



New Features 2023-24.00



Technical Inquiries

Please contact our support department!

E-Mail: e3-support@de.zuken.com

Note:

Zuken is not responsible for any errors which may appear in this documentation. Liability, due to direct and indirect losses resulting from the delivery or use of this documentation, is excluded to the extent permitted by law.

This documentation contains copyrighted information.

All rights, especially those pertaining to the duplication and distribution as well as the translation, are reserved. This documentation, whether wholly or in part, may not be reproduced in any form (photocopy, microfilm, etc.), or processed, duplicated or distributed using an electronic system without Zuken's prior written consent.

Contact

Zuken E3 GmbH
Lämmerweg 55
D-89079 Ulm/Einsingen

Phone: +49 7305/9309-0

Fax: +49 7305/9309-99

Web: <http://www.zuken.com>

E-Mail: e3-info@de.zuken.com

Contents

- E3.series Release Notes 1**
- Update Information of Version 2023..... 2
 - Windows Support 2
 - Installation 2
 - Databases 3
 - Projects 3
 - Multi-User Installation 3
 - ORACLE 3
 - Microsoft SQL Server 3
 - Using an Existing Work Environment..... 3
- New Features in 2023 Build 24.00 5
- Overview - New Features in COM-Interface 7
- COM-Interface - Enhancements and Changed Behavior of Interfaces and Functions 8
 - Customer Request: Enhancement to *e3DbeModel - ImportStep* 8
 - Customer Request: Enhancement to *e3Pin - GetColourDescription* 8
 - Enhancement to *e3Device - Create2DView, Create2DViewEx and Create2DViewOfSlots* 9
 - Enhancement to *e3Device - SetPanelLocation and SetPanelLocationEx* 9
 - Enhancement to *e3Dimension - Create and CreateEx* 10
 - Enhancement to *e3Job - GetBomPartList* 10
 - Enhancement to *e3Pin - SetWireType* 10
- COM-Interface - New Interfaces and Functions 11
 - Customer Request: New Function *e3Pin - GetInternalColourDescription* 11
 - Customer Request: New Functions *e3Application - SuppressMessages* and *e3DbeApplication - SuppressMessages* 11
 - Customer Request: New Functions *e3Device - IsPlacedInPanel, GetPanelLocationXYZEulerAngles and SetDisplayModelViewAtPosition* 12
 - Customer Request: New Functions *e3Sheet - GetDisplayModelViewAtPosition and SetDisplayModelViewAtPosition* 14
 - New Function *e3Application - GetCursorPositionInteractively* 15
 - Customer Request: New Function *e3DbeApplication - createDXFfromSTEP* 16
 - New Function *e3DbeModel - DisplayOverviewOfExistingViews* 17
 - New Function *e3Job - UpdateComponentAttributes* 17
- General Changes 19
 - Start Parameter for Changing Interface Language 19
- Valid Language Codes 21
- Database Editor 22
 - Create Connectors without Pins 22
- Importing and Exporting Data 24
 - Importing Data 24
 - Exporting Data 24
 - Display Configuration Files when Importing and Exporting DNG, DWG and DXF Files 24
 - Place Images on Sheets via Copy & Paste and Drag & Drop 24
- Multiuser 25
 - Expert Mode for Installing Multiuser Servers 25

Project Handling	28
Automatically Create Sheet Names with Multiple-Digit Sheet Numbers	28
Update Device Attributes without Changing Other Component Properties.....	29
Retain Manually Changed Device Attributes when Updating or Exchanging Components	32
New Text Type for Cable Ducts.....	34
Search by Component Name	35
Text Token for Description of Wire Colors	37
Panel (also: 3D)	39
Define Opposing Slots and Penetration Depth in 2D Views	39
Adjust Size of STEP Models when Importing	40
Rotate and Tilt Models in Any Dimension	42
Tables 45	
Distinguish Objects with Unassigned Attributes	45
Display Variants/Options in Device and Sheet Tables	47

E³.series Release Notes

Date: 4/27/2023

This manual describes the following functions of version:

2023 **Build** 24.00

!! Please Note when Switching to New Version !!

Projects and databases are converted when opening them with this new **E³.series** Release. Please keep in mind that these projects can no longer be edited with an older version. That's why we recommend backing up projects beforehand.

Note:

- o The latest version of the license server must be installed when using a FlexNet license server.
- o Starting with Microsoft Office Version 2016, the TrueType font **,Arial Unicode MS'** is no longer included in the standard delivery of Microsoft Office due to financial reasons.

This font has been and is still in use for **E³.series**, **E³** template projects and **E³** example projects. You have most likely used this font to documents for **E³** projects. Presently, Zuken is not aware of any comparable functionality that is free-of-charge for **,Arial Unicode MS'**.

To ensure that your new and older projects and the documents generated from them are displayed correctly on devices without older MS Office versions, we recommend you purchase and install the original font **,Arial Unicode MS Regular'** on your devices. Furthermore, we recommend you change your templates to another font style so that new projects no longer use this font.

- **New Features**

See also:

Update Information of Version 2023

- o [Windows Support](#)
- o [Installation](#)
- o [Databases](#)
- o [Projects](#)
- o [Multi-User Installation](#)
- o [ORACLE](#)
- o [Microsoft SQL Server](#)
- o [Using an Existing Work Environment](#)

Windows Support

The following Windows systems are supported with **E3.series** 2023:

- Windows 10
- Windows 11 Pro
- Windows Server 2008/2012/2016/2019

For more information please refer to the installation description.

Installation

Before installing, we recommend you make a backup of your current installation (at least the data and databases).

The installation of **E3.series** 2023 is started using

SETUP.EXE

which can be found on the DVD's main directory.

The Version can be installed in addition to an existing **E3.series** Version.

Note: If internal IT policies prohibit the installation of Microsoft Visual C++ 2008 Redistributable, the following converters must **not** be installed in the installation process:

- Converter 2001-2003
- Converter 2004-2008
- Converter 2009-2011

The converters can be selected in the installation dialog along with the installation parameters. For more information about the installation parameters, refer to the installation help of **E3.series**.

The default directory is: `\Program Files\Zuken\E3.series_<version>`

Therefore, this version can also be used alongside existing versions of **E3.series**. However, note that you cannot use your existing database(s), or the existing projects, in both versions, as they are updated when opened in **E3.series** 2023, so copies should be used for **E3.series** 2023.

Databases

With this new version, a new default database is installed. Existing databases are not overwritten.

The new database contains new components and symbols.

Projects

Existing projects are automatically converted upon opening with the new version. So long as the file has not been saved in the new version, it can still be opened with an older version. After saving a project with **E3.series** Version 2023, a project can no longer be opened with an earlier version.

Multi-User Installation

Existing multi-user projects from Version 2022 can directly be opened with Version 2023. They are automatically converted upon opening. However, this cannot be undone, and the projects can no longer be opened with Version 2022.

That's why we recommend installing Version 2023 multi-user projects in parallel to the existing version, so that it's also possible to still access these project with **E3.series** Version 2022.

To transfer the projects to Version 2023, these projects must be saved in Version 2022 as Single-User projects. They can then be stored in the multi-user environment of Version 2023.

In order to run the Multi-User databases in parallel

- a new SQL Server process must be installed for Microsoft SQL Server or another server must be installed
- a new database user account for ORACLE must be used

The E3 Server process must then access the new database or the new database user account.

ORACLE

When setting up the **E3-MU-Server**, a new name must be specified for the Multi-User Administrator.

Thus a new database user account is created and both **E3.series** versions can work simultaneously in the multi-user environment.

Microsoft SQL Server

First, a new instance of the SQL Server processes must be installed.

Enter a new instance name, which must be used with the set-up installation of the **E3** MU-Server.

The rest of the installation continues as described in the separate **MuSetup** installation description.

Using an Existing Work Environment

Due to the new software component for the user interface, adjustments to the work environment (arrangement of windows, structure of toolbars, own programs added, hidden commands, assignment of keys to commands, ...) cannot be adopted by versions before 2020.

These assignments must be recreated and saved in the working environment.

New Features in 2023 Build 24.00

The following functionality is new in Version 2023 Build 24.00:

- [Overview - New Features in COM-Interface](#)
 - [COM-Interface - Enhancements and Changed Behavior of Interfaces and Functions](#)
 - **Customer Request: Enhancement to e3DbeModel - ImportStep**
 - **Customer Request: Enhancement to e3Pin - GetColourDescription**
 - **Enhancement to e3Device - Create2DView, Create2DViewEx and Create2DViewOfSlots**
 - **Enhancement to e3Device - SetPanelLocation and SetPanelLocationEx**
 - **Enhancement to e3Dimension - Create and CreateEx**
 - **Enhancement to e3Job - GetBomPartList**
 - **Enhancement to e3Pin - SetWireType**
 - [COM-Interface - New Interfaces and Functions](#)
 - **Customer Request: New Function e3Pin - GetInternalColourDescription**
 - **Customer Request: New Functions e3Application - SuppressMessages and e3DbeApplication - SuppressMessages**
 - **Customer Request: New Functions e3Device - IsPlacedInPanel, GetPanelLocationXYZEulerAngles and SetDisplayModelViewAtPosition**
 - **Customer Request: New Functions e3Sheet - GetDisplayModelViewAtPosition and SetDisplayModelViewAtPosition**
 - **New Function e3Application - GetCursorPositionInteractively**
 - **Customer Request: New Function e3DbeApplication - createDXFfromSTEP**
 - **New Function e3DbeModel - DisplayOverviewOfExistingViews**
 - **New Function e3Job - UpdateComponentAttributes**
- [General Changes](#)
 - **Start Parameter for Changing Interface Language**
- [Database Editor](#)
 - **Create Connectors without Pins**
- [Importing and Exporting Data](#)
 - **Display Configuration Files when Importing and Exporting DNG, DWG and DXF Files**
 - **Place Images on Sheets via Copy & Paste and Drag & Drop**
- [Multiuser](#)
 - **Expert Mode for Installing Multiuser Servers**
- [Project Handling](#)

- **Automatically Create Sheet Names with Multiple-Digit Sheet Numbers**
- **Update Device Attributes without Changing Other Component Properties**
- **Retain Manually Changed Device Attributes when Updating or Exchanging Components**
- **New Text Type for Cable Ducts**
- **Search by Component Name**
- **Text Token for Description of Wire Colors**

- **Panel (also: 3D)**
 - **Define Opposing Slots and Penetration Depth in 2D Views**
 - **Adjust Size of STEP Models when Importing**
 - **Rotate and Tilt Models in Any Dimension**

- **Tables**
 - **Distinguish Objects with Unassigned Attributes**
 - **Display Variants/Options in Device and Sheet Tables**

Overview - New Features in COM-Interface

The following new features are available in the COM Interface and described in the individual chapters:

COM-Interface - Enhancements and Changed Behavior of Interfaces and Functions

- **Customer Request: Enhancement to e3DbeModel - ImportStep**
- **Customer Request: Enhancement to e3Pin - GetColourDescription**
- **Enhancement to e3Device - Create2DView, Create2DViewEx and Create2DViewOfSlots**
- **Enhancement to e3Device - SetPanelLocation and SetPanelLocationEx**
- **Enhancement to e3Dimension - Create and CreateEx**
- **Enhancement to e3Job - GetBomPartList**
- **Enhancement to e3Pin - SetWireType**

COM-Interface - New Interfaces and Functions

- **Customer Request: New Function e3Pin - GetInternalColourDescription**
- **Customer Request: New Functions e3Application - SuppressMessages and e3DbeApplication - SuppressMessages**
- **Customer Request: New Functions e3Device - IsPlacedInPanel, GetPanelLocationXYZEulerAngles and SetDisplayModelViewAtPosition**
- **Customer Request: New Functions e3Sheet - GetDisplayModelViewAtPosition and SetDisplayModelViewAtPosition**
- **New Function e3Application - GetCursorPositionInteractively**
- **Customer Request: New Function e3DbeApplication - createDXFfromSTEP**
- **New Function e3DbeModel - DisplayOverviewOfExistingViews**
- **New Function e3Job - UpdateComponentAttributes**

COM-Interface - Enhancements and Changed Behavior of Interfaces and Functions

This chapter provides an overview of which interfaces or functions have been enhanced in **E³.series** 2023 or had their behavior changed.

All enhancements that could be assigned to a customer request, which was documented by the **E³.series** Support in the form of a work item, are listed separately at the beginning of the chapter.

For detailed documentation, refer to the Help integrated in **E³.series**:

- **Customer Request: Enhancement to e3DbeModel - ImportStep**
- **Customer Request: Enhancement to e3Pin - GetColourDescription**
- **Enhancement to e3Device - Create2DView, Create2DViewEx and Create2DViewOfSlots**
- **Enhancement to e3Device - SetPanelLocation and SetPanelLocationEx**
- **Enhancement to e3Dimension - Create and CreateEx**
- **Enhancement to e3Job - GetBomPartList**
- **Enhancement to e3Pin - SetWireType**

Customer Request: Enhancement to e3DbeModel - ImportStep

The behavior of the following function has been enhanced in the **e3DbeModel** interface:

return = e3DbeModel.ImportStep (filename, flags)

- o The parameter **flags** supports the new value **0x0400**.
If the parameter has this value, the size of the model is not updated when a STEP file is imported.

References: Designer-26441 and Designer-29158

Customer Request: Enhancement to e3Pin - GetColourDescription

The behavior of the following function has been enhanced in the **e3Pin** interface:

return = e3Pin.GetColourDescription ()

- o The function can now also output the translations of the colors in the selected project languages using translation abbreviations in the format **&#lt;Number>;** (e.g. **ý** for **red**).
To output the names of the colors in all project languages, the translation abbreviation must be entered in the database editor under **Format** → **Multilingual Descriptions...** → **Wire Colors and Hose/Tube Materials** for each color and language to be translated.
- o The text for every translation abbreviation is output for each language. A line break is additionally output after each language.

References: Designer-17310 and Designer-37373

Enhancement to *e3Device* - *Create2DView*, *Create2DViewEx* and *Create2DViewOfSlots*

The behavior of the following function has been enhanced in the **e3Device** interface:

return = e3Device.Create2DView (modi, name, symbol, position, before)

- o The parameter **modi** supports the new value **0x0400**.
If the parameter has this value, the 2D view of the model is generated with the original orientation of the model.
This means the 2D view does not reflect the rotated orientation of the model.

return = e3Device.Create2DViewEx (modi, name, symbol, position, before, shtid, xMin, yMin, xMax, yMax, scale)

- o The parameter **modi** supports the new value **0x0400**.
If the parameter has this value, the 2D view of the model is generated with the original orientation of the model.
This means the 2D view does not reflect the rotated orientation of the model.

return = e3Device.Create2DViewOfSlots (modi, name, symbol, slotslist, position, before, shtId, xMin, yMin, xMax, yMax, scale)

- o The parameter **modi** supports the new value **0x0400**.
If the parameter has this value, the 2D view of the model is generated with the original orientation of the model.
This means the 2D view does not reflect the rotated orientation of the model.

Enhancement to *e3Device* - *SetPanelLocation* and *SetPanelLocationEx*

The parameter **world** has been removed from the following functions in the **e3Device** interface.

Instead of **world** the function supports the new parameter **flags**:

return = e3Device.SetPanelLocation (flags, shti, x, y, z, rot, use_start_z, place_combined, pivot, shift_key)

- o The new parameter **flags** controls whether models are rotated and pivoted during placement or placed with their original rotation and pivoting.
The following values are available:
0x0001: The model is moved as well as rotated and pivoted according to the parameters **red** and **pivot**.
The model is reset to the original positioning with which it is stored in the database and then rotated and pivoted with the values for **red** and **pivot**.
0x0002: The model is moved and retains the rotation and pivot with which it is placed.
The values for **red** and **pivot** are ignored.

return = e3Device.SetPanelLocationEx (flags, shti, x, y, z, rot, use_start_z, place_combined, pivot, shift_key)

- o The new parameter **flags** controls whether models are rotated and pivoted during placement or placed with their original rotation and tilting.
The following values are available:
0x0001: The model is moved as well as rotated and pivoted according to the parameters **red** and **pivot**.

The model is reset to the original positioning with which it is stored in the database and then rotated and pivoted with the values for **red** and **pivot**.

0x0002: The model is moved and retains the rotation and pivot with which it is placed. The values for **red** and **pivot** are ignored.

Enhancement to *e3Dimension* - *Create* and *CreateEx*

The return values of the following functions have been enhanced in the *e3Dimension* interface:

return = e3Dimension.Create (shtid, x1, y1, x2, y2, flags, distance, text, tx, ty)

- o In the future, the return value 0 will also be output if the function is either applied to a dimension object placed on a locked sheet or called with the module **E³.view** or **E³.viewPlus**.

return = e3Dimension.CreateEx (dimtyp, sheet, x, y, flags)

- o The return value -5 is new and means an error occurred when calling. The return value indicates that the function is applied to a dimension object placed on a locked sheet.
- o The return value -6 is new and means an error occurred when calling. The return value indicates that the function is called with the module **E³.view** or **E³.viewPlus**.

Enhancement to *e3Job* - *GetBomPartList*

The behavior of the following function has been enhanced in the *e3Job* interface:

return = e3Job.GetBomPartList (consumer, outputFormatVersion, flags, keyAttribut, quantityAttribut, lengthAttribut, additionalAttributes, result)

- o The parameter **flags** supports the new value **0x0100**. If the parameter has this value, special characters are masked that are used for multi-instance attributes, apostrophes and separators. Therefore the masking of special characters complies with the RFC 4180 standard.

Enhancement to *e3Pin* - *SetWireType*

The behavior of the following function has been enhanced in the *e3Pin* interface:

return = e3Pin.SetWireType (comp, name)

- o The return value -1 is new and means an error occurred when calling. The return value indicates that the parameter **name** is an empty string and therefore the change to the wire group could not be made.

COM-Interface - New Interfaces and Functions

This chapter provides an overview of which interfaces or functions that are new in **E³.series** 2023.

All enhancements that could be assigned to a customer request, which was documented by **E³.series** support in the form of a work item, are listed separately at the beginning of the chapter.

For detailed documentation, refer to the Help integrated in **E³.series**:

- **Customer Request: New Function e3Pin - GetInternalColourDescription**
- **Customer Request: New Functions e3Application - SuppressMessages and e3DbeApplication - SuppressMessages**
- **Customer Request: New Functions e3Device - IsPlacedInPanel, GetPanelLocationXYZEulerAngles and SetDisplayModelViewAtPosition**
- **Customer Request: New Functions e3Sheet - GetDisplayModelViewAtPosition and SetDisplayModelViewAtPosition**
- **New Function e3Application - GetCursorPositionInteractively**
- **Customer Request: New Function e3DbeApplication - createDXFfromSTEP**
- **New Function e3DbeModel - DisplayOverviewOfExistingViews**
- **New Function e3Job - UpdateComponentAttributes**

Customer Request: New Function *e3Pin* - *GetInternalColourDescription*

The following function has been added to the **e3Pin** interface for querying whether the description of a wire color uses a translation abbreviation or free text:

```
return = e3Pin.GetInternalColourDescription ( )
```

Return Values:

- o **return "Color Description"**: Successfully called the function - the function returns either the used translation abbreviation in the format **&#lt;Number>**; or the free text description of the color.
- o **return "Several Values"**: Failed calling the function - the color description is different depending on the option used.
- o **return <Empty String>**: Inconclusive - the wire has no color or an error occurred.

References: Designer-17310 and Designer-37373

Customer Request: New Functions *e3Application* - *SuppressMessages* and *e3DbeApplication* - *SuppressMessages*

The following functions have been added to the **e3Application** and **e3DbeApplication** interfaces for suppressing all messages in **E³.series** that are not sent by another function from **e3Application** or **e3DbeApplication**:

```
return = e3Application.SuppressMessages ( suppress, flags )
```

Parameter:

- o **suppress[in]**: Specifies whether messages are suppressed in **E³.series**.
If the value is **true**, only messages sent by the following functions are displayed:
 - o **e3Application.PutError**

- o **e3Application.PutErrorEx**
- o **e3Application.PutInfo**
- o **e3Application.PutInfoEx**
- o **e3Application.PutMessage**
- o **e3Application.PutMessageEx**
- o **e3Application.PutWarning**
- o **e3Application.PutWarningEx**
- o **e3Application.PutVerify**
- o **e3Application.PutVerifyEx**
- o **flags** [in] [optional]: An optional parameter with which the function can be enhanced. Presently, the parameter does not support any values.

Return Values:

- o **return 1**: Successfully called the function - message suppression was enabled or disabled according to the value of the variable **suppress**.
- o **return -1**: Failed calling the function - the script with the function was executed outside the script transaction. This is the case, for example, when more than one script is executed at the same time.

return = e3DbeApplication.SuppressMessages (suppress, flags)

Parameter:

- o **suppress** [in]: Specifies whether messages are suppressed in **E3.series**. If the value is **true**, only messages sent by the following functions are displayed:
 - o **e3DbeApplication.PutError**
 - o **e3DbeApplication.PutErrorEx**
 - o **e3DbeApplication.PutInfo**
 - o **e3DbeApplication.PutInfoEx**
 - o **e3DbeApplication.PutMessage**
 - o **e3DbeApplication.PutMessageEx**
 - o **e3DbeApplication.PutWarning**
 - o **e3DbeApplication.PutWarningEx**
 - o **e3DbeApplication.PutVerify**
 - o **e3DbeApplication.PutVerifyEx**
- o **flags** [in] [optional]: An optional parameter with which the function can be enhanced. Presently, the parameter does not support any values.

Return Value:

- o **return 1**: Successfully called the function - message suppression was enabled or disabled according to the value of the variable **suppress**.

Reference: Designer-28711

Customer Request: New Functions e3Device - IsPlacedInPanel, GetPanelLocationXYZEulerAngles and SetDisplayModelViewAtPosition

The following functions have been added to the **e3Device** interface for querying whether models are placed in the panel, or for querying and specifying the Euler angles of placed models:

return = e3Device.IsPlacedInPanel ()

Return Values:

- o **return 1**: Successfully called the function - the model is placed in the panel.
- o **return 0**: Successfully called the function - the model is not placed in the panel.
- o **return -1**: Failed calling the function - no project opened or no model ID specified.

return = e3Device.GetPanelLocationXYZEulerAngles (x, y, z, xrot, yrot, zrot)

Parameter:

- o **x [out]**: Specifies the X coordinate of the model placed in the panel.
- o **y [out]**: Specifies the Y coordinate of the model placed in the panel.
- o **z [out]**: Specifies the Z coordinate of the model placed in the panel.
- o **xrot [out]**: Specifies the Euler angle of rotation around the X axis.
- o **yrot [out]**: Specifies the Euler angle of rotation around the Y axis.
- o **zrot [out]**: Specifies the Euler angle of rotation around the Z axis.

Return Values:

- o **return 1**: Successfully called the function - the coordinates and Euler angles of the model are output.
- o **return 0**: Successfully called the function - the model is not placed in the panel.
- o **return -1**: Failed calling the function - no project opened or no model ID specified.

return = e3Device.SetPanelLocationXYZEulerAngles (flags, shti, x, y, z, xrot, yrot, zrot, use_start_z, place_combined, shift_key)

Parameter:

- o **flags [in]**: A parameter that depends on the placement of the model.
The following values are possible:
0: An unplaced model is placed for the first time.
1: A placed model is moved.
- o **shti [in]**: Specifies the sheet ID.
- o **x [in]**: Specifies the X coordinate of the model placed in the panel.
- o **y [in]**: Specifies the Y coordinate of the model placed in the panel.
- o **z [in]**: Specifies the Z coordinate of the model placed in the panel.
- o **xrot [in]**: Specifies the Euler angle of rotation around the X axis.
- o **yrot [in]**: Specifies the Euler angle of rotation around the Y axis.
- o **zrot [in]**: Specifies the Euler angle of rotation around the Z axis.
- o **use_start_z [in] [optional]**: Specifies whether the value of **z** is used as the start coordinate for placement.
The following values are possible:
0: **z** is not used as start coordinate.
1: **z** is used as start coordinate.
- o **place_combined [in] [optional]**: Specifies whether the model is placed as an assembly with all connectors connected.
The following values are possible:
0: The model is not placed as an assembly.
1: The model is placed as an assembly with all connectors connected.
- o **shift_key [in] [optional]**: Specifies whether the model is rotated automatically.
The following values are possible:
0: The model is rotated automatically.
1: The model is not rotated automatically.

Return Values:

- o **return 0**: Successfully called the function - the model is placed with the coordinates and Euler angles.
- o **return -1**: Failed calling the function - no project opened or no model ID specified.
- o **return -2**: Failed calling the function - the model or sheet is locked.
- o **return -3**: Failed calling the function - all variants are displayed.
- o **return -4**: Failed calling the function - the value for **shti** is not valid.
- o **return -5**: Failed calling the function - the sheet has no area.
- o **return -6**: Failed calling the function - the normal vector of the sheet does not fit the Z-axis.
- o **return -7**: Failed calling the function - an error occurred.

Reference: Designer-17956

Customer Request: New Functions *e3Sheet* - *GetDisplayModelViewAtPosition* and *SetDisplayModelViewAtPosition*

The following functions have been added to the **e3Sheet** interface for querying whether a view position was selected for 2D views of models from which the respective view was generated, or to specify the view position for the view:

return = e3Sheet.GetDisplayModelViewAtPosition (view_distance, flags)

Note: 2D views of models with view positions can only be created if the model has slots.

Parameter:

- o **view_distance** [out]: Specifies the numerical value of the view position used for the 2D view.
- o **flags** [in] [optional]: An optional parameter with which the function can be enhanced. The parameter currently supports only the value 0.

Return Values:

- o **return 1**: Successfully called the function - the 2D view of the model was created with a defined viewing position. The position is specified with the parameter **view_distance**.
- o **return 0**: Successfully called the function -the 2D view was created without setting a viewing position.
- o **return -1**: Failed calling the function - no project opened.
- o **return -2**: Failed calling the function - no sheet selected.
- o **return -3**: Failed calling the function - the value for **flags** is not valid.
- o **return -4**: Failed calling the function - the selected sheet is not a 2D view.

return = e3Sheet.SetDisplayModelViewAtPosition (display_model_view_at_position, view_distance, flags)

Parameter:

- o **display_model_view_at_position** [in]: Specifies whether and which viewing position the 2D view has. The following values are possible:
Values > 0: Generates a viewing position with the value specified with **view_distance**.
0: Removes the viewing position of the 2D view.

- o **view_distance** [in]: Specifies the distance the viewing position has.
- o **flags** [in] [optional]: An optional parameter with which the function can be enhanced. The parameter currently supports only the value 0.

Return Values:

- o **return 1**: Successfully called the function - the viewing position was generated or reset.
- o **return -1**: Failed calling the function - no project opened.
- o **return -2**: Failed calling the function - no sheet selected.
- o **return -3**: Failed calling the function - the value for **flags** is not valid.
- o **return -4**: Failed calling the function - the selected sheet is not a 2D view.
- o **return -5**: Failed calling the function - the model has no slot.
- o **return -6**: Failed calling the function - the value for **view_distance** is not valid.

Reference: Designer-14139

New Function e3Application - GetCursorPositionInteractively

The following function has been added to the **e3Application** interface for querying the real time position of the mouse cursor:

return = e3Application.GetCursorPositionInteractively (xpos, ypos, flags, keysAndMouseButtons, grid, gridX, gridY, boxWidth, boxHeight)

Note: The function is not supported in the panel in 3D.

Parameter:

- o **xpos** [out]: Specifies the X coordinate of the position where the mouse cursor is located.
- o **ypos** [out]: Specifies the Y coordinate of the position where the mouse cursor is located.
- o **flags** [in] [optional]: An optional parameter with which the function can be enhanced. The following value is possible:
 - 0x0001**: The position of the mouse cursor is output.
- o **keysAndMouseButtons** [out] [optional]: Indicates which key was pressed on the mouse or keyboard. The following values are possible:
 - 0x0001**: The left mouse button was pressed.
 - 0x0002**: The right mouse button was pressed.
 - 0x0004**: The middle mouse button was pressed.
 - 0x0008**: The Shift key was pressed.
 - 0x0010**: The Ctrl key was pressed.
- o **grid** [out] [optional]: Specifies the name of the sheet and position of the mouse cursor in the grid of the sheet in the following format: /<Sheet>.<Grid>
- o **gridX** [out] [optional]: Specifies the position of the mouse cursor in the grid of the sheet.
 - gridX** Indicates in which column of the grid the mouse cursor is located.
- o **gridY** [out] [optional]: Specifies the position of the mouse cursor in the grid of the sheet.
 - gridY** Indicates in which row of the grid the mouse cursor is located.
- o **boxWidth** [in] [optional]: Specifies the width of an optional rectangle that is displayed at the mouse cursor.

The rectangle can be used, for example, to better visually estimate the exact position of the mouse cursor.

- o **boxHeight** [in] [optional]: Specifies the height of an optional rectangle that is displayed at the mouse cursor.

The rectangle can be used, for example, to better visually estimate the exact position of the mouse cursor.

Return Values:

- o **return > 0**: Successfully called the function - gives the position of the mouse cursor and the optional information about the position of the grid, the pressed keys and the rectangle displayed at the mouse cursor.
- o **return 0**: Failed calling the function - the call was terminated because the Esc key was pressed.

Customer Request: New Function *e3DbeApplication - createDXFfromSTEP*

The following function has been added to the *e3DbeApplication* interface for generating a DXF file on the basis of a STEP file:

return = e3DbeApplication.createDXFfromSTEP (stepFile, outputDirectory, dxfVersion, views, color, flags)

Parameter:

- o **stepFile** [in]: Specifies the name of the STEP file.
- o **outputDirectory** [in]: Specifies the directory where the DXF file to be generated is saved.
Note: If there is already a DXF file with the same name in the directory, the existing DXF file will be overwritten.
- o **dxfVersion** [in]: Specifies the DXF version with which the DXF file is generated. Directory in which the DXF file to be generated is saved..
The following values are possible:
11
13
14
2000
2004
2007
2008
2009
2010
2013
2018
- o **views** [in] [optional]: Optionally specifies the views of the STEP file that are displayed in the DXF file.
The following values are possible:
0x01: front
0x02: back
0x04: right
0x08: left
0x10: top
0x20: bottom
If no value is specified, the front view is generated by default.

- o **color** [in] [optional]: Optionally specifies the color that graphics have in the DXF file. The parameter supports integer values between **-1** and **255**. **-1** stands for automatic color selection. The colors for the values **0** to **255** are shown in the documentation of the API in the chapter **Common Methods** → **Parameters** → **Colors**.
- o **flags** [in] [optional]: An optional parameter with which the function can be enhanced. Presently, the parameter does not support any values.

Return Values:

- o **return 0**: Successfully called the function - the DXF file was generated.
- o **return -1**: Failed calling the function - the STEP file **stepFile** does not exist or cannot be read.
- o **return -2**: Failed calling the function - the directory **outputDirectory** does not exist or cannot be created.
- o **return -3**: Failed calling the function - **views** describes valid views.
- o **return -4**: Failed calling the function - **colors** does not describe a valid color value.
- o **return -5**: Failed calling the function - the value for **flags** is not valid.
- o **return -6**: Failed calling the function - an error occurred.
- o **return -7**: Failed calling the function - the parameter **dxfVersion** and the corresponding DXF version are not supported.

References: Designer-26441 and Designer-29158

New Function e3DbeModel - DisplayOverviewOfExistingViews

The following function has been added to the **e3DbeModel** interface for displaying all existing 2D views of a model in the database editor:

return = e3DbeModel.DisplayOverviewOfExistingViews (flags)

Parameter:

- o **flags** [in] [optional]: An optional parameter with which the function can be enhanced. Presently, the parameter does not support any values.

Return Values:

- o **return 1**: Successfully called the function - the 2D views of a model are displayed. The 2D view **Front** of the selected model is selected as the active view.
- o **return -1**: Failed calling the function - no project opened or no model selected.
- o **return -2**: Failed calling the function - no valid view is active.
- o **return -3**: Failed calling the function - the value for **flags** is not valid.
- o **return -4**: Failed calling the function - the selected sheet is not a 2D view.

New Function e3Job - UpdateComponentAttributes

The following function has been added to the **e3Job** interface for updating the attributes of specific component versions of devices used in the project:

return = e3Job.UpdateComponentAttributes (name, version)

Parameter:

- o **name** [in]: Specifies the component name.
- o **version** [in] [optional]: An optional parameter with which the function can be enhanced.
Specifies a component version.
If **version** is a valid version of the component **name**, all attributes of the component are updated in the specified version.
If **version** is not specified, all versions of the component are updated.

Return Values:

- o **return 1**: Successfully called the function - the attributes of the component have been updated.
- o **return 0**: Failed calling the function - an error occurred.
- o **return -1**: Failed calling the function - the component **name** does not exist in the database.
- o **return -2**: Failed calling the function - the component **name** is not used in the project.
- o **return -3**: Failed calling the function - the component with the specified version is not used in the project.

General Changes

- **Start Parameter for Changing Interface Language**

Start Parameter for Changing Interface Language

A new start parameter is available in **E3.series** that enables the interface language to be changed.

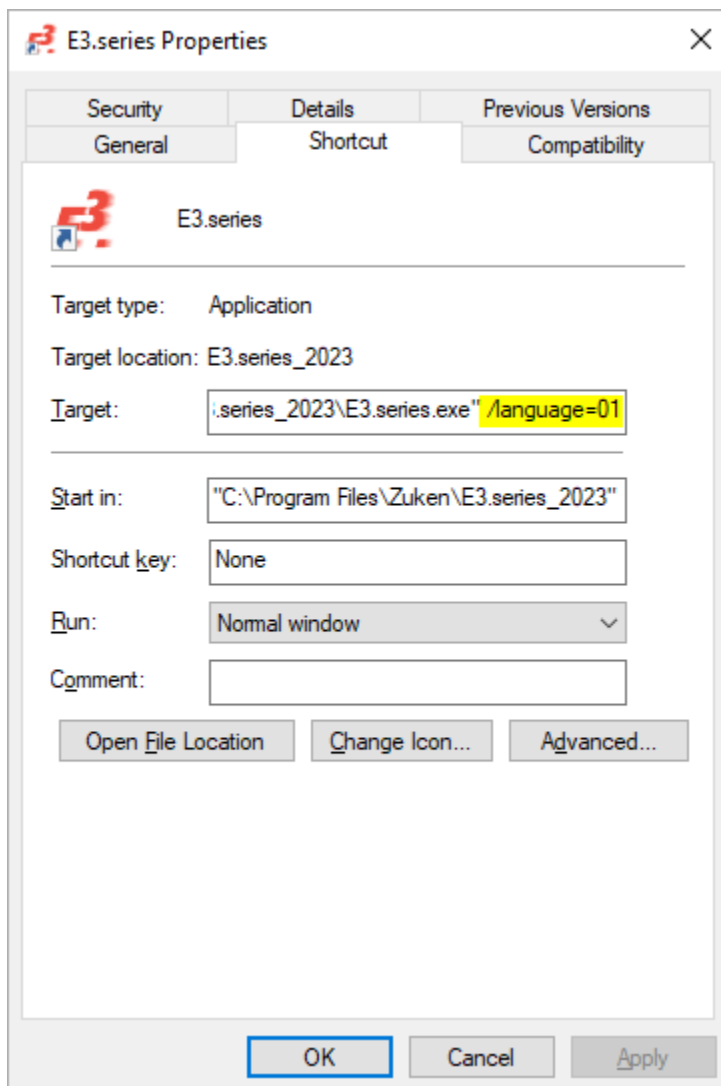
This makes it possible to change the interface language even if the user has no rights to change the registry.

This reduces the risk of accidentally making the wrong changes in the registry and also simplifies the procedure for changing the language.

To use a start parameter for the interface language of **E3.series**, first create a link to the file `E3.series.exe`.

Then open the properties of the link and specify the parameter `/language`, an equal sign (=) and a valid **Language Code**.

Example:



The language code used for the start-up parameter is treated preferentially and overrides the language definition set in the registry.

Note: If the language code of the start-up parameter or the registry is invalid or no language is defined, **E3.series** is started in British English.

Valid Language Codes

The following language codes are available:

Language	Value
US English	01
British English	44
German	49
Italian	39
French	33
Spanish	34
Polish	48
Portuguese	55
Japanese	81
Russian	07
Chinese	86
Turkish	90

Reference: Designer-31436

Database Editor

- **Create Connectors without Pins**

Create Connectors without Pins

Connectors can also be created in the Component Wizard without pins. Such components can be used, for example, as cover plates or cavity parts without connection possibilities in connectors with inserts.

This allows connectors to be defined as cavity parts for connectors with inserts with as little effort as possible.

To define connector components without pins, select **Tools** → **Start Database Editor** in the main menu.

The database editor opens.

Right-click in the component tree of the database editor and select the context menu command **New Component**.

The component wizard opens.

Select the component type **Connector** in the **Component Wizard - Identification** dialog and then define the connector properties on the following page. Confirm both dialogs with **Next**.

Finally, set the number of pins to 0 in the **Component Wizard - Connector Structure** dialog:

Component Wizard - Connector Structure

Please decide whether to use the existing connector structure or create a new structure for the new connector.

Component name: NewComponent1
Version: 1

Use current structure.
The structure cannot be modified

Make new copy.
The existing structure can be modified

Completely new structure.
An empty structure is used that must then be completed

Connector symbol
<no entry> ...
No Preview Available

Block symbol
<no entry> ...
No Preview Available

Number of pins:
0

< Back Next > Finish Cancel

Finish defining the connector by pressing **Finish** or select **Next** to edit the connector graphically.

For connectors that do not have pins, the following applies:

- Connectors without pins automatically receive the sub-type **Cavity Plug**. Such components can be used, for example, as cover plates or cavity parts for connectors with inserts.
- Connectors without pins can only be assigned to models without pins and symbols of type **Attribute text template**.
- No pins can be defined subsequently for connectors without pins.
- To use connectors without pins in the project, drag them into the device tree.
- Connectors without pins cannot be placed on sheets. However, models and attribute text templates assigned to them can be placed.
- It is possible to exchange components of connectors without pins.

Importing and Exporting Data

Importing Data

- **Display Configuration Files when Importing and Exporting DNG, DWG and DXF Files**
- **Place Images on Sheets via Copy & Paste and Drag & Drop**

Exporting Data

- **Display Configuration Files when Importing and Exporting DNG, DWG and DXF Files**

Display Configuration Files when Importing and Exporting DNG, DWG and DXF Files

When importing and exporting DNG, DWG and DXF files, the output window of **E³.series** shows which configuration file is used.

This makes it clear which configuration file was used and in which directory it is located every time the files are imported and exported. This enables you to determine the path to the files even faster and make any necessary adjustments more user-friendly.

Place Images on Sheets via Copy & Paste and Drag & Drop

Images can now also be placed with insert operations.

The enhancement enables images to be inserted more quickly and independently from import dialog.

To place images with the operating system's insert operations, the following options are available:

- Copy an image in a folder or file with embedded images and insert it on a sheet in **E³.series**.
To do this, select the image, copy it using the keyboard shortcut Ctrl + C via the context menu, paste it using the keyboard shortcut Ctrl + V or via the context menu and place it by clicking on the position on the sheet where the image is to be placed.
Note: If images are copied from other programs, only one image can be copied at a time. If multiple images are copied, only the first image whose file type is supported is inserted.
Additionally selected text is not inserted.
- Select an image file in a folder and drag & drop the file onto a sheet in **E³.series**.

Reference: Designer-04891

Multuser

- **Expert Mode for Installing Multuser Servers**

Expert Mode for Installing Multuser Servers

When installing multuser servers with the `MuSetup.exe` installation file, there is an expert mode that allows configuring the multuser server directly in the operating system's **Data Link Properties** dialog.

All database drivers can be selected, which are installed.

To use expert mode for installing multuser servers, double-click on the file `MuSetup.exe`. The installation wizard opens.

First select the language to guide you through the installation process.

Then, under **Installation Path**, specify the directory in which the data for accessing the multuser server will be installed.

Finally, select the setup method **Expert mode** in the **Database** window. The **Data Link Properties** dialog opens:

E³.enterprise Setup [Close]

Database
Please choose the database for E³.enterprise.

Database

Oracle SQL Server **Expert mode**

Data source: [Dropdown]

Database Administrator

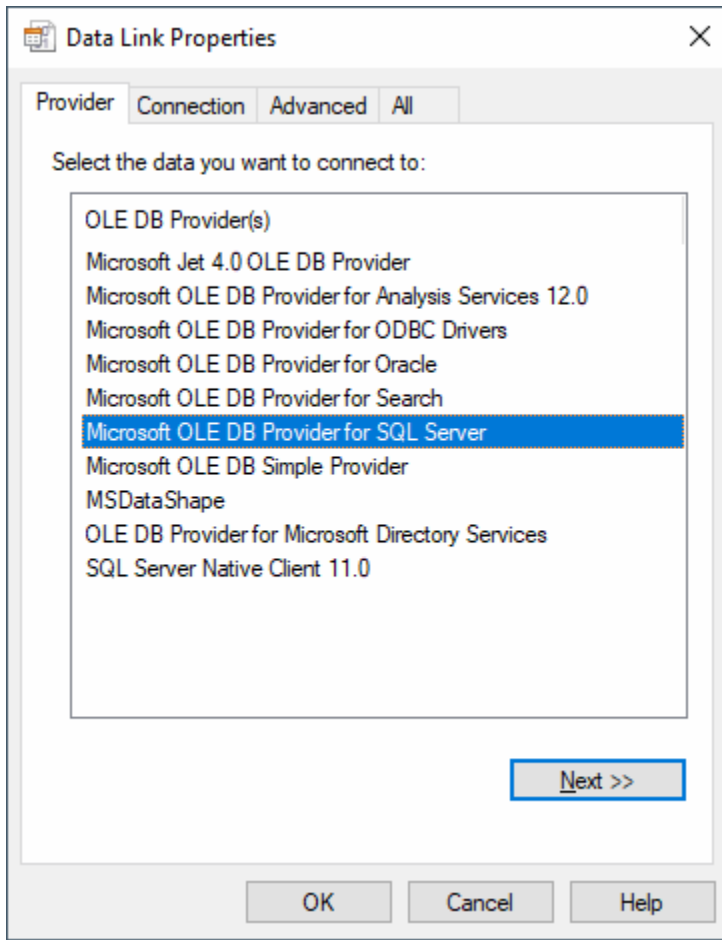
[Blurred]

Windows account: [system] [...]

Database account: []

Password: []

< Back **Next >** Cancel Help



Specify the following information in the dialog:

- In the **Provider** tab:
 - the Database Provider
 - In the **Connection** tab:
 - Server
 - Database on the Server
 - Credentials of the Database Administrator
 - Note:** The user that is specified must have rights as a database administrator. The user is entered under **Database Administrator**.
- One of the following log-in methods can be selected for the database administrator:
- o The Windows user connecting to the server.
 - This log-in procedure is implemented in the system dialog with the setting **Use Windows NT Integrated security**.
 - o A defined, specific user.
 - This log-in procedure is implemented in the system dialog with the setting **Use a specific user name and password**; the desired user account (**User name:**) and the appropriate password (**Password:**).
 - Note:** If a user is specified for the log-in procedure, the option **Allow saving password** must be enabled.

The information for the connection is entered into the registry in the form of a so-called *Connection String*.

Project Handling

- **Automatically Create Sheet Names with Multiple-Digit Sheet Numbers**
- **Update Device Attributes without Changing Other Component Properties**
- **Retain Manually Changed Device Attributes when Updating or Exchanging Components**
- **New Text Type for Cable Ducts**
- **Search by Component Name**
- **Text Token for Description of Wire Colors**

Automatically Create Sheet Names with Multiple-Digit Sheet Numbers

Sheet names of automatically generated sheets can be generated with multi-digit sheet numbers.

This allows one to define the format of the names of sheets that are to be automatically generated.

Note: With the change the setting **Create unique sheet names** was moved from the **Placement** → **Import** menu to the **General** menu.

To specify the name format of sheets to be automatically generated, select **Tools** → **Settings...** in the main menu.
The **Electric Settings** dialog opens.

Note: The setting can be made via the **Electric Settings** and **Fluid Settings** dialogs.

Then specify how many characters automatically generated sheet names have under **General** → **Number characters incl. leading zero**.

The setting is project-specific:

Electric Settings

Categories:

- General
- Connection
- Placement
- Graphic
- Dimensions
- Panel
- Variants/Options
- MIL-Standard
- Electrical Checks
- Auto Routing

General

Working Grid
Grid size: 4 mm
Snap size: 1.5 mm

Alternative Grid
Grid size: 0 mm

Measurement Units
 Millimeters
 Inches

Save
 Automatically generate backup file after 60 minutes
 Write messages and results to file

Template
Initial settings can be read from a file.
File name: C:\Users\Public\Documents\Zuken\E3.series_2023\PROJECT_IEC.E3T

Layout
Layout measure:

Compatibility Mode
 CR-5000

Align Distances
Horizontal: 0 mm
Vertical: 0 mm

Snap Size
Snap size: 4 mm

Sheets
Number characters incl. leading 0: 3
 Create unique sheet names

OK Apply Cancel Help

Sheets generated automatically receive a name with the specified number of characters.

Note: If a new sheet is generated with **Copy** and **Paste**, the rule for the number of characters is only used if the **Create unique sheet names** setting is also active.

The setting **Create unique sheet names** has been moved from the **Placement** → **Import** menu to the **General** menu as part of the enhancement.

Reference: Designer-35816

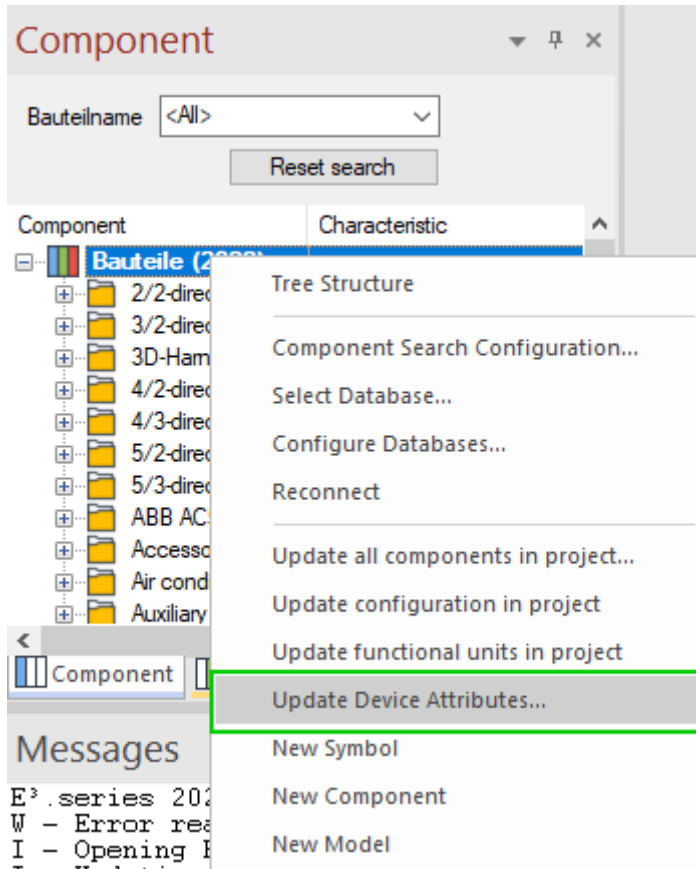
Update Device Attributes without Changing Other Component Properties

Selected device attributes can be updated, which reference component attributes.

This enables the device attributes to be updated without updating and thus changing other properties of the component, for example, the symbol assignment.

There are two ways to update device attributes:

1. To update all device attributes, right-click on a folder in the component view of the database tree or on the uppermost element and select **Update Device Attributes...**
The **Update Device Attributes** dialog opens:



Update Device Attributes

Overwrite	Name	Assign	Location	Compon	Attribut	Copy	
<input type="checkbox"/>	<All>	=A1	+BAZ1	0432132	<All>	<All>	
<input checked="" type="checkbox"/>	-S3	=A1	+BAZ1	0432132	Hyperlink	1	data sheet ref="Kugelhahn.pdf"
<input checked="" type="checkbox"/>	-S3	=A1	+BAZ1	0432132	Data she	1	Kugelhahn.pdf
<input checked="" type="checkbox"/>	-S3	=A1	+BAZ1	0432132	Technical	1	ball valve
<input checked="" type="checkbox"/>	-S3	=A1	+BAZ1	0432132	Weight	1	0,515
<input checked="" type="checkbox"/>	-S4	=A1	+BAZ1	0432132	Hyperlink	1	data sheet ref="Kugelhahn.pdf"
<input checked="" type="checkbox"/>	-S4	=A1	+BAZ1	0432132	Data she	1	Kugelhahn.pdf
<input checked="" type="checkbox"/>	-S4	=A1	+BAZ1	0432132	Technical	1	ball valve
<input checked="" type="checkbox"/>	-S4	=A1	+BAZ1	0432132	Weight	1	0,515
<input checked="" type="checkbox"/>	-S5	=A1	+BAZ1	0432132	Hyperlink	1	data sheet ref="Kugelhahn.pdf"
<input checked="" type="checkbox"/>	-S5	=A1	+BAZ1	0432132	Data she	1	Kugelhahn.pdf
<input checked="" type="checkbox"/>	-S5	=A1	+BAZ1	0432132	Technical	1	ball valve
<input checked="" type="checkbox"/>	-S5	=A1	+BAZ1	0432132	Weight	1	0,515
<input checked="" type="checkbox"/>	-S6	=A1	+BAZ1	0432132	Hyperlink	1	data sheet ref="Kugelhahn.pdf"
<input checked="" type="checkbox"/>	-S6	=A1	+BAZ1	0432132	Data she	1	Kugelhahn.pdf
<input checked="" type="checkbox"/>	-S6	=A1	+BAZ1	0432132	Technical	1	ball valve
<input checked="" type="checkbox"/>	-S6	=A1	+BAZ1	0432132	Weight	1	0,515
<input checked="" type="checkbox"/>	-S7	=A1	+BAZ1	0432132	Hyperlink	1	data sheet ref="Kugelhahn.pdf"
<input checked="" type="checkbox"/>	-S7	=A1	+BAZ1	0432132	Data she	1	Kugelhahn.pdf
<input checked="" type="checkbox"/>	-S7	=A1	+BAZ1	0432132	Technical	1	ball valve
<input checked="" type="checkbox"/>	-S7	=A1	+BAZ1	0432132	Weight	1	0,515

OK Cancel Apply Help

All device attributes used in the project are displayed line by line in the table. The columns can be filtered and sorted.

Select all attributes to be overwritten in the column **Overwrite** and confirm the selection by pressing **OK** or **Apply**

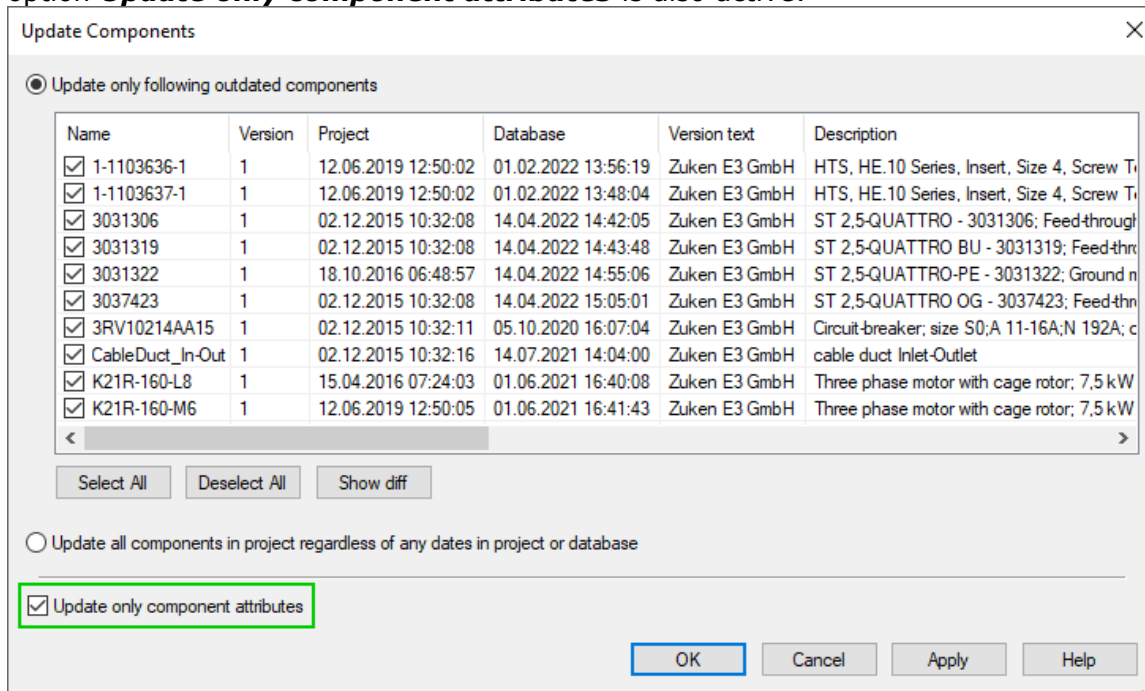
To select all attributes, which are displayed, select the context menu command **Select all (visible)** in the column **Overwrite**.

With **Deselect all (visible)** the selections of all displayed attributes are removed.

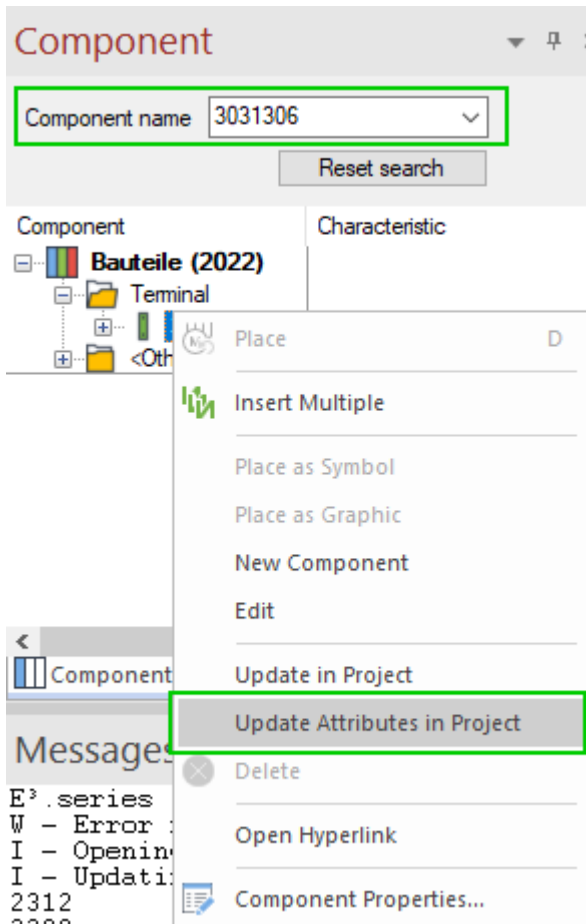
Then, in the **Overwrite** column, select all referenced components whose attributes are to be updated and confirm the selection with **OK**.

The attributes of the components used in the project are updated.

Note: Component attributes can also be updated in the **Update Components** dialog if the option **Update only component attributes** is also active:



2. To update the attributes of an individual component, locate the desired component in the database, right-click on it in the database tree and select **Update Attributes in Project**:



All attributes of the component are updated.

Retain Manually Changed Device Attributes when Updating or Exchanging Components

When updating or exchanging the component, manually changed device attributes can be retained.

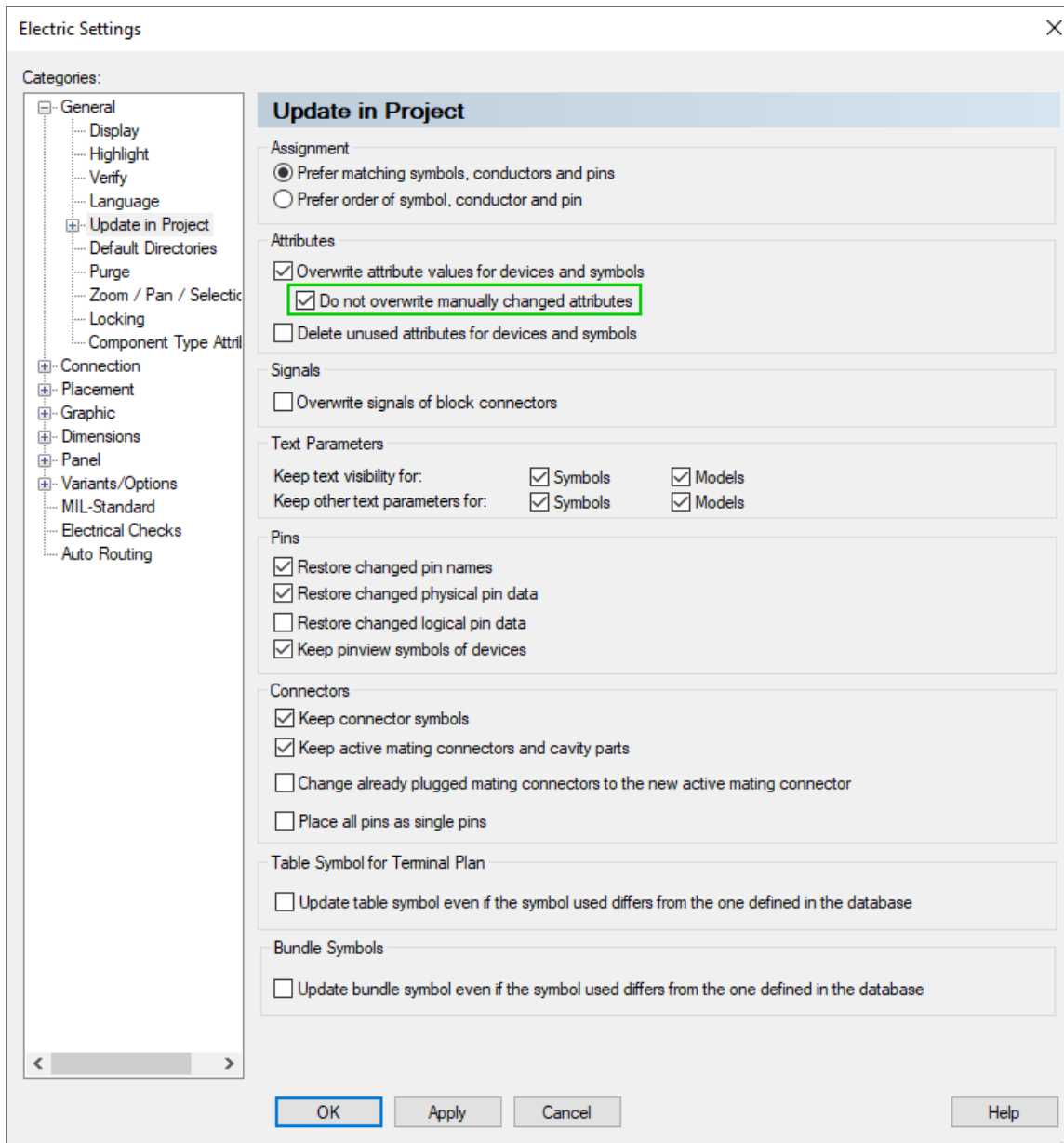
This allows you to adjust the attributes of devices so that they differ from the used component and you do not have to confirm the changes again and again when updating or exchanging components..

To retain manually changed device attributes when updating components, select **Tools** → **Settings...** in the main menu.
The **Electric Settings** dialog opens.

Note: The setting can be made via the **Electric Settings** and **Fluid Settings** dialogs.

Then go to the menu **General** → **Update in Project** and activate the setting **Do not overwrite manually changed attributes**.

The setting is project-specific:



If attributes or values, which were changed manually, receive different attributes or values via the **Update in Project** function, the attributes and values of the device keep the same status they had before updating the component.

Attributes, which have been deleted from the device but are still used by the device, are not assigned to the device again when updating the component.

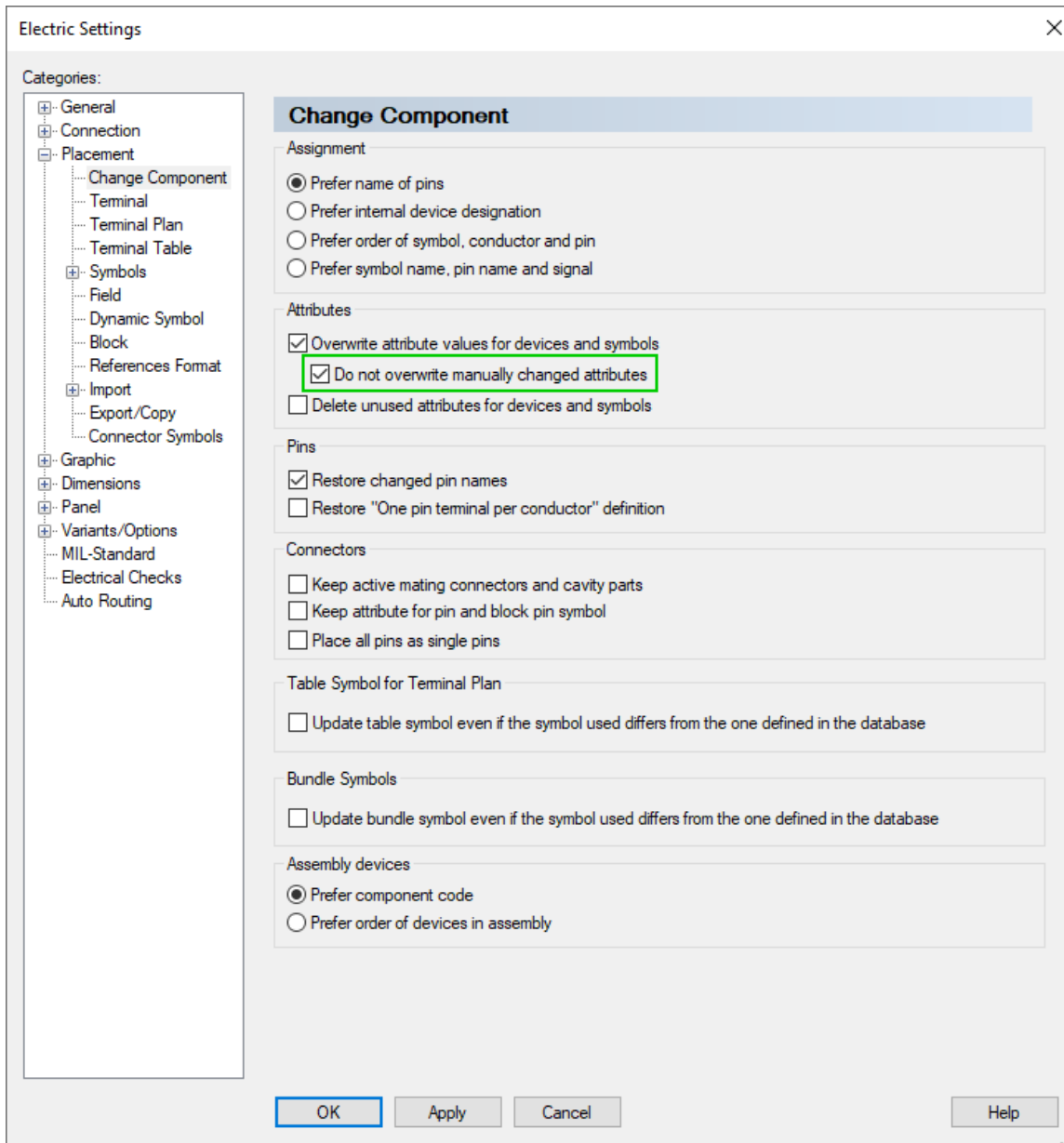
To retain manually changed device attributes when updating components, select **Tools** → **Settings...** in the main menu.

The **Electric Settings** dialog opens.

Note: The setting can be made via the **Electric Settings** and **Fluid Settings** dialogs.

Then go to the menu **Placement** → **Change Component** and activate the setting **Do not overwrite manually changed attributes**.

The setting is project-specific:



If attributes or values that were changed manually receive different attributes or values due to exchanging a component, the attributes and values of the retain the status they had before the part was updated.

Attributes that have been deleted from the device but are still referenced by the component in use are not assigned to the device again when exchanging the component.

New Text Type for Cable Ducts

E³.series provides a new text type to cross-reference cable duct inlets and outlets between projects and generated PDF and SVG files:

Text type	Text name	Max. length	Owner	Description	Line break permitted in text
-----------	-----------	-------------	-------	-------------	------------------------------

1186	Reference between cable duct inlet and outlet	255	Model	Is filled in automatically when placing and serves as a cross-reference between cable duct inlets and outlets	Yes
------	-----------------------------------------------	-----	-------	---------------------------------------------------------------------------------------------------------------	-----

References: Designer-25783 and Designer-28408

Search by Component Name

The search function can also find components that are used for devices. With the enhancement you can also search wires of wire groups.

This enables **E3.series** to be more efficient when searching for objects used in projects.

To search for components of devices, select **Edit** → **Search...** in the main menu. The **Search** dialog opens.

Then select the object type **Component Name** under **Type:** and specify all additional search criteria to be used for the search.

Search

Search Options

Object

Type: **Devices/Cables/Fields** Value:

Device de Value:

Higher lev Value:

Location: Value:

View num Value:

Attributes

Name: Value:

Variants

All Current Value:

Redlining Information

Search for all redlining information

Retain search options

Highlight Options

Color: Width: 2 mm

To search wires of wire groups, enter a part of the name or the full name of a wire group and an asterisk(*).

Example:

- The search term **HO*** finds all wires of wire groups whose names start with **HO** in the project, so for example the wire groups **H05V-K** and **V07V-K**.
- The search term **H05V-K*** finds all wires of the wire group **H05V-K** in the project.

Then start the search with **OK**.

All objects found are displayed in the **Results** tab of the Results window, which is activated automatically.

Double-click on a line to jump to the desired object in the drawing.

If the result is a wire group with placed symbols, it jumps to the first symbol.
For wire groups without placed symbols, it jumps to the device tree.

Reference: Designer-06101

Text Token for Description of Wire Colors

Text tokens also be used to describe wire colors.

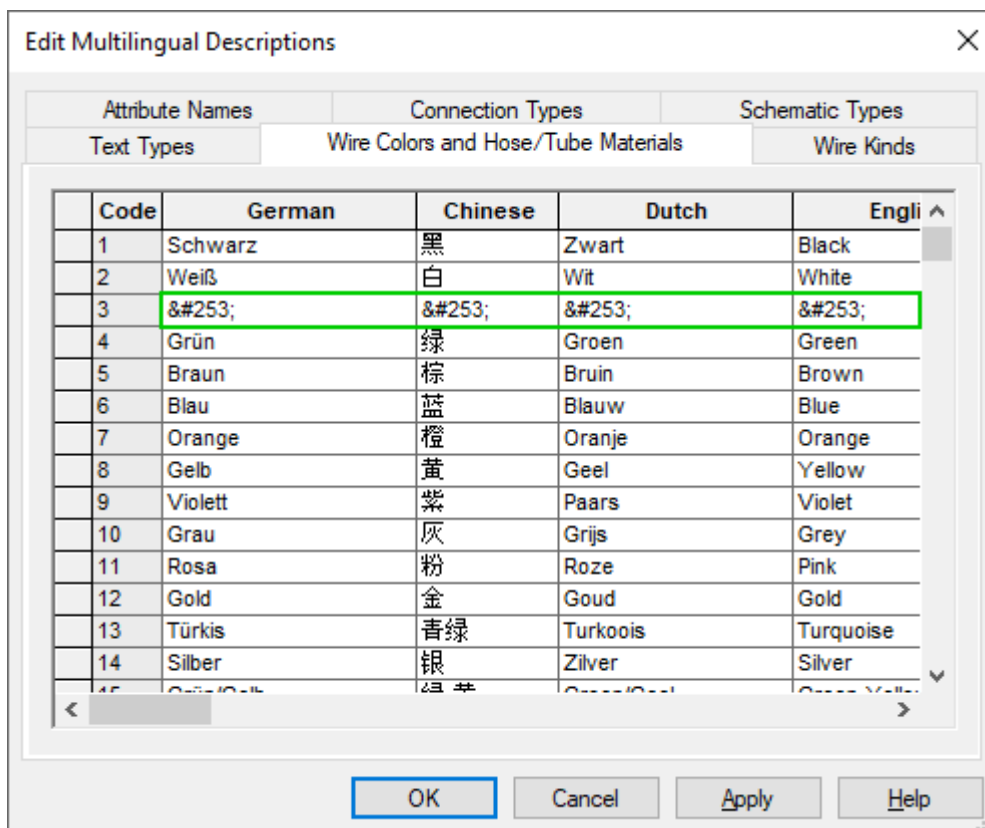
This enables, for example, text types used to specify the wire color to display the respective translated wire color name for all project languages.

To use text tokens for wire colors and translate the colors to all project languages, select **Tools** → **Start Database Editor** in the main menu.
The database editor opens.

Then under **Format** → **Multilingual Descriptions...** select the menu with multilingual descriptions used in **E3.series**.
The **Edit Multilingual Descriptions** dialog opens.

Then enter the text token for each color and language to be translated in the tab **Wire Colors and Hose/Tube Materials**.

To do so, enter the text token in the format **ý**; for all languages, for example **ý** for **red**:



Note: You can also enter the text token using the translation table. To do so, click on the icon :

Edit Multilingual Descriptions

Attribute Names		Connection Types		Schematic Types	
Text Types		Wire Colors and Hose/Tube Materials		Wire Kinds	
Code	German	Chinese	Dutch	Engli	
1	Schwarz	黑	Zwart	Black	
2	Weiß	白	Wit	White	
3	Rot	紅	Rood	Red	

Then find the translation text to be used for the color in the **Translation Table** and double-click on it:

Translation Table

Outdated	Description	<mainClass>	<subClass>	American E	British	Castellano	Chinese	Chinese(CH)	Deutsch	Francais	Italiano	Japan
<input checked="" type="checkbox"/>	<All>	<All>	<All>	red	<All>	<All>	<All>	<All>	<All>	<All>	<All>	<All>
<input type="checkbox"/>		Colours	single	red	red	rojo	紅	紅	rot	rouge	rosso	赤

OK Cancel Help

References: Designer-17310 and Designer-37373

Panel (also: 3D)

- Define Opposing Slots and Penetration Depth in 2D Views
- Adjust Size of STEP Models when Importing
- Rotate and Tilt Models in Any Dimension

Define Opposing Slots and Penetration Depth in 2D Views

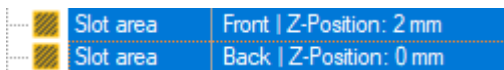
When generating 2D views of slots, it is possible to specify from which placement side the 2D view is generated and the distance of the observation perspective to the view. Furthermore, it is now also possible to generate multiple 2D views from the same side.

This allows the external and internal sides of models with slots to be displayed. In addition, the distance of the observation perspective from the 2D view allows different depths of penetration into the model to be displayed.

1. Generate 2D Views of Selected Placement Sides

To display the placement side of the model in a 2D view, select slots of two opposite sides in the device tree.

So either slots from **Front** and **Back**, slots from **Left** and **Right** or slots from **Top** and **Bottom**:



Right-click on the selection, open the sub-menu **Create 2D View** in the context menu and then select the side from which the view should be generated.

The displayed placement side results from the positions of the selected slots and the direction of the generated view. The following table summarizes the relationship between the positions of slots, the directions of the views and the resulting placement side:

Placement Side	Position of Selected Slots	Direction of View to be Generated
Inside left	Left and Right	Right
Outside left		Left
Inside right		Left
Outside right		Right
Inside front	Front and Back	Back
Outside front		Front
Inside back		Front
Outside back		Back
Inside top	Above and Below	Bottom
Outside top		Top
Inside bottom		Top
Outside bottom		Bottom

2. Consider the Penetration Depth of Slots in 2D Views

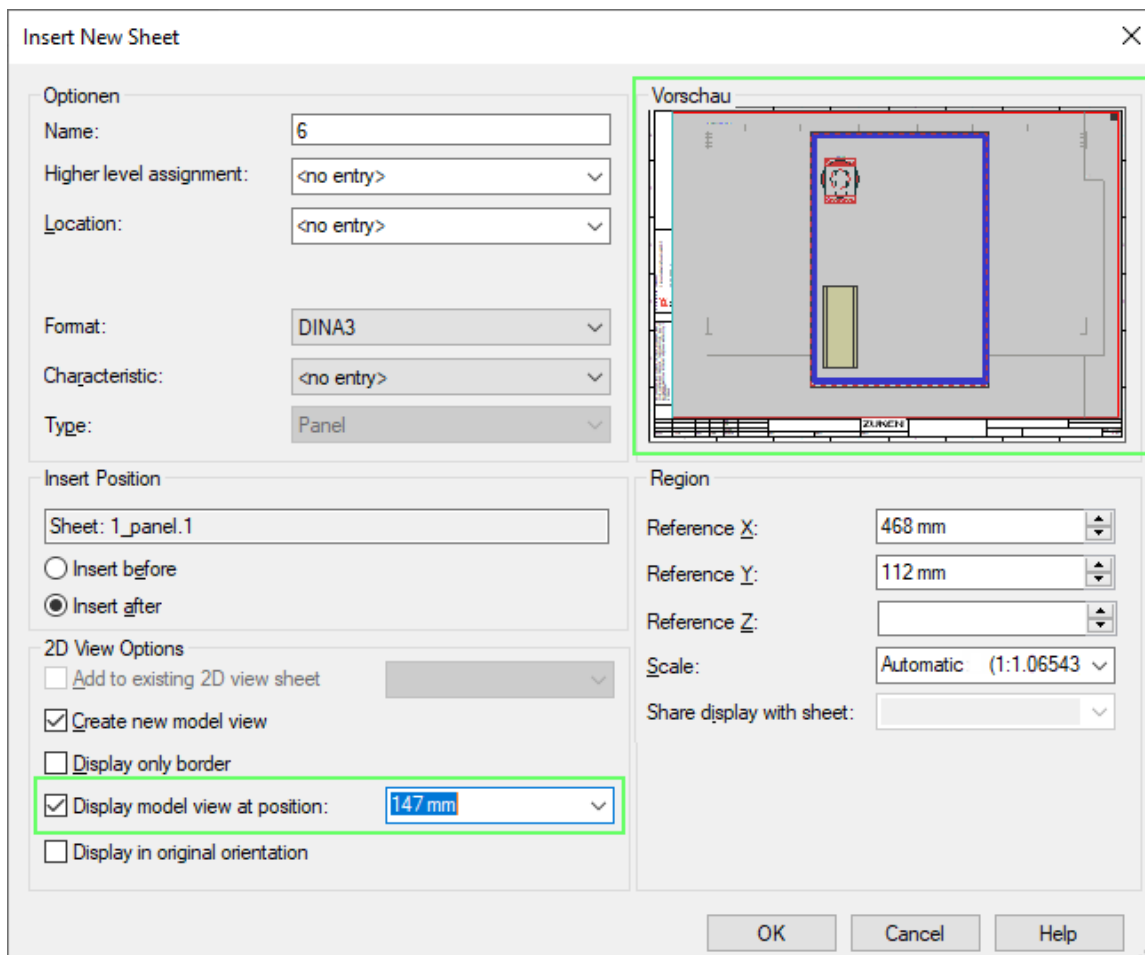
To take into account the penetration depth of slots in 2D views, right-click on a model in the device tree or on a sheet.

Select the context menu command **Create 2D View**. If several views can be generated for the model, select the respective direction in the sub-menu.

The **Insert New Sheet** dialog opens.

Define the properties of the 2D view sheet to be created, activate the option **Display model view at position** and select the penetration depth in the model.

The preview shows which slots are visible with the set penetration depth:



Note: The option is only available if the option **Display only border** is not selected and the model has slots with the direction for which the 2D view is generated.

Confirm the settings by pressing **OK**.

The 2D view of the model is generated with the defined penetration depth.

The option and penetration depth can also be changed in the sheet properties.

Reference: Designer-14139

Adjust Size of STEP Models when Importing

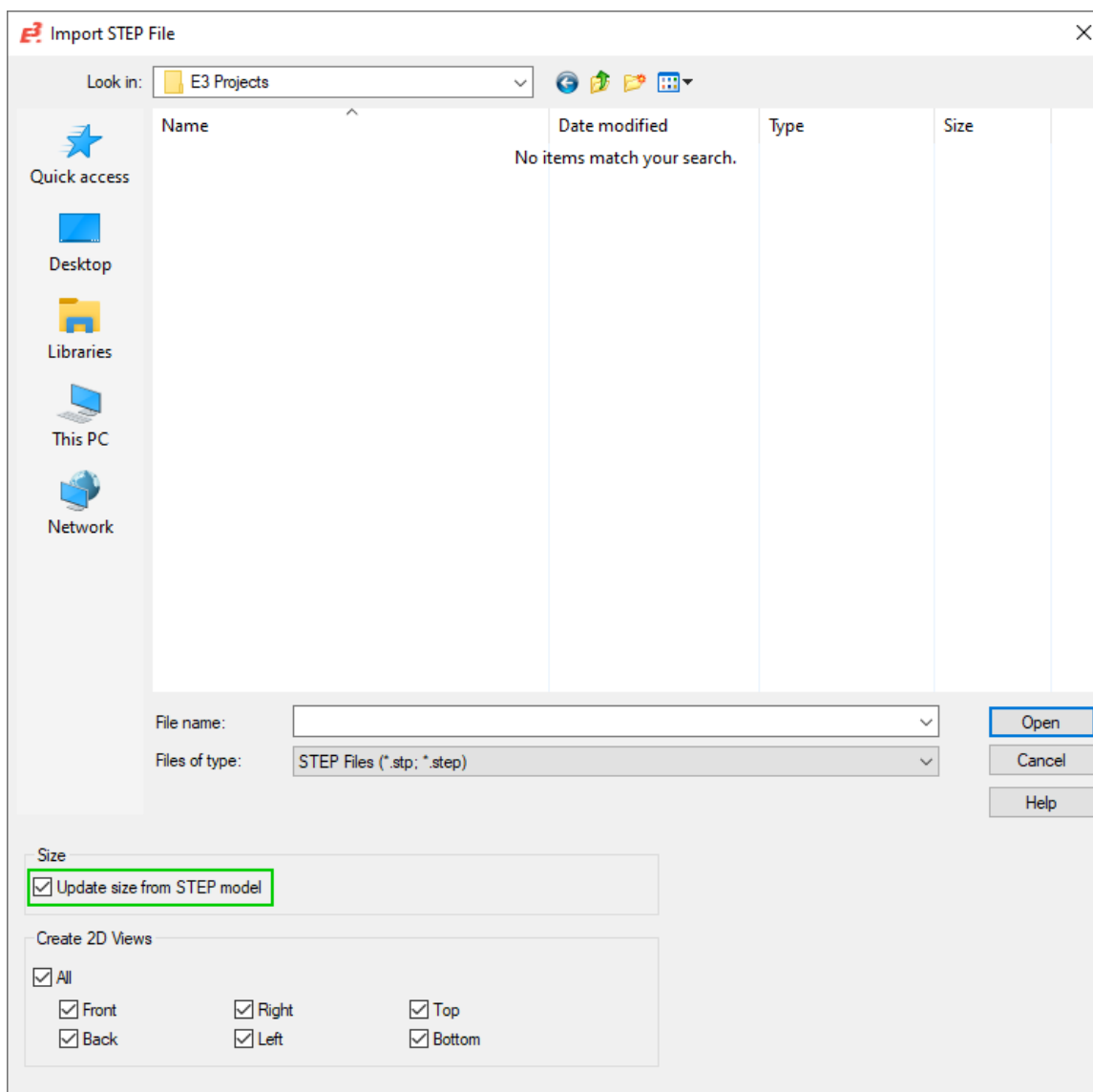
The size of STEP models can now be adjusted when importing.

Thus, the model properties in **E3.series** and the dimensions of the imported STEP model fit to each other.

To assign a STEP model, whose size can be adjusted when importing, to models in **E3.series**, select **Tools** → **Start Database Editor** in the main menu.
The database editor opens.

To create a new model or edit an existing model, select **Insert** → **STEP Model...** in the main menu.
The **Import STEP File** dialog opens.

Activate the option **Update size from STEP model** in the dialog, select the 2D views of the STEP model that will be created and confirm the selection by pressing **OK**:



The size of the STEP model is adjusted during import so that it fits the size of the model properties.

References: Designer-26441 and Designer-29158

Rotate and Tilt Models in Any Dimension

The information about the rotation of models is displayed in the device properties and the tooltip.

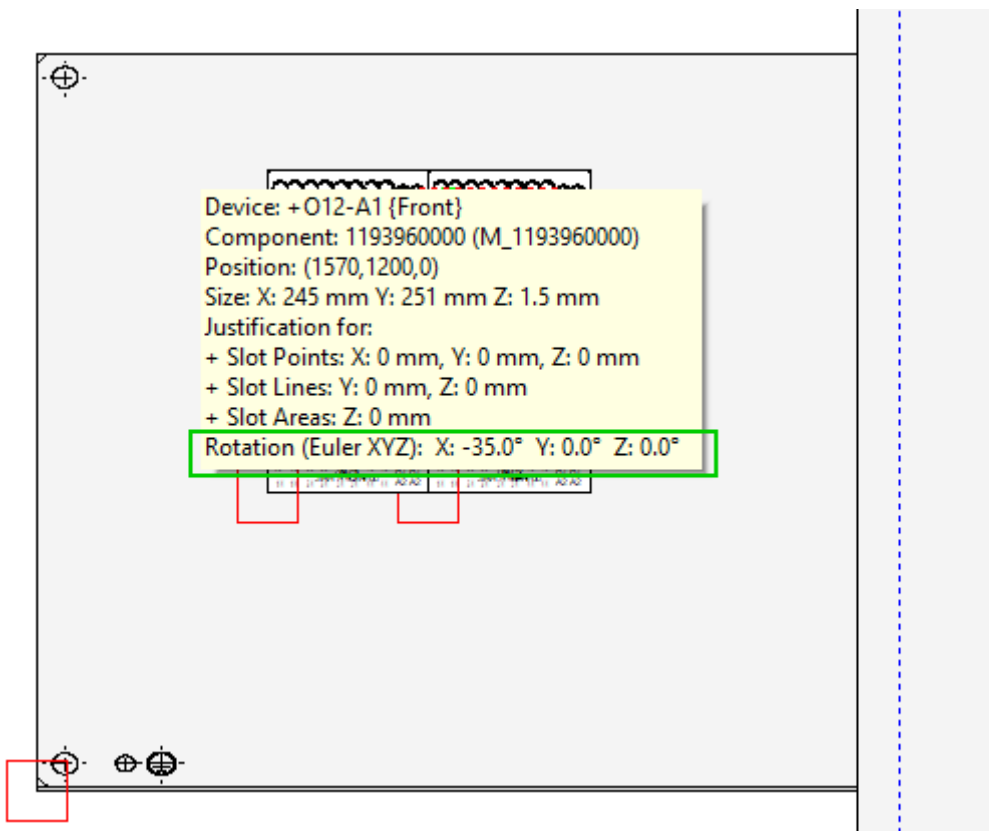
This makes the display of the individual rotation values even more user-friendly.

The rotation values can be displayed in two different ways:

1. Open a sheet, on which a rotated model is placed.

Move the mouse over the model and then leave it motionless for a short time.
The tooltip opens.

The rotation values around the respective axes are displayed with **Rotation (Euler XYZ)**:



Note: The tooltip is only displayed if the model has at least one rotation value not equal to 0°.

2. Right-click on a model in the device tree or on a sheet.
The context menu opens.

Select the context menu command **Device Properties...**
The **Device Properties** dialog opens.

Switch to the **Shape** tab.

The individual rotation values are displayed under **Rotation (Euler XYZ)**:

Device Properties
✕

Device Device II Component Model Shape Slots/Pins Assign Variants / Options Variant Overview

Mounting Description

Available descriptions:

▼
Add

List more descriptions Delete

Position

X:

Y:

Z:

Free placement:

Lock position:

Size

X:

Y:

Z:

Changeable:

Justification for:

Slot points

X:

Y:

Z:

Slot lines

Y:

Z:

Slot areas

Z:

Preview

Rotation (Euler XYZ)

X: 0.0° Y: 0.0° Z: 0.0°

OK
Cancel
Apply
Help

If several models are selected at the same time or the model is not rotated, no rotation values are displayed.

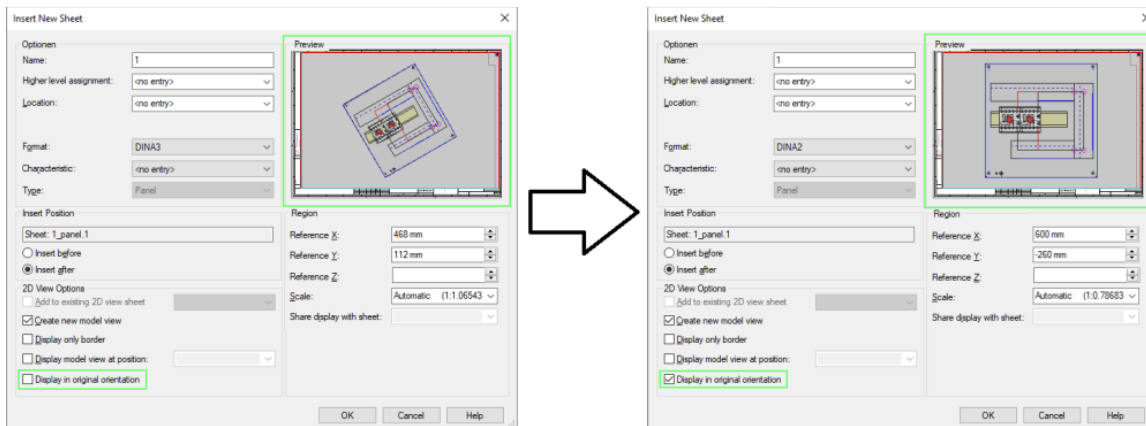
When 2D views of models are created, it is possible to set whether the rotation values of the model are used on the 2D view or whether the original orientation is used. This means that the 2D view does not reflect the rotated orientation of the model.

To create 2D views of models with their original orientation, right-click on a model in the device tree or on the sheet, select **Create 2D View**. If several views can be generated for the model, select the respective direction in the sub-menu.

The **Insert new sheet** dialog opens.

Specify the properties of the 2D view sheet to be created and activate the option **Display in original orientation**.

The preview displays the original orientation of the model:



Confirm the settings by pressing **OK**.

The 2D view of the model is created.

Reference: Designer-17956

Tables

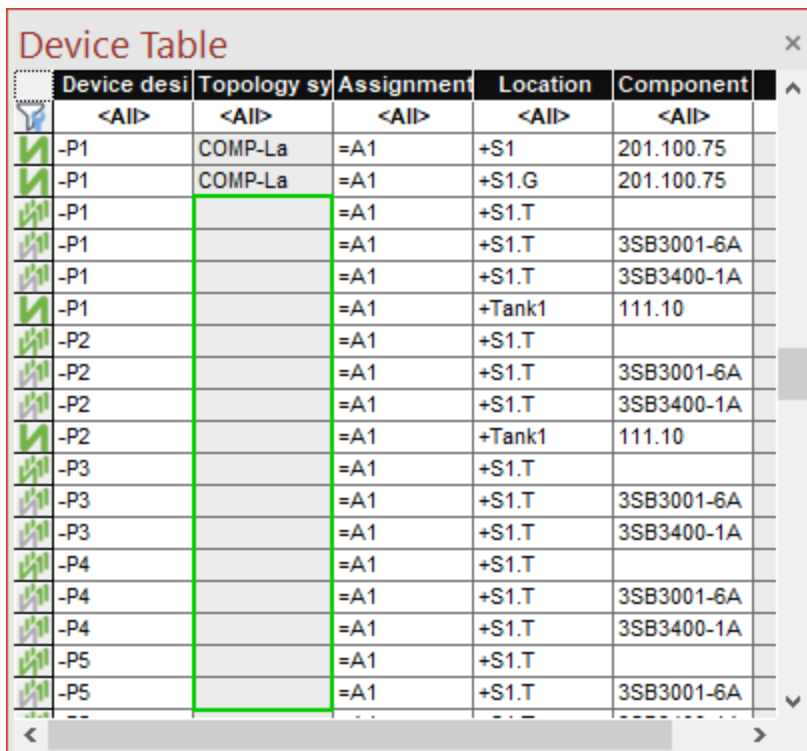
- Distinguish Objects with Unassigned Attributes
- Display Variants/Options in Device and Sheet Tables

Distinguish Objects with Unassigned Attributes

In all project tables, for example device and connection tables, a filter is available that can be used to highlight attributes that have not been assigned.

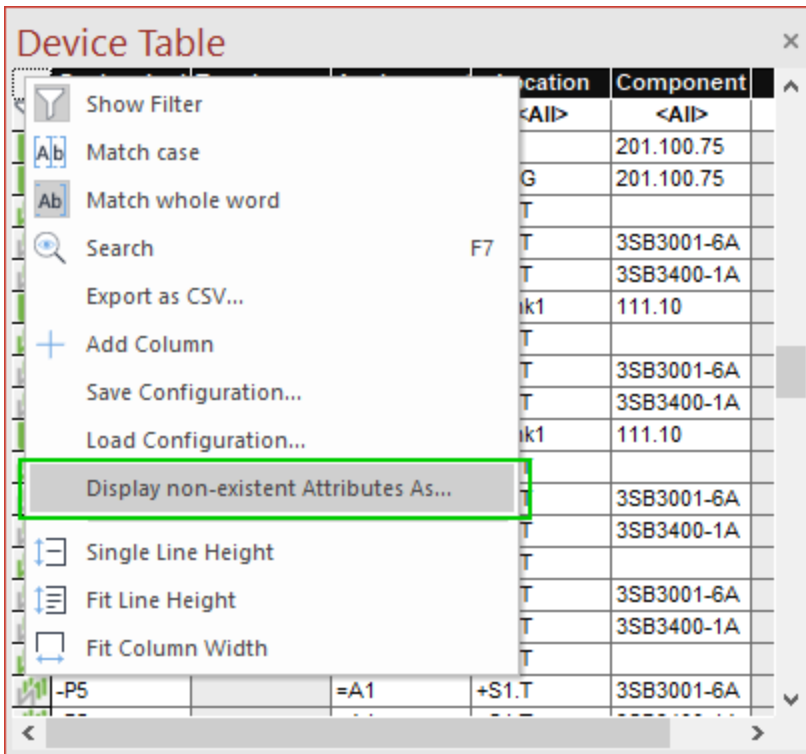
This makes it easier to see whether attributes have not been assigned or whether only no values have been entered for them.

The following example shows a device table in which it is not clear whether the attribute **Topology Symbol** has been assigned for the listed devices or whether no value has been entered:

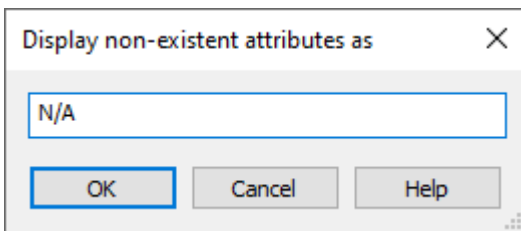


Device desi	Topology sy	Assignment	Location	Component
<All>	<All>	<All>	<All>	<All>
-P1	COMP-La	=A1	+S1	201.100.75
-P1	COMP-La	=A1	+S1.G	201.100.75
-P1		=A1	+S1.T	
-P1		=A1	+S1.T	3SB3001-6A
-P1		=A1	+S1.T	3SB3400-1A
-P1		=A1	+Tank1	111.10
-P2		=A1	+S1.T	
-P2		=A1	+S1.T	3SB3001-6A
-P2		=A1	+S1.T	3SB3400-1A
-P2		=A1	+Tank1	111.10
-P3		=A1	+S1.T	
-P3		=A1	+S1.T	3SB3001-6A
-P3		=A1	+S1.T	3SB3400-1A
-P4		=A1	+S1.T	
-P4		=A1	+S1.T	3SB3001-6A
-P4		=A1	+S1.T	3SB3400-1A
-P5		=A1	+S1.T	
-P5		=A1	+S1.T	3SB3001-6A

To display whether the attribute is available in tables, right-click in the upper left corner of the table and select the context menu command **Display non-existent Attributes as....**. The **Display non-existent Attributes as** dialog opens.



Enter a text that will be used for non-existent attributes and confirm the selection with **OK**. The text **N/A** is used in the example. The text is used in all tables in **E³.series**:



Device desi	Topology sy	Assignment	Location	Component
<All>	<All>	<All>	<All>	<All>
-P1	COMP-La	=A1	+S1	201.100.75
-P1	COMP-La	=A1	+S1.G	201.100.75
-P1		=A1	+S1.T	
-P1	N/A	=A1	+S1.T	3SB3001-6A
-P1	N/A	=A1	+S1.T	3SB3400-1A
-P1	N/A	=A1	+Tank1	111.10
-P2		=A1	+S1.T	
-P2	N/A	=A1	+S1.T	3SB3001-6A
-P2	N/A	=A1	+S1.T	3SB3400-1A
-P2	N/A	=A1	+Tank1	111.10
-P3		=A1	+S1.T	
-P3	N/A	=A1	+S1.T	3SB3001-6A
-P3	N/A	=A1	+S1.T	3SB3400-1A
-P4		=A1	+S1.T	
-P4	N/A	=A1	+S1.T	3SB3001-6A
-P4	N/A	=A1	+S1.T	3SB3400-1A
-P5		=A1	+S1.T	
-P5	N/A	=A1	+S1.T	3SB3001-6A

Note: The filter is user-specific and is used in all projects until it is changed or reset. The filter has no effect on the tables of other users of **E3.series**.

Reference: Designer-34922

Display Variants/Options in Device and Sheet Tables

Variants and options can be displayed in device and sheet tables.

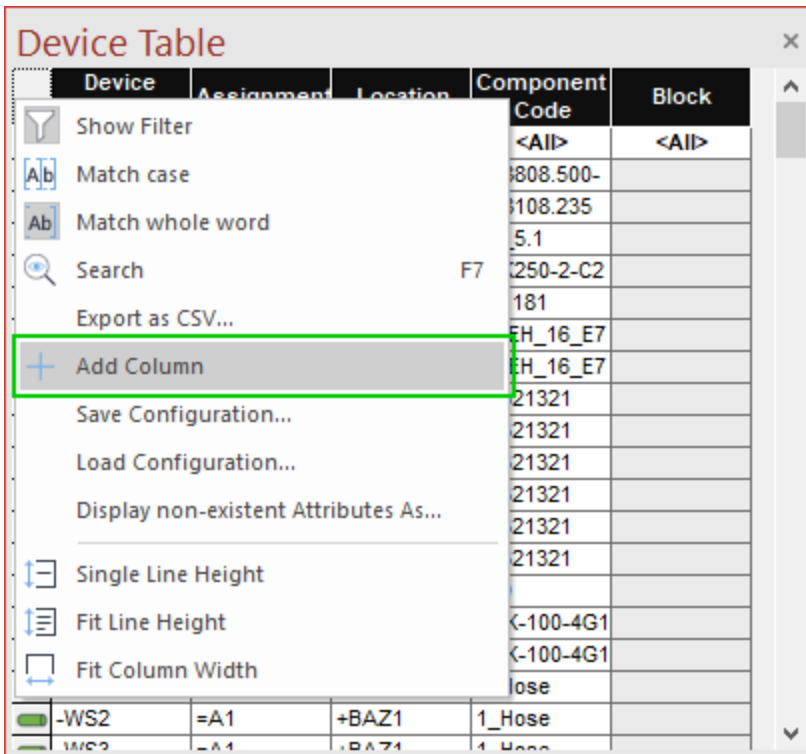
For device tables, the extended variant information can also be displayed.

This means that the search functions of device and sheet tables can also be used to search for variants, options and extended variant options.

To supplement the device or sheet table, select **View** → **Device Table** or **Sheet Table** in the main menu.

The desired table opens.

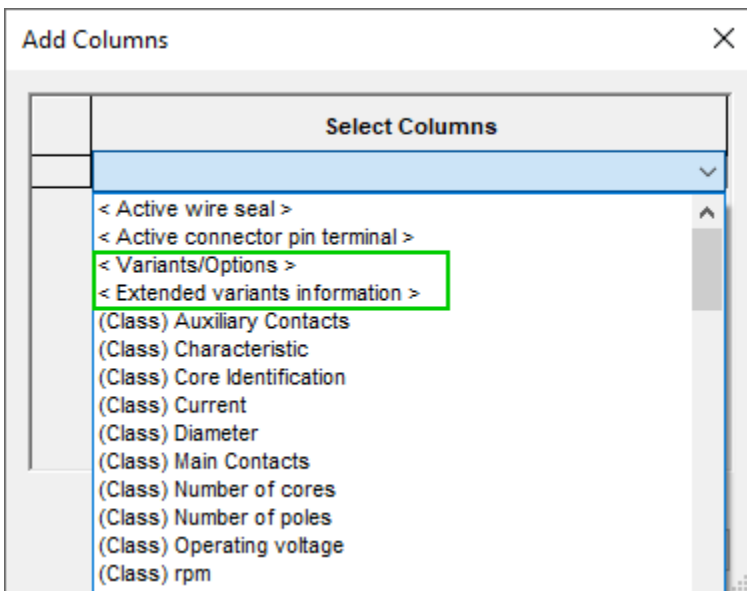
Right-click on a header and select the context menu command **Add column**:



The **Add column** dialog opens.

Then click in the empty rows and choose from the drop-down menu **Variants/Options** or **Extended variants information**.

Repeat the process for all additional columns to be added:



The desired columns are added to the table.

With these enhancements it is also possible to search and filter in the tables for variants/options and extended variants information.

References: Designer-11749 and Designer-20931



Americas

North America

Zuken USA Inc.
Westford, MA 01886, USA
Tel: +1 978 692 4900

Asia

Japan

Zuken (Worldwide Head Office)
Yokohama, Kanagawa 224-8585, Japan
Tel: +81 45 942 1511

China

Zuken (Shanghai) Technical Center Co.,
Ltd.
Room 301, No.555 Nanjing West Road,
Shanghai, 200041, People's Republic of
China
Tel: +86-21-3218-1784

Korea

Zuken Korea Inc.
Seoul 135-283, Korea
Tel: +82 2 5648031

Singapore

Zuken Singapore Pte Ltd.
#22-05 Gateway East, Singapore 189721
Tel: +65 6392 5855

Taiwan

Zuken Taiwan Inc.
Taipei 110, Taiwan
Tel: +886 2 7718 1116



Europe

Germany

Zuken GmbH (European HQ)
D-85399 Hallbergmoos, Germany
Tel: +49 89 7104059 00

Zuken E3 GmbH
D-89079 Ulm, Germany
Tel: +49 7305 9309 0

Zuken E3 GmbH
D-30659 Hannover, Germany
Tel: +49 511 8595 9489

Switzerland

Zuken E3 GmbH
CH-5504 Othmarsingen, Switzerland
Tel: +41 62 561 08 00

United Kingdom

Zuken UK Ltd.
Bristol, BS32 4RF, UK
Tel: +44 1454 207 801

France

Zuken S.A.
#91974 Les Ulis Cédex, France
Tel: +33 1 69 29 48 00

Italy

Zuken S.r.l.
20090 Milanofiori Assago, Milan, Italy
Tel: +39 02 575 921

Netherlands

Zuken GmbH
NL-6075 HA Herkenbosch, The Netherlands
Tel: +31 475 520 998