zenon manual

Project conversion

v.7.20

COPADATA
do it your way

Contents

1. Welcome to COPA-DATA help

GENERAL HELP

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

PROJECT SUPPORT

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

LICENSES AND MODULES

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

2. Project conversion

When using new version of zenon projects which were created with earlier versions must be converted to the new version. If certain properties or changes bust be considered at converting, they are described in the hints for converting in this manual.

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

CONVERTING AT UPDATE AND DOWNGRADE

If you want to update zenon to a higher version, projects are converted automatically to the higher version when opened in the new version. From this time you cannot edit the project with earlier versions of the Editor. However a project backup is created automatically of the converted project in the original version.

Converting a project to an earlier version which was created with a older version (downgrade) is not possible.

BACKWARD COMPATIBILITY

As of version 6.21 projects are downward compatible for the Runtime. You can work in an Editor with a higher version number. This Runtime backwards compatibility is particularly suited for use of mixed versions such as: A project which was engineered with version 6.50 can be used in a Runtime with version 6.22.

> ⚠ **Attention**
>
> *If, in a project with a later version of the Editor, properties are shown that are not available in the earlier version, it can lead to undesired results in the Runtime.*

3. Recommended procedure for converting Runtime files

At the project conversion take care that the Runtime changeable files are also converted correctly.

To convert the Runtime changeable files:

1. Import the files from the Runtime to the original version of the Editor.

2. Carry out the update to the new version.

3. Navigate to the **General** section in project properties.

4. Click on the **RT changeable data** property.

5. The dialog for the settings of the Runtime changeable files is opened.
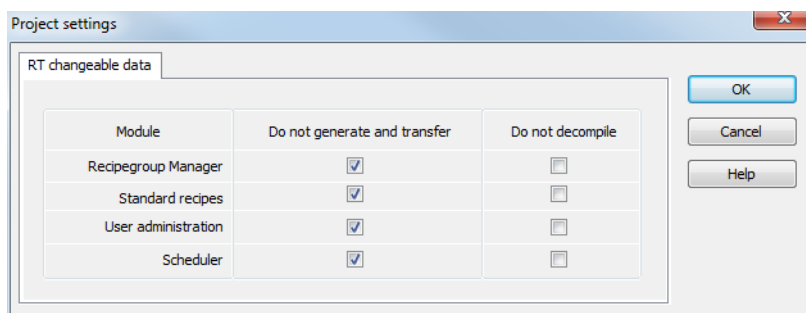
6. Deactivate all check boxes in column `Do not generate and transfer`

7. Compile the project (**Create all Runtime files**)

8. Change the check box in the **RT changeable data** property back to the status before the conversion

4. Converting from version 2.20 to 3.04

4.1 Restructuring the database

1. Create a temporary directory and a subdirectory. Name of the subdirectory: XX. For example: `C:\\TEMP\\XX` where `C:\\TEMP` is the path of the original database and `C:\\TEMP\\XX` is the path for the restructured database. Do not forget to make a backup of your original database.

2. Copy the original database and `REV.BAT` and `DATAMGR.RDL` to `C:\\TEMP`.

3. Copy the new DBD files (of the new database) and the file `DBAREORG.EXE` to the XX directory.

4. Set the paths to the folder, where DBREV.EXE and INITDB.EXE are, or copy these files to `C:\\TEMP` and also to `C:\\TEMP\\XX`. Check if these two programs can be started in the DOS window without entering their path.

5. Make sure, that enough space (min. 500 K) is available in the DOS memory area.

6. Execute the batch file `REV.BAT` in the DOS window:

Conte nts:	del *.taf
	del *.log
	cd xx
	del *.taf
	del *.log
	initdb datamgr
	cd..
	dbrev -r -v -q -s datamgr.rdl datamgr xx/datamgr

7. Start `DBAREORG.EXE` in the directory zenon directly from Windows (execute file).

8. Define the directory where the database is (XX) with CONVERT 2xx -> 3xx and PATH SETTINGS.

9. Start restructuring with CONVERT 2xx -> 3xx and CONVERT.

10. Restructuring is finished. The new database is in the XX path. If existing please delete the TAF files from the XX directory. The original database still is in the TEMP directory.

11. If restructuring is not successful on the first attempt, you have to restart with step 1. Otherwise the restructured database would be converted another time.

Result of restructuring	all screens are converted
	all variables are converted
	all functions are converted, but the functions have to be relinked. Exception: the "Switch to screen" function no longer needs to be relinked.
	a log file (`DBAREORG.LOG`) is created in the folder from which `DBAREORG.EXE` has been started. There you can find information on the unlinked functions.

4.1.1   Restructuring the database when the structures changed

1. Create a temporary subdirectory. Name of the subdirectory: XX

2. Copy the new DBD files to the XX directory.

3. Execute the batch file `REV.BAT`:

Contents:	del *.taf
	del *.log
	cd xx
	del *.taf
	del *.log
	initdb datamgr
	cd..
	dbrev -r -v -q datamgr xx/datamgr

5. Converting from version 3.4, 3.50 to 3.52

5.1 Operating system

Version 3.52 only runs under Windows NT 4.0 and WIN 95.

5.2 AML/CEL-Export

The export name of alarm and CEL entries has changed. Now `Ayymmdd.TXT` or `Cyymmdd.TXT`. For the export there is a new function, which can be executed e.g. with time control. The old settings are omitted.

5.3 AML / CEL, Archives

Now archives for alarm list and CEL are stored as a default. These day archives have no file limitations. For this reason you should engineer a delete function. With this you make sure that not the whole storage capacity of the hard-disk is filled with archives (see function file operations). The available memory for reading archives is limited with the following entry in the zenon6.ini.

[ARCHIV]

…

SPEICHER=1000   (default=1000, that is 1MB)

5.4   Network operation

The drivers have to be set to local. In the section [Befehlsgabe] of the `zenon6.ini` the entry **Treiber=** has to be deleted.

5.5   Archives

Archives now are stored in a subdirectory of the project path. On the first start existing ARV files are moved there.

5.6   Layer

In DXF import layers are no longer supported. Layers now are called symbols.

5.7   Arcs

Arcs have been replaced by segments of circles. Existing arcs retained until changes are made.

5.8   Backward compatibility

Version 3.52 is not downwards compatible. (Concerns e.g. `zenon.ini`, symbols and functions administration)

5.9   Database

A new database structure file is delivered. (`DATAMGR.DBD`,20.11.1996,14.759 byte)

This file is needed for lot archives and has to be copied to the project directory.

Delete *.TAF and *.LOG files in the project directory.

5.10 Backup documentation (QRF inquiries)

In the QRF files the inquiries for VARIABLE and LANGTEXT have been changed.

- ▶ field VARIABLE = dwkanalNr translate using 1 display as left (30);
- ▶ field LANGTEXT = dwkanalNr translate using 6 display as left (39);

5.11 American time formats

As a time format AM/PM is not supported.

5.12 Alarm printer

If the alarm printer fails, the control system now automatically switches to the Windows standard printer. If no print out should happen, the alarm printer in the Editor has to be deleted.

5.13 Symbols

After rescaling symbols have to be resolved and relinked before storing them in the symbol library. Bitmaps in symbols have to be stored separately. Symbol files are saved in folder ../zenon.

5.14 Zoom function

The zoom function may be used to check or edit elements.

5.15 Project backup

A database now consists of the following files:

*.vbf
*.idx
*.DAT
*.BIN
*.dbd
*.INI
*.mdb
*.MIB (for NWM III)
*.NWM (for NWM III)
*.EMS (for 230 EMS)

For saving the archives (*.ARV, *.CEL, *.AML) the according subdirectory has to be regarded.

⚠ **Attention**

The file `ALARM.BIN` *in the archive directory saves the last contents of the alarm list when closing the Runtime and should not be saved.*

5.16   Filter for system windows

We recommend checking the filters for the system screens with the new functionalities (e.g. scheduler, archives for AML/CEL).

5.17   Export variable

For usingthe export file under dBase, Foxpro the max. column width for variable name and identification (long text) has to be limited from 128 characters to max. 100 characters.

Therefore the following `zenon6.ini` entry is necessary:

[EXPORT]

MAX_LAENGE= z. B. 40

5.18 Element Link text

In old projects texts are no longer centered but left aligned. This can be changed in the element parameters. If this leads to extensive changes in existing projects, please proceed in the following way:

1. enter [Editor] Konvert=1 in the `zenon6.ini`

2. Set the option old text elements in the Editor under Open screen; then open the according screen and save it. Repeat this procedure for each screen concerned.

3. delete [Editor] Konvert=1 in the `zenon6.ini`

6. Converting from version 3.52 to 5.11

6.1 Things to be done in version 3.52

6.1.1 Preparatory works

We recommend to create a backup of the project in the old version.

The project database should be checked with the provided tool DBCHECK in version 3.52. If any errors occur during the check, they can be fixed with the also provided tool DBFIX in version 3.52.

6.1.2 Rotated texts

Rotated texts behave differently in version 3.52 and version 5.50 – as far as positioning is concerned – and have to be edited.

In order to get a display compatible to version 3.52, the following entry in the `project.ini` has to be set:

[VERSION]

TEXT352=1

6.1.3   Screen names with special characters

Screen names must not contain special characters such as / / : * ? < > | Ä Ö Ü because the screens are saved as files. The Windows operating system does not allow such special characters.

If existing screen names contain such special characters, these screen names have to be changed in the old version e.g. 3.52, before the screens can be converted to version 5.x.

> ⚠ **Attention**
>
> *Additionally all functions containing invalid screen names have to be changed.*

If these changes are not done, the according screens are not converted and are not available in version 5.x.

> ⚠ **Attention**
>
> If in a 3.5x project there are missing frames or empty allocations, the following error message opens on starting the Runtime 5.11: It was tried to access a file behind the end of the file and the Runtime crashed.
>
> *So before conversion old 3.5x projects have to be checked on missing frames or empty allocations.*

6.2   Things to be done in version 5.11

zenon version 5.11 must be installed.

6.2.1   Licensing

The license numbers of version 5.x are not compatible with the license numbers of previous versions. For updates new license numbers have to be ordered and entered in the Editor. Licenses for all versions up to 3.52 stay untouched, as the new serial numbers are stored in new entries in the `zenon.ini`.

    [DEFAULT]

SERIAL5=xxx

ACTIVATIONKEY5=xxx

  No login name can be stored in the start buttons any more.

An existing Project has to be inserted into a new workspace.

The simulation driver now works with the same driver model as all other drivers. Under Driver/Configuration Hardware has to be selected.

After loading the 3.52 project and converting the screens the correct monitor profile has to be selected and the Editor has to be restarted.

In the `zenon.ini` there is an entry defining the size of the driver buffer. As a rule of thumb we recommend Number of variables * 10.

    [RT]

DRIVER_QUE =50000

If the value is too small, this can result in update problems in the visualization.

We recommend to alarm an overflow. There is a system driver variable for this: Project info/Driver queue.

7.   Converting from version 3.52 to 5.x or 6.x

7.1   Convert dynamic element switch

Switches which were created in a project of version 3.52 must be converted via XML export when switching to a higher zenon version.

Procedure in the new zenon version:

1. Context menu screens -> XML export all

2. Export screens in new file

3. Context menu screens ->Import XML

4. select exported file and import it

8. Converting from version 5.11 to 5.50

8.1 Things to be done in version 5.11

We recommend to create a backup of the project before converting it. The Editor data and the Runtime changeable data have to be saved.

The project database should be checked and repaired if necessary. You will find more information on this on the installation CD under `Software\\TOOLS\\DB_TOOLS\\DB_DOKU.DOC`.

8.2 Things to be done in version 5.50

zenon version 5.50 must be installed.

After starting the Editor the old workspace can be opened. The projects then are converted automatically.

After creating the Runtime files the Runtime can be started.

9. Converting from version 5.50 to 6.01

⚠ **Attention**

Not translated keywords in the language table are deleted during the conversion process.

9.1 Things to be done in version 5.50

Before the converting can be carried out, the data must be read back to the engineering computer. After that the converting may be started.

In the properties window of the project under '`General`' you can find the dialog '`RT changeable data`'. Here you determine which data should be changed. For more information refer to chapter RT changeable data.

9.1.1 Function names

You have to care, that the functions names are unique. As no function names have to be defined in version 5.50, zenon does not check the uniqueness. This is not necessary for version 5.50. In version 6.x however the functions are identified by their names.

9.1.2 Projects

Before a 5.50 project can be opened in the Editor version 6, it has to be converted. (File - Insert project 5.50...). The old project stays unaffected and a copy of it is created in the SQL server. Please be aware, that some external files and directories are `not` automatically added to the new Editor and Runtime directory structure. These files/directories have to be adapted or inserted into the project manually (in the project tree / Files)

This is true for e.g.: user-defined subdirectories in the project directory, extended list directories, databases, export folders, etc....

9.1.3 Runtime changeable project data (users, recipes, etc.)

Have to be read back to the Editor before the conversion, so that no changes done in the Runtime get lost.

9.1.4 Data

The files created by the Runtime such as AML files, CEL files, HD data, bin files etc. are compatible. These have to be copied to the corresponding Runtime directories of version 6 by hand. Please be aware, that these files are converted on the first start of the Runtime. This may take some time for large amounts of data.

9.1.5   Profiles

In version 5.50 the profiles for alarming, Extended Trend, etc. are saved in the file with the name of the project and the extension zrt. Example: For the project Project1 this file is named Project1.zrt. This file has to be renamed to `project.zrt` after having converted the project. Then the file has to be moved to the same folder in the RT path, where also the project.ini is saved.

9.1.6   3S Arti driver (since SP2)

The name of the variable allocation file for the 3S Arti driver has changed. So the file has to be renamed, `before` new variables can be browsed from the PLC. You will find more information in the driver documentation.

9.2   Things to be done in version 6.01

zenon version 6.01 must be installed.

9.2.1   Licensing

The license numbers of version 6.x are not compatible with the license numbers of previous versions. For updates new license numbers have to be ordered and entered in the Editor. Licenses for all versions up to 5.x stay untouched, as the new serial numbers are stored in new entries in the `zenon6.ini` instead of the `zenon.ini`.

9.2.2   User administration / password system

In the user administration of version 6.x there is the new property Administrator. Now only administrators are allowed to edit user data in the Runtime. As this property did not exist in older versions, as a default it is inactive for the existing users. If users still should be able to edit user data in the Runtime, this has to be defined accordingly in version 6.x.

9.2.3   Configuration settings

Additionally all properties of the projects as well as all settings under Options/Settings have to be checked.

As the `zenon.ini` has been replaced by the `zenon6.ini`, probable manual changes (e.g. the entry SPEICHER= for archiving) have to be copied from the `zenon.ini` to the `zenon6.ini` manually.

9.2.4   the Simulator driver

As also the current settings for the simulation driver are stored in the `zenon.ini`, they now are taken from the `zenon6.ini`. After the new installation there are the default settings. So after the installation they have to be edited according to the old `zenon.ini`.

After creating the Runtime files the Runtime can be started.

9.2.5   Screen functions

In version 5.50 and lower, only scripts (but not functions) could be linked to screens. These scripts which were executed automatically when a screen was closed or opened.

In version 6.01 and above, only functions can be linked screens, so all links to scripts are lost. The screens have to be relinked with the desired functions.

10. Converting from version 5.50 to S7 dBase export

When converting a 5.50 project, a converting error may occur. Typical error message:
"`Error driver 'SIMUL32 – SIMULATORDRIVER' variable 'Sub_VISU_E1_IGEF' type for primitive object '34' data type 8 not defined.`"

REASON:

The variables were

> ▸ created on a S7 driver
>
> ▸ exported as dBase
>
> ▸ and then imported on the simulator driver

SOLUTION

> ▸ Export variables with zenon 5.50 as dBase
>
> ▸ in the dBase file, check `Driver` object type and `Data type` and adapt the driver correspondingly

11. Conversion from version 6.01 to 6.20

Projects of the version 6.01 have to be converted when loaded in the current Editor for the first time. A backup of the project is automatically created in the directory `\SQL\Backup`.

> ⚠ **Attention**
>
> *Not translated keywords in the language table are deleted during the conversion process.*

11.1   Archive files

From version 6.20 SP0, archive files are stored in ARX format; before that, the ARV format was used. If a project is converted, a check is made when Runtime is started to see whether there are archive files in ARV format in the Runtime folder. After confirmation of the conversion, the files are converted into ARX format. This conversion is done for all projects, before the projects start. The ARV files are deleted during the conversion. For evacuation to SQL (on page 23), further columns must be added to the table.

Recommendation: Back up the ARV files before conversion.

11.2   Evacuating archives to SQL database

The database table <Project name>_<shortcut> gained two columns.

▶ GUID: char 36. Contains the project GUID of the variable from another project or is ZERO if in the own project.

▶ STRVALUE: varchar. Contains the archived string values. For numerical variables this field has the value ZERO.
The length of the varchar datafield depends on the longest string variable to be archived. The length of the string variable is defined in the variable properties.

The database table <project name>_VARIABLES has one more column.

▶ GUID: char 36. Contains the project GUID of the variable from another project or is ZERO if in the own project.

These new columns either have to be added to the SQL database by hand, or they are added from the Editor. In the Editor you have to switch to the to the property page **Save** in all concerned archives. There

you open the connection string to the database and confirm the dialog. On closing the dialog the changes in the database are done.

> ⚠ **Attention**
>
> *If these changes are not done, no archive data will be evacuated to the SQL database.*

12. Conversion for multi-user projects with different SQL database versions

If multi-user projects are converted to a zenon version with a more recent version of the SQL server, the following process must be noted.

Attention: Multi-user projects can only be converted if no elements are checked out. This means that all engineers have to check in their changes first. If this is not possible for any reason, first a project backup of the project on the project SQL Server has to be done and to be restored immediately. This resets all the `Under construction` information.

> ⚠ **Attention**
>
> *All changes in the local project versions are lost.*

The conversion can only be done on the PC, on which the central project database resides. If this is on a computer in which no Editor is installed (standalone database server), the Editor of the older version must be installed on it first. Only after that can the conversion be done on this PC.

INSTRUCTIONS ON THE UPDATE TO THE NEW VERSION WITH A NEW DATABASE

With this procedure, we assume that the old database is no longer used. In doing so:

- ▶ Files that can be amended in Runtime are taken into account
- ▶ Multi-user clients are removed before the project is backed up

TASKS IN PREVIOUS ZENON VERSION

The following steps are carried out on all client computers for multi-user projects:

1. On all clients, for all objects with the status `Check out`, either `Check in` or `Undo check out`. No object can have the status `Check out`.
2. Delete the respective local projects in the Editor.

The following steps are carried out on the computer with the database for multi-user projects:

1. Optional: Import all changes that have been made in Runtime:

    a) For all modules: Set the status `Allow changes`

    b) Navigate to the project properties **General** -> **RT changeable data** and set all check boxes for **Do not decompile** to `inactive`.

    c) Read all files from Runtime back into the Editor via Remote Transport.

    d) Set the check boxes in **RT changeable data** back to the desired status.

    e) For all modules: Set the status `Accept changes`.

    f) Repeat these steps for all projects.

2. in the context menu of the workspace, select the `Add existing project` command.

3. Navigate to the project, expand the subnode `Connected local project copies`.

4. Carry out the following step for all listed computers, with the exception of the computer with the role of database server:

    • Right click on the connected computer and select `Disconnect connection`.

5. Repeat steps 2 and 4 for all projects.

6. Create a backup of the workspace and a project backup for each project.

TASKS IN NEW ZENON VERSION

The following steps are carried out on the computer with the database for multi-user projects:

1. Install the new Editor

2. Restart the computer
   Note: From this moment, the old database is no longer used.

3. Read back the workspace backup:

    a) Select the same folder as that in which the workspace of the version was located.

    b) Select `Yes` to overwrite the existing workspace.

    c) Select `Yes` to convert all projects.

4. Ensure that port `1433` is not blocked by a firewall or add the SQL Browser to the exceptions for the firewall rules.

5. Ensure that the new SQL server allows remote connections.
   To do this, connect to SQL Server Management Studio and check the properties in the `Connections` section.

The following steps are carried out on all client computers for multi-user projects:

1. Install the new Editor

2. Restart the computer

3. Open the existing workspace.
   In the process, error messages are displayed in the output window.

4. In the context menu of the workspace, select the `Add existing project` command

5. Enter the name of the computer with the database for multi-user projects.

6. Select Refresh.

7. Select the project that is to be added.

   a) Start with the global project.

   b) Then add, if applicable, the integration project.

   c) Then add all other desired projects.

The projects should now be in the workspace and synchronized.

13. Converting from version 5.50 to version > 6.22SP1

When converting a project from version 5.50 to a later version, note the following:

GENERAL

▶ Writing the set value and combined element:

After conversion, the **Write set value** property is activated for all combined elements. Because this property does not exist in version 5.50, the conversion must be manually deactivated again for each combined element after conversion.

▶ Copying projects:

When copying projects in version 5.50, the same GUID are in the `project.ini` file. The first project here would be overwritten in the conversion.

Solution: Delete the INI entry before conversion. This ensures that another GUID number is not issued during conversion.

▶ Switch graphics files to invisible:

If a graphics file in a combined element is to be invisible in a certain status, no graphics file was selected in version 5.50. After conversion, a rectangle is displayed in the combined element, because the **Transparent** property was not activated.

Solution: Activate the **Transparent** property for the graphics file. This enables you to have the same functionality.

▶ Names for screens:

Allowed are: numbers, letters and the following special characters: `$, -, @, _, {}, ~, !, ()`, umlauts and all other special characters in screen names must be removed before conversion.

RECIPEGROUPS AND WINDOWS CE

When converting a project of version 5.50 with active property **Windows CE project** to a version higher than 6.22SP1, recipe groups are not converted to the higher version.

Background: With active property **Windows CE project** the saving type is set to `binary` before the RGM database is converted. Thus no new recipe groups can be created in the new version.

Solution: Deactivate property **Windows CE project** before the conversion.

For this you can either:

▸ open the project in the Editor of version 5.50 and deactivate the property
    or

▸ change the respective entry in the INI file of the project. To do this:

  • open the `projectname.ini` of the project

  • go to area **[RT]**

  • set entry **WIN_CE**=0

14. Converting from version x to 6.21

14.1   zenon web client CAB files are no longer available

The automatic installation of the zenon web clients via CAB file is no longer possible due to security restrictions by Windows 7.

15. Converting from version x to 6.22

15.1   Function authorizations for Acknowledging Alarms

Since zenon version 6.22, the existing function authorizations for acknowledging alarms in the project properties are replaced by three new authorizations. The old function authorization 'Acknowledge alarms' no longer exists.

When converting from existing projects, the new function authorizations are configured according to the old single authorization. For example, if the authorization group 15 used to be necessary for the function authorization 'Acknowledge alarms', it is now also necessary for the three new functions.

For the Editor backward compatibility it may be necessary to transfer the authorization group from the three new function authorizations to the old function. The highest authorization group will be used for that. This means, if the function 'Acknowledge alarm via alarm status line' has authorization group 5, 'Acknowledge alarm via screens' has authorization group 2 and 'Acknowledge alarm via function' has authorization group 12, the function 'Acknowledge alarm' of older versions will receive the authorization group 12.

> **Information**
>
> *Notice regarding online compatibility in the Runtime: If you start a project older than zenon version 6.22 with the current version, the system will offer you the three new functions for configuration. However, only the highest authorization group will be used for processing, according to the mechanism described above. The old project will not be able to use the new functionality.*

15.2   Report Generator function fixed

The function `fixed()` also has the argument `no_seps` in versions up to 5.50. The argument is optional and controls the display of thousand separators.

This argument has no longer an effect on the display as thousands separators are no longer used in zenon.

15.3   VBA - Direct variable access via request is no longer possible

VBA gives our customers a powerful tool for project design. Practical experience has shown that access on variables via VBA often leads to mix-ups in the use of the methods "Advise" via an online container and "Request" with direct call. Too many requests can slow down communication significantly.

We have therefore decided to meet the demands of our customers by closing this error source. From now on, variables can only be addressed via "Advise" in VBA.

For existing projects, this leads to certain incompatibilities. You will be affected if you address variables via "Request" in VBA. In this case you will have to change these projects. After making these adjustments you will profit from an increased communication performance.

NEW:

Direct VBA read access on variables is only possible if the variable has been registered in the driver (advised) and if it has a value, i.e. if IsOnline() is true . Simple spontaneous queries (requests) are no longer possible.

The following functions of the "Variable" object return an error if this is not the case:

- ▸ Value
- ▸ StatusValue
- ▸ StatusExtValue
- ▸ StatusString
- ▸ LastUpdateTime
- ▸ LastUpdateTimeMilliSeconds

This means that existing projects will no longer run in the Runtime after the conversion, as the functions mentioned above will fail.

Possible solutions:

- ▸ Define an online container for all affected variable accesses.
  Advantage: The variables will only be requested if you really need them.
  Disadvantage: Increased programming effort.

- ▸ Set the option "DDE active" in the group "Additional settings" of the variable properties for all affected variables.
  Advantage: Easy to perform.
  Disadvantage: The driver continuously requests all variables.

⚠ **Attention**

Limited compatibility!

After a project conversion to the current version, please check whether there is direct access on variables in any Runtime VBA macro. If this is the case, you must perform the measures mentioned above!

15.4 RGM - database changed

In 6.22 SP1 the format of the used databases for data storage was modified. This means that when converting a project in the Editor the database is modified automatically.

If you changed data of the RGM in the Runtime, you must carry out the following steps:

1. Start the Editor in the original version before you convert the project.

2. In the project properties **RT changeable data** make sure that the data of the RGM can be read back and decompiled.

3. If you use the Runtime on a remote system: Establish a connection to the remote system and read the Runtime files back. You can find this function in the Remote Transport toolbar.

4. Read the Runtime data back to the Editor. Use command `Import Runtime Data` in the toolbar `Runtime files` in order to do so.

5. Close the old Editor and open the new Editor. Now you can convert the project to the new version safely. All data are available in the new version.

> ⚠ **Attention**
>
> *Older RGM database from lower versions than 6.22 SP1 cannot be read in Runtime 6.22 SP1 or higher! If you do not convert the data as described using the Editor, you will lose all changes you made to recipes and recipe groups in the Runtime.*

WINDOWS CE

If you convert a project with activated option **Windows CE project**, the data are saved automatically binary in zenon 6.22. This means that when opening the RGM it looks like all recipes are gone.

15.5 Template - maximum name length

Template names may have a maximum of 29 characters. In a previous version, it was possible to create names with 30 characters. Before converting a project, all template names with 30 characters must be reduced to 29 characters.

16. Converting from version X to version 6.50

Projects from older versions are automatically converted when loaded into the current editor. A backup of the project is automatically created in the directory `\SQL\Backup`.

> ⚠ **Attention**
>
> *Before converting the editable data into Runtime editor, read it back into the old version. Otherwise it will be lost!*

16.1   zenon Logic

When converting from zenon Logic projects, the following must be taken into account:

⚠ **Attention**

The following applies for zenon Logic:

▶ All projects must be recompiled after conversion into Workbench, so that they work in zenon Logic Runtime.

▶ Projects from an older version which have not been converted to version 6.50 cannot be started with Runtime version 6.50.

16.2   Extended graphical settings for AML and CEL

As of version 6.50 property **Extended graphical settings** is available for control element alarm message list in screen Alarm Message List (AML) and control element event list for screen Chronological Event List (CEL).

It activates the use and the customization of the horizontal and vertical scroll bars, the header and the grid for the control element via the corresponding properties in group **Header and grid**.

If projects from earlier version are converted to version 6.5x, property **Extended graphical settings** is missing. To access the property, you must delete control element `Alarm Message List` or `Event List` from the screen and create it again.

16.3   Status bits - new short name

Short names for status bits differ since version 6.20 in the different languages. With zenon version 6.50, common short names were introduced. To ensure compatibility with earlier versions, the short names can be changed to the previous ones in project.ini. In this chapter, you will find:

1. List of new short names including comparison to previous short names

2. List of short names with long name

3. Instruction to reactivate the old short names in project.ini

1. NEW SHORT NAME STATUS

Bit no.	From 6.50 All	Up to 6.50 German	Up to 6.50 English	Up to 6.50 French	Up to 6.50 Italian	Up to 6.50 Spanish	Up to 6.50 Russian
0	M1	ST_M1	ST_M1	ST_M1	ST_M1	ST_M1	ST_M1
1	M2	ST_M2	ST_M2	ST_M2	ST_M2	ST_M2	ST_M2
2	M3	ST_M3	ST_M3	ST_M3	ST_M3	ST_M3	ST_M3
3	M4	ST_M4	ST_M4	ST_M4	ST_M4	ST_M4	ST_M4
4	M5	ST_M5	ST_M5	ST_M5	ST_M5	ST_M5	ST_M5
5	M6	ST_M6	ST_M6	ST_M6	ST_M6	ST_M6	ST_M6
6	M7	ST_M7	ST_M7	ST_M7	ST_M7	ST_M7	ST_M7
7	M8	ST_M8	ST_M8	ST_M8	ST_M8	ST_M8	ST_M8
8	NET_SEL	SELEC	SELEC	SELEC	SELEC	SELEC	ВЫБОР
9	REVISION	REV	REV	REV	REV	REV	РЕВ
10*	PROGRESS	LAUF	DIREC	DIREC	RUN	LAUF	ХОД
11	TIMEOUT	LZÜ	RTE	DRT	RTE	LZÜ	КВИ
12	MAN_VAL	HWERT	MVALUE	VAL_M	V_MAN	VWERT	МЗНАЧ
13	M14	ST_14	ST_14	ST_14	ST_14	ST_14	ST_14
14	M15	ST_15	ST_15	ST_15	ST_15	ST_15	ST_15
15	M16	ST_16	ST_16	ST_16	ST_16	ST_16	ST_16
16	GI	GA	GI	RG	IG	CG	ГО
17	SPONT	SPONT	SPONT	SPONT	SPONT	SPONT	SPONT
18	INVALID	I-BIT	I-BIT	I-BIT	NV-BIT	I-BIT	I-Bit
19	T_CHG_A	SO/WI	SU/WI	ET/HI	ES/IN	VE/IN	ЛТ/ЗМ
20	OFF	N_AKT	N_UPD	N_RAF	N_UPD	N_AKT	N_AKT
21	T_EXTERN	EZ_E	RT_E	HR_E	RT_E	EZ_E	EZ_E
22	T_INTERN	EZ_I	RT_I	HR_E	RT_I	EZ_E	EZ_E
23	N_SORTAB	NSORT	NSORT	NTRI	NORD	NSORT	НЕСОРТ
24	FM_TR	SM_TR	DM_TR	MD_TR	SM_TR	SM_TR	SM_TR
25	RM_TR	LM_TR	RM_TR	MM_TR	RM_TR	LM_TR	СИ_ТР
26	INFO	INFO	INFO	INFO	INFO	INFO	INFO
27	ALT_VAL	EWERT	AVALUE	VALR	RVAL	EWERT	ЕЗНАЧ
28	RES28	RES13	RES13	RES13	RES13	RES13	RES13
29	N_UPDATE	!AKTUAL	!AKTUAL	!AKTUAL	!ATTUALE	!ACTUAL	!АКТУАЛЬНО

30	T_STD	WINTER	WINTER	HIVER	INVERNO	INVIERNO	ЗИМА
31	RES31	RES16	RES16	RES16	RES16	RES16	RES16
32	COT0	UEK0	TCB0	CTB0	CTB0	TCB0	ТРУ0
33	COT1	UEK1	TCB1	CTB1	CTB1	TCB1	ТРУ1
34	COT2	UEK2	TCB2	CTB2	CTB2	TCB2	ТРУ2
35	COT3	UEK3	TCB3	CTB3	CTB3	TCB3	ТРУ3
36	COT4	UEK4	TCB4	CTB4	CTB4	TCB4	ТРУ4
37	COT5	UEK5	TCB5	CTB5	CTB5	TCB5	ТРУ5
38	N_CONF	PN_BIT	PN_BIT	PN_BIT	PN_BIT	PN_BIT	PN_BIT
39	TEST	T_BIT	T_BIT	T_BIT	T_BIT	T_BIT	T_BIT
40	WR_ACK	WR-ACK	WR-ACK	ECR-ACK	WR-ACK	WR-ACK	WR-ACK
41	WR_SUC	WR-SUC	WR-SUC	ECR-OK	WR-SUC	WR-SUC	WR-SUC
42	NORM	NORM	NORM	NORM	NORM	NORM	НОРМ
43	N_NORM	ABNORM	DEVNORM	DEVNORM	Fuori norma	DEVNORM	ОКРНОРМ
44	BL_870	BL_BIT	BL_BIT	BL_BIT	BL_BIT	BL_BIT	BL_BIT
45	SB_870	SB_BIT	SB_BIT	SB_BIT	SB_BIT	SB_BIT	SB_BIT
46	NT_870	NT_BIT	NT_BIT	NT_BIT	NT_BIT	NT_BIT	NT_BIT
47	OV_870	OV_BIT	OV_BIT	OV_BIT	OV_BIT	OV_BIT	OV_BIT
48	SE_870	SE_BIT	SE_BIT	SE_BIT	SE_BIT	SE_BIT	SE_BIT
49	T_INVAL	TIME_INVAL	TIME_INVAL	TIME_INVAL	TIME_INVAL	TIME_INVAL	TIME_INVAL
50	RES50	RES50	RES50	RES50	RES50	RES50	RES50
51	RES51	RES51	RES51	RES51	RES51	RES51	RES51
52	RES52	RES52	RES52	RES52	RES52	RES52	RES52
53	RES53	RES53	RES53	RES53	RES53	RES53	RES53
54	RES54	RES54	RES54	RES54	RES54	RES54	RES54
55	RES55	RES55	RES55	RES55	RES55	RES55	RES55
56	RES56	RES56	RES56	RES56	RES56	RES56	RES56
57	RES57	RES57	RES57	RES57	RES57	RES57	RES57
58	RES58	RES58	RES58	RES58	RES58	RES58	RES58
59	RES59	RES59	RES59	RES59	RES59	RES59	RES59
60	RES60	RES60	RES60	RES60	RES60	RES60	RES60
61	RES61	RES61	RES61	RES61	RES61	RES61	RES61
62	RES62	RES62	RES62	RES62	RES62	RES62	RES62
63	RES63	RES63	RES63	RES63	RES63	RES63	RES63

* Note: Status Nr. 10 was renamed to D_DIREC in version 6.50 and to PROGRESS as of version 6.51.

2. SHORT NAME - LONG NAME

The short names are the same in all languages from version 6.50. The long names remain language-dependent:

Bit number	Short term	Long name	zenon Logic label
0	M1	User status 1	_VSB_ST_M1
1	M2	User status 2	_VSB_ST_M2
2	M3	User status 3	_VSB_ST_M3
3	M4	User status 4	_VSB_ST_M4
4	M5	User status 5	_VSB_ST_M5
5	M6	User status 6	_VSB_ST_M6
6	M7	User status 7	_VSB_ST_M7
7	M8	User status 8	_VSB_ST_M8
8	NET_SEL	Select in the network	_VSB_SELEC
9	REVISION	Revision	_VSB_REV
10	PROGRESS	In operation	_VSB_DIREC
11	TIMEOUT	Runtime exceedance	_VSB_RTE
12	MAN_VAL	Manual value	_VSB_MVALUE
13	M14	User status 14	_VSB_ST_14
14	M15	User status 15	_VSB_ST_15
15	M16	User status 16	_VSB_ST_16
16	GI	General interrogation	_VSB_GR
17	SPONT	Spontaneous	_VSB_SPONT
18	INVALID	Invalid	_VSB_I_BIT
19	T_CHG_A	Daylight saving time/winter time announcement	_VSB_SUWI
20	OFF	Switched off	_VSB_N_UPD
21	T_EXTERN	Real time external	_VSB_RT_E
22	T_INTERN	Realtime internal	_VSB_RT_I
23	N_SORTAB	Not sortable	_VSB_NSORT
24	FM_TR	Error message transformer value	_VSB_DM_TR
25	RM_TR	Working message transformer value	_VSB_RM_TR
26	INFO	Information for the variable	_VSB_INFO
27	ALT_VAL	Alternate value If no value was transferred, the defined alternate value is used otherwise the last valid value is used.	_VSB_AVALUE

28	RES28	Reserved for internal use (alarm flashing)	_VSB_RES28
29	N_UPDATE	Not updated	_VSB_ACTUAL
30	T_STD	Standard time	_VSB_WINTER
31	RES31	Reserved for internal use (alarm flashing)	_VSB_RES31
32	COT0	Cause of transmission bit 1	_VSB_TCB0
33	COT1	Cause of transmission bit 2	_VSB_TCB1
34	COT2	Cause of transmission bit 3	_VSB_TCB2
35	COT3	Cause of transmission bit 4	_VSB_TCB3
36	COT4	Cause of transmission bit 5	_VSB_TCB4
37	COT5	Cause of transmission bit 6	_VSB_TCB5
38	N_CONF	Negative acceptance of Select by device (IEC 60870)	_VSB_PN_BIT
39	TEST	Test bit (IEC870 [T])	_VSB_T_BIT
40	WR_ACK	Writing acknowledged	_VSB_WR_ACK
41	WR_SUC	Writing successful	_VSB_WR_SUC
42	NORM	Normal status	_VSB_NORM
43	N_NORM	Deviation normal status	_VSB_ABNORM
44	BL_870	IEC 60870 Status: blocked	_VSB_BL_BIT
45	SB_870	IEC 60870 Status: substituted	_VSB_SP_BIT
46	NT_870	IEC 60870 Status: not topical	_VSB_NT_BIT
47	OV_870	IEC 60870 Status: overflow	_VSB_OV_BIT
48	SE_870	IEC 60870 Status: select	_VSB_SE_BIT
49	T_INVAL	Time invalid	not defined
50	CB_TRIP	Breaker tripping detected	not defined
51	CB_TR_I	Breaker tripping detection inactive	not defined
52	RES52	reserved	not defined
53	RES53	reserved	not defined
54	RES54	reserved	not defined
55	RES55	reserved	not defined
56	RES56	reserved	not defined
57	RES57	reserved	not defined
58	RES58	reserved	not defined

59	RES59	reserved	not defined
60	RES60	reserved	not defined
61	RES61	reserved	not defined
62	RES62	reserved	not defined
63	RES63	reserved	not defined

ℹ Information

In formulas all status bits are available. For other use the availability can be reduced.

3. ACTIVATION OF OLD SHORT NAMES

If you wish to use the language-dependent short names as in versions before 6.50, you can enable this with an entry in `project.ini`. The first short names apply for:

▶ VBA

▶ Recipegroup Manager

▶ Combined element dialog

▶ Reaction matrix dialog

Note: The short name must be entered in capital letters, must not contain any spaces and must contain a maximum of 7 characters.

THIS IS HOW YOU OPEN PROJECT.INI

1. select the project in Project Manager

2. press shortcut `Ctrl+Alt+E`

3. the SQL folder of zenon opens in the Windows Explorer

   `C:\ProgramData\COPA-DATA\[SQL folder]\[UID]}FILES`

4. navigate to `\zenon\system\`

5. open the file `project.ini` with a text editor.

ASSIGNMENT OF NEW LANGUAGE DEPENDENT SHORT NAMES

Copy the entry of the desired language from `[STATUS]` up to and including `STATUS63=RES63` and paste this into `project.ini`.

German	English	French
[STATUS]	[STATUS]	[STATUS]
STATUS0=ST_M1	STATUS0=ST_M1	STATUS0=ST_M1
STATUS1=ST_M2	STATUS1=ST_M2	STATUS1=ST_M2
STATUS2=ST_M3	STATUS2=ST_M3	STATUS2=ST_M3
STATUS3=ST_M4	STATUS3=ST_M4	STATUS3=ST_M4
STATUS4=ST_M5	STATUS4=ST_M5	STATUS4=ST_M5
STATUS5=ST_M6	STATUS5=ST_M6	STATUS5=ST_M6
STATUS6=ST_M7	STATUS6=ST_M7	STATUS6=ST_M7
STATUS7=ST_M8	STATUS7=ST_M8	STATUS7=ST_M8
STATUS8=SELEC	STATUS8=SELEC	STATUS8=SELEC
STATUS9=REV	STATUS9=REV	STATUS9=REV
STATUS10=LAUF	STATUS10=DIREC	STATUS10=DIREC
STATUS11=LZÜ	STATUS11=RTE	STATUS11=DRT
STATUS12=HWERT	STATUS12=MVALUE	STATUS12=VAL_M
STATUS13=ST_14	STATUS13=ST_14	STATUS13=ST_14
STATUS14=ST_15	STATUS14=ST_15	STATUS14=ST_15
STATUS15=ST_16	STATUS15=ST_16	STATUS15=ST_16
STATUS16=GA	STATUS16=GI	STATUS16=RG
STATUS17=SPONT	STATUS17=SPONT	STATUS17=SPONT
STATUS18=I-BIT	STATUS18=I-BIT	STATUS18=I-BIT
STATUS19=SO/WI	STATUS19=SU/WI	STATUS19=ET/HI
STATUS20=N_AKT	STATUS20=N_UPD	STATUS20=N_RAF
STATUS21=EZ_E	STATUS21=RT_E	STATUS21=HR_E
STATUS22=EZ_I	STATUS22=RT_I	STATUS22=HR_E
STATUS23=NSORT	STATUS23=NSORT	STATUS23=NTRI
STATUS24=SM_TR	STATUS24=DM_TR	STATUS24=MD_TR
STATUS25=LM_TR	STATUS25=RM_TR	STATUS25=MM_TR
STATUS26=INFO	STATUS26=INFO	STATUS26=INFO
STATUS27=EWERT	STATUS27=AVALUE	STATUS27=VALR

STATUS28=RES13	STATUS28=RES13	STATUS28=RES13
STATUS29=!AKTUAL	STATUS29=!AKTUAL	STATUS29=!AKTUAL
STATUS30=WINTER	STATUS30=WINTER	STATUS30=HIVER
STATUS31=RES16	STATUS31=RES16	STATUS31=RES16
STATUS32=UEK0	STATUS32=TCB0	STATUS32=CTB0
STATUS33=UEK1	STATUS33=TCB1	STATUS33=CTB1
STATUS34=UEK2	STATUS34=TCB2	STATUS34=CTB2
STATUS35=UEK3	STATUS35=TCB3	STATUS35=CTB3
STATUS36=UEK4	STATUS36=TCB4	STATUS36=CTB4
STATUS37=UEK5	STATUS37=TCB5	STATUS37=CTB5
STATUS38=PN_BIT	STATUS38=PN_BIT	STATUS38=PN_BIT
STATUS39=T_BIT	STATUS39=T_BIT	STATUS39=T_BIT
STATUS40=WR-ACK	STATUS40=WR-ACK	STATUS40=ECR-ACK
STATUS41=WR-SUC	STATUS41=WR-SUC	STATUS41=ECR-OK
STATUS42=NORM	STATUS42=NORM	STATUS42=NORM
STATUS43=ABNORM	STATUS43=DEVNORM	STATUS43=DEVNORM
STATUS44=BL_BIT	STATUS44=BL_BIT	STATUS44=BL_BIT
STATUS45=SB_BIT	STATUS45=SB_BIT	STATUS45=SB_BIT
STATUS46=NT_BIT	STATUS46=NT_BIT	STATUS46=NT_BIT
STATUS47=OV_BIT	STATUS47=OV_BIT	STATUS47=OV_BIT
STATUS48=SE_BIT	STATUS48=SE_BIT	STATUS48=SE_BIT
STATUS49=TIME_INVAL	STATUS49=TIME_INVAL	STATUS49=TIME_INVAL
STATUS50=RES50	STATUS50=RES50	STATUS50=RES50
STATUS51=RES51	STATUS51=RES51	STATUS51=RES51
STATUS52=RES52	STATUS52=RES52	STATUS52=RES52
STATUS53=RES53	STATUS53=RES53	STATUS53=RES53
STATUS54=RES54	STATUS54=RES54	STATUS54=RES54
STATUS55=RES55	STATUS55=RES55	STATUS55=RES55
STATUS56=RES56	STATUS56=RES56	STATUS56=RES56
STATUS57=RES57	STATUS57=RES57	STATUS57=RES57

STATUS58=RES58	STATUS58=RES58	STATUS58=RES58
STATUS59=RES59	STATUS59=RES59	STATUS59=RES59
STATUS60=RES60	STATUS60=RES60	STATUS60=RES60
STATUS61=RES61	STATUS61=RES61	STATUS61=RES61
STATUS62=RES62	STATUS62=RES62	STATUS62=RES62
STATUS63=RES63	STATUS63=RES63	STATUS63=RES63

Italian	Spanish	Russian
[STATUS]	[STATUS]	[STATUS]
STATUS0=ST_M1	STATUS0=ST_M1	STATUS0=ST_M1
STATUS1=ST_M2	STATUS1=ST_M2	STATUS1=ST_M2
STATUS2=ST_M3	STATUS2=ST_M3	STATUS2=ST_M3
STATUS3=ST_M4	STATUS3=ST_M4	STATUS3=ST_M4
STATUS4=ST_M5	STATUS4=ST_M5	STATUS4=ST_M5
STATUS5=ST_M6	STATUS5=ST_M6	STATUS5=ST_M6
STATUS6=ST_M7	STATUS6=ST_M7	STATUS6=ST_M7
STATUS7=ST_M8	STATUS7=ST_M8	STATUS7=ST_M8
STATUS8=SELEC	STATUS8=SELEC	STATUS8=ВЫБОР
STATUS9=REV	STATUS9=REV	STATUS9=РЕВ
STATUS10=RUN	STATUS10=LAUF	STATUS10=ХОД
STATUS11=RTE	STATUS11=LZÜ	STATUS11=КВИ
STATUS12=V_MAN	STATUS12=VWERT	STATUS12=МЗНАЧ
STATUS13=ST_14	STATUS13=ST_14	STATUS13=ST_14
STATUS14=ST_15	STATUS14=ST_15	STATUS14=ST_15
STATUS15=ST_16	STATUS15=ST_16	STATUS15=ST_16
STATUS16=IG	STATUS16=CG	STATUS16=ГО
STATUS17=SPONT	STATUS17=SPONT	STATUS17=SPONT
STATUS18=NV-BIT	STATUS18=I-BIT	STATUS18=I-Bit
STATUS19=ES/IN	STATUS19=VE/IN	STATUS19=ЛТ/ЗМ
STATUS20=N_UPD	STATUS20=N_AKT	STATUS20=N_AKT
STATUS21=RT_E	STATUS21=EZ_E	STATUS21=EZ_E
STATUS22=RT_I	STATUS22=EZ_E	STATUS22=EZ_E
STATUS23=NORD	STATUS23=NSORT	STATUS23=НЕСОРТ
STATUS24=SM_TR	STATUS24=SM_TR	STATUS24=SM_TR
STATUS25=RM_TR	STATUS25=LM_TR	STATUS25=СИ_ТР
STATUS26=INFO	STATUS26=INFO	STATUS26=INFO
STATUS27=RVAL	STATUS27=EWERT	STATUS27=ЕЗНАЧ

STATUS28=RES13	STATUS28=RES13	STATUS28=RES13
STATUS29=!ATTUALE	STATUS29=!ACTUAL	STATUS29=!АКТУАЛЬНО
STATUS30=INVERNO	STATUS30=INVIERNO	STATUS30=ЗИМА
STATUS31=RES16	STATUS31=RES16	STATUS31=RES16
STATUS32=CTB0	STATUS32=TCB0	STATUS32=ТРУ0
STATUS33=CTB1	STATUS33=TCB1	STATUS33=ТРУ1
STATUS34=CTB2	STATUS34=TCB2	STATUS34=ТРУ2
STATUS35=CTB3	STATUS35=TCB3	STATUS35=ТРУ3
STATUS36=CTB4	STATUS36=TCB4	STATUS36=ТРУ4
STATUS37=CTB5	STATUS37=TCB5	STATUS37=ТРУ5
STATUS38=PN_BIT	STATUS38=PN_BIT	STATUS38=PN_BIT
STATUS39=T_BIT	STATUS39=T_BIT	STATUS39=T_BIT
STATUS40=WR-ACK	STATUS40=WR-ACK	STATUS40=WR-ACK
STATUS41=WR-SUC	STATUS41=WR-SUC	STATUS41=WR-SUC
STATUS42=NORM	STATUS42=NORM	STATUS42=НОРМ
STATUS43=Fuori norma	STATUS43=DEVNORM	STATUS43=ОКРНОРМ
STATUS44=BL_BIT	STATUS44=BL_BIT	STATUS44=BL_BIT
STATUS45=SB_BIT	STATUS45=SB_BIT	STATUS45=SB_BIT
STATUS46=NT_BIT	STATUS46=NT_BIT	STATUS46=NT_BIT
STATUS47=OV_BIT	STATUS47=OV_BIT	STATUS47=OV_BIT
STATUS48=SE_BIT	STATUS48=SE_BIT	STATUS48=SE_BIT
STATUS49=TIME_INVAL	STATUS49=TIME_INVAL	STATUS49=TIME_INVAL
STATUS50=RES50	STATUS50=RES50	STATUS50=RES50
STATUS51=RES51	STATUS51=RES51	STATUS51=RES51
STATUS52=RES52	STATUS52=RES52	STATUS52=RES52
STATUS53=RES53	STATUS53=RES53	STATUS53=RES53
STATUS54=RES54	STATUS54=RES54	STATUS54=RES54
STATUS55=RES55	STATUS55=RES55	STATUS55=RES55
STATUS56=RES56	STATUS56=RES56	STATUS56=RES56
STATUS57=RES57	STATUS57=RES57	STATUS57=RES57

STATUS58=RES58	STATUS58=RES58	STATUS58=RES58
STATUS59=RES59	STATUS59=RES59	STATUS59=RES59
STATUS60=RES60	STATUS60=RES60	STATUS60=RES60
STATUS61=RES61	STATUS61=RES61	STATUS61=RES61
STATUS62=RES62	STATUS62=RES62	STATUS62=RES62
STATUS63=RES63	STATUS63=RES63	STATUS63=RES63

16.4 Structures for UDFBs in zenon Logic

As of version 6.50 in zenon the zenon Logic UDFB data types are no longer available. To preserve the compatibility with version 6.22, you can create them in zenon with a project prefix: "`PROJEKTNAME/UDFBName`". For this to work you must add an entry in file `K5DBXS.ini`:

1. open file `K5DBXS.ini`
2. go to area **[XS]**
3. create entry **UseUDFBPrefix=1**

When you have questions concerning the adaption of invisible UDFBs please contact support@copadata.com or the hotline mentioned in your support contract.

16.5 VSTA and VBA - naming of objects

Some changes to the object model have been made due to limitations in naming VSTA objects. These changes have an effect on VBA code because this continues to access the old name and therefore no longer work.

The following changes have been implemented:

- ► **IDriver**

  - **Name** -> **Identification**:

    The **Name** property does not receive the name returned, but the driver identification. It was renamed in **Identification**.

  - **Driver** -> **Name**:

    Driver is incompatible with VSTA, because this property returns the name of the driver. It is renamed in **Name**.

- ► **IApplication**

- **Close** -> Method: **Close** / Event: **OnClose**:

  IApplication has both a method as well as event with the name **Close.** This is not compatible with VSTA. The event was renamed in **OnClose**.

▶ IZenWorkspace

- **Startup** -> **OnWorkspaceStartup**

- **Exit** -> **OnWorkspaceExit**

  Because both these names are used for VSTA internal events, they were changed to **OnWorkspaceStartup** and **OnWorkspaceExit**.

  > ⚠ **Attention**
  >
  > *Check the VBA code for changed names and adapt it to the new model accordingly.*

17. Converting from version X to version 6.51

Projects from older versions are automatically converted when loaded into the current editor. A backup of the project is automatically created in the directory `\SQL\Backup`.

> ⚠ **Attention**
>
> *Measures to carry out before conversion:*
>
> *Before converting the editable data into Runtime editor, read it back into the old version. Otherwise it will be lost!*

17.1   Calculation column width

As of version 6.51 the average character width of the selected font is used to calculate the column width (e.g. Alarm Message List or CEL). Before that a default value was used. This may cause columns to be displayed in other widths than expected after the conversion.

17.2   Settings SQL database

As of version 6.51 the SQL instance can be defined and the password is saved in an encrypted form in the Startup Tool.

For this the Dialog for setting database properties was changed. New entries have a higher priority than existing entries. The display of the dialog is automatically adopted to the selected version (previous 6.51, as of 6.51).

NEW ENTRIES

`zenondb.ini` contains new entries as of version 6.51:

```
CONNECTION_SQL2005]
USER=zenOnSrv
PW=0x9C 0x94 0xC6 0x50 0x15 0x80 0x79 0x06 0x32 0xED 0x4E 0xE1 0x15 0xDD 0x7C 0x90
SQLINSTANCE=COMPUTERNAME\ZENON_DEV
```

These entries replace entry:

```
[CONFIG]
PROVIDER_SQL2005
```

If the new entries are not available or empty, this entry is still used.

COMPATIBILITY

As long as no property is changed, the existing entries remain valid. If you change an entry for version 6.51 or higher, the new entries are valid. Older versions must be maintained separately. You can find the settings for version previous to 6.51 in chapter Database previous version 6.51.

Attention: As the encrypting of the user password is now taking place in the new dialog in the Startup Tool, as of version 6.51 all settings must be made via the Startup Tool.

17.3   Extended Trend XY axis

Due to performance reasons archive data for the X axis are no longer loaded automatically in the Extended Trend diagram as of version 6.51. To display the X-axis in the diagram anyway:

▶   Add the variable selected for the **variable for X-axis** as well as the curve in the diagram

▶   Deactivate the display for this curve

17.4 GUID for project conversion for version 5.50

At the conversion of projects of version 5.50 take care that when converting several projects an individual GUID is created for each project.

REASON

In the project.ini of version 5.50 there is the entry `GUID` which contains a project GUID. If you convert a project to version 6.x, this GUID is used.

In zenon 5.50 it was possible to copy folders on file level, then open the project in the Editor and rename it. This project copy still contained the original GUID in the INI file which was no problem for 5.50.

If you convert two such projects in version 6.x, the conversion for the second project is canceled with the note that the GUID already exists.

SOLUTION

For converting several copied projects from 5.50 there are three possibilities:

1. Save as
   - Convert project A
   - execute "Save project as" in version 6.x
   - a new GUID is created
   - delete original project
   - convert project B

   Attention: This method is not suitable for global projects.

2. Project backup
   - Convert project A
   - create project backup in version 6.x
   - delete original project
   - Restore backup via "`Restore project backup`" and at that activate property "Create new project".

   Note: This method is also suitable for global projects.

3. editing project.ini
   - open project.ini of version 5.50 In this version "project.ini" is called projectname.ini
   - delete the entry with the `GUID`

- convert

- a new GUID is created

Note: If this project is opened again with version 5.50, a new GUID is also created and entered in the INI file.

17.5   Clickable buttons combined element

As of version 6.51 you can create clickable buttons in the combined element for option `Symbol from library` in any from.

For projects for earlier versions, the property **Symbol form defines the click area** (node: `Display`) is treated as `inactive`.

17.6   Context menus command processing

Previous to version 6.51 text at automatic menu items was ignored. At converting projects which were created with earlier versions than 6.51 Macros $ALL$$NOTE$ are automatically inserted before the engineered text. Therefore the menu items behave as before.

17.7   Record shift times in the PFS

At recording shift times in the Production & Facility Scheduler the table name for the recording was fixed to **PFSSHIFTHISTORY** in version 6.50.

As of version 6.51 it is created after the following pattern: **ProjectGUID_SHIFT_GUID of the equipment group**
For example:
292af0ac-d33d-4123-8484-e359cd0a6ae3_SHIFT_989ef705-d6a6-4b81-9eb5-f76483ecaa c1.

17.8   Convert symbol colors of the general symbol library from color palette to absolute color

When using palettes, only the palette index is saved. The actual color is assigned in Runtime.

PROBLEM

If you define colors for symbols of the `general symbol library` via palettes (similar to function as palette as of 6.51) in versions earlier than zenon 6.51 and the colors of a symbol are changed in version 6.51, all user-defined colors of the symbols are adapted to the change when the Editor is restarted. This action is correct from compatibility's point of view. However absolute colors can be necessary.

SOLUTION

If the symbol library is saved again in zenon 6.51, the palette indices are saved as absolute colors.

Procedure:

1. Activate and open the project with the correct palette in zenon 6.51.

2. Add a new symbol to the general symbol library.

3. Save the general symbol library. It is saved in the new format.

4. Close the Editor and restart it.

5. Rename the newly created symbol and save it.

6. In the global symbol library the palette colors are replaced by the absolute colors.

17.9 Wizards - remove VBA/VSTA properties

At filtering for screen switch functions to a screen of type `Extended Trend`, the following dynamic properties were removed as they no longer have a function:

▶ `PictFilter[0].Curve["0 "].VarInfo.Channel`

▶ and `ArvName`, `Titel`, `Group` and `Amplitude` of the same object

If you use these properties in a wizard, you must remove them.

17.10 Character # not valid in the object name

As of version 6.51 character # is no longer valid for object names such as variables or functions. The character cannot be entered when giving the name via the user interface.

Background: A # in the name may for example cause problems during the import.

18. Converting from version X to version 7.00

AttentionProjects from older versions are automatically converted when loaded into the current editor. A backup of the project is automatically created in the directory \SQL\Backup. The automatically-generated backups have the SQL Server used in the filename from version 7.00 onwards, for example:

before converted to 7.00 SP0 (sql server 2008 r2).zip

> **Attention**
>
> *Measures to carry out before conversion:*
>
> *Before converting the editable data into Runtime editor, read it back into the old version. Otherwise it will be lost!*

Note: Special characters such as ß-/\ are not permitted in zenon project names.   Allowed are: Numbers, letters and the special underscore character (_). You can find more information about this in the `Project administration and workspace` manual, `Create new project` chapter.

18.1   User administration with Active Directory

From version 7.00 SP0 on Active Directory is only available for the user administration in the zenon Runtime. This means for the zenon Editor:

▶   AD users are not used for the Editor.

▶   AD users are no longer validated in the Editor.

▶   Via AD log in to the Editor is not possible.

Attention: If you implemented the log in to the zenon Editor via Active Directory in a project, you must create a zenon user with all necessary rights before you convert the project.

18.2   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 that

1.   Restart the system

2.   On the installation medium, open the path `Additional_Software\COPA-DATA Multiple Network Protocol Driver x64`

3. Execute the file `setup64.exe` (setup32.exe on 32 -bit systems from the path
   `Additional_Software\COPA-DATA Multiple Network Protocol Driver x86`)

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

18.3   Diagnosis Server with new service

With zenon 7.00 SP0 the diagnosis system adapted. From this version on all logging tasks are carried out by service `zenLogSrv`. The service `zenSysServ` is now only responsible for Remote Transport activities. The maximum number of modules per Diagnosis Client was increased from 32 to 64.

That means:

▶ Diagnosis systems up to version 6.51 and from version 7.00 are each compatible among themselves.

▶ The diagnosis mechanism of zenon 6.51 SP0 and zenon 7.00 SP0 are not compatible.

Compatibility	Diagnosis Server 6.51 SP0 and earlier	Diagnosis Server 7.00 SP0 and higher
Diagnosis Client 6.51 SP0 and earlier	`compatible`	**`incompatible`**
Diagnosis Viewer 6.51 SP0 and earlier	`compatible`	**`incompatible`**
Diagnosis Client 7.00 SP0 and higher	**`incompatible`**	`compatible`
Diagnosis Viewer 7.00 SP0 and higher	**`incompatible`**	`compatible`

With the Diagnosis Viewer version 7.00 SP0 and higher you can open log files which were created by Diagnosis Server version 6.51 SP0 (or earlier). It does not work the other way round.

18.4   Dynamic Combo/List Box

As of version 7, combo-/listboxes can also be created dynamically. Via property **Entries from string variable** you define whether the entries in the box are created in the Editor or via a linked string variable.

At converting projects from a version older than zenon 7, the new properties are assigned with valid values:

▶ static combo/listbox (property **Entries from string variable** inactive)

▶ no visibility variable (property **Variable** empty)

If files for an older version are created or saved, the properties for the dynamic comb/listbox are not loaded in order to ensure backward compatibility.

18.5   IPv6

As of zenon version 7 you can use IPv6 in the network.

> 💡 **Information**
>
> With IPv6, more users and devices can communicate via the Internet in that they use greater numbers to create IP addresses. With IPv4, each IP address is 32- bits long, as a result of which 4.3 billion unique addresses can be formed. Example for an IPv4 address:
>
> `172.16.254.1`
>
> For comparison: IPv6 addresses are 128- bits long, which allows formation of approximately 340 sextillion (3,4e+38) unique IP addresses. Example of an IPv6 address:
>
> `2001:db8:ffff:1:201:02ff:fe03:0405`
>
> However IPv6 offers other advantages for network traffic. In most cases, computers and programs recognize IPv6-compatible networks and use the corresponding advantages without the user having to do anything more. IPv6 also frees other network problems that can occur due to the limited addressing area of IPv4. Example: IPv6 reduces the necessity of network address translations (NAT), a service that allows several clients to use a joint IP address, but which does not always work reliably.

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

- ▶ Network configuration in the Startup Tool
  or

- ▶ In zenon6.ini

Attention: IPv6 only works with version 7 onwards.
No versions prior to version 7 can be started if this is active. This concerns zenAdminSrv, zenSysSrv, zenLogSrv and zenDBSrv in particular.

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

- ▹ Driver communication with the PLCs

- ▹ Protocol communication in the Process Gateway plug-ins

- ▹ Workbench and Runtime communication in zenon Logic

DIAGNOSIS VIEWER

The Diagnosis Server also works with Diagnosis Clients which addresses via IPv6 addresses. For this the format of the log file has been adapted. The Diagnosis Viewer only reads the new format of the log files. If files from older zenon versions are opened (or vice versa), the IP address of the Diagnosis Client is not displayed correctly.

18.6   Licensing

With version 7.00 licensing is adapted. To adapt existing zenon version to version 7.00, you must purchase a license for version 7.00 and enter the new activation number. The serial number remains the same.

The entries in zenon6.ini are now:

[DEFAULT]

SERIAL7=

ACTIVATIONKEY7=

18.7   Message Control

As of version zenon 7.00 SP0 module Message Control differs basically from earlier versions in terms of technology and configuration.

Important technical changes:

- ▶ COM Server is no longer used

- ▶ the additional component of company DerDack is no longer used

- ▶ the ZenMsgQueue is replaced by an own screen of type `Message Control`

- ▶ the configuration is carried out in property **Message Control** of the workspace (sending) and properties **Project-specific settings** for module Message Control in the project (project-specific)

- ▶ there is no detail view anymore

- ▶ the shift model and the calendar functionality has been removed

- ▶ there are no Runtime changeable files anymore

- ▶ Sending e-mails is possible via Outlook or a SMTP Server whereas SMTP allows the sending of attachments

- ▶ the configuration of the sending type is no longer saved in file message32.ini but in file zenon6.ini

- ▶ Evaluating the limit texts: Up to now the evaluation of compound texts in module Message Control differed from the evaluation of standard limit texts. From version 7.00 on both are evaluated in the same way. **@StringTabelle+%var1**

You can find details about the configuration in chapter Configure Message Control.

> ⚠ **Attention**
>
> *Only projects from version 5.50 SP7 on can be converted to version 7.*

CONVERSION

Due to the profound changes a 100% compatibility cannot be guaranteed for the conversion. This is also true for compiling RT files for older versions. At converting especially take care for:

▶ User:

- Users with the same name (first name, last name) existing: User is used and information is added.

- No according user available: A new user is created. The link to the replacement and to the user group is resolved. The user is added to the existing or at the conversion to the created group.

▶ User groups:

- User group with same name exists: User group is used and information is added.

- No according user group available: A new user group is created.

▶ Functions:

  `Show recipient-database` function was removed.
  This function can no longer be created with the Editor. At the conversion it is not deleted however. Its call up in the Runtime has no effect and creates a log entry.

▶ Paging:
  Paging is no longer available as sending type. Existing functions with sending type `Paging` are changed to sending type `GSM` at the conversion. A message in the output window indicates this. After the conversion you must check the settings of the function.

▶ RT changeable files:
  Because the user administration was changed, RT changeable files are no longer required for Message Control. There is no possibility in version 7 to read back old Runtime data. If the Runtime files of a project prior to version 7 are needed, you must read them back in an Editor prior to zenon 7 and then converted (on page 9).

▶ Shifts and calendars:
  The functionality for shifts and calendars was removed. Existing functions with target type `Shift` are changed to target type `Group` at the conversion. However no group is linked. A message in the output window indicates this. After the conversion you must check the settings of the function.

▶ SMS-Gateway:
  As the simple interface does not offer a technical possibility to assign messages distinctly, from version 7 on only the enhanced interface is supported. At conversion you must make sure that the SMS Server from company Dialogs is configured correspondingly. Otherwise the sending fails.

18.8 RGM - error behavior at screen switch

If for the screen switch of the RGM a recipe is selected faulty by:

▶ recipe not available in the Runtime

▶ no selection made

▶ recipe not included in the filter

then the behavior of the drop-down list `recipe` changes in the Runtime:

▶ up to version 6.51 SP0 the first recipe in the list is offered

▶ as of version 7.00 SP0 the selection remains empty.

18.9 Read RGM recipe - new operation

As of version 7.00 SP0 at reading in variable values to recipes (teaching) it is checked:

▶ whether the values of the properties `min. value` and max. `value` have been adhered to.

▶ which status the variable has.

If the values are gone below or exceeded, or the variable has the status INVALID, the values are no longer written to the recipe and no longer saved.

Additional system variables (sysdrv.chm::/25964.htm) are analyzed. As of version 7.00 possible variable values:

▶ `0`: Set before the reading and only changes when the reading process is done.

▶ 1: Finished reading successfully.

▶ 2: During reading an error not defined in greater detail has occurred.

▶ 3: During readnig at least one variable hat status INVALID (main.chm::/24148.htm).

▶ 4: At least on value is not within the min-max limits.

▶ 5: During reading a timeout (30000 + 100*VarCount in [ms]) occurred.

18.10   Driver Allan Bradley RS-Linx

From version 7.00 SP0 the driver supports `Unsolicited Messages`. With this the configuration of the driver changes. It is now done on the tab.

▶ `General:` Unchanged
   Note: If you use Unsolicited Messages, you should deactivate **Update time global**

▶ `Configuration` (`allanbnt.chm::://11111.htm`): The previous KT number is replaced by the label of the RS-LINX driver.

▶ `Unsolicited Messages configuration` (`allanbnt.chm::/33547.htm`): New. Setting for Unsolicited Messages.

> ⚠ **Attention**
>
> *In existing projects the driver configuration must be adapted.*

18.11   SQL Server change

With zenon 7.00 SP0, the `Microsoft SQL Server 2008 R2 Express` is installed and used for zenon projects. The zenon Editor only connects to `SQL Server 2008 R2` by default from version 7.00 onwards. Projects that have their databases in a different SQL server instance (such as ZENON_DEV with SQL Server 2005), cannot be opened.

The SQL Server instance that zenDBSrv connects to can be changed using the startup tool. The startup tool sets, in `zenDB.ini`, the corresponding entries for the respective zenon Version:

▶ Before 6.00: no database

▶ 6.00 to 6.20: Entries for SQL Server 2000 (MSDE)

▶ 6.21 to 6.51: Entries for SQL Server 2005 Express Edition
   The password is stored in encrypted form with 6.51

▶ Version 7.00 and later: Entries for the SQL Server 2008 R2 Express Edition with encrypted password

Dual operation of SQL Server 2005 instance "ZENON_DEV" and SQL Server 2008 R2 instance "ZENON_2008R2" is not possible.

> ⚠ **Attention**
>
> *Projects from previous versions of zenon must be imported in the original version an then restored in zenon 7.00.*
>
> *Hint: If no export has been made and the applicable version is no longer available, the transfer can be made manually:*
>
> ▸ Copy the complete folder, including the GUID, to a new location
>
> ▸ Establish the database connection manually

PROJECT CONVERSION

The following procedure is recommended for the conversion of projects from versions prior to version 7.00:

1. Create project backups in the version from which they are to be converted.

2. Export project backups to the hard drive.

3. zenon Editor 7.00 Editor.

4. Create new workspace.

5. Read the project backups into the new workspace.

The project backups can also be read back into the same workspace in the 7.00 Editor. Because the GUID remains the same when a project backup is read back, the workspace in zenon 7.00 and in versions between 6.21 and 6.51 can be opened.

> 💡 **Information**
>
> *Although the projects have the same name and the same GUID, projects from zenon 7.00 are independent from projects from zenon 6.21 to 6.51 due to the different SQL server instance. Backups in zenon 7.00 do not appear in versions 6.21 to 6.51 of the Editor. It is therefore recommended that these are stored in a new workspace.*

19. Converting from version X to version 7.00

Note: Special characters such as ß-/\ are not permitted in zenon project names.   Allowed are: Numbers, letters and the special underscore character (_). You can find more information about this in the `Project administration and workspace` manual, `Create new project` chapter.

19.1   32 and 64-bit version

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

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

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

GENERAL LIMITATIONS

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

CONVERSION OF ZENON 5.50 PROJECTS

No zenon 5.50 projects can be converted with the 64-bit editor. These must be converted beforehand with the 32-bit Editor.

RGM LIMITATION

The Access database is no longer supported in the RGM. In order to be able to use MS Access data from previous versions under 64-bit, the project must first be converted in the 32-bit Editor.   The **DataSource** property is no longer available from version 7.10. For details, see the Converting Recipegroup manager database (on page 59) chapter.

VBA

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

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

```
#if Win64 then
    Declare PtrSafe Function MyMathFunc Lib "User32" (ByVal N As LongLong) As LongLong
```

```
#else

   Declare Function MyMathFunc Lib "User32" (ByVal N As Long) As Long

#end if

#if VBA7 then

   Declare PtrSafe Sub MessageBeep Lib "User32" (ByVal N AS Long)

#else

   Declare Sub MessageBeep Lib "User32" (ByVal N AS Long)

#end if
```

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

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

  ▶   32-bit and 64-bit declares for API calls: http://www.jkp-ads.com/articles/apideclarations.as
      (http://www.jkp-ads.com/articles/apideclarations.as)p

COMPONENTS IN 32-BIT ONLY

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

  ▶   Some programs, such as SIC.exe and DiagViewer.exe

  ▶   Licensing

  ▶   Process Gateway

  ▶   Startup Tool

  ▶   Windows CE

  ▶   Driver

  ▶   zenon Logic Runtime and Workbench

19.2   Installation

zenon 7.10 cannot be installed on systems on which the `Microsoft SQL Server Data Engine` (`MSDE`) is already installed. This affects all systems on which zenon 6.20 or an earlier version has been installed.

19.3   Batch Control

Before converting a project to the new zenon version, all recipes must be ended. Recipes that are running continue to be executed after a restart. The restart only functions within the same zenon version.

19.4   Conversion of projects in versions up to zenon 6.21

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

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

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

19.5   Converting Recipegroup Manager database

From version 7.10, the MS Access database is no longer supported in the Recipegroup Manager. When opening an existing project, the data storage is automatically converted to binary data. A project backup is created in the process. This makes it possible for you to open the project with the version in which it was created.

CONVERSION WITH 64-BIT EDITOR

The 64-bit Editor cannot access the MS Access database. To convert this, open the project in the 32-bit Editor first. There is a mechanism available that with the RGM setting **DataSource**: `MS AccessDB` automatically transfers the data to binary files. The   property **DataSource** is no longer available from version 7.10. If the data storage has already been set to binary files, the database data is rejected. For this, the following applies:

- ▶ Copying the data from the Access database to binary data storage only occurs with conversion in the 32-bit Editor. The data from the Access database is always rejected with 64-bit.

- ▶ When converting under 64-bit, a check is made to see if the data storage of the RGM is set to MS Access. In this case, corresponding information with notification of conversion is displayed in the 32-bit Editor.

- ▶ If, when copying over in the 32-bit Editor, it is established that at the target (binary files) data has already been configured, the user is asked which data is to be kept (MS Access or binary). MS Access and binary data cannot be combined.

After conversion, you can also open and edit the project with the 64-bit Editor.

If you want to convert the project again, use automatically-created backup during the conversion.

AMENDMENT OF RECIPEGROUP NAMES AND RECIPE NAMES FOR 32-BIT ZENON

Recipegroup names and recipe names that contain invalid characters for "binary data" are automatically renamed when converting a project to the 32-bit version of zenon 7.1x.   The renamed elements are shown in the output window. Check the output window for corresponding messages after conversion.

Attention: If recipe groups or recipes are renamed, the following elements must be checked and amended manually in the project:

- ▶ All RGM functions
- ▶ Variables that could contain recipegroup names or recipe names
- ▶ VBA code that could contain recipegroup names or recipe names

19.6   SQL Server 2012

The SQL Server is switched to 2012 Express Edition version with zenon 7.10 SP0. The settings are defined with the Startup Tool.

The correct entry for the zenDB.ini can be found in the zenDB.ini chapter.

> ⚠ **Attention**
>
> *Projects from other versions cannot be opened directly in zenon 7.10. Projects which should be edited in   zenon 7.10 must first be edited in the version in which they were created. The project backup can then be converted in zenon 7.10.*
>
> *If numerous versions are installed in parallel, the various SQL servers must have different paths.*

19.7   Report Viewer

TIME DOMAINS

The Report Viewer can use several time domains from version 7.10 and up. For upwards compatibility the first two configured time filters on the list are used.

DATA SETS

The datasets for AM, CEL, ARCHIVE and ONLINE were supplemented with the `RESOURCESLABEL` entry.

19.8   DirectDraw removed from VBA

All classes, methods and events that are used for direct drawing in zenon screens using the zenon API (VBA/VSTA) were removed for performance reasons. This concerns:

▶   `Draw` class

▶   `DrawAPI` method

▶   `Draw` event

19.9   WPF

The WPF element `CircularGauge` was expanded from 3 to 6 display areas. Backward compatibility is not guaranteed here.

Note: If two versions of a group file are available in a project, the user is asked which version is to be used. No further actions are needed for the maintenance of the versions used up until now. If a newer version is chosen, all corresponding CDWPF files in all symbols and images in all projects must be adapted.

19.10   Time filter

The configuration and display of the time filter and the filter results have been revised, optimized and harmonized.

The combination of the new **Modify time range** option and the new filter behavior can lead to the filter results being different in versions 7.10 from previous versions.

20. Converting from version x to 7.11

Note: Special characters such as ß-/\ are not permitted in zenon project names.   Allowed are: Numbers, letters and the special underscore character (_). You can find more information about this in the `Project administration and workspace` manual, `Create new project` chapter.

20.1 API - change to data type

In the **CreateArrayVarEx** method for the **variables** object, the data type for lAdrMode has been changed from **int** to zenOn.**tpAdrMode**

Existing code must be amended accordingly.

UP TO ZENON 7.10:

public virtual zenOn.**IVariable** CreateArrayVarEx(**string** strName, **object** lpDriver, zenOn.**tpKanaltypes** kTypes, **object** lpVarType, int lLBound, **int** lDimension1, **int** lDimension2, **int** Dimension3, **int** lAdrMode, bool bStartAtNewOffset)

FROM ZENON 7.11

public virtual zenOn.**IVariable** CreateArrayVarEx(**string** strName, **object** lpDriver, zenOn.**tpKanaltypes** kTypes, **object** lpVarType, int lLBound, **int** lDimension1, **int** lDimension2, **int** Dimension3, zenOn.**tpAdrMode** lAdrMode, **bool** bStartAtNewOffset)

20.2 Batch Control

DISPLAY COMMAND TAGS

With version 7.11, in the **Parameter list settings** *tab in screen switching, the options for* ***Display command parameters*** *have been enhanced and switched to radio buttons with as many combinable check boxes as desired.*

In terms of backwards-compatibility, this means that backward-compatible writing is only possible if:

▸ Only one checkbox is set for the changeable parameters

▸ or the combination of all checkboxes lets all parameters through

If the combination of the checkboxes results in a setting that was not previously configurable with zenon 7.10 or earlier, no parameters are displayed in the list.

RESTARTING RUNTIME

When restarting after Runtime has been restarted, the respective status is stored with the information in the recipes. The execution status is also displayed in the unit information. The execution status (numerical and text) in the unit information contains a number and text that corresponds to that of the variables in the screen. Including information on whether triggered by a restart, information on objects with a different status and objects that delay a status change.

Caution: The content of these variables is not compatible between zenon 7.10 and 7.11.

RDL FILES

The **default.rdl** from zenon 7.10 can have configuration errors. If RDL files are used based on default.rdl for Batch Control in zenon 7.11, it can lead to errors on output. If you want to use the correspondent RDL files with zenon 7.11, make the following changes before use:

▶ Data set for **BatchOperationInstances**

   • The field **OpertionType** must be renamed as **OperationType**.

▶ **BatchMasterRecipes** and **BatchControlRecipes**:

   • The field **OpertionType** must be renamed as **OperationType**.

   • Add the data field **MrVersion** with the data type Integer

▶ **BatchPhases1**

   Add the following data fields:

   • **CondPlcError** (data type: String)

   • **CSName** (data type: String)

   • **CSDescription** (data type: String)

   • **CSTag** (data type: String)

   • **ActiveCSNumber** (data type: Integer)

20.3 Command Processing

When creating the Runtime files - via `Create all Runtime files` - all `auto/remote command` and `mandatory command` actions are filtered out.

These actions are only available in zenon from version 7.11. The Editor compilation validation checks whether, for a variable (or variables when using wildcards) the same command or the same switching direction has been configured twice. If this is the case, a corresponding error message appears in the output window.

> 💡 **Information**
>
> The `auto/remote command` and `mandatory command` actions are not available in Runtime versions before version 7.11.

20.4   IMM table names

Tables names are now fixed and issued as follows:

- ▶ **Table for devices**: `Devices`
- ▶ **Table for maint. works**: `MaintenanceWorks`
- ▶ **Table for history**: `MaintenanceHistory`
- ▶ **Table for documents**: `Documents`

These tables have fixed table names as standard.

Different table names are retained when projects are converted. New projects use the fixed, prescribed table names.

20.5   Modify variables via API

In the zenon object model, the **`variable object`** has been supplemented with the **`ModifyVariable`** and **`ModifyArrayVariable`** functions. Properties of simple variables, structure variables and arrays can therefore be edited via the API. All dynamic properties that cannot be changed via the API have been set to "read only".

- ▶ **`ModifyVariable`** function:
  For simple variables. Allows the modification of:
  - • Driver
  - • Channel type
  - • Data type
  - • Addressing mode
  - • **Each data type starts with new offset** setting
- ▶ **`ModifyArrayVariable`** function:
  For array variables. Allows the modification of:
  - • Driver
  - • Channel type
  - • Data type
  - • Addressing mode
  - • Calculation of offset
  - • Lower bound (0 or 1) and dimensions of the array

Attention: In addition, the following properties have been set to READ-ONLY for the API:

> ▶ **LBound**

> ▶ **Dim1**

> ▶ **Dim2**

> ▶ **Dim3**

> ▶ **OfsAccordingType**

> ▶ **IsOffsetManuell**

> ▶ **IsStartAtNewOffset**

> ▶ **Driver**

> ▶ **ID_DataTyp**

ID_DriverTyp

20.6   Lot filter

Filtering for lots has been enhanced and the configuration has been harmonized.

COMPATIBILITY

> ▶ The `Lots` setting in the `Time` tab is now in the `Lots` tab and is called `Display lot selection dialog`.
> Difference: Up to version 7.11, the dialog could be configured in the time filter with the **Lot** selection and the filter could be switched in the `Lot` tab of the filter.
> From version 7.11, only one of the two is possible. When converting a project, the **Display lot selection dialog** option is selected and the time filter is set to its fixed default value of `Relative time filter with one hour`.
>
> > • Note on XML import: When converting a project from version 7.10 to version 7.11 via an XML export and XML import, there is a visual error. The conversion is carried out for the XML import with `absolute time filter with one hour` instead of with `relative time filter with one hour`. This has no effect in Runtime because the setting is not evaluated when the dialog is called up. However when the filter is opened in Runtime or in the Editor, an error message is displayed, because the start time is later than the end time.
>
> ▶ The possibilities for last lots have been expanded. Completed and ongoing lots can be displayed and both can be combined. If the current lots or the combination of current and completed lots are selected and the project is compiled for a version before 7.11, the completed lots are shown in Runtime.
>
> ▶ The following options are available when filtering for archives and lot names:
>
> > • **No filter**: Corresponds to the earlier setting `*` as filter
> >
> > • **Static**:
> >
> > • **From variable**: new setting

When converting from projects created in version 7.10 or earlier, the "*" or empty string setting is converted up as **No filter**.

▶ If Runtime files are created for versions before 7.11, only the options **no filter** and **static** can be converted correctly. With **static**, the string is written as it is; with **no filter** "*" is saved as a filter. The **From variable** option is ignored; no filter is set.

20.7   Network

VALIDATION DURING COMPILATION

The following validation for an evaluated network is undertaken during project compilation:

▶ Used variables still exist

▶ Used variable is not a `string` variable and not a `structure variable`

▶ Hysteresis is within the sum of all weightings

▶ Used variable is not read by standby server.

▶ Variables are only checked if the redundancy mode is `evaluated`.

> 💡 **Information**
>
> A configuration entry is only checked it it is also active.

BACKWARD COMPATIBILITY

▶ COPA-DATA Runtime can connect to a server with version < 7.11.

▶ The COPA-DATA Editor can create a project for COPA-DATA Runtime version < 7.11.

20.8   OPC UA server multi-project compatible

The OPC UA server is multi-project compatible from zenon 7.11 onwards. Variable from the Runtime project and all its subprojects can be selected. In doing so, the object name from the variable name and the project name are combined. Configurations for OPC UA clients that were created before zenon 7.11 are thus not compatible. These must be amended when using zenon 7.11 or higher.

20.9   VBA functions for frame list changed

The functions to query the number of frames has been changed. Up to version 7.10 inclusive, all frames of the local project and the global project have been counted together. From version 7.11, only the frames of the local project or the global project are counted.

The following VBA functions have been changed:

- ▶ `CSchabliste::vba_Count()`: As of version 7.11, only provides the number of local or global frames.

- ▶ `CSchabliste::vba_Item(const VARIANT FAR& vID)`: As of version 7.11, only iterates the global frame list

These changes can lead to incompatibilities with existing projects. If frames are queried in a project using said VBA functions, these must be adapted.

EXAMPLE

There are frames in local project 5 and global project 10. Then:

- ▶ Previously 15 frames were counted and iterated
- ▶ From version 7.11, either 5 or 10 frames are counted and iterated
- ▶ If the iteration goes beyond the respective size, for example 11, this leads to an error
- ▶ Separate queries must be created for the local project and the global project

21. Converting from version x to 7.20

Note: Special characters such as ß-/\ are not permitted in zenon project names.   Allowed are: Numbers, letters and the special underscore character (_). You can find more information about this in the `Project administration and workspace` manual, `Create new project` chapter.

21.1   BACnetNG

The driver was amended for version 7.20.

SEPARATOR

For the **Property seperator** option, the use of the `@` or `#` as a separator is now not permitted. The characters are left in converted projects, but cannot be reentered.

Attention: If one of the two characters is used, no communication takes place.

ADDRESSING

SELECTION OF ADDRESS

The **Use identification to define object name and property** check box in the `Settings` configuration dialog has been replaced with the **Property used for adressing** drop-down list. Addressing using the following methods can be selected:

- ▶ `Variable name`
- ▶ `Identification`
- ▶ `Symbolic address`

The corresponding entry from the drop-down list is set during project conversion.

NEW CHECK BOX FOR STATUS EXTRACTION

The new **Do not read property/event state from address string** check box makes it possible to not extract the `Property-ID` and `Event State` from the selected address.

If this option is activated:

- ▶ The **Property-ID** or **Event State** must be set using the driver-specific properties envisaged for this. This configuration is carried out automatically with online import.

- ▶ Individual elements of BACnet _{Arrays}/lists can no longer be addressed.

- ▶ The variable name cannot be used for addressing, because this must be unique. Addressing must be either by means of identification or symbolic address.

- ▶ The **Property seperator** is not always searched for in the address field. The address thus now only consists of: `<Device name>.<object name>`.
  The following is applicable in the process:

  - The device name must not include a period (.).

  - No restrictions for object names.

21.2 Batch Control: No more grouped CEL entries with a value change in the control recipe

There is no longer a grouped CEL message any more when values change in a control recipe. Instead, each message has all information that concerns the recipe. The header entries have been removed and all changes of parameter values now use the same entries as for editing via the parameter list. The entries for changes were also adapted to the minimum execution time. New entries were added for this.

The following entries were removed:

- ▶ `IDS_STRING2460 "@Engineering of the phase@ %s""%s/%s"" @(column@ %i - @row@ %i@) changed, control recipe@ ""%s""@, master recipe@ ""%s"""`

- ▶ `IDS_STRING2627 "@Reason:@ ""%s"""`

- ▶ `IDS_STRING2624 "  @Value of the parameter@ ""%s"" @was changed. Old value@ ""%s"", @new value@ ""%s"""`

- ▶ `IDS_STRING2628 "  @Minimum duration of execution was changed. Old value@ ""%s"", @new value@ ""%s"""`

The following entries were added:

- ▶ `IDS_STRING2624 "@Minimal processing duration of the phase@ %s""%s/%s"" @(column@ %i - @row@ %i@) changed. Old value@ ""%s"", @new value@ ""%s""@, control recipe@ ""%s""@, master recipe@ ""%s"""`

- ▶ `IDS_STRING2627 "@Minimal processing duration of the phase@ %s""%s/%s"" @(column@ %i - @row@ %i@) changed. Old value@ ""%s"", @new value@ ""%s""@, control recipe@ ""%s""@, master recipe@ ""%s"" @reason@ ""%s"""`

21.3   Screen type specific functions

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

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

21.4   Display sequence of the elements

The display sequence of the individual element components was made the same.

21.5   DirectX 11.1

The following applies when using DirectX 11.1:

- ▶ DirectX 11.1 is available natively under Windows 8 and later versions of the operating system.

- ▶ DirectX 11.1 cannot be used under Windows 7 and Server 200R2 without a Service Pack and versions earlier than this.

- ▶ For Windows 7 SP1 and Server 2008 R2 SP1, a Windows Service Pack must be installed

To install the Service Pack:

1. Download the platform update KB2670838 from the Microsoft support website.

2. Select the version that corresponds to your operating system:

   - 32-Bit

   - 64-Bit

3. Select the corresponding installation file.

   Note: This only concerns Windows 7 SP1 and Server 2008 R2 SP1. DirectX 11.1 is already present on more recent versions. The update cannot be installed on older versions.

21.6   DirectX in the Editor

The native DirectX dsplay is available in the Editor through the settings of the **Graphics quality** property.

There is no dotted line for the grid. It is replaced by a solid line during project conversion.

When a new project is created, the **Graphics quality** is `DirectX hardware` by default. This can be changed manually to `Windows basic` or `DirectX software`. During project conversion, the previous value `Windows Enhanced` is changed to `DirectX hardware`. If a   project that was configured with `Windows Enhanced` is started in Runtime, it is loaded with `DirectX hardware`.

Note: The graphic display of `DirectX` is different from `Windows Enhanced`. You can read details in the Differences between the graphics qualities of "Windows Enhanced" and "DirectX" (on page 77).

If, from version `7.00` a different graphics quality to `Windows Basic` is set for a project (including the corresponding setting for the **Create RT files for**), Runtime up to zenon version `6.51` loads the `Windows extended` graphics quality. If the `Windows basic` graphics quality is set, this is retained.

21.7   DirectX: Graphic display when screen switching

When switching, closing or calling up a screen, the behavior in Runtime when using DirectX has changed. As of now, only the background color of a screen is shown up to the point where it is shown in full or after it is closed.

21.8   DNP3_TG driver replaces DNP3_NG

With version 7.20, the DNP3_TG driver replaces the DNP3_NG driver. The DNP3_NG driver is no longer automatically offered for driver selection. However it can be added manually. (for details, see the driver manual)

When converting projects that use the DNP3_NG driver, this driver list is added to the driver list again.

21.9 Dynamic text: "Text from variable" property replaced

For the configuration of the Dynamic Text screen element, the **Text from variable** check box has been replaced by the new **Display text** drop-down selection.

This allows the selection of the content to be displayed in Runtime from a drop-down list:

▸ `Resources label`: The content of the **Resources label** variable property is displayed.

▸ `Limit text`: As long as no limit value has been breached, the content of the **Text** screen property is displayed. If a limit value is breached, the content of the **Limit text** variable property is displayed.

▸ `Measuring unit`: The content of the **Measuring unit** variable property is displayed.

▸ `Variable identification`: The content of the **Identification** variable property is displayed.

▸ `Variable name`: The content of the **Name** variable property is displayed.

▸ `Variable value`: The value of the variable is displayed.

When converting projects to version 7.20, the property is set according to the status of the check box:

Check box in projects before 7.20	Selection in drop-down list in 7.20
`inactive`	`Limit text`
`active` (variable is numerical)	`Variable value`
`active` (variable is not numerical)	`Limit text`

When compiling Runtime files in version 7.20 for version or 7.11 or lower, the value for the check box is set in the compiled version:

▸ `active`: Selection for drop-down selection is `variable value`

▸ `Inactive`: for all other setups

21.10 Antialiasing property removed

The **Use antialiasing** property was removed from the project properties. **Antialiasing** is now always used if the set graphics quality supports it.

21.11   Text angle and letter slant properties removed

The **text angle** and **letter slant** properties have been removed. These were used with the `static text` element if the script was embedded. **Text angle** can be implemented by rotating the complete element. There is no replacement for **letter slant**.

21.12   Entries in zenon6.ini amended for [AlarmFilterDialog]

In `zenon6.ini`, an entry in the `[AlarmFilterDialog]` section can be used to predefine filter screens for **variable name** and **identification** in the filter dialogs for AML, CEL and filter screens. These entries were amended with version 7.20. Entries that have already been defined thus no longer work.

Because all entered filters are automatically saved on the local computer using this entry in `zenon6.ini`, configuration via the INI file is not necessary. You can find details on the entries that are currently valid in the documentation for `zenon6.ini`.

21.13   Extended Trend: Cross-hair, XY-view

The cross-hair in Runtime for the XY-view is configured in screen switching from zenon 7.20. To do this, activate the **Show cross-hair in XY diagram** option.

Configuration was previously carried out by means of the **SHOW_XY_CROSS**= entry in `project.ini`. When converting a project from a version before zenon 7.20, the setting is read from `project.ini` and entered. From now on, the entry in `project.ini` is ignored.

21.14   Format database table Aggregated archive

From zenon 7.20, the GUID is contained in the index when evacuating archives into an SQL database. The following applies for these: It must be NOT zero. From zenon version 7.20, an empty string is entered instead of ZERO for the evacuation of variables of your own project for the GUID.

If Runtime files for version 7.11 or earlier are compiled, there is a compatibility problem as a result: Because ZERO is written in the GUID column here, the evacuation does not work.

Solution: The table in the SQL Server must be created manually without GUID in the primary key or completely without a primary key.

  For example, with the following syntax:

```
CREATE TABLE [$projectname$_$archivename$]

(
```

```
[VARIABLE] int,

[CALCULATION] int,

[TIMESTAMP_S] int,

[TIMESTAMP_MS] int,

[VALUE] float,

[STATUS] int,

[GUID] varchar(36),

[STRVALUE] varchar(?),

CONSTRAINT [PK_$projectname$_$archivename$] PRIMARY KEY CLUSTERED

(

[TIMESTAMP_S] ASC,

[TIMESTAMP_MS] ASC,

[VARIABLE] ASC,

[CALCULATION] ASC

)

)
```

21.15 "Windows Enhanced" removed

The `Windows Enhanced` graphics quality option was removed from zenon. When a new project is created, the **Graphics quality** is `DirectX hardware` by default. This can be changed manually to `Windows basic` or `DirectX software`. During project conversion, the previous value `Windows Enhanced` is changed to `DirectX hardware`. If a   project that was configured with `Windows Enhanced` is started in Runtime, it is loaded with `DirectX hardware`.

Note: The graphic display of `DirectX` is different from `Windows Enhanced`. You can read details in the Differences between the graphics qualities of "Windows Enhanced" and "DirectX" (on page 77).

If, from version `7.00` a different graphics quality to `Windows Basic` is set for a project (including the corresponding setting for the **Create RT files for**), Runtime up to zenon version `6.51` loads the `Windows extended` graphics quality. If the `Windows basic` graphics quality is set, this is retained.

21.16 Limit text

The limit value text and the status text of a reaction matrix can now be up to 1024 characters long. In the table format for the SQL export of alarms, the TEXT column has been extended to 1024 characters.

The buffer for the dynamic parts of the dynamic limit value texts when using long dynamic limit value texts (`D*.AML` and `D*.CEL`) can now vary, with a maximum size of 1024 bytes. The save format of D*.AML and D*.CEL files has been changed for this.

Existing files remain unchanged as a fixed-length record format with 264 bytes and are not converted. Newly-created files are created with a variable record length in the new format version.

21.17   IMM table names

Tables names are now fixed and issued as follows:

- ▶ **Table for devices**: `Devices`
- ▶ **Table for maint. works**: `MaintenanceWorks`
- ▶ **Table for history**: `MaintenanceHistory`
- ▶ **Table for documents**: `Documents`

These tables have fixed table names as standard. These correspond to the naming in the metering point administration.

Different table names are retained when projects are converted. New projects use the fixed, prescribed table names. If tables are used together in IMM and metering point administration, they must correspond to this convention. Different names in IMM can be amended using a dialog.

21.18   Navigation with Multi-Touch in the worldview

When converting to zenon 7.20 from an earlier versions, the following values are set for the **Move horizontally** and **Move vertically** properties:

- ▶ Screen is bigger than the frame: `Move`
- ▶ Screen is the same size or smaller than the frame: `No reaction`

21.19   Process Control Engine (PCE) removed

Starting from version 7.20, PCE will not be supported anymore and it will not be shown in the module tree of zenon anymore. PCE will not further be developed and documented.

While converting projects from versions lower than 7.20, which contain PCE tasks, the node PCE will be shown for these projects again. PCE can continue to be used for converted projects.

Recommendation: Please use `zenon Logic` instead of PCE.

21.20   Remote Transport: Save password in encrypted form

The password for the Remote Transport connection can be stored in encrypted form in `zenon6.ini` from version 7.20.

Note the following for project conversion:

▸   In the Editor (source system), when converting the project from version X to 7.20, the **KEY**= entry is replaced by **KEYCRYPT**= and the password is saved in encrypted form here.

▸   On the target system, there is a query to see whether **KEY**= in `zenon6.ini` has an entry. Yes: This entry has priority. For conversion, the entry on the target system must be deleted manually.

For details, see the Remote Transport manual, Encryption chapter.

Remote Transport communication can also be encrypted. To do this, the encryption of the zenon network need only be activated in the Startup Tool.

21.21   Frame properties

The frames have new properties. The new properties make it possible to configure the size of a screen when it is called up, as well as the movement of screens on the monitor. In addition, two existing properties have been renamed.

NEW PROPERTIES

Property	Description
Opening size:	Defines the size with with which a screen based on this frame is called up in Runtime.
Value (Opening size)	Value for the selected type of size setting.
Limitation Minimum:	Defines limits for minimum.
Value (minimum):	Value for the selected type of limit.
Move:	Defines possible settings when moving the object.
Limitation:	Defines the type of limit when moving.
Minimum frame margin:	Defines the area that must remain on the monitor if an object is moved beyond the border of the monitor.

When converting a project, the values for the new properties are set to the default values.

RENAMED PROPERTIES

zenon 7.11	zenon 7.20
Width [pixels]:	**Width (maximum) [pixels]:** Defines the maximum width.
Height [pixels]:	**Height (maximum) [pixels]:** Defines the maximum height.

The values are applied during project conversion.

21.22   Frames: "Border type" property changed

The possible settings for the **Border type** were changed for frames:

Up to 7.11	From version 7.20
Bold border	Size adjustable
Thin border	Size fixed
No border	No border

The display in the Editor and Runtime is thus changed. In the Editor, the border of the frame is now shown as it would look in Runtime on the same PC with the same settings in the operating system. The appearance of the frame in converted projects can thus change.

21.23   Frame background color

The background of the frames is now shown in Runtime by the color of the screen that is called up.

21.24   Shading in a combined element

In a combined element, the shading effects act in the same way as for other elements or symbols. Rotations have effects on the actual angle of the shade.

21.25 Differences between the graphics qualities "Windows Enhanced" and "DirectX"

The `Windows Enhanced` graphics quality was removed from zenon, see the "Windows Enhanced" graphics quality removed (on page 73). There is the possibility of pixel differences, and things being displayed differently in the graphics output, as a result of the project conversion.

If the project needs to be amended after project version, note the following differences between the graphics qualities `Windows Enhanced` and `DirectX`:

Differences can occur in the following areas:

- ▶ ClearType of fonts
- ▶ Antialiasing
- ▶ Positioning of displays within elements

  In doing so, note:

  - The dynamic elements and
  - all areas in which text occurs.

- ▶ Dynamics/symbols:

  - Rotate
  - Scaling
  - Move

Also possible differences in:

- ▶ Color gradient
- ▶ Rounding
- ▶ Effects, such as shading
- ▶ Rounded corners
- ▶ End of line
- ▶ Line type
- ▶ Graphics file
- ▶ Zoom in zenon Editor
- ▶ Worldview in zenon Runtime

Note: Differences that are not listed here can also occur in other areas.

21.26   Unlinked elements

COMBINED ELEMENT:

If the main variable is linked, the **combined element** is no longer displayed in Runtime.

TREND ELEMENT:

If no curves have been defined in the Editor, the **trend element** is not shown in Runtime.

CLOCK ELEMENT:

If the **clock element** is configured with the time difference setting, but no variable is linked, it is no longer shown in Runtime.

WPF ELEMENT:

A **WPF element** for which no valid XAML file has been linked is not available in Runtime.

21.27   Variable properties for zenon Analyzer

For use in the COPA-DATA product zenon Analyzer, a new group was added to the variable properties:

The following properties in the zenon   **Analyzer** variable properties group provide information for reports in the zenon Analyzer:

- ▶   **Visual name**: Entry of a display name of the variable in zenon Analyzer. This must be unique in the project. The check is not carried out when issued in zenon, but when imported into zenon Analyzer. If this property is changed after the first export to a zenon Analyzer, these changes are not applied in the zenon Analyzer.

- ▶   **Meaning**: Entry of the (Meaning) of a variable in the zenon Analyzer. Entry is manual or by means of the `Meaning and Waterfall Chart Wizard`. Several meanings are separated by a comma. Syntax: `[Meaning1],[Meaning2],…,[MeaningN]`

- ▶   **Parameter for waterfall diagram**: Parameters of a variable for a waterfall diagram in zenon Analyzer. Entry is manual or by means of the `Meaning and Waterfall Chart Wizard`. The individual parameters are separated by a comma. Several waterfalls are divided by a semicolon. Syntax: `[model name],[row index],[index in row],[color code];`

Up to version 7.11, the entries of the new **Meaning** and **Parameter for waterfall diagram** properties were entered into the **Resources label** property.

When converting a project to version 7.20, the entries are kept in the **Resources label** property. When exporting to zenon Analyzer, both the previous property and the new one are checked. If both are assigned, the entries of the new properties are taken on. Entries that are created using the Meaning and Waterfall Chart Wizard are always entered into the new properties.

21.28   Create zenon Logic projects

In the zenon Editor, no zenon Logic projects with the name `__Global` can be created.
Attention: If there are already projects with the name `__Global`, these are no longer shown.