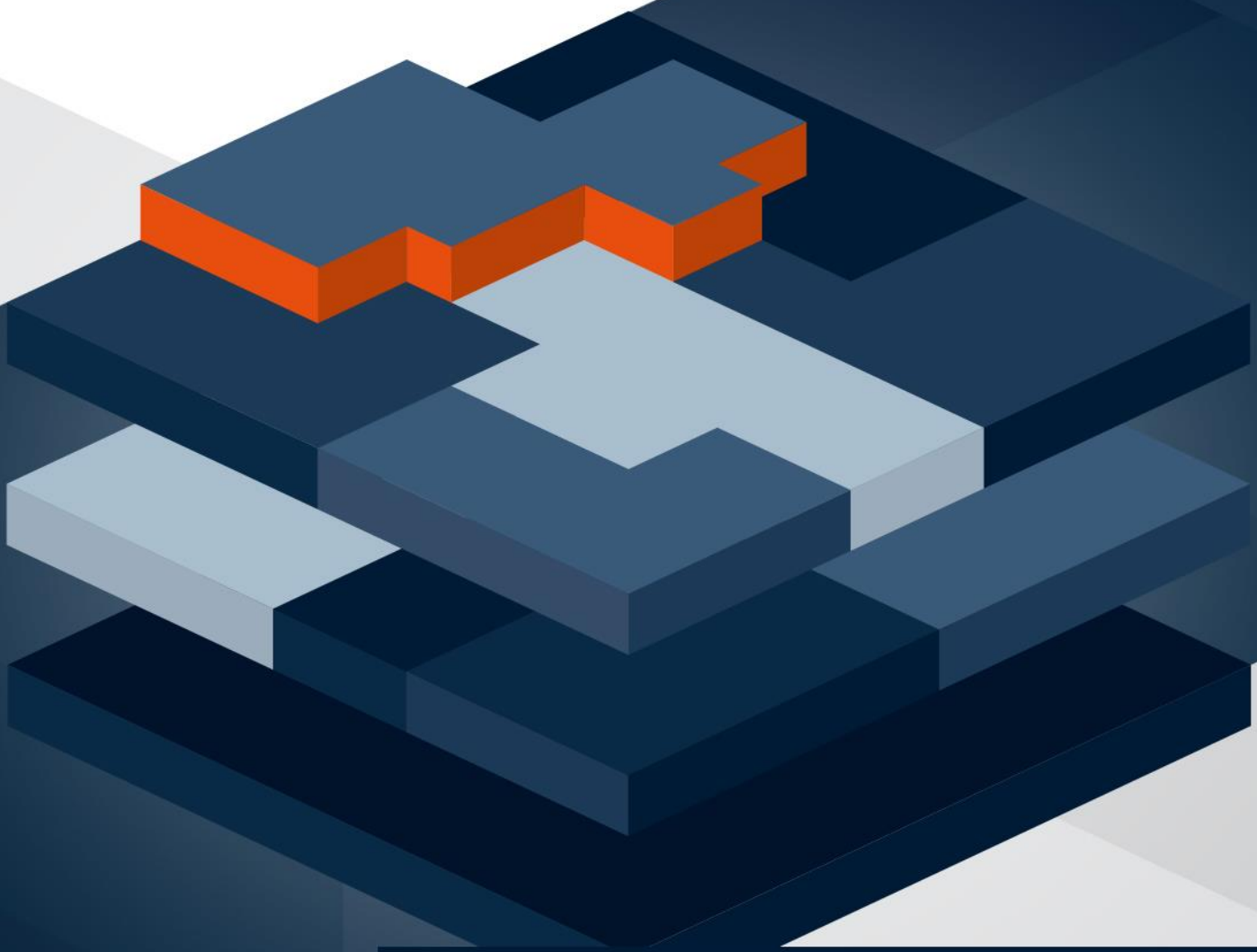
zenon
by COPA-DATA



zenon manual

Installation and updates

v.11

Table of contents

1 Welcome to COPA-DATA help

ZENON VIDEO TUTORIALS

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

GENERAL HELP

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

PROJECT SUPPORT

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

LICENSES AND MODULES

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

2 Installation and updates

During the first installation of zenon, the installation routine automatically starts and leads you through the whole installation process. If the autoplay of media is disabled in the operating system of the computer, the installation routine will not start automatically. Start the installation by executing the **START.exe** file in the root folder of your zenon installation medium.

Hint

The autoplay of media can be enabled in the settings of your operating system.

- Press the **Windows button + I** to open the Windows settings dialog.
- Enter *Enable/Disable Auto Play* in the input field.
  This opens the system configuration dialog.
- Select the **Use AutoPlay for all media and devices** option.

Notes for the installation:

- Before installing zenon:
  - All current operating system updates must be installed
    Note: If you always use the latest version (Service Pack) of your operating system, you not only avoid compatibility issues but also security problems.
  - There must not be a restart pending
- During the installation of zenon, the **COPA-DATA Multiple Network Protocol Driver (cdprotdrv.sys)** is installed. To start the driver, the operating system must be restarted after installation.

Attention

From Version 7.10 on, zenon cannot be installed on systems on which the **Microsoft SQL Server Data Engine** (**MSDE**) is already installed. This affects all systems in which zenon 6.01 or 6.20 has been installed.

Information

If you receive an error message during installation stating that a service cannot be started, then:

- first reboot the computer
- then start the zenon setup again

3 zenon Software Platform standard installation

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

The zenon software platform is available in different embodiments with different names. Product names and the scope may differ from the standard installation described.

> **Attention**
>
> The computer is automatically restarted during installation if necessary. Close all other programs before installation.

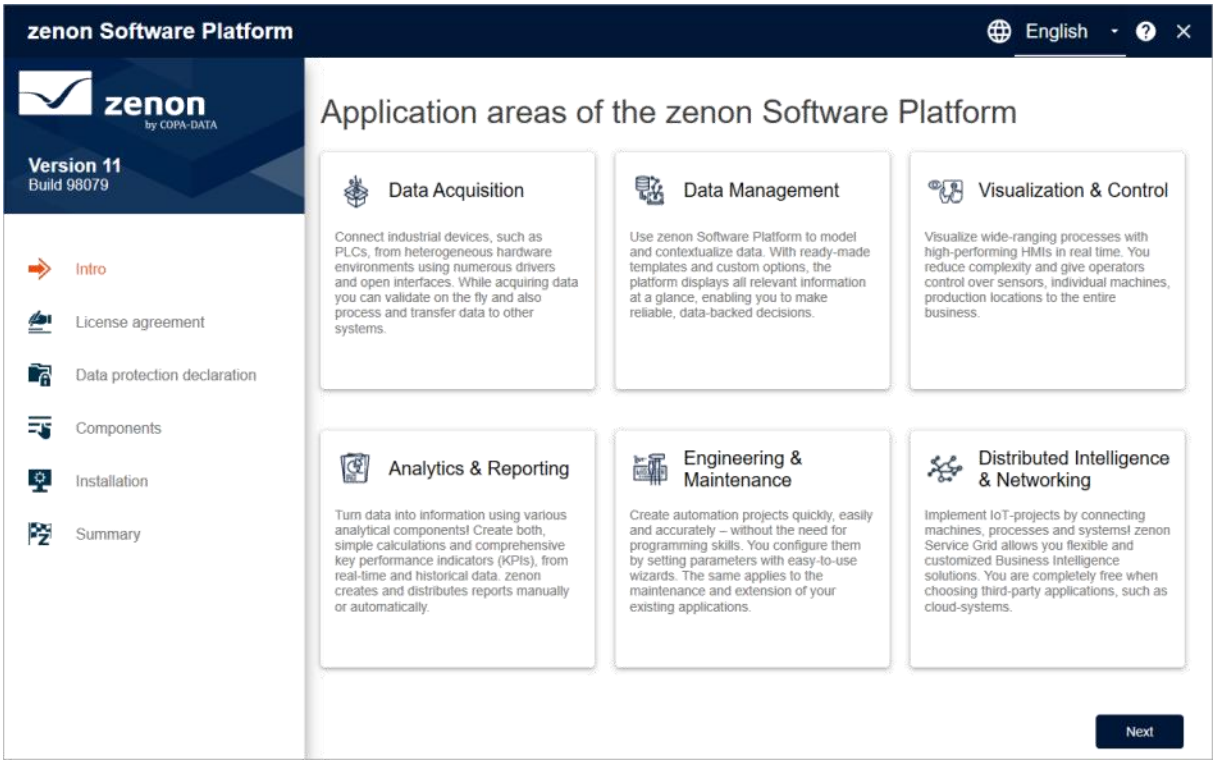
Administrator rights are required for the installation process on the computer!

> **Attention**
>
> The minimum screen resolution for the setup is 1280 x 800 pixels.

3.1 Start window

You are given general information about the zenon Software Platform in the start window.

The information in the left window shows you the current status of the installation process. You switch to the next respective window with the **Next** button.
You can get help on installation by clicking on the Help symbol at the top right.



1. From the drop-down list at the top right, select the desired language for installation.
   The following languages are available:
   - German
   - English
   - Italian
   - French
   - Spanish
   - Czech
   - Japanese
   - Korean
   - Chinese
   - Russian

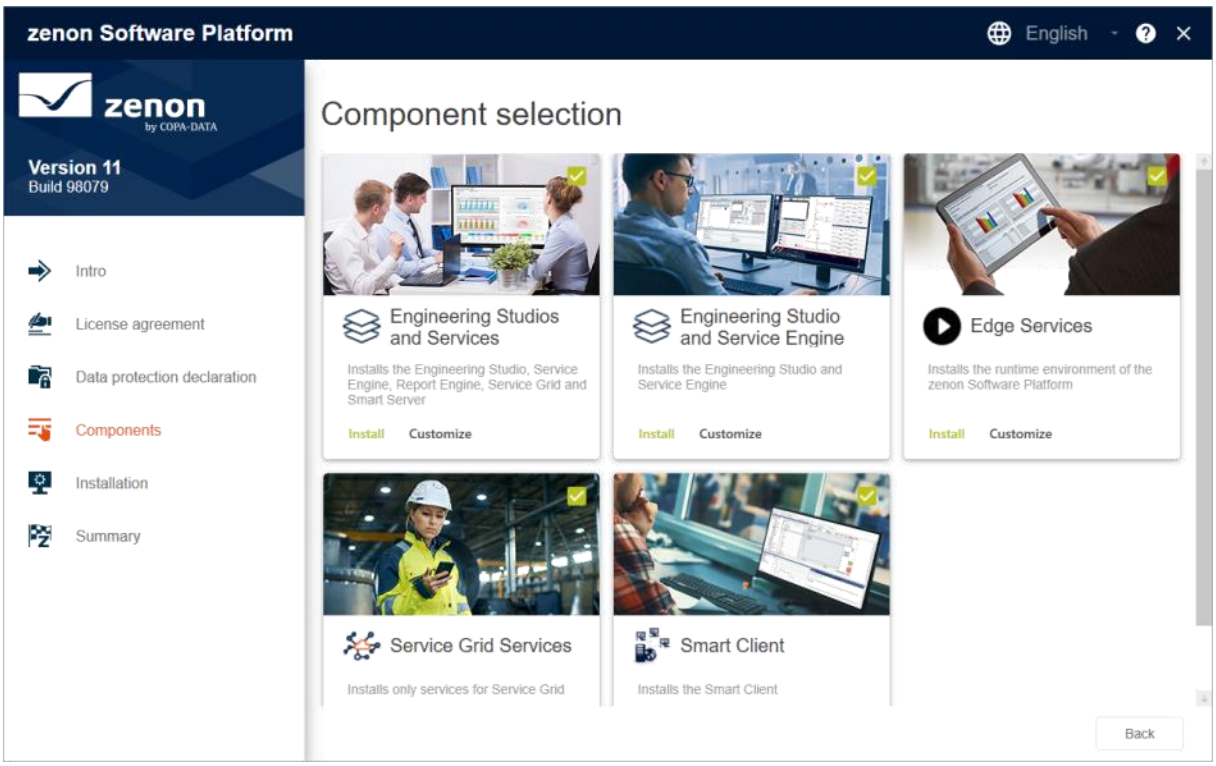
   Note: The language can only be changed on this page. In the following steps, the language is shown but can no longer be amended.
2. Clicking on the **Next** button opens the window with the license conditions.

3. Confirm the license conditions by activating the corresponding checkbox.
   If you do not accept the license conditions, you cannot install the product.

   You can also print the license conditions out by clicking on the **Print** button.
4. Clicking on the **Next** button opens the privacy policy.
   Read the privacy policy carefully.
   You can print out the privacy policy by clicking on the **Print** button.
5. Activate the checkbox for the privacy policy.
   This will confirm that you have read this. If you do not accept the privacy policy, the product cannot be installed.
6. Clicking on the **Next** button opens the window to select the desired product.
   Note: The **Next** button is only available if you have agreed to the license conditions by clicking on the option field.

3.2 zenon Standard installation

Select the desired components It is only possible to select components that have not already been installed. If you want to carry out a reinstallation, you must first uninstall the previously-installed component using the Control Panel.



INSTALLATION PACKAGES

There are five collections of packages available for installation.
you can amend these individually before installation.

Package	Components
Engineering Studios and zenon Softwareplattform Services	Installed: ▸ Engineering Studio ▸ Report Engine ▸ Smart Server ▸ Service Grid Services ▸ License Manager ▸ Service Engine ▸ Reporting Studio Smart Client can be added using the **Amend** button.
Engineering Studio and Service Engine	Installed: ▸ Engineering Studio ▸ License Manager ▸ Service Engine Further components can be added using the **Amend** button.
Edge Services	▸ Service Engine ▸ Report Engine ▸ Smart Server ▸ Service Grid Services ▸ Reporting Studio ▸ License Manager Further components can be added using the **Amend** button.
Service Grid services	Installed: ▸ Service Grid Services ▸ License Manager Further components can be added using the **Amend**

Package	Components
	button.
Smart Client	Installs Smart Client.

3.3 Configuration and installation

All packages can be installed with a click. You also have the possibility to amend the installation packages individually.

COMPLETE INSTALLATION

To install a package in full:

1. Select the desired package.
2. Click on the **Install** button.

The installation is started. The computer may be restarted automatically during installation. Follow the instructions of the wizard

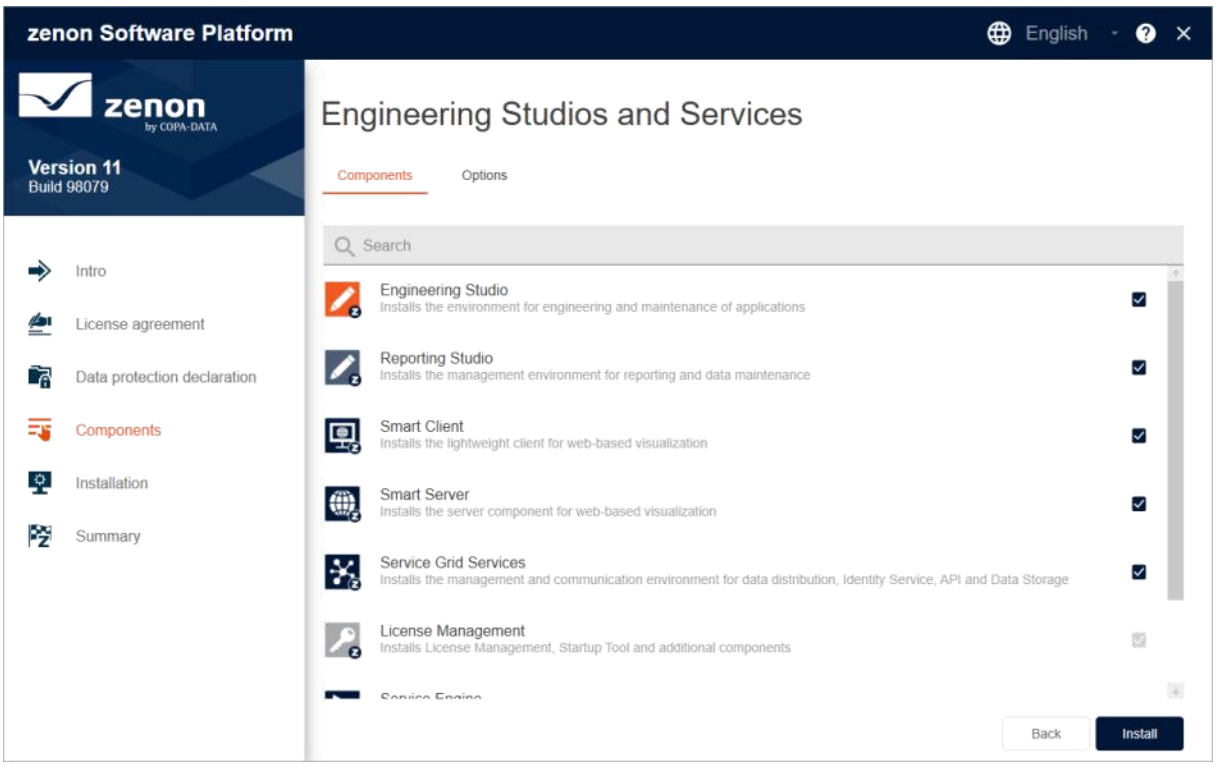
CUSTOMIZED INSTALLATION

You can amend the packages individually.

To install a package with an amended installation:

1. Select the desired package.
2. Click on the **Customize** button.

The dialog to amend the installation is opened.
The standard components have already been pre-selected.



3. In the **Components** tab, select or deselect the desired components.
   Components can only be deselected if they are not required by another component.
   Already-installed components cannot be deselected.
4. If necessary, configure the paths for installation in the **Options** tab.
5. If necessary, select the **Harden installation** option in the **Options** tab.
   In this case note the additional necessary configuration of the connections (on page 15) after installation.
6. Click on the **Install** button.

The installation is started.
The computer may be restarted automatically during installation. Follow the instructions of the wizard

Attention

If there is already a version of Service Grid on the system, it is strongly recommended that you back up this installation before an update. You can find information about this in the **Backup and Restore – Persistence Instance** (on page 20) section.

3.3.1 Hardening zenon

If you activate the **Harden Installation** option, TCP/IP communication is limited:

- zenon only permits local access.
- The SQL Server only permits local access.
- The firewall rules only allow zenon services to have local access

This primarily has effects on distributed engineering. You must amend the SQL Server, firewalls and connection to your requirements after installation. You can find information about the configuration of hardened systems in the Help in the **Security - distributed engineering with hardened system** section.

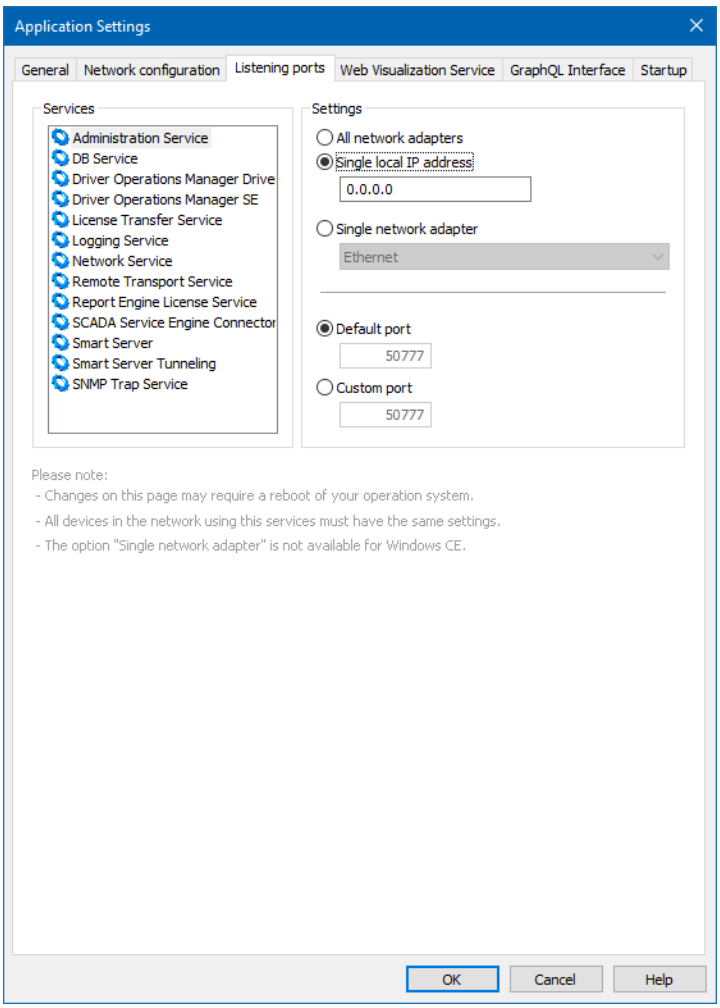
AMENDMENT IN THE STARTUP TOOL

When zenon is installed, exceptions are created in the Windows firewall by the setup. They are configured for applications and services that open a TCP Listening Port.

After installation, configure the exceptions in the Windows firewall more restrictively, appropriate to their environment and to the necessary applications and services.

On multi-homed systems with multiple network cards, zenon applications and services with their default settings open the TCP Listening Port for all network cards present in the system. However communication throughout all network cards is often not necessary and not desirable.

After installation of zenon, configure the TCP Listening Ports for the respective services and applications, according to their environment and requirements. Use the **Startup-Tool** to do this. Only allow communication between the network card or IP address that is absolutely necessary for this. If you assign a service to the Loopback adapter or the IP address *127.0.0.1*, you only allow local communication. This way, local Diagnosis Server access to local diagnosis clients can be limited, for example.



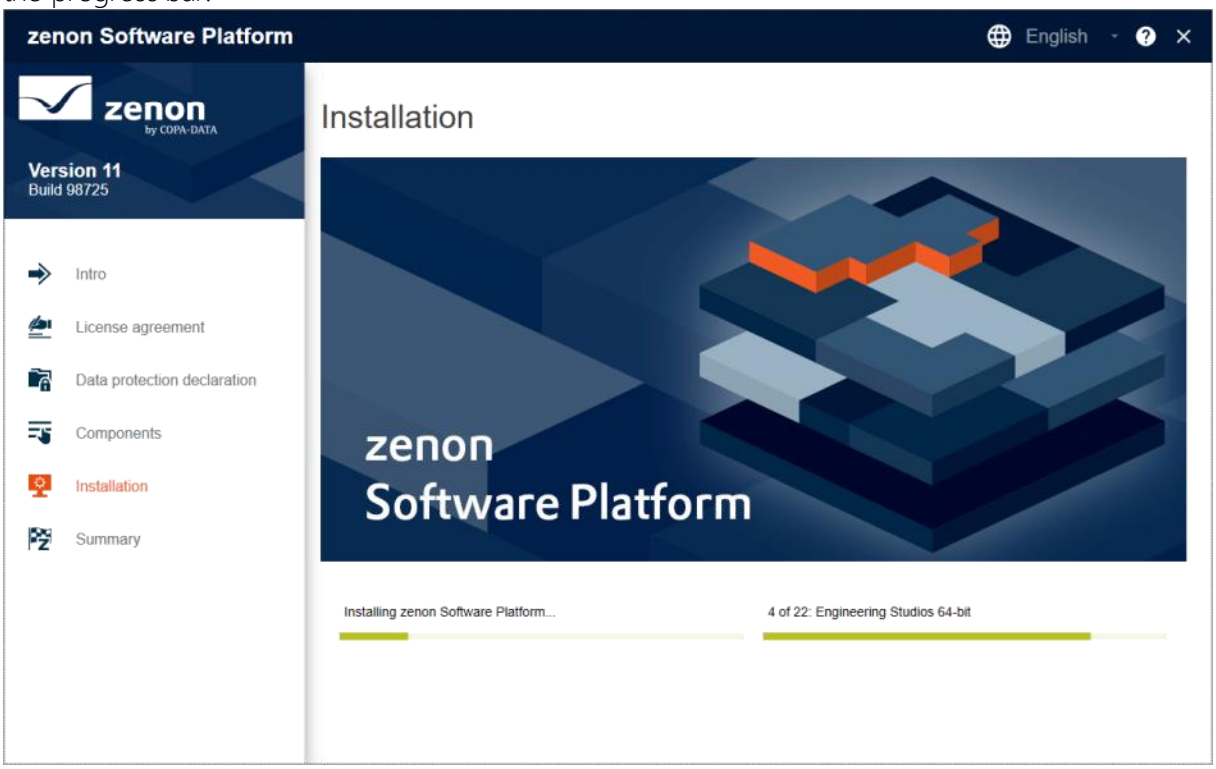
Note the following for platform functionalities that are independent of the configuration of the services:

Service	Platform functionality
Administration Service	Provides general functionalities for the operation of zenon.
DB Service	The database service is only needed by Engineering Studio. If you want to use **distributed engineering**, the service must be able to communicate via the network.
Driver Operations Manager	The **Driver Operations Manager** administers the local

Service	Platform functionality
	driver instances of Service Engine. It must be contactable from outside.
License Transfer Service	For **Remote Licensing** of zenon components, they must be able to communicate via the network. The service can also be limited again after concluding the licensing.
Logging Service	For **Remote Logging** via the **Diagnosis Viewer**, the service must be contactable via the network.
Network Service	This service must be reachable via the network for the use of : ▸ zenon network ▸ Smart Client
Remote Transport Service	Serves to manually transfer Service Engine files from an external instance of Engineering Studio. The service must be reachable via the network for this. With local instances of Engineering Studio, the service can only be operated locally.
Report Engine Licensing Service	For **Remote Licensing** of Report Engine, it must be able to communicate via the network. The service can also be limited again after concluding the licensing.
SCADA Service Engine Connector	This service must be able to be contacted via the network if Report Engine, Service Grid or remote Service Engine drivers: ▸ are used with a remote instance ▸ access online Service Engine data
Smart Server	This service is necessary for the operation of the Smart Server. Recommendation: Operate it as a separate instance, separate from Service Engine.
Smart Server Tunneling	This service is necessary for the operation of the Smart Server. Recommendation: Operate it as a separate instance, separate from Service Engine.
SNMP Trap Service	This service must be contactable via the network if you use zenon SNMP services.

3.4 Installation and finishing

During installation, you are informed of the installation progress and the current installation stage in the progress bar.



The installation process may take some time. Do not turn your computer off in this time. Please also ensure that your computer is not automatically put into sleep mode.

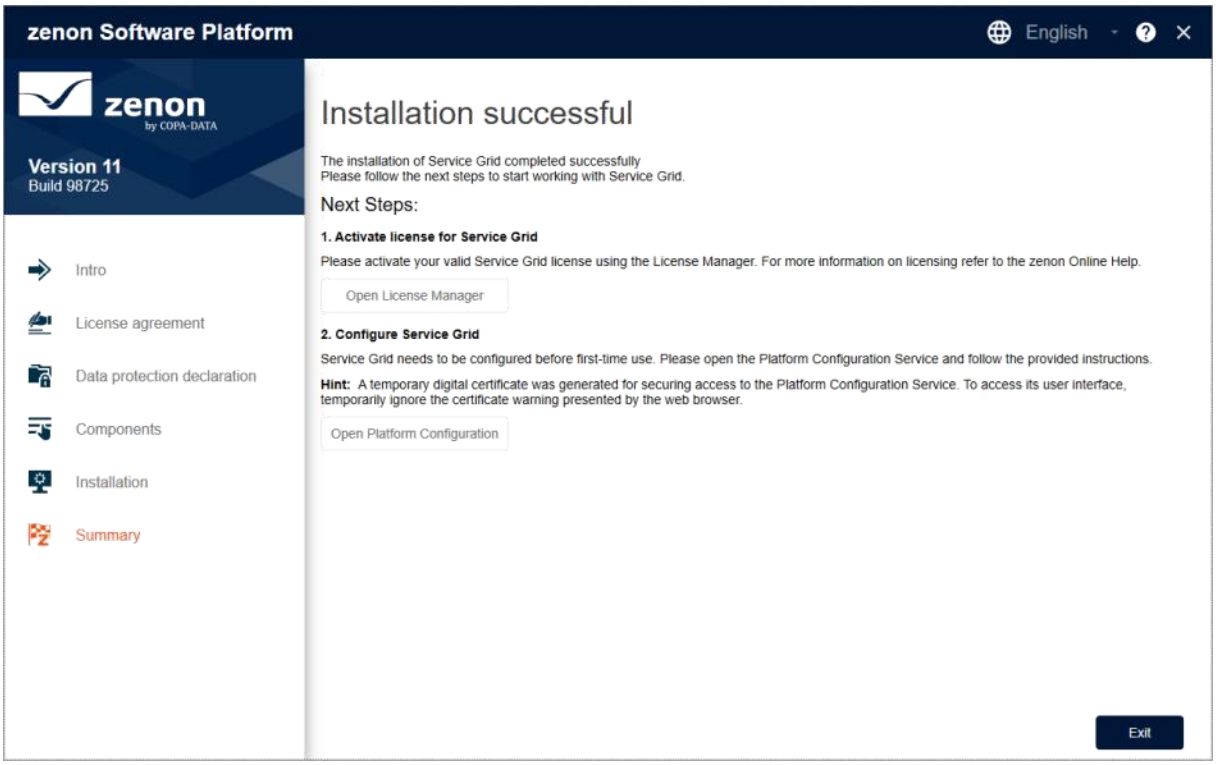
INSTALLATION IS COMPLETED

You receive a message about the success of the installation at the end of installation.

If Service Grid has been installed:

1. Now activate the license for Service Grid using the corresponding button.

2. Configure Service Grid using the corresponding button.



CANCEL INSTALLATION

You can cancel the installation by clicking on the **Cancel** button.
Before canceling the installation there is a security query.
Possible actions:

- **Yes**: The installation is canceled.
  The dialog for an invalid installation is shown.
- **No**: The installation is continued.

CANCELED OR INCORRECT INSTALLATION

If an error occurs during installation or the installation was canceled, this is shown in a dialog.

1. Click on the **Finish** button to finish the setup.
2. Check the log messages when canceling took place, if there are any.
3. Attempt the installation again.

Note: If installation of the SQL Server fails, the error message will provide a link to the log files. Click on the link to receive a detailed error report.

POSSIBLE CAUSES OF THE ERROR:

Important possible causes for a cancellation:

- A pending update to the Windows operating system.
- The SQL server required for Engineering Studio could not be installed.

3.5 Backup and Restore – Persistence Instance

The following applies for the persistence:

- The Persistence Instance should be backed up before every update of Service Grid. This is a precaution.
- A restore of the backup is only necessary in rare cases. This is the case, for instance, if a problem occurs during an update.

The Persistence Instance is based on MongoDB. The CLI tools mongodump and mongorestore of the database manufacturer can be used for backup and restore. Both tools are described in the following chapters.

You can find further information in the documentation on *www.mongodb.com*.

Info: Host operating system and backup folder

The following applies for all tutorials in this chapter:

- You execute the CLI tools locally on the host operating system on which Service Grid is installed natively or in Docker.
- The backup is stored in the host operating system in the *backups* folder.

The backup commands described create the *backups* folder relative to the folder path in which you are located during the command processing in PowerShell.

3.5.1 CLI tools: mongodump and mongorestore

You can use the CLI tools mongodump and mongorestore to back up and restore the Persistence Instance via the command line.

The installation option determines whether the tools are preinstalled:

1. Service Grid (Windows native) installation option:
   The CLI tools are automatically installed by Service Grid setup in this path:
   %ProgramFiles%\MongoDB\Server\<version>\bin

2. Service Grid (Docker) installation option:
   You must download the tools yourself from *www.mongodb.com* and install them on the host operating system.

The installation of CLI tools is documented in detail on *www.mongodb.com*.

Hint: Start CLI tools under Windows

A corresponding environment variable is automatically saved in the host operating system when CLI tools are installed under Windows.

Thus the following applies:

- You can basically start the CLI tools via PowerShell from any folder path. This requires that the environment variable is active.
- In some cases, the operating system must be restarted after installation for the environment variable to be active.

You can start the CLI tools at any time – regardless of the environment variable – via the installation path.

3.5.2 Determine user credentials

The commands contain the following placeholders:

- <*username*>: The user name for MongoDB
- <*password*>: The password for MongoDB

You must replace the placeholders in the commands with the individual user credentials for your system. Where the user credentials can be found depends on your Service Grid installation option.

See the comparison in the table below:

Placeholder	Service Grid (Windows native)	Service Grid (Docker)
<username>	*"AdminUser"*	*"SG_Persistence_Username"*
<password>	*"AdminUserPassword"*	*"SG_Persistence_Password"*
	Path to the user credentials: *C:\ProgramData\COPA-DATA\System\ServiceGrid\common.json*	**Path to the user credentials:** *.env* file in the installation directory of Service Grid.

3.5.3 Perform backup

These instructions basically work for all installation options of Service Grid.

Please note the following:

- Information is provided at each configuration step on which installation of Service Grid it refers to.
- You can use the configuration steps for Service Grid (Docker on Windows) for Service Grid (Docker on Linux) too. However, you have to change the folder paths for linux and use a Linux Shell.

The tools and backup commands used are basically the same under Linux and Windows.

PREPARATION (DOCKER ON WINDOWS)

In Docker you must open the database containers beforehand for access from the host operating system.

Proceed as follows:

1. Open an elevated PowerShell.
2. In PowerShell go to the installation directory of Service Grid.
   cd C:\servicegrid
   Note: You have created this folder path yourself for the installation of Service Grid. It contains all Service Grid configuration files such as *docker-compose.yml*.
3. Stop all containers:
   docker-compose down
4. Start all containers with an additional configuration file:
   docker-compose -f docker-compose.yml -f docker-compose.expose-db.yml up

Now you can access the database in the Docker containers via the Windows host system.

PERFORM BACKUP (WINDOWS NATIVE, DOCKER ON WINDOWS)

Back up Persistence Instance using mongodump as follows:

1. Open an elevated PowerShell.
2. In PowerShell go to the directory path where the backup folder should be created.
3. Use the following command to create a backup folder in the selected directory path and back up the database there:
   mongodump --username='<username>' --password='<password>' --host='localhost' --port=27017
   --archive='backups\ServiceGridSystem.archive'

Note: You must replace the <*username*> and <*password*> placeholders with the appropriate user credentials.

You have thus backed up your Persistence Instance.

POST-BACKUP ACTIVITIES (DOCKER ON WINDOWS)

You must restart all containers in Docker:

1. Stop all containers:
   docker-compose down
2. Restart the containers:
   docker-compose up

The database is thus protected again from access via the host operating system.

3.5.4 Apply restore

These instructions basically work for all installation options of Service Grid.

Please note the following:

- Information is provided at each configuration step on which installation of Service Grid it refers to.
- You can use the configuration steps for Service Grid (Docker on Windows) for Service Grid (Docker on Linux) too. However, you have to change the folder paths for linux and use a Linux Shell.

The tools and backup commands used are basically the same under Linux and Windows.

PREPARATION (DOCKER ON WINDOWS)

In Docker you must open the database containers beforehand for access from the host operating system.

Proceed as follows:

1. Open an elevated PowerShell.
2. In PowerShell go to the installation directory of Service Grid.
   cd C:\servicegrid
   Note: You have created this folder path yourself for the installation of Service Grid. It contains all Service Grid configuration files such as *docker-compose.yml*.
3. Stop all containers:
   docker-compose down

4. Start all containers with an additional configuration file:
   docker-compose -f docker-compose.yml -f docker-compose.expose-db.yml up

Now you can access the database in the Docker containers via the Windows host system.

APPLY RESTORE (WINDOWS NATIVE, DOCKER ON WINDOWS)

You can restore your Persistence Instance from a backup using mongorestore:

1. Open a PowerShell.
2. Go to the directory path of the backup folder.
3. Perform the restore of the database:
   mongorestore --username='<username>' --password='<password>' --host='localhost' --port=27017 --archive='backups\ServiceGridSystem.archive' --drop

You have thus restored the Persistence Instance from the backup.

> **Attention**
>
> The *--drop* argument in the restore causes all existing data in Persistence Instance to be deleted and replaced with data from the backup.

> **Tip**
>
> With the *--dryRun* argument, it is possible to simulate the restore of the data. In doing so, existing data in Persistence Instance are not overwritten.

POST-BACKUP ACTIVITIES (DOCKER ON WINDOWS)

You must restart all containers in Docker:

1. Stop all containers:
   docker-compose down
2. Restart the containers:
   docker-compose up

The database is thus protected again from access via the host operating system.

4 Installing additional components

You can install components of the zenon software platform that you have not yet installed at any time, via the Setup.

To do this:

1. Start the Setup for the zenon software platform.

   Because there are already components on the system, the page with the available components is opened immediately.

2. Select the desired components by clicking on the respective checkbox.
   If a further component is needed for a component, this is automatically selected too.

3. Click on **Install**.

The selected components are installed.

Note: To uninstall components open the Windows **Apps** application.

5 Uninstalling components

To uninstall the zenon software platform or individual components:

Uninstall complete software platform:

- Open the Windows **Apps** settings.
- Click on the **zenon Platform Setup** entry.
- Click on **Uninstall**.
- Confirm the confirmation prompt.

All components of the zenon software platform are uninstalled.

Uninstall components:

- Open the Windows **Apps** settings.
- Click on the **zenon Platform Setup** entry.
- Click on **Change**

  The dialog to select the components is opened.

  - Select the desired components
  - Click on Uninstall.

The selected components are uninstalled.

6 zenon Logic for Windows (standalone installation)

On the installation medium, in the *%AdditionalSoftware%\COPA-DATA* Logic Service directory, you will find the installation packages for a standalone installation for **Logic Service for Windows**.

LOGIC SERVICE - CONTENTS OF THE INSTALLATION PACKAGE

With the standalone setup for **Logic Service for Windows** all components for operating the 61131-3 compliant Logic Service are installed on the target system.This includes among other things, components for licensing and diagnostics.No configuration components are installed.

INSTALLATION REQUIREMENTS

Keep in mind the general system requirements for installing the product. Pre-installation of **zenon Operator/Supervisor** or **zenon Logic for Windows (Standalone)** is not permitted.

The product requires software already installed on the target system. You can also find them in the *%AdditionalSoftware%* directory of the installation medium. Therefore, if necessary, manually install the following packages:

- **Microsoft Visual Studio C++ Redistributables**
- **WIBU-SYSTEMS CodeMeter Runtime Kit**

After installation, execute the appropriate installation package (x86 or x64) for your target system.

> **Information**
>
> The installation does not include a license for the product. Therefore Logic Service starts in test mode. Licensing can be done using the general licensing tools.

UPDATE

To update an already installed version, uninstall it and perform a new installation.

7 Silent installation and uninstallation

zenon can also be installed and uninstalled silently (Silent Installation) and (Silent Remove).

As part of Silent Installation, it is possible to exclude certain standard components from the installation specifically:

- Firewall rules: The **CDPROP_INSTALLFIREWALL** parameter decides whether rules for the firewall are set.

- **Codemeter** Software: Can be configured using the *PREREQUISITES_* argument for the **ISFeatureInstall** parameter. Codemeter is not installed if the argument is not used.
  Attention: This argument is applicable for all *Prerequisites*. Other Prerequisites are also not installed in this case!

> **Information**
>
> All zenon versions from 7.10 on support silent installation and uninstalling.

PASSWORD CONVENTIONS FOR SA USERS

The random password created during a standard installation for the user *SA* on the SQL Server can be replaced with your own password. To do this, use an individual password for the argument **CDP_SQLADMINPW=** (Version 10 onwards) or **CDPROP_SQLADMINPASSWORD=** (before version 10).

Rules:

- Default length: 20 characters
- Permitted characters:
  - Letters: *A - Z, a - z*
  - Digits: *0 - 9*
  - Special characters: *!@$?#%&**
- Composition:
  - at least 1 lower case letter
  - at least 1 upper case letter
  - at least 1 number
  - at least 1 special character

> **Attention**
>
> A user-defined password is not validated.

7.1 As of version zenon 10

Initiation of silent *silent* installation can be carried out with the following parameters:

- **/***silent -silent /s -s*
- **/quiet -quiet /q -q**

For example:

- **SoftwarePlatform.exe** *-s* **CDP_WORKLOAD="***WlSmartClient***"**
- **SoftwarePlatform.exe** */quiet* **CDP_WORKLOAD=**"*WlFullSoftwarePlatform*"
- **SoftwarePlatform.exe** */silent* **CDP_WORKLOAD="***WlEdgeServices***"**

CDP_WORKLOAD does not make sense for installations that are not *silent*. The argument is therefore also not supported and is ignored.

Parameters	Arguments	Description
softwareplatform.exe		Call-up of the installation.
/silent		Silent installation.
CDP_WORKLOAD=	Workload to be installed. Must correspond to the ID from **WorkloadsSetup.config**. Example: *WlSmartClient* for a SmartClient installation. **Attention:** Workloads only, no features! Incorrect IDs or IDs that do not exist lead to the installation being aborted.	Entry is mandatory for *silent*. Is ignored with non-*silent*.
CDP_INSTALLFIREWALL=	▸ *1*: is installed ▸ *0*: is not installed Default:*1*	Whether firewall rules are installed.
CDP_INSTALLDEMO=	▸ 1: is installed ▸ 0: is not installed Default:*1*	Whether the demo project is also to be installed. Is only transferred to MSI.
CDP_LANGUAGE=	▸ 1031: German ▸ 1033: English ▸ 1034: Spanish ▸ 1036: French ▸ 1040: Italian ▸ 1041: Japanese ▸ 1042: Korean ▸ 1049: Russian	Selection of the language. Is only transferred to MSI.

Parameters	Arguments	Description
	Default:*1033*	
CDP_SQLPATH=	Path to SQL. *Empty*: Standard path Default: *Empty*	Path for SQL installation, as in GUI.
CDP_INSTALLDIR=	Path to 64-bit zenon installation folder. Default: default installation path	zenon software platform installation directory for 64-bit components
CDP_SQLADMINPW=	Any desired password. Must comply with SQL guidelines	SQL administrator password.
CDP_POSTINSTALLEXE=	Whether **PostInstall.exe** is executed after installation: ▸ *1*: is executed ▸ *0*: is not executed Default:*0*	
CDP_POSTINSTALLARGS=	Default:*empty*	Arguments for **PostInstall.exe**.
CDP_SERVICEHUB_PW=	Any desired password. Default:*ServiceGrid*	Password for **Service Hub**.
CDP_SQLADMINPW=	Any desired password.	Password for SQL server instance of Service Engine.
CDP_SQLADMINPW_REPORTING=	Any desired password.	Password for SQL server instance of Report Engine .

SILENT UNINSTALLATION

Uninstallation must be carried out using the same **SoftwarePlatform.exe** that was used for installation. Because this is saved in a folder with an execution-specific GUID, the following lines are also logged with each successful installation.

"For uninstalling of the currently installed producty via CommandLine use:"
"For silent uninstall C:\\ProgramData\\Package Cache\\{bundleProviderGuid}\\SoftwarePlatform.exe /silent /uninstall"
"For uninstall via UI C:\\ProgramData\\Package Cache\\{bundleProviderGuid}\\SoftwarePlatform.exe /uninstall"

In doing so, **{bundleProviderGuid}** is always replaced with the execution-specific GUID. The full path to the EXE is thus given in the LOG file.

In principle, all actions are documented in the log. Certain queries that are displayed as GUI feedback during normal installation are written here in the LOG file.

7.2 zenon 7.20

Instigation of silent installation for version 7.20.

Syntax: **scada.exe /silent /language:***[number]* **CDPROP_EDITION=***[edition]* **CDPROP_TYPE=***[type]* **ISFeatureInstall=***[features]*

Examples:

- Installation of Engineering Studio, German, Energy Edition:
  scada.exe /silent /language:1031 CDPROP_EDITION=ENERGY CDPROP_TYPE=ED ISFeatureInstall=PREREQUISITES_EDITOR,SCADA
- Installation of Service Engine, English, Supervisor Edition:
  scada.exe /silent /language:1033 CDPROP_EDITION=SUPERVISOR CDPROP_TYPE=RT ISFeatureInstall=PREREQUISITES_RUNTIME,SCADA
- Installation of Smart Server, German:
  scada.exe /silent /language:1031 ISFeatureInstall=PREREQUISITES_WEBSERVER,WEBSERVER
- Installation of Smart Client, German:
  scada.exe /silent /language:1031 ISFeatureInstall=PREREQUISITES_WEBCLIENT,WEBCLIENT

PARAMETERS

TAGs	Arguments	Description
scada.exe		Call-up of the installation.
/silent		Silent installation.
/language:	▸ *1031*: German	Selection of the language.

TAGs	Arguments	Description
	▸ *1033*: English ▸ *1034*: Spanish ▸ *1036*: French ▸ *1040*: Italian ▸ *1041*: Japanese ▸ *1042*: Korean ▸ *1049*: Russian	Example: English: **language:** *1033*
CDPROP_EDITION=	▸ *ENERGY* ▸ *SUPERVISOR* ▸ *OPERATOR* ▸ *PHARMA*	Selection of the edition. Example: Energy Edition: **CDPROP_EDITION=** *ENERGY* Is not needed for Smart Server and Smart Client.
CDPROP_TYPE=	▸ *ED*: Engineering Studio and Service Engine ▸ *RT*: Service Engine	Selection Engineering Studio or Service Engine. Example Service Engine: **CDPROP_TYPE=***RT* Is not needed for Smart Server and Smart Client.
CDPROP_INSTALLFIREWALL	▸ *0* or *1*	Denotes whether rules for the firewall have been created: ▸ *0*: Rules are not created ▸ *1*: Rules are created
CDPROP_SQLADMINPASSWORD=	User-defined password You can find further information on passwords in the **Password conventions for SA users** chapter.	Password for the *SA* user in SQL Server. This password is created for the *SA* user and used during installation. Example: **CDPROP_SQLADMINPASSWORD=***H1342DFAhzgs§*464578* If no password is transferred, a random password is created during installation.

TAGs	Arguments	Description
		Attention: User-defined passwords are not validated for validity and compliance with password rules!
ISFeatureInstall=	▸ *PREREQUISITES_EDITOR,SCADA*: Engineering Studio ▸ *PREREQUISITES_RUNTIME,SCADA*: Service Engine ▸ *PREREQUISITES_WEBSERVER, WEBSERVER*: Web Server ▸ *PREREQUISITES_WEBCLIENT,WEBCLIENT*: Web Client	Selection of features to be installed. Arguments: ▸ *PREREQUISITES_*: Decides whether Prerequisites are installed. The reasons why Prerequisites are installed is given after the underscore. E.g.: **EDITOR Codemeter** is not installed if the argument is left out. ▸ *SCADA*: Installs Engineering Studio and/or Service Engine, depending on the parameters for **CDPROP_TYPE=**. ▸ *WEBSERVER*: Installs the Web Server. ▸ *WEBCLIENT*: Installs the Web Client. Examples: ▸ Service Engine with Prerequistes: **ISFeatureInstall=***PREREQUISITES_RUNTIME,SCADA* ▸ Service Engine without Prerequistes: **ISFeatureInstall=***SCADA*

SILENT UNINSTALLATION AS OF ZENON 7.20.

The **GUID** is part of the path and depends on the version. The attendant version is visible in the file properties of a **GUID**.

Without LOG file:

- Path: **%ProgramFiles(x86)%\InstallShield Installation Information\{GUID}**
  Example: **C:\Program Files (x86)\InstallShield Installation Information\{9BE6EDFE-3465-486F-87EE-1C439DE5EA9A}**
- Syntax: **SCADA.exe /remove /silent**

With LOG file:

- Path: **%ProgramFiles(x86)%\InstallShield Installation Information\{GUID}**
  Example: **C:\Program Files (x86)\InstallShield Installation Information\{9BE6EDFE-3465-486F-87EE-1C439DE5EA9A}**
- Syntax: **SCADA.exe /remove /silent /log"%TEMP%"**

7.3 zenon 7.10 and 7.11

Syntax: **scada.exe /silent /language:[number] CDPROP_EDITION=[edition] CDPROP_TYPE=[type] ISFeatureInstall=[features]**

Parameters	Arguments	Description
scada.exe		Call-up of the installation.
/silent		Silent installation.
/language:	▸ **1031**: German ▸ **1033**: English ▸ **1034**: Spanish ▸ **1036**: French ▸ **1040**: Italian	Selection of the language. Example: English: **language:1033**
CDPROP_EDITION=	▸ **ENERGY** ▸ **SUPERVISOR** ▸ **OPERATOR** ▸ **PHARMA**	Selection of the edition. Example: Energy Edition: **CDPROP_EDITION=ENERGY** Is not required for Smart Server.
CDPROP_TYPE=	▸ **ED**: Engineering Studio and Service Engine ▸ **RT**: Service Engine	Selection Engineering Studio or Service Engine. Example Service Engine:

Parameters	Arguments	Description
		CDPROP_TYPE=RT Is not required for Smart Server.
ISFeatureInstall=	▸ **WIBU,SCADA,MS,MSALL,SQL,COMMON**: Engineering Studio ▸ **WIBU,SCADA,MS,MSALL,COMMON**: Service Engine ▸ **WIBU,WS,MSALL,COMMON**: Web Server	Selection of features to be installed. Example Engineering Studio: **ISFeatureInstall=WIBU,SCADA,MS,MSALL,SQL,COMMON**

Examples:

- Installation of Engineering Studio, German, Energy Edition:
  scada.exe /silent /language:1031 CDPROP_EDITION=ENERGY CDPROP_TYPE=ED ISFeatureInstall=WIBU,SCADA,MS,MSALL,SQL,COMMON
- Installation of Service Engine, English, Supervisor Edition:
  scada.exe /silent /language:1033 CDPROP_EDITION=SUPERVISOR CDPROP_TYPE=RT ISFeatureInstall=WIBU,SCADA,MS,MSALL,COMMON
- Installation of Smart Server, German:
  scada.exe /silent /language:1031 ISFeatureInstall=WIBU,WS,MSALL,COMMON

SILENT UNINSTALLATION IN ZENON VERSION 7.10 AND 7.11

ZENON 7.10

Path: **C:\Program Files (x86)\InstallShield Installation Information\{860C41F0-6034-4822-BCF1-88D4849AE897}**

Syntax: **SCADA.exe /remove /silent**

ZENON 7.11

Path: **C:\Program Files (x86)\InstallShield Installation Information\{ED00D319-77B8-4C58-8D67-2DA2D48E90DB}**

Syntax: **SCADA.exe /remove /silent**

8 Error treatment

CHECK BEFORE INSTALLATION:

The system requirements are checked before installation. If the requirements are not met, you are shown these on a separate page with notices on how to rectify this.

ERROR DURING INSTALLATION

You will receive an error message if there are errors during installation.

If you need help from the Technical Consulting department of COPA-DATA:

1. If possible, create a screenshot of the error message
2. Navigate to the *%Temp%/SoftwarePlatform* folder.
3. Here you can find the log files of the installation.
4. Create a ZIP file with the content of the folder.
5. Forward the file and the screenshot to support@copadata.com

If you have already closed the error message window, you can find the log files with the installation information for the SQL Server in the folder:
C:\Program Files\Microsoft SQL Server\150\Setup Bootstrap\LOG

Tip: The file **summary.txt** provides information for troubleshooting.

Information

Firewalls: zenon automatically configures the firewall installed with Windows during installation. Firewalls from other providers must be properly configured by the user

FREQUENT SOURCES OF ERROR DURING INSTALLATION:

- The virus scanner is active and blocks the installtion because the scanner thinks it´s a virus. Solution: Seperate the system from the network, disable the virus scanner, execute the installation again.
- The firewall was not configured correctly. Solution: Seperate the system from the network, disable the firewall, execute the installation again.
- Erroneous SQL-installation on the system. Solution: Create project backups; if possible, deinstall SQL server, rename the SQL folder, and restart installation.

9 Windows Updates

Attention: Automatic Windows updates influence the installation

If an update of the Windows operating system is carried out while the zenon setup is running, it can cause problems during the zenon installation.

To prevent this:

- Deactivate the automatic Windows update during the time of installation.
- carry out the Windows update before starting the zenon installation

10 Virus scan

Anti-virus software can slow down or even prevent the installation of zenon.

Note: If the anti-virus software you use leads to problems during installation, deactivate the anti-virus software for the duration of the installation. The computers concerned can be exposed to higher risks during this time. Activate your anti-virus software immediately after the installation of zenon.

11 File Structure

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

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

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

Folder	Path
Program folder	32-bit system: *%ProgramFiles%\COPA-DATA\zenon Software Platform 10* 64-bit system: *%ProgramFiles%\COPA-DATA\zenon Software Platform 10* *%ProgramFiles (x86)%\COPA-DATA\zenon Software Platform 10*
Program data folder, e.g. global symbols, print	*%ProgramData%\COPA-DATA\zenon1000*

Folder	Path
templates, log files etc.	
Database folder (SQL)	*%ProgramData%\COPA-DATA\SQL2019*
System folder	*%ProgramData%\COPA-DATA\System*
Settings Engineering Studio and profiles	*%Users%\UserName\AppData\Local\COPA-DATA\zenon*Engineering Studio
Setting for Diagnosis Viewer.	*%Users%\UserName\AppData\Local\COPA-DATA\zenon\DiagView*

DEFINITION SERVICE ENGINE FOLDER AND DATA FOLDER

SERVICE ENGINE FOLDER

Engineering Studio creates Service Engine files in the Service Engine folder, or they are transferred to this folder by means of Remote Transport. The Service Engine folder is created or updated when compiling a project in Engineering Studio. This folder can be parameterized in Engineering Studio with the **Service Engine folder** project property. With remote transfer, the Service Engine folder is defined in the Remote Transport settings.

DATA FOLDER

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

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

Attention

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

12 Free ports

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

This is how you check the port assignments:

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

   You can reach the command line in Windows:

   - by pressing the Windows-key and R
   - Enter *cmd* and confirm with **OK**.
   - A DOS-window pops up
   - enter the command *netstat*
2. A list of all currently used TCP and UDP ports will pop up.
3. Check the listening ports (status: *LISTEN*) if the process-ID (PID) of the ports needed by zenon and zenon Logic corresponds with the processes of zenon and zenon Logic.

   These PIDs can be read in the **Windows Task Manager**. To do this, open the **Windows Task Manager** and switch to the **Services** tab.
4. If another software uses these ports, reconfigure this software.
   You can see the ports that zenon and zenon Logic use in the **Port assignment by zenon and zenon Logic** table. Here you can also see if these ports can be amended in these programs.

PORT SETTINGS BY ZENON AND ZENON LOGIC

Application	Description	Ports	Transport log
stratonrt[k].exe	Logic Service polling communication and Logic Studio.	*1200-1210*	**TCP**
stratonrt[k].exe		*4500-4510*	**TCP**
stratonrt[k].exe	zenon Logic redundancy.	*7000-7010*	**TCP**
stratonrt[k].exe	Logic Service - spontaneous communication	*9000-9010*	**TCP**
zennetsrv.exe	zenon network service.	*1100-1100*	**TCP**

Application	Description	Ports	Transport log
zensyssrv.exe	zenon transport service.	▸ *1101*	**TCP**
zendbsrv.exe	zenon database service.	▸ *1103*	**TCP**
zenAdminsrv.exe	zenon administration service.	▸ *50777*	**TCP**
zenLogSrv.exe	zenon logging service.	▸ *50780*	**TCP**
CodeMeter.exe	Code Meter dongle service.	▸ *22350* (changeable but must not be changed)	**TCP**
WkSvW32.exe	WibuKey Network service	▸ *22347* (fixed)	**TCP**
Zenrt32.exe	**Message Control** with Voice over IP.	▸ *5060*: SIP ▸ *4000*: RTP ▸ *4001*: RTCP (fixed) SIP and RTP can be configured using Engineering Studio. RTCP is automatically set by the system.	**UDP**
zenHelpService.exe	zenon Help provisioning service	▸ *50790*	**TCP**

13 Installation of an older version after installation of zenon 11 (64-bit operating system)

If, on a 64-bit operating system, after installation of zenon 11, a version of zenon with a version number lower than 7.10 is installed, the 64-bit services of version 7.10 must be re-registered afterwards using the command line. Registration can be carried out using a batch file or manually.

Registration with a batch file:

1. Copy the file named **Register.bat** from the zenon installation medium.
2. You can find this in the following folder: *...\AdditionalSoftware\Register Admin Service and Log Service (x64)*

3. Execute the file on the respective computer as an administrator.

manual registration:

1. run the command line with administrative rights
2. Go to the folder *%ProgramFiles%\Common Files\COPA-DATA\zenAdminSrv*
3. Start the service **zenAdminSrv.exe** with the parameter **-service**
4. Go to the folder *%ProgramFiles%\Common Files\COPA-DATA\zenLogSrv*
5. Start the service **zenLogSrv.exe** with the parameter **-service**

> **Example**
>
> **zenAdminSrv:**
>
> - Folder: *C:\Program Files\Common Files\COPA-DATA\zenAdminSrv*
> - Command: **zenAdminSrv.exe -service**
>
> **zenLogSrv:**
>
> - Folder: *C:\Program Files\Common Files\COPA-DATA\zenLogSrv*
> - Command: **zenLogSrv.exe -service**

14 Installation of version 7.x and version 6.51 on the same computer

If a version 7.x is installed on a system that already has zenon 6.51 installed, the **Multiple Network Protocol Driver** must be reinstalled after a reboot.

FOR X64 SYSTEMS

For new installation:

1. Restart the system.
2. On the installation medium, open the following path: *AdditionalSoftware\COPA-DATA Multiple Network Protocol Driver*.
3. Execute the file called **MNDPx64Setup.bat**.

This means that the driver is reinstalled and properly linked to the network adapters.

FOR X86 SYSTEMS

For new installation:

1. Restart the system.
2. On the installation medium, open the following path: *AdditionalSoftware\COPA-DATA Multiple Network Protocol Driver*
3. Execute the file called **MNDPx86Setup.bat**

This means that the driver is reinstalled and properly linked to the network adapters.

15 Licensing preview versions

The following time-limited licenses are available for zenon preview programs. They have a fixed expiry date.
If a preview version is installed, it needs an internet connection. The licensing will fail if this is not present.
In this case, the time stamp can be updated manually.
To do this:

1. Open the command line.
2. Enter: *%programfiles(x86)%\CodeMeter\Runtime\bin\cmu32.exe --time-update*

The time stamp is updated and the license is valid.

16 System requirements

You can find information on operating systems supported by zenon in the **Desktop operating systems** (on page 43) and **Server operating systems** (on page 44) sections

Changes due to Service Packs, Hotfixes or Patches from Microsoft can cause incompatibilities and affect the functionality of the software. In this case, COPA-DATA will provide an updated version of zenon as soon as possible. In this case you can get more information from COPA-DATA support: support@copadata.com.

Attention

Note when configuring the project:

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

Note: According to Windows conventions, hostnames may not contain more than 15 characters.

16.1 Desktop operating systems

Supported desktop operating systems and minimum required Windows Service Pack/version:

Operating system	zenon Engineering Studio	Service Engine	zenon Smart Server	zenon Smart Client	zenon HTML Web Engine Web Visualization Service	Logic Service	zenon Report Engine
Windows 10 (Pro, Enterprise, IoT Enterprise)	Only x64 from version 1607	From version 1607	From version 1607	From version 1607	From version 1607	From version 1607	From version 1607
Windows 10 (Enterprise LTSC, IoT Enterprise LTSC)	Only x64 from version LTSC 2016	From version LTSC 2016	From version LTSC 2016	From version LTSC 2016	From version LTSC 2016	From version LTSC 2016	From version LTSC 2016
Windows 11 (Pro, Enterprise)	Only x64 from version 21H2	From version 21H2	From version 21H2	From version 21H2	From version 21H2	From version 21H2	From version 21H2

16.2 Server operating system

Supported server operating systems and minimum required Windows Service Pack/version:

Server operating system	zenon Engineering Studio	Service Engine	zenon Smart Server	zenon Smart Client	zenon HTML Web Engine	Logic Service	zenon Report Engine
Windows Server 2016 (All editions with the exception of Core)	Only x64 from version 1607	From version 1607	From version 1607	From version 1607	From version 1607	From version 1607	From version 1607
Windows Server 2019 (All editions with the exception of Core)	Only x64 from version 1809	From version 1809	From version 1809	From version 1809	From version 1809	From version 1809	From version 1809
Windows Server 2022 (All editions with the exception of Core)	Only x64 from version 21H2	From version 21H2	From version 21H2	From version 21H2	From version 21H2	From version 21H2	From version 21H2

16.3 System requirements when using DirectX

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

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

Parameter	Minimum requirements	Recommended
CPU:	Single core with SSE2 support.	Quad Core or more cores
Graphics adapter: (DirectX hardware only)	DirectX 11 mainstream graphics card. **Note:** When an integrated graphics chip is used in particular, it is possible, depending on the driver used, that there are impairments to the display quality.	Dedicated DirectX 11 AMD or nVidia high-end graphics card
Graphics memory: (DirectX hardware only)	1 GB VRAM **Note:** The size that is actually needed depends on the number of screens called up and the elements displayed.	2 GB VRAM
Graphics card driver: (DirectX hardware only)	The graphics card manufacturer's most recent driver.	
Operating system:	*DirectX Hardware* and *DirectX Software* only work on operating systems with *DirectX 11.1* support. If the system does not support *DirectX 11.1*, it is automatically switched to *Windows based* . The current *DirectX*-Service Engine must be installed. For zenon it is installed together with the setup. It must be manually installed for Smart Client.	

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

16.4 User authorization

Windows administrator rights are required for installation.

Standard Windows user rights are required for ongoing operation. The user account control (UAC) can be activated at the highest security level.

16.5 Hardware requirements

In this chapter, you can find the hardware requirements for the individual versions of Engineering Studio and Service Engine, as well as Smart Server and Smart Client.

Engineering Studio uses a Microsoft SQL Server as an SQL Server and has higher hardware requirements than Service Engine. If Engineering Studio and Service Engine are to be running on a system simultaneously, the requirements increase.

> **Attention**
>
> Graphics cards with their own graphics memory and DirectX support are recommended. Shared-memory graphics cards may require too much working memory and may thus lead to performance impairments. Note the system requirements when using DirectX (on page 45) chapter. The recommended configuration from this chapter is to be noted for the use of Multi-Touch.

PERFORMANCE OPTIMIZATION

Note that all information stated only constitutes the minimum requirements for your system. Better hardware equipment improves the performance of zenon considerably.

- Equip your hardware - both clients and most of all the server - with sufficient memory (RAM).
- Optimize the hardware for data backup, for example with fast SSD data storage.
- Match the hardware of the clients and the network to one another. A system is only as powerful as its weakest component.
- Optimize your network architecture, for example with the use of cabling with a high data transfer rate and corresponding devices (switches and routers).
- When using a virtual environment, ensure that the virtual system is correspondingly configured with performance optimization.

16.5.1 Engineering Studio

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

Hardware	Minimum requirements	Recommended
CPU	Single core with SSE2 support.	Quad Core
Memory	From 4 GB **Note**: The more projects you have simultaneously available in memory, the more memory you need.	8 GB
Harddisk	At least 13 GB free space for a complete installation plus additional space for the projects.	
Monitor resolution	Extended VGA with 1024 x 768 pixels. **Attention:** Some dialogs, e.g. the filter of the **AML/CEL** image, are difficult to operate or possibly non-operable at a height of less than 850 pixels.	Double monitor setup: 2 times *1920 x 1080*.
Graphics adapter	64 MB dedicated memory. Cards with shared memory can lead to performance loss. Note the system requirements when using DirectX (on page 45) chapter.	
Input devices	Standard keyboard or standard mouse.	
USB interface or DVD drive	For the installation, regardless of installation medium. The installation is also possible via network. Installation files can also be downloaded from the customer area of the COPA-DATA website.	
Parallel or USB interface	In case of dongle licensing required for dongle. For network dongle only required for the dongle server.	
Network connection (optional)	Recommended 10 MBit/s with TCP/IP protocol for Remote Transport, network dongle, project backups on central file server, multi-user	1000 MBit/s

Hardware	Minimum requirements	Recommended
	capable Engineering Studio, etc.	

Note: When using comprehensive multi-user projects, note the information in the hardware requirements chapter.

16.5.2 Service Engine

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

Hardware	Minimum requirements	Recommended
CPU	Single core with SSE2 support.	Quad Core
Memory	From 512 MB. **Note:** Projects with big amounts of data, Network projects, multiple projects simultaneously and projects in redundance mode need more memory.	4096 MB
Harddisk	2 GB of free memory space for Service Engine installation plus additional space for the projects. **Attention:** If you log historical data (e.g. Archive data or Alarm-/CEL-Data), you need sufficient harddrive space or you have to make sure during engineering that the historical data is evacuated or deleted.	
Monitor resolution	1024 x 768. Note also the information in the infobox under the table.	
Graphics adapter	64 MB dedicated memory. Cards with shared memory can lead to performance loss. Note the System requirements when using DirectX (on page 45) chapter in relation to this.	
Input devices	Keyboard and/or mouse. Operation via touchscreen is also possible. Many individual, customizable softkeyboards for the touchscreen are available for you. In addition, there is the possibility of Multi-Touch	

Hardware	Minimum requirements	Recommended
	operation.	
USB interface **(optional)**	▸ For installation. Installation also possible via network or other storage media. ▸ For dongle. Network dongle also available.	
Network connection (optional)	64 kBits/s for standard Client/Server projects. 100 Mbit/s full duplex for redundant operation.	1000 kBits/s full duplex for standard Client/Server projects.
Remote connection (optional)	Minimum requirements: Dial-up modem with 9600 Bit/s.	1 Mbit/s full duplex.
WAN connection (optional)	Any desired connection via router, e.g. per ISDN or DSL Data transfer is slower in a WAN than in a local network for technical reasons. Be sure to check the possible data transfer rates of your WAN technology already at the time when you create the project.	
Message Control (optional):	Please refer to chapter Message Control for the requirements.	
Interfaces (optional)	The necessary interfaces depend on the requirements of the PLC and/or the bus connection, for example serial RS232 or RS422/485 interfaces, ISA/PCI slots, etc.	

Information

The minimum recommended resolution in Service Engine is 1024 x 768 pixels. Smaller resolutions can also be configured. However it may then not be possible to operate some online dialogs. If these are not used, the resolution can be selected as lower.

16.5.3 Smart Server

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

Hardware	Minimum requirements	Recommended
CPU	Single core with SSE2 support.	
Memory	From 1024 MB	
Harddisk	256 MB free harddrive space.	1 GB free harddrive space.
Network connection	10 Mbit/s full duplex.	1000 Mbit/s full duplex.
Remote connection (optional)	Minimum requirements: Dial-up modem with 9600 Bit/s.	1 Mbit/s full duplex.

16.5.4 Smart Client

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

Hardware	Minimum requirements	Recommended
CPU	Single core with SSE2 support.	Quad Core
RAM memory	From 1024 MB	
Harddisk	64 MB of free space for Smart Client plus space for the projects.	80 GB free harddrive space.
Network connection	10 Mbit/s full duplex.	1000 Mbit/s full duplex.
Remote connection (optional)	Minimum requirements: Dial-up modem with 9600 Bit/s.	1 Mbit/s full duplex.
Graphics adapter	64 MB dedicated memory. Cards with shared memory can lead to performance loss. Note the	

Hardware	Minimum requirements	Recommended
	System requirements when using DirectX (on page 45) chapter in relation to this.	

17 Software and paths for installation and operation

Paths for zenon:

- Installation
- Engineering Studio
- Service Engine

> **Info**
>
> You can display many default paths with the help of the **set** command:
>
> - start the command line (enter **cmd** in the Windows start area)
> - enter command **set**
> - By pressing the **Enter** key, the default folder for Windows and zenon are displayed.

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

INSTALLATION

During installation, paths are set for:

- Engineering Studio
- zenon SQL folder

Only the paths for Engineering Studioand zenon SQL folder can be customized.
The setup needs administrator rights. This is also true for changing the installation paths.

REQUIREMENTS

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

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

- Engineering Studio

- Service Engine
- Smart Client
- Logic Service

Requirements	Engineering Studio	Service Engine	Smart Client	Logic Service
Microsoft .NET Framework 4.8	+	+	+	--
Microsoft SQL Server 2019 Express (with Report Engine Standard Edition)	+	--	--	--
Microsoft Visual C++ 2010 Redistributable	+	+	+	+
Microsoft Visual C++ 2013 Redistributable	+	+	--	--
Microsoft Visual C++ 2022 Redistributable	+	+	+	+
Microsoft .NET Core Hosting Bundle 6.0.3	+	+	+	+
CodeMeter Runtime Kit 7.40a	+	+	--	+
Microsoft Web Deploy 3.6	--	--	--	--
Visual Basic for Applications | VBA 7.1	+	+	+	--
Visual Basic for Applications Language Pack | VBA 7.1	+	+	+	--
COPA-DATA Multiple Network Protocol Driver	+	+	--	+

Note: **Microsoft Visual Studio Tools for Applications** (VSTA 2.0) is not required and is not installed during setup. It can be installed separately using the **Startup Tool**. To do this, the **VSTA_AddOn.exe** file must be on the system.

ZENON

The installation of Engineering Studio sets two paths:

- Engineering Studio:
  - 32-bit systems: *%ProgramFiles(x86)%\COPA-DATA\zenon Software Platform [Version]*
  - 64-bit systems: *%ProgramFiles%\COPA-DATA\zenon Software Platform [Version]*

  These paths can be customized during the installation.
- zenon SQL folder:
  - Version 10 and higher: %ProgramData%\COPA-DATA\SQL2019\

These paths can be customized manually via zenDB.ini.

ENGINEERING STUDIO

In Engineering Studio, the following paths are used by default:

Object	Path
Workspace	*%CD_USERDATA%* For example: *C:\Users\Public\Documents\zenon_Projects*
Projects	*%CD_USERDATA%* For example: *C:\Users\Public\Documents\zenon_Projects* **Hint for short cuts:** highlight the project -> **Ctrl+Alt+D**
SQL folder of the project	*%ProgramData%\COPA-DATA\[SQL-Ordner]\[GUID]\FILES* **Hint for short cuts:** highlight the project -> **Ctrl+Alt+E**
project.ini	*%ProgramData%\COPA-DATA\[SQL folder]\[GUID]\FILES\zenon\system*
zenon6.ini	*%ProgramData%\COPA-DATA\System* For example: *C:\ProgramData\COPA-DATA\System*
Backup	*%ProgramData%\COPA-DATA\[SQL folder]\GUID]\BACKUP* *%ProgramData%\COPA-DATA\[SQL folder\GUID]\FILES\[Projekte]*
Compiled files	*%CD_USERDATA%\[Workspace]\[Projekt]\RT*
External files	*%CD_USERDATA%\[Workspace]\[Projekt]\RT\FILES\...* **Note:** Can be set in the project using the file storage property.
System files	Windows system folder.

SERVICE ENGINE

In Service Engine, the following paths are used by default:

Object	Path
Projects	*%CD_USERDATA%\[Workspace]\[Projekt]\RT*

Object	Path
External files	*%CD_USERDATA%\[Workspace]\[Projekt]\RT\FILES\...*
Exported archives, Chronological Event List and Alarm Message List	*%CD_USERDATA%\[Workspace]\[Projekt]\Export* **Note:** Is created at the first export.
System files	Windows system folder.

ZENON LOGIC

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

18 Smart Server

To install Smart Serveror Smart Server Pro:

1. Activate the WWW services on the computer.

   Folder *C:\inetpub\wwwroot* must exist.
2. Start the zenon installation medium. The start screen is displayed

   If you have deactivated the autostart feature, execute **start.exe** from the installation medium.
3. Select Smart Server.
   The 32-bit or 64-bit version of Smart Server is installed automatically according to the version of the operating system.
4. Follow the installation routine.
5. Restart the computer.

The setup files for the web client can be found after installation in subdirectories of the Smart Server installation path.
For example: *C:/Programs (x86)/COPA-DATA/zenonWebserver/zenon/controlversion***/SmartClientStandalone.exe**
or

*C:/Inetpub/wwwroot/zenon/controlversion***/SmartClientStandalone.exe**

The example web pages (index*.htm und init*.htm) are also installed. They can be found in the zenon subdirectory of the Smart Server installation path.
For example: *C:/Programs/zenon Web Server/zenon/index.htm*
or
C:/Inetpub/wwwroot/zenon/index.htm

Information

The service for Smart Server is only started automatically in the licensed version. In demo mode, Smart Server must be started manually via the Smart Server console in the system properties.

ADDITIONAL INFORMATION

You can find details on Smart Server in the Smart Server and Smart Server Pro manual, and details on licensing in the Licensing manual.

19 Smart Client

The Smart Client is mainly an ActiveX control displaying the information in a browser. The display is 1:1 like in Service Engine client. The connection to the Service Engine server is established via Smart Server using TCP/IP communication.

You can find the setup files for Smart Client after installation in subdirectories of the Smart Server installation path (xxx stands for the respective version of zenon), for example:
%Programfiles%/COPA-DATA/Smart_Server/zenon/controlversions/Versionxxx/zenon_Webclient_Setup_ENGLISH.EXE
or

C:/Inetpub/wwwroot/zenon/controlversions/Versionxxx/zenon_Webclient_Setup_ENGLISH.EXE

All zenon Logic Web Client setups are digitally signed and can also be provided for download from the Internet without any problems.

The Smart Client Starter is also installed with Smart Client. This makes it possible to open Smart Client from any desired browser.

Info

<Smart Client sends error and LOG files. The application Diagnosis Server (necessary for the evaluation of these files) is included in the installation of Smart Client.

20 Version changes and updates (build setups)

In zenon, you can change to new versions for example, from 10 to 11. Or you can install updates within a version.
New versions can be installed in parallel with existing versions. They mainly offer new features.
Updates are provided in the form of build setups. Updates change a previously-installed version. They mainly fix bugs.

With each setup for new build or new versions, you receive a link to the COPA-DATA website with the changes between the previous version and the newly-installed one.

UPDATE (BUILD SETUP)

An update only updates those files which are more current than the previously installed files. All projects and individual settings will remain unchanged. Build setups can contain changes for all installed components. When calling up the setup, the components that have been changed and the version to which they have been changed are shown.

Build setups are never 100% quality assured. Only the bug fixes are tested. If unwanted side effects should occur because of a bug fix, it might be possible that these side effects will not be noticed during testing.

Note: Build setups can only be installed locally. Installation on network paths (UNC) is not possible.

VERSION CHANGE

If you want to install a new version of zenon, start the installation routine. The new version is being installed parallel to the old one. All projects and individual settings will remain unchanged. Projects aren´t converted to the new version during installation. The respective project is converted when it´s being opened for the first time in Engineering Studio. A dialog box notifies you about this procedure. The old version is automatically backed up. If you want to use only the most up-to-date version, use the Windows control panel software deinstallation routine to remove the old version.

Attention

If an installation involves changing the SQL server (for example, from zenon 10 to zenon 11 or higher), you must back up all projects or the workspace must be backed up before the installation. This backup is read back after the installation. For details see also section **Multi-user projects/Update with change of SQL servers**.

If you want to use multiple versions of zenon simultaneously, you have to manage them using the **Startup Tool**. You can start only one version at a time. You can select which version you want to run using the **Startup Tool** that automatically adjusts all necessary settings. You can find details in chapter **Startup Tool**.

Information

New versions always bring about structural changes. Projects and settings remain untouched during installation. If you open Engineering Studio for the first time, projects are converted to the new version. Simultaneously, an automatic backup of the old version is created.

Converted projects cannot be edited in legacy versions. From version 6.2 on, Engineering Studio is able to create projects for different Service Engine versions.

Important tips for converting projects can be found in the revision text and in the **Project conversion** manual.

MULTI-USER PROJECTS

To ensure a change to a new zenon version in multi-user projects without data loss:

1. Check in all checked out elements on all Clients by clicking *Apply changes*.
   Nothing must be checked out. This is true for all projects.
2. Install the new zenon version on the server computer.
3. Convert all server projects to the new version.
   To do this, load each project on the server computer into Engineering Studio and accept the conversion.
4. Install the new zenon version on the client computers.
5. Load the projects to the clients.

CHANGE THE SQL SERVER

If an installation involves changing the SQL Server (e.g. from zenon 7.00 to zenon 11), additional steps are needed.
These steps are carried out:

- after all projects are checked in
- before the new version is installed

Procedure when changing the SQL Server:

1. Check in all checked out elements on all Clients -> *Apply changes*.
2. On the multi-user server, open Engineering Studio in the original version.
3. Create backups of all projects which you want to edit or open with the new version:
   - either as single project backups
   - or as backup of the complete workspace

4. Install the new version on the Server.
5. Convert all Server projects to the new version by loading the previously created project backups one time in Engineering Studio.
6. Install the update on every Client.
7. Transfer the projects from the multi-user Server to the Clients
   Keep in mind the new name of the SQL instance.

The projects are converted, synchronized and ready for use

Attention: Make sure that the settings of the firewall allows the data traffic between the multi-user Server and the Clients.

20.1 Compatibility

Compatibility in zenon concerns:

- Service Engine: Cooperation of different Service Engine versions.
- Engineering Studio: Up-converting existing projects to new Engineering Studio versions.
- Engineering Studio: Creating Service Engine files for different Service Engine versions in Engineering Studio.

SERVICE ENGINE

Service Engine online compatibility enables Service Engine systems to work together in the zenon network, as well as via Smart Clients.

The following is applicable here: The version of the client Service Engine must be the same or higher than the version of the server Service Engine.
For example:

- An 8.20 client can work together with an 8.10 server.
- An 8.00 client does not work together with an 8.10 server. In this case, the client Service Engine must be upgraded to version 8.10 or higher.

The current Service Engine can load projects of the following versions:

- *6.20 SP4*
- *6.21 SP0*
- *6.21 SP1*
- *6.22 SP0*
- *6.22 SP1*
- *6.50 SP0*

- *6.51 SP0*
- *7.00 SP0*
- *7.10 SP0*
- *7.11 SP0*
- *7.20 SP0*
- *7.20 SP0[current build no.]*
- *7.50 SP0*
- *7.60 SP0*
- *8.00 SP0*
- *8.10 SP0*
- *8.20 SP0*
- *10:00:00*
- *11:00:00*

Due to the multi-project administration, projects from different versions can be loaded. For example, the integration project can be from version 8.20, a subproject from version 8.10 and another subproject from version 7.60.

ENGINEERING STUDIO

Engineering Studio can open projects from the previous versions in each new version. These can be edited further in the new version. If adjustments are required by the user, information can be found in the current revision text and in the **Project conversion** manual. When opening a project with a lower version number in a higher Engineering Studio version:

- the project is automatically converted
- a backup of the project is automatically created

Attention

There is no backward compatibility between Engineering Studio versions. Backward compatibility is only ensured between Engineering Studio and Service Engine

That means:

- Converted projects can no longer be opened in a Engineering Studio with a lower version number.
- The project backup created during conversion can still be opened and edited.

Also avoid transferring projects via XML import from newer to older versions. This can lead to undesirable results. Drivers in particular can perform differently than expected.

COMPATIBILITY BETWEEN ENGINEERING STUDIO AND SERVICE ENGINE

With Engineering Studio, Service Engine files can be created for different versions of Service Engine. The Service Engine version therefore does not need to correspond to the Engineering Studio version. This backward compatibility is particularly suited for use of mixed systems.

For example: A project that has been configured with Engineering Studio version 10.00 and compiled for 8.00, can also be started with Service Engine8.00.

Attention

If possible, the same version of Engineering Studio and Service Engine should always be used. Configurations of properties that are not available in older versions can lead to unwanted results in older versions of Service Engine.

CREATE SERVICE ENGINE FILES

To create Service Engine files for earlier versions in Engineering Studio:

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

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

ERROR CREATING SERVICE ENGINE FILES AND MICROSOFT OFFICE 365

In certain configurations, an error may occur when creating Service Engine files:

- You can create Service Engine files on a computer with:
  - Windows 10
    operating system and
  - Office 365.
- The creation of Service Engine files has failed and is ended with an error message.

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

Solution:

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

The creation of Service Engine files is possible again.

XML

Data exported in XML is then available for import into later Engineering Studio versions.

Exception: If data from the RGM is saved in Service Engine directly as an XML file using the export function, this cannot be reimported.

Recommendation: Avoid transferring projects via XML import from newer to older versions. This can lead to undesired events in Engineering Studio and Service Engine.

21 FAQ

Errors during the installation mostly occur when the replacement or creation of files is prevented by a virus scanner or by existing installations. Here you can find the most frequent reasons for installation errors and their solution.

ZENON

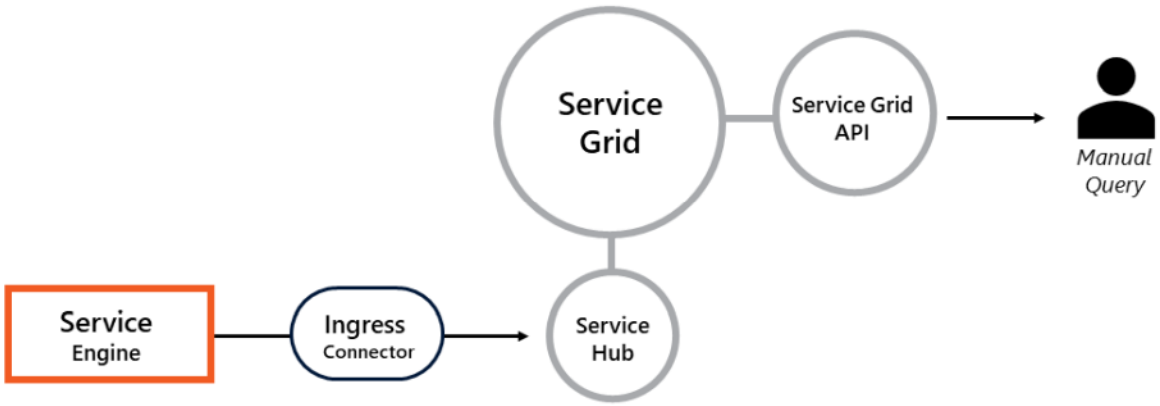
Problem	Solution
Installation is terminated. Typical error message: **Error 1304. Error writing to file...**	Deactivate the virus scanner. Close unnecessary programs.
Demo projects were installed but are not displayed. New projects cannot be created.	Check the computer name. The computer: ▸ must not consist of more than 15 characters ▸ must be in accordance with the convention of the NetBIOS computer name
Error message that a service cannot be started.	▸ first reboot the computer ▸ then start the zenon setup again

SQL SERVER

GENERAL

Problem	Solution
The installation is unsuccessful because the password does not meet the requirements.	The installation of SQL Server is not possible if the security requirements do not permit a password length of 20 characters (A-Z, a-z, 0-9 and special characters). Each character can only be used once.

22 Getting Started Guide (Windows native)



Test environment: Querying of process data of a Service Engine via Service Grid API.

In this guide, you will install and configure a simple use case for Service Grid in a test environment. With the necessary prior knowledge (on page 92), you can work through this guide in 3 to 4 hours.

INSTALLATION OPTION

This guide only refers to the Service Grid (Windows-native) installation option.

The following applies for Service Grid (Windows native):

- Host operating system is Windows
- Service Grid is installed as a native Windows application

You can find further information and a comparison of all supported Service Grid installation options in the Help.

LEARNING OBJECTIVES

After working through this guide, you can:

- Initialize Service Grid
- Configure a zenon project for data transfer with Service Grid
- Query variable values of the zenon project using the Service Grid API

In doing so, you will have mastered the basic functions of Service Grid in a test environment.

OTHER USE CASES

All supported use cases for Service Grid are documented in the Help.

22.1 Test environment vs. productive environment

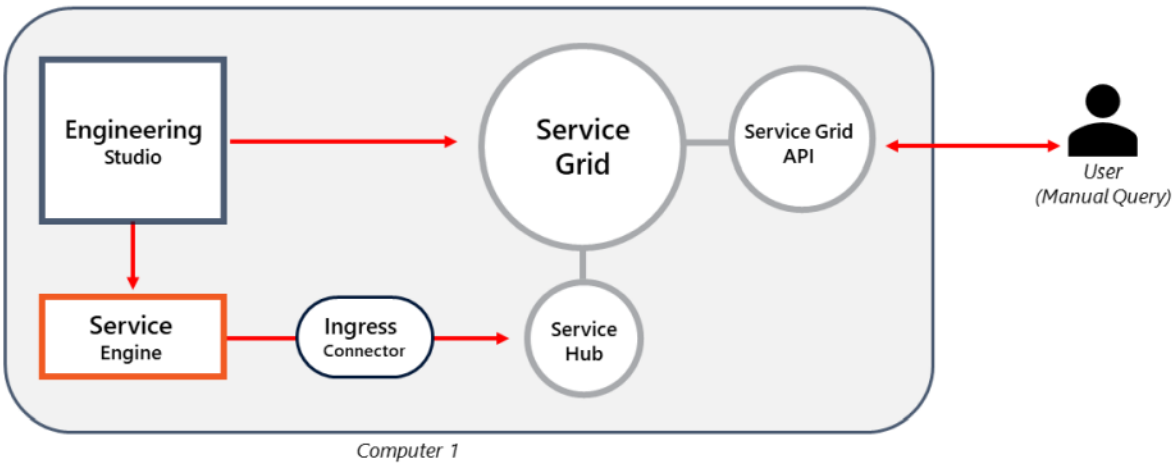
The test environment described in this guide is quicker and easier to set up than a typical productive environment.

The basic differences are:

	Test environment	Productive environment
Installation option	▸ Service Grid (Windows native)	▸ Service Grid (Windows native) ▸ Service Grid (Docker on Linux)
Number of computers	▸ 1 computer for Service Grid and all clients	▸ 1 computer for Service Grid ▸ Dedicated computers for clients

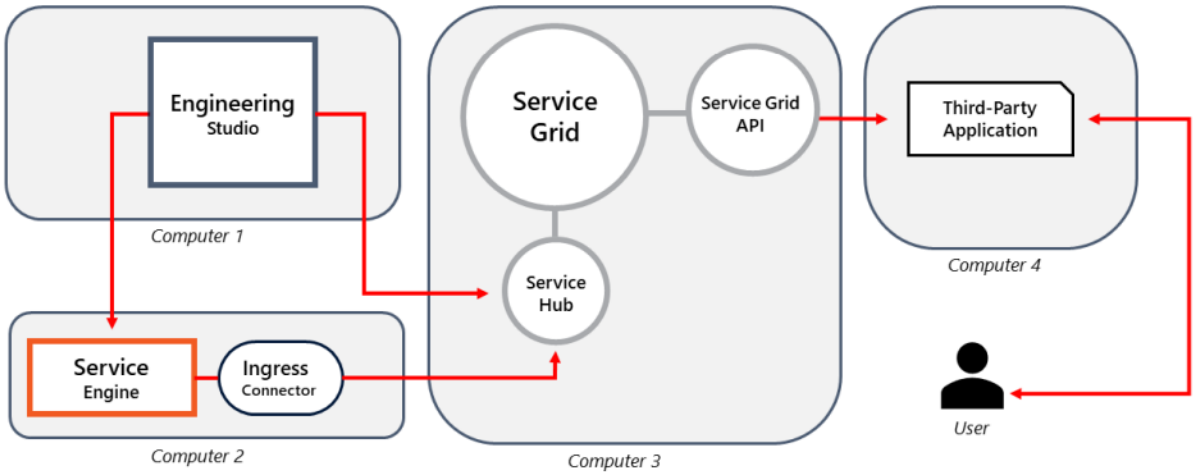
	Test environment	Productive environment
Network topology	All applications run on the same computer.	The applications run on different computers. The computers can be distributed over different remote locations.
Passwords	It is possible to use predefined passwords in a protected test environment.	For all logins, it is essential that you assign your own secure passwords.

TEST ENVIRONMENT



Test environment: Service Grid and all clients are installed on the same computer. Service Grid API is queried by the user using a manual query.

PRODUCTIVE ENVIRONMENT



Productive environment: Each application is installed on a dedicated computer. Service Grid API is automatically queried for each third-party application.

22.2 Necessary prior knowledge

You require prior knowledge of zenon in order to use this guide.

You must know:

- How to create a zenon project
- How to start a project in Service Engine
- How to define variables and modify variable properties

Knowledge in these areas is advantageous (but not required):

- Use of a REST API
- HTTPS certificates and certificate infrastructures

In general, the following applies: You require knowledge of all applications that you would like to connect with each other using Service Grid.

22.3 Prepare the system

Before you can install Service Grid, the test environment must be prepared properly.

22.3.1Test computer

The test computer for Service Grid must meet the following requirements:

1. Sufficient resources for the smooth operation of all installed applications (CPU, RAM, memory)
2. Functioning internet connection
3. No other software should be installed yet on the computer except for the operating system
4. In particular, no zenon applications may be installed yet.

Recommendation: For this guide, use a dedicated test computer with a newly setup operating system.

22.3.2 Test VM

You can install Service Grid both on a physical test computer as well as in a virtual machine (test VM).

The following particular features must be noted when installing in a test VM:

- zenon demo licenses have significantly shorter license periods on a VM than on a physical computer
- The additional virtualization level of the VM can make troubleshooting more difficult

Secure operation of Service Grid is guaranteed both on a test computer as well as in a test VM.

22.3.3 DNS & network ports

Service Grid is a networked system. The services communicate with one another and with other computers.

The following network addressing is supported:

- Naming resolution: Domain Name System (DNS)
- Hostname: Fully Qualified Domain Name (FQDN)

The use of DNS and FQDN ensures that all services both in LAN and in WAN (internet) can be addressed securely.

These hostnames are explicitly not supported:

- "localhost"
- All hostnames that are not FQDN

Reason: Because the reliable addressing of services is not ensured.

The following applies for network ports:

- All network ports for Service Grid are documented.
- Make sure that these ports are not being used otherwise in your network or are not blocked, for instance, by firewalls.

Note: If required, you can reconfigure network ports.

22.3.4 Operating sytem and language settings

For the host operating system, you require:

- Windows 10, 64-bit (with the latest updates)

This guide uses English for:

- Windows operating system
- Language settings in the browser
- Service Grid
- All other applications

Recommendation: Configure your test computer accordingly.

22.3.5 Browser

You need a web browser to access the web interfaces of Service Grid.

The following web browsers are supported:

- Google Chrome
- Mozilla Firefox
- Microsoft Edge
- Apple Safari

Always use the most recent version of the respective browser.

22.4 Determine configuration values

In this chapter, you will compile all configuration values you need for the initialization and configuration of Service Grid.

Please note the following for working with configuration templates:

1. **System-specific values**: Must always be determined individually for your system (e.g. hostname, project ID). You may not apply the sample values.
2. **Non-system-specific values**: Can be defined by you (e.g. usernames, passwords). In a protected test environment, you can also apply the suggested sample values.

You must always assign your own passwords in productive environments or unprotected test environments.

22.4.1Own template file

It is good idea to create your own template file for the configuration. A simple text file will do here.

It has the following advantages:

- You can look up all the configuration values in one place.
- You can apply configuration files using copy & paste.

The configuration template must be protected from unauthorized access.

22.4.2 Bookmarks for web interfaces

Create boomarks in the browser for the following URLs:

Name	Sample values	Description
Identity Service	*https://mycomputer.mydomain.com:9443/identity-service* System-specific value*	The **Identity Service** web interface allows each user to manage certain settings of their own user account.
Service Grid Studio	*https://mycomputer.mydomain.com:9443* System-specific value*	The **Service Grid Studio** interface allows exclusively users with administrator privileges to fully administer Service Grid.

* You must replace mycomputer.mydomain.com in the URLs with the hostname of your computer (on page 69).

22.4.2.1 Determine hostname

To determine the actual hostname of a Windows computer:

1. Open a Windows command prompt.
2. Execute the following command:
   ping localhost
3. The command line interface shows your computer's actual hostname. This is usually a **Fully Qualified Domain Name (FQDN)**.
4. Examples of actual FQDNs are:
   - *MYCOMPUTER.mydomain.com*
   - *MyComputer.mydomain.com*
   - *mycomputer.myddomain.com*
5. Convert the actual hostname to lowercase letters: *mycomputer.mydomain.com*

You have thus determined the hostname you need for use in Service Grid.

22.4.3 zenon Project

In the test environment, you use Service Grid API to access variable data in a zenon project.

DETERMINE PROJECT PARAMETERS

You need the following values:

Parameters	Sample value	Description
Project	*ZENON10_DEMO*	This project is installed by default with

Parameters	Sample value	Description
Name		Engineering Studio. You also have the option to use a different project of your choice.
Project ID	***a0f4d8f9-c009-41d5-bc30-457dd92f6a29*** System-specific value: You must determine the project ID (on page 70) yourself.	The unique identification number of your zenon project.
Variable	*ALC_GLOBAL_GROUND*	This variable is contained in the *ZENON10_DEMO* project. You also have the option to use a different variable of your choice.

Note: You can only determine these parameters once the zenon installation has been completed (on page 97).

22.4.3.1 Determine project ID

The Project ID is individual for each zenon project. It is an automatically generated ID.

It is based on the following schema:

a0f4d8f9-c009-41d5-bc30-457dd92f6a29

To determine the project ID:

1. In Engineering Studio, select the demo project of zenon.
2. Select the main project node in the project.
3. Press the **Ctrl+Alt+E** shortcut.
   This opens the file path of your zenon project on your hard disk.
4. Select the superordinate folder name.
   Reason: The Project ID is identical to the superordinate folder name.
5. Open the context menu by right clicking on *Rename* with the mouse.
6. You can then copy the Project ID to the clipboard and use it again as you wish.

You have thus determined your Project ID. You can use this ID to uniquely address the project using the Service Grid API.

22.5 Installation

For the installation of Service Grid 11.2, you must strictly adhere to the order of individual installation steps set out in this guide. If you deviate from this order, it can lead to malfunctions.

The right order of installation is:

1. Installation of Service Grid 11.2 (Windows native)
2. Installation of zenon 11 platform setup
3. Installation of the most recent zenon build update.
4. Installation of Service Grid Gateway 11.2

This order of installation ensures that Service Grid 11.2 will work properly.

22.5.1Download

You can download all the installation files for Service Grid for free in the download section of the COPA-DATA website. This requires a one-time registration free of charge.

Download the following files:

1. Service Grid 10.4 (Windows native)
2. zenon 10 platform setup
3. Both installers for Service Grid Gateway 10.4:
   - x86 installer
   - x64 installer
4. Service Grid build update

You have thus downloaded all the required files for setup.

22.5.2 Service Grid 10.4

There is a separate installer for the setup of Service Grid 10.4.

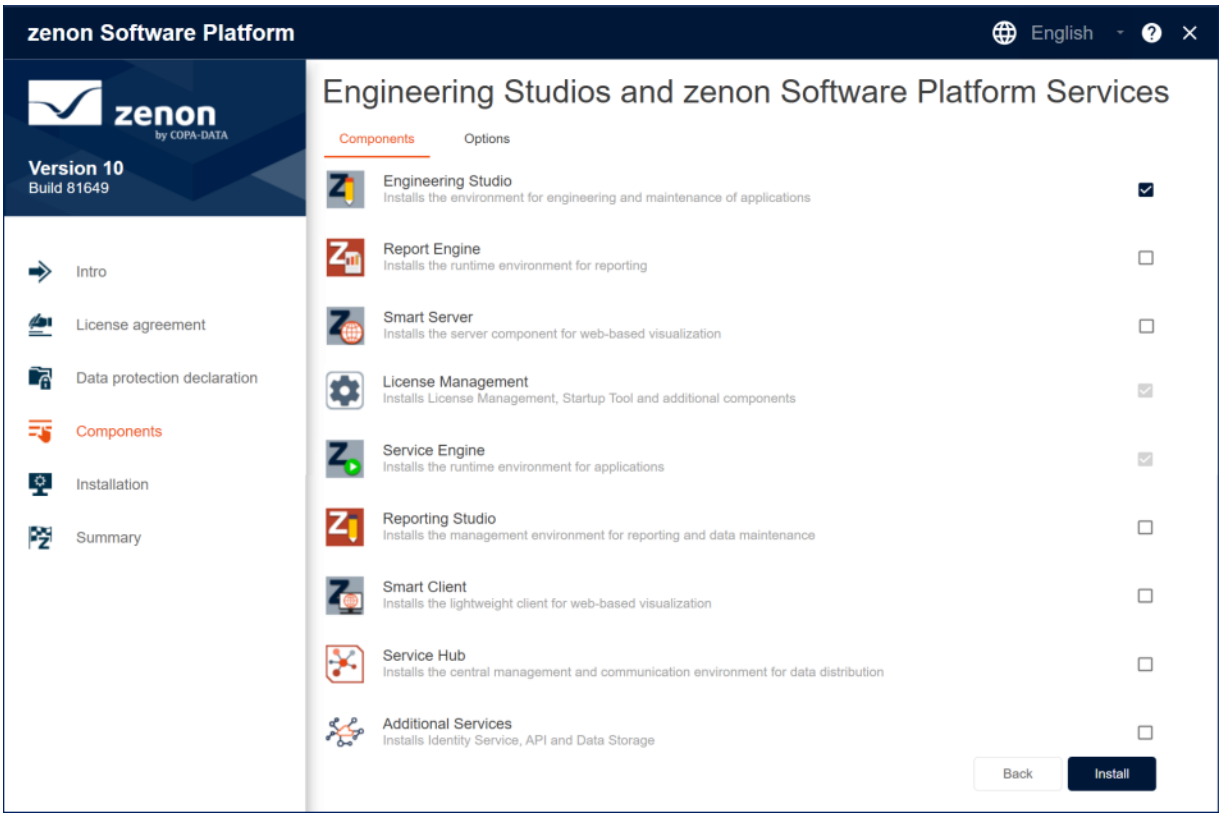
To install < NAME_SERVICE_GRID >:

1. It is absolutely essential to ensure beforehand that no zenon applications are installed on the test computer.
   Reason: Existing zenon applications can cause conflicts with the services of Service Grid.
2. Double-click on the file to start the installer for Service Grid.
3. Click on the **Install** button. Setup will then run through automatically.

4. After installation is successful, close the program window.
   Note: You cannot activate the license or configure Service Grid until later.

You have thus completed the first part of the Service Grid installation.

22.5.3 zenon platform



The test environment may only be installed with the highlighted components.

For the test environment, you must install zenon applications on the same test computer as Service Grid.

To install zenon:

1. It is absolutely essential that you ensure that Service Grid 10.4 has already been installed (on page 71) beforehand.
   Note: This is required for a functioning setup.
2. Mount the ISO file for the zenon platform setup.
3. Start the platform setup.
4. Click the **Customize** button in the **Engineering Studios and zenon Software** selection category.

5. Deselect all applications except Engineering Studio.
   Note: Service Engine and required additional applications such as **Licence Management** remain preset (checkbox highlighted in gray).
6. Make sure that only Engineering Studio is selected (blue checkbox).
7. Perform the installation.

You have thus installed zenon on the test computer.

22.5.3.1 Build update

You use Build update to update the zenon platform.

To install Build update:

1. Ensure that the setup of the zenon platform (on page 72) has been completed beforehand.
   Important: No applications of the platform may be open.
2. Open Build update.
3. Start the installation by clicking the **Install** button.
4. When installation is complete, close the window of the application.

You have thus installed the latest build of zenon.

22.5.3.2 Service Grid Gateway

Service Grid Gateway ensures compatibility between Service Grid and connected zenon clients.

To install Service Grid Gateway:

1. Make sure beforehand that the build update has been completed (on page 73).
2. First start the installer for Service Grid Gateway (x86)
3. Then start the installer for Service Grid Gateway (x64)
   Note: You always require both versions (x86 and x64).

You have thus installed Service Grid Gateway.

22.5.3.3 Activate demo license

You need licenses to run Service Grid. You can use the provided zenon demo license in a test environment.

The following applies for the demo license:

- You can work through the entire test case with the demo license

- You must manually activate the demo license
- The term of the demo license depends on the computer platform selected
  - Physical test computer: *30-day* term
  - Test VM: *1-day* term

To activate the demo license:

1. Open the **zenon Startup Tool**
2. Switch to the **Tools** tab
3. Start the **Licence Manager**
4. Click on **Advanced options...**
5. Click on **Advanced license administation...**
6. Activate the demo license.

A valid activated license is required for the operation of Service Grid.

22.5.4 Monitor services

All services of Service Grid are automatically started by the operating system.

To monitor the status of the services of Service Grid:

1. Open the command line using the *Windows+R* shortcut.
2. Enter *services.msc*.
3. Confirm by clicking the Enter key. This then opens the console for the administration of services.
4. You can find the services under: *zenon Service Grid <servicename>*
5. In general, all zenon Service Grid services must be running in the *running* status.
   Exception: **Data-Storage** may be in the *exited* status (because it is not used).

Note: After changes to the configuration, it may be necessary in exceptional cases to restart services manually.

22.6 Create initial user (administrator)

You must create the first user account in **Identity Service** yourself.

To create the initial user:

1. Open the web interface for **Identity Service** in the browser using the bookmark.
2. Confirm the HTTPS certificate warning shown.
   Note: You cannot install the root certificate until later.

3. You receive the following message from **Identity Service**:
   Please create the initial user. This user will have administrator permissions.
4. Assign the User Credentials for the Initial User:
   - "*Administrator*" (sample value for **Username**)
   - "*Changeme123!*" (sample value for **Password**)
     Important: You must assign your own secure **Password** in productive systems or unprotected test systems.
5. Confirm your entries. You are then forwarded to the login for the **Identity Service**.
6. Log in.

You have thus created the initial user account and logged in to the web interface of the **Identity Service**. You are now logged in for all web interfaces in Service Grid.

22.7 Set up HTTPS trust

Set up a HTTPS trust using the Root Certificate.

A HTTPS trust is needed in Service Grid for:

- All client computers that connect with Service Grid.
- The computer with the Service Grid installation.

Not having a trust has the following consequences:

- Service Grid web interfaces: Display certificate warnings. You can confirm the certificate warning and establish the connection manually.
- Client applications: Do not display certificate warnings. There is no option to establish the connection manually.

You can find further information on certificates and trusts in the Service Grid Help.

INSTALL ROOT CERTIFICATE

You must install the Service Grid Root Certificate on the test computer.

To download the certificate file:

1. Open the Service Grid Studio web interface using the bookmark. The HTTPS connection is currently shown as insecure.
2. Confirm the HTTPS certificate warning.
   Note: You cannot install the root certificate until later.
3. Go to the **Hub Controller** menu item.
4. Go to the **Certificates** subpage.

5. Click on the **Download CA Certificate** button.

You have thus saved the certificate file on your local computer.

To install the certificate:

1. Open the Windows **Run** dialog with the *WIN+R* keyboard shortcut.
2. Execute the following command at the command line interface:
   certlm.msc
3. This opens the Management Console (MMC) with the Windows Certificate Manager.
4. Go to this folder:
   Trusted Root Certification Authorities\Certificates.
5. Right-click on the *Certificates* folder. This opens the context menu.
6. Select the following option from the context menu:
   All Tasks\Import...
7. Import the certificate file.

You have thus installed the certificate and set up the HTTPS trust.

CHECK THE HTTPS TRUST

To check the HTTPS trust:

1. Restart your browser.
2. Open the web interface of Service Grid Studio.
3. The HTTPS connection is shown as secure. You no longer receive any certificate warnings.

You have thus checked the HTTPS trust.

22.8 Create Client Certificate Bundles

You must create several Client Certificate Bundles (CCBs) with the Service Node Configuration Tool. Each client needs an individual CCB.

A missing CCB has consequences:

- The client can no longer connect to Service Grid.
- In applications, you cannot configure a connection to the Service Hub (in Service Engine for example).

The CCB is a requirement for the setup of an encrypted network connection to the **Service Hub**. The **Service Hub** does not support unencrypted network connections.

START SNCT

This is how you start the Service Node Configuration Tool:

1. Check in advance whether Service Engine and/or Engineering Studio are registered on the computer.
   Reason: In the case of a registered **Smart Client**, **SNCT** does not recognize Service Engine nor Engineering Studio.
2. Open the **zenon Startup Tool**.
3. Click on the **Tools** tab.
4. Select the **Service Node Configuration Tool** entry.
5. Click on **Start**.

This opens the **SNCT**.

CREATE CLIENT CERTIFICATE BUNDLES

This is how you configure Client Certificate Bundles:

1. Ensure in advance that the HTTPS trust setting for the test computer (on page 75) has been set up.
2. In the **SNCT**, select all displayed client entries under **Selection**.
3. Click on the **Next** button.
4. Under **Connection** in the **Host name or address** field, enter the host name of the test computer:
   [mycomputer.mydomain.com]
   Note: Enter the host name using only lowercase letters!
5. Use the pre-set port:
   9410
6. Enter **Username** and **Password** for the login to **Service Hub**.
   Note: To do this, you need an **Identity Service** user account with the *Service Hub Administrator* user role. This user role is assigned by default to all users of the *Administrators* group.
7. Click on **Execute**.
8. Check under **Retrieval** if the following confirmation has been displayed:
   "Successfully wrote configuration for all selected products."
9. Close the **Service Node Configuration Tool** with the **Close** button.

You have thus configured the Client Certificate Bundles.

22.9 Engineering Studio

You must configure Engineering Studio for the data transfer between Service Engine and Service Grid.

22.9.1Configure Service Hub

This is how you configure the connection to the Service Hub:

1. Start Engineering Studio.
   Note: If Engineering Studio is already running, you must stop and restart it.
2. Select the zenon project.
3. Select the **Network** property group in the project properties.
4. Go to property group **Service Grid - General**.
5. In the **Service Hub** drop-down menu, select the entry for the computer on which Service Grid is installed.
   Note: Entry is only visible if a CCB has been created for Engineering Studio.
6. Activate the **Execute Service Grid Ingress Connector** property.
   Note: This checkbox is only visible if the connector is licensed.

You have thus configured Engineering Studio for the connection with the **Service Hub**.

22.9.2 Release variable

Service Engine can exchange data with Service Grid in different ways. The main method of data exchange is variables.

To configure Service Grid access rights for a variable:

1. In Engineering Studio, select the following **variable** in the **project manager**:

   ALC_GLOBAL_GROUND
2. Open the properties of the variable.
3. Go to property group **Authorization/eSignature**.
4. Go to **Service Grid settings**.
5. Select the *write-protected* entry from the drop-down list of the **Access permission** property.
   Note: This setting protects the *ALC_GLOBAL_GROUND* variable from write access from Service Grid.

You have thus assigned Service Grid access rights for this variable.

22.9.3 Starting Service Engine

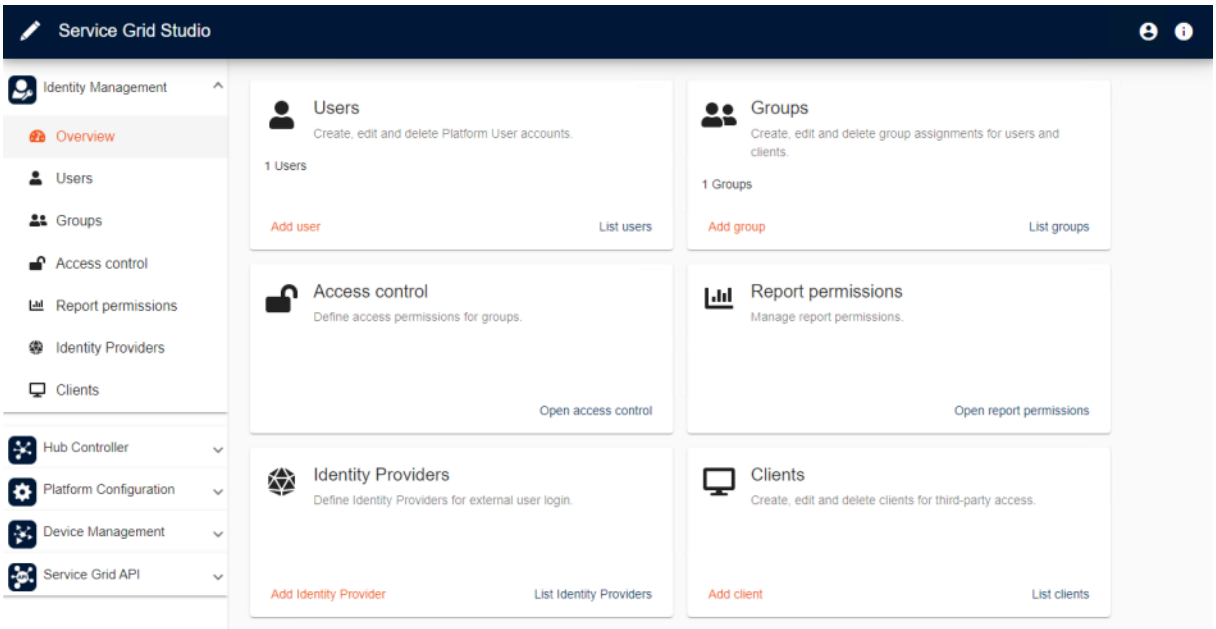
After completing configuration, you can start Service Engine.

To start Service Engine:

1. Save the project with all the changes.
2. Create all Service Engine files.
3. Start Service Engine.

Service Engine is now ready for the data transfer with Service Grid.

22.10 Identity Management



In Identity Management, you administer users, groups, resources and privileges.

Assign the Administrator user the privilege to access Service Engine via Service Grid.

To do this, the following configurations are necessary:

- You must create the new *Users* user group.
- You must assign the *Users* user group to the already existing *Administrator* user.
- The zenon project is a resource in Service Grid. You must assign this resource to the *Users* user group.
- You must now assign the **Service Grid API – Data Read** role to the resource.

The *Administrator* thus has read access to released variables and variable values in Service Engine via Service Grid API.

22.10.1 Create group and add users

To create a group:

1. Go to:
   - **Service Grid Studio** web interface
   - **Identity Management** menu entry
   - **Groups** submenu
2. Click in the middle column on the **Create Group** button.
3. In the pop-up, enter the following group name: *Users*
4. Click on **Add**.
   The group will then be displayed in the middle column.

You have thus created the *Users* group.

To add a user to the group:

1. Select the *Users* group.
2. Click on the **Add user** button.
3. Select the *Administrator* user.
   Note: This user is displayed by default in the list as *admin admin*.
4. Click on **Add**.

You have thus added the *Administrator* user of the *Users* group.

22.10.2 Add resource and add role

To add the Service Engine resource to the Users group:

1. Make sure beforehand that Service Engine is running.
2. Open the web interface of **Service Grid Studio**.
3. Go to the **Identity Management** menu item.
4. Open the **Access Control** submenu.
5. Select the *Users* under **Groups**.
6. Click on the **Add Resources** button.
7. Select the project: *ZENON10_DEMO*.
8. Click on the **Add** button.

You have thus added the resource to the user group.

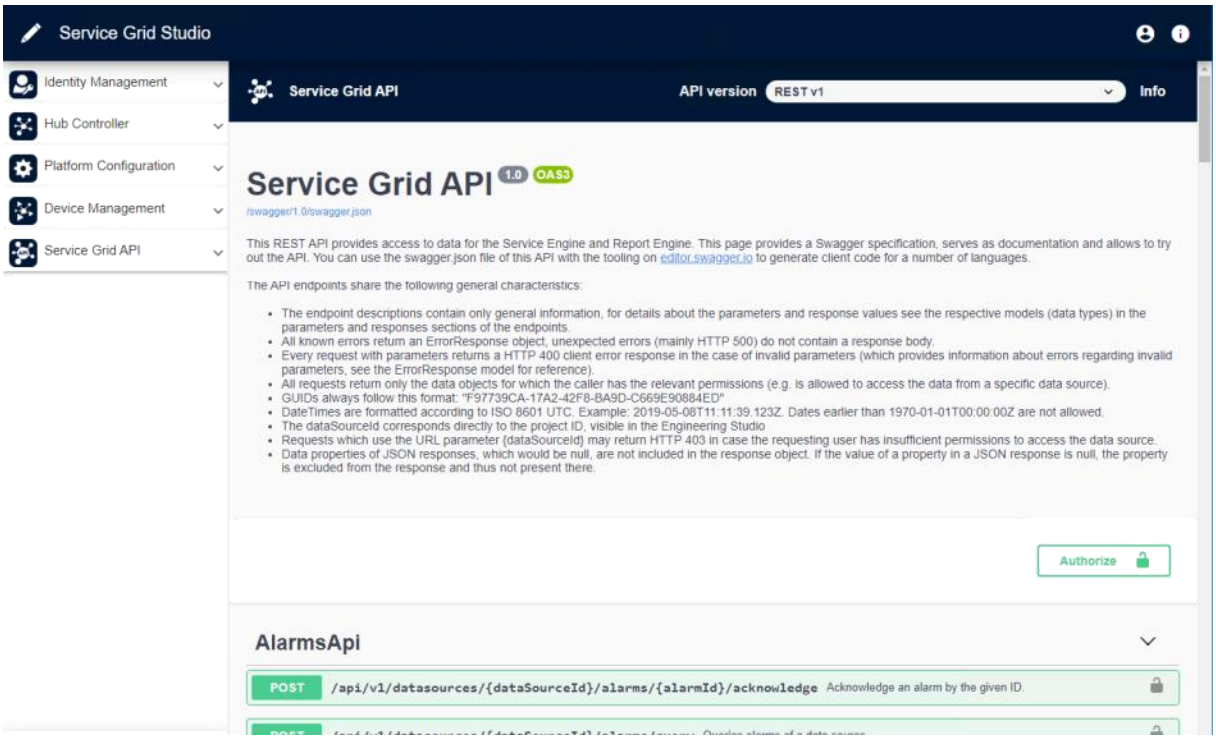
To assign the necessary role to the resource:

1. **Assigned Resources**: Click on the **...** button in the line for your Service Engine project.
2. Select **Manage roles** from the context menu.
3. Select the following permissions: **Service Grid API – Data Read**
4. Click on the **Submit** button.

You have thus assigned the necessary role to the resource.

From now on the *Administrator* user has read access to the released variable in Service Engine via Service Grid API.

22.11 Service Grid API



You can as a user access the Service Grid API via the web interface.

You can retrieve data from Service Grid via the Service Grid API .

<u>There are basically two options to do this:</u>

- In a test environment, you must access the Service Grid API **manually as the user**. To do this, use the Service Grid Studio web interface.
- In a productive environment, a **client application automatically** accesses the Service Grid API. To do this, you need an accordingly programmed third-party application.

For the test case described in this guide, you need neither a third-party application nor knowledge of API programming. Simply follow the instructions.

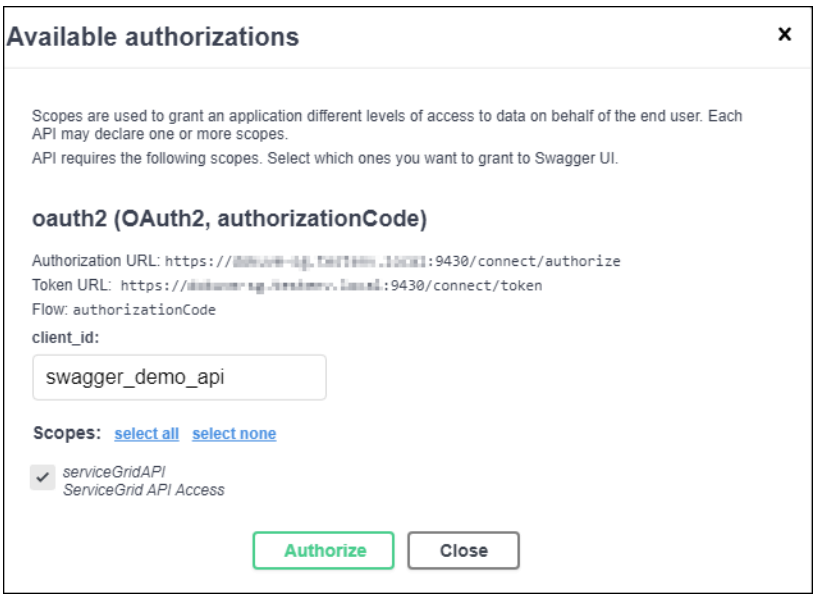
22.11.1 User authorization

For manual retrieval using the Service Grid API, you must be authorized:

To authorize a user in Service Grid API:

1. Make sure beforehand that Service Engine is running.
2. Open the web interface of **Service Grid Studio**.
3. Go to the **Service Grid API** menu item.
4. Click on the green **Authorize** button. Then a pop-up opens.
   Note: You are not authorized by default. The icon displays an opened lock.
5. Make sure that the value for the **client_id** field is set to *swagger_demo_api*.
6. Activate the following checkbox:
   - **serviceGridAPI**
   - **ServiceGrid API Access**
     Note: You thus determine the scope of the application.
7. Click on the **Authorize** button.
8. After successful authorization, the system shows the message *Authorized*.
9. You can now close the pop-up by clicking the **Close** button. The authorization remains active.
   Note: If you are authorized, you will see the locked icon.

You have thus successfully authorized your user account to the Service Grid API .



Users must authorize themselves in the web interface in order to query Service Grid API.

> **Hint: Interpret error codes**
>
> If the authorization fails, Service Grid API outputs error codes. The error codes are documented in the Help.

22.11.2 Test 1: Query available project

In this test, you check which data sources Service Grid API can access.

SELECT ENDPOINT

1. Make sure beforehand:
   - That Service Engine is running.
   - That user authorization to the Service Grid API (on page 82) has been completed.
2. Open the web interface of Service Grid Studio.
3. Go to Service Grid API in the menu.
4. Check whether the value *REST v1* is set as **API version** in the header.
5. Go to the **DataSourcesApi** category.
6. Go within the category to the line with the */api/v1/datasources* endpoint.

You must configure this endpoint for the following query.

QUERY PROJECT

1. Click on the blue **Get** button in the line. This expands the endpoint.
2. Click on the **Try it out** button.
3. Click on the **Execute** button.
4. The result of the query shows:
   - **DataSourceId**: "*ZENON10_DEMO*"
5. Copy the *dataSourceId* into a text file. You need this value for the following test.
   Note: It is identical to the project ID of your project (on page 70).

You have thus queried the zenon project available to Service Grid.

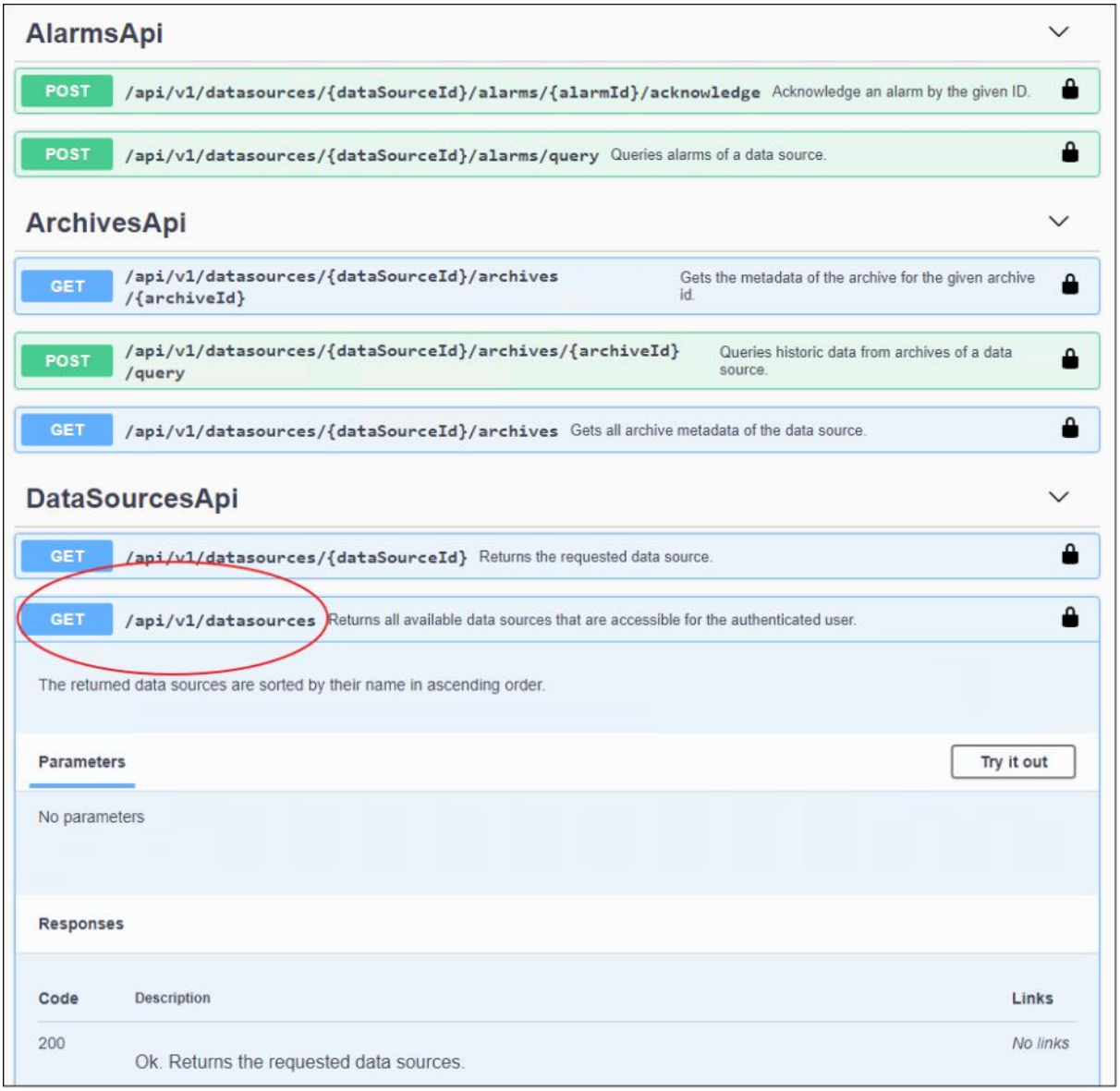
RESULT

The result of the query shows the available project:

Code Sample: Response body

```
{
"dataSources": [
{
"name": "ZENON10_DEMO",
"dataSourceId": "d3058681-c6a8-4b2e-908d-610676fce605",
"state": "Online"
}
]
}
```

SCREENSHOT



You can query available projects using the endpoint circled in red.

22.11.3 Test 2: Query available variables and variable values

In this test, you access released variables and variable values in the zenon project via Service Grid.

OPEN ENDPOINT

1. Make sure beforehand:
   - That Service Engine is running.

- That user authorization to the Service Grid API (on page 82) has been completed.

2. Open the Service Grid Studio web interface.
3. Go to the **Service Grid API** menu item.
4. Check whether the value *REST v1* is set as **API version** in the header.
5. Go to the **Variables API** category.
6. Go to the the line with the */api/v1/datasources/{dataSourceId}/variables/query}* endpoint.

You must configure the query in this endpoint.

CONFIGURE QUERY

1. Expand the line by clicking on the green **Post** button.
2. Click on the **Try it out** button. You have thus activated the input field for the **dataSourceId**.
3. Enter the **dataSourceId** (identical to the zenon project ID (on page 70)).
   Note: You have thus defined the target project for the query.
4. Change the following points in the **Query specification** (compare also code samples):
   a) *fields*: Replace the predefined "string" with "name", "value".
   You have thus defined the data fields for the query.
   b) *nameFilter*: Replace the predefined "string" with "*".
   You can use this placeholder to query all values unfiltered.
5. Then click on **Execute** to perform the query.
6. The query is acknowledged as follows: "Code 200" "Ok. Returns the queried variables."
7. The "Response body" section shows the query result (see code sample).

The query result shows the released variables and their variable values from the specified zenon project.

QUERY SPECIFICATION

Default Query:

Code Sample:

```
{
"fields": [
"string"
],
"nameFilter": {
"variableNames": [
"string"
]
}}
```

Custom Query (query of variables and variable values):

Code Sample:

```
{
"fields": [
"name", "value"
],
"nameFilter": {
"variableNames": [
"*"
]
}}
```

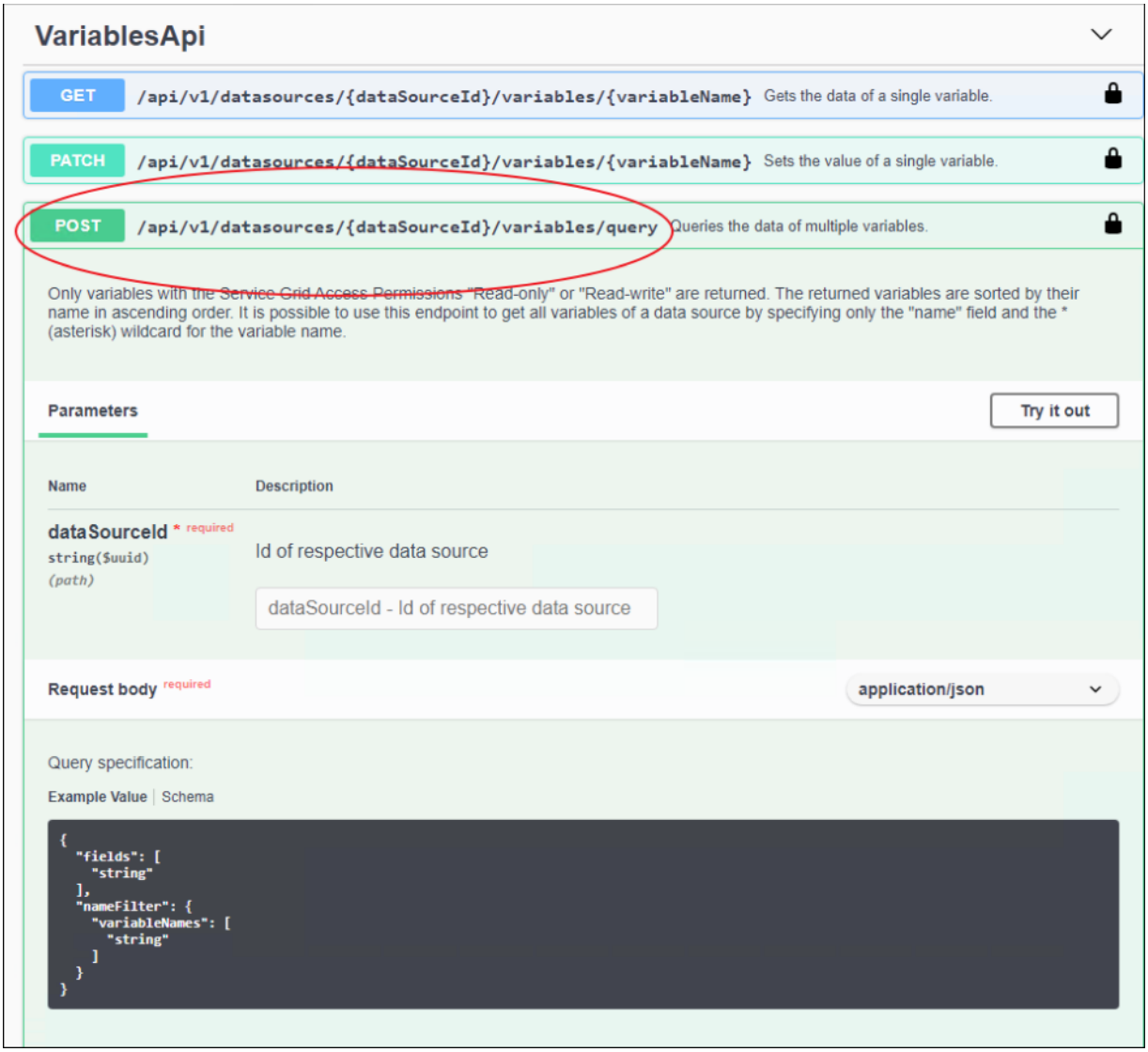
QUERY RESULT

The shared variable and the variable value are in the "Response body" section:

Code Sample:

```
{
"variables": [
{
"name": "ALC_GLOBAL_GROUND",
"value": "1"
}
]
}
```

SCREENSHOT



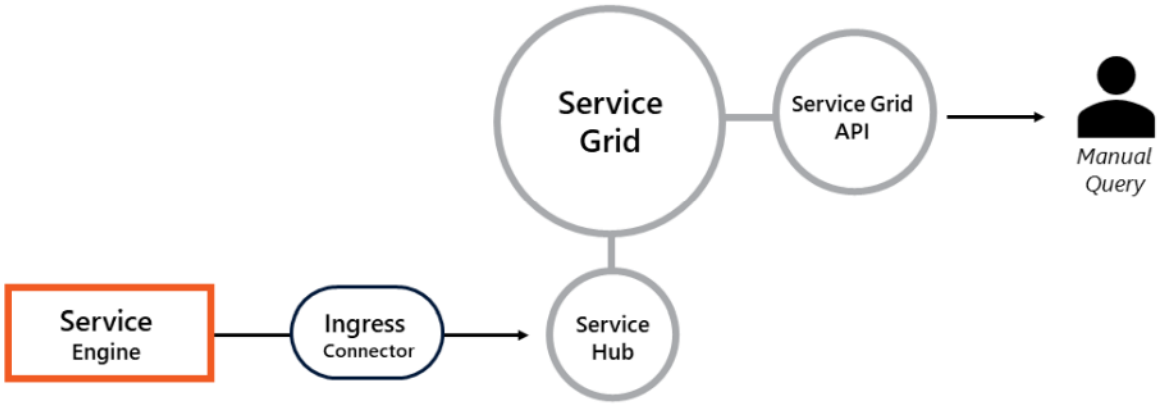
You must configure the query for this endpoint in the web interface.

22.12 Congratulations!

You have set up Service Grid and checked its functionality. You have succcessfully completed the Getting Started Guide.

You can find further information and detailed use cases in the Service Grid Help.

23 Getting Started Guide (Docker)



Test environment: Querying of process data of a Service Engine via Service Grid API.

In this guide, you will install and configure a simple use case for Service Grid 10.4. With the necessary prior knowledge (on page 92), you can work through this guide in 3 to 4 hours.

INSTALLATION OPTION

This guide refers to the Service Grid (Docker on Windows) installation option.

The following applies for Service Grid (Docker on Windows):

- Host operating system is Windows
- Docker Desktop on Windows is the Docker version used
- Service Grid containers are based on Linux

Note: If you are using other Docker versions (such as Docker on Linux, Kubernetes or hosting platforms), you must adapt this guide accordingly. You can find further information on supported installation options in the Help.

LEARNING OBJECTIVES

After working through this guide, you can:

- Initialize Service Grid
- Configure a zenon project for data transfer with Service Grid
- Query variable values of the zenon project using the Service Grid API

In doing so, you will have mastered the basic functions of Service Grid in a test environment.

OTHER USE CASES

All supported use cases for Service Grid are documented in the Help.

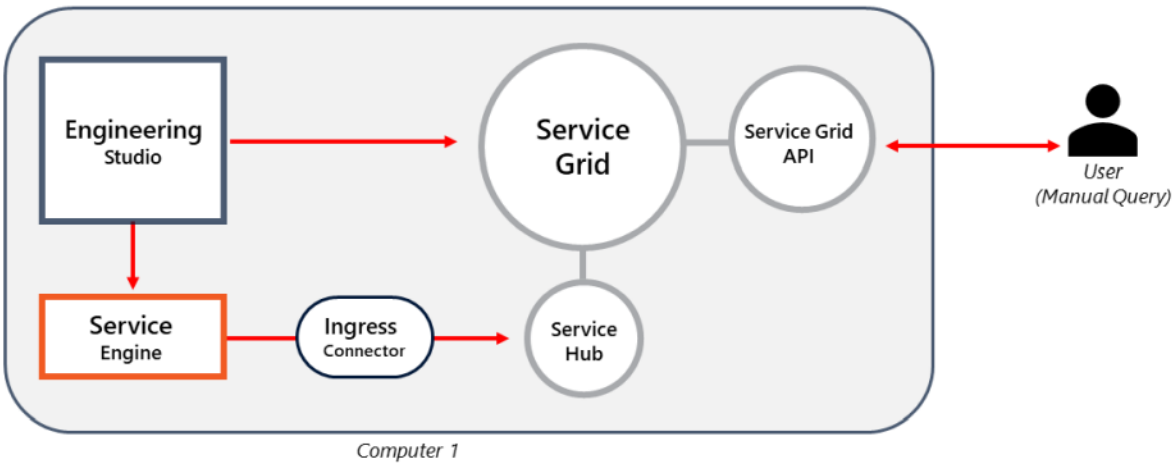
23.1 Test environment vs. productive environment

The test environment described in this guide is easier to set up than a typical productive environment.

The basic differences are:

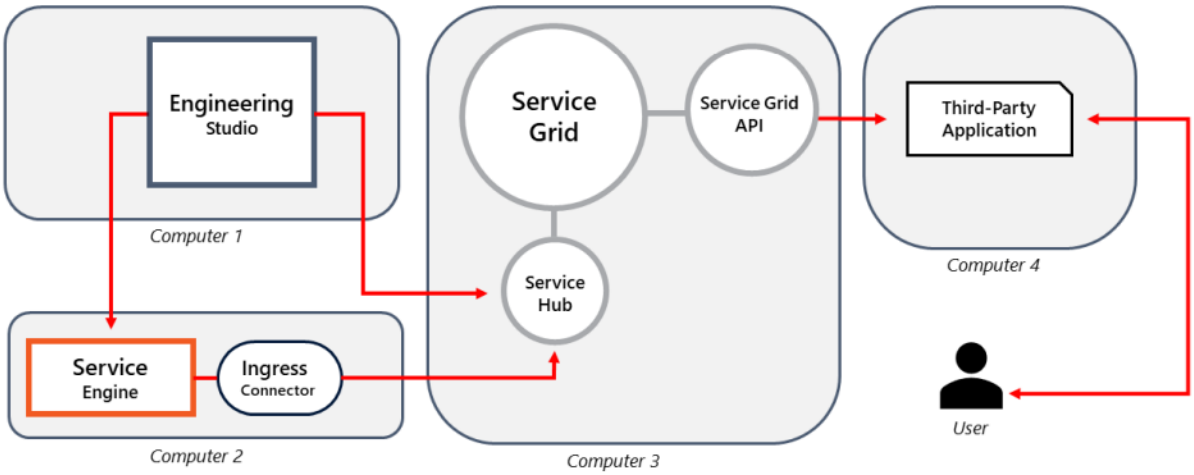
	Test environment	Productive environment
Installation option	▸ Service Grid (Docker on Windows)	▸ Service Grid (Windows native) ▸ Service Grid (Docker on Windows)
Number of computers	▸ 1 computer for Service Grid and all clients	▸ 1 computer for Service Grid ▸ Dedicated computers for clients
Network topology	All applications run on the same computer.	The applications run on different computers. The computers are usually distributed across different remote sites.
Multi-user system	Not suitable as a multi-user system.	Suitable as a multi-user system.
Passwords	It is possible to use predefined passwords in a protected test environment.	For all logins, it is essential that you assign your own secure passwords.

TEST ENVIRONMENT



Test environment: Service Grid and all clients are installed on the same computer. Service Grid API is queried by the user using a manual query.

PRODUCTIVE ENVIRONMENT



Productive environment: Each application is installed on a dedicated computer. Service Grid API is automatically queried for each third-party application.

23.2 Necessary prior knowledge

You require prior knowledge of zenon in order to use this guide.

You must know:

- How to create a zenon project
- How to start a project in Service Engine
- How to define variables and modify variable properties

Knowledge in these areas is advantageous (but not required):

- Operation of Docker containers
- Use of a REST API
- HTTPS certificates and certificate infrastructures

In general, the following applies: You require knowledge of all applications that you would like to connect with each other using Service Grid.

23.3 Prepare the system

Before you can install Service Grid, the test environment must be prepared properly.

23.3.1Test computer

The test computer must meet the following requirements:

1. Sufficient resources for the smooth operation of all installed applications (CPU, RAM, memory)
2. CPU must support hardware virtualization
3. CPU hardware visualization must be activated in BIOS
4. Functioning internet connection
5. No other software should be installed yet on the computer except for the operating system
6. In particular, no zenon applications may be installed yet.

Recommendation: For this guide, use a dedicated test computer with a newly setup operating system.

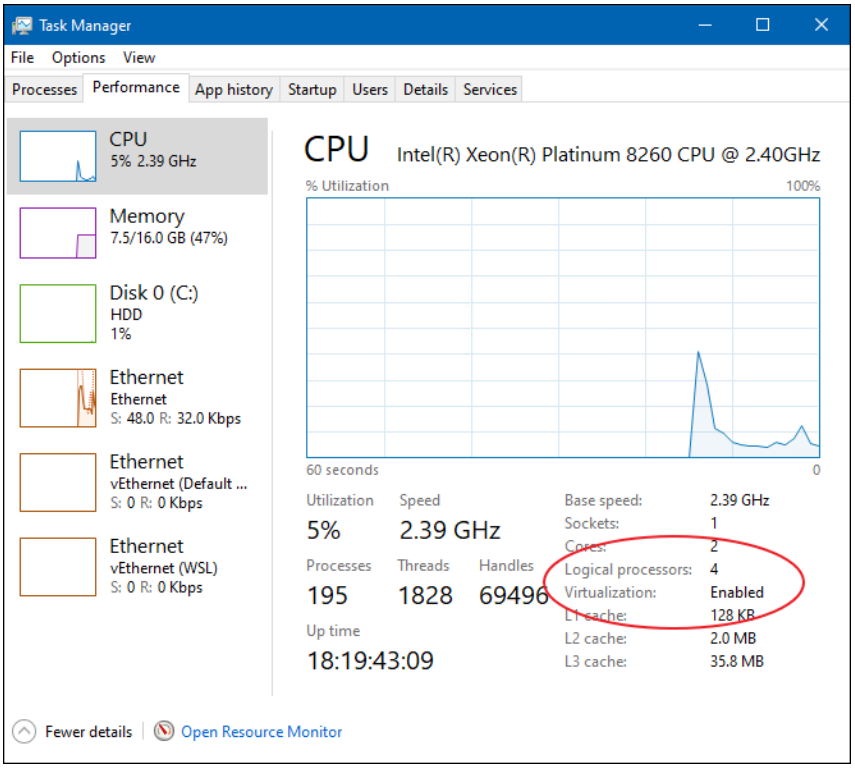
23.3.1.1 Check CPU hardware visualization

To check in Windows whether hardware virtualization is activated for your CPU:

1. Open the **Task-Manager** in Windows 10.

2. Click on the **Performance** tab.
3. Go to the **CPU** category there.
4. If your system is correctly configured, you will find the **Virtualization: Enabled** entry under the CPU graph.

Your computer thus meets the requirements for the installation of Docker.



The Windows Task Manager shows whether the CPU supports hardware virtualization. This applies equally to physical computers and VMs.

23.3.2 Test VM

You can install Service Grid both on a physical computer (test computer) as well as in a virtual machine (test VM).

<u>The following particular features must be noted when installing in a test VM:</u>

- zenon demo licenses have significantly shorter license periods on a VM than on a physical computer
- The host system of the VM must support Nested Virtualization
- The additional virtualization level of the VM can make troubleshooting more difficult

Secure operation of Service Grid is guaranteed both on a physical computer as well as in a virtual machine (VM).

23.3.3 DNS & network ports

Service Grid is a networked system. The services communicate with one another and with other computers.

The following network addressing is supported:

- Naming resolution: Domain Name System (DNS)
- Hostname: Fully Qualified Domain Name (FQDN)

The use of DNS and FQDN ensures that all services both in LAN and in WAN (internet) can be addressed securely.

These hostnames are explicitly not supported:

- "localhost"
- All hostnames that are not FQDN

Reason: Because the reliable addressing of services is not ensured.

The following applies for network ports:

- All network ports for Service Grid are documented.
- Make sure that these ports are not being used otherwise in your network or are not blocked, for instance, by firewalls.

Note: If required, you can reconfigure network ports.

23.3.4 Operating sytem and language settings

For the host operating system, you require:

- Windows 10, 64-bit (with the latest updates)

This guide uses English for:

- Windows operating system
- Language settings in the browser
- Service Grid
- All other applications

Recommendation: Configure your test computer accordingly.

23.3.5 Elevated PowerShell

This guide describes installation using an *elevated PowerShell*. This is a PowerShell with adminstrator privileges.

To create an elevated PowerShell:

1. Make sure that you have Windows adminstrator privileges on the test computer.
2. Create this link to your desktop:*%SystemRoot%\system32\WindowsPowerShell\v1.0\powershell.exe*
3. Right-click on the link to open the context menu.
4. Select the **Run as Administrator** option
   This action now opens the elevated PowerShell.

You can install and administer Service Grid using the elevated PowerShell.

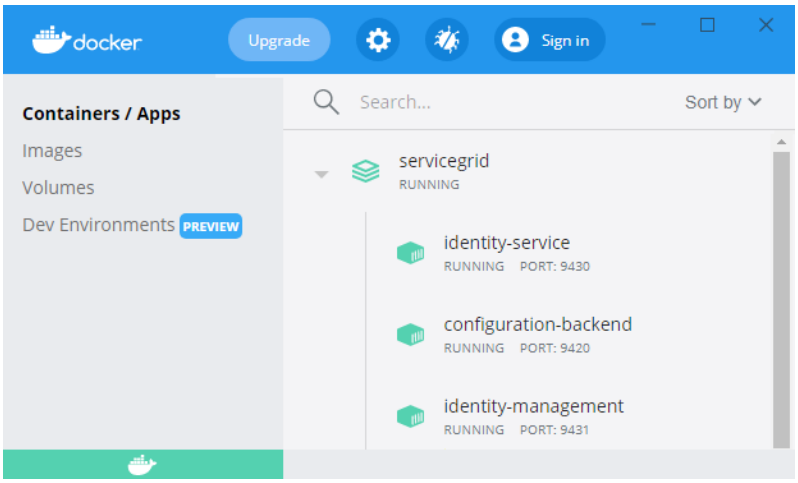
23.3.6 Docker Desktop for Windows

To install Docker Desktop for Windows:

1. Load the current version of the website of the provider:
   http://docker.com/get-started
2. Install **Docker Desktop for Windows** with the **WSL 2 engine**. To do this, you must select this checkbox during installation:
   Install required Windows components for WSL 2
3. Restart the Windows operating system after installation.
4. Start **Docker Desktop for Windows**.
   Note: Two error messages will be shown (*WSL 2 installation is incomplete* and an error message of the .Net Framework).
5. Follow the link shown to *https://aka.ms/wsl2kernel*.
6. Download the *WSL2 Linux kernel update package for x64 machines*.
7. Install the Update Package.
8. Restart the Docker application. You will know if the start was successful if you see "*No containers running*" and the Docker symbol highlighted in green.
9. Check whether Docker is configured for the use of Linux containers. This is the default setting of Docker.
   Note: The **Switch to Windows Containers...** entry is then shown in the Docker tray symbol.

By successfully starting Docker, you have also tested the functionality of the application.

23.3.6.1 Docker dashboard



Docker dashboard is a GUI for administering Docker containers.

Docker Dashboard allows you to start, stop and monitor Service Grid services.

To monitor and manage containers by mouse click:

- Start Docker Dashboard via the context menu of the Docker tray icon.
  At this point, the dashboard should display the message **No containers running**.

Hint: Leave Docker Dashboard permanently open during the installation of Service Grid. In this way, you can see at a glance which containers probably need a restart.

Dashboard allows you to:

- **Start all containers:** Select *servicegrid* and click on **Start**.
- **Start individual containers:** Click on **Start** next to the respective container entry.

Note: The Docker Dashboard is only included in Docker Desktop for Windows. In Docker for Linux, you only work with the command line interface.

23.3.7 Browser

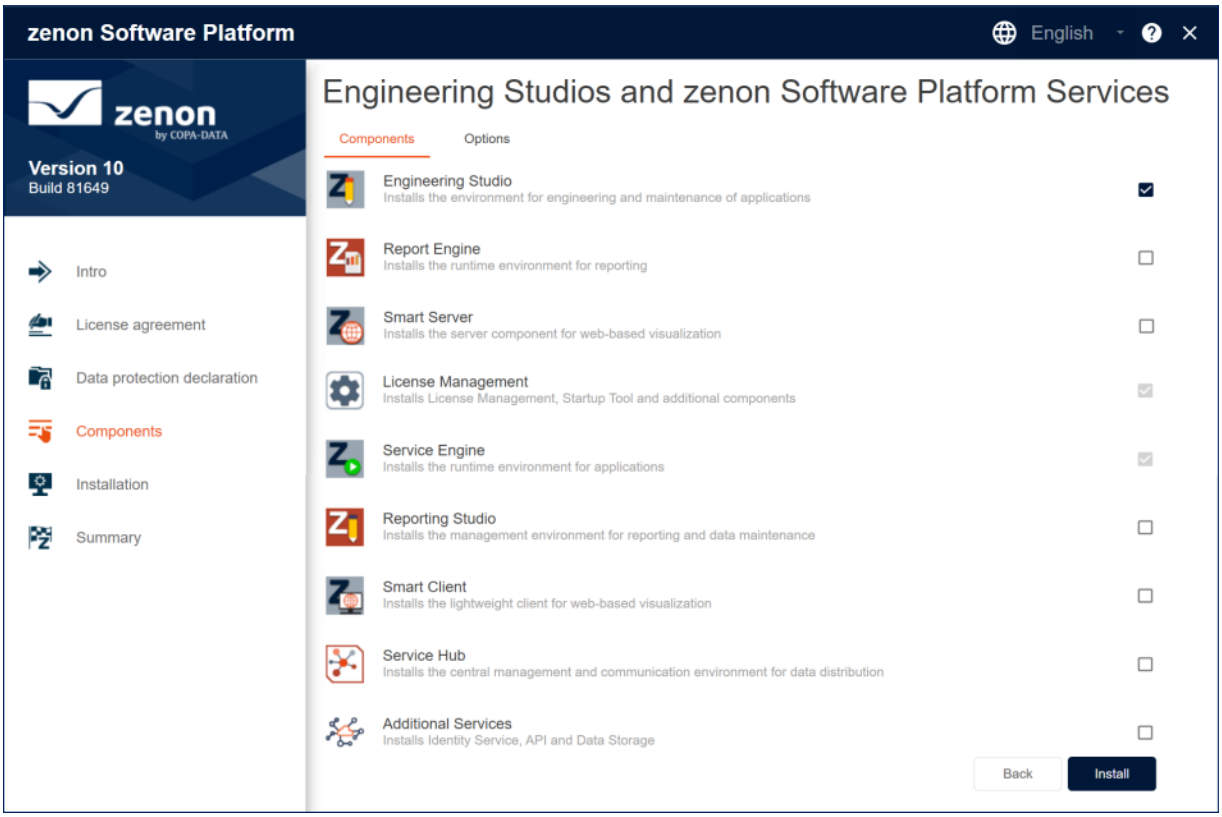
You need a web browser to access the web interfaces of Service Grid.

The following web browsers are supported:

- Google Chrome
- Mozilla Firefox
- Microsoft Edge
- Apple Safari

Always use the most recent version of the respective browser.

23.3.8 zenon platform



The test environment may only be installed with the highlighted components.

For the test environment, you must install zenon applications on the same test computer as Service Grid.

To install zenon:

1. Mount the ISO file for the zenon platform setup.
2. Start the platform setup.
3. Click the **Customize** button in the **Engineering Studios and zenon Software** selection category.
4. Unselect all applications except Engineering Studio.
   Note: Service Engine and required additional applications such as **Licence Management** remain preset (checkbox highlighted in gray).
5. Make sure that only Engineering Studio is selected (blue checkbox).
6. Perform the installation.

You have thus installed zenon on the test computer.

23.3.8.1 Build update

You use Build update to update the zenon platform.

To install Build update:

1. Ensure that the setup of the zenon platform (on page 72) has been completed beforehand.
   Important: No applications of the platform may be open.
2. Open Build update.
3. Start the installation by clicking the **Install** button.
4. When installation is complete, close the window of the application.

You have thus installed the latest build of zenon.

23.3.8.2 Activate demo license

You need licenses to run Service Grid. You can use the provided zenon demo license in a test environment.

The following applies for the demo license:

- You can work through the entire test case with the demo license
- You must manually activate the demo license
- The term of the demo license depends on the computer platform selected
  - Physical test computer: *30-day* term
  - Test VM: *1-day* term

To activate the demo license:

1. Open the **zenon Startup Tool**
2. Switch to the **Tools** tab
3. Start the **Licence Manager**
4. Click on **Advanced options...**
5. Click on **Advanced license administation...**
6. Activate the demo license.

A valid activated license is required for the operation of Service Grid.

23.3.9 Service Grid Gateway

Service Grid Gateway ensures compatibility between Service Grid and connected zenon clients.

To install Service Grid Gateway:

1. Make sure beforehand that the build update has been completed (on page 73).
2. First start the installer for Service Grid Gateway (x86)
3. Then start the installer for Service Grid Gateway (x64)
   Note: You always require both versions (x86 and x64).

You have thus installed Service Grid Gateway.

23.3.10 Service Grid configuration files

You can download the configuration files for Service Grid from the zenon website.

The download contains:

- Service Grid configuration files: *.env* and *docker-compose.yml*
- PDF file: **Service Grid manual** (including **Getting Started Guide**)

To download the configuration files:

- Go to the **copadata.com** website
- Go to the following subpage:
  Downloads -> Product-Downloads -> Software -> Current versions -> Service Grid
- Download the *.zip* file with the latest version of Service Grid (Docker).

Note: You must log in to the COPA-DATA website with your user account for this download. Registration is free.

Save the configuration files as follows:

1. Create the folder *c:\servicegrid*.
2. Unzip the *.zip* file in this folder.
3. Check if the ".env" file still contains the leading dot (**"."**).
   Reason: Some file operations in the Windows operating system can remove the dot. You then need to rename the file from **"env"** to **".env"**.

You have thus saved the configuration files for Service Grid in the installation folder.

23.4 Determine configuration values

In this chapter, you will compile all configuration values you need for the initialization and configuration of Service Grid. There are corresponding configuration templates for this.

Please note the following for working with configuration templates:

1. **System-specific values**: Must always be determined individually for your system (e.g. hostname, project ID). You may not apply the provided sample values.
2. **Non-system-specific values**: Can be assigned by you (e.g. usernames, passwords). In a protected test environment, you can also apply the suggested sample values.

You must always assign your own passwords in productive environments or unprotected environments. This is also recommended for protected test environments.

You enter configuration values in two different places:

- Text-based configuration in the *.env* file (only in the "Service Grid Docker" installation option)
- GUI-based configuration in the web interfaces of Service Grid (all installation options)

In this chapter, there are configuration templates with sample values for each configuration required.

23.4.1Own template file

It is good idea to create your own template file for the configuration. A simple text file will do here.

It has the following advantages:

- You can look up all the configuration values in one place.
- You can apply configuration files using copy & paste.

The configuration template must be protected from unauthorized access.

23.4.2 .env file

You configure the basic settings for Service Grid in the *.env* file.

To configure the .env file:

1. Determine all the necessary configuration values (see the table below).
2. Open the **.env** file with a Text Editor (Notepad++ for example).
3. Copy and paste the values in the respective variable fields of the **.env** file.
4. Save the **.env** file.
5. Check if the ".env" file still contains the leading dot (**"."**).
   Reason: Some file operations in the Windows operating system can remove the dot. You then need to rename the file from **"env"** to **".env"**.

Thus, the **.env** file is fully configured for your system.

DETERMINE CONFIGURATION VALUES

You require the following variable values for the configuration of the .env file:

Name of the variables	Sample values	Description
Datenbank		After Service Grid has been successfully installed, the database credentials are only needed for administrative work such as the backup and restore of Persistence Instance (on page 20).
SG_Persistence_Username=	*sgp_user*	You can choose the usernames yourself.
SG_Persistence_Password=	*sgp_Changeme123!*	You can define the password yourself. Please note the minimum password requirements (on page 102)! Unsuitable passwords lead to malfunctions between services.
SG_Persistence_Uri=	Do not fill in variable value	Is not needed in the test environment.
Machine settings		
MACHINE_HOSTNAME=	*mycomputer.mydomain.com* **System-specific value:** ▸ Determine the FQDN hostname (on page 69) of your computer ▸ FQDN must be entered in continuous lowercase letters.	**Common configuration errors in the MACHINE_HOSTNAME are:** ▸ Use of uppercase letters (e.g. "*MyComputer.mydomain.com*", "*MYCOMPUTER.mydomain.com*") ▸ Use of *localhost* ▸ Use of the loopback address *127.0.0.1* This leads to malfunctions.

Note: In this test environment, you should leave all the other variable values of the *env* file unchanged.

23.4.2.1 Minimum password requirements

The minimum password requirements must be met for every password assigned in Service Grid. This is due to technical reasons and thus also applies for protected test environments.

Unsuitable passwords can cause malfunctions in Service Grid.

The minimum password requirements in Service Grid are:

- Password length: at least 8 characters
- One uppercase letter (*A - Z*)
- One lowercase letter (*a-z*)
- One numeric character (*0 - 9*)
- One special character (*!#$%&'()*+,-./\:;<=>?[]@^_'`{}|~*)

Only when all the requirements have been met is a password regarded as being suitable. Unsuitable passwords fail to meet at least one of the above-mentioned requirements.

When configuring passwords, a difference must be made between two cases:

- Configuration using GUI:
  The web interfaces of Service Grid check during the configuration of a password if the minimum requirements have been met. You are only able to configure suitable passwords here.
- Configuration using the configuration file:
  It is technically possible to configure unsuitable passwords in the *.env* file. This leads to authentication problems between services.

Important: Only use suitable passwords that fully meet the minimum password requirements.

23.4.2.2 Determine hostname

To determine the actual hostname of a Windows computer:

1. Open a Windows command prompt.
2. Execute the following command:
   ping localhost
3. The command line interface shows your computer's actual hostname. This is usually a **Fully Qualified Domain Name (FQDN)**.
4. Examples of actual FQDNs are:

- *MYCOMPUTER.mydomain.com*
- *MyComputer.mydomain.com*
- *mycomputer.myddomain.com*

5. Convert the actual hostname to lowercase letters: *mycomputer.mydomain.com*

You have thus determined the hostname you need for use in Service Grid.

23.4.3 Bookmarks for web interfaces

Create boomarks in the browser for the following URLs:

Name	Sample values	Description
Identity Service	*https://mycomputer.mydomain.com:9443/identity-service* System-specific value*	The **Identity Service** web interface allows each user to manage certain settings of their own user account.
Service Grid Studio	*https://mycomputer.mydomain.com:9443* System-specific value*	The **Service Grid Studio** interface allows exclusively users with administrator privileges to fully administer Service Grid.

* You must replace mycomputer.mydomain.com in the URLs with the hostname of your computer (on page 69).

23.4.4 zenon Project

In the test environment, you use Service Grid API to access variable data in a zenon project.

DETERMINE PROJECT PARAMETERS

You need the following values:

Parameters	Sample value	Description
Project Name	*ZENON10_DEMO*	This project is installed by default with Engineering Studio. You also have the option to use a different project of your choice.
Project ID	***a0f4d8f9-c009-41d5-bc30-457dd92f6a29*** System-specific value:	The unique identification number of your zenon project.

Parameters	Sample value	Description
	You must determine the project ID (on page 70) yourself.	
Variable	*ALC_GLOBAL_GROUND*	This variable is contained in the *ZENON10_DEMO* project. You also have the option to use a different variable of your choice.

Note: You can only determine these parameters once the zenon installation has been completed (on page 97).

23.4.4.1 Determine project ID

The Project ID is individual for each zenon project. It is an automatically generated ID.

It is based on the following schema:

a0f4d8f9-c009-41d5-bc30-457dd92f6a29

To determine the project ID:

1. In Engineering Studio, select the demo project of zenon.
2. Select the main project node in the project.
3. Press the **Ctrl+Alt+E** shortcut.
   This opens the file path of your zenon project on your hard disk.
4. Select the superordinate folder name.
   Reason: The Project ID is identical to the superordinate folder name.
5. Open the context menu by right clicking on *Rename* with the mouse.
6. You can then copy the Project ID to the clipboard and use it again as you wish.

You have thus determined your Project ID. You can use this ID to uniquely address the project using the Service Grid API.

23.5 Initialization

Once you have prepared all the configuration values, you can initialize Service Grid.

23.5.1Download Docker images

To download Docker images:

1. Start Docker.
2. Open an elevated PowerShell
3. Switch to the local working folder:
   cd C:\servicegrid
4. Execute this command:
   docker-compose -f .\docker-compose.yml pull
5. The images are loaded via the internet connection of *hub.docker.com*.

You have thus downloaded the docker containers for Service Grid on your local computer.

Hint

Successful execution of the command is confirmed as follows:

Pulling hub-controller ... done

Pulling data-hub ... done

Pulling service-grid-persistence ... done

Pulling configuration-backend ... done

Pulling service-grid-platform-configuration ... done

Pulling service-grid-studio ... done

Pulling servicegrid-redis ... done

Pulling service-grid-api ... done

Pulling identity-service ... done

Pulling identity-management ... done

Pulling data-storage ... done

23.5.2 Initialize Service Grid

To initialize Service Grid:

1. Check beforehand if you have completely configured the *.env*file.
2. Open an elevated PowerShell.
3. Switch to the local working folder:
   cd C:\servicegrid
4. Execute this command:
   docker-compose -f .\docker-compose.yml up
5. Confirm the Windows Firewall approval for Service Grid.
   Note: The firewall blocks the services until they are approved. This can lead to timeouts, which can cause the initialization to fail. In this case, you must restart the initialization.
6. **Docker Dashboard**: Check under **Containers / Apps** if all services are in the *Running* status.

You have thus successfully initialized Service Grid.

23.6 Start and monitor services

You must start the services in Docker manually.

There are the following options for starting an already installed Service Grid:

1. Using a PowerShell command:
   docker-compose -f .\docker-compose.yml up
   Note: This is the same command as for initialization.
2. With the GUI via Docker Dashboard (on page 107).

Generally, all services once started are in the *running* status. Exception: **Data-Storage** may be in the *exited* status (because it is not used).

The following applies for the monitoring of services:

- You can find the complete list of services in the Help.
- **Docker Dashboard** allows you to conveniently monitor services with the GUI.

After changes to the configuration, it may be necessary to restart individual services or Service Grid.

23.7 Create initial user (administrator)

You must create the first user account in **Identity Service** yourself.

To create the initial user:

1. Open the web interface for **Identity Service** in the browser using the bookmark.
2. Confirm the HTTPS certificate warning shown.
   Note: You cannot install the root certificate until later.
3. You receive the following message from **Identity Service**:
   Please create the initial user. This user will have administrator permissions.
4. Assign the User Credentials for the Initial User:
   - "*Administrator*" (sample value for **Username**)
   - "*Changeme123!*" (sample value for **Password**)
     Important: You must assign your own secure **Password** in productive systems or unprotected test systems.
5. Confirm your entries. You are then forwarded to the login for the **Identity Service**.
6. Log in.

You have thus created the initial user account and logged in to the web interface of the **Identity Service**. You are now logged in for all web interfaces in Service Grid.

23.8 Set up HTTPS trust

Set up a HTTPS trust using the Root Certificate.

A HTTPS trust is needed in Service Grid for:

- All client computers that connect with Service Grid.
- The computer with the Service Grid installation.

Not having a trust has the following consequences:

- Service Grid web interfaces: Display certificate warnings. You can confirm the certificate warning and establish the connection manually.
- Client applications: Do not display certificate warnings. There is no option to establish the connection manually.

You can find further information on certificates and trusts in the Service Grid Help.

INSTALL ROOT CERTIFICATE

You must install the Service Grid Root Certificate on the test computer.

To download the certificate file:

1. Open the Service Grid Studio web interface using the bookmark. The HTTPS connection is currently shown as insecure.
2. Confirm the HTTPS certificate warning.
   Note: You cannot install the root certificate until later.
3. Go to the **Hub Controller** menu item.
4. Go to the **Certificates** subpage.
5. Click on the **Download CA Certificate** button.

You have thus saved the certificate file on your local computer.

To install the certificate:

1. Open the Windows **Run** dialog with the *WIN+R* keyboard shortcut.
2. Execute the following command at the command line interface:
   certlm.msc
3. This opens the Management Console (MMC) with the Windows Certificate Manager.
4. Go to this folder:
   Trusted Root Certification Authorities\Certificates.
5. Right-click on the *Certificates* folder. This opens the context menu.

6. Select the following option from the context menu:
   All Tasks\Import...
7. Import the certificate file.

You have thus installed the certificate and set up the HTTPS trust.

CHECK THE HTTPS TRUST

To check the HTTPS trust:

1. Restart your browser.
2. Open the web interface of Service Grid Studio.
3. The HTTPS connection is shown as secure. You no longer receive any certificate warnings.

You have thus checked the HTTPS trust.

23.9 Create Client Certificate Bundles

You must create several Client Certificate Bundles (CCBs) with the Service Node Configuration Tool. Each client needs an individual CCB.

A missing CCB has consequences:

- The client can no longer connect to Service Grid.
- In applications, you cannot configure a connection to the Service Hub (in Service Engine for example).

The CCB is a requirement for the setup of an encrypted network connection to the **Service Hub**. The **Service Hub** does not support unencrypted network connections.

START SNCT

This is how you start the Service Node Configuration Tool:

1. Check in advance whether Service Engine and/or Engineering Studio are registered on the computer.
   Reason: In the case of a registered **Smart Client**, **SNCT** does not recognize Service Engine nor Engineering Studio.
2. Open the **zenon Startup Tool**.
3. Click on the **Tools** tab.
4. Select the **Service Node Configuration Tool** entry.
5. Click on **Start**.

This opens the **SNCT**.

CREATE CLIENT CERTIFICATE BUNDLES

This is how you configure Client Certificate Bundles:

1. Ensure in advance that the HTTPS trust setting for the test computer (on page 75) has been set up.
2. In the **SNCT**, select all displayed client entries under **Selection**.
3. Click on the **Next** button.
4. Under **Connection** in the **Host name or address** field, enter the host name of the test computer:
   [mycomputer.mydomain.com]
   Note: Enter the host name using only lowercase letters!
5. Use the pre-set port:
   9410
6. Enter **Username** and **Password** for the login to **Service Hub**.
   Note: To do this, you need an **Identity Service** user account with the *Service Hub Administrator* user role. This user role is assigned by default to all users of the *Administrators* group.
7. Click on **Execute**.
8. Check under **Retrieval** if the following confirmation has been displayed:
   "Successfully wrote configuration for all selected products."
9. Close the **Service Node Configuration Tool** with the **Close** button.

You have thus configured the Client Certificate Bundles.

23.10 Engineering Studio

You must configure Engineering Studio for the data transfer between Service Engine and Service Grid.

23.10.1 Configure Service Hub

This is how you configure the connection to the Service Hub:

1. Start Engineering Studio.
   Note: If Engineering Studio is already running, you must stop and restart it.
2. Select the zenon project.
3. Select the **Network** property group in the project properties.
4. Go to property group **Service Grid - General**.

5. In the **Service Hub** drop-down menu, select the entry for the computer on which Service Grid is installed.
   Note: Entry is only visible if a CCB has been created for Engineering Studio.
6. Activate the **Execute Service Grid Ingress Connector** property.
   Note: This checkbox is only visible if the connector is licensed.

You have thus configured Engineering Studio for the connection with the **Service Hub**.

23.10.2 Release variable

Service Engine can exchange data with Service Grid in different ways. The main method of data exchange is variables.

To configure Service Grid access rights for a variable:

1. In Engineering Studio, select the following **variable** in the **project manager**:

   ALC_GLOBAL_GROUND
2. Open the properties of the variable.
3. Go to property group **Authorization/eSignature**.
4. Go to **Service Grid settings**.
5. Select the *write-protected* entry from the drop-down list of the **Access permission** property.
   Note: This setting protects the *ALC_GLOBAL_GROUND* variable from write access from Service Grid.

You have thus assigned Service Grid access rights for this variable.

23.10.3 Starting Service Engine

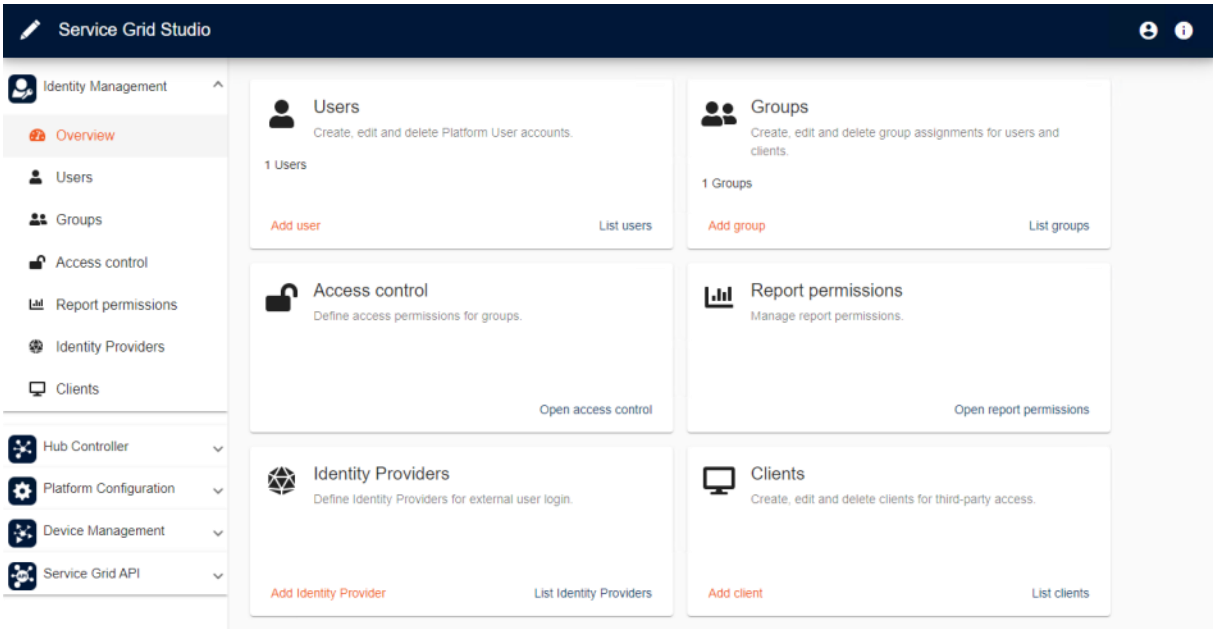
After completing configuration, you can start Service Engine.

To start Service Engine:

1. Save the project with all the changes.
2. Create all Service Engine files.
3. Start Service Engine.

Service Engine is now ready for the data transfer with Service Grid.

23.11 Identity Management



In Identity Management, you administer users, groups, resources and privileges.

Assign the Administrator user the privilege to access Service Engine via Service Grid.

To do this, the following configurations are necessary:

- You must create the new *Users* user group.
- You must assign the *Users* user group to the already existing *Administrator* user.
- The zenon project is a resource in Service Grid. You must assign this resource to the *Users* user group.
- You must now assign the **Service Grid API – Data Read** role to the resource.

The *Administrator* thus has read access to released variables and variable values in Service Engine via Service Grid API.

23.11.1 Create group and add users

To create a group:

1. Go to:
   - **Service Grid Studio** web interface
   - **Identity Management** menu entry
   - **Groups** submenu
2. Click in the middle column on the **Create Group** button.

3. In the pop-up, enter the following group name: *Users*
4. Click on **Add**.
   The group will then be displayed in the middle column.

You have thus created the *Users* group.

To add a user to the group:

1. Select the *Users* group.
2. Click on the **Add user** button.
3. Select the *Administrator* user.
   Note: This user is displayed by default in the list as *admin admin*.
4. Click on **Add**.

You have thus added the *Administrator* user of the *Users* group.

23.11.2 Add resource and add role

To add the Service Engine resource to the Users group:

1. Make sure beforehand that Service Engine is running.
2. Open the web interface of **Service Grid Studio**.
3. Go to the **Identity Management** menu item.
4. Open the **Access Control** submenu.
5. Select the *Users* under **Groups**.
6. Click on the **Add Resources** button.
7. Select the project: *ZENON10_DEMO*.
8. Click on the **Add** button.

You have thus added the resource to the user group.

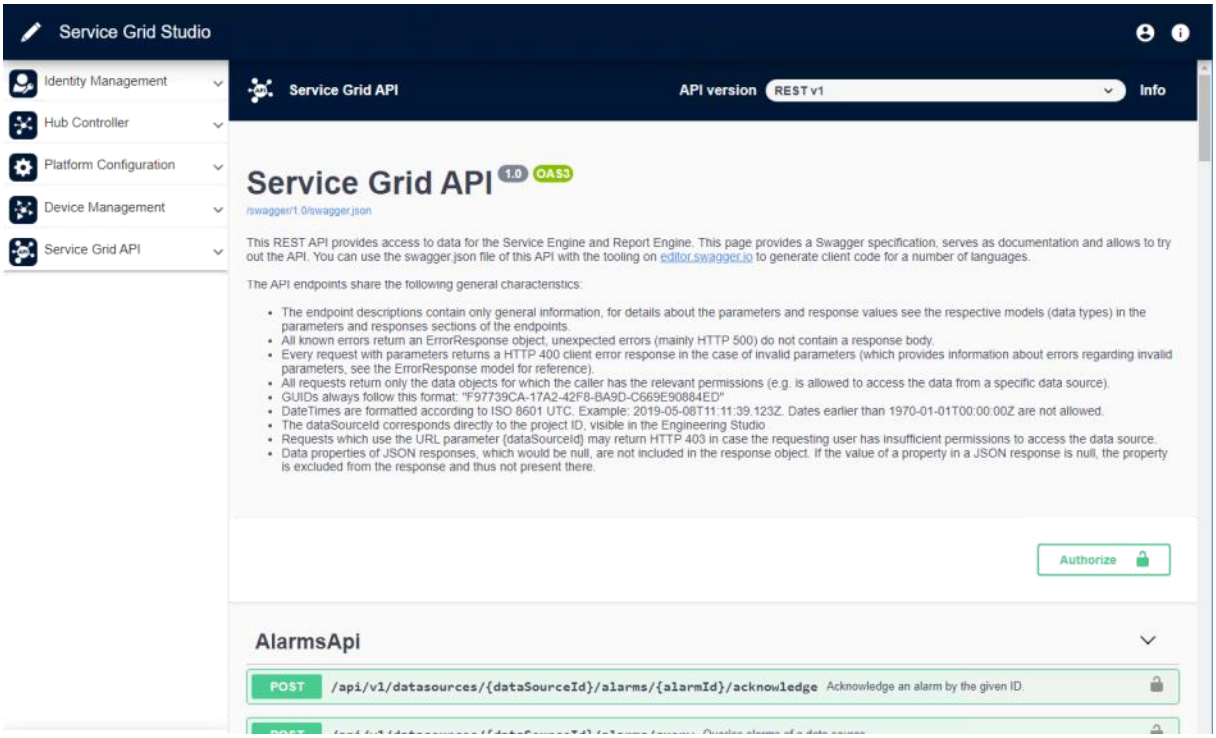
To assign the necessary role to the resource:

1. **Assigned Resources**: Click on the **...** button in the line for your Service Engine project.
2. Select **Manage roles** from the context menu.
3. Select the following permissions: **Service Grid API – Data Read**
4. Click on the **Submit** button.

You have thus assigned the necessary role to the resource.

From now on the *Administrator* user has read access to the released variable in Service Engine via Service Grid API.

23.12 Service Grid API



You can as a user access the Service Grid API via the web interface.

You can retrieve data from Service Grid via the Service Grid API .

<u>There are basically two options to do this:</u>

- In a test environment, you must access the Service Grid API **manually as the user**. To do this, use the Service Grid Studio web interface.
- In a productive environment, a **client application automatically** accesses the Service Grid API. To do this, you need an accordingly programmed third-party application.

For the test case described in this guide, you need neither a third-party application nor knowledge of API programming. Simply follow the instructions.

23.12.1 User authorization

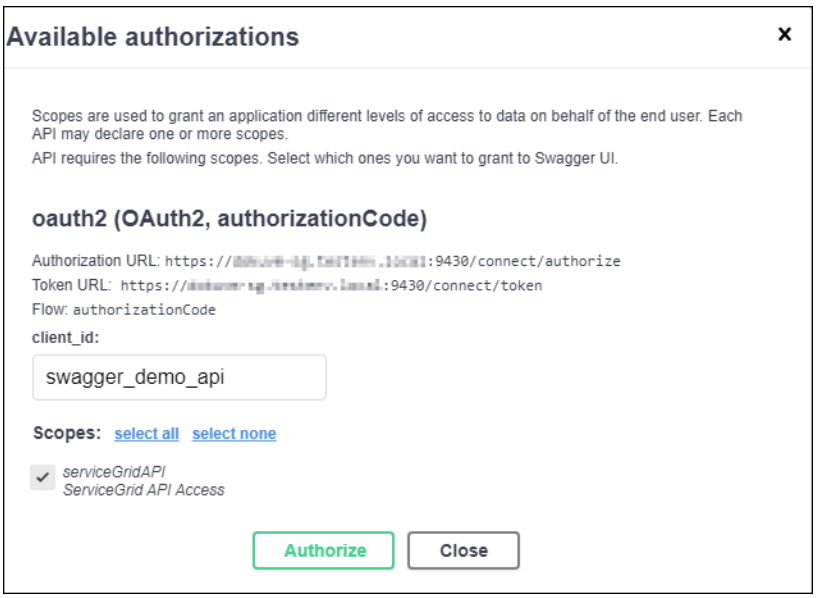
For manual retrieval using the Service Grid API, you must be authorized:

<u>To authorize a user in Service Grid API:</u>

1. Make sure beforehand that Service Engine is running.
2. Open the web interface of **Service Grid Studio**.
3. Go to the **Service Grid API** menu item.

4. Click on the green **Authorize** button. Then a pop-up opens.
   Note: You are not authorized by default. The icon displays an opened lock.
5. Make sure that the value for the **client_id** field is set to *swagger_demo_api*.
6. Activate the following checkbox:
   - **serviceGridAPI**
   - **ServiceGrid API Access**
     Note: You thus determine the scope of the application.
7. Click on the **Authorize** button.
8. After successful authorization, the system shows the message *Authorized*.
9. You can now close the pop-up by clicking the **Close** button. The authorization remains active.
   Note: If you are authorized, you will see the locked icon.

You have thus successfully authorized your user account to the Service Grid API .



Users must authorize themselves in the web interface in order to query Service Grid API.

Hint: Interpret error codes

If the authorization fails, Service Grid API outputs error codes. The error codes are documented in the Help.

23.12.2 Test 1: Query available project

In this test, you check which data sources Service Grid API can access.

SELECT ENDPOINT

1. Make sure beforehand:
   - That Service Engine is running.
   - That user authorization to the Service Grid API (on page 82) has been completed.
2. Open the web interface of Service Grid Studio.
3. Go to Service Grid API in the menu.
4. Check whether the value *REST v1* is set as **API version** in the header.
5. Go to the **DataSourcesApi** category.
6. Go within the category to the line with the */api/v1/datasources* endpoint.

You must configure this endpoint for the following query.

QUERY PROJECT

1. Click on the blue **Get** button in the line. This expands the endpoint.
2. Click on the **Try it out** button.
3. Click on the **Execute** button.
4. The result of the query shows:
   - **DataSourceId**: "*ZENON10_DEMO*"
5. Copy the *dataSourceId* into a text file. You need this value for the following test.
   Note: It is identical to the project ID of your project (on page 70).

You have thus queried the zenon project available to Service Grid.

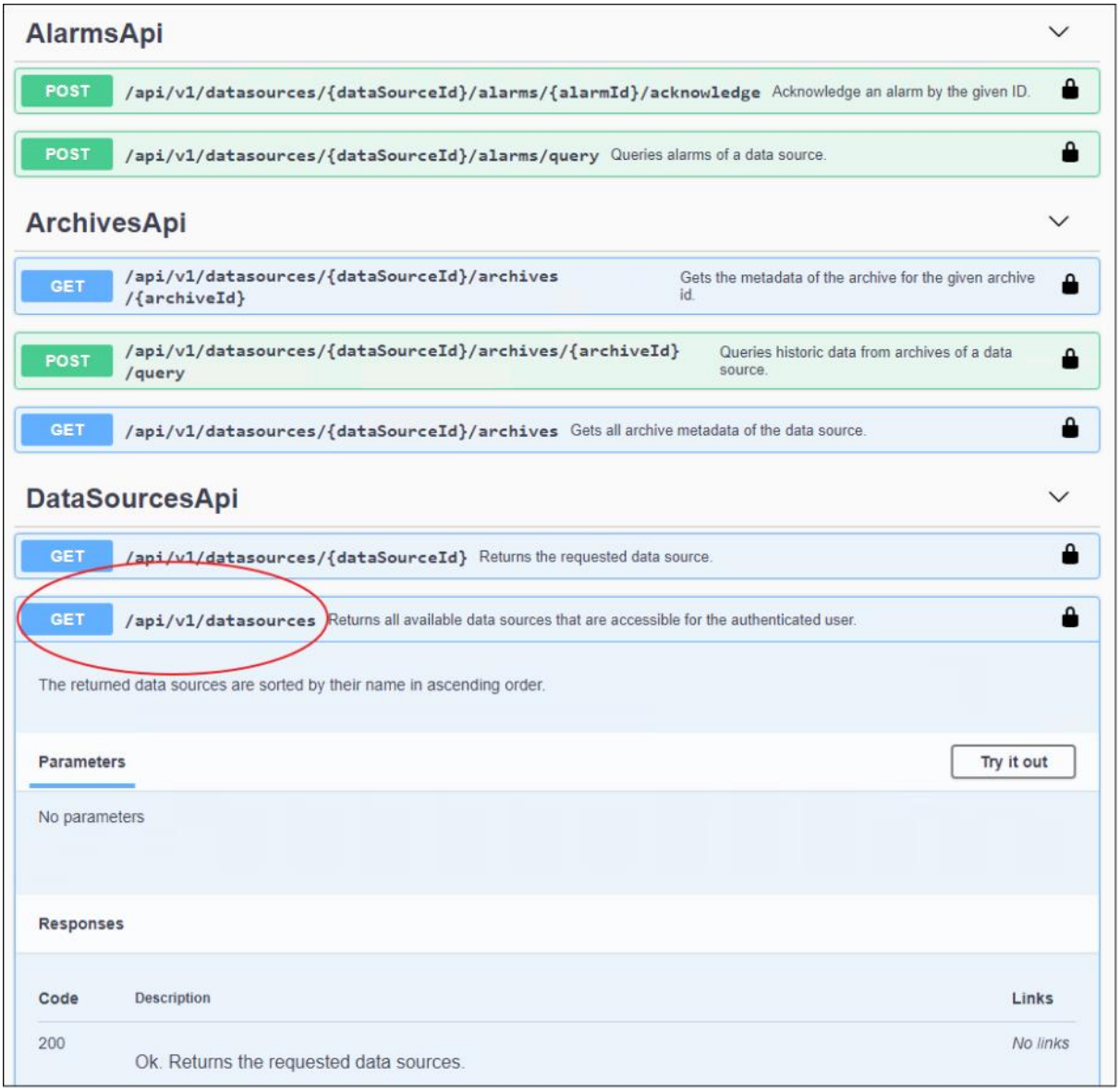
RESULT

The result of the query shows the available project:

Code Sample: Response body

```
{
"dataSources": [
{
"name": "ZENON10_DEMO",
"dataSourceId": "d3058681-c6a8-4b2e-908d-610676fce605",
"state": "Online"
}
]
}
```

SCREENSHOT



You can query available projects using the endpoint circled in red.

23.12.3 Test 2: Query available variables and variable values

In this test, you access released variables and variable values in the zenon project via Service Grid.

OPEN ENDPOINT

1. Make sure beforehand:
   - That Service Engine is running.

- That user authorization to the Service Grid API (on page 82) has been completed.

2. Open the Service Grid Studio web interface.
3. Go to the **Service Grid API** menu item.
4. Check whether the value *REST v1* is set as **API version** in the header.
5. Go to the **Variables API** category.
6. Go to the the line with the */api/v1/datasources/{dataSourceId}/variables/query}* endpoint.

You must configure the query in this endpoint.

CONFIGURE QUERY

1. Expand the line by clicking on the green **Post** button.
2. Click on the **Try it out** button. You have thus activated the input field for the **dataSourceId**.
3. Enter the **dataSourceId** (identical to the zenon project ID (on page 70)).
   Note: You have thus defined the target project for the query.
4. Change the following points in the **Query specification** (compare also code samples):
   a) *fields*: Replace the predefined "string" with "name", "value".
   You have thus defined the data fields for the query.
   b) *nameFilter*: Replace the predefined "string" with "*".
   You can use this placeholder to query all values unfiltered.
5. Then click on **Execute** to perform the query.
6. The query is acknowledged as follows: "Code 200" "Ok. Returns the queried variables."
7. The "Response body" section shows the query result (see code sample).

The query result shows the released variables and their variable values from the specified zenon project.

QUERY SPECIFICATION

Default Query:

Code Sample:

```
{
"fields": [
"string"
],
"nameFilter": {
"variableNames": [
"string"
]
}}
```

Custom Query (query of variables and variable values):

Code Sample:

```
{
"fields": [
"name", "value"
],
"nameFilter": {
"variableNames": [
"*"
]
}}
```

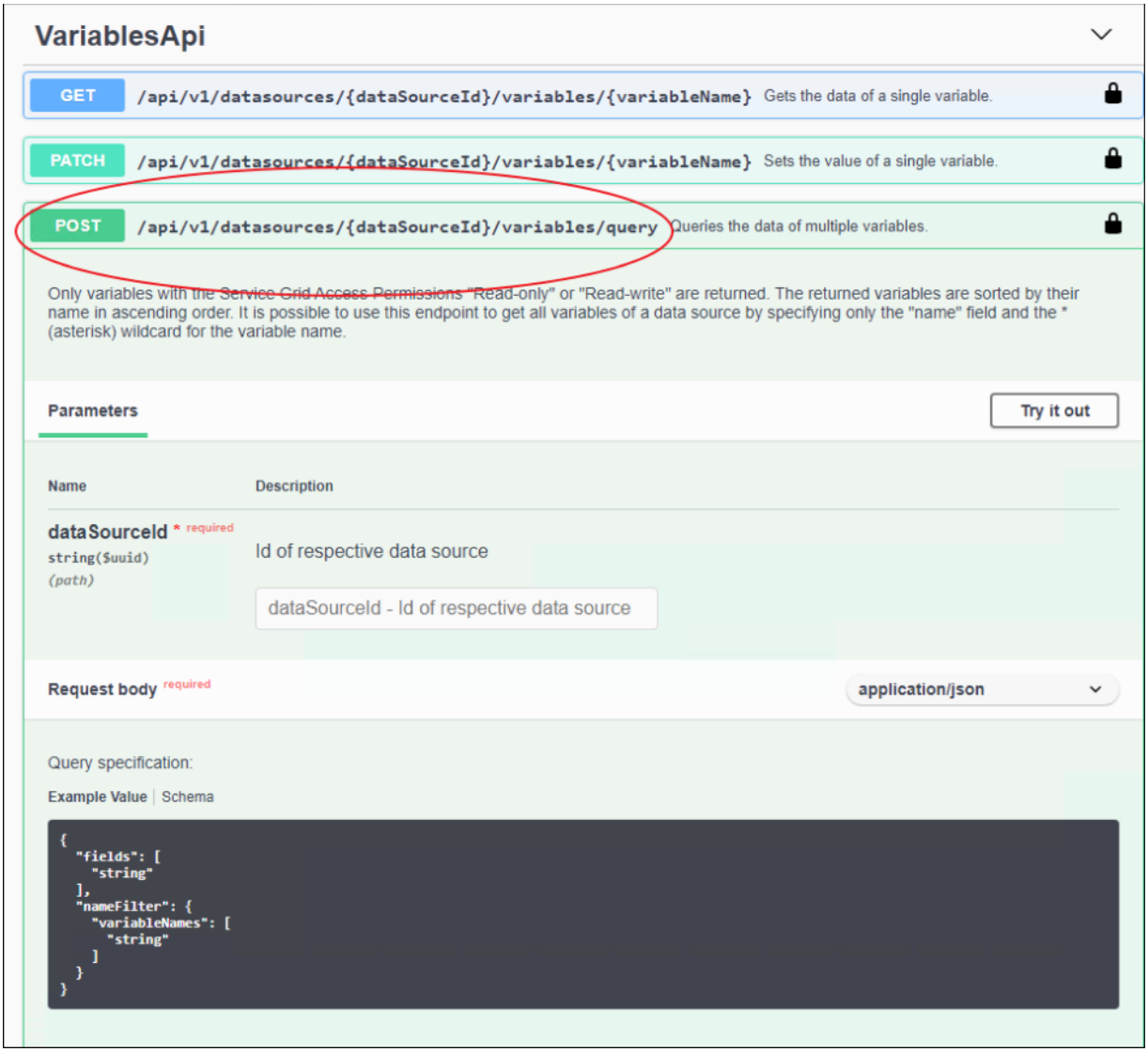
QUERY RESULT

The shared variable and the variable value are in the "Response body" section:

Code Sample:

```
{
"variables": [
{
"name": "ALC_GLOBAL_GROUND",
"value": "1"
}
]
}
```

SCREENSHOT



You must configure the query for this endpoint in the web interface.

23.13 Congratulations!

You have set up Service Grid and checked its functionality. You have succcessfully completed the Getting Started Guide.

You can find further information and detailed use cases in the Service Grid Help.

24 Technical support

BASIC SUPPORT

If you need support for the installation, our employees in Technical Consulting would be happy to help you.

User with basic support can reach the hotline at the following e-mail address: support@copadata.com.

ADVANCED AND PREMIUM SUPPORT

If you have an Advanced or Premium service agreement, please use the telephone number or email address provided in that. Our sales employees (sales@copadata.com) will gladly assist you, if you want to upgrade your free basic service agreement to an Advanced or Premium service agreement.