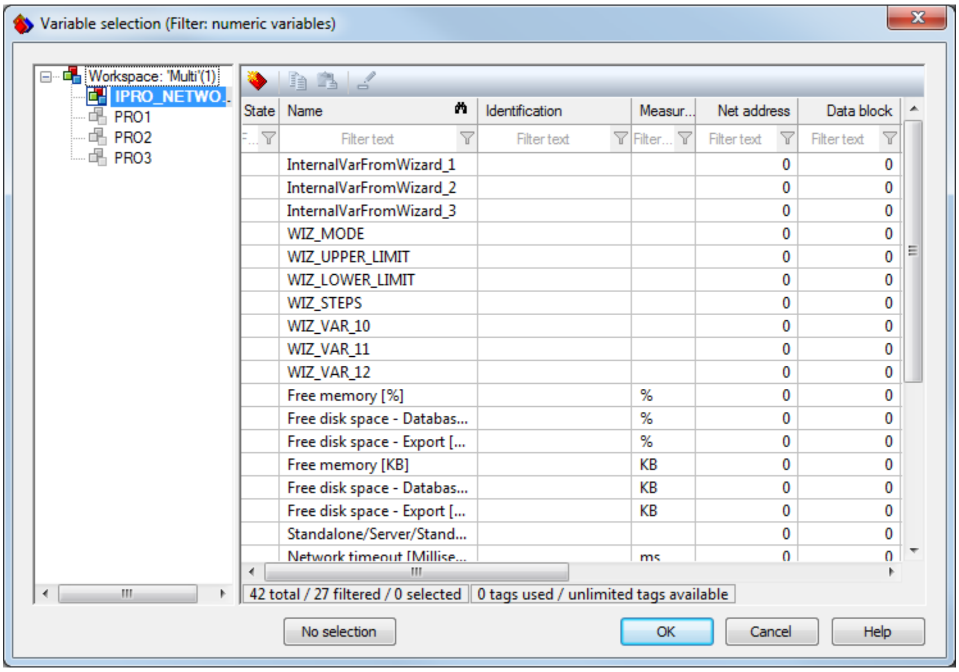
You can access variables and functions of other projects from the same workspace using **`Dynamic elements`**.

VARIABLE EXAMPLE

1. Open the start screen of the I-PRO.

2. Add a new **`numerical value`** dynamic element.

3. Now the variable selection dialog opens.



Variable selection (Filter: numeric variables)

Workspace: 'Multi'(1)
- IPRO_NETWO
- PRO1
- PRO2
- PRO3

State	Name	Identification	Measur...	Net address	Data block
	Filter text	Filter text	Filter...	Filter text	Filter text
	InternalVarFromWizard_1			0	0
	InternalVarFromWizard_2			0	0
	InternalVarFromWizard_3			0	0
	WIZ_MODE			0	0
	WIZ_UPPER_LIMIT			0	0
	WIZ_LOWER_LIMIT			0	0
	WIZ_STEPS			0	0
	WIZ_VAR_10			0	0
	WIZ_VAR_11			0	0
	WIZ_VAR_12			0	0
	Free memory [%]		%	0	0
	Free disk space - Databas...		%	0	0
	Free disk space - Export [...		%	0	0
	Free memory [KB]		KB	0	0
	Free disk space - Databas...		KB	0	0
	Free disk space - Export [...		KB	0	0
	Standalone/Server/Stand...			0	0
	Network timeout [Millise...		ms	0	0

42 total / 27 filtered / 0 selected | 0 tags used / unlimited tags available

4. Here, you can select not just variables from the I-PRO. To select a variable from another project:
    a) Click on a project in the left list area
    b) You are offered the attendant variables
    c) select the variable you want to assign
5. Select a variable from PRO1 or PRO2.

As the I-PRO is a client to the servers PRO1 and PRO2, the connection will work without any problems in Runtime.

The procedure is the same 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 will not work in Runtime however!

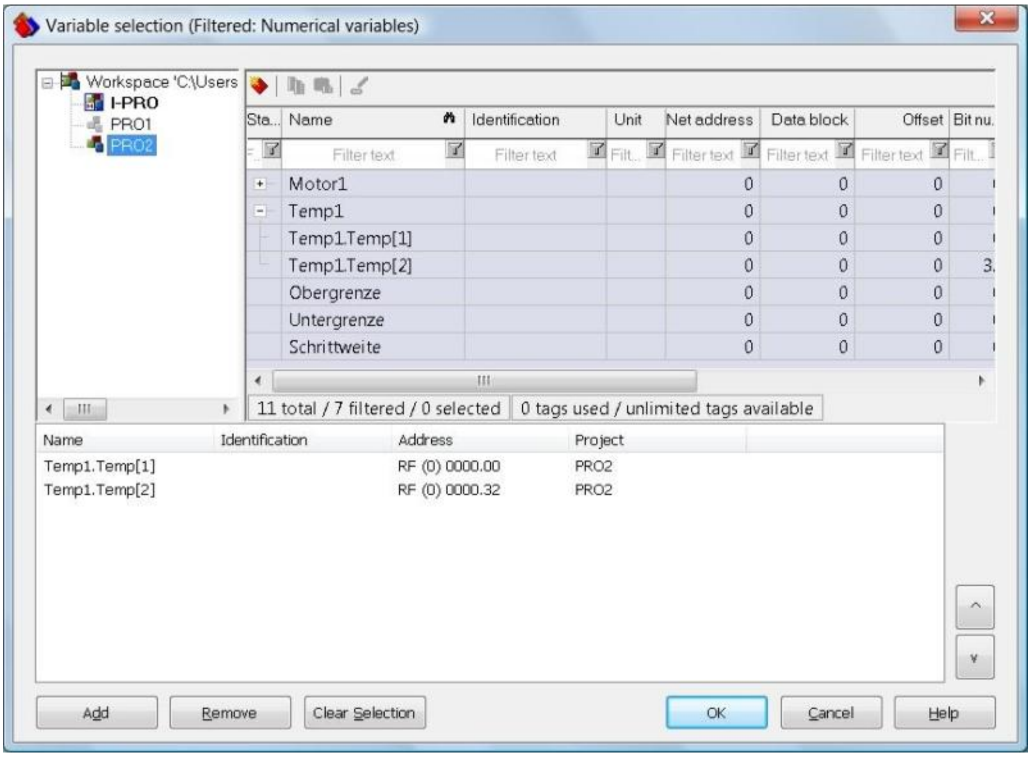
Recipes

You can set values of variables for different projects of the workspace in a recipe.

EXAMPLE OF A RECIPE

1. In the project I-PRO open the branch **`Recipes`**.
2. Under **`Standard Recipes`** create a new recipe named RECIPE1.
3. Open the context menu of RECIPE1 and select **`Add variable`**.

4. The dialog for selecting variables will be opened.



Sta...	Name	Identification	Unit	Net address	Data block	Offset	Bit nu.
+	Motor1			0	0	0	
−	Temp1			0	0	0	
	Temp1.Temp[1]			0	0	0	
	Temp1.Temp[2]			0	0	0	3.
	Obergrenze			0	0	0	
	Untergrenze			0	0	0	
	Schrittweite			0	0	0	

11 total / 7 filtered / 0 selected | 0 tags used / unlimited tags available

Name	Identification	Address	Project
Temp1.Temp[1]		RF (0) 0000.00	PRO2
Temp1.Temp[2]		RF (0) 0000.32	PRO2

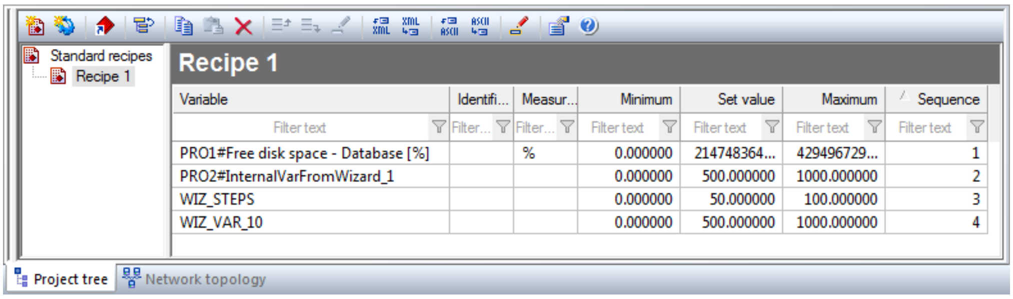
5. Here, you can select not just variables from the I-PRO. To select variables from other projects:

   a) Click on a project in the left list area.

   b) You are offered the attendant variables.

   c) Select variables from PRO1 and PRO2.

   d) In the variable list of the recipe, the project name is written in front of the variable name.



Recipe 1

Variable	Identifi...	Measur...	Minimum	Set value	Maximum	Sequence
PRO1#Free disk space - Database [%]		%	0.000000	214748364...	429496729...	1
PRO2#InternalVarFromWizard_1			0.000000	500.000000	1000.000000	2
WIZ_STEPS			0.000000	50.000000	100.000000	3
WIZ_VAR_10			0.000000	500.000000	1000.000000	4

As the I-PRO is a client to the servers PRO1 and PRO2, the connection will work without any problems in Runtime.

⚠ 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 will not work in Runtime however!*

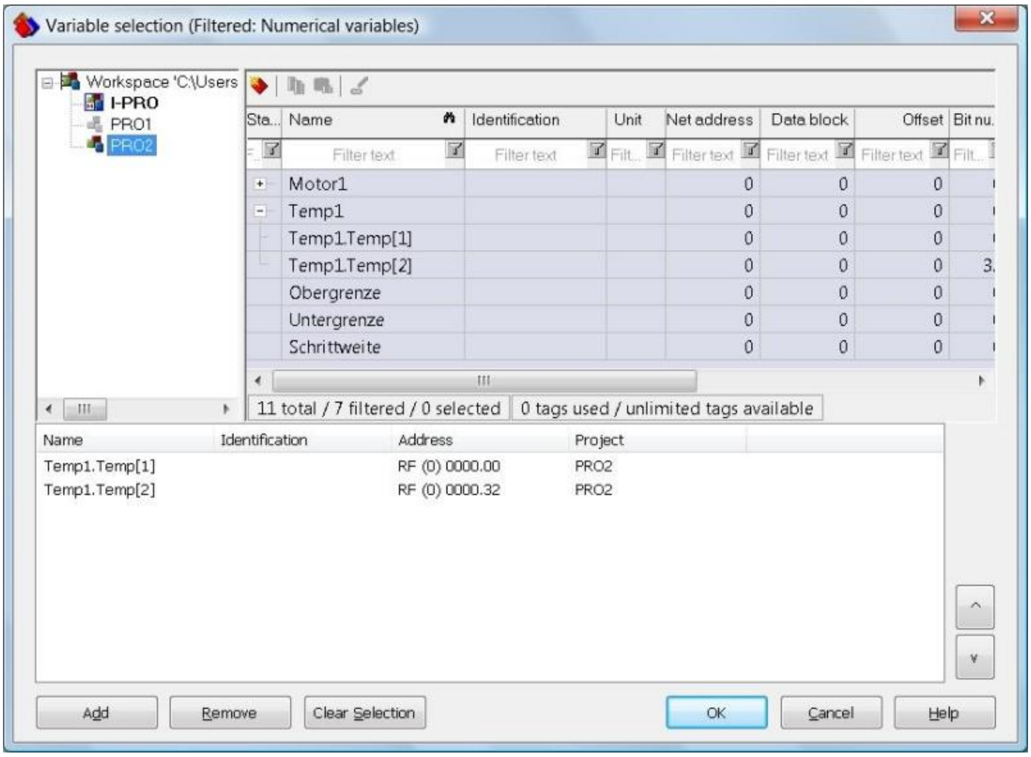
Archives

You can record values of variables from different projects of the workspace 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 I-PRO open the branch **`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 (Filtered: Numerical variables)

Sta...	Name	Identification	Unit	Net address	Data block	Offset	Bit nu.
+	Motor1			0	0	0	
−	Temp1			0	0	0	
	Temp1.Temp[1]			0	0	0	
	Temp1.Temp[2]			0	0	0	3.
	Obergrenze			0	0	0	
	Untergrenze			0	0	0	
	Schrittweite			0	0	0	

11 total / 7 filtered / 0 selected

0 tags used / unlimited tags available

Name	Identification	Address	Project
Temp1.Temp[1]		RF (0) 0000.00	PRO2
Temp1.Temp[2]		RF (0) 0000.32	PRO2

5. Here, you can select not just variables from the I-PRO. To select variables from other projects:

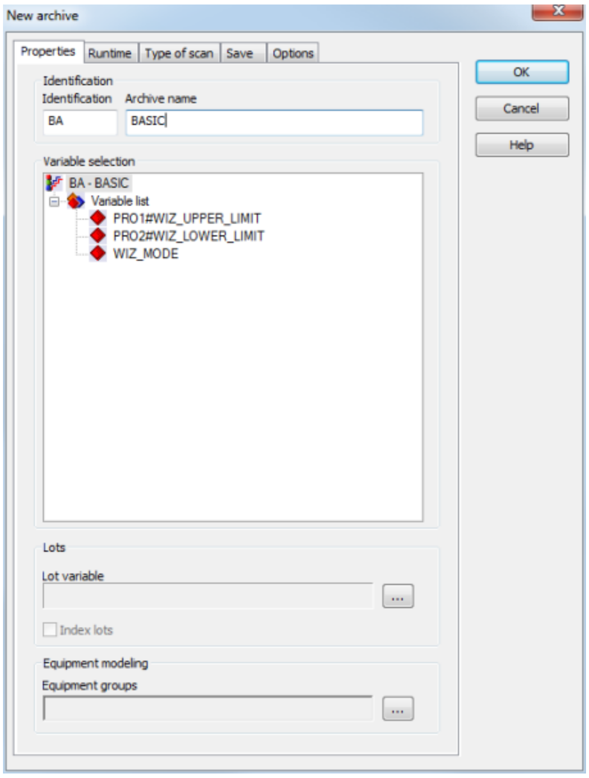
   a) Click on a project in the left list area

   b) You are offered the attendant variables

   c) select the variable you want to assign

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

Properties | Runtime | Type of scan | Save | Options

Identification

Identification: BA

Archive name: BASIC

Variable selection

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

Lots

Lot variable

□ Index lots

Equipment modeling

Equipment groups

As the I-PRO is a client to the servers PRO1 and PRO2, the connection will work without any problems in Runtime.

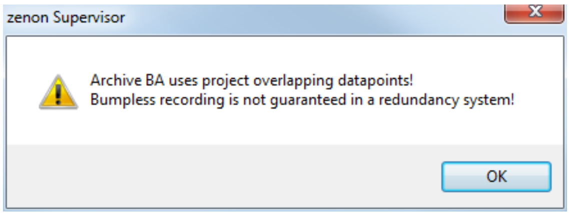
Attention

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 will not work in Runtime however!

REDUNDANCY

After the selection of variables has been concluded, a message box indicates that seamless recording is not guaranteed under all circumstances.



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

> *In this structure, the I-PRO is only a client for the two servers PRO1 and PRO2. So it only gets its data from the two server projects. If one of the two servers fails, the I-PRO no longer receives the corresponding data. So in the archives you would only get alternate values!*

To ensure seamless recording, run all projects that provide data to an archive as redundant.

Alarms and CEL

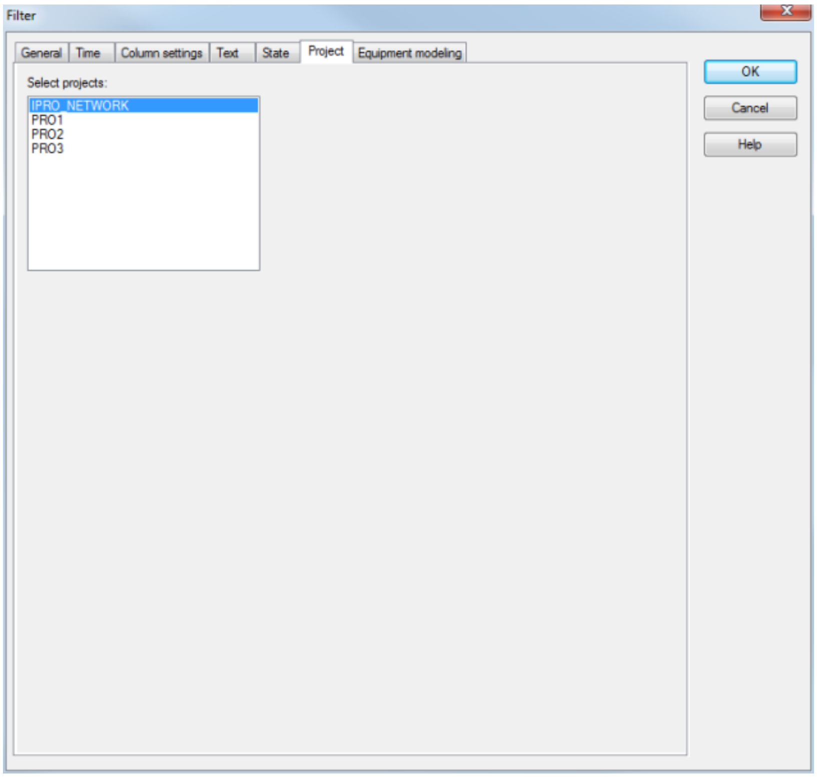
In zenon, you can display system messages and alarms from different projects of one workspace together in a list. These entries then can be filtered, displayed, printed or exported just the data from normal Alarm Message Lists or Chronological Event Lists.

AML example

1. Create a screen of the screen type `Alarm`.
2. Add controll 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 project that is to be displayed in the AML of the I-PRO.

   Multiple selection button `Ctrl` plus a mouse click.)



Filter

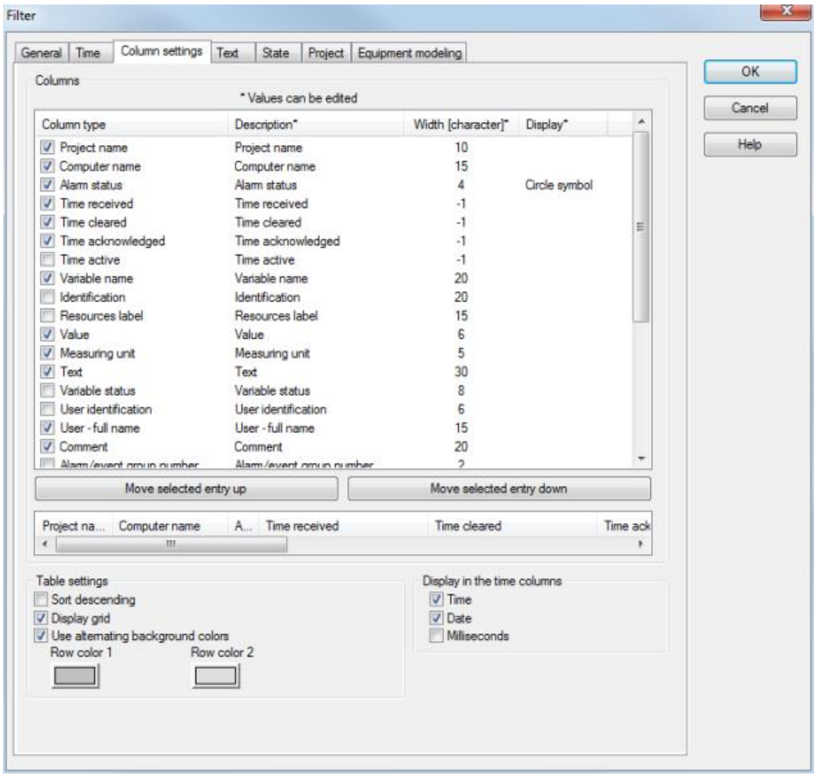
Select projects:

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

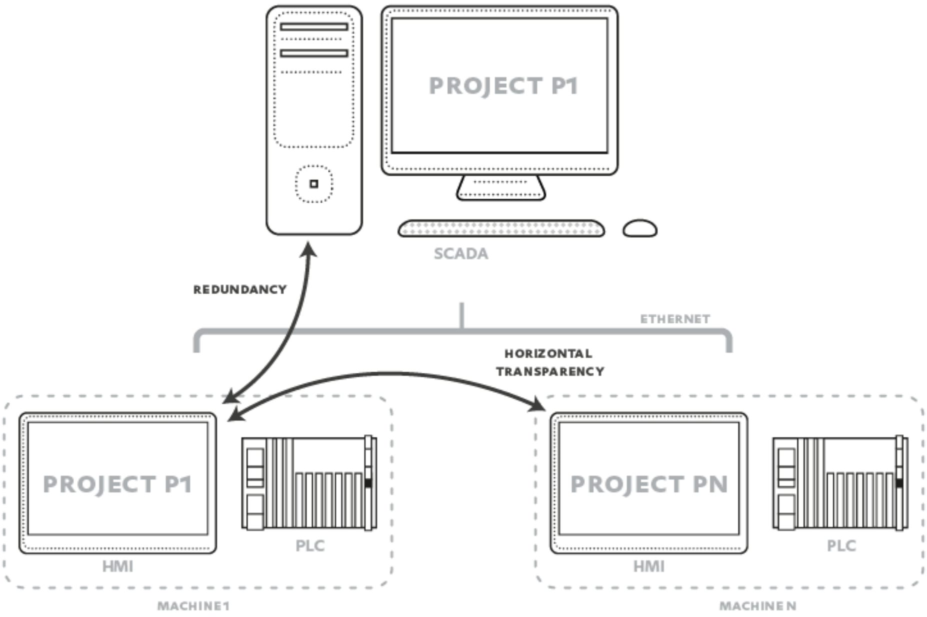
□ Milliseconds

4.5 Horizontal transparency

Horizontal transparency increases the performance of Runtime by:

- Machine operators working together
- Communication of the actual values and target values of the performance indicators

Horizontal transparency is made possible by the multi-project administration (on page 25) in zenon. With this, all projects that are on the same level can be transferred/switched to one computer.



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 the terminal's own project and that of the neighboring terminals on each terminal. This way the entire machine can be monitored and operated from each terminal.

4.6 Optimization of large projects

Large network projects can, with standard settings, place a large load on zenon Server when reloading is carried out at the same time.
Large means:

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

In this case, the behavior on reloading may need to be optimized so that not all clients load data at the same time. You can make this happen with the `RELOADDELAY_SEC 0` entry in `project.ini`. With this,

reloading is delayed by a random value.
To do this:

1. Open **project.ini** in the Project_SQL_Ordner/FILES/zenon/system/ folder
   Tip: Highlight the project in the project manager and press the keys Ctrl+Alt+E; Windows Explorer opens the Project_SQL_directory/FILES/ folder
2. Navigate to the **[NETZ]** section
3. Create the entry **RELOADDELAY_SEC 0**
4. Select a value for the delay for 0

When reloading, a random delay in seconds is set for each client, which is between 0 and the selected value. 0 means no delay.

The selected value has no influence on single user projects, the server or the standby

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

5. 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 (except Windows XP SP2) and for the web client.

If encryption is active, communication between the server, client and web client is in encrypted form; the zenon web server only forwards data packets and is not affected by encryption.

> **Info**
>
> *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.*

5.1 Basics

Encryption for zenon Runtime is available from version 7.0. 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**. Above all, this 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 switched on or off for all servers, clients and web clients in the zenon network. Communication between encrypted and unencrypted systems is not possible. Web servers 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.

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

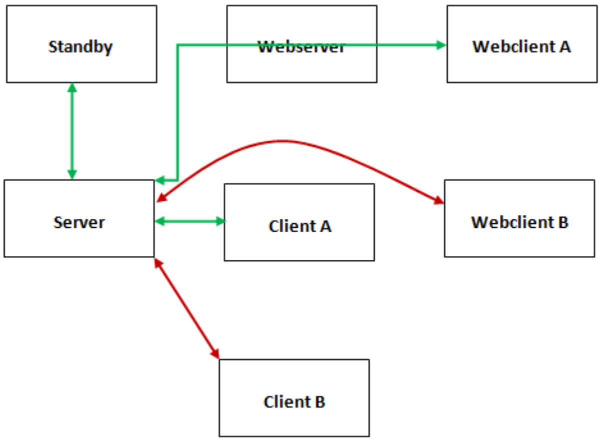
Errors (on page 50) are logged in the Diagnosis Viewer's log file.

EXAMPLE:

The following illustration shows an example of a network with server, standby, two clients, one 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 server using the Startup Tool (on page 46).
- Encryption is also activated on the standby server and client A via Remote Transport (on page 46) when Runtime files are transferred.
- Client B and web client B still communicate without encryption.
- On web client A , encryption is activated by means of `network communication encryption` (on page 48) (zenWebCryptConfig.exe).
- Because the 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 server.
- Client A can log in to the server and exchange data.
- Because client B sends unencrypted messages and these are rejected by the server because encryption is active, client B cannot communicate with the server and is therefore offline.
- Web client A logs on to the server via the web server and can exchange data.

- The unencrypted messages from web client B are forwarded from the web server to the server, but the server rejects these. Web client B cannot communicate with the server and is therefore offline.

As soon as encryption via Remote Transport or the Startup Tool configuration on client B and via **`network communication encryption`** on web client B is activated, these connections can also make connections to the server.

5.2 Activate encryption

Encryption can be activated in different ways:

- Locally via the Startup Tool (on page 46)
- By Remote Transport (on page 46)
- in the web client (on page 48)

HINT:

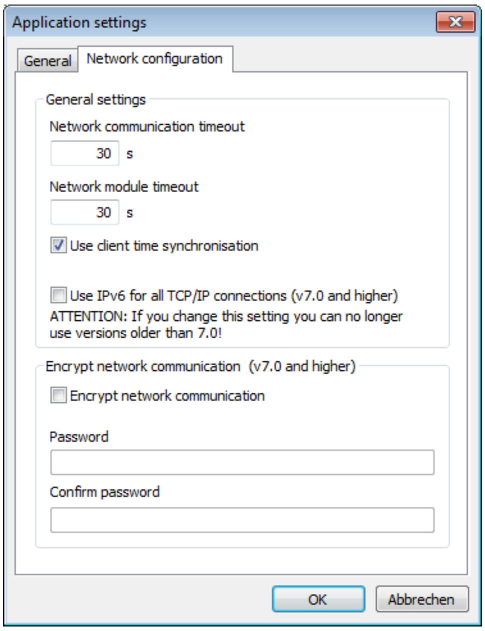
For quick, easy activation of the encryption, it is recommended that the configuration is carried our on a computer using Remote Transport (on page 46). This has the following advantages:

- The configuration be transferred from one device to another once the connection is made with a few clicks:
  - Activating **`Configure remote encryption`**
  - Click on read **`from zenon6.ini`**
  - Click on **`OK`**
- All configurations can be made from one computer.
- Typing mistakes when entering the password are picked up by the password confirmation, but in the event that the same typing mistake was made in both input screens, the transfer of the configuration by Remote Transport ensures that all computers use the same password.

5.2.1 Locally via the Startup Tool

To activate encryption on the local computer:

1. Open the zenon Startup Tool
2. If you operate several zenon versions at the same time, highlight the desired version
3. Click on *Application -> Options*
4. The dialog for the zenon Startup Tools settings is opened
5. Select the **`Encrypt network communication`** tab



General settings

Network communication timeout

30 s

Network module timeout

30 s

☑ Use client time synchronisation

☐ Use IPv6 for all TCP/IP connections (v7.0 and higher)
ATTENTION: If you change this setting you can no longer
use versions older than 7.0!

Encrypt network communication (v7.0 and higher)

☐ Encrypt network communication

Password

Confirm password

6. Check the `Encrypt network communication` checkbox
7. Enter the password and verify this
8. confirm the dialog by clicking **OK**

5.2.2 By Remote Transport

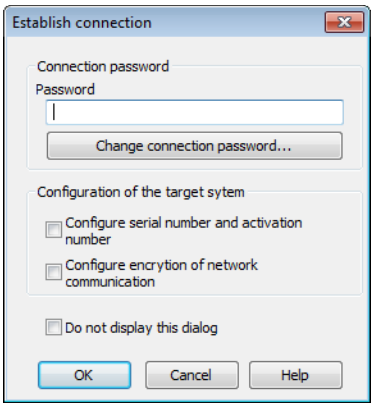
Encryption can be be activated on remote computers using Remote Transport. However, this is only possible if the remote connection is protected with a password.

To activate encryption using a remote connection:

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

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

2. 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

□ Configure encrytion of network communication

□ Do not display this dialog

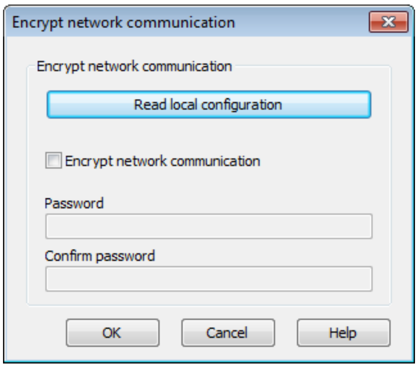
OK Cancel Help

3. Enter the connection password or create one, if none has been set

4. Activate the `Configure encryption of network communication` checkbox

5. click on OK

6. 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

7. Activate the `Encrypt network communication` checkbox

8. Give it a password (to find out the criteria for this, see "**network encryption password (on page 48) section)**

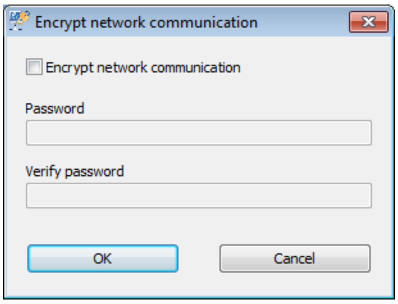
9. Confirm the dialog by clicking on the OK button.

5.2.3 in the web client

Network communication can also be encrypted for access with the zenon Web Client. To access an encrypted network with a web client:

1. In the Start menu, launch the link `\Programme\COPA-DATA\Network communication encryption`
   or the file **`zenWebCryptConfig.exe`** in the zenon program file.

2. The dialog for configuration is opened


Encrypt network communication
□ Encrypt network communication
Password
Verify password
OK
Cancel

3. Check the checkbox in front of `Encrypt network communication`

4. Enter the password and confirm this

5. Close the dialog by clicking on the OK button.

5.3 Password network encryption

The following applies to the password to encrypt 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
- Characters that are not permitted:
  - space
  - Return key
- Composition: A password must contain at least 1 digit and 1 character

5.4 Checklist for errors

In the event of errors, check:

- Do all computers have access to the network and does the name resolution work?
- Was the Network active property activated for the project in the Editor?
- For projects with encryption, Is zenon Runtime Version 7.00 SP0 or later running on all computers?
- 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.)
- Are the required 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 50) is written to the log file when Runtime is started.
- 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`

Errors (on page 50) are logged in the Diagnosis Viewer's log file.

5.5 Error messages

Errors are displayed either in the output window of the zenon Editor or in pop-ups and/or logged in the log files of the Diagnosis Viewer.

NO CONNECTION

If a client was configured with an incorrect encryption password (not the same as that on the server), then this is evident from the following events:

- The client is offline, although the server can be reached by pinging.
- The server writes error messages into the log file:

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

or:

NET Error During Decryption [error number]

NOTICE BOXES AND ERROR MESSAGES

Errors in encryption are indicated by notice boxes (on page 51) (pop-ups) and entries to log files (on page 54) or in the zenon output window (on page 54).

5.5.1 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 `network communication encryption` for the configuration of the web clients. These messages are always in English.

Error message	Description
The network password has to be entered in both textboxes!	When configuring the encryption, the user has left one of the two input fields (**Password** or **Password confirmation**) empty.
The entered network password and the retyped network password are different!	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 fulfill the password conventions! **Password conventions:** **Minimum length = 8 symbols** **Maximum length = 20 symbols** **At least one character of the Latin charset** **At least one number** **The blank character is not allowed**	The password entered does not fulfill the password conventions. The password conventions are also 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 files zenon6.ini and startup.ini are invalid. **Please enter a new configuration and register any item with the startup tool to update both files with valid configurations.**	When opening the **Encrypt network communication** tab in the zenon Startup Tool, it was established that neither **zenon6.ini** nor **startup.ini** have valid configurations for network encryption. A new configuration must be entered.
The network encryption configuration in the file zenon6.ini is invalid. **As the configuration in the file startup.ini is valid, the file zenon6.ini will be updated with a valid configuration when any item is registered with the startup tool.**	A check when opening the **Encrypt network communication** tab in the zenon Startup Tool resulted in: The encryption configuration in **zenon6.ini** is invalid, but the encryption configuration in **startup.ini** is valid. The invalid configuration in **zenon6.ini** will be overwritten by the valid configuration from startup.ini next time an entry is registered.
The network encryption configuration in the file startup.ini is invalid. **The file zenon6.ini currently contains a valid configuration. Upon registration of any item with the startup tool the file zenon6.ini will be updated with the invalid configuration from the file startup.ini. Please enter a new configuration to**	A check when opening the **Encrypt network communication** tab in the zenon Startup Tool resulted in: The encryption configuration in **zenon6.ini** is valid, but the encryption configuration in **startup.ini** is invalid. The configuration in **zenon6.ini** will also become invalid next time an entry is registered. A new configuration must be entered in order to ensure that both INI files contain valid configurations.

`ensure both files contain valid configurations after the registration of an item.`	
`The network encryption password in zenon6.ini is invalid`	The password read off from **`zenon6.ini`** is invalid.
`The password for network encryption is invalid and must be reentered!`	Message when Runtime starts if the password cannot be verified.

REMOTE TRANSPORT

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

Error message	Description
`To configure the network encryption, the Remote Transport connection must be password protected.`	An attempt was made to configure remote encryption without the Remote Transport connection by means of a password.
`The network password must be entered into both text boxes.`	When configuring the encryption, the user has left one of the two input fields (**`Password`** or **`Password confirmation`**) empty.
`The password confirmation does not correspond to 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 conventions. The password conventions are displayed in the error message.

`An error occurred when encrypting the network password.`	An error occurred when encrypting the network 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.`	The password stored in zenon6.ini cannot be decrypted.
`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 set again.

5.5.2 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.
`*** The configuration of network encryption on the remote device was updated:`	ACCENT	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 encryption configuration was successfully saved on the server.`	TEXT	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.

5.5.3 Error messages in log files

ENCRYPTION

Errors in encrypted network traffic are documented in log entries. The `Error IDs` of the error messages in the following table 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 [Error-ID]	ERRORS	The processing of a hash value was unsuccessful.
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 Calculation [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.
`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 Password Not Properly Formatted`	ERRORS	The hex dump of the encrypted password is in an invalid format.
`NET Error During Password Decryption [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 Disabled By Remote Configuration`	DEBUG	zenSysSrv reports that encryption of network traffic on the computer was activated by the Remote Transport configuration.
`Network Cryptography Remote`	ERRORS	A configuration sent by Remote Transport for network

Configuration Error		encryption is erroneous.
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 Password Decryption: 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 cancelled 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 Password Decryption 0x80090005**" error, where instead of "**NET Error During Password Decryption", The Password is Invalid!**" is displayed. 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 error message "**NET Error Cryptography Not Initialized!**" is written to the log file.

6. Remote transport in the network

With remote transport, files are transferred to other computers. You can find the basics and details about remote transport in the Remote Transport manual.

You control the actions with the remote transport toolbar. You can find the settings for connection and files to be transferred in the project settings. The paths for the development computer must be assigned to the paths of the Runtime computer. It is therefore also possible to update Runtime files during online operation. The updated Runtime files are loaded into the running project by means of the **Reload (on page 110)** function.

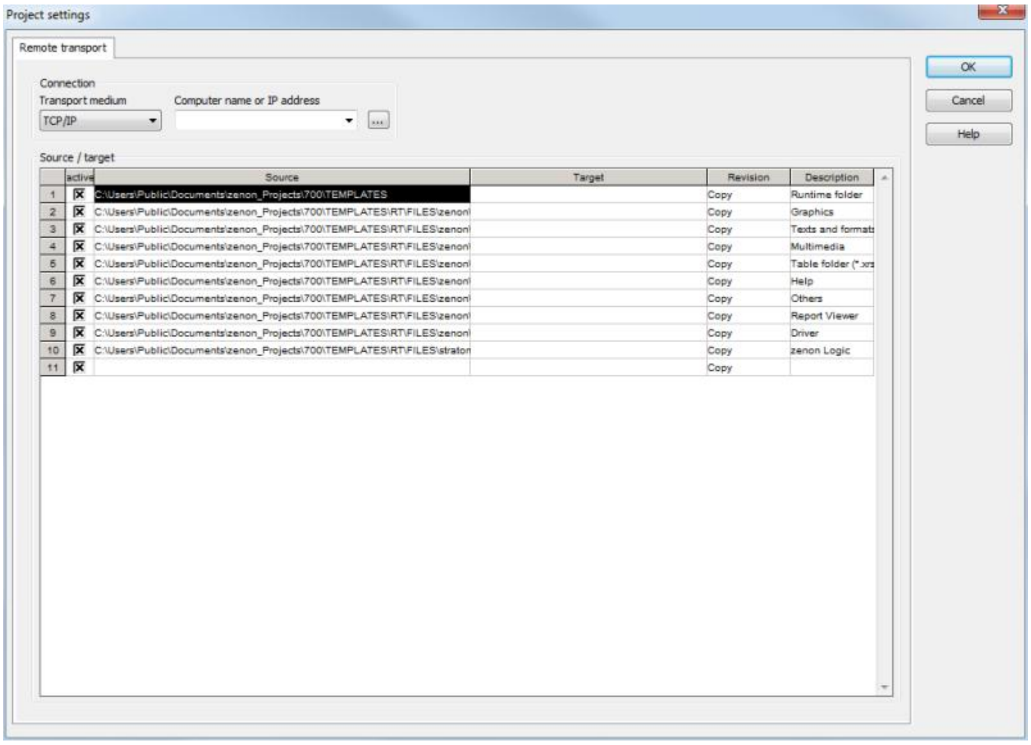
Note: For connection to CE devices, see the Windows CE section in the Remote Transport handbook.

CREATING A CONNECTION AND SELECTING A SAVE LOCATION

1. Open the `General` node in Project Properties.

2. Click on the `Remote transport` property.

3. The **Remote Transport** dialog is opened.



Connection

Transport medium: TCP/IP

Computer name or IP address:

Source / target

	active	Source	Target	Revision	Description
1	☒	C:\Users\Public\Documents\zenon_Projects\700\TEMPLATES		Copy	Runtime folder
2	☒	C:\Users\Public\Documents\zenon_Projects\700\TEMPLATES\RT\FILES\zenon\		Copy	Graphics
3	☒	C:\Users\Public\Documents\zenon_Projects\700\TEMPLATES\RT\FILES\zenon\		Copy	Texts and formats
4	☒	C:\Users\Public\Documents\zenon_Projects\700\TEMPLATES\RT\FILES\zenon\		Copy	Multimedia
5	☒	C:\Users\Public\Documents\zenon_Projects\700\TEMPLATES\RT\FILES\zenon\		Copy	Table folder (*.xrs
6	☒	C:\Users\Public\Documents\zenon_Projects\700\TEMPLATES\RT\FILES\zenon\		Copy	Help
7	☒	C:\Users\Public\Documents\zenon_Projects\700\TEMPLATES\RT\FILES\zenon\		Copy	Others
8	☒	C:\Users\Public\Documents\zenon_Projects\700\TEMPLATES\RT\FILES\zenon\		Copy	Report Viewer
9	☒	C:\Users\Public\Documents\zenon_Projects\700\TEMPLATES\RT\FILES\zenon\		Copy	Driver
10	☒	C:\Users\Public\Documents\zenon_Projects\700\TEMPLATES\RT\FILES\straton		Copy	zenon Logic
11	☒			Copy	

Parameters	Description
`Transport medium`	
`Serial`	Transfer via a serial connection, e.g. to a CE Terminal.
▸ `Port`	Selecting the COM ports for the serial connection.
`TCP/IP`	Transfer via TCP/IP in a network or via a modem.
▸ `Computer name or IP address`	The computer name or the TCP/IP address is entered as target: ▸ Enter the computer name manually or via clicking button `...` ▸ Enter the IP address manually The IP address must have conform the defined IP version (on page 8) (IPv4 or IPv6). **Note:** At the connection with name you can also use port numbers. For example: `Runtime1;PORT=1105`

Parameters	Description
`Source/Target`	List of connections. In the first line you can define a target for the top most folder of the structure. Right click in the cell in order to open the context menu for selection. **Attention:** This path must be permanently available on the target system. This means no integrated network device and no removable data device.
`active`	Defines files which should be transferred optionally. Project base path cannot be deselected.
`Source`	Folder for files which should be transferred.
`Target`	Target folder. Target for the top most folder and new entry can be defined. Sub-folders cannot be changed. This makes sure that all files are found on the target system in the Runtime. **Hint:** The default folder is the Runtime folder defined in the project properties. If the target folder is entered manually, take care that it ends with the project name. This is important for the multi-user administration. For example: C:\Users\Public\Documents\zenon_Projects\MY_PROJECT
`Editing`	Type of transfer. Can be defined freely for the top most fodler and new entries. Right click to open the drop-down list: ▸ **`Copy`**: copies files ▸ **`Copy and register`**: copies files and registerd them in the system. Helpful for ActiveX elements and for fonts (ttf files). ▸ **`Copy and execute`**: copies files and then executes them
`Description`	Optional text input for new entries for describing the files which should be transferred.

⚠ Attention

For redundant projects: If drivers are used which need a configuration file, you must deactivate entry `\zenon\custom\drivers` .

Parameters	Description

> ⚠ **Attention**
>
> *Note the effects of settings if the project is used in the network.*
>
> ***For example:***
>
> *Transport of graphics is deactivated, because these are already on the server. Then a client is started. The client finds out that the graphics should not be transported and thus does not transport them to itself. No graphics are then displayed on the client.*
>
> *Or vice-versa, files are transferred to the client and the files there are overwritten.*

FILES FOR TRANSFER

With Remote Transport, the following files are transferred to the target system: In doing so, all files are always transferred to the folder:

1. Standard

   All files that are in the project's Runtime directory.
   These files determine the appearance and behavior of the project and are transferred as standard:

   a) all screen files (screen name.zpp)

   b) amlcel.cmp

   c) archiv.cmp

   d) fpm.cmp

   e) functions.cmp

   f) project.cmp

   g) projekt.mdb (not CE)

   h) remas.cmp

   i) rezepturen.cmp

   j) scripts.cmp

   k) templates.cmp

   l) variables.cmp

   m) zuweisung.cmp

   n) Plus: project.ini and projekt.vba, which are always in the project directory

> **Info**
>
> *Files with the following suffixes are not transferred by default:*
>
> - .hot
> - .ho
> - .ret
> - .re

1. 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 subdirectories of the project directory:

a) `\zenon\custom\graphics`: for graphics

b) `\zenon\custom\lists`: for language tables

c) `\zenon\custom\media`: for all media files

d) `\zenon\custom\reports`: For Report Generator tables

e) `\zenon\custom\help`: For help files

f) `\zenon\custom\additional`: For additional files

g) `\zenon\custom\drivers:` for drivers

h) `\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.

2. Setting the start project

For Runtime, the start project must always be entered in `zenon6.ini`. To do this, click on the **`Set remote Runtime start project`** in the Remote Transport toolbar. In doing so, the following entries are set:

```
[PATH]

VBF30=project path

[DEFAULT]

DEFANWENDUNG30=project name
```

GLOBAL PROJECT

If there is a global project in the workspace, it will automatically be listed for the transport. No additional settings have to be entered. Always all files necessary for the global project will be transported.

REMOTE CONTROL IN THE NETWORK

In the context menu select projects *-> Remote Transport -> <Command>*:

Parameters	Description
`Establishing a connection`	With the service zenSysSvr a connection to the target device is established. On both devices the service zenSysSvr.exe or on the CE device the service `SysSrvCE.exe` has to be started. The versions have to be identical.
`Transport changed Runtime files`	If this setting is selected, the Runtime changed since the last transport are transported. If no Runtime files exist on the target device, all Runtime files are transported.
`Transport all Runtime files`	If this entry is activated, all Runtime files are transported. The only exception are files like e.g. standard recipes or Message Control, which are defined in the setting: 'RT changeable data'.
`Read all Runtime files`	If this setting is selected, all Runtime files of the modules Recipes, Recipegroup Manager, Message Control , User administration and Production and Facility Scheduler (PFS) are transported from the target system to the local Runtime directory. Then the changes in the Runtime can be read to the Editor with the option 'Import Runtime files'.
`Define project as start project`	The selected project is defined as the start project. This project is loaded on each Runtime start.
`Start Runtime`	Starts the Runtime.
`Stopping the Runtime`	The Runtime is closed.
`Reload project`	The project is reloaded. Changes Runtime files are read.
`Restart the operating system`	The device with which the connection exists is restarted after a confirmation message. If necessary, it must be logged on to the operating system again. Under Windows CE this option is not supported.
`Get system status`	It is checked, if the Runtime is running on the target system and with which status. Also the installed zenon version is checked. The following are determined: - Computer name - Operating system - Runtime active/not active - Start project - Real memory - Drives

	- Remote serial number - Remote activation number - Remote zenon version
Change password and display licensing.	Opens dialog for connection establishing. Enables: - Changing of the password - Display and change of the licensing - Configuration of the encryption
Start remote desktop connection	Starts a connection to operate the target computer using Remote Desktop.

Batch mode

Several computers can be addressed at the same time in batch mode. For that

1. Create a file named **hosts.txt**.A-Z
2. Enter the IP address of each computer on a separate line

Example:

HOST=192.168.0.24;
HOST=192.168.0.15;

3. Enter the target address of the remote transport: `HOST=&hosts.txt;`
4. Place **hosts.txt** in the `\zenon\custom\additional` directory.

STRONG ENCRYPTION

You can find information on strong encryption in the network via remote transport in the Activating encryption (on page 45)/by means of remote transport (on page 46) chapter.

7. zenon at the terminal server

The zenon Runtime can also be used together with a terminal server solution. The Editor cannot run on a terminal server.

> **Info**
>
> *Keep in mind that the name of the terminal client is resolved. If you are using a firewall, make sure that corresponding ports are enabled.*

Terminal servers are offered by several manufactures. All tests with zenon were carried out using the Windows terminal server.

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

7.1 Mode of operation of terminal servers

Terminal servers allow to start several separated shell instances (desktop) on one computer (the host computer).

If a terminal client connects to the server, a separate GUI is allocated to it. Only a small program runs on the client, which displays the graphical data sent from the server. All programs started on the client run on the terminal server. Only the screen information (graphical data) is sent to the client via the network.

7.2 Advantages and disadvantages

ADVANTAGES

- Only one computer (the terminal server) has to be maintained.
- Clients do not have to be very performant (Thin Clients).
- Clients can have different operating systems (Windows 7, Windows CE, Linux, Unix, etc.).

- High data security at the client.

DISADVANTAGES

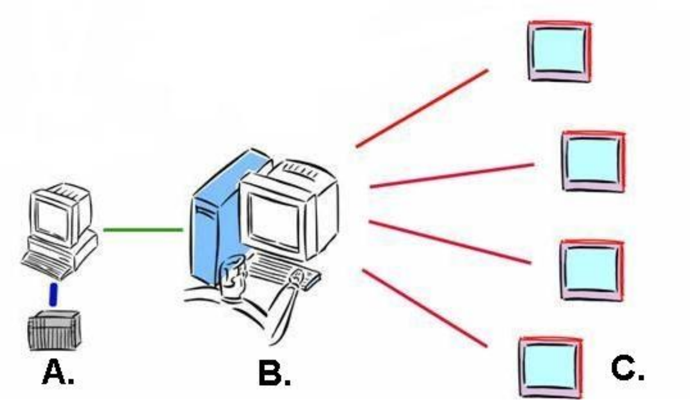
- All started programs of all instances run on one computer (the terminal server). The computer:
  - must have sufficient computing power for all started programs.
  - must have sufficient RAM for all started programs.
- All interfaces have to be shared, e.g. network adapters, COM ports, parallel ports.
- The network load gets accordingly high, as the data of the programs as well as the graphical data for the clients have to be sent via the network.
- All started programs use the same file system and the same files. For zenon this means that:
  - each client increases the memory need
  - the screen resolution is defined by the client started first. If screens in different resolutions should be used, you can manage this via an entry in zenon6.ini (SERIALZE=0) at the terminal server. With this all screens are calculated newly for the client which increases the needed performance for the terminal server.

7.3 Operation zenon at the terminal server

Only one zenon client can run on the terminal server. zenon as server or single-user system is not possible.

SCHEMATIC DISPLAY

This is how the topology of a termnial server network with zenon could look:



Parameters	Description
A.	zenon Runtime Server
B.	Terminal Server and n-fold Runtime Client
C.	n Terminal Clients (only graphical display)

7.4 Required settings

As the zenon Runtime is started several times as client at the terminal server, you must adjust several settings.

GENERAL

1. INI entry

In the zenon6.ini the following entry has to be added on the terminal server. On the Runtime server no settings are needed.

```
[TERMINAL]
CLIENT=1
```

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

0: The Runtime can only be started once. Operation on the terminal server is not possible. (Default)

2. Automatic adjustment of the screen resolution

Per default the first client at the terminal server defines the screen resolution. You can adjust this with the following entry at terminal server in zenon6.in:

[TERMINAL]

SERIALIZE=0

3. Up to 6.21
   The network service (zennetsrv.exe) has to be registered as standard COM server and not as a service. For this the program has to be started with the option -regsrv from the command line. for example: C:/Program Files/COPA-DATA/zenon700/zenNetSrv.exe -regsrv

Comment: From 6.21 on this is the default registration and you do not have to set this manually.

4. Transfer

The transport service (zensyssrv.exe) must be registered and started as a Windows service, not as a standard EXE file. For this:

a) the program has to be started with the option -service from the command line.

   For example: C:/Program Files/COPA-DATA/zenon700/zenSysSrv.exe -service

b) then the Windows service manager is started. The service will be started automatically during every computer restart.

Please consider that the startup tool and also the setup program always register the transport service as a standard EXE. Therefore the transport service must be register newly as a Windows service after every execution of the startup tool and after every reinstallation.

5. Runtime folder

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

6. Rename zenprocess.exe

If a user terminates the Runtime during terminal operation, `zenprocess.exe` also terminates the Runtimes of all other users. In order to prevent this, you must rename file

zenprocess.exe in terminal operation. **zenprocess.exe** is then no longer started. You can find the file in the zenon installation directory.

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 file **zenon6.ini** with a text editor
2. go to area **[TERMINAL]**
3. edit or create entry: CLIENT_NO_FILE_ALIGN=
4. 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 file **reloadindicator.tmp** in the directory which contains file **project.ini** of the program . The session clients at the terminal server check every 10 seconds whether this file is available. Does it exist and is its file time stamp newer that 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

SERIALIZE=0
```

7.5 Remote Desktop versus Terminal Server

Terminal servers distinguish from remote desktop programs or the zenon Remote Desktop especially by the displayed information:

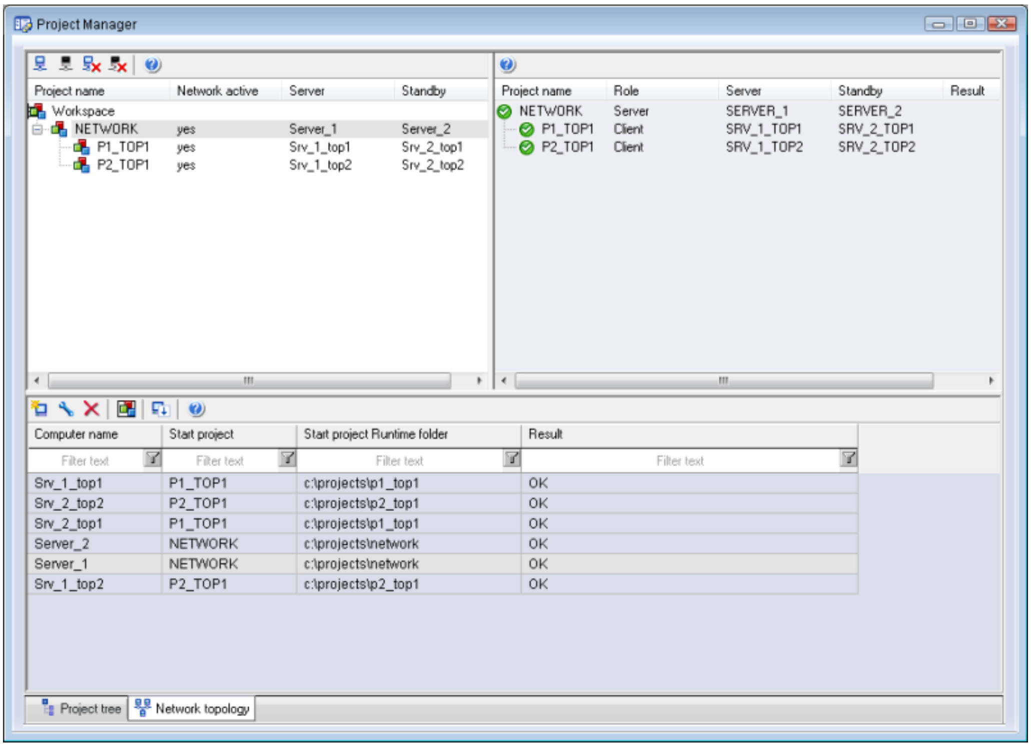
Remote Desktop	Terminal server
All connected stations always see one and the same desktop. If e.g. one user starts a program, all see the same program, the same mouse cursor, the same keyboard input, etc.	Each connected station has its own desktop - an own instance. Only it sees, what happens there.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

8. Administering and checking network topology

The network topology is displayed in a separate project manager tab. It consists of three areas:

- Topology tree (on page 72)(upper left): shows active projects; the global project is not displayed
- Result tree (on page 73)(upper right): only the result is displayed; represents the topology tree of a selected computer

- Computer list (on page 74)(bottom): List display and configuration of computers in the network



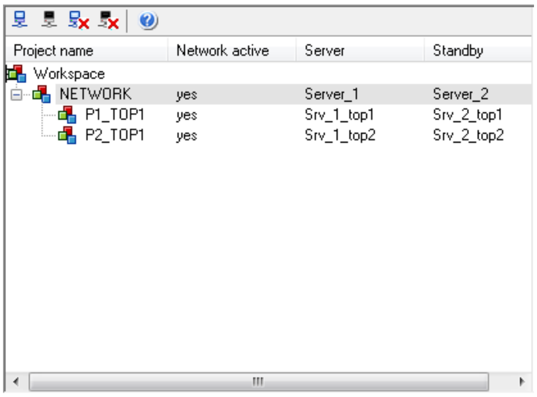
Project name	Network active	Server	Standby
Workspace			
NETWORK	yes	Server_1	Server_2
P1_TOP1	yes	Srv_1_top1	Srv_2_top1
P2_TOP1	yes	Srv_1_top2	Srv_2_top2

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

8.1 Topology tree

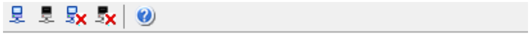
The topology tree displays active projects in topological form. The following are displayed:



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

Parameters	Description
Project name	Is defined in the project tree tab and cannot be changed here.
network active	Shows if the network is active for this project. The setting can be changed via the `Network active` property.
Server	Shows the server defined for this setting. The setting can be changed via the context menu, the symbol in the toolbar or the `Server` property.
Standby	Shows the standby server defined for this setting. The setting can be changed via the context menu, the symbol in the toolbar or the `Standby` property.

TOPOLOGY TREE TOOLBAR AND CONTEXT MENU



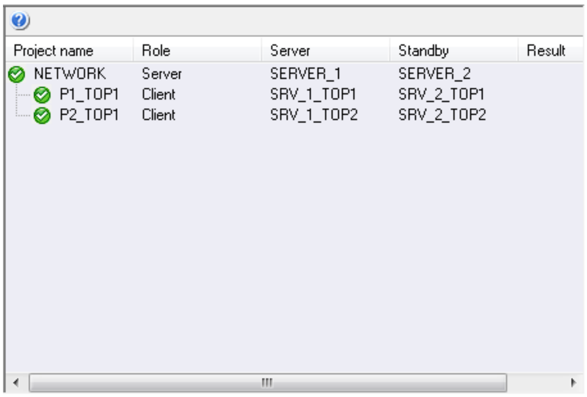
Property	Description
Set computer as server	Defines the computer highlighted in the computer list (on page 74) as the server for the project highlighted in the topology tree.
Set computer as standby	Defines the computer highlighted in the computer list (on page 74) as the standby server for the project highlighted in the topology tree.
Delete server	Deletes the server defined for the highlighted project.
Delete standby	Deletes the standby server defined for the highlighted project.
Help	Opens online help.

8.2 Result tree

The result tree represents the project tree of the computer selected in the computer list (on page 74) 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

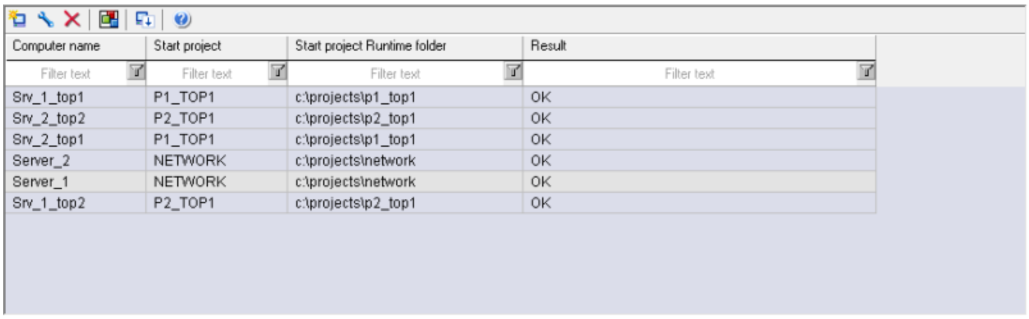


Project name	Role	Server	Standby	Result
✓ NETWORK	Server	SERVER_1	SERVER_2	
✓ P1_TOP1	Client	SRV_1_TOP1	SRV_2_TOP1	
✓ P2_TOP1	Client	SRV_1_TOP2	SRV_2_TOP2	

Parameters	Description
Project name	Projects that are assigned to the selected computer.
Role	Role of the computer: Server Standby Client
Server	Name of the computer that acts as a server to Runtime.
Standby	Name of the computer that acts as a standby server to Runtime.
Result of test	Shows detailed error messages (on page 78) for topology test.

8.3 Computer list

The computer list displays all network devices and allows them to be configured. The list relates to the workspace and is saved each time the workspace is saved.



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

Parameters	Description
Computer name	Name of the computer. Can be changed by - Click in the cell: Clicking on ... opens a drop-down list of the computers currently available in the network. - Edit computer entry in the context menu or the toolbar - `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. - Set start project entry in the context menu or the toolbar Sets the project selected in the topology tree (on page 72) 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:\Projekte\Top = location where start project is stored. Subprojects are stored in C:\\Projects . **Hint:** Use the project name as folder name to automatically create the same structure as on the engineering computer. The `Start project Runtime folder` can be changed by: - double clicking on the computer: Opens computer configuration dialog (on page 77). - 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 found - for details, see detail view: One or more projects have an error. - Serious error found - for details, see detail view: A project has a serious

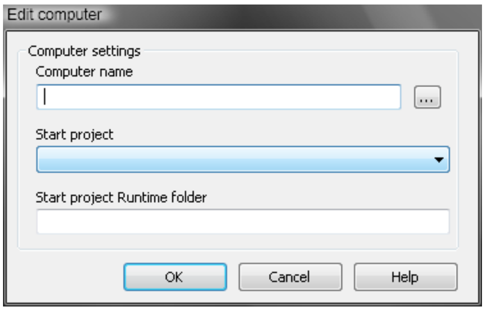
	error. Serious errors halt further testing. - Not tested, because there is a serious error in the structure: The computer was not fully tested, because the test was ended due to a serious error. Detailed error messages (on page 78) are displayed in the result tree.

COMPUTER LIST TOOLBAR AND CONTEXT MENU-

Entry	Description
Add computer...	Opens the Configure computer dialog (on page 77) in the network.
Edit computer...	Opens the dialog to configure the computer (on page 77) in the network for these computers.
Delete computer	Deletes computer from the topology after requesting confirmation. Note: Note that deleting a server or standby server leads to serious errors in the topology.
Setting the start project	Sets the project selected in the topology tree (on page 72) as the start project.
Copy Runtime files from all projects on the computer	Copies all projects valid for the selected computer to the target computer. The result is displayed in an information window.
Help	Opens online help.

8.3.1 Computer network configuration dialog

The following data is required to configure a computer in the network:



Edit computer

Computer settings

Computer name

Start project

Start project Runtime folder

Parameters	Description
Computer name	Clicking on ... opens a drop-down list of the computers currently available in the network.
Start project	Select from drop-down list.
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 to automatically create the same structure as on the engineering computer. For example: Project name = I-Project at `Start project Runtime folder` enter: `C:\Projects\I-Project` The subprojects in relation to this 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 level in the editor. 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: The integration project has the following set up as a Runtime folder: `C:\Workspace\Projects\I-Project`. The subproject has the following set up as a Runtime folder: `C:\Subproject`. The start project Runtime folder is set to `C:\Project`. The subproject cannot be transferred, because the relative folder would be `..\..\Project`. This does not work, because the Runtime folder for the subproject would be below `C:\`. Workaround: 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.

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 server defined?
- Were different computers defined for standby and server?
- Can the client achieve a server/standby?
- Can the server reach its clients?
- Can the standby reach its clients?
- Is the server available for a project in the topology?
- Is the standby available for a project in the topology?
- Is a computer included more than once in the path from client to server?

NOT TESTED:

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

CLIENT TO SERVER

- Does the client reach its server via the server's chain?
- Was a computer that routes switched to its standby?

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

ERROR MESSAGES

Errors that are recognized during the topology test are displayed in the result tree (on page 73) 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!	The server and standby server must be different computers.	Define different computers as server and standby server.
No computer is registered as a 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)! This is however necessary, because it is accessed in in higher hierarchy levels.	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.
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 tested, because there is a serious error in the topology	No test was carried out due to a serious error.	Rectify serious error.
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	Adapt topology or start project for the computer or deactivate the `Routing active` property.

	computer is a server or standby. - > identifies the following computer as a server. - "+" labels the following computer as a standby. e.g.: Computer1+Computer2>Computer3	
`There is circular access from the server to the 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) does not 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 client.)	Adapt topology or start project for the computer or deactivate the `Routing active` property.

9. Redundancy

Redundancy ensures that processes are not interrupted even in the event of a server failure and that no data is lost. For redundant systems, a distinction is made between:

- **`Software redundancy:`** Only the server writes to the PLC. The standby has read access only.
- **`Hardware redundancy:`** PLCs are also designed as redundant. Standby writes to standby PLC.

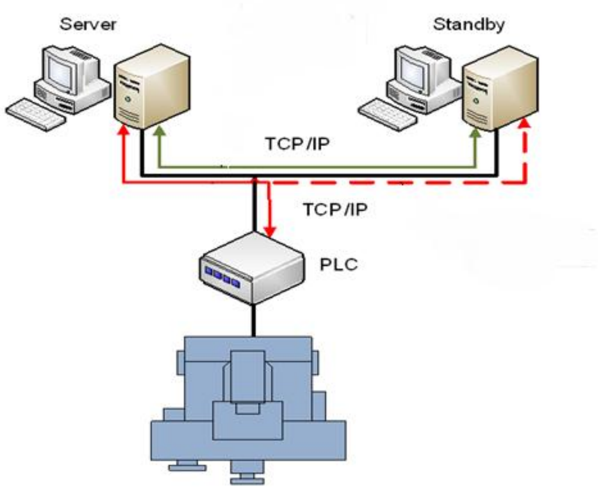
You can find the selection of this in the configuration of the standby server (on page 87).

> **Info**
>
> *Redundancy is not available under zenon Operator. Redundancy settings are ignored.*

SOFTWARE REDUNDANCY

Software redundancy consists of: A PLC and two redundant computers (server and standby server).

In operation, the server communicates both ways with the PLC; the standby reads from the PLC. The standby synchronizes with the server.



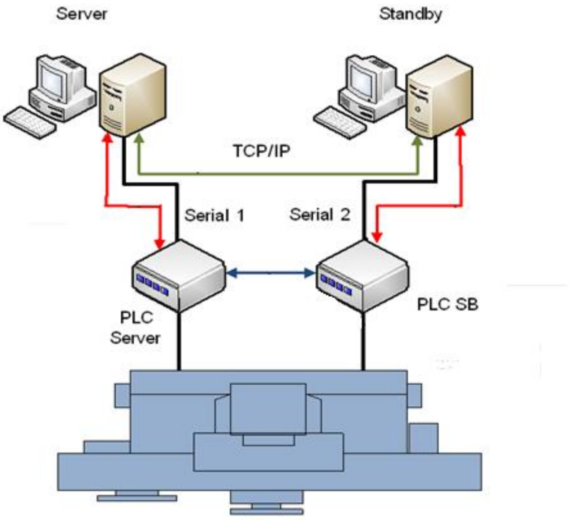
In the event of a server failure:

- The standby upgrades itself to become the server
- Seamless redundancy (on page 85) ensures that all data is complete without omissions, including data from the downtime between failure and switching
- The standby communicates with the PLC both ways as a server

HARDWARE REDUNDANCY

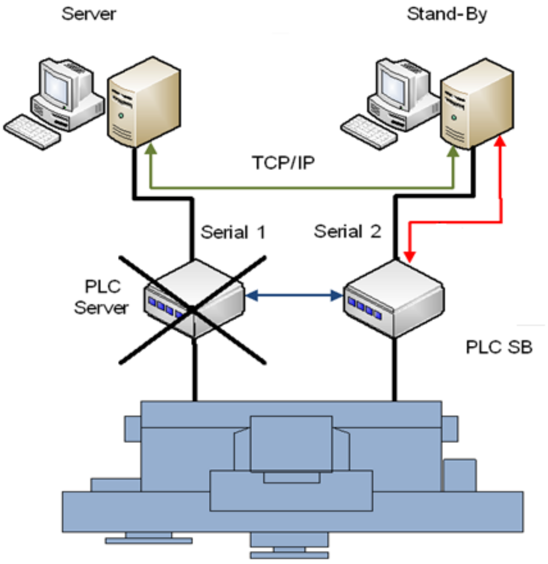
Software redundancy consists of: Two redundant PLCs and two redundant computers (server and standby server).

In operation, the server communicates both ways with the PLC; the standby communicates with the PLC both ways. The standby synchronizes with the server; the second PLC synchronizes with the first PLC.



In the event of a server failure or a failure of the server's PLC:

- The standby upgrades itself to become the server
- The standby communicates with the PLC as a server
- Seamless redundancy (on page 85) ensures that all data is complete without omissions, including data from the downtime between failure and switching



REDUNDANCY WITH ZENON

Protection from redundancy is achieved in that each project server is backed up by a second server, the standby server.

zenon makes it possible for you to configure:

- Seamless redundancy (on page 85): Prevents loss of data; can be implemented with just two computers.

  The standby recognizes a server failure and automatically assumes complete functionality of the servers. In order to avoid data loss in the time between the server failure and recognition of the failure (down time), the standby buffers all data. 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.

- Circular redundancy (on page 88): Backs up several projects at the same time; can be implemented with three computers.

  Combination of seamless redundancy with multi-project administration.

Both forms of redundancy can be configured very easily. You can find details on the configuration of redundant networks here:

- zenon seamless redundancy (on page 85)
- zenon circular redundancy (on page 88)
- Redundancy for zenon Logic and zenon (zenon Logic Runtime manual)

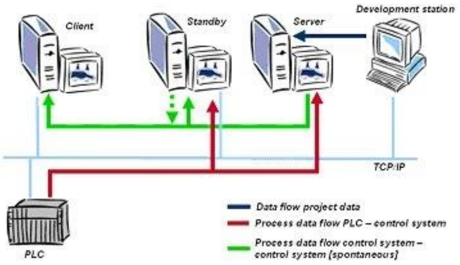
For zenon redundancy, only the server and standby server need to be defined. The server and standby server can continue to be used as operator workspaces. Project changes are administered on the project server; the standby server and the connected clients automatically synchronize online data. All computers always have the same project status.

> **Info**
>
> *If only one PLC connection is available which offers only one-way communication, you can activate option* `Stop at the Standby Server`*in the general settings of the driver configuration. For this the driver is stopped at the Standby Server and only started at the upgrade.*

9.1 Seamless Redundancy

Seamless redundancy prevents loss of data in the event of server failures. The down time between failure and recognition of the failure is backed up.

For seamless redundancy in a project, you only need two simple, standard PCs:


Client
Standby
Server
Development station
TCP/IP
PLC
Data flow project data
Process data flow PLC – control system
Process data flow control system – control system (spontaneous)

PROCEDURE FOR SEAMLESS REDUNDANCY

- A server and a standby-server are jointly in charge of a project.
- Like in a normal client/server network, the server has the data ownership.
- The standby server receives all data and remains fully operable.
- The standby server records all historical data such as alarms, CEL and archives and synchronizes recipes, users, etc. with the server.
- Because the information is always coming from the server, it is guaranteed to be up-to-date and consistent.
- If the server breaks down, the standby server upgrades itself and takes over all tasks. No data is lost in the downtime (the time between the failure and recognition of the failure) either (seamless redundancy).
- All connected clients are informed of the server failure and automatically connect to the new server.
- If the original server goes back online again
  - It connects to the standby server
  - Synchronizes the data

- Upgrades itself to the server again.

No data is lost in the process.

- All linked clients, including the standby server, reassign themselves to the server.

CONFIGURATION OF SEAMLESS REDUNDANCY

1. Define a standby server (on page 87)
2. The standby buffers all data
3. In the event of a server failure, the standby takes over
4. The buffer is reloaded and the downtime is thus backed up

SPECIAL SETUPS IN THE COMMUNICATION BETWEEN SERVER AND STANDBY SERVER

Note the communication rules between the server and standby in the following setup:

- The server is idle -> the standby server is running -> the server is started -> the server obtains all Runtime data from the standby

There may be conflicts in exceptional cases:

1. Project changes with the server stopped
2. Server role not clear due to network problems

1. PROJECT CHANGES WITH THE SERVER STOPPED

If you make changes to a project whilst the server is not operational and bring these into effect on this server only before a synchronization, these changes are overwritten again if the serer obtains its data from the standby.

To prevent this: Bring the changes into effect on the standby too before updating.

2. SERVER ROLE NOT CLEAR DUE TO NETWORK PROBLEMS

It is possible that both computers see themselves as the server in exceptional cases. The cause of this can be, for example, a loss of network connection due to a switching failure, a loose network cable etc. In this case, the communication between the 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, then the configured server has data sovereignty. That means: The standby server's current data could be overwritten.

To prevent this:

1. Always check the **`current server`** **`SYSDRV.chm::/25959.htm`** with the system variable**:**
   You will see the role that Runtime has and discover duplicate servers.
2. End the zenon project on the server that lost the network connection.
3. Set up the network connection again.
4. Restart the zenon project on the server.
5. The project then starts as a standby, updates its data and only then reverts to its server role.

Hint: Monitor the network connection with the Redundancy Management Tool (on page 90).

9.1.1 Configuring a standby server

To set up a standby server:

1. Open, in the project properties in the Editor, the `Network` group.
2. Enter, in the `Standby` property, the name of the computer that is to serve as a redundant server. (the computer must have a connection to the PLC.)

You can enter the computer name:

a) By selecting from the drop-down list after clicking on the `...` button

b) Type it in manually

Select, in the `Redundancy type` property, the desired redundancy form from the drop-down list:

- `Software redundancy:`

  The system consists of one PLC and two redundant control system computers. Both computers must have a connection to the PLC. Both computers communicate with the control and at the same time keep the data from the control updated. The communication to the control is managed by the computer which is the server. The server communicates bidirectionally, the standby communicates unidirectionally. If the server crashes, the standby server takes over the bidirectional communication with the PLC.

Only zenon is executed redundantly. The PLC is not redundant.

- `Hardware redundancy:`

The system consists of two redundant PLCs and two redundant control system computers. Each server communicates bidirectionally with one PLC. Both computers and both PLCs are synchronizing their data. If one component in the first system crashes, the second system takes over.

Both zenon and the PLC are are executed redundantly.

9.2 zenon circular redundancy

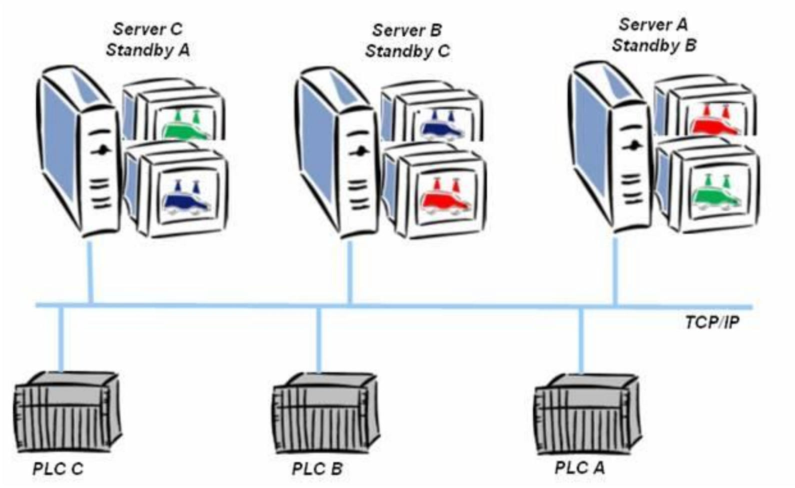
zenon circular redundancy allows seamless redundancy for several projects with a low amount of hardware being used. B

Two computers are normally required for each redundant project: One server and one standby. Three projects therefore require at least six computers. Just three computers are sufficient for protecting three projects with zenon circular redundancy. Another PC is added for each further project. For this, zenon combines multi-project administration (on page 25) with seamless redundancy (on page 85).

CONCEPT OF CIRCULAR REDUNDANCY

Circular redundancy uses the possibility of multi-project administration: Several projects can run simultaneously on one PC. Each PC is the server for one project and at the same time the standby server for the neighboring project; and additionally, it can be the client for other 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



- Project A is running as a server project on computer 1; project B is running as a standby project.
- On computer 2, project B is running as a server project; project C is running as a standby project.
- On computer 3, project C is running as a server project; project A is running as a standby project.

The circle is closed!

- Each computer can be a client for all projects at the same time.
- Expense: Three computers and three Runtime licenses

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 PCs can also be clients for other projects allows the easy realization of a low-cost, fail-safe, highly available production line.

TIME SYNCHRONIZATION FOR ZENON CIRCULAR REDUNDANCY

If zenon time synchronization (on page 11) is active, the standby and clients always receive the current time from the server. This makes no sense when using zenon circular redundancy, because the individual PCs are server and standby 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 11).

9.3 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.

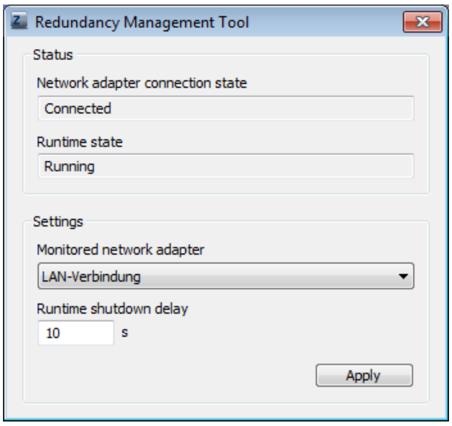
START AND CONFIGURATION

The Redundancy Management Tool can be configured via a dialog or via command line.

To open the dialog, there a three possible ways:

- From the Windows start folder: *Start -> All programs -> COPA-DATA -> Tools -> Redundancy Management Tool*
- Via the Startup Tool: *Tools -> zenon_redman*
- Direct start of the file **`zenon_redman.exe`** from the zenon program folder

After the start the Redundancy Management Tool is also displayed as symbol in the right area of the Windows task bar. Double click on the symbol to open the configuration dialog:



Redundancy Management Tool

Status

Network adapter connection state: Connected

Runtime state: Running

Settings

Monitored network adapter: LAN-Verbindung

Runtime shutdown delay: 10 s

Parameters	Description
Status	Status of the network adapter.
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	Settings.
Monitored network adapter	Selection of the network adapter which should be monitored from the drop-down list. List displays all found adapters in the device.
Runtime shutdown delay	Setting of the delay time in seconds before the Runtime is closed. In this period of time the operator can cancel the closing of the Runtime. - Maximum value: 2147483647 s. Values above this are interpreted as 0.
Apply	Applies the settings, writes values in the INI file and closes the dialog.

INI FILE

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

Parameters	Description
[DEFAULT]	
ADAPTER=	Selected LAN connection.
DELAY=	Value for delay time.

COMMAND LINE

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

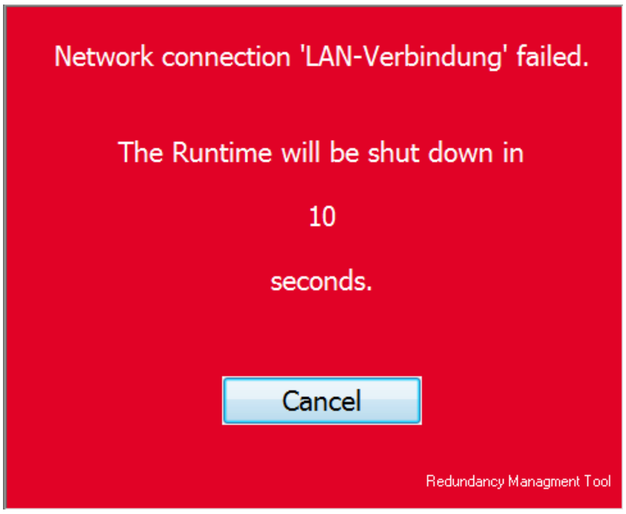
Possible parameters:

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

> **Info**
>
> *At the configuration via command line:*
>
> - 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 monitors continuously the network connection. If the connection is interrupted, the Redundancy Management Tool displays a warning and closes the Runtime after the configured delay time.



Network connection 'LAN-Verbindung' failed.

The Runtime will be shut down in

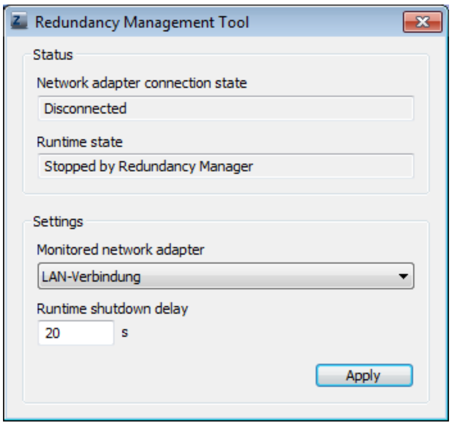
10

seconds.

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

Click on button `Cancel` 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.

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



Redundancy Management Tool

Status

Network adapter connection state: Disconnected

Runtime state: Stopped by Redundancy Manager

Settings

Monitored network adapter: LAN-Verbindung

Runtime shutdown delay: 20 s

TROUBLESHOOTING

ERROR MESSAGES

Error are displayed by pop-up messages.

Message	Meaning
`GetAdapterAddresses not supported on this patform! 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

In the log file of the Diagnosis Viewer the following is documented:

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. Routing

For routing, the packets of subordinate projects are sent to the first client project (**FCP**) in the branch.

- 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` and the `Standby` of the subordinate projects are amended on that of the FCP; this is the `Server`/`Standby` active in runtime.

GENERAL NOTES ON ROUTING

BASIC RULES

Two basic rules should 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 in one level (circular redundancy) act as a server or Standby several times. It must not be defined as a server a level above or below.

- Rule 2: Single-user projects

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 client connection to the client, the packet is sent there.
2. If there is no client 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 client 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.

> **Info**
>
> *The server and standby must 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 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.*

RULES FOR ROUTING BEFORE ZENON VERSION 6.50:

1. The first client network project of a branch on a PC defines the server and standby for all subordinate projects in the branch. This also applies
   - If a subordinate project on this PC server or standby were
   - 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.

CHECKING THE ROUTING

To check the routing settings, use the procedure from "Administering network topology (on page 71)".

10.1 Routing example

Routing for:

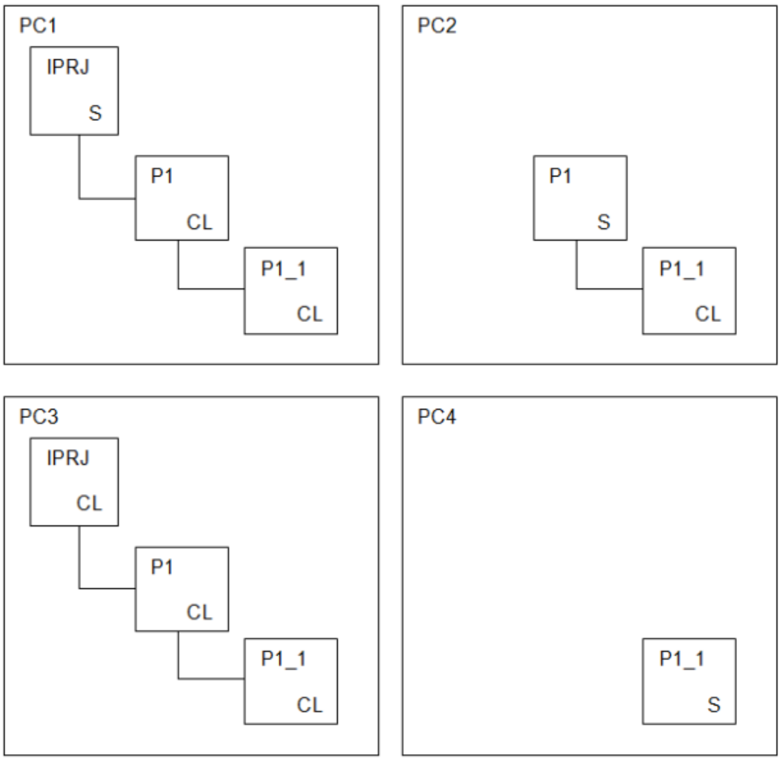
- PC3 P1 + P1_1 via PC1
- PC1/P1_1 via PC2

INITIAL SITUATION

The following graphics include the computer, the projects that are running on it and the server name configured.

- PC1 ... PC4 = computers 1 to 4
- Px = project number
- IPRJ = integration project
- S = server

- CL = client



SERVER IN RUNTIME:

PC1	
IPRJ	PC1
P1	PC2
P1_1	PC2

PC2	
P1	PC2
P1_1	PC4

PC3	
IPRJ	PC1
P1	PC1
P1_1	PC1

PC4	
P1_1	PC4

EXAMPLE

If a packet is sent from PC3/P1_1 to its server, the route is as follows:

PC3/P1_1 -> PC1 -> PC2 -> PC4

Connection between the computers:

Client makes connection	Server
PC3	PC1
PC1	PC2
PC2	PC4

11. Authorization in the network

A network project can be operated from all stations with the basic settings. Operation here means: To actively intervene in the process, i.e: setting values, executing recipes, acknowledging 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 values that is entered last overwrites all previous ones

In zenon you have the possibility to allow operation of the project only from one station at the time. In this case a user has to get the authorization, before he can operate the project. Opening screen, reading lists, etc. still is possible from all stations.

PROCEDURE

If the `Authorization in network active` property is active:

- Authorization must be obtained if active operation takes place.
- If operation is blocked by another computer, a dialog is opened on the compute that is blocking it.
- The user that is blocking it can approve the authorization or keep it blocked.
- If there is no response, the authorization is approved after a pre-set time-out.
- If an interruption in the network connection is recognized, then the authorization for this computer is reset.

For details, see the chapter:

- Configuring authorization (on page 101)
- Authorization in Runtime (on page 103)

SYSTEM VARIABLES FOR AUTHORIZATION

The system variables inform you about authorization:

- `Computer with authorization`: Name of the computer that has the authorization (string)
- `Authorization present for this computer`: Computer has authorization (Bool)
- `Authorization not granted`: Computer requests authorization, but does not receive it (Bool)

For details, see the Network messages from the system driver (on page 123) chapter.

11.1 Configuring authorization

To enable authorization in the network, you must:

- Activating authorizations and setting time-outs
- Configuring functions for operation in Runtime
- Note the behavior of user administration in the network (on page 107)

ACTIVATING AUTHORIZATIONS

This user authorization in the network has to be activated in the project properties.

- navigate to the `Network` group in Project Properties.
- Activate the `Authorization in network active` property.
- Define the `Timeout for request [s]`:

  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.

  Default: `60` seconds

- Define the `Timeout for authorization [s]`:

  Defines the period of time in which a computer that has authorization must report to the 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 compute that cannot be contacted.

  Default: `60` seconds.

  Attention: Select a time period shorter than the network time-out in the `Timeout [s]` property

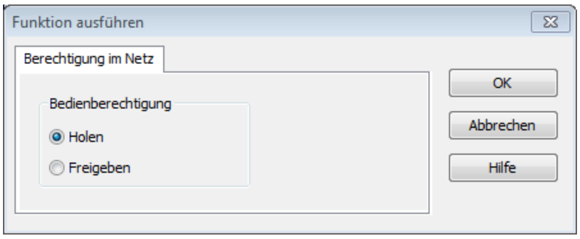
FUNCTIONS FOR AUTHORIZATION IN RUNTIME

To obtain such authorizations or to approve these, the corresponding functions must be available in Runtime. To do this, draft 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 `Authorization in network` function in the **Network** group.
3. The selection dialog for authorizations in the network is opened.

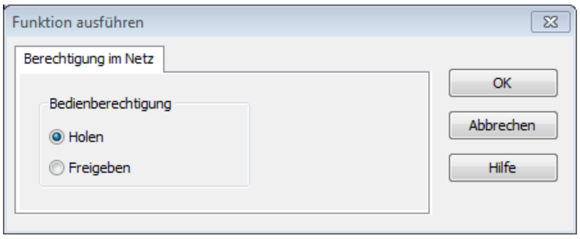


4. Select `Get`.

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 `Authorization in network` function 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.

11.2 Authorization in Runtime

If the Authorization in network active property is active, active operations are only executed in Runtime if the station is authorized.

EXAMPLE

A set value should be set for a variable:

- The set value is not sent to the hardware.
- Instead, a message box opens informing you that you do not have the authorization for this project.
- Click on the button to obtain the authorization.

DOES NOT BLOCK AUTHORIZATION FOR ANY OTHER COMPUTER:

If the operation is freely available:

- You receive the authorization
- You can now set the set value
- You can, after the operation, make this available to others using the 'Approve authorization' function

BLOCKS AUTHORIZATION FOR ANOTHER COMPUTER:

If authorization is blocked:

- A dialog is opened on the computer that is blocking
- The user of the computer that is blocking must explicitly release the authorization
  - Yes: The authorization is passed over to the other computer
  - No: Authorization remains blocked
  - No reaction: Countdown corresponding to the time-out time defined in the Timeout for authorization [s] property runs out. The authorization is automatically released at 00:00.

12. zenon functions in the network

When using functions in the network, the place of execution (on page 110) must be noted. The place of execution can be freely configured for some functions, this is stipulated for others.

Special functions for the network are:

- Authorization in network (on page 104)
- Redundancy switch (on page 105)

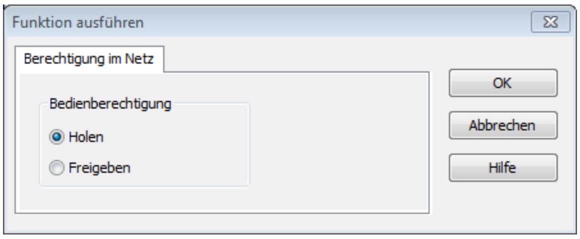
12.1 Authorization in network

To obtain such authorizations or to approve these, the corresponding functions must be available in Runtime. To do this, draft 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 `Authorization in network` function in the **`Network`** group.
3. The selection dialog for authorizations in the network is opened.



Funktion ausführen

Berechtigung im Netz

Bedienberechtigung

● Holen

○ Freigeben

4. Select `Get`.

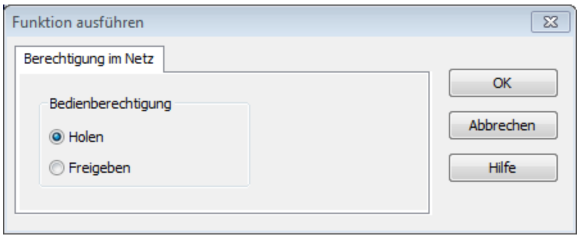
If this function is executed in Runtime, the authorization can be obtained from the user's own station.

APPROVE AUTHORIZATION

1. Create a new function.

2. Select the `Authorization in network` function in the **Network** group.

3. The selection dialog for authorizations in the network is opened.



Funktion ausführen

Berechtigung im Netz

Bedienberechtigung

● Holen

○ Freigeben

4. Select `Approve`.

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

12.2 Redundancy switch

With this function, switching between the server and standby server is possible. The current server thus becomes the standby and vice versa. The change is permanent until:

- The function is executed again

or

- Runtime files from the Editor are reloaded

Scenarios of this being used in practice are, for example: Maintenance work on the server, improved hardware connection to the standby, etc.

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

To configure the function:

1. Create a new function.
2. Select, in the Network group, the Redundancy switching function.

   (further adjustments are not necessary.)
3. Link the function to a button.

13. Behavior of zenon modules in the network

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

13.1 AML and CEL

ALARMING

Alarming is administered by the server. The server answers requests for alarming from the clients. Changes are synchronized between the server and standby.

CHRONOLOGICAL EVENT LIST

The CEL is administered on the server. Changes are synchronized between the server and standby.

13.2 Archiving

Archiving is carried out on the server.

The server synchronizes the archive data with the standby and responds to enquiries from the clients.

13.3 User administration

User administration is administered on the server. Log-in procedure:

1. The login request is sent to the 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 server.

13.4 Files

FILES

Lists for the files of all modules are created when data is exchanged between the server and the standby server. The server monitors these lists for changes. Changes that are detected are transferred to the standby.

> ⚠ **Attention**
>
> *The server does not react to watchdogs that are sent by the standby when lists are created. Note the time for the* `Timeout [s]` *property in the network properties when configuring.*

With Remote Transport, the following files are transferred to the target system: In doing so, all files are always transferred to the folder:

1. Standard

All files that are in the project's Runtime directory.
These files determine the appearance and behavior of the project and are transferred as standard:

a) all screen files (screen name.zpp)

b) amlcel.cmp

c) archiv.cmp

d) fpm.cmp

e) functions.cmp

f) project.cmp

g) projekt.mdb (not CE)

h) remas.cmp

i) rezepturen.cmp

j) scripts.cmp

k) templates.cmp

l) variables.cmp

m) zuweisung.cmp

n) Plus: project.ini and projekt.vba, which are always in the project directory

> **Info**
>
> *Files with the following suffixes are not transferred by default:*
>
> - .hot
> - .ho
> - .ret
> - .re

1. 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 subdirectories of the project directory:

a) `\zenon\custom\graphics`: for graphics

b) `\zenon\custom\lists`: for language tables

c) `\zenon\custom\media`: for all media files

d) `\zenon\custom\reports`: For Report Generator tables

e) `\zenon\custom\help`: For help files

f) `\zenon\custom\additional`: For additional files

g) `\zenon\custom\drivers:` for drivers

h) `\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.

2. Setting the start project

For Runtime, the start project must always be entered in `zenon6.ini`. To do this, click on the **`Set remote Runtime start project`** in the Remote Transport toolbar. In doing so, the following entries are set:

```
[PATH]

VBF30=project path

[DEFAULT]

DEFANWENDUNG30=project name
```

GLOBAL PROJECT

If there is a global project in the workspace, it will automatically be listed for the transport. No additional settings have to be entered. Always all files necessary for the global project will be transported.

⚠ **Attention**

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

13.5 Extended Trend

Extended Trend shows information from archives and online data. This data is saved on the server.

The server synchronizes the amended data with the standby and responds to enquiries from the clients.

13.6 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

> **Info**
>
> *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.

- `Server`: Function will be executed on the server.
- `Standby`: Function will be executed on the 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
  - Server
  - Standby
  - Client

Function	Adjustable	Current computer	Server	Standby	Client
AML and CEL					
Alarms: ackn. flashing	-	O			
Alarms: delete	-		O	O	
Alarms: Acknowledge	-		O	O	
Activating/deactivating an alarm/event group	-	O			
Activate/deactivate alarm message list / alarm/event groups / alarm/event classes	-		O	O	
Active alarm message list	-		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 alarm list or CEL	+	O			
Create/print IPA document	-		O		
Switch online printing on/off	-		O	O	
Start online printing on a new page	+	O			
Switch online printer	-		O		
Application					
Select printer	+	O			
Start EMS	-		O		
Stop EMS	-		O		
Print extended trend diagram	+	O			
Switch color palette	+	O			

Functions with active limit values	-		O	O	
Functions with active/inactive limit values	-		O	O	
Functions with inactive limit values	-		O	O	
Open Help	+	O			
Reload	+	O			
Determine open maintenances	-		O		
PFS - execute user-defined event	+	O			
Activate/deactivate project simulation	-	O			
Simulate right mouseclick	+	O			
Save remanent data	+	O			
Stopping the Runtime	+	O			
Analyze S7 Graph heuristics	+	O			
Execute SAP function	+	O			
Language switch	+	O			
Topology - ground fault search	-		O		
Topology - LoadShedding	-		O		
Historian					
Archive: Stop	-		O	O	
Index Archive	-		O		
Archive: Start	-		O	O	
Export archive	-	O			
Display open archive	-		O	O	
User administration					
Change user	+	O			
Login with dialog	+	O			

Login without password	+	O			
Logout	+	O			
Change password	-	O			
Screens					
Change ALC source color	+	O			
Indexed screen	-	O			
Screen : close	+	O			
Screen: Return to last	-	O			
Screen: Move center	+	O			
Screen switch	+	O			
Focus: Activate input to the element with the focus	+-	O			
Focus: set to frame	+	O			
Move focus	-	O			
Focus: Delete from frame	+	O			
Show menu	+	O			
Monitor assign	+	O			
Runtime profiles	+	O			
Close the frame	+	O			
Set point input for keyboard screen	-	O			
Displaying the overview window	+	O			
Error detection in the electric network					
Acknowledge ground fault message	+	O			
End ground fault search	+	O			
Start ground fault search	+	O			
Acknowledge ground fault message	+	O			
Message Control					

Note: Place of execution can be set freely in theory. Changes have no effect however. Message Control is always executed on the server.					
Show recipient-database	+	O			
Send a Message	+	O			
Send Message: activate	+	O			
Send Message: deactivate	+	O			
Network					
Authorization in network	+	O			
Redundancy switch	-			O	
Report Generator					
Print report	+				
Execute report	+				
Export report	+				
Recipes					
Recipegroup Manager	-	O			
Standard Recipe	-	O			
Standard recipe single directly	+	O	O	O	O
Standard recipe single with dialog	-	O			
Standard recipe single with online dialog	-	O			
Script					
Script: Execute	+	O			
Script: Select online	+	O			
Variable					
Export data	-		O		
Read a dBase-file	+	O			

Print current values	+	O			
Unit conversion	+	O			
Trend-Values on	-		O	O	
Trend-Values off	-		O	O	
Trend values inactive/active	-		O	O	
Send value to hardware	-		O		
Driver commands	-	O			
Transfer simulation image to standby	-				O
Write time to variable	+	O			
Read time from variable	+	O			
VBA					
Open PCE editor	-		O		
Open VBA Editor	+	O			
Execute VBA Macro	+	O			
Show VBA macro dialog	+	O			
VSTA					
Open VSTA editor	+	O			
Execute VSTA macro	+	O			
Show VSTA macro dialog	+	O			
Windows					
Play audio file	+	O			
File operations	+	O			
Start continuous tone	+	O			
Stop continuous tone	+	O			
Window to the background	-	O			
Window to foreground	-	O			
Print screenshot	+	O			

`Start program`	+	O			

13.7 Message Control

Message Control is server-dependent.

- Data is administered on the server.
- Messages are always sent by the server.
- Changes must be made on the server.
- Changes that are carried out on a client are lost.

> **Info**
>
> *A place of execution other than the server can, in theory, be defined using the properties in the* `Execution` *group. These settings have no effect however. The corresponding functions continue to be carried out automatically on the server.*

13.8 Programming interfaces

VBA AND VSTA

Code in VBA or VSTA is always executed locally on the system on which it is started.

The place of execution can however be defined otherwise when this is called up via the function (on page 110).

PCE

The PCE is always executed locally on the system on which it is started.

13.9 Report Generator

The **`*.xrs`** files of the Report Generator are synchronized on all systems in the network (clients, standby, server).

EDITOR

If the file in the zenon Editor is modified and transferred to the server, the server sends this amended file to all other computers in the network via the push service.

RUNTIME

If the file is amended in Runtime, the changes are only saved on a temporary basis and replaced at the next reload or when Runtime is restarted.

13.10 Recipes

The execution of recipes is different for standard recipes and the RGM.

STANDARD RECIPES

Standard recipes are administered on the server and standby.

If a standard recipe is changed by a user in Runtime, the client requests the full recipe list from the server. In the event of changes, the recipe list is sent back to the server.

> **Info**
>
> *This list is not identical to that of the file rezepturen.cmp*

If a recipe is changed and executed in Runtime on the client, it is executed with the new values. When the standard recipe, you are given the option to save the changes.

RECIPEGROUP MANAGER

When the Recipe Group Manager screen is loaded on the client, a list of all recipe names is requested by the server. As soon as a recipe is selected, it is loaded by the server.

13.11 Scripts

Scripts combine several functions. The place of execution depends on the settings of the **Execute script** 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: Server Single-user project: Active computer
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: 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.	Client
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.	Client
AUTOSTART_SRVPRJ	Script is executed automatically when Runtime is started for any desired project on the project server before the start screen is loaded.	Server
AUTOSTART_SRVPRJ	Script is automatically executed when Runtime of a desired project is ended on the project server.	Server

13.12 Driver - Variables - Rema

Only the driver and the standby communicate with the PLC in the zenon network. Client requests are routed via the server. It obtains the information from the PLC and forwards this to the client. Limit values are monitored by the server.

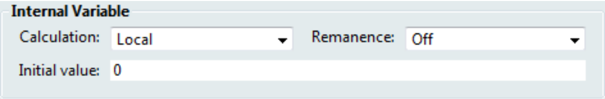
DRIVERS

Drivers are only executed on the server.

INTERNAL VARIABLES

For internal variables, it is possible to define whether each individual variable is calculated locally or in the network in zenon. To do this:

1. Navigate to the `Internal Variable` node in the properties of internal variables



Internal Variable

Calculation: Local

Remanence: Off

Initial value: 0

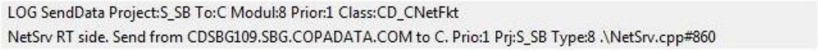
2. Define the place of execution using the `Calculation` property:
   a) Local: The internal variables are evaluated and administered locally for network projects, i.e. on the client. The values are not synchronized with other computers in the network.
   Attention: The limit values of locally-calculated variables are also monitored locally. This has a consequence:
   - The values of the variables can be different on each client.
   - Functions are executed.
   - Alarming does not work; alarms are only administered by the server.

   b) Network: For network projects, the internal variable is assessed and administrated on the Server of the project. It has the same value on the server and on all clients.

FUNCTIONS AND INTERNAL VARIABLES

1. Calculation in the network: Client

A limit value of an internal variable calculated in the network triggers a function that is to be executed on a client. The following log entries are generated in the event that a limit value is exceeded.

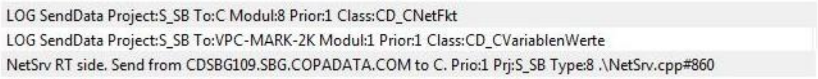


```
LOG SendData Project:S_SB To:C Modul:8 Prior:1 Class:CD_CNetFkt
NetSrv RT side. Send from CDSBG109.SBG.COPADATA.COM to C. Prio:1 Prj:S_SB Type:8 .\NetSrv.cpp#860
```

The server is responsible for breaches of limit values and thus triggers the function on the client.

2. Calculation in the network: Standby instead of client

A limit value of an internal variable calculated in the network triggers a function that is to be executed on a client. However no client is available, just a standby. The following log entries are generated in the event that a limit value is exceeded.

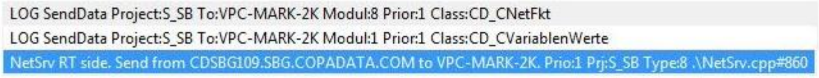


```
LOG SendData Project:S_SB To:C Modul:8 Prior:1 Class:CD_CNetFkt
LOG SendData Project:S_SB To:VPC-MARK-2K Modul:1 Prior:1 Class:CD_CVariablenWerte
NetSrv RT side. Send from CDSBG109.SBG.COPADATA.COM to C. Prio:1 Prj:S_SB Type:8 .\NetSrv.cpp#860
```

The server treats the standby as a client.

3. Calculation in the network: Standby

A limit value of an internal variable calculated in the network triggers a function that is to be executed on a standby. The following log entries are generated in the event that a limit value is exceeded.

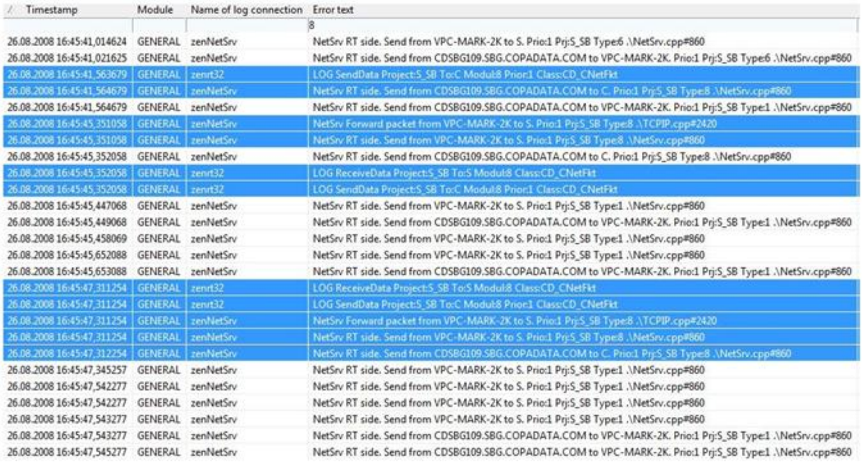


```
LOG SendData Project:S_SB To:VPC-MARK-2K Modul:8 Prior:1 Class:CD_CNetFkt
LOG SendData Project:S_SB To:VPC-MARK-2K Modul:1 Prior:1 Class:CD_CVariablenWerte
NetSrv RT side. Send from CDSBG109.SBG.COPADATA.COM to VPC-MARK-2K. Prio:1 Prj:S_SB Type:8 .\NetSrv.cpp#860
```

The server sends the standby a message to execute the function.

4. Calculation in the network: Client and standby

A limit value of an internal variable calculated in the network triggers a function that is to be executed on a client and the standby. The following log entries are generated in the event that a limit value is exceeded.



Timestamp	Module	Name of log connection	Error text
			8
26.08.2008 16:45:41,014624	GENERAL	zenNetSrv	NetSrv RT side. Send from VPC-MARK-2K to S. Prio:1 Prj:S_SB Type:6 .\NetSrv.cpp#860
26.08.2008 16:45:41,021625	GENERAL	zenNetSrv	NetSrv RT side. Send from CDSBG109.SBG.COPADATA.COM to VPC-MARK-2K. Prio:1 Prj:S_SB Type:6 .\NetSrv.cpp#860
26.08.2008 16:45:41,563679	GENERAL	zenrt32	LOG SendData Project:S_SB To:C Modul:8 Prior:1 Class:CD_CNetFkt
26.08.2008 16:45:41,564679	GENERAL	zenNetSrv	NetSrv RT side. Send from CDSBG109.SBG.COPADATA.COM to C. Prio:1 Prj:S_SB Type:8 .\NetSrv.cpp#860
26.08.2008 16:45:41,564679	GENERAL	zenNetSrv	NetSrv RT side. Send from CDSBG109.SBG.COPADATA.COM to VPC-MARK-2K. Prio:1 Prj:S_SB Type:1 .\NetSrv.cpp#860
26.08.2008 16:45:45,351058	GENERAL	zenNetSrv	NetSrv Forward packet from VPC-MARK-2K to S. Prio:1 Prj:S_SB Type:8 .\TCPIP.cpp#2420
26.08.2008 16:45:45,351058	GENERAL	zenNetSrv	NetSrv RT side. Send from VPC-MARK-2K to S. Prio:1 Prj:S_SB Type:8 .\NetSrv.cpp#860
26.08.2008 16:45:45,352058	GENERAL	zenNetSrv	NetSrv RT side. Send from CDSBG109.SBG.COPADATA.COM to C. Prio:1 Prj:S_SB Type:8 .\NetSrv.cpp#860
26.08.2008 16:45:45,352058	GENERAL	zenrt32	LOG ReceiveData Project:S_SB To:S Modul:8 Class:CD_CNetFkt
26.08.2008 16:45:45,352058	GENERAL	zenrt32	LOG SendData Project:S_SB To:C Modul:8 Prior:1 Class:CD_CNetFkt
26.08.2008 16:45:45,447068	GENERAL	zenNetSrv	NetSrv RT side. Send from VPC-MARK-2K to S. Prio:1 Prj:S_SB Type:1 .\NetSrv.cpp#860
26.08.2008 16:45:45,449068	GENERAL	zenNetSrv	NetSrv RT side. Send from CDSBG109.SBG.COPADATA.COM to VPC-MARK-2K. Prio:1 Prj:S_SB Type:1 .\NetSrv.cpp#860
26.08.2008 16:45:45,458069	GENERAL	zenNetSrv	NetSrv RT side. Send from VPC-MARK-2K to S. Prio:1 Prj:S_SB Type:1 .\NetSrv.cpp#860
26.08.2008 16:45:45,652088	GENERAL	zenNetSrv	NetSrv RT side. Send from VPC-MARK-2K to S. Prio:1 Prj:S_SB Type:1 .\NetSrv.cpp#860
26.08.2008 16:45:45,653088	GENERAL	zenNetSrv	NetSrv RT side. Send from CDSBG109.SBG.COPADATA.COM to VPC-MARK-2K. Prio:1 Prj:S_SB Type:1 .\NetSrv.cpp#860
26.08.2008 16:45:47,311254	GENERAL	zenrt32	LOG ReceiveData Project:S_SB To:S Modul:8 Class:CD_CNetFkt
26.08.2008 16:45:47,311254	GENERAL	zenrt32	LOG SendData Project:S_SB To:C Modul:8 Prior:1 Class:CD_CNetFkt
26.08.2008 16:45:47,311254	GENERAL	zenNetSrv	NetSrv Forward packet from VPC-MARK-2K to S. Prio:1 Prj:S_SB Type:8 .\TCPIP.cpp#2420
26.08.2008 16:45:47,311254	GENERAL	zenNetSrv	NetSrv RT side. Send from VPC-MARK-2K to S. Prio:1 Prj:S_SB Type:8 .\NetSrv.cpp#860
26.08.2008 16:45:47,312254	GENERAL	zenNetSrv	NetSrv RT side. Send from CDSBG109.SBG.COPADATA.COM to C. Prio:1 Prj:S_SB Type:8 .\NetSrv.cpp#860
26.08.2008 16:45:47,345257	GENERAL	zenNetSrv	NetSrv RT side. Send from VPC-MARK-2K to S. Prio:1 Prj:S_SB Type:1 .\NetSrv.cpp#860
26.08.2008 16:45:47,542277	GENERAL	zenNetSrv	NetSrv RT side. Send from VPC-MARK-2K to S. Prio:1 Prj:S_SB Type:1 .\NetSrv.cpp#860
26.08.2008 16:45:47,542277	GENERAL	zenNetSrv	NetSrv RT side. Send from VPC-MARK-2K to S. Prio:1 Prj:S_SB Type:1 .\NetSrv.cpp#860
26.08.2008 16:45:47,543277	GENERAL	zenNetSrv	NetSrv RT side. Send from VPC-MARK-2K to S. Prio:1 Prj:S_SB Type:1 .\NetSrv.cpp#860
26.08.2008 16:45:47,543277	GENERAL	zenNetSrv	NetSrv RT side. Send from CDSBG109.SBG.COPADATA.COM to VPC-MARK-2K. Prio:1 Prj:S_SB Type:1 .\NetSrv.cpp#860
26.08.2008 16:45:47,545277	GENERAL	zenNetSrv	NetSrv RT side. Send from CDSBG109.SBG.COPADATA.COM to VPC-MARK-2K. Prio:1 Prj:S_SB Type:1 .\NetSrv.cpp#860

The server triggers the function on the client and sends the standby a message to execute the function.

REMA

Reaction matrixes are calculated on the client and server.

If a function is executed by a Rema, the place of execution is determined by the settings for the function.

13.13 Time control

Time control is executed on the server and standby.

The function triggered is executed on the systems that were selected for execution of the function in the settings.

13.14 Allocations

Assignments are always carried out by the server and not for the original variable.

Assignments from the clients are always ignored.

14. Network messages from the system driver

The following system driver variables are available for this subject area:

Name	Data type	Comment
`Current server`	STRING local	Computer name of the server that is currently handling the process If the name was acquired from the hosts file, this is the name entered there. For DNS, this is the Fully Qualified Doman Name. **Note:** If the network is deactivated, the variable sends the status INVALID. The **`Current standby server`** remains empty in contrast.
`Current standby server`	STRING local	Computer name of the server which is currently not handling processes. If the name was acquired from the hosts file, this is the name entered there. For DNS, this is the Fully Qualified Doman Name.
`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.
`Authorization: not granted`	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 = Request for Authorization granted 1 = Request for Authorization denied
`Authorization exists (yes(1)(no(0))`	BOOL	Shows whether there is an authorization for the current project on the local computer. 0 = no 1 = yes
`Authorization: Computer that owns it.`	STRING	Shows the name of the computer that has the authorization for the currently loaded project.
`Names of connected clients`	STRING	Delivers the names of the clients currently connected to the server. The standby server, if there is one, is also included.
`Network timeout [milliseconds]`	UDINT	Shows the timeout in milliseconds for the zenon network as configured in the project configuration.
`Redundancy switch`	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 redundancy switch 1 = Redundancy switch
`Server offline`	BOOL	Indicates that the connection to the process handling server

	local	was lost. 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 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 was promoted to be the process handling server. EVALUATION: Preferably via a 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.
`Server stop`	BOOL local	Indicates the regular stop of the process handling server. The value changes to TRUE if the process handling 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 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 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.
`Server-Standby in data update`	BOOL	A binary variable that takes on the value 1 if the server and the standby server are synchronizing files after a redundancy switch. 0 = No file synchronization 1 = File synchronization active
`Standalone/Server/Standby-Server/Client`	DINT	Shows the type of the local computer in the network. -1 = Single user 0 = Client 1 = Server 2 = Standby Server
`Standby-Server offline`	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.
`Standby-Server stop`	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. 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.

`Standby-Server start`	BOOL	TRUE if the non-process handling server has registered at the process-handling server and if the data update was performed 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.
`Switch from Server to Standby`	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.
`Switch from Standby to Server`	BOOL	A binary variable that takes on the value 1 if the standby server becomes the server during a 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.