

CONVERTING FROM VER22 TO NEW FORMAT



This booklet describes how to convert from ver22 to the new format and what happens during the conversion.

If you don't want to convert old projects and components but only want to view old projects, the documents isn't relevant.

Developed by PCSHEMATIC A/S

Last edit April 2024



PREFACE

The booklet describes how to convert ver22 projects and their components, so that you can keep working on those projects in ver24.

Function and workflow are more or less as in ver23, but there are a few changes in the dialogs in ver24 opposed to those in ver23. The changes, however, mean that we recommend that you upgrade to ver24, as the changes have been made based on the challenges some have experienced in ver23.

Short resume

Start by saving your old settings in ver22 and read pages 7 and 8 in this document.

The recommended method to fetch components from the old database to the new one is to convert only the necessary components when they are needed!

That means, that you only convert the missing components from the individual projects. In this way, you only fetch the components that you need.

The import functions from ver24 are made to make it easy to import those components, including accessories and symbols.

If you want to upgrade ver22 projects, we recommend that you keep your old installation. We use it to get old settings and old symbols.

It is, of course, also possible to import larger parts of or the complete database, which takes longer time to retrieve, and means that you also fetch components that are no longer valid. See how to setup filters to import more components.

If you want to know more about the database and its datafields and setup, you can read the database manual, which is found on our website.

If you only want to view projects, use the Viewer, which can be downloaded for free on our website.

You don't need to convert the project when using the Viewer.



CONTENTS

PREFACE	3
CONTENTS	5
CONVERT VER22 COMPONENTS	7
First time	7
Every time	8
CONVERT A VER22 PROJECT	10
When you open the project.....	10
What happens in the project.....	11
Subdrawings and templates	12
Remap the components to the database.....	13
ComponentID.....	13
Componens with mixed usr/pcs-symbols.....	13
MORE ABOUT SETTING UP THE IMPORT WIZARD	14
Tab 0 – Connect to the old database	15
Use your old settings.....	15
If you do not have your old settings	15
Tab 1 – Select components from a list or a filter	16
Use filter to import a lot of components	18
Tab 2 – Ver22 fields that are imported	19
If you used our standard database in ver22	19
If you used your own article number in ver22	20
Tab 3 – Discontinued and own ver22 fields.....	21
If you used our default article number in ver22	21
If you used your own article number in ver22	22
Tab 4 – Mapping of fields from ver22 to ver24 database	23
If you used your own article number in ver22	24
Tab 5 – Symbols	25
Tab 6 – Import components.....	26
Merge ver22 components with Portal components	27
DATABASE SETTINGS IN VER24.....	28
Which database and other basic settings	28
Converted projects	29
MORE ABOUT THE DATABASE FOLDER.....	30
If you have a *sql-database	30
Files and subfolder	30
WHY CONVERT – WHAT DID WE DO?.....	31
The major changes from ver22 to ver24.....	31
New database structure	32
The important difference between article numbers in ver22 and ver24.....	32
Old settings.....	33
Import user settings	33
New symbols	34
If you have imported the full ver22 Alias list	35
We have created new PLC symbols and several functions in ver23.....	36
New plc symbols.....	36
You can read more	37
Your own menu table	38
MY NOTES	39



CONVERT VER22 COMPONENTS

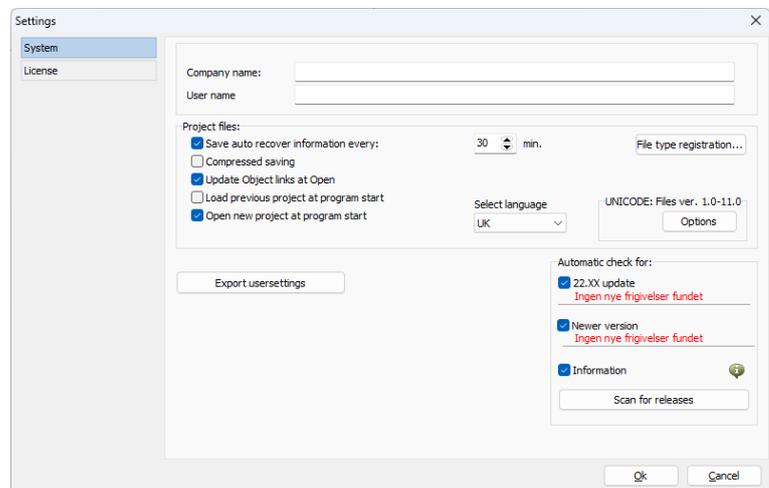
Below, you can find a quick overview of what to do, if you want to keep working with your old projects and old components in ver24. And here 'old' means, that you use the same components (and symbols) as previously.

First time

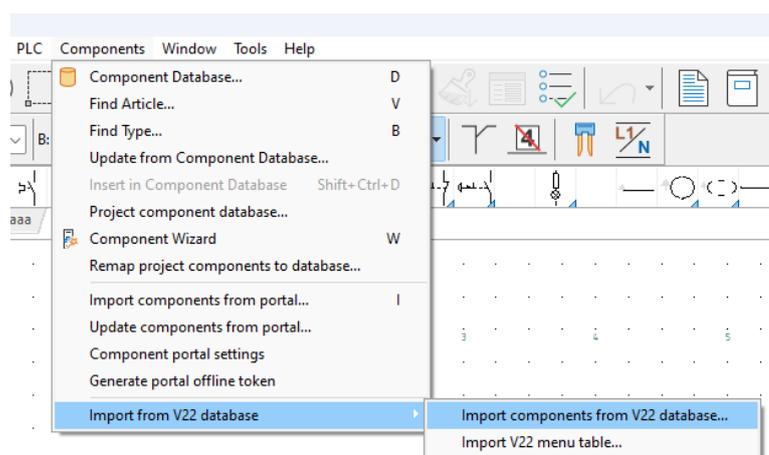
We recommend, that you install your ver24 on the same machine as your ver22, as your new installation can find old folders during the conversion.

1. Open your ver22 installation, which you should keep as long as you want to work with it and want to be able to fetch old components.

2. Export your old user settings in ver22. You find the function in Settings| System|License. The created file contains all settings that you can reuse in your new installation. In the new installation, you choose the settings you want to reuse. The old settings also helps you during the conversion of old projects and components.



3. Open your new version 24 installation.
4. Go to Components| Import from V22 database| Import components from V22 database ...



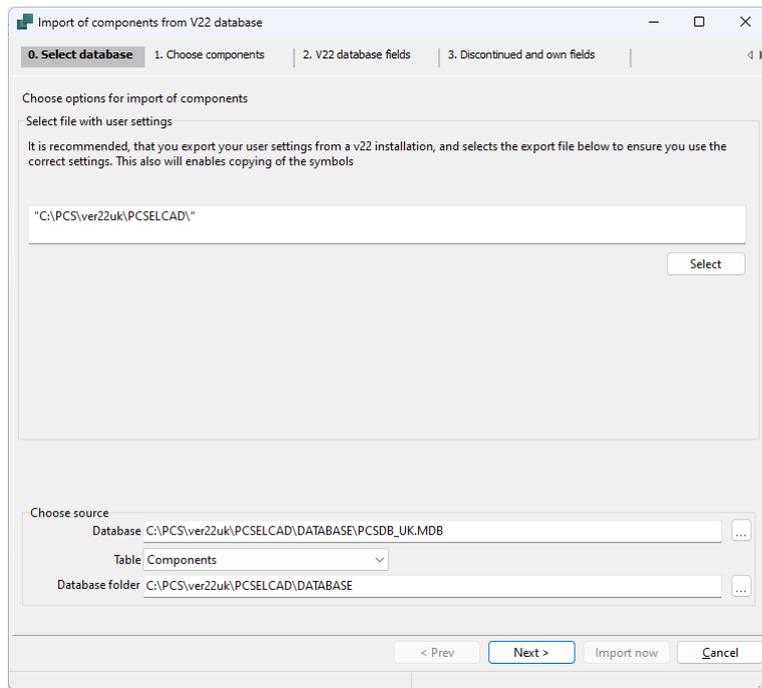
Select the file with your old user settings.

It knows your old installation (here ver22uk\PCSELCAD).

It also knows your previously connected database (bottom section) and your old symbol folders.

You can change to another database if you have used different databases for different projects.

Continue by pressing the Next button.



You can go through the various settings, but if you have a standard installation of ver22, you don't need to.

You can read more about the settings from page 14.

Every time

Open a ver22 project that contains components that you want to import. This is the easiest method; thus, this is the method explained first!

If the component used in this project are not in your database, you get the box seen below.

Buttons

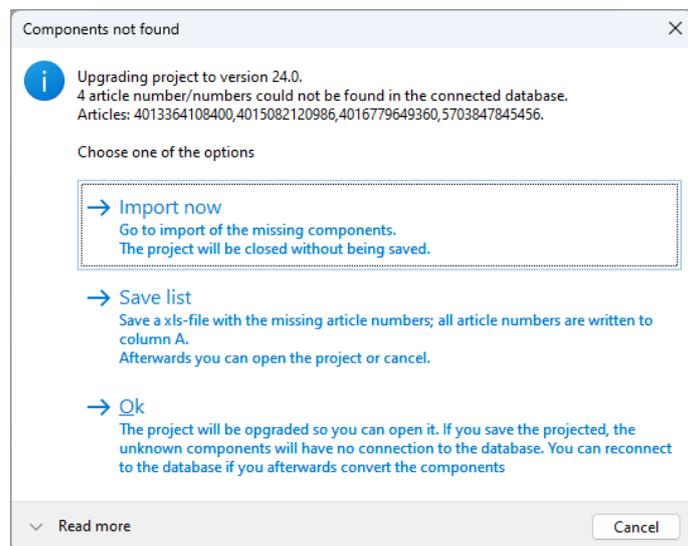
Import now: Go directly to import of the missing components.

Save list: Save a xls-list with the missing article numbers in column A. You import the list with the function Components | Import from v22 database.

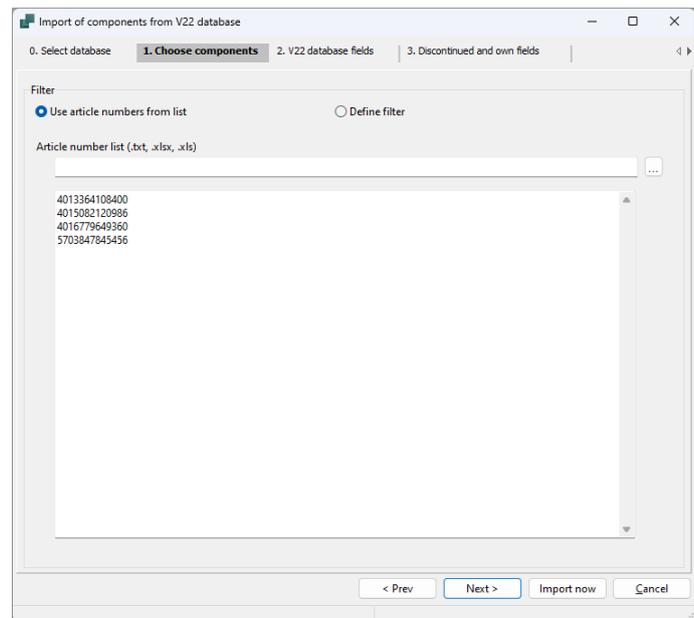
OK: Convert the file WITHOUT converting the components.

You can convert the missing components later.

When you press Import now, you go directly to the import function.



Here, you see the list of missing article numbers from the current project. Press the Import now button at the bottom and the listed components are imported into your database.



When you open an 'old' project, the program searches for its article numbers in the connected database's HistoricalArticleNumber. If the article numbers aren't found in this field, they are on the list of missing components.

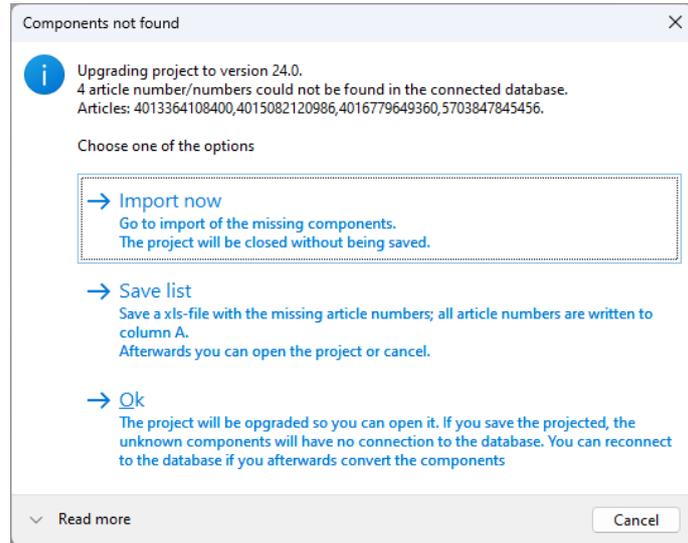
CONVERT A VER22 PROJECT

Here, you can read more about, what happens in the project.

When you open the project

When you open a project in ver24, that was made in ver22 and which contains components not available in the ver24 database, you get this window.

Do as described previously.



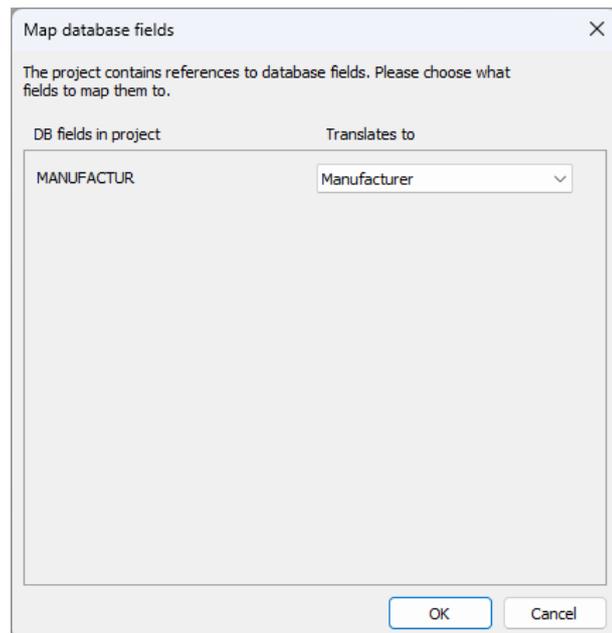
If all components are in the database already, the box will not appear.

If you don't convert immediately, you can open the file and have a look. The only thing is, that you can't look up other data on the components that are not in the database.

If you subsequently import the components, you can reconnect to the database – find the function in Components|Reconnect xxx

The next box appears, when the project contains lists or other objects in the file, which refers to the old database.

In the current project, the parts or component list uses the old database field for Manufacturer - MANUFACTUR. In the new database, the field is called Manufacturer.



What happens in the project

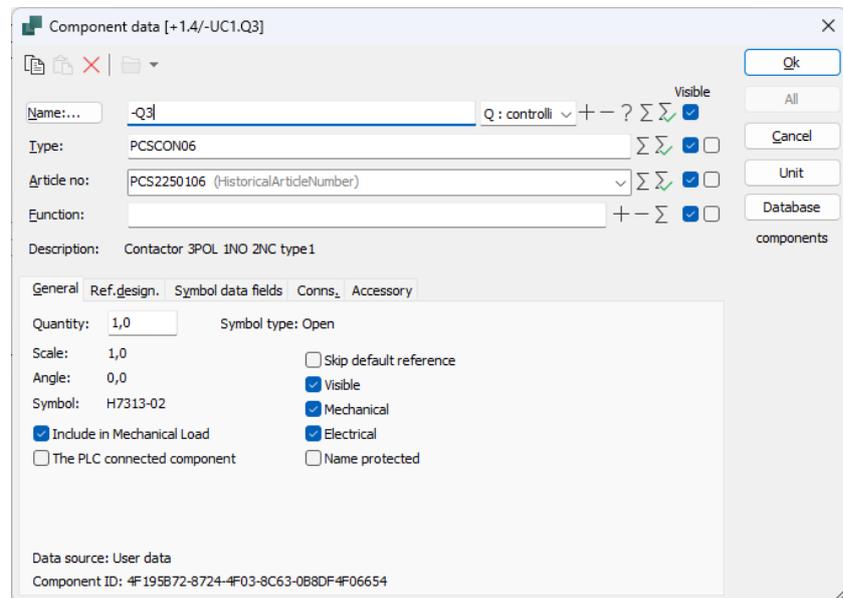
When we convert the projects, you - fortunately - cannot immediately see the difference.

This means that the project looks like itself, with the same names, article numbers, symbols etc. as before. And when you update the lists, you will get the same result as before.

Once the components and their symbols have been imported into your database, you can continue drawing with the same components with the same symbols as before. However, you can also download new components from our Component Portal and insert them into the project. Or create your own components and insert them into the project, i.e. you can do everything you were able to do with the project before.

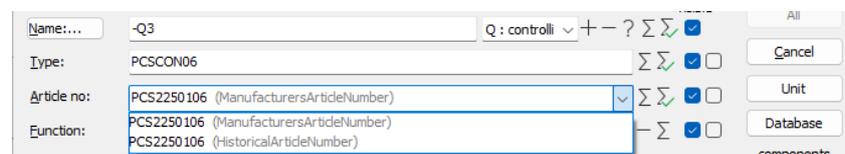
However, we have done something:

The field Article no. was linked to EANNUMBER in ver22, when the file is converted to ver23 it is linked to Historical ArticleNumber – which contains the old EANNUMBER, because it is imported into this field.



In ‘the old days’ the link to the database was the article number, ie the EANnumber, but from ver23 is the Component ID, which is seen at the bottom of the dialog.

If you press the drop-down arrow, you can see the other article numbers for the component.



Converted projects will always prioritize the Historical article number, so that the old components are recognized in the project itself and in the lists.

Old components are also usr-components, meaning that they only have data from the user – usr.

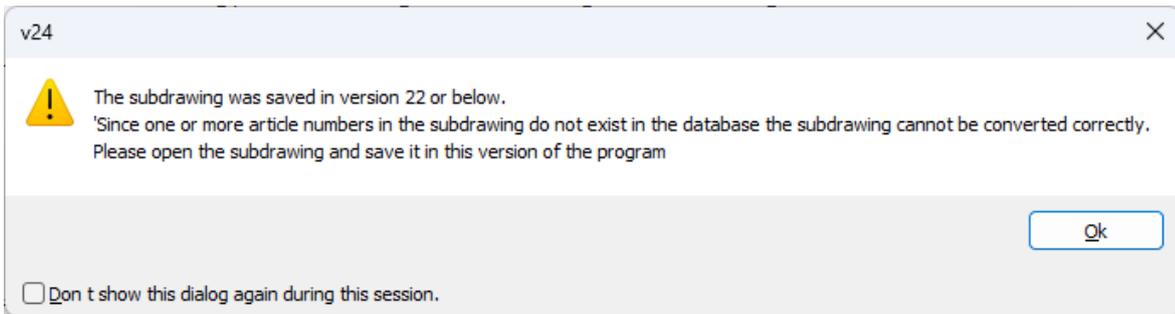
If the components are not in your database there is no Component ID at the bottom of the dialog.

Subdrawings and templates

When we convert a file, we change the connection between components and the database.

Templates and subdrawings are also files and as a starting point, the same rules are valid.

- If you want to use the files directly in your new installation, the components should be found in your new database. If not, you will get the message below.



- If the components are in the database, you can drag the old subdrawings directly into your new projects.
- If you save the file in ver24, you cant use it again with your old ver22.
- If you keep them in ver22 format, you can use them with both installations.
- Lists with database fields are changed to the new fields, ie MANUFACTUR from the old database is changed to Manufacturer.

As you get your components into the new database, the warning will appear less and less.

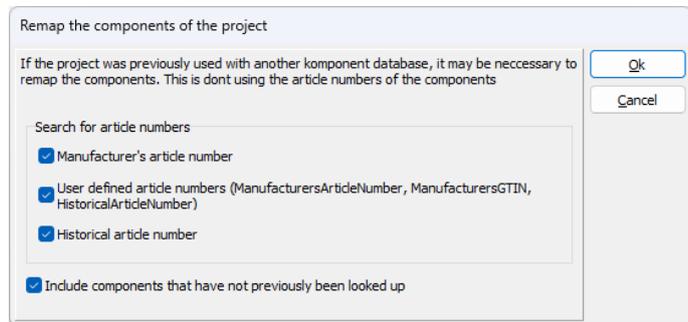


Remap the components to the database

If your project seems to have no connection to the component database, then there is a function in the Components menu – Remap the components....

Missing connection to the database can be due to

- you opened an old project and saved it without the components having a proper connection to the database
- The project has used another component database earlier



When you remap the components of the project, they keep their article numbers, but their ComponentIDs change.

ComponentID

The link to the database is the individual component's ComponentID. For components that you download from the Component Portal, the ComponentID is always the same.

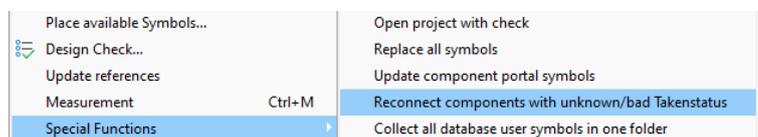
When you create components yourself and when you convert old components, then the ComponentIDs for these components are new and unique in your database. If you delete the component and create it again, it will get a new and unique ComponentID, different from its first one.

That means, that if/when you exchange files with colleagues, customers and suppliers, who use other databases – or if you start all over by converting your old database – you will experience, that the project loses connection to (some of) the components in the database. You reestablish the connection by using the Remap function.

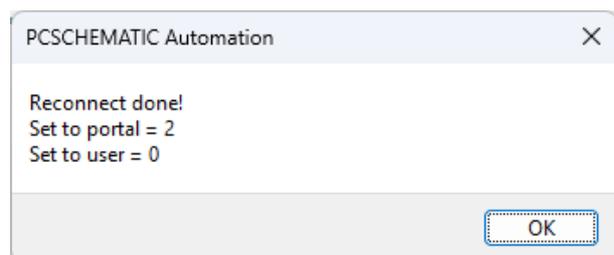
Components with mixed usr/pcs-symbols

Some components may have mixed symbols, ie symbols from both the usr and the pcs symbol fields. It is probably because, they were created with an early ver23.

You fix it by using the Function|Special Function here.



The components in the the current project are fixed – here there were two.



MORE ABOUT SETTING UP THE IMPORT WIZARD

On the next pages, you can read more about setting up the import wizard.

You can see the different options, among them how to import components from a project, a list and how to import a selection of (maybe all) components from your old database.

The settings you make are saved for your next import.

If you have used anything else than 'our' old EANNUMBER as article number, you can see examples of this on the following pages.

The wizard opens directly when you open a ver22 project, but you can open it directly via Components|Import components from ver22 database.

The starting point is that it is your own old projects that must be loaded, and that you therefore have access to your old database and the old symbol folders.

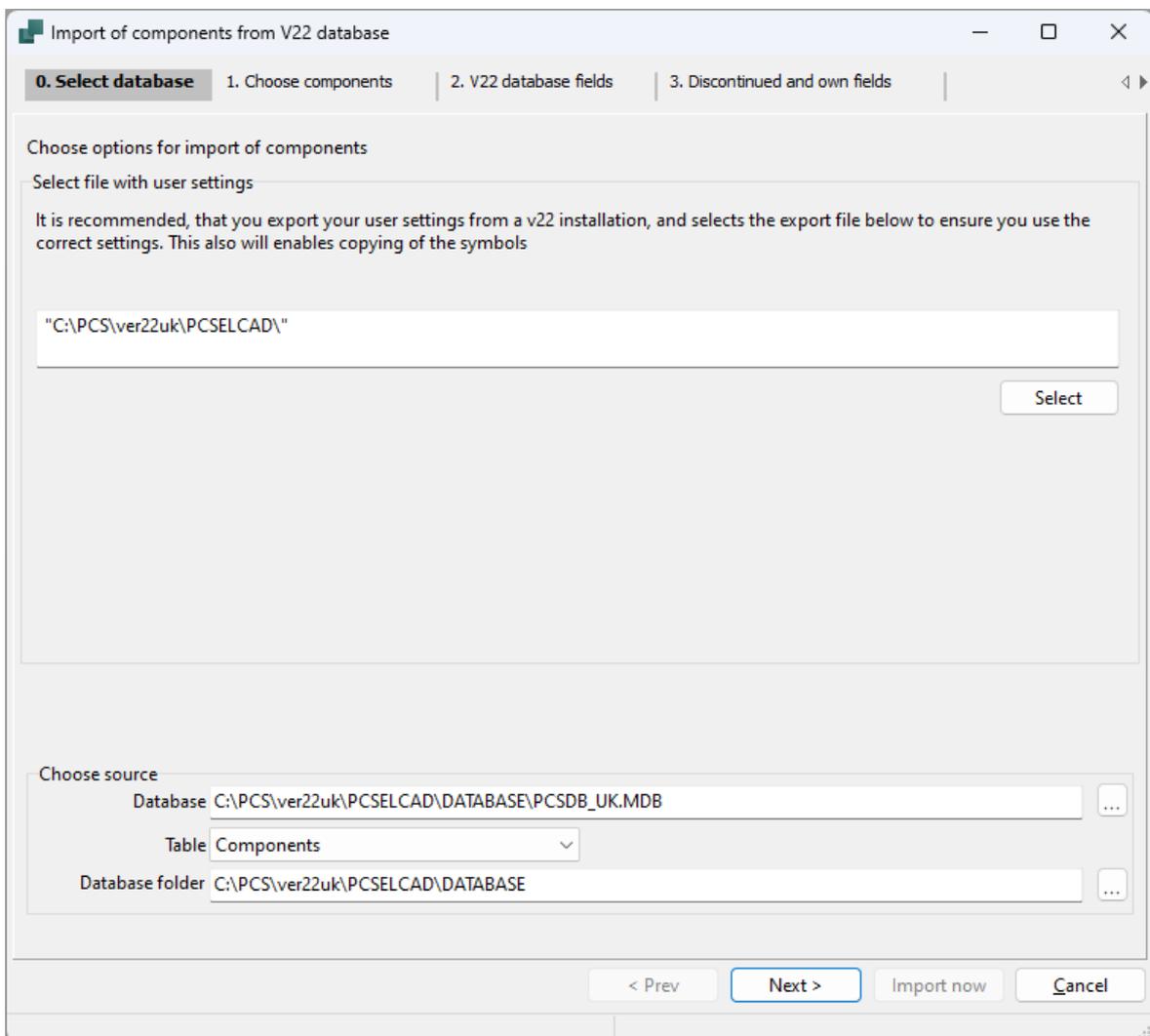
If you don't have it, you can easily open the file, but you can't convert the components if they are not in an old database.

You can download Automation ver22 from our webpage. It contains a database, with the components that we have created.



Tab 0 – Connect to the old database

In Tab 0 in the Import Tool, you need to point at your old ver22 installation, as this is the location that you fetch old components and symbols from.



Use your old settings

If you want to import your old symbols automatically, you need to export your old ver22 user settings, so that the program can fetch your old symbols. This is of course only the case, if you have access to the same drives with the new installation.

Your old settings-file also contains information about your last database, its tables, and folders, which is seen below.

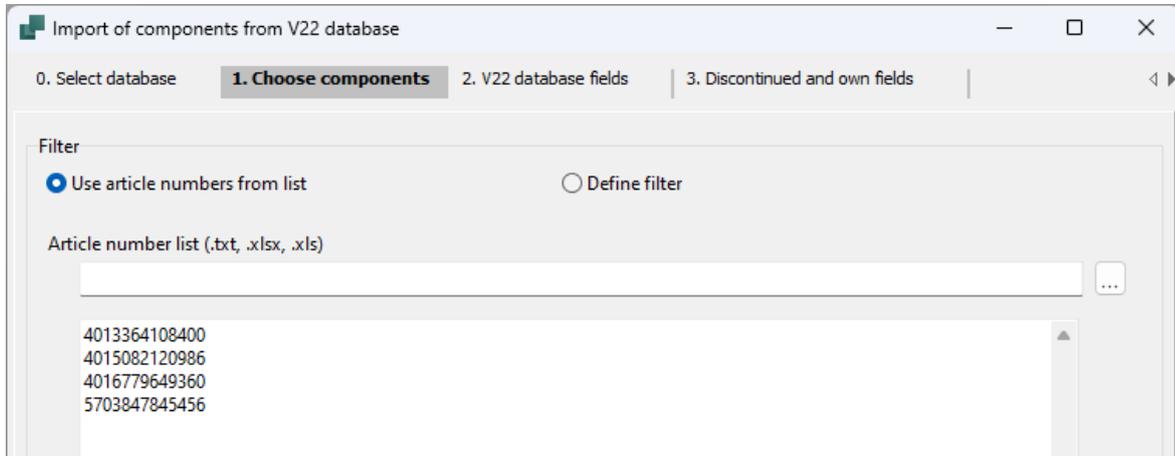
If you do not have your old settings

If you want to use another database or want to fetch component data only (and no symbols), you can select the desired database manually. In that case, you also need to select the table containing components (in our database it is called Components). If the database folder does not pop up automatically, you need to select it manually. The folder contains pin- and pcs-files and the Panelbuilder database. During the conversion, pin-, pcs- and Panelbuilder-data is automatically imported from the old files.

Tab 1 – Select components from a list or a filter

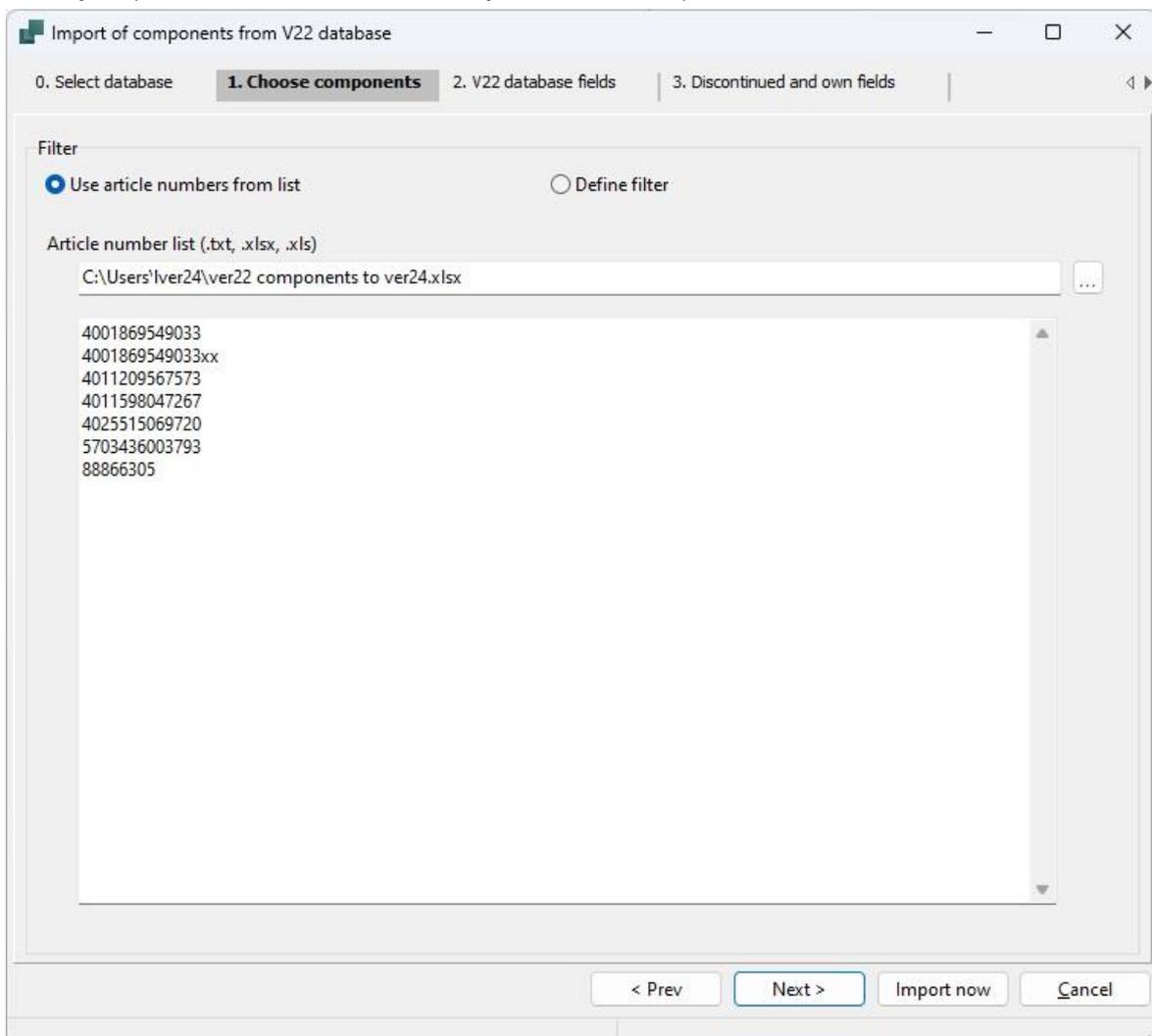
On this tab, you select the components that you want to import.

First option is to import the components in the window – this appears when you go directly to Import now after opening the ver22 file.



Option 2 is to select a list with components, like the list you can save during import of a project, see page 8 or page 10.

Here you press the ... to find the list you want to import.



Accessories

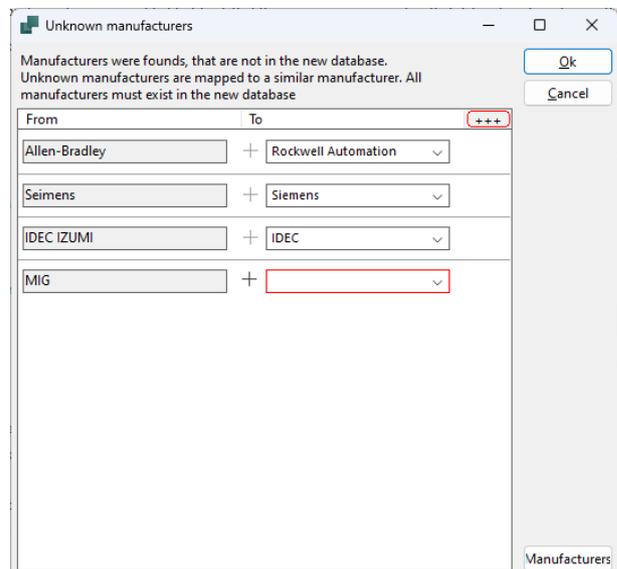
If the components contain information about any accessories, they are automatically added to the import – if they can be found in the database – and you can see the number of accessories at the bottom of the wizard.

Unknown manufacturers

If the components have unknown manufacturers, you get a list so that you can add and approve them.

Known manufacturers with new names (or spelling errors) are changed, unknown manufacturers are marked with red, and can be added to the list individually by pressing the single + at the side.

If you press the +++ button at the top, the program creates all unknown manufacturers in the new database with their old names.



Use filter to import a lot of components

You can select to import a lot of components directly from your old database by setting up filters.

Below, the filter is set to import surge protection devices from ABB.

Import of components from V22 database

0. Select database | **1. Choose components** | 2. V22 database fields | 3. Discontinued and own fields

Filter

Use article numbers from list Define filter

Column	Filter operation	Value	A<>a
DESCRIP	Contains	surge	<input type="checkbox"/> + -
And			
MANUFACTUR	Contains	ABB	<input type="checkbox"/> + -
And			
		Select a column first	<input type="checkbox"/> + -

Equal
Not equal
Larger than
Smaller than
Contains
Ends with
Begins with
Larger or equal

< Prev Next > Import now Cancel

You can set up a lot of filters on different fields from your old database, if you want to import a lot of components in one operation, ie

- 'Type' 'begins with' 'abc' -> fetches all components where Type begins with abc
- 'MyArticleNo' 'is larger than' '1' fetches all components that has a 'MyArticleNo'.



Tab 2 – Ver22 database fields that are imported

Here you see the mapping of your ver22 database, meaning that you can see which database fields are set up to be used in Automation.

Information about the mapping is in the old IDB-file.

If you used our standard database in ver22

You might miss some of the fields, either because they are not in your old database or because they are not mapped.

Data from the showed fields are always imported into the new database.

When you convert components from an old database with a default setting, the EANNUMBER comes into the new database's HistoricalArticleNumber field. If the old database was set up with an alternative article number, this number will come into the HistoricalArticleNumber2 field.

Read more about the other fields in the following.

Import of components from V22 database

0. Select database | 1. Choose components | **2. V22 database fields** | 3. Discontinued and own fields

Below are previously used mappings of fields. Check that the mappings are correct so that data is transferred correctly

Component data	
Article	EANNUMBER
Alt. article	
Type	TYPE
Function	
Description	UKDESCRIPT
Units/Pack	UNITPRPACK
Manufacturer	MANUFACTUR
Approved	APPROVED
Obsolete	OBSOLETE
Fixed accessory	ACCESSORY
Optional accessories	
Electrical	OPT_ADDON
Mechanical	OPT_ACCESSORY
Symbol	PCSTYPE
SLD symbol	SINGLELINESYMBOL
Symbol ref.	REFID
Connections	PINDATA
Mec. symbol	MECTYPE
Thumbnail	PICTURE
Menu link	TABLECODE
Preferred Link field	CATALOGUE

< Prev Next > Import now Cancel

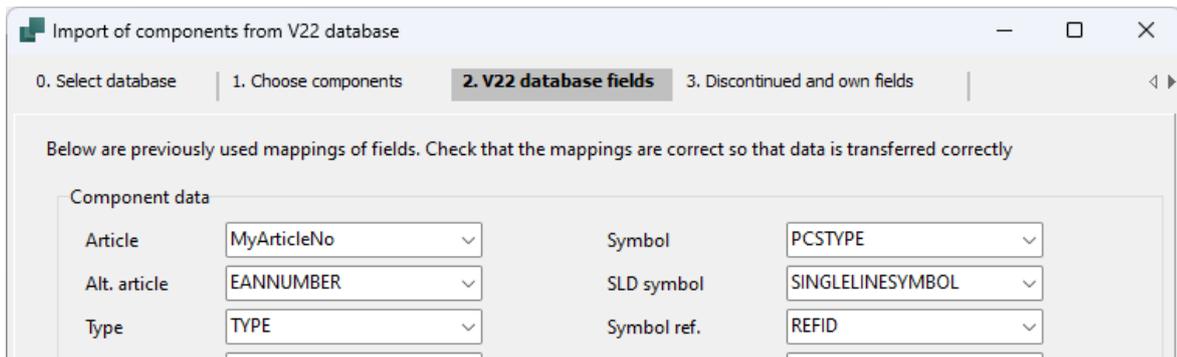
Import of 9 components (1 accessories)

If you used your own article number in ver22

In this (old) database, the field MyArticleNo contains the primary article number, and the EANUMBER is alternative article number.

When the components are imported from the old database, the contents from MyArticleNo are imported into HistoricalArticleNumber and the contents of the EANUMBER is imported into HistoricalArticleNumber2.

The field MyArticleNo is also created as an extra field in the ver24 database, see next section.



The screenshot shows a software window titled "Import of components from V22 database". The window has a progress bar at the top with four steps: "0. Select database", "1. Choose components", "2. V22 database fields" (which is currently selected and highlighted), and "3. Discontinued and own fields". Below the progress bar, there is a text instruction: "Below are previously used mappings of fields. Check that the mappings are correct so that data is transferred correctly". Underneath this instruction is a section titled "Component data" containing a table of field mappings. Each mapping consists of a label on the left and a dropdown menu on the right.

Component data	
Article	MyArticleNo
Alt. article	EANUMBER
Type	TYPE
Symbol	PCSTYPE
SLD symbol	SINGLELINESYMBOL
Symbol ref.	REFID



Tab 3 – Discontinued and own ver22 fields

On this tab, you can see the fields that we do not import by default, because they were not mapped in ver22.

If you have used some of those fields, that we don't use in the future – you must select them to transfer data.

The fields that you select here, will be created as new fields in the new database, and data in the fields will be included.

If you haven't used the fields actively, simply press Next.

If you used our default article number in ver22

If you created your own data fields in your old database, they are showed in **bold** writing at the top of the list, but you still need to select them, if you want them in the future.

Below, that are NOT used as article number previously, thus they are not selected!

Import of components from V22 database

0. Select database | 1. Choose components | 2. V22 database fields | **3. Discontinued and own fields**

Userdefined databasefields from the old database can be imported to the new.
If you haven't added your own fields (marked with bold) to the database, then press 'Next'

<input type="checkbox"/> MyArticleNumber	The field is not a standard field, but was added by a user. You should import this field if the values are to be transfered to the new database
<input type="checkbox"/> SettingsInterval	The field is not a standard field, but was added by a user. You should import this field if the values are to be transfered to the new database
<input type="checkbox"/> DEPTH	The field is no longer standard on the system. You should import the field if you have changed/added values to this field that you want to be transfered to the new database
<input type="checkbox"/> DISCOUNT	The field is no longer standard on the system. You should import the field if you have changed/added values to this field that you want to be transfered to the new database
<input type="checkbox"/> NETPRICE	The field is no longer standard on the system. You should import the field if you have changed/added values to this field that you want to be transfered to the new database
<input type="checkbox"/> ORDERNO	The field is no longer standard on the system. You should import the field if you have changed/added values to this field that you want to be transfered to the new database
<input type="checkbox"/> PB_MEC	The field is no longer standard on the system. You should import the field if you have changed/added values to this field that you want to be transfered to the new database
<input type="checkbox"/> PRICE	The field is no longer standard on the system. You should import the field if you have changed/added values to this field that you want to be transfered to the new database
<input type="checkbox"/> RATING	The field is no longer standard on the system. You should import the field if you have changed/added values to this field that you want to be transfered to the new database
<input type="checkbox"/> RENMARK	The field is no longer standard on the system. You should import the field if you have changed/added values to this field that you want to be transfered to the new database
<input type="checkbox"/> STOCKNO	The field is no longer standard on the system. You should import the field if you have changed/added values to this field that you want to be transfered to the new database
<input type="checkbox"/> TAX	The field is no longer standard on the system. You should import the field if you have changed/added values to this field that you want to be transfered to the new database
<input type="checkbox"/> WIDTH	The field is no longer standard on the system. You should import the field if you have changed/added values to this field that you want to be transfered to the new database
<input type="checkbox"/> ID	The field is deprecated. It is not recommended to import the field

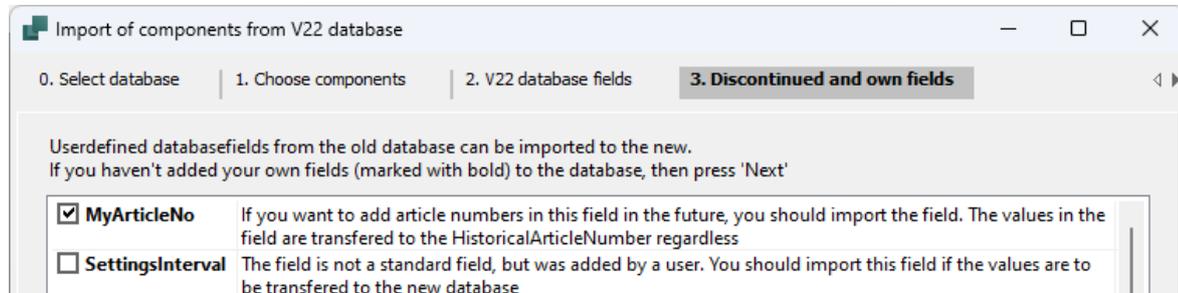
< Prev **Next >** Import now Cancel

Import of 90 components

If you used your own article number in ver22

Here, MyArticleNo is pre-selected when you import. This is because it was mapped as Article number in the old database.

The field is created automatically in the new database, and the contents of this field is imported into MyArticleNo and HistoricalArticleNumber in the new database.



If you want to use your own article number as MyArticleNo in the future, you need to set it up, read more from page 28.



Tab 4 – Mapping of fields from ver22 to ver24 database

On this tab you can see – and select – how the old data is transferred from ver22 to ver24. In the example below, you can see that the old EANNUMBER is transferred to HistoricalArticleNumber, as it was the historical article number,

This field **must** be filled out (and it happens automatically) for old components to make the program recognize an old, existing component.

Descriptions are transferred to DescriptXX, TYPE to Type etc.

It shouldn't be necessary to change anything, but it is a good idea to go through the settings during the first import, to see where data goes from and to.

Import of components from V22 database

1. Choose components | 2. V22 database fields | 3. Discontinued and own fields | **4. Mapping to new fields**

Choose which databasefields to import from 1:90 components

From	To
	ComponentID 1F9E7D50-8B2E-4B85-B368-929A:
MANUFACTUR WAGO	Manufacturer Wago
	ManufacturersArticleNumb UNKNOWN_1F9E7D50-8B2E-4B8:
EANNUMBER 4044918454124	HistoricalArticleNumber 4044918454124
	HistoricalArticleNumber2
TYPE 282-696	Type 282-696
CATALOGUE	usrDatashet
PICTURE %PIC%\282-696.jpg	Picture %PIC%\282-696.jpg
APPROVED False	usrFavoriteComponent False
SOURCE WAGO Danmark	usrFavoriteSupplier WAGO Danmark
UKDESCRIPT	DescriptUK

Show all fields

< Prev Next > Import now Cancel

Import of 90 components

If you used your own article number in ver22

If you used your own article number in ver22, the contents of this field will be imported into HistoricalArticleNumber and the field will also be created in the new database with the same contents.

This happens partly because the old projects are searching for the old article number in the HistoricalArticleNumber field, but new projects and components will also use MyArticleNo (when it has been set up).

From	To
	ComponentID 1F9E7D50-8B2E-4B85-B368-929A:
MANUFACTUR WAGO	Manufacturer Wago
MyArticleNumber	ManufacturersArticleNumb UNKNOWN_1F9E7D50-8B2E-4B85-B368-929A:
EANNUMBER 4044918454124	HistoricalArticleNumber 4044918454124

Show all fields

< Prev Next > Import now Cancel

Import of 90 components

MyArticleNo must be set up as article number in ver24. See how from page 28.



Tab 6 – Import components

The import itself happens on this tab.

Here you can see the result of your choices.

And a little info

In our Component Portal and in the new database, all components are created with a Manufacturer and a ManufacturersArticleNumber.

The components that we – PCSHEMATIC – had created in the old database and that can still be bought – are imported with the right article number. Because we know it.

'Our' discontinued components, and probably all components that you created earlier, get the ManufacturersArticleNumber 'UNKNOWN_XXXX'. You can change it later, but this is what it looks like for now.

Press the Import now, and the components are imported into your new database.

Import of components from V22 database

3. Discontinued and own fields | 4. Mapping to new fields | 5. Symbols | **6. Import components** | 7. Status for import

Ready to import 5 records.

	Manufacturer	ManufacturersArticleNumber	BuiltInDepth	CwCode	DescriptDE	D
1	Gewiss	20000				Ff
2	Siemens	345678				E
3	Phoenix Contact	30000				PI
4	OMRON	UNKNOWN_12DBF7A2-BE13-4813-BADD-B33B4122CC75				IV
5	Schneider Electric	10000				H

< Prev Next > **Import now** Cancel

Import of 5 components



Merge ver22 components with Portal components

If you have created components in your ver22 database, that are found on the Component Portal today, you can merge them by doing this:

- Import the old components to the new database
 - Check that the Manufacturer has the same name, ie AllenBradley is spelled Rockwell Automation!
 - The manufacturers article number – ManufacturersArticleNumber – combined with the manufacturer's name is the key. Therefore both fields must be identical!
- Download the same components from the Component Portal
 - The components are merged: you keep your old symbols in the usr-fields, and you get the new symbols in the pcs-fields.

DATABASE SETTINGS IN VER24

The database settings determine how Automation retrieves information in the database.

Many of the settings are fixed from ver23, which should result in fewer questions in the long run.

Below is an overview of the most important settings.

Which database and other basic settings

The selected database is shown at the top. In ver24 it MUST contain our fields as a minimum.

The project database, i.e. what is stored in the file – we recommend system and list fields.

The project database can be prioritized in terms of updating lists - similar to the old 'locked project database'.

Setting up the article number, i.e. which article number is displayed in the symbol's article number field and thus in lists, can be prioritized based on the article numbers you have in the database by pressing the 'cogwheel'.

Primary data source – or rather symbol selection – when there are symbols both from the portal (pcs) and from yourself (usr) – here you can choose which one you prefer.

Most people will probably choose usr as first priority.

The screenshot shows the 'Settings' dialog box with the 'Database' tab selected. The 'Database file name' is set to 'C:\PCS\PCAutomation24\PCAutomation\Database\MY_new.mdb;Components'. Under 'System settings', the 'Database' is set to 'On', and 'Database fields saved in project files' is set to 'System fields and fields used in lists'. The 'Update symbols from database' section has 'Update Components when changing article no.' checked. Under 'Project settings', the 'The component's article number comes from the database field ...' is set to 'Default' with a list of fields: 'MyArticleNo, ManufacturersArticleNumber, HistoricalArticleNumber, ManufacturersGTIN'. The 'Project component database' section has 'Prioritize when updating lists (locked)' and 'Cache update' unchecked. The 'Primary data source' is set to 'Using user data as first priority' and the 'Number of letter codes for the project' is set to '3-letter code'. The 'Ok' and 'Cancel' buttons are at the bottom right.



On the 'Basis mapping' tab, you can set up your own article numbers and choose a description from the database. Most other settings are done.

'MyArticleNo' is chosen as an article number here, and therefore it can also appear in the list above and therefore on the symbol. And you can also search for it in the database.

All symbol selections are set in the system. All symbol fields are twin fields – pcs or usr. See the Symbols mapping tab.

All selections for accessories are set in the system, also with twin fields; here though you get a distinct list of accessories instead of a selection between pcs and usr.

Converted projects

When a project is converted, then

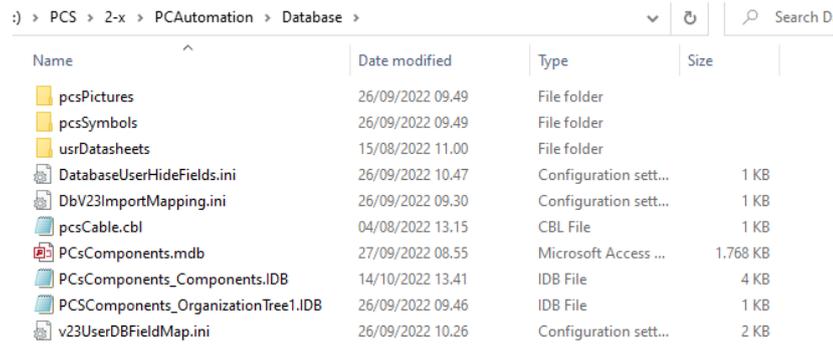
- The HistoricalArticleNumber has the highest priority among article numbers. All components found in the database use this article number
- The primary data source is User data – all components are 'old' and a lot of them only has user data.

You can change this, if you want to continue working with the project.

MORE ABOUT THE DATABASE FOLDER

The database folder contains the database (*mdb) itself and the other files, that you can read about below.

The location for the folder is selected in Settings|Directories.



Name	Date modified	Type	Size
pcsPictures	26/09/2022 09.49	File folder	
pcsSymbols	26/09/2022 09.49	File folder	
usrDatasheets	15/08/2022 11.00	File folder	
DatabaseUserHideFields.ini	26/09/2022 10.47	Configuration sett...	1 KB
DbV23ImportMapping.ini	26/09/2022 09.30	Configuration sett...	1 KB
pcsCable.cbl	04/08/2022 13.15	CBL File	1 KB
PCsComponents.mdb	27/09/2022 08.55	Microsoft Access ...	1.768 KB
PCsComponents_Components.IDB	14/10/2022 13.41	IDB File	4 KB
PCSComponents_OrganizationTree1.IDB	26/09/2022 09.46	IDB File	1 KB
v23UserDBFieldMap.ini	26/09/2022 10.26	Configuration sett...	2 KB

If you have a *sql-database

Then the udl-file should be in this folder. The database itself is in another location.

Files and subfolder

pcsPictures

When you download components from the Portal, pictures for every component are seen in the database, in the Component data dialog and maybe in parts lists and more. The pictures are downloaded to this folder. You can also save your own pictures here.

pcsSymbols

All downloaded components come with symbols, which are downloaded to this folder. They can be normal diagram symbols, plc symbols, mechanical symbols etc. The symbols are automatically transferred to the relevant folders when you open Automation. Only the Portal saves symbols here.

usrDatasheets

You can save your datasheets for your components here. Remember, you have to find the datasheets yourself.

PcsComponents.mdb

Our ver24 database, which contains all required data fields. You may make a copy of it so that your own database has another name, preferably using the database program.

*idb-filer

Database settings for the database and some tabels, respectively.

DatabaseUserHideFields

If you have chosen to hide fields in the database record window, information is saved here.

DbV23ImportMapping.ini

Settings made relating to database import is saved here, so that you don't need to decide every time.

V23UserDBFieldMap

More database settings.

*cbl-filer

Text files, that may be used, when you create cables in an easy way.



WHY CONVERT – WHAT DID WE DO?

Automation ver24 works in the same way as the program has done always! But, from ver23, we decide the database structure. And that means, that components from the old – and more free structure – must be converted to the new – more fixed – structure.

For most users, it will be a simple pressing the Next-buttons, for a few, there is a little more work.

And everybody wants to know, what happens, thus this small booklet.

The changes mean, that you need to *install* the first time (ver23 or ver24) instead of *updating* the old installation. The next version, however, is a normal update 😊

The major changes from ver22 to ver24

From ver23 you can download components to your own database from a **Component Portal**. To be able to use the Portal data, you need a database that complies with our requirements. We have made a new database structure, that supports the requirements. That also means, that you need to convert data if you want to keep working with your ‘old’ components.

Old settings – ie text settings and printer settings – are not included, when you *install* ; when you *update* you keep your text and printer settings.

We have **renamed all standard symbols**, and the new symbol names are used on the Portal. That means that when you want to keep working with existing components, you must be able to access old symbols. This happens when you convert.

Finally, we have expanded our **plc functions** from ver23. We have made new symbols, and new symbol datafields, that support some longtime missing functions. The new symbol datafields are created in the program, so that they work as ‘normal’ datafields on the plc symbols, and this in itself needs a new installation. You can see a short overview of those functions at the end of this document.

Don't delete your old installation

The three points above means, that you can't make a normal update of the old installation, but that you need to make a more extensive conversion, if you want to continue working with your existing projects.

That is the reason why we advice you to keep your old ver22 installation and only transfer the projects you want to be able to continue working on. And do it as the need arises.

New database structure

The database contains a lot of new data fields, and ALL these fields must be in the database, meaning that you cannot delete the uninteresting fields as you could earlier. However, you can hide selected fields.

All 'our' fields are meant for a purpose, and we have a hint on all fields with info about their purpose. Ie, we have a lot of fields which purposes are to keep hold of the connection to our Portal and other fields with info about of article numbers.

We have double fields for all symbols – pcs and usr – which makes it possible to have our symbols as well as your own symbols for the same component.

Also important, all pin-files are gone, as the new database allows you to have all data in the database itself, and the separate Panelbuilder database has also been included into the new database.

You can create own datafields as previously.

Open the Component database, select and open a component to see this.

The important difference between article numbers in ver22 and ver24

In ver22, the link between the project's components and the database is not unanimous: the component's article number is (default) fetched from the database's EANNUMBER field. However, you can also choose another field to contain the article number, often a 'SAP-number', and finally you can choose to have an alternative field if your primary field is empty. This means, that the article number = the database link is not unanimous across projects but works ok in the individual projects.

This is different in ver24, as every component is linked directly to a componentID in the database. The article number, that you use in the project is chosen between different article numbers, including ie SAP-number.

Which article number to use/display can be prioritized and changed in the individual project.

When we convert a project from ver22 to ver24, we automatically set the project to fetch its article numbers from the database's HistoricalArticleNo-field, as this will make it have the same values as previously. Remember though, that the project's components must be in the database, otherwise you don't have a correct link to the database.



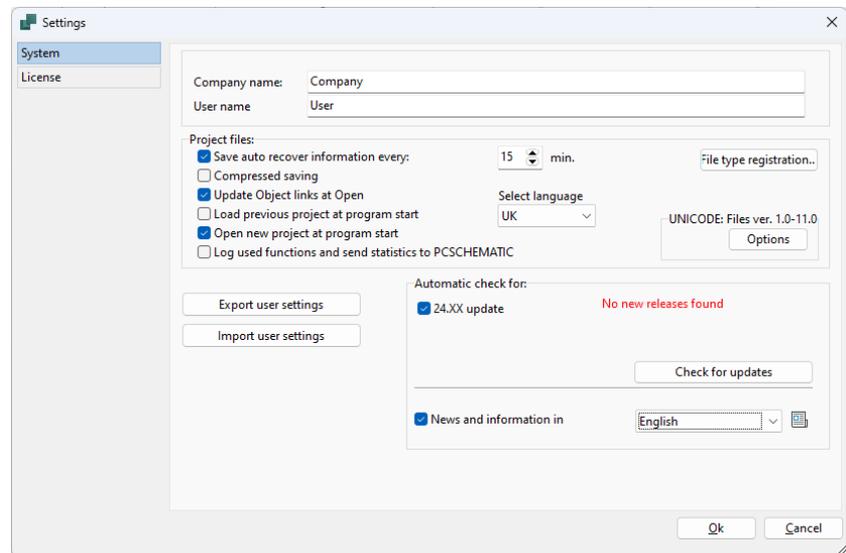
Old settings

When you update your installation, as you have done many times, then you also for better or worse keep your user settings.

That means, that many of the changes in the program, that are visible for *new* users when they open the program, stay 'secret' for for *old* users.

Old settings are many things, and some of the old settings, you want to use also in the new installation.

On the Settings| System tab, you can – from ver22 – *export* user settings, and in ver24 you can import and export user settings.



Use this function from now on, if you want to copy settings to your co-workers. DON'T copy the ini-file! It doesn't contain all settings anyway, and the receiver of it, risk to damage some important settings ...

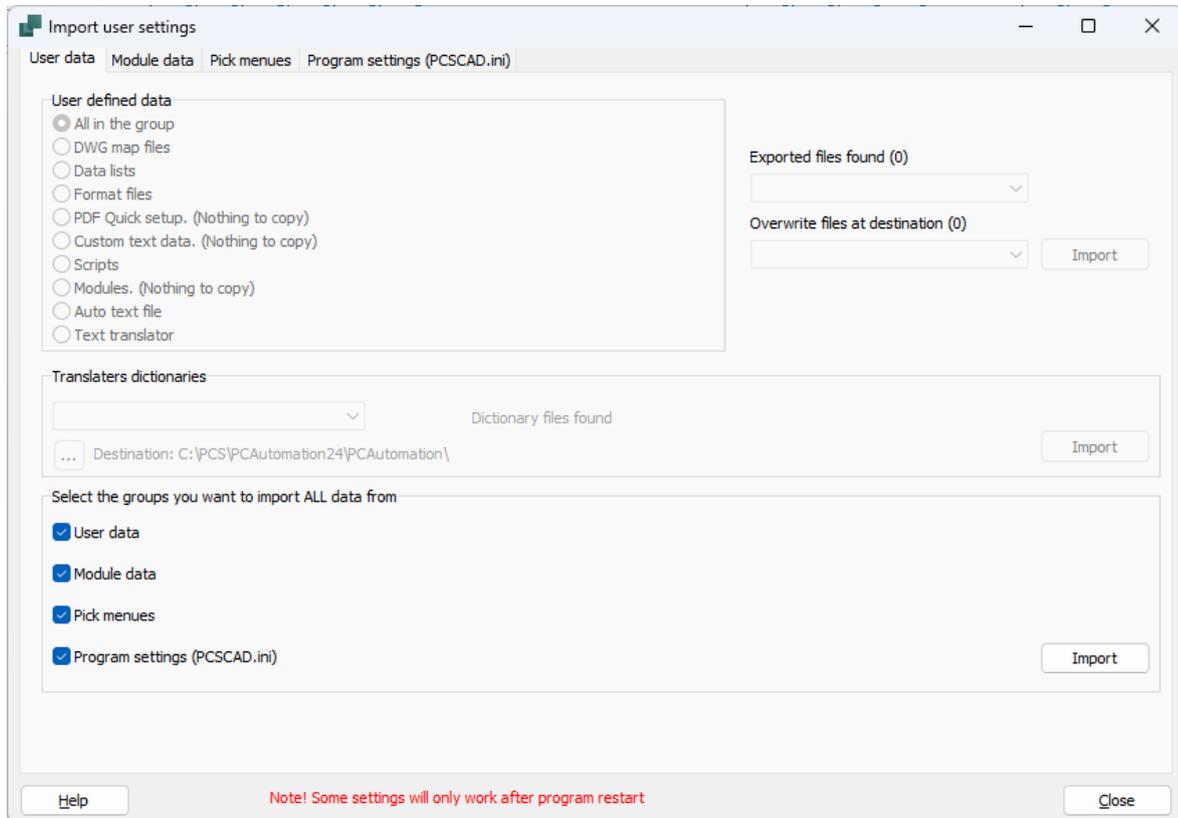
Import user settings

In the 'old days', many users would copy ini-files to each other, because they thought, that it contained all settings. It does/did not. That is the reason why we have made the function Export/Import user settings.

When you *import* user settings, you get the dialog, where you are guided through various program settings. You can press Help to open a document that explains all settings.

Here you can import the different settings from another user. There has been a wish to make it possible to import 'the same settings as Peter'. This is the reason why the first tab looks like this.

The import is divided into different sections, and you can select settings from one or more sections.



New symbols

In ver23/24 we have renamed all IEC symbols, so that they have the name from the standard IEC60617.

That means, that a terminal is renamed from 03-02-02 to PCS-S00017.

When you download a terminal from the Portal, you automatically get the new symbol (also when you have it already). When you work with old components, you might not have the old symbols (anymore).

The new import function from ver24 imports the used symbols.

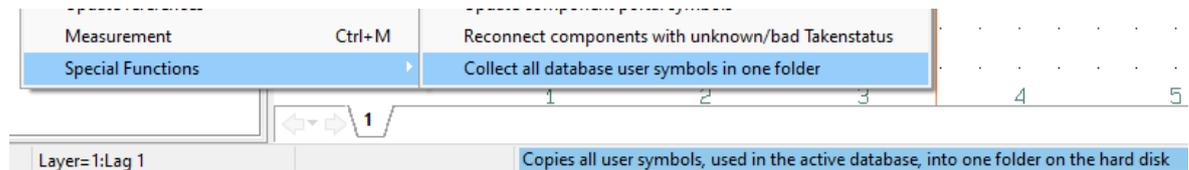
We use the old settings to find the symbols for old components when we import components.



If you have imported the full ver22 Alias list ...

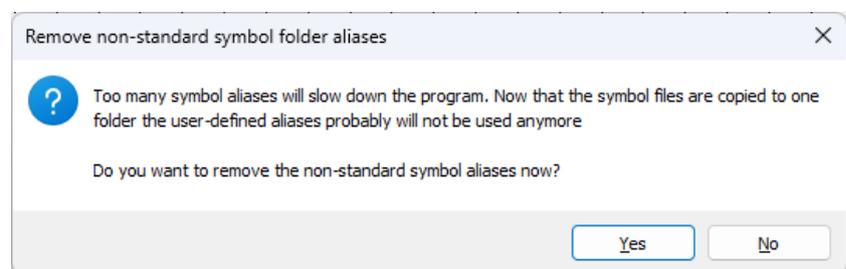
If you have imported all old Alias'es, and in that way have gotten a long, long list, it is possible to get all used symbols out of the many folders and into the OldSymbols folder as described earlier.

In Functions|Special functions, select the 'Collect all database user symbols in one folder'.



You should also tidy up in the Alias list itself, as a long list slows down the program.

When you run the function above, it can be done automatically. You could also do it manually.



You don't delete symbols or folders, you only delete them from the Alias-list!

We have created new PLC symbols and several functions in ver23

And maybe it could be an idea to consider going over to these.

All plc components on the Portal use these symbols.

Among the advantages is that all components use the same symbol when it comes to the same function. In addition, we have expanded the Component Guide's functionality, so that you can create PLCs in a simple way, and you can edit (some) existing PLCs.

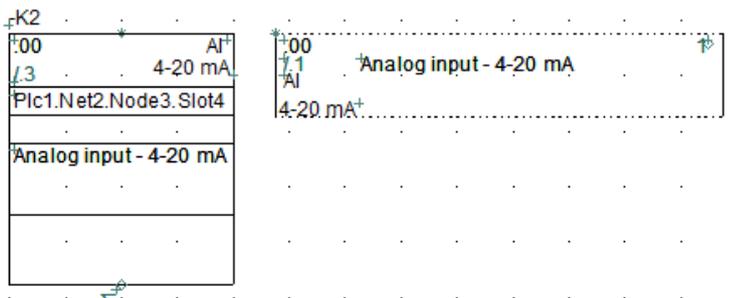
Finally, the symbols contain other functions that have been in demand for a long time, such as PlcID (plc.net.node.slot) and signal type, both of which are displayed on the symbols.

New PLC symbols

The IO symbols themselves and the other symbols in the 'field' have been given a uniform appearance:

-K1	-K1	-K1	-K1	-K1
.00 /.3	I .01 /.3	AO .02 0-20mA /.3	O /.3	
Plc1.Net2.Node3.Slot4	Plc1.Net2.Node3.Slot4	Plc1.Net2.Node3.Slot4	Plc1.Net2.Node3.Slot4	Plc1.Net2.Node3.Slot4
Label - Input	Label - output	Label		
Description. By default, we display an 'I' for inputs	Description for the output. Signal type is displayed, too	Output with five connections (large ref.symbol)	Symbol for the plc's communication	Symbol for supply (no cross reference to ref.symbol)

- The width is 1 current path (40 mm)
- All connections point downwards
- New symbol data field – PlcID (composed of Plc.Net.Node.Slot) – on all symbols
- Space for label and description
- Space for description on connections
- All texts in Arial, 2.5 mm



Connection details

On terminal list

Is I/O connection

I/O direction: Input Output

I/O status type: Analog Digital Generic

I/O Status type:

With reference

No dot generation Lock I/O address



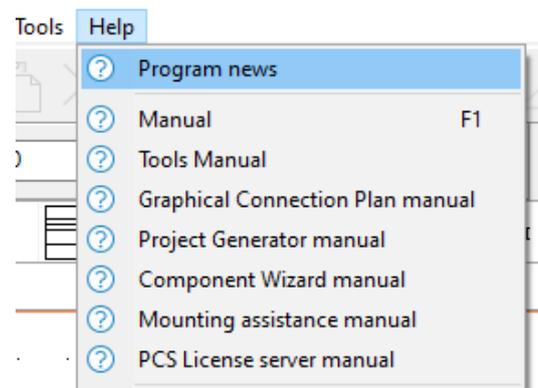
The reference symbols have also changed appearance:

- Width is 2 current paths (80 mm), meaning you can have 4 columns on a page
- 15/30 mm high per address
- Common top, common bottom which is automatically grouped together with the symbols per address
- All texts in Arial, 2.5 mm

-K1		Plc1.Net2.Node3.Slot4
.00 /2 I	Label - Input Description. By default, we display an 'I' for inputs	1
.01 /3 AO 0-20mA	Label - output Description for the output. Signal type is displayed, too	2 3
.02 /3 O	Label Output with five connections (large ref. symbol)	4 5 6 7 8
CH0 /5	Symbol for the plc's communication	9

You can read more ...

You can read much more in the Program News document for ver23 and ver24. Find it in the Help menu.



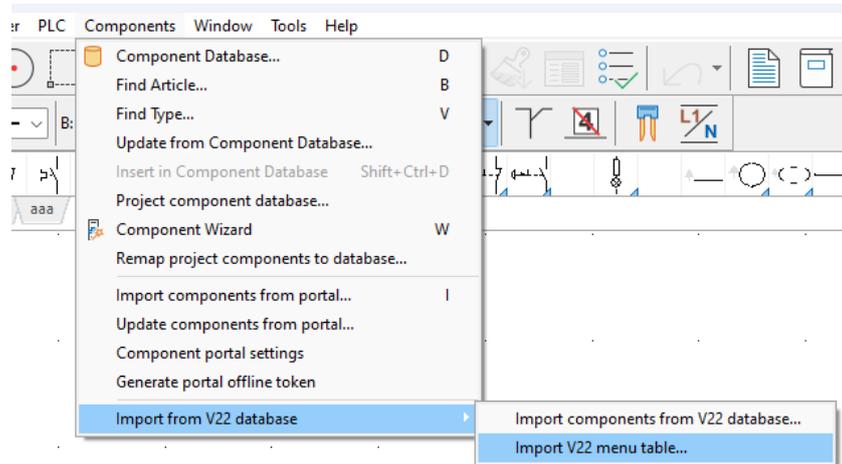
You can also read about the PLC functions in the 'Get a great start with PC|Automation', that you can download from our webpage.

Your own menu table

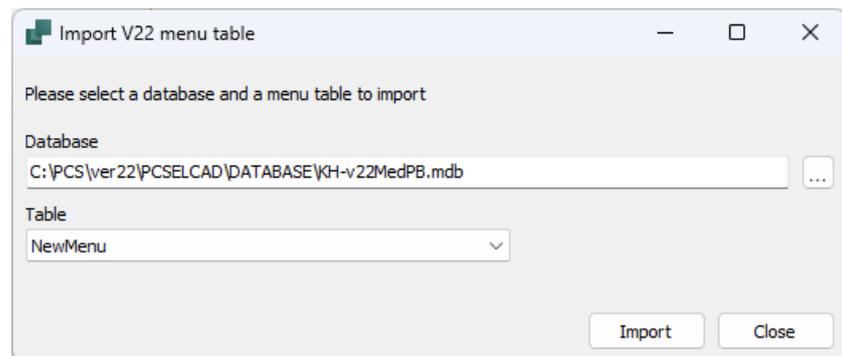
The previous pages have been dealing with the new database's component table, which is the table that must comply with our requirements.

The database also contains a menu table, that divides the components into different groups. Our default menu table is identical to the one we use in ver22. If you have made your own menu table, it is possible to import this one, too.

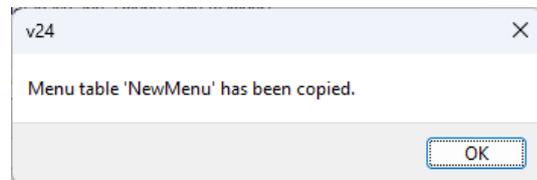
The function is found in the menu Components.



Select the database with the menu table, you want to import to your connected database.



Click Import and it is copied.



If you use your own menu tabel, you need to put the components into the right group yourself – the Component Portal only knows our table and links to it.



MY NOTES





