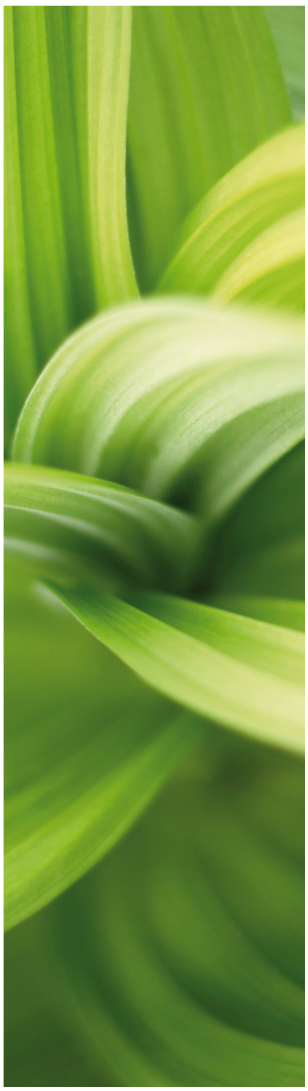


AUTOMATION

DATABASE

Developed by: PC|SCHEMATIC A/S Bygaden 7 4040 Jyllinge Denmark
T: +45 4678 8244 www.pcschematic.com info@pcschematic.com

PC|SCHEMATIC

TABLE OF CONTENTS

PREFACE 5
PREFACE 7
SURVEY OF THE MANUAL 8
TYPOGRAPHY 9

GETTING STARTED 11
ON DATABASES 13
GETTING STARTED WITH PC|SCHEMATIC DATABASE 16
THE WORK AREA 18

BASIC FUNCTIONS 21
DISPLAYING THE CONTENTS OF DATABASES 23
EDITING RECORDS 28
SEARCHING IN DATABASES 32
SQL QUERIES 35

MAINTAINING DATABASES 37
CREATING MICROSOFT ACCESS DATABASES 39
CREATING TABLES 42
EDITING TABLE PROPERTIES 48
CREATING TABLE MENUS 51
CONNECTING TABLE MENUS TO TABLES 59

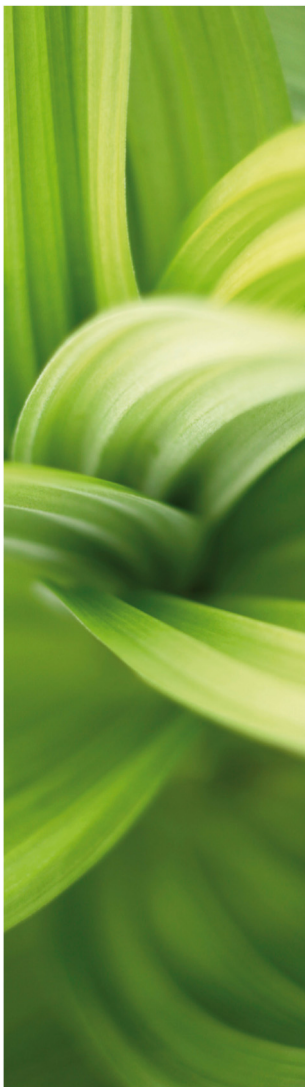
IMPORTING DATA 61
IMPORTING DATABASES FROM COMPONENT VENDORS 63
IMPORTING COMMA SEPARATED FILES 69
FILTERED IMPORT 73
IMPORTING MICROSOFT EXCEL DOCUMENTS 78

EXPORTING DATA 81
EXPORTING COMMA SEPARATED FILES 83
EXPORTING TO MICROSOFT EXCEL 87
GIVING AWAY DATABASES 91
LOADING DATABASES INTO OTHER PROGRAMS 94

OTHER DATABASE FORMATS 95
SUPPORTING OTHER DATABASE FORMATS 97

MDAC AND UDL FILES 99
CREATING BDE DATABASE ALIASES 103

INDEX 109



Preface

PC SCHEMATIC

PREFACE

PC|SCHEMATIC Database is a database program, developed by the Danish software house PC|SCHEMATIC A/S.

The program enables you to import databases from component vendors, and adapt the databases as you please.

PC|SCHEMATIC Database works directly in the database formats Microsoft Access and dBase, and can furthermore work in practically any other database format.

PC|SCHEMATIC Database works with the following programs:

- **PC|SCHEMATIC Automation** for documentating wiring- and power control circuit diagrams, as well as PLC diagrams.
- **PC|SCHEMATIC Electronics** for documentation of electronic circuits and for transferring data to PCB layout.

PC|SCHEMATIC A/S also develops software for interfaces, and user-defined modules for the **PC|SCHEMATIC** programs.

SURVEY OF THE MANUAL

This manual is divided in the following sections:

PART	DESCRIPTION	SUBJECTS
"GETTING STARTED" (P. 11)	SHORT INTRODUCTION TO THE PROGRAM	WHAT IS A DATABASE STARTING THE PROGRAM OPENING A DATABASE THE WORK AREA
"BASIC FUNCTIONS" (P. 21)	HOW TO WORK WITH THE CONTENTS OF A DATABASE	DISPLAYING THE CONTENTS OF A DATABASE SELECTING, DELETING AND INSERTING SEARCHING IN A DATABASE FINDING AND REPLACING ADVANCED FILTERING WITH SQL
"MAINTAINING DATABASES" (P. 37)	HERE YOU ADAPT THE DATABASE YOUR NEEDS	CREATING DATABASES CREATING TABLES EDITING THE PROPERTIES OF A TABLE CREATING TABLE MENUS CONNECTING A TABLE MENU TO A TABLE
"IMPORTING DATA" (P. 61)	IMPORT OF COMPONENT VENDOR DATABASES AND OTHER RELEVANT FILES	LOADING A COMPONENT VENDOR DATABASE IMPORTING A COMPONENT VENDOR DATABASE INTO A TABLE IMPORTING COMMA SEPARATED FILES OR ZIP FILES FILTERED IMPORT IMPORTING AN MICROSOFT EXCEL SPREADSHEET
"EXPORTING DATA" (P. 81)	EXPORT OF DATABASES - OR PARTS OF DATABASES - TO OTHER PROGRAMS	CREATING A COMMA SEPARATED FILE FOR EXPORT EXPORTING TO MICROSOFT EXCEL GIVING AWAY A DATABASE LOADING A DATABASE IN OTHER PROGRAMS
"OTHER DATABASE FORMATS" (P. 95)	SUPPORT OF OTHER DATABASE FORMATS THAN MICROSOFT ACCESS AND DBASE	SUPPORTING OTHER DATABASE FORMATS MDAC AND UDL FILES CREATING BDE DATABASE ALIASES

— Please also apply the index on page 111.

Typography

The following typographical elements are used in this manual:

Typography	Use
Italics	A text written in *italics*, is a text you have to type in yourself.
Bold	Texts written in **bold** — not including headlines — indicates functions, buttons or menus.
Italics and bold	Applied for file names — or for options you can click on in dialog boxes.
Keys ***< >*** ***[ ]*** ***[Ctrl+G]***	The control key is written as <Ctrl>, the Escape key as <Esc>, the Enter key as <Enter>, the Delete key as <Del>, while the Spacebar is written as [Spacebar]. Single letters in brackets, such as [z], indicates that the "z" key is a short-cut key. If it for instance says [Ctrl+G], this signifies that you activate the short-cut by pressing down <Ctrl> and press down the "g" key.
Menus ***=>***	**File => Save** indicates that you have to select the menu File, and then Save. An underscored letter, such as **H** in **Help**, means that the menu can also be activated by pressing [Alt+H].

Furthermore, if you have to activate one of the program buttons, you will see the mentioned button in the left margin of the text — facing the place where the button is mentioned.

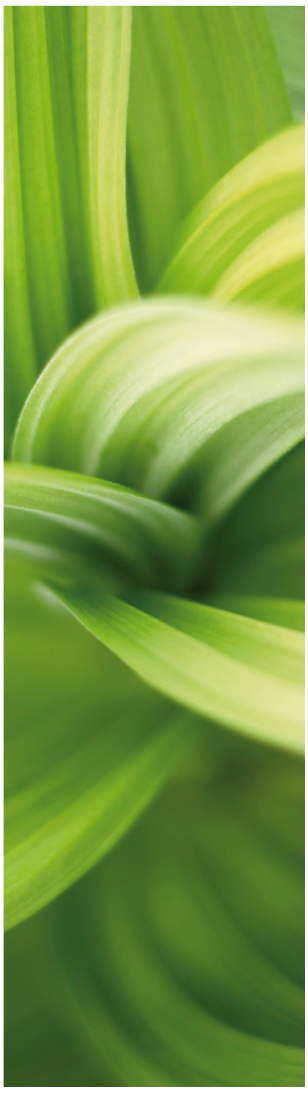
Buttons mentioned shortly after one another, are displayed in the order in which they are mentioned.

Short-cut keys

In the program you can use the following short-cut keys:

Command	Short-cut	Command	Short-cut
Copy	[Ctrl+C]	***Insert***	[Ctrl+V]
Cut	[Ctrl+X]	***Open***	[Ctrl+O]
Find	[Ctrl+F]	***Replace***	[Ctrl+R]
Find next	[F3]	***Show used only***	[Ctrl+U]

Short-cut keys in PC|SCHEMATIC Database



Getting started

This section covers the following subjects:

- What is a database.....(p. 13)
- How to start the program.....(p. 16)
- How to open a database.....(p. 16)
- The work area .....(p. 18)

On databases

This chapter describes
This chapter explains the terms *database*, *table*, *record*, *data field* and *field name*.

If you are already familiar with the database concept, please skip the first part of the chapter, and jump to “Table menus” on page 14 and continue reading from there.

What is a database?

A database is a collection of information, which has been gathered in a number of tables.

The contents of a database

A database in PC|SCHEMATIC typically just consists of one table containing component information (and one table containing a menu system for grouping the components).

All information placed on the same row in a *table*, contains information about the same component. This collection of information about the same component is called a *record*.

Each individual record consists of a number of *data fields*, each containing *one* type of information about the component.

The top row contains the names of the individual data fields, the *field names*.

Below you see a small table, containing five records (components), each containing four pieces of information about each component (EANNUMBER, STOCKNO, ORDERNO and TYPE). This information is placed in individual data fields:

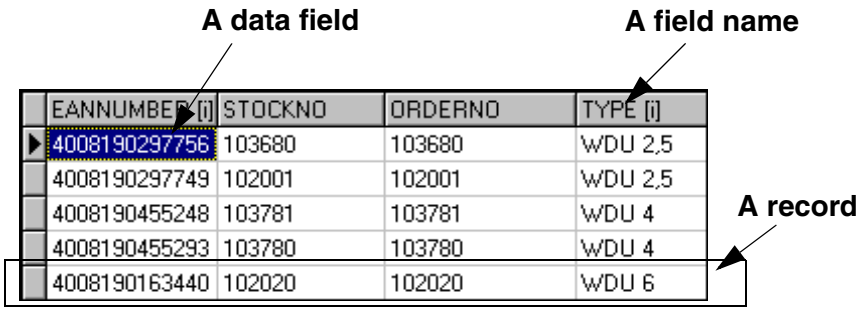
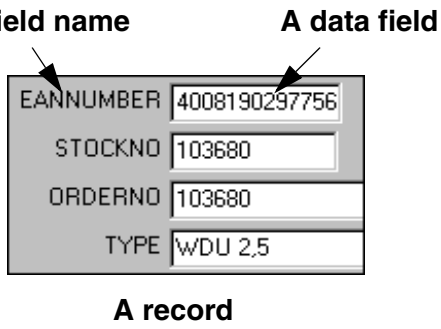


***Table* containing five *records*.**
Each record contains four *data fields*.

A *table* typically consist of many records. These records can either be displayed several at a time in a table, or individually, like below.

In a table containing four data fields for each component, an individual record can also be displayed like this:



Thus you distinguish between ***Record view*** and ***Table view***. Please read more about this in “Table or Record viewing” on page 24.

TABLES IN PREVIOUS VERSIONS OF PC|SCHEMATIC Database

IN PREVIOUS VERSIONS OF **PC|SCHEMATIC Database**, THE TABLE CONTAINING COMPONENT INFORMATION WERE REFERRED TO AS “A DATABASE”.

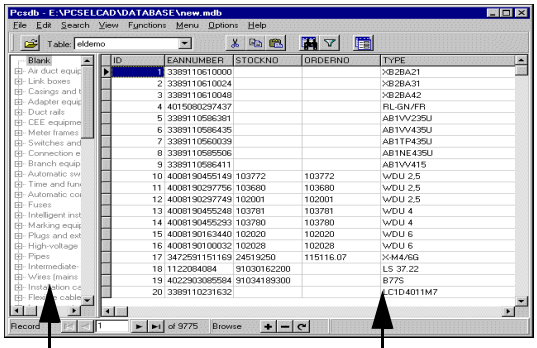
IN THIS VERSION THE COMPONENT INFORMATION IS PLACED IN A *TABLE IN A DATABASE*. — READ MORE ABOUT DATABASES AND TABLES IN “MAINTAINING DATABASES” ON PAGE 37.

Table menus

In **PC|SCHEMATIC Database** you can furthermore group the components in a table, by indicating which menupoints they belong to — for instance by component function.

These menupoints are displayed in the left hand side of the database program, and are called the *table menu*.

You can choose whether you wish to see all the components of the table, or just the components belonging to a specific menupoint.



— Read more about this in “Table menus in PC|SCHEMATIC Database” on page 24.

Working in PC|SCHEMATIC tables

When you edit the contents of a table, the changes are carried out the moment you have finished editing one record, and start working on another.

Thus, the changes are carried out on the fly, which do not leave you any possibility to cancel the changes by not saving the table.

If more than one person is working on a table in a network, it is very important that only one version of the table exist — and that this table is being updated for all the users at the same time, each time a record is changed.

If you need to test the database program, you should therefore start by making a copy of an existing table. This gives you a table, in which incorrect changes are not fatal. See how to do this in “Creating a new Microsoft Access table” on page 42.

One table at the time

You can only work in one table at the time. Therefore, if you are working in one table, and then open another, this automatically causes the first table to be closed down. This is done without prior warning.

GETTING STARTED WITH PC|SCHEMATIC DATABASE

THIS CHAPTER DESCRIBES

— HOW TO OPEN **PC|SCHEMATIC Database**.....(P. 16)
— HOW TO OPEN A DATABASE IN **PC|SCHEMATIC Database**.....(P. 16)

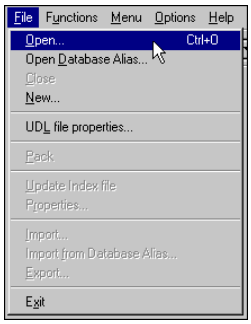
STARTING PC|SCHEMATIC Database

You can either start **PC|SCHEMATIC Database** by choosing **Start => Programs => PCschematic => Database** in Windows, or by choosing **Tools => Database** in the **PC|SCHEMATIC Automation** program.

When you only wish to open a table in a database and add records, delete records, or change the contents of a record, you can do this at any time, without taking any special action.

You have now started **PC|SCHEMATIC Database**.

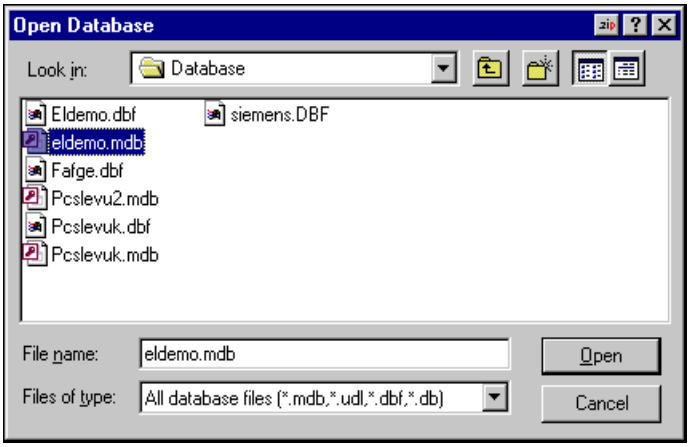
OPENING A DATABASE



In order to have a database to work on, you can for instance open the demo database **eldemo.mdb**, which comes along with the program.

In the emerging **Database dialog box**, you click on **File**, and then on **Open**.

You then enter the **Open** dialog box, where you choose which database to open:



Click on **eldemo.mdb**, and then on **Open**.

You have now opened the database **eldemo**.

You must the choose which table to open:

Click on the *table* **Eldemo** in the database **eldemo.mdb**.

If you need to change to another table in the database, click on the down-arrow in the ***Table*** field in the upper-left corner of the program.

SEE THE FULL CONTENTS OF THE TABLE

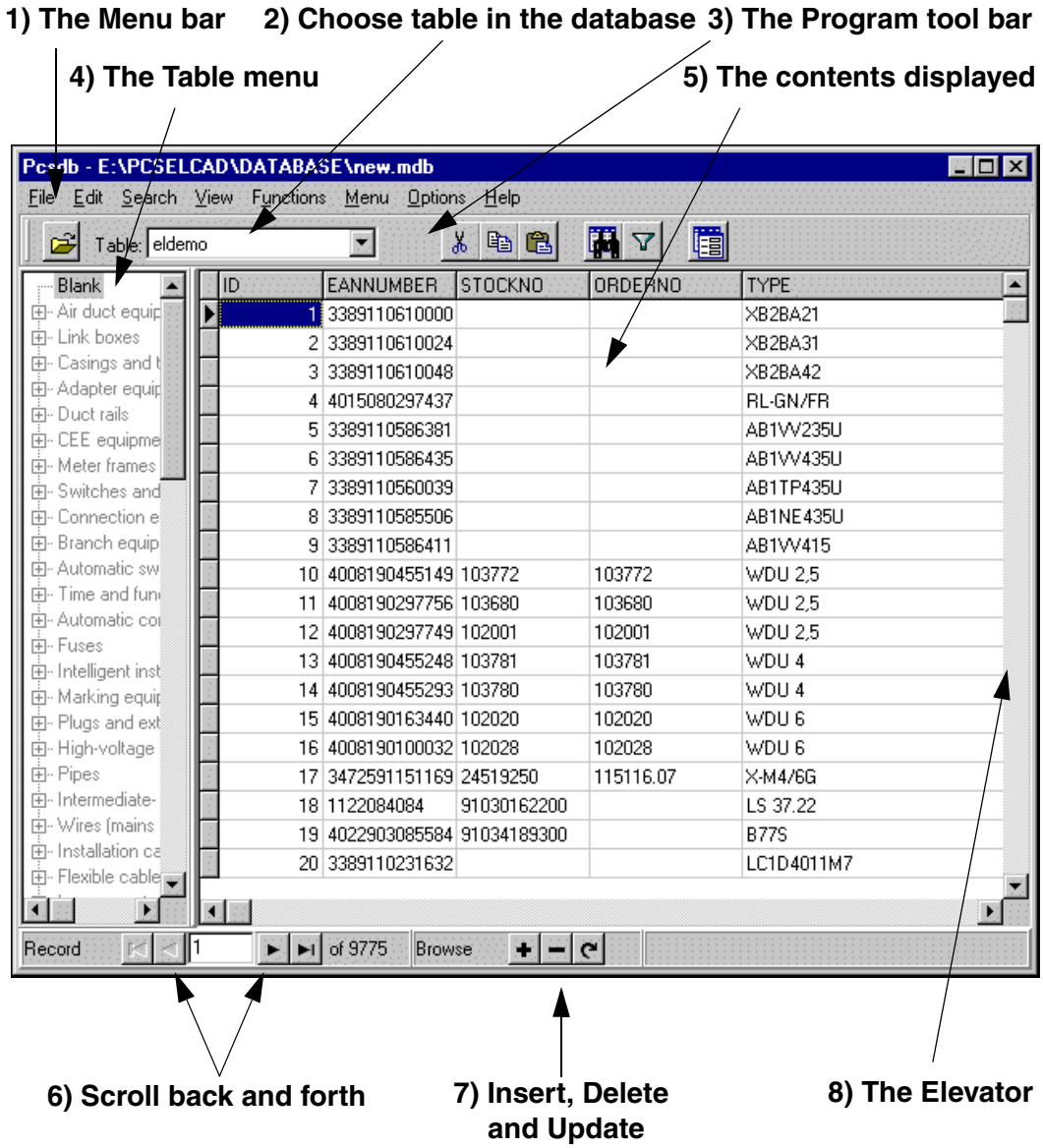
IN ORDER TO SEE THE FULL CONTENTS OF THE TABLE, CHOOSE **MENU => ACTIVE**.

— READ MORE ABOUT THIS IN "TABLE MENUS IN PC|SCHEMATIC DATABASE" ON PAGE 24.

THE WORK AREA

THIS CHAPTER DESCRIBES
THIS CHAPTER DESCRIBES THE WORK AREA OF PC|SCHEMATIC DATABASE.

When you have started **PC|SCHEMATIC Database** you enter the **Database dialog box**, that looks like this:



1. The Menu bar

The Menu bar consists of menus, which gives access to all the functions in the program. Choosing in the menu bar is either done by using the mouse, or by using the <Alt> key. **File** can for instance be chosen by pressing [Alt+f].

2. Choose table in the database

By clicking on the down-arrow in the ***Table*** field, you can choose which table in the database you wish to see/work on.

3. The Program tool bar

The Program tool bar contains program buttons. Here you can open files, cut, copy, paste, search for information, filter which information is displayed and adjust how to display the information in the Contents part of the dialog box (**5**).

4. The Table menu

The contents of a table can be divided into various menupoints, which are displayed in the Menu. When the table menu is active, you will only see the part of the table, which is belonging to the chosen menupoint. If the menu is inactive, the contents of the entire table is displayed. In that case the text in the menu will be written in grey. You enable and disable the table menu by choosing **Menu => Active**.

— Read more about the table menu in “Table menus in PC|SCHEMATIC Database” on page 24.

5. The contents

Here the contents of the chosen part of the table is displayed. Either one record is displayed at the time (record view), or several records (record view). Shift back and forth between these two views by clicking the **Record view** button.

Read more about which part of the table is displayed in “Displaying the contents of databases” on page 23.

6. Scroll back and forth

By clicking these buttons you scroll back and forth between the displayed records.
— Please read more about manoeuvring in the database in “Scrolling back and forth between the records” on page 34.

7. Insert, Delete and Update

By clicking on these buttons you determine whether to insert or delete records. You can also update the displayed database, which is important if more than one person is working on the database in a network.

— Please read more about these functions in “Editing records” on page 28.

8. The Elevator

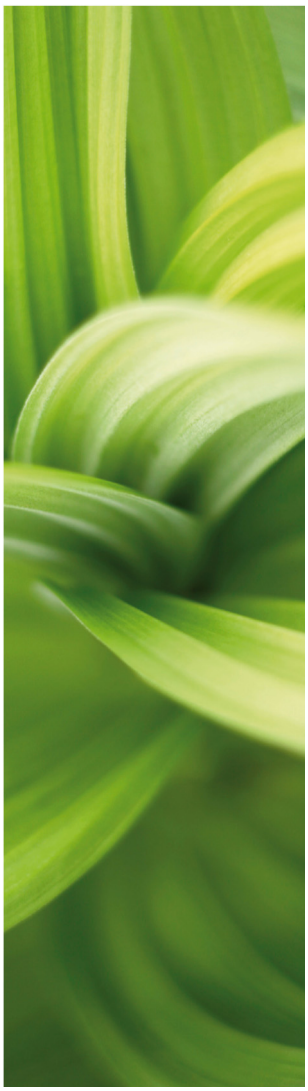
By clicking on the elevator, and keeping the mouse button pressed down, you can drag the elevator up and down. This enables you to scroll between the chosen records, if they cannot be displayed all at a time.

If you click above the elevator, you will move one page up, and if you click below the elevator you will move one page down. There is also an elevator in the lower part of the dialog box, which enables you to see other information about the components.

The table menu in the left-hand side of the screen also uses elevators.

Flat buttons

If you wish the program buttons to be displayed in the Windows98-style, you choose **Options => Flat Buttons**. To return to the Windows95-style buttons, choose **Options => Flat Buttons** again.



Basic functions

This section deals with the basic functions in PC|SCHEMATIC Database:

- Displaying the contents of a database.....(p. 23)
- The menu system.....(p. 24)
- Data filters .....(p. 26)
- Inserting, deleting and updating records.....(p. 28)
- Searching facilities .....(p. 32)
- Advanced datamanipulation and filtering with SQL queries .....(p. 35)

Displaying the contents of databases

This chapter describes

— What determines what is displayed on the screen ..... (p. 23)
— The consequences of what is displayed ..... (p. 23)
— Table viewing or Record viewing ..... (p. 24)
— Using table menus ..... (p. 24)
— Using the data filter ..... (p. 26)

What determines what is displayed on the screen?

When you see the contents of a table in a database, mainly two factors have impact on which records are displayed:

1) Whether you see the entire table, or whether you just see the contents of a menupoint in the table menu.

2) Whether you use a data filter, with which you select the components you wish to see. If you apply a filter, the table menu is set out of effect.

Finally, as an general setting, you can determine whether to display the records individually, or whether to see them gathered in a table. This is called *record viewing* and *table viewing* respectively.

The consequences of what is displayed

The fact that you can determine which components are displayed, serves two purposes in the program:

1) It is easier to locate information in the active table.

2) It determines which records you export from the table, when you choose **File => Export**.

Table or Record viewing

In the right-hand side of the **Database dialog box**, you can either see all the records at a time, or display the records individually.

By clicking the **Record view** button, you can change back and forth between these two ways of displaying the contents of the active table.

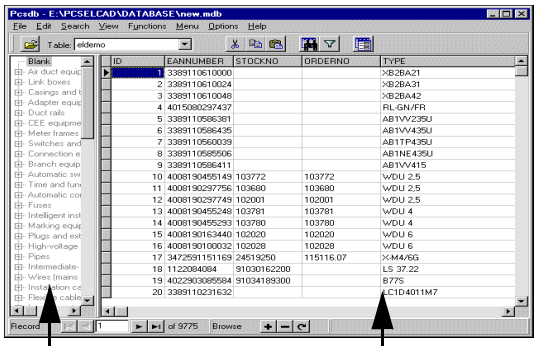
You can also change back and forth between the two views by choosing **View => RecordView** and **View => TableView** respectively.

Table menus in PC|SCHEMATIC Database

As mentioned in “On databases” on page 13, all records/components in a table can belong to a specific menupoint.

Therefore you can choose whether to display all the components in the table, or just the components that belongs to a specific menupoint.

In the **Database dialog box**, the left side is a survey of the various menu-points, to which the components can belong. This is called the ***Table menu***.



The *table menu*

Table with Components/ records

This menu can either be enabled or disabled — see “Activate the table menu” on page 24.

When it is enabled, the names of the menupoints are written in black, and when it is disabled, they are written in grey (Dimmed).

Please note, that it can be a bit hard to see the difference.

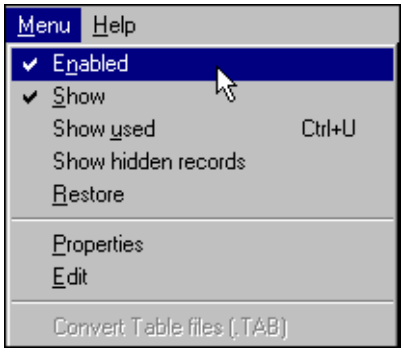
When the menu is active, you can *only* see the components, that belongs to the menupoint you have chosen in the menu.

Otherwise, you will see all the components in the table.

Activate the table menu

When you wish to activate the menu part, you have to choose **Menu => Enabled**.

When you wish to display the entire table, you choose **Menu => Enabled** again, which removes the check mark.



Right-click

When you right-click on the menupoints in the left-hand side of the dialog box, a menu pops up, in which you can activate some of the same functions.

Options using the menu

When you have activated the menu, you see all the menupoints in the menu in the left-hand side of the **Database dialog box**. If you only wish to see the menupoints, that contains components, you choose **Menu => Show used**.

To return to displaying all the menupoints, choose **Menu => Show used** once again.

SEE RECORDS THAT DOES NOT FIT THE MENU STRUCTURE

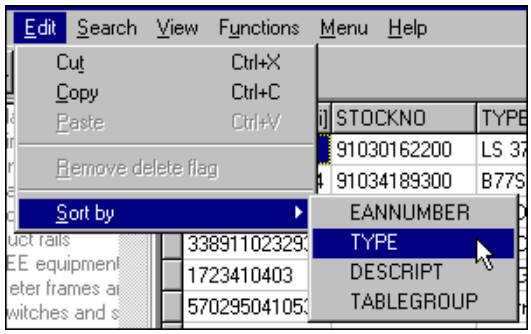
WHEN YOU WISH TO SEE RECORDS, THAT DOES NOT FIT INTO YOUR MENU STRUCTURE, CLICK ON **MENU => SHOW HIDDEN RECORDS**. THEN CLICK ON THE DISPLAYED RECORDS, AND *DRAG* THEM OVER TO THE CORRECT MENU-POINTS.

DISABLE THE FUNCTION BY CHOOSING **MENU => SHOW HIDDEN RECORDS** AGAIN.

If you choose **Menu => Restore**, the program examines all the components in the table, to check whether there are new menupoints containing components.

The choice **Menu => Properties** is elaborated in “Connecting table menus to tables” on page 59, while **Menu => Edit** is elaborated in “Creating table menus” on page 51.

SORTING BY FIELD NAME



When all the table components are displayed — that is when the menu is *not* active — you have the option of sorting the components by their field names.

To do this choose **Edit => Sort by** and click on the field name you wish the data to be sorted by.

You can only sort the components by the field names you have indexed in the table. — Read more about index fields in “Index fields” on page 50.

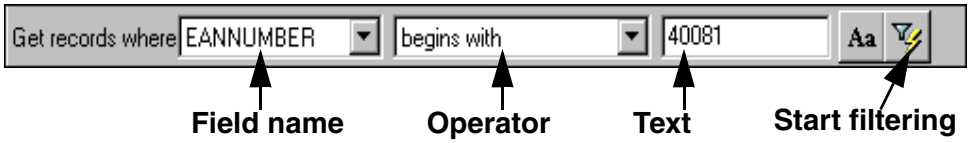
The Data filter

By clicking on the **Data filter** button — or by choosing **Functions => Data Filter** — you can set up conditions for which records in the table you wish to display.

The data filter disables the table menu

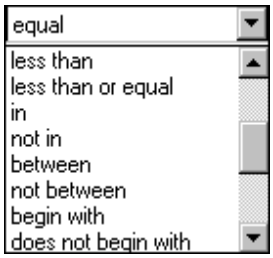
Using the data filter automatically disables the table menu. As a consequence you will always be searching in the full table when you are using the data filter.

When you have clicked the button, the following bar appears:



Here you first specify which data field you wish to use in the filtering. Then you choose an operator, and a text or a number you wish to compare the contents of the data field with.

When you click on the down-arrow in the operator field, you get a lot of different operators to choose from.



If for instance you choose the field *EANNUMBER*, set the operator to ***begins with*** and type in *40081*, you get all the records with a *EANNUMBER* beginning with *40081*.

If you wish to distinguish between upper- and lower-case letters, you click the **Aa** button, which activates the function (highlighted). You have to do this before starting the filtering.

Click the **Activate** button to carry out the filtering.

Using the Data filter to check new records

Among the operators, you will find the operator ***is repeated***. This tells you whether there exists more records having the same contents in a specific data field.

When you have typed in new records in the table, you can therefore assure yourself that you have not typed in more than one record containing information on the same component.

To do this, you have to specify a data field, that uniquely identifies the records in the table.

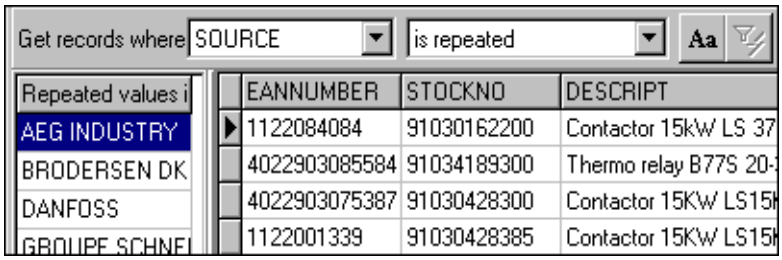
The fields STOCKNO or the field EANNUMBER does this, because no components have the same STOCKNO or EANNUMBER. (At the other hand several components can have the same MANUFACTUR).

When you have typed in new components in your table, you can therefore choose the field EANNUMBER, set the operator to ***is repeated*** and click the **Activate** button. If this gives any result, it means that you have more than one component having the same EANNUMBER in your table.

You must then correct your table, so this is no longer the case.

Separating by component vendor

If you want the components separated by component vendor, you choose the **SOURCE** field, apply the operator ***is repeated***, and click the **Activate** button.



This groups the components by component vendors. By clicking on the vendor names in the left-hand side of the dialog box, you will see which components belongs to which vendors in the right-hand side.

Disabling the data filter

When you wish to see the contents of the entire table, you just click the **Data filter** button once again, so that it is no longer highlighted.

You can also disable the filter choosing **Functions => Data Filter**.

ADVANCED FILTERING USING SQL QUERIES

IF YOU NEED TO MAKE MORE ADVANCED SELECTIONS OF DATA IN A TABLE, THIS IS POSSIBLE IF YOU KNOW OF SQL QUERIES.

READ MORE ABOUT THIS IN “SQL QUERIES” ON PAGE 35.

EDITING RECORDS

This chapter deals with how to select records or data fields, and how to insert, delete and copy records.

Furthermore, you can see how to update databases/tables working in a network.

THIS CHAPTER DESCRIBES

— INSERTING, DELETING AND UPDATING.....(P. 28)
— MOVING RECORDS TO A NEW MENUPOINT.....(P. 31)

If you are not working in a table in a database for the moment, please open for instance the database **eldemo.mdb** and select the table **Eldemo**, as described in “Opening a database” on page 16.

If you need a survey of the **Database dialog box**, please refer to “The work area” on page 18.

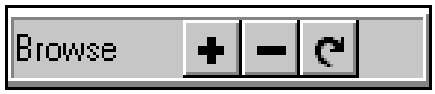
INSERTING, DELETING AND UPDATING

NO OPTION TO UNDO CHANGES

WHEN YOU EDIT RECORDS IN A TABLE, YOU MAKE THESE CHANGES IN THE TABLE *DIRECTLY*.

YOU HAVE NO OPTION TO UNDO THE CHANGES YOU MAKE.

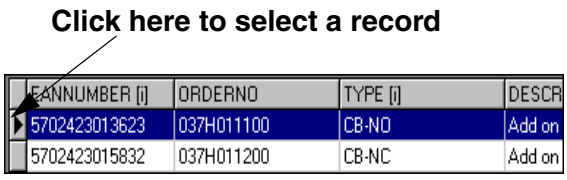
In the middle of the downmost line of the dialog box — also called the status line — you will find some buttons, which gives you the option to insert or delete records in the table. You can also update the table (which can be necessary working in a network).



Selecting records or data fields

To select a data field, you just click on the data field once. If you wish to edit the contents of the data field, you click on the data field once more, and type in the new contents of the data field. (You have to click twice, with a short interval between the two clicks — which means you shall *not* double-click).

When you wish to select an entire record, you click on the tiny field on the left-hand side of the record.



Deleting records

If you wish to delete a record, you first select the record and then click the **Delete** button. Then you will be asked whether you want to delete the record, and click **Yes**.

You can also delete a record by selecting it — or just a data field inside the record — and then press [Ctrl+Del].

If you wish to delete the contents of a data field, you have to select the data field (by clicking twice) and then press [Del].

Records only delete-marked in dBase

When you delete a record in dBase, this causes a delete flag to be set for the record, but the record is not completely removed from the table.

If you wish to remove the delete-mark from a record, select the record — or just a data field in the record — and click on the **Remove delete flag** button, or choose **Edit => Remove delete flag**. You can only remove the delete flag for an entire record.

Inserting new records

When you wish to insert a new record, click on the **Insert new record** button, or press <Ins>. This inserts a new empty record, in which you can fill in information. Click on the first data field, and fill out the data field. Then click on the succeeding data field, and fill it out as well. When you wish to jump to the succeeding data field in the record, you can also press the <Tab> button. Pressing <Shift+Tab> brings you to the preceding data field.

In the **PC|SCHEMATIC Automation** manual you can see how to attach electrical and mechanical symbols to a record.

Copying data fields or records

By selecting a record and clicking the **Copy** button, you can copy the contents of an entire record. In stead of clicking the Copy button, you can also press [Ctrl+c].

If you then select another record and click the **Paste** button, the contents of the selected record is replaced by the contents of the copied record. In stead of clicking the Insert button, you can also press [Ctrl+v].

If you only select a *data field* in another record and press the **Paste** button, only the data field is replaced by the corresponding data field in the copied record.

Copying records in Record view

If you are working in record view — and therefore only see one record at the time — you copy the entire record by pressing down <Ctrl> while clicking the **Copy** button.

Applying the Cut button

When you have selected field in a record and click the **Cut** button, the contents of the field is deleted. When you then click on another field and click on the **Paste** button, the contents of the selected field is replaced by the contents of the field you just removed. In stead of using the Cut button, you can press [Ctrl+x].

Updating

If you are working on a table in a network, it may occur that others are working on the table at the same time as you are. Therefore, they may have changed some of the information you see for the moment.

To see an updated version of the table, click on the **Update** button.

NO OTHER USER HAS ACCESS TO THE RECORD YOU EDIT

WHEN EDITING A DATA FIELD IN A RECORD, THE ENTIRE RECORD IS UPDATED AUTOMATICALLY. AT THE SAME TIME THE RECORD IS LOCKED, AND NO OTHER USER HAS ACCESS TO THE RECORD UNTIL YOU HAVE FINISHED EDITING IT. — BUT THEY ARE FREE TO WORK ON THE OTHER RECORDS, WHICH YOU HAVE NOT SELECTED.

WHEN YOU LEAVE A RECORD, THE CHANGES YOU HAVE PERFORMED, ARE AUTOMATICALLY UPDATED IN THE TABLE.

DELETED RECORDS

WHEN YOU CHOOSE **VIEW => DELETED RECORDS**, YOU WILL ALSO SEE THE DELETED RECORDS IN THE TABLE. THIS IS BECAUSE WHEN A RECORD IS DELETED, IT IS NOT REMOVED COMPLETELY, BUT A DELETE FLAG IS SET FOR THE RECORD. IN THIS WAY YOU CAN FOR INSTANCE INDICATE THAT A COMPONENT IS EXPIRED, AND WAIT TO REMOVE IT COMPLETELY UNTIL IT IS NO LONGER ON STOCK, AND NO LONGER IS USED IN EXISTING PROJECTS.

IF A COMPONENT IS NO LONGER BEING USED, YOU CAN NOTICE IT IN A **REMARKS** DATA FIELD FOR THE COMPONENT. — OR, YOU CAN CREATE A SPECIAL MENUPOINT IN THE TABLE MENU FOR EXPIRED COMPONENTS. MORE ABOUT THIS LATER.

DELETED RECORDS ARE DISPLAYED WITH STRIKETHROUG TEXT ON THE SCREEN. AS A DEFAULT, THE PROGRAM IS SET TO DISPLAY DELETED RECORDS, BECAUSE THESE CAN BE USED IN OLDER PROJECTS.

IF YOU WISH TO COMPLETELY REMOVE ALL DELETE MARKED RECORDS FROM THE TABLE, YOU MUST CHOOSE **FILE => PACK**. PLEASE NOTE, THAT THIS REMOVES *ALL* DELETE MARKED RECORDS FROM THE TABLE. REMOVING DELETE-MARKED RECORDS FROM THE TABLE INDIVIDUALLY IS NOT AN OPTION.

Moving records to a new menupoint

When you wish to move records from one menupoint to another, you can right-click on the record you want to move, and choose **Move record** in the menu that pops up.

Then right-click on the menupoint you want to transfer the record to, and choose **Insert record**.

You can also move a record by clicking on it and dragging it to the desired menupoint.

The record is now transferred from the one menupoint to the other, and the data field specifying the menulink of the record is automatically changed. — Please refer to “Menulink” on page 56 for further details about menulinks in the table.

Records outside menupoints

If there exists records without a filled out menulink data field, they are placed in the menupoint ***Blank*** first in the table menu.

Records, containing a menulink data field, which does not correspond to any menupoint in the menu structure, is displayed when you choose **Menu => Show hidden records**.

You can now transfer the displayed records to the relevant menupoints, as described above.

Disable displaying the hidden records by choosing **Menu => Display hidden records** once again.

SEARCHING IN DATABASES

In **PC|SCHEMATIC Database** you can find and replace the contents of the individual data fields in the tables in the database. You also have the option of scrolling back and forth between the components in the table, as well as specifying which record number you want to jump to.

This chapter describes how to apply these facilities.

THIS CHAPTER DESCRIBES

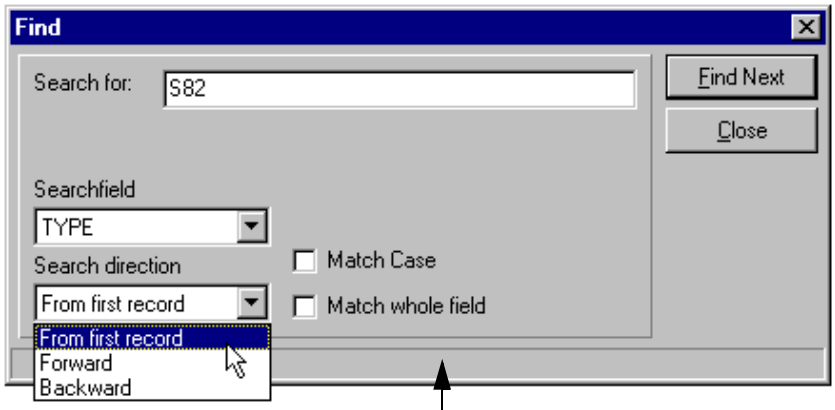
— SEARCHING ON DATA FIELDS .....(P. 32)
— FIND AND REPLACE .....(P. 33)
— SCROLLING BACK AND FORTH BETWEEN THE RECORDS .....(P. 34)

Also, you have the option of filtering which parts of the table you wish to see. Please refer to “The Data filter” on page 26 for further details.

SEARCHING ON DATA FIELDS

When you wish to locate a component in the table, you can click the **Find** button. Or you can choose **Search => Find** or press [Ctrl+f].

You then enter this dialog box, where you type in the text you are looking for.



Here you type in the text in the upper-left corner, and by clicking the down-arrow below ***Searchfield***, you specify for which data field the search should be carried out.

By clicking the down-arrow below ***Search direction***, you determine whether the program should search ***Forward*** or ***Backward*** compared to where you are located in the table for the moment. Or the table could be examined ***From first record***.

If you enable ***Match Case***, the program will only find the data fields containing the text ***S82*** in the example above. If it is not enabled, data fields containing the text ***S82*** as well as the text ***s82*** will be found.

If you enable ***Match whole field***, the program only finds the fields matching the exact text you have typed in. In the dialog box above, it would then locate the data fields where it *only* says ***S82***.

When you click **OK** in the dialog box above, the program will select the first component having a *type* data field containing the text ***S82***.

When you press [F3] — or choose **Search => Find next** — the program will locate the next component having a *type* data field containing the text ***S82***.

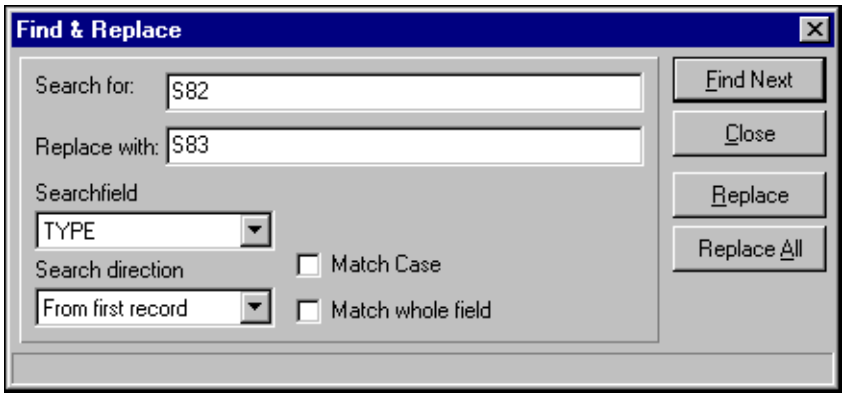
SEARCH ONLY PERFORMED AMONG THE DISPLAYED COMPONENTS

YOU ONLY SEARCH AMONG THE COMPONENTS DISPLAYED ON THE SCREEN FOR THE MOMENT.

If you are applying the table menu or the data filter function, you have to disable these to be able to search in the entire table. — To see how to disable these two functions, please refer to “Table menus in PC|SCHEMATIC Database” on page 24 and “The Data filter” on page 26 respectively.

FIND AND REPLACE

If you wish to replace the text you find by another text, you should instead choose **Search => Replace** or press [Ctrl+r]. You then enter this dialog box:



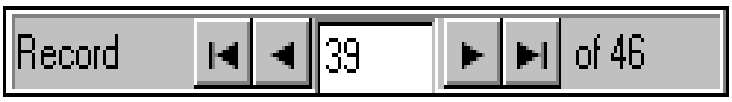
The dialog box function as described above, having the extra facility that you in the ***Replace by*** field type in the text you want the original text to be substituted by.

In order to find the next data field containing the text ***S82***, you click on **Find next**. When you have found a data field fulfilling the specifications, you click on **Replace** if you wish to replace the text. you will then be asked to confirm this, and you click **Yes**.

If you wish to replace every occurrence of the text ***S82*** with the text ***S83***, you click **Replace all**. You will then be asked whether you wish to replace all occurrences of ***S82***, or you just wish to replace the text you have found for the moment. Click **Replace all**.

Scrolling back and forth between the records

In the lower-left corner of the dialog box, some buttons are located, which you use for scrolling back and forth between the components in the table.



By clicking the **Next** and **Previous** buttons, you can scroll back and forth among the selected components.

By clicking the **Top** and **Bottom** buttons, you jump to the first and last component respectively.

By clicking in the field, you can type in which record number you want to locate. When you press <Enter> the program jumps to the desired record.

By choosing **Search => Go to record no.**, the cursor place itself in the field, where you can specify which record number you want to jump to.

SQL QUERIES

This chapter describes how to execute SQL queries in **PC|SCHEMATIC Database**.

THIS CHAPTER DESCRIBES

— OPTIONS WHEN USING SQL ..... (P. 35)
— EXECUTING SQL QUERIES ..... (P. 36)

WHAT IS SQL?

SQL — STRUCTURED QUERY LANGUAGE — IS A COMMONLY USED LANGUAGE FOR EXECUTING QUERIES AND MANIPULATING DATA IN DATABASES.

THE SYNTAX FOR SQL QUERIES IS NOT DESCRIBED IN THIS MANUAL. PLEASE REFER TO THE SQL LITERATURE.

OPTIONS WHEN USING SQL

In **PC|SCHEMATIC Database** you can execute SQL queries on the database, which is open for the moment.

Hereby you can for instance create — or delete — tables.

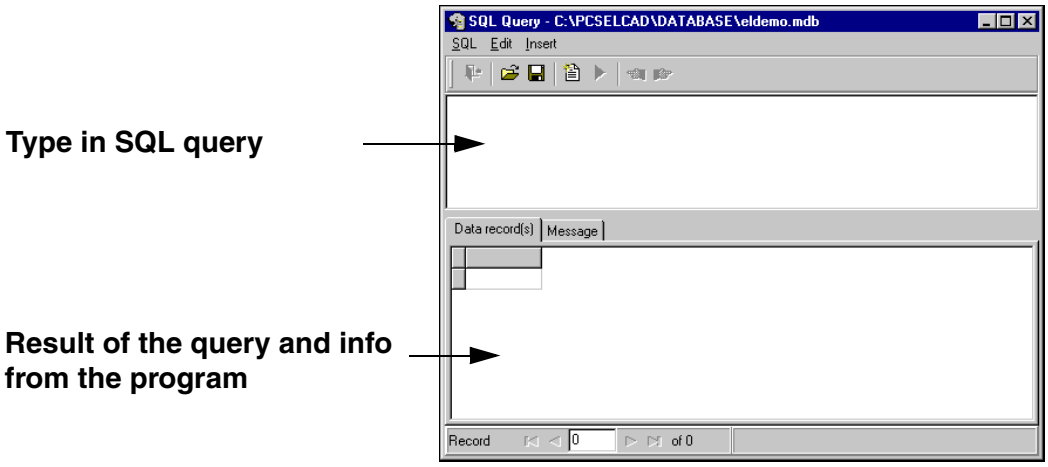
You can also execute compound queries, which are not covered by the data filter function in **PC|SCHEMATIC Database**, as described in “The Data filter” on page 26.

ONLY USE SQL IF YOU KNOW WHAT YOU ARE DOING

BECAUSE SQL QUERIES CAN HAVE DRAMATIC CONSEQUENCES — SUCH AS DELETING ENTIRE DATABASES WITH NO UNDO — YOU SHOULD ONLY USE THIS OPTION IF YOU HAVE WORKING KNOWLEDGE OF SQL QUERIES.

Executing SQL queries

To execute an SQL query in **PC|SCHEMATIC Database** choose **Functions => SQL query**, and you enter this dialog box:



Executing a SQL query

Here you type in an SQL query in the upper part of the dialog box, and press the **Execute SQL** button. The result of the query is displayed in the lower part of the dialog box on the tab **Data record(s)**. Information from the program is displayed on the tab **Message**.

If you need the name of a table in a query, choose **Insert => Tablename** and choose one of the suggested table names.

Click on the **New SQL** button to execute a new SQL query.

When you have selected the information you need, click on the **Exit and Return SQL** button. This closes the dialog box, and the selected records are displayed in the database dialog box. You can then for instance export these records, as described in “Exporting comma separated files” on page 83.

Saving or opening an SQL query

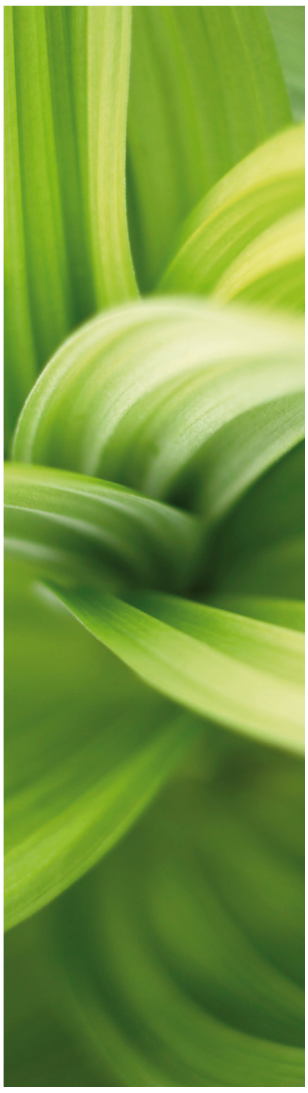
If you wish to save an SQL query, click on the **Save SQL file** button and save the query.

If you wish to open a saved SQL query, click on the **Open SQL file** button, and open the saved SQL query.

Quote tablenames

Under **Options => Quote Tablenames** in the **Database dialog box** you can specify whether to quote tablenames in SQL queries.

The applied driver sets this setting correct. It should therefore only be changed if problems occurs in SQL queries as a consequence of misinterpretations of tablenames.



Maintaining databases

This section deals with:

- Creating a new database ..... (p. 39)
- Creating a new table ..... (p. 42)
- Editing the data fields in a table ..... (p. 48)
- Creating and editing a table menu..... (p. 51)
- Connecting a menu to a table ..... (p. 59)

Creating Microsoft Access databases

This section is a description of how to create a Microsoft Access database in **PC|SCHEMATIC Database**.

It is *not* described how to create the *tables* of the database, but only how you create a database, in which you can place tables in *later*. — Read more about database and tables in “On databases” on page 13.

This chapter describes
— Creating a new Microsoft Access database....(p. 39) — Copying an existing database....(p. 41) — Databases and the dBase format....(p. 41)

— How to create *tables* is described in “Creating tables” on page 42. How to edit the properties of a table is described in “Editing table properties” on page 48.

Installation of Microsoft Access not necessary

In **PC|SCHEMATIC Database** you can create and maintain Microsoft Access databases without having installed Microsoft Access on your PC.

Creating a new Microsoft Access database

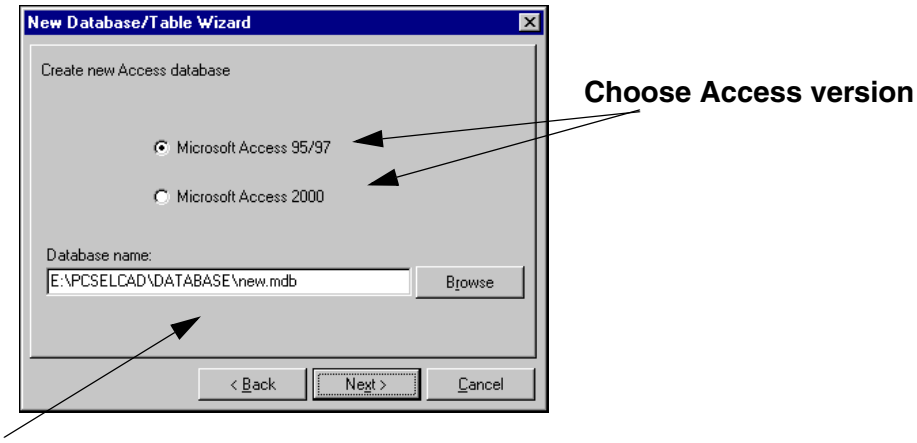
To create a new Microsoft Access database choose **File => New**:

Here you get the following options:

Option	Description of the option
Microsoft Access Database (mdb file)	Described in the following.
Microsoft Access Table	Described in “Creating tables” on page 42.

Option	Description of the option
Microsoft Data Link (udl file)	Described in "MDAC and UDL files" on page 99 and "Creating an UDL file" on page 100.
dBase Table (dbf file)	Described in "Creating tables" on page 42.

Because you are about to create a new Microsoft Access database, choose ***Microsoft Access Database (mdb file)*** and click **Next**:



Now choose in which Microsoft Access version you wish to create the database, and type in the name of the database. Click **Next**.

Names for databases, tables and data fields

A good policy for naming databases, tables and data fields is only to use the following characters:

1) The letters from A to Z

2) The digits from 0 to 9

In other words, you should not use any special characters or special letters from alphabets other than the English.

The database is now created, but contains no tables.

You get the following message:

Click on **Yes** to start creating a new table in the database now, or click **No**.

If you click **Yes**, you can create a table as described in “Creating tables” on page 42.

Copying a table from an existing database and adapting its contents

If you wish to create a database including tables quickly by copying from an existing database, you can do it like this:

1) Create a database — as described in “Creating a new Microsoft Access database” on page 39

2) Make a copy of an existing table — as described in “Creating tables” on page 42 (see “Copying a table” on page 45 in particular)

3) Adapt the properties of the table — the names and lengths of its fields — as described in “Editing table properties” on page 48

4) Create a menu — as described in “Creating table menus” on page 51

5) Attach the menu to the table — as described in “Connecting table menus to tables” on page 59

Please note that you may not need to execute all the steps mentioned above, when creating your own database:

- If you are pleased with the structure of the copied database, there is no reason executing step 3
- If you are pleased with the menu attached to the table, you can copy the menu (like you copy any other table) in stead of executing step 4

However, it might be a good idea to carry out all the steps to get familiar with the database program.

Databases and the dBase format

Microsoft Access works with database in the traditional way, where a database file is created containing information about which tables etc. are included in the database. — Read more about this in “On databases” on page 13.

As opposed to this, dBase only works with tables. When working with a table in dBase, **PC|SCHEMATIC Database** understands the *folder* in which the table is placed as a database.

If other dBase tables exists in this folder, they can be choosen from the ***Table*** window in the upper left-hand corner of the program.

CREATING TABLES

This chapter describes how to create tables in **PC|SCHEMATIC Database**.

THIS CHAPTER DESCRIBES

— CREATING A NEW MICROSOFT ACCESS TABLE ..... (P. 42)
— CREATING A NEW DBASE TABLE ..... (P. 47)
— DELETING A DATABASE ..... (P. 47)
— EMPTYING A TABLE ..... (P. 47)

— See how to create Microsoft Access *databases* in “Creating Microsoft Access databases” on page 39.

INSTALLATION OF MICROSOFT ACCESS NOT NECESSARY

IN **PC|SCHEMATIC Database** YOU CAN CREATE AND MAINTAIN MICROSOFT ACCESS AND DBASE TABLES WITHOUT HAVING INSTALLED EITHER DBASE OR MICROSOFT ACCESS.

CREATING A NEW MICROSOFT ACCESS TABLE

To create a new table in the Microsoft Access database you have opened for the moment, choose **File => New**. Then choose ***Microsoft Access Table*** and click **Next**.

You then enter this dialog box:

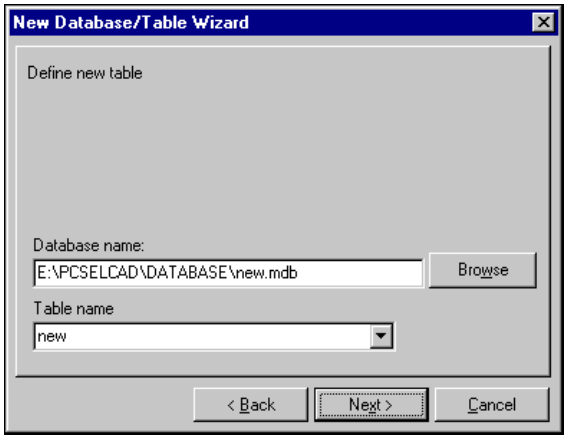
You then get the following options:

Option	Description of option
Define New Table	Here you create a table from scratch. No fields are defined in advance. However, in Microsoft Access tables an ID field is created as a primary key. — See "Defining a new table" on page 43.
Copy Table Structure	Here you copy a table structure from an existing table. Thus you get a complete table, where the fieldnames are created, but the table contains no records. — See "Copying a table structure" on page 44.
Copy Table	Here you copy of a complete table in a database. You get an exact copy of the table — including all records. — See "Copying a table" on page 45.

These options are described in the following.

1. Defining a new table

To create a new table from scratch, choose ***Define New Table*** and click **Next**. You then enter the following dialog box:

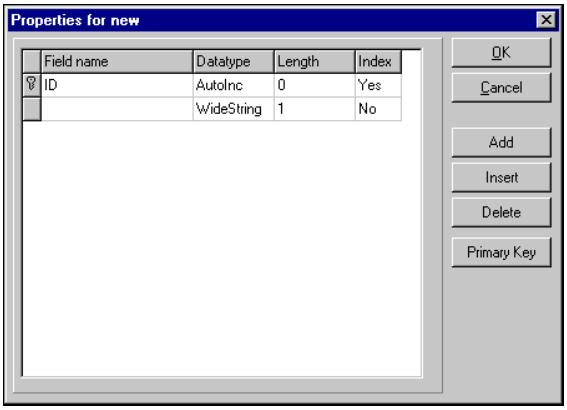


Specify the name of the database — if a database is opened this field is filled out automatically — and type in the name of the table.

— See "Names for databases, tables and data fields" on page 40.

Click **Next**.

You now enter the following dialog box:



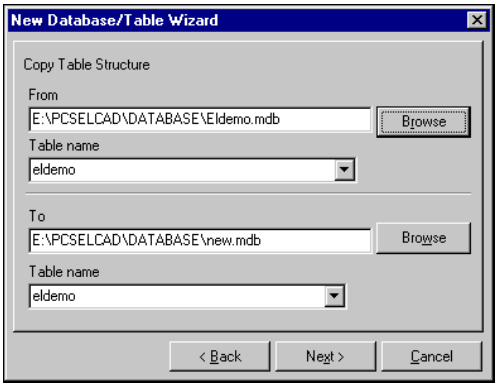
Here you can create the necessary field names in the table. Read more about this in “Editing table properties” on page 48.

2. Copying a table structure

To copy the table structure from an existing table select ***Copy Table Structure*** and click **Next**.

Copy the structure from this table →

Name on the new table →



Here you specify which table you wish to copy — see “Choosing which table to copy” on page 45 — and type in the name of the new table.

Doing this you get a complete table where the field names are created, but the table contains no records.

CHECK THAT THE TABLE HAS A PRIMARY KEY

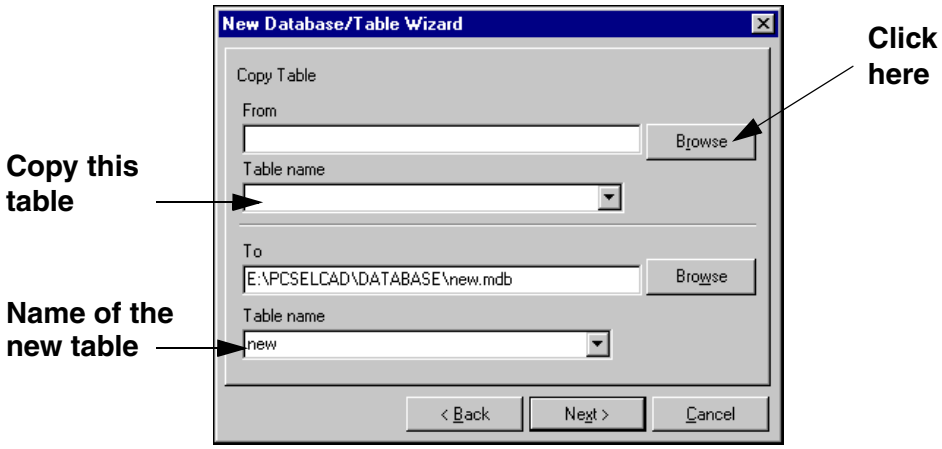
REMEMBER TO CHECK THAT THE TABLE HAS A PRIMARY KEY. IF THIS IS NOT THE CASE, YOU HAVE TO CREATE ONE YOURSELF. (IF YOU ARE CREATING A DBASE TABLE YOU CAN NOT CREATE A PRIMARY KEY).

— READ MORE ABOUT THIS IN “EDITING TABLE PROPERTIES” ON PAGE 48.

3. Copying a table

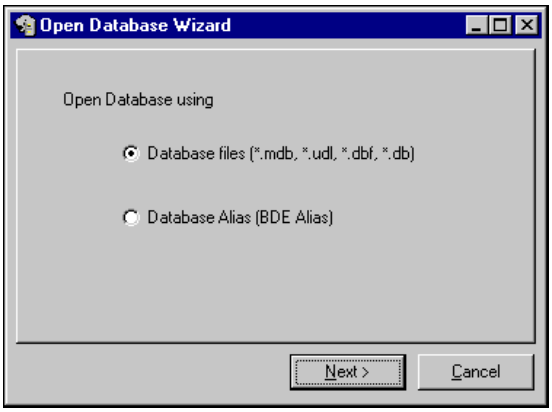
To copy a table including all of its records (here the table **EIdemo** from the database carrying the same name) you choose ***Copy Table*** and click on **Next**.

You then enter this dialog box where you can select which table you wish to copy, and type in the name of the new table.



Choosing which table to copy

Click on **Browse** to choose the table you wish to copy:

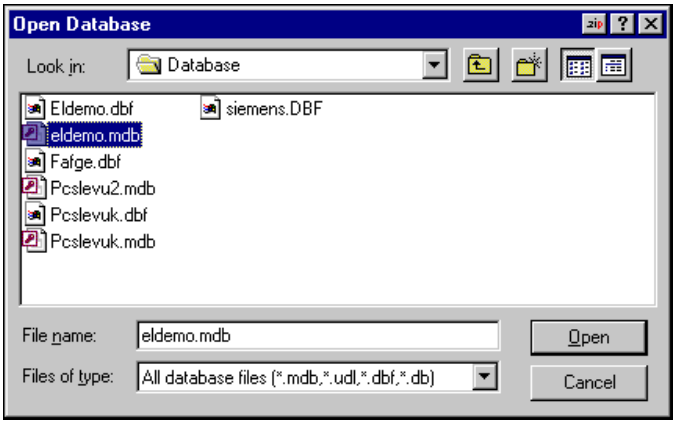


You now have to choose how to open the database in which the table is created.

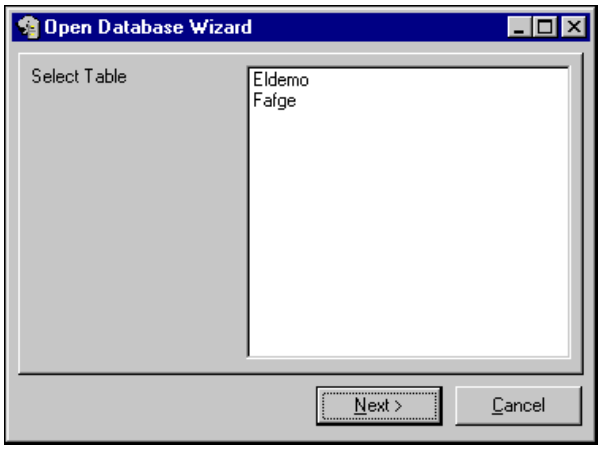
COPYING TABLES FROM OTHER DATABASE FORMATS

EVEN THOUGH YOU ARE CREATING A MICROSOFT ACCESS TABLE, YOU CAN CHOOSE TO COPY A TABLE FROM ANOTHER DATABASE FORMAT. FOR INSTANCE YOU CAN COPY A DBASE TABLE, OR A DATABASE OF A FORMAT OPENED VIA AN UDL FILE OR A BDE ALIAS.

Click on ***Database files (*.mdb, *.udl, *.dbf, *.db)*** and click on **Next**.

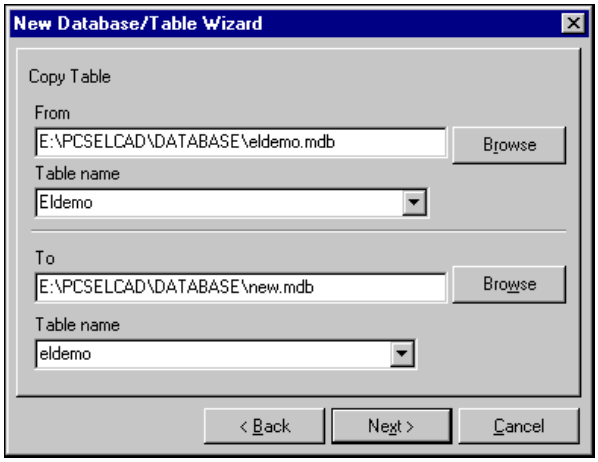


Click on the *database* **eldemo.mdb** and click on **Open**.



Click on the *table* **Eldemo** and click **Next**.

The previously displayed dialog box has now been filled out:



Click on **Next** to execute the copying.

When the copying is completed, the new table in the database is opened automatically.

CHECK THAT THE TABLE HAS A PRIMARY KEY

REMEMBER TO CHECK THAT THE TABLE HAS A PRIMARY KEY. IF THIS IS NOT THE CASE, YOU HAVE TO CREATE ONE YOURSELF. (IF YOU ARE CREATING A DBASE TABLE YOU CAN NOT CREATE A PRIMARY KEY).

— READ MORE ABOUT THIS IN "EDITING THE PROPERTIES OF A TABLE" ON PAGE 48.

CREATING A NEW DBASE TABLE

To create a new dBase table choose **File => New**. Here you choose ***dBase table (dbf file)*** and click on **Next**.

You now get the option of specifying how to create your new dBase table, which takes place exactly as described for Microsoft Access tables. — See how to do this in "Creating a new Microsoft Access table" on page 42.

If for instance you choose to copy a Microsoft Access table, you can have a new dBase table created from a Microsoft Access table.

— Please also read "Databases and the dBase format" on page 41.

DELETING TABLES

To delete the currently active table, do the following:

1) Press down <Shift> and choose **File => Delete**

2) Answer **Yes** to delete the table

3) Press <F5>, and the table disappears from the ***Table*** field in the toolbar

The table is now deleted.

EMPTYING TABLES

To delete all records from the currently active table, do the following:

1) Press down <Shift> and choose **File => Empty**

2) Answer **Yes** to empty the table

All records are now deleted from the table.

EDITING TABLE PROPERTIES

THIS CHAPTER DESCRIBES

— EDITING THE PROPERTIES OF A TABLE.....(P. 48)
— PRIMARY KEYS.....(P. 50)
— INDEX FIELDS.....(P. 50)

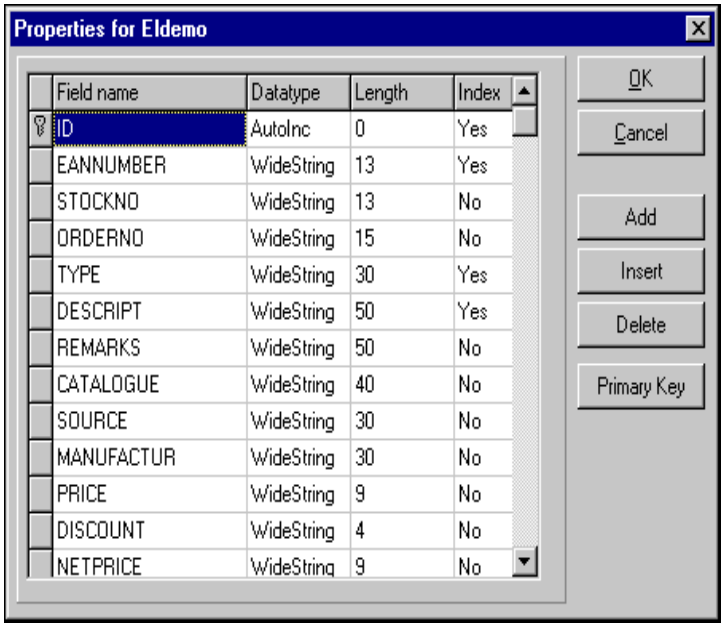
NEWLY CREATED MICROSOFT ACCESS TABLE

IF YOU HAVE CREATED A NEW MICROSOFT ACCESS TABLE, THERE HAS ONLY BEEN CREATED AN ID FIELD FOR THE TABLE.

YOU CAN NOW CREATE THE OTHER FIELDS IN THE TABLE, AS DESCRIBED BELOW.

EDITING THE PROPERTIES OF A TABLE

If you wish to edit some of the data fields in the active table, choose **File => Properties** and you enter this dialog box:



Here you can add or remove data fields, and change the length of the data fields. Finally, you determine which data fields shall be index fields in the data fields — which means that searching on these data fields will be speeded up if you are working in dBase.

DENIED ACCESS TO EDIT TABLE PROPERTIES

IF SOMEBODY ELSE IS USING A TABLE YOU WISH TO *RECONSTRUCT*, YOU WILL BE DENIED ACCESS TO DO SO.

IF YOU ARE WORKING IN ***PCSCHEMATIC*** AND WISH TO EDIT THE PROPERTIES OF A TABLE (AND NOT JUST ITS CONTENTS), YOU THEREFORE HAVE TO CHOOSE **SETTINGS => DATABASE** AND CLICK ON ***DISABLED***. OTHERWISE YOU YOURSELF WILL BE BLOCKING YOUR OWN ACCESS TO THE TABLE.

Adding/inserting a data field in a table

When you wish to add a data field to the end of the table, click on **Add**.

Type in the name of the data field below ***Field name***, and press <Enter> — see “Names for databases, tables and data fields” on page 40.

Under ***Datatype*** choose *WideString* — unless it is a primary key ID field in Microsoft Access. If this is the case choose *AutoInc*. — See “Primary keys” on page 50.

Datatypes and primary keys in dBase

In dBase all ***Datatypes*** are of the type *String*.

Also, no **Primary key** button exists for dBase tables, because primary keys can not be created in dBase.

Then click in the ***Length*** field of the new data field, type in the length of the data field, and press <Enter>.

— Please note, that the program does not memorize what you type in until you press <Enter>.

If you wish to place a new data field in the middle of the table, click on the data field, which is located where you want the new data field to be located. Then click on **Insert**, and fill out ***Field name*** and ***Length*** as described above.

Deleting a data field in the table

If you wish to delete a data field from the table, select the data field, and click on **Delete**.

Updating the changes

The changes in the properties of the data fields in the table is not carried out until you click **OK**. When you have deleted a data field from the table, you will be asked to confirm the deletion when you click **OK**.

Click on **Cancel** if you regret any of the changes you have carried out.

Primary keys

Please note that primary keys does not exist in the dBase format.

What is a primary key?

A primary key is a field in a table, which uniquely identifies the individual records in the table. If a field is pointed out to be a primary key, **PC|SCHEMATIC Database** will not allow the creation of two records with the same contents in the primary key field.

If you wish to turn a data field in a table into a primary key, choose **File => Properties**, click on the field, and then click on **Primary key**. When a field is a primary key, a small key is displayed in front of the field in the **Properties** dialog box.

When you create a Microsoft Access table in **PC|SCHEMATIC Database**, a field with the name ID will be created automatically. This field will be set to be primary key.

Primary key for Microsoft Access table created from dBase table

When you create a Microsoft Access table from a dBase table, you must add an ID data field (or another data field name) of the type *AutoInc* and set it to be primary key. The data field should be placed as the first data field in the table.

It is important to create a primary key, in order not to have double records in the table.

Index fields

When you want a data field to be an index field, you click on its ***Index*** field, and choose **Yes** in the appearing menu.

Index fields speeds up searching in dBase

If you turn a data field into an index field, searching on the data field will be increased considerably. Furthermore, you got the option to sort the database by its index fields. *But* at the same time the index file of the database will take up more space on your hard disk. If you are not careful, this index file could easily take up more space than the database itself. So, make sure only to create the index fields that are really necessary.

CREATING TABLE MENUS

In **PC|SCHEMATIC Database** you have the option of designing your own table menus.

THIS CHAPTER DESCRIBES

— CREATING A NEW TABLE MENU.....(P. 48)
— ON EDITING TABLE MENUS .....(P. 53)
— THE TABLE MENU EDITOR .....(P. 55)
— COPYING AN EXISTING TABLE MENU.....(P. 58)

When you have created a table menu, you can connect it to a table, as described in “Connecting table menus to tables” on page 59.

TABLE MENUS WERE PREVIOUSLY CALLED DATABASE MENUS

WHAT WAS CALLED A *DATABASE MENU* IN PREVIOUS VERSIONS OF **PC|SCHEMATIC Database** IS NOW CALLED A *TABLE MENU*.

THIS IS BECAUSE THE MENU IS ONLY ATTACHED TO A SINGLE *TABLE* IN THE DATABASE.

CREATING A NEW TABLE MENU

— If you just need to copy an existing table menu, please read “Copying a table menu” on page 58.

To create a new menu, do the following:

1) Choose **File => Close** if a database is open in **PC|SCHEMATIC Database**

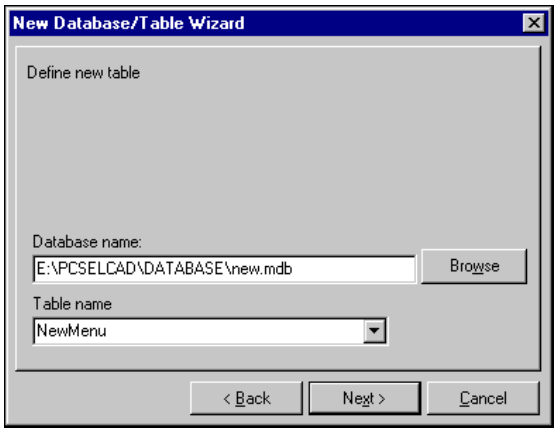
2) Choose **Menu => Edit**

3) Choose **File => New**

4) You now have to choose whether to create a table menu in Microsoft Access or dBase format:

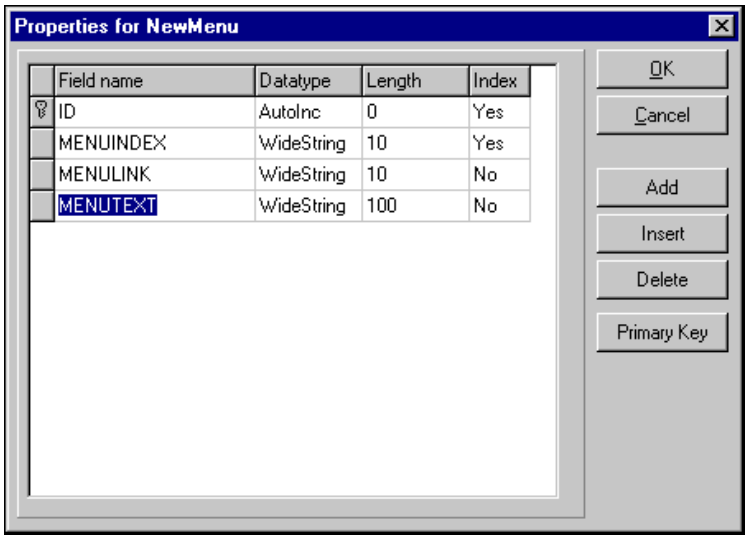
Choose ***Microsoft Access Table*** and click **Next**.

5) You must now specify in which database to save the menu, and type in the name of the menu in the field ***Table name***:



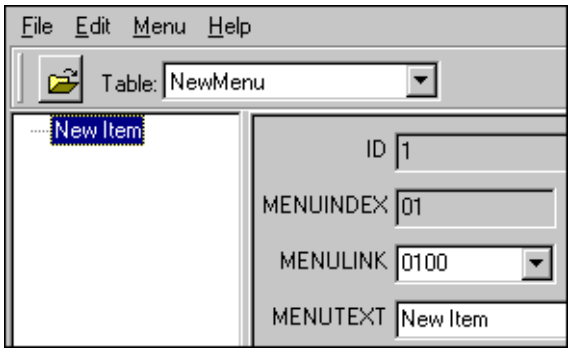
Click on **Next**.

6) You can now add and edit data fields in the table menu, which you edit as an ordinary table:



See how to do this in “Editing the properties of a table” on page 48.
Click **OK** when you have finished.

7) You now return to the **Table Menu Editor**, where you can create menupoints, as described in “The Table Menu Editor” on page 55:



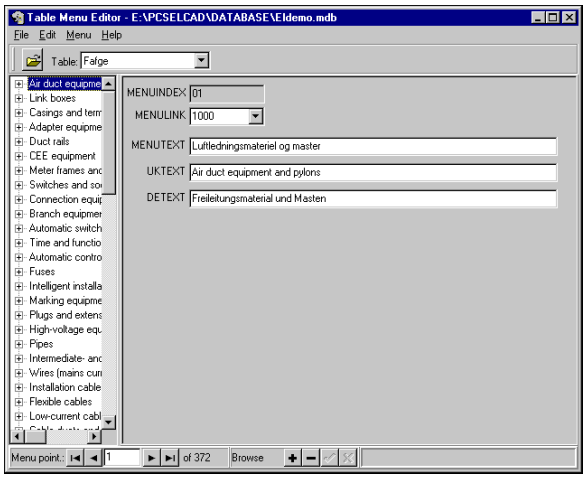
8) Choose **File => Exit** when you have finished.

You can then connect the menu to a table, as described in “Connecting table menus to tables” on page 59.

On editing table menus

In the *table* **eldemo** — in the database of the same name — a table menu is applied, but you do not necessarily have to use this menu in your database.

When you choose **Menu => Edit** you can edit the table menu, which is connected to the database you are working on for the moment.



ALWAYS EDIT TABLE MENUS FROM MENU => EDIT

THE TABLE MENU YOU CREATE HERE, ALSO IS CREATED AS A DATABASE IN THE **PC|SCHEMATIC Database** FORMAT. THEREFORE, YOU *CAN* ALSO OPEN IT AS AN ORDINARY DATABASE. HOWEVER, IT IS IMPORTANT THAT YOU DO *NOT* DO THIS. THE CONSTRUCTION OF THIS DATABASE HAVE TO FULFIL SOME SPECIFICATIONS, WHICH MAKES IT USABLE AS A MENU FOR ANOTHER DATABASE. THIS IS HANDLED AUTOMATICALLY WHEN YOU EDIT THE MENU USING **MENU => EDIT**.

No access

When the database is located in a network, it may occur that you are not allowed to edit the structure of a table menu — such as adding an extra field for a new language for the menupoints. If this is the case, it is because someone else is working in the database, and you are therefore not allowed to edit it.

In situations where you have started **PC|SCHEMATIC Database** from **PC|SCHEMATIC Automation** having the database connected, you can therefore block for your own access.

To disconnect the database, you can either close down **PC|SCHEMATIC Automation**, or enter **PC|SCHEMATIC Automation**, choose **Settings => Database**, click on ***Off***, and then click **OK**.

Databases and table menus

Because more than one database can use the same table menu, you have to look out for the following:

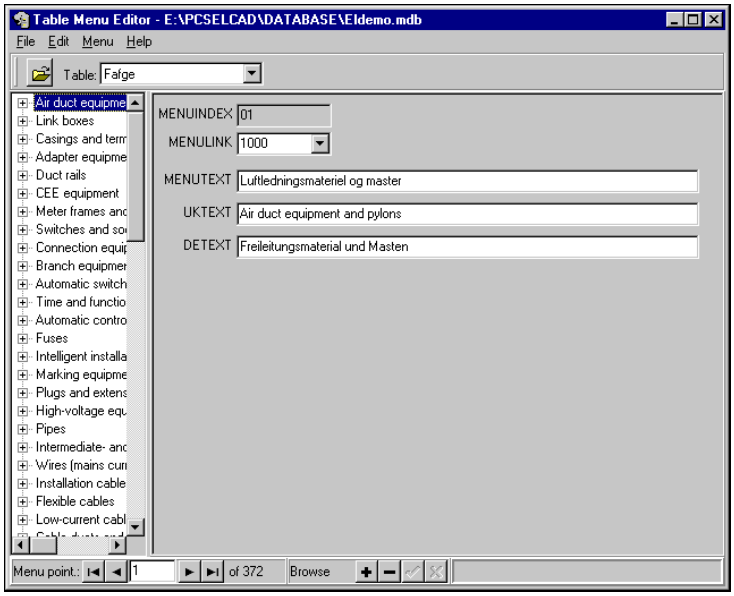
1) Making changes in a table menu for a database, also has consequences for other databases using the same table menu.

2) In **PC|SCHEMATIC Automation**: The changes are first updated when **PC|SCHEMATIC Automation** is restarted, or when you inactivate the menu and activate it again.

3) In **PC|SCHEMATIC Database**: You have to press <F5> to see the changes.

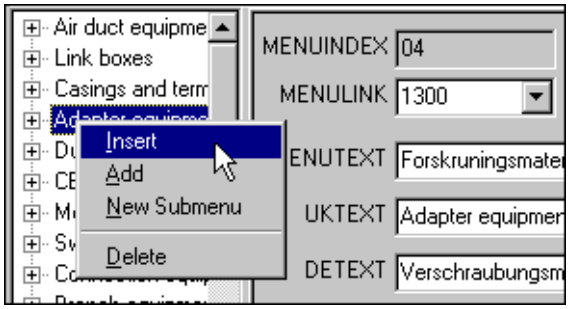
The Table Menu Editor

When you have chosen **Menu => Edit** you enter the **Table Menu Editor**:



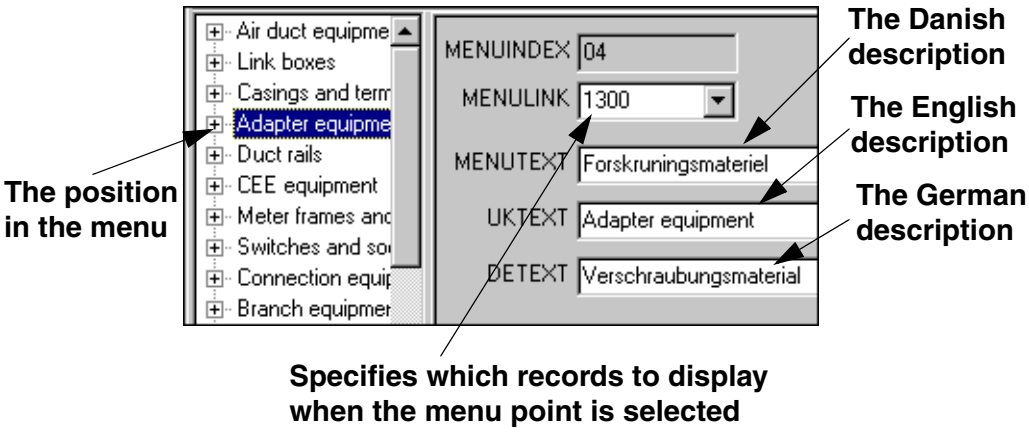
If you right-click on a menupoint in the dialog box, a menu pops up, giving you the option of deleting, inserting or adding new menu points in the menu — as well as adding submenus.

You will get the same menu clicking on **Edit** in the menu tool bar.



Edit menupoint

When you wish to edit an entry in the table menu, you click on the menu point in the left-hand side of the dialog box, and make the desired changes in the right-hand side of the dialog box.



Here you can edit the name/description in Danish, English and German, and change its *menulink*. When you click on the down-arrow in the ***Menulink*** field, you get a list of the available menulinks for the level you are placed on in the Table Menu Editor — see “Menulink” on page 56.

Make your changes and choose **File => Exit** when finished. The changes are made directly in the table menu, and you have no options for undo.

Menulink

A menulink has two functions:

1) The menulink determines which records in a table are displayed when you choose the menupoint — see “The menu structure and menulink” on page 56

2) By using a special naming in the menulink fields, you can have the records of the submenupoints displayed, when you click on a menupoint — see “Displaying records on several levels in the menu” on page 56

MENULINKS AND THE CONTENTS OF MENUPOINTS

IN ORDER TO LET THE MENULINK SPECIFY WHICH RECORDS ARE DISPLAYED FOR A MENUPOINT, A FIELD IN EACH OF THE RECORDS IN THE COMPONENT TABLE MUST CONTAIN A NUMBER CODE, SPECIFYING WHICH MENUPOINT THIS INDIVIDUAL RECORD BELONGS TO.

WHICH FIELD THAT CONTAINS THIS NUMBER CODE IN THE DATABASE, IS SPECIFIED IN **MENU => SETTINGS**. READ MORE ABOUT THIS IN “CONNECTING TABLE MENUS TO TABLES” ON PAGE 59.

The menu structure and menulink

The menu structure for a table (the table menu) consists of menupoints, each displaying different components depending on the nature of the components. If you need further division of the components, the menupoints can also contain submenus.

The *menulink* determines which records are displayed when you click on a menupoint. If for instance a menupoint has got the menulink *1310*, all records in the component table containing the menulink *1310* will be displayed when you click on the menupoint.

A menulink must be a number containing at least four digits. You can have menupoints in up till five different levels.

MENULINKS DOES *NOT* DETERMINE THE LOCATION OF MENUPOINTS

THE MENULINK DOES *NOT* DETERMINE WHERE THE MENUPOINT IS PLACED IN THE MENU STRUCTURE. THIS IS HANDLED AS DESCRIBED IN “ADDING/INSERTING A MENUPOINT” ON PAGE 57.

Displaying records on several levels in the menu

When you click on a menupoint, the program displays all components in the component table, that has the same menulink as the menupoint.

If the menupoint contains submenus, it is also possible to have displayed all components from these.

If for instance the menupoint ***Terminals*** is a submenu to ***Branch equipment***, all the terminals from ***Terminals*** can also be displayed when you click on ***Branch equipment***.

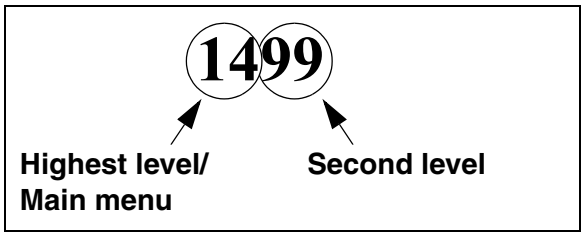
In order to make this work, you must apply number codes following a special system in the name of the menulink.

To do so, you have to know the following:

1) The first two digits in the menulink identifies it on the highest level. If the third and fourth digit are both zero, the menupoint will be considered to be on the highest level in this context.

2) If the third and fourth digit have non-zero values in a four-digit menulink, the menupoint belongs to the second level.

3) If you need menupoints on third level, you just assign a six-digit menulink to the menupoint. Accordingly you apply eight digits for a fourth level, and ten for a fifth.



When for instance the menulink for a record is 162657, the record will be displayed when you click on:

1) The menupoint with the menulink 1600 (highest level)

2) The menupoint with the menulink 1626 (second level)

3) The menupoint with the menulink 162657

You can decide for yourself whether you wish to apply the number code system described above.

Adding/Inserting a menupoint

When you wish to insert a menupoint in the menu, right-click where you want the menupoint to be inserted, and choose **Insert** in the appearing menu.

In the right-hand side of the dialog box you fill out the ***UKtext*** field, and choose ***Menulink*** by first clicking on the down-arrow in the ***Menulink*** field, and then click on one of the menulinks displayed.

If you wish to add a menupoint to the end of the menu, choose **Add** instead.

Delete menupoint

When you wish to delete a menupoint, you right-click on the menupoint, choose **Delete**, and answer **Yes** to delete the menupoint.

New submenupoint

When you wish to create a new submenupoint to a menupoint, you right-click on the menupoint, and choose **New submenu**. Here you fill out the field with ***Description***, and choose ***Menulink*** by first clicking the down-arrow in the ***Menulink*** field. This gives you all available sub-menulinks for the chosen menu.

Click on one of the menulinks displayed, and click **OK**.

Returning to the component table

When you have finished working on the table menu, and have saved your changes, you return to the component table by choosing **File => Exit**.

Copying a table menu

When you wish to copy an existing table menu, this is done just like you copy an ordinary table. This is described in “Creating a new Microsoft Access table” on page 42 — see “Copying a table” on page 45 in particular.

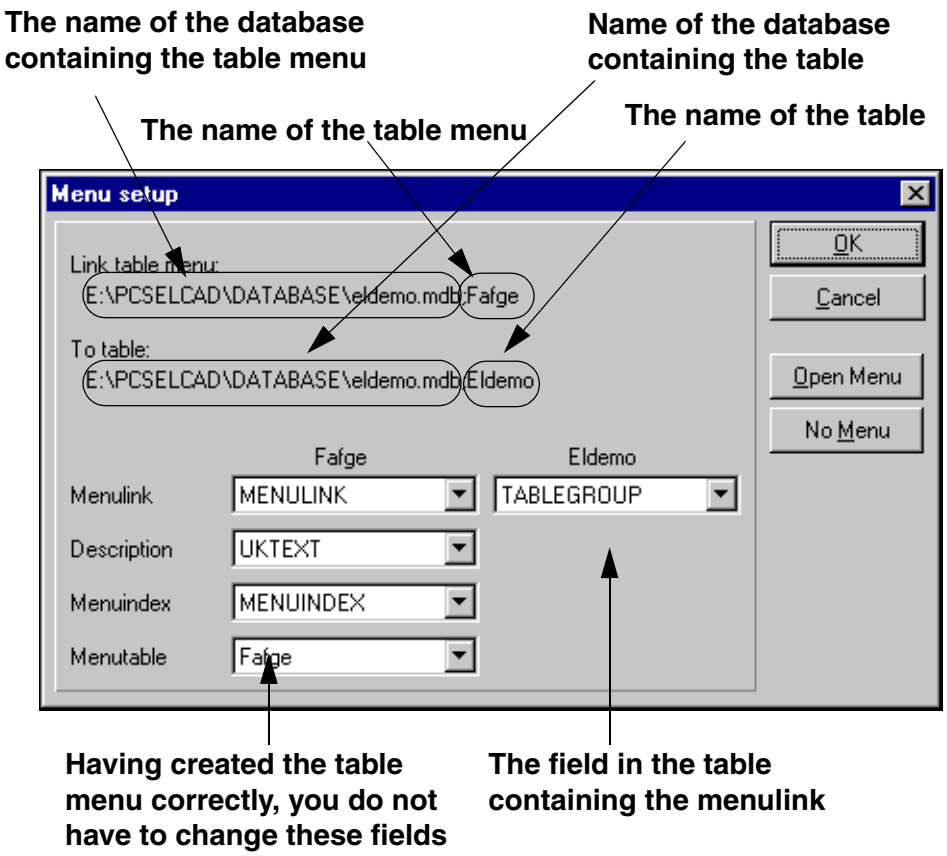
When you have copied the table menu, you attach the table menu to a table as described in “Connecting table menus to tables” on page 59.

CONNECTING TABLE MENUS TO TABLES

THIS CHAPTER DESCRIBES
THIS CHAPTER DESCRIBES HOW TO CONNECT A TABLE MENU TO A TABLE IN A DATABASE.

— SEE HOW TO CREATE TABLE MENUS IN “CREATING TABLE MENUS” ON PAGE 51.
— SEE HOW TO CREATE TABLES IN “CREATING TABLES” ON PAGE 42.

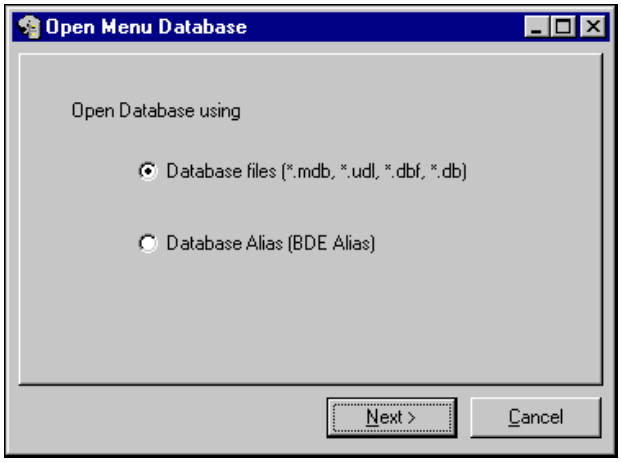
When you have a table in a database, for which you want to use a specific table menu, you must first open the table in the database. Then choose **Menu => Properties**:



In this example, the name of the database is *eldemo.mdb*. In this database the table *Eldemo* is placed, which is connected to the table menu *FAFGE*. The table menu *FAFGE* is also placed in the database *eldemo.mdb*.

To select the menu click on **Open Menu**, and do the following:

1) Choose how you wish to open the database in which the table menu is placed:



Click on ***Database files (*.mdb, *.udl, *.dbf, *.db)*** and click **Next**

2) You now enter the dialog box **Open**: Click on *eldemo.mdb* and click **OK**

3) You now enter the **Choose table** dialog box: Choose the table *FAFGE* (containing the table menu) and click **Next**

You now return to the dialog box **Menu setup**, where you can see that the table menu now is attached to the table. Click **OK**.

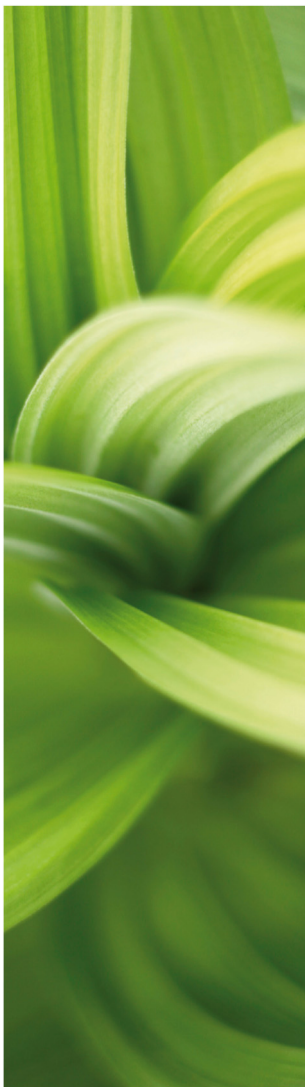
You have now connected a table in a database to a table menu.

The fields in the lower part of the menu

Please note, that as long as you are applying the FAFGE menu, and have adapted the table menu as described above, there is no reason to change the contents of the four fields in the lower part of the menu.

In the table *Eldemo* the field **TABLEGROUP** specifies to which FAFGE group the various components belong.

The program now makes sure that the records in the table are displayed in the menupoint in the table menu, which is specified by the **TABLEGROUP** field in the table.



Importing data

This section covers:

- Loading a database from a component vendor ..... (p. 63)
- Importing a component vendor database into your own database ..... (p. 64)
- Importing comma separated files ..... (p. 69)
- Import using filters ..... (p. 73)
- Importing from Microsoft Excel ..... (p. 78)

IMPORTING DATABASES FROM COMPONENT VENDORS

In this chapter you load a component vendor database from a diskette/CD, and import it into your own database.

THIS CHAPTER DESCRIBES

— IMPORTING A COMPLETE COMPONENT VENDOR DATABASE ..... (P. 63)
— LOADING A COMPONENT VENDOR DATABASE ..... (P. 64)
— IMPORTING THE COMPONENT VENDOR DATABASE ..... (P. 64)
— IMPORTING PART OF A COMPONENT VENDOR DATABASE ..... (P. 67)

IMPORTING A COMPLETE COMPONENT VENDOR DATABASE

When you wish to import a complete component vendor database into a table in your own database, do the following:

MAKE A COPY OF YOUR TABLE BEFORE IMPORTING

BEFORE IMPORTING ANYTHING INTO A TABLE IN YOUR DATABASE, IT IS A GOOD IDEA TO MAKE A COPY OF THE TABLE.

IF YOU REGRET THE IMPORT, YOU CAN THEN APPLY THE COPY OF THE DATABASE.

1) Create your own database and a component table, if you have not already done so — see “Creating Microsoft Access databases” on page 39 and “Creating tables” on page 42

2) Install the component vendor database — see “Loading a PC|SCHEMATIC component vendor database” on page 64

3) Import the entire component vendor database into a table in your own database — see “Importing the component vendor database into your own database” on page 64

When you only wish to import part of the component vendor database, please read “Importing part of a component vendor database” on page 67.

Loading a PC|SCHEMATIC component vendor database

Start by inserting a CD/diskette from a component vendor, click **Start** in Windows, and choose **Run**. Now type in *a:\setup* and press <Enter>.

PC|SCHEMATIC Database does not have to be started

PC|SCHEMATIC Database does not have to be started in order to install a component vendor database.

Follow the installation guide until the database is properly installed.

You have now installed the database, but have not loaded it into your own database.

Importing the component vendor database into your own database

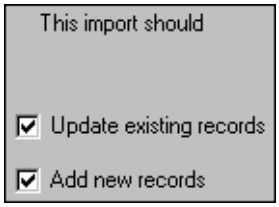
Enter the database program. Choose **File => Open** and open the table in your database to which you wish to import the components.

Choose **File => Import**, select the new component database in the dialog box and click **Open**.

Select the table you wish to import from the database, and click **Next**.

Adding/updating records

You now enter a dialog box, in which you have to specify the following:



If it is not the first time you import data from the chosen component vendor, you already have records with components from this vendor. However, the information might have changed. Enable ***Update existing records***, if you want your database updated with these changes, or disable the function if this is not the case.

Also enable ***Add new records***, if you wish to add new records to your database.

Click **Next**.

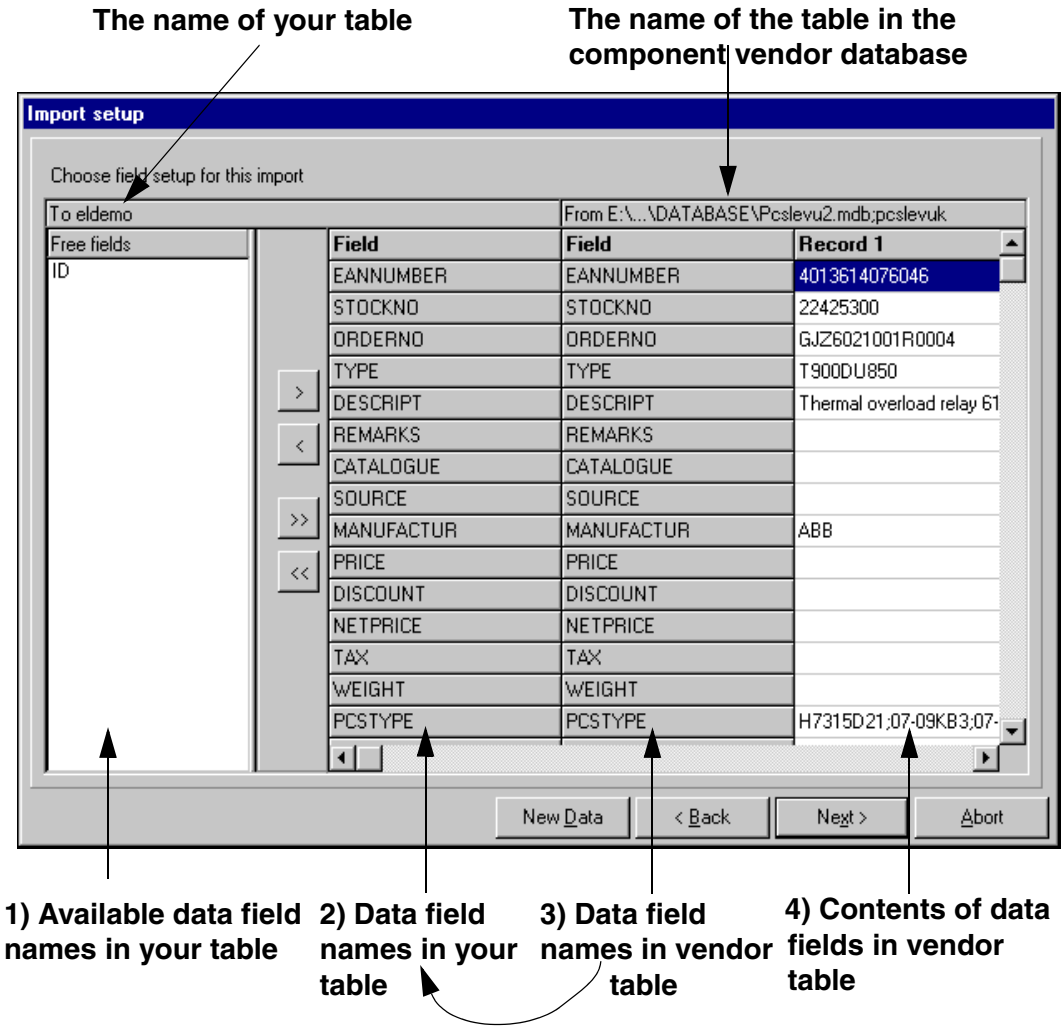
Import settings

You now enter a dialog box, in which you specify which fields in the imported database, you wish to transfer to which fields in your own database.

DO *NOT* IMPORT ID FIELDS

IF THE RECORDS OF THE COMPONENT VENDOR DATABASE CONTAINS AN ID FIELD, YOU SHOULD *NOT* IMPORT THIS FIELD.

THIS FIELD IS UPDATED AUTOMATICALLY.



In the right-hand side of the dialog box — above **3)** — you see the data fields of the table in the component vendor database. Above **1)** you see the data fields of the component table in your own database. By transferring the field names from **1)** to **2)** you specify which data fields in the table in your own database are going receive data from which data fields in the component vendor database table.

Transfering field names

Using the four buttons between **1)** and **2)** on the figure above, you transfer data field names from **1)** to **2)**. If you might loose data due to too short field lengths in your table, you will receive a warning.

Transfering all data fields

Because all the component vendor databases are adapted to ***PCschematic***, all data fields within the component vendor database table have the same field names and field lengths as the table **Eldemo**.

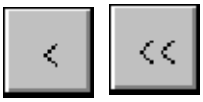
Therefore you can just click the **Transfer all** button, which causes all the field names to be aligned facing the corresponding field names in the other table.

Transfering individual data fields

For each individual data field you can also specify, that the contents of for instance the data field named ***EANNUMBER*** in your database table, shall receive data from the data field that (also) is named ***EANNUMBER*** in component vendor database. You do this by clicking on the data field ***EANNUMBER*** in **1) Available data field names in your table**, and then click on the **Transfer** button.

When you transfer the data field names in this way, they are arranged in the order in which you transfer them. When they are on the list (**2**), you can move the fields by clicking on them and dragging them to another position.

Reverting data fields



If you regret transferring some data fields, you can transfer them back individually by selecting the data fields and clicking the **Revert** button, or transfer all data fields back clicking the **Revert all** button.

Drag and drop

You can also transfer the data fields individually using *drag* and *drop*.

Viewing records in the file you import from

In the right-hand side of the dialog box (**4**), you can see the contents of the file you are going to import.

When you wish to see another record in the same file, click on the **New data** button in the lower part of the dialog box.

Memorizes settings

If you have previously imported a component vendor database, the program memorizes the settings you specified at that time.

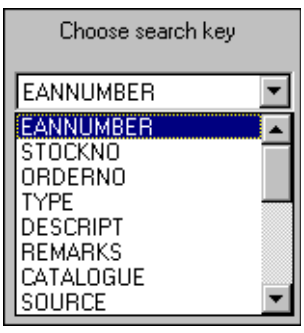
Transfer the data fields, and click **Next**.

Search key

The field ***Choose search key*** then appears in the left-hand side of the dialog box.

A *search key* is a data field, which uniquely identifies the records in the table.

The data field *remarks*, for instance, does not uniquely identify a record, because several components can have the same remarks.



On the other hand, the *EANNUMBER* of the component, precisely identifies the component. Therefore choose *EANNUMBER* as the search key in this case.

Using the search key, the program examines whether there exists components in your table, which also exists in the table you import. Depending on what you specified earlier — in “Updating/adding records” on page 71 — the program will update these records or not.

— If you need another search key (in another table in another context), click the down-arrow and choose another search key.

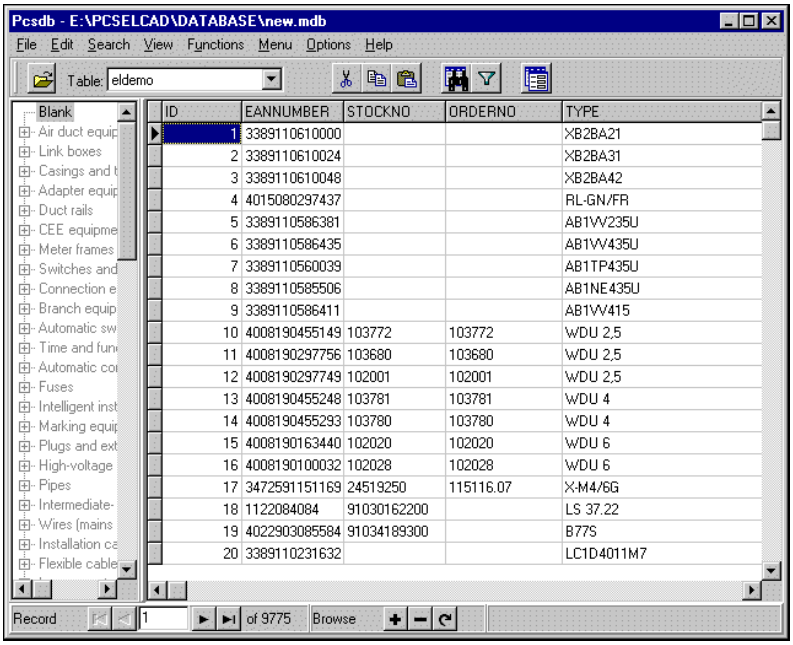
Import the file

Click on **Import**.

You now enter a dialog box, in which you can see which table is being imported, and which table it is imported into. Click **Start**.

The program now imports the table, and after completing the import, it is displayed how many records have been updated and added. Please note, that existing records cannot be deleted this way. Click **OK**.

You have now imported the table, and see the contents of the table on the screen.



Importing part of a component vendor database

When you wish to import a complete component vendor database into your own database, do the following:

Make a copy of your table before importing

Before importing anything into a table in your database, it is a good idea to make a copy of the table.

If you regret the import, you can then apply the copy of the database.

1) Create your own database and a component table, if you have not already done so — see “Creating Microsoft Access databases” on page 39 and “Creating tables” on page 42

2) Install the component vendor database — see “Loading a PC|SCHEMATIC component vendor database” on page 64

3) Export the desired part of the component vendor database to a comma separated file — see “Exporting comma separated files” on page 83

4) Import the comma separated into your own database — see “Importing comma separated files” on page 69

When you wish to import the complete component vendor database, please read “Importing a complete component vendor database” on page 63.

IMPORTING COMMA SEPARATED FILES

THIS CHAPTER DESCRIBES
THIS CHAPTER DESCRIBES HOW TO IMPORT A COMMA SEPARATED FILE INTO A TABLE IN **PC|SCHEMATIC Database**.

To import a comma separated file, you will typically need to know in which order the data fields are placed in the file. However, you get the option to see what the file looks like before importing it.

If you do not know what a comma separated file is, please read “What is a comma separated file?” on page 83.

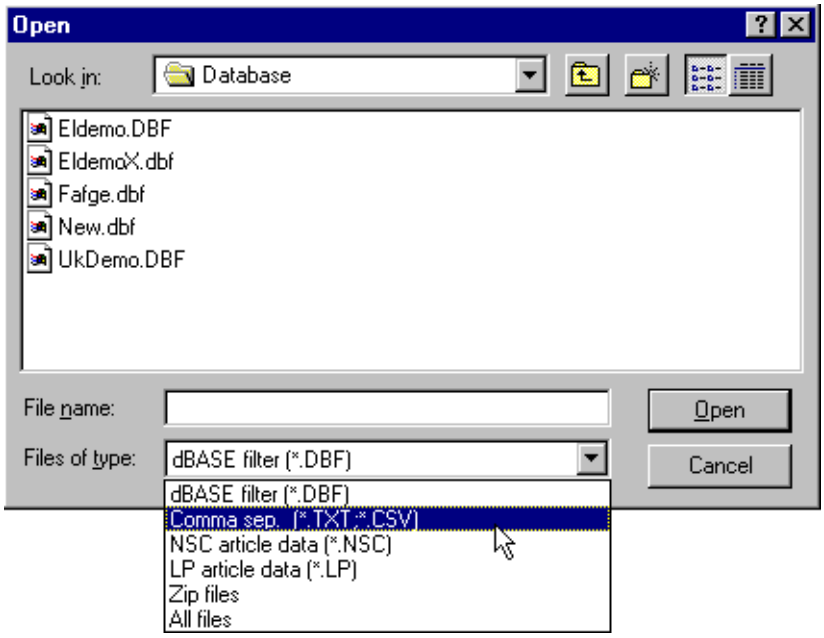
MAKE A COPY OF YOUR TABLE BEFORE IMPORTING

BEFORE IMPORTING ANYTHING INTO A TABLE IN YOUR DATABASE, IT IS A GOOD IDEA TO MAKE A COPY OF THE TABLE.

IF YOU REGRET THE IMPORT, YOU CAN THEN APPLY THE COPY OF THE DATABASE.

OPENING A COMMA SEPARATED FILE

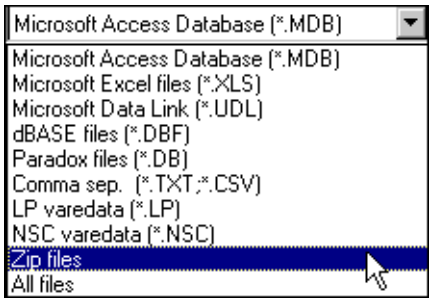
To open a comma separated file, you first choose **File => Import**.



Choose the file type ***Comma sep.***, and double-click on the file you wish to import.

Opening zip files

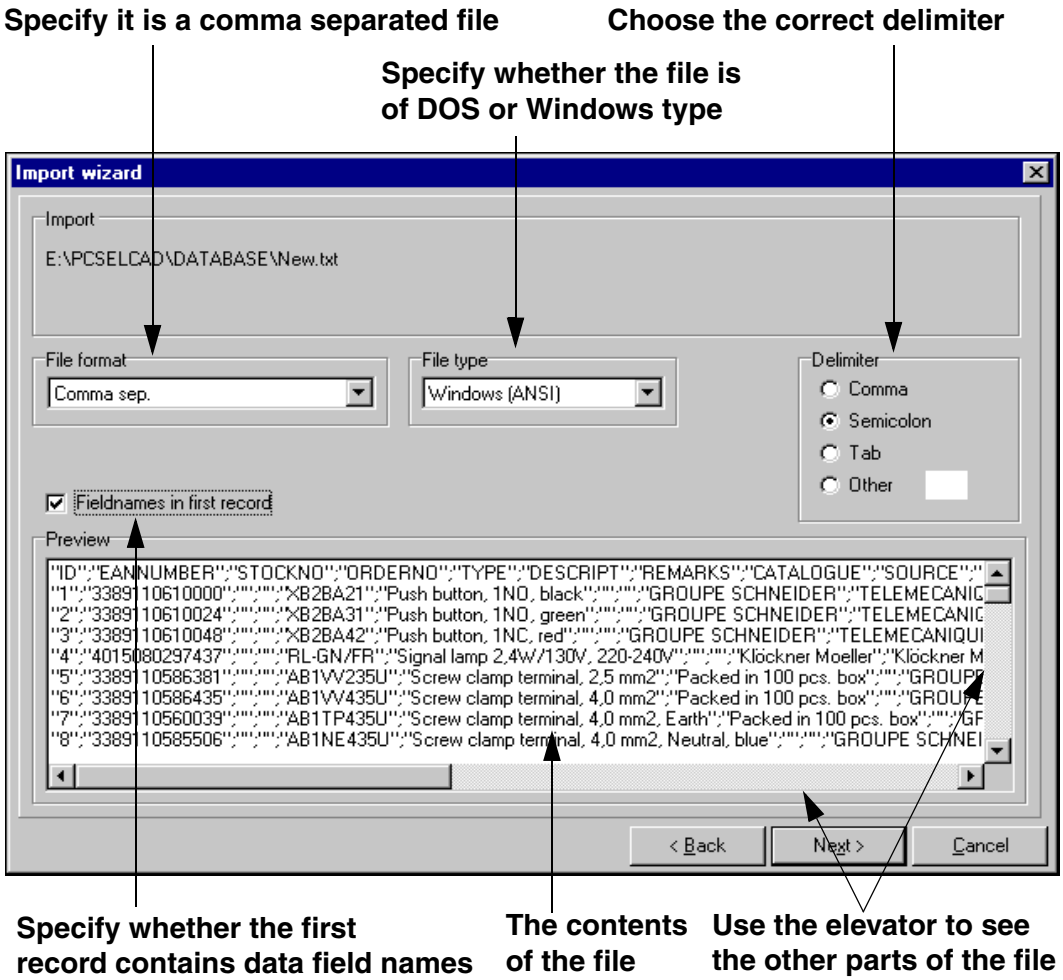
If the comma separated file is saved as a zip file, choose the file type ***Zip file*** instead, and double-click on the file you wish to import.



The program then automatically opens the zip file. — Then continue as described in the following.

Specifying the file setup

You now enter this dialog box, where you specify information on the type of the file:



Here you first make sure that the ***File format*** is set to ***Comma sep.***. If this is not the case, you must choose the right format under ***File format***.

Under ***Preview*** in the lower part of the dialog box, you can see how the program understands the contents of the file, when it uses the settings you have specified in the upper part of the dialog box.

This might be used to see whether the file is of the type ***Windows (ANSI)*** or of the type ***DOS (PC-8)***. These two file types use different signs for languages having special characters. Click on the type below ***File type*** which gives the right result.

Furthermore, you can see which delimiter is used in the file, and then specify the correct separator under ***Delimiter***.

If the first record contains field names, you have to enable ***Field names in first record***.

Click **Next** when the settings are correct.

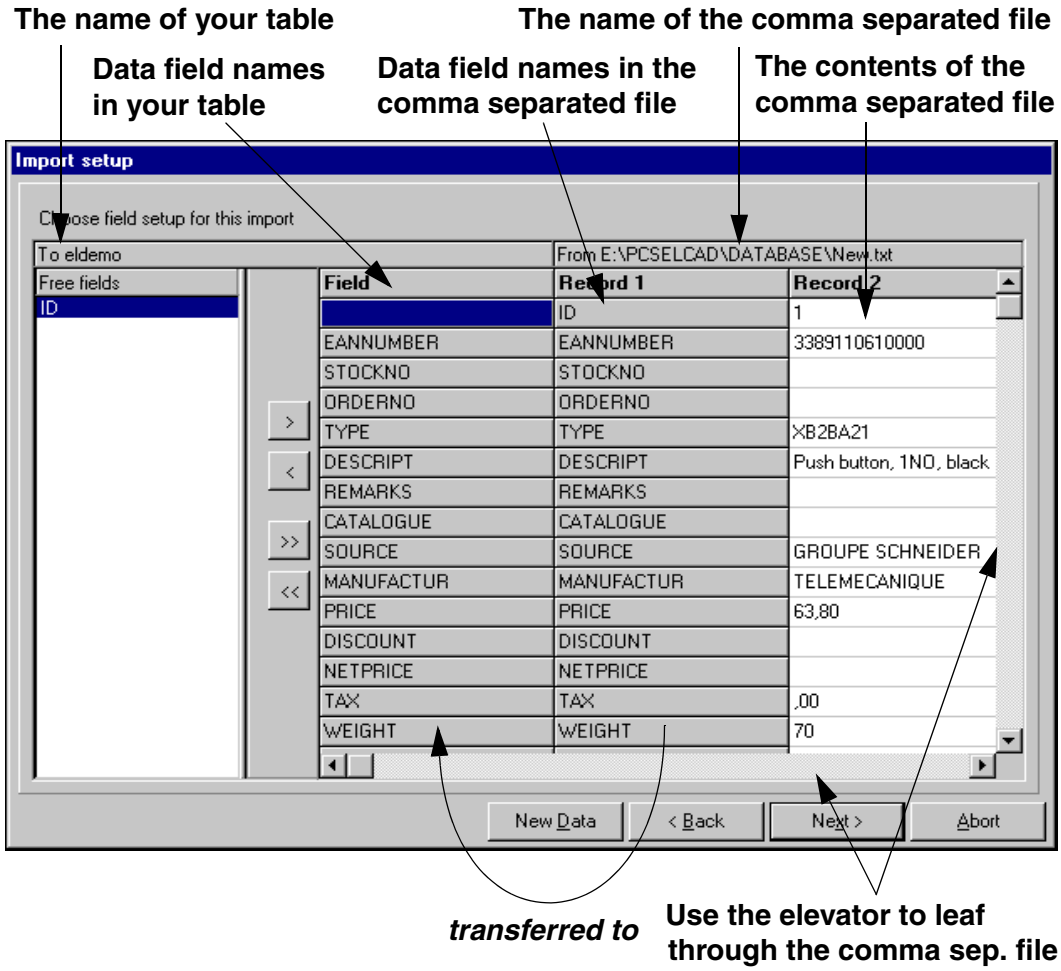
Updating/adding records

You now have to specify whether you wish to update and/or add records to the database. This is elaborated in “Updating/adding records” on page 71.

Click **Next**.

Import settings

You now enter this dialog box:



If you do not already know which information the individual fields in the comma separated file contains, you can use the elevator to leaf through the file, which is displayed in the right-hand side of the dialog box. Using the horizontal elevator enables you to leaf back and forth between five different records at the time. If you wish to see the five following records, you click **New data**.

This enables you to specify which data fields in your table, that are going to receive data from which data fields from the comma separated file.

On the image above, the first record contains data field names, which makes it easier to find the corresponding data fields in your own table. The first record will always be available in this dialog box, even though it is not going to be imported because you enabled ***First record contains field names*** in the previous dialog box.

Place the field names below **Data field names in your own DB** facing the appropriate data fields in the file, as described in “Transfering field names” on page 65.

DO *NOT* COPY ID FIELDS

IF THE RECORDS OF THE COMPONENT VENDOR DATABASE CONTAINS AN ID DATA FILED, YOU SHOULD *NOT* IMPORT THIS DATA FIELD.

THIS FIELD IS UPDATED AUTOMATICALLY.

Click **Next**.

Now specify the search key and click **Next**, as described in “Search key” on page 66. Finally, import the file by clicking **Import**, as described in “Import the file” on page 67.

You have now loaded the contents of the comma separated file into your table.

Click **OK**.

FILTERED IMPORT

THIS CHAPTER DESCRIBES
THIS CHAPTER DESCRIBES HOW TO IMPORT FILE TYPES MEANT FOR OTHER PROGRAMS.

Because **PC|SCHEMATIC Database** contains specially designed filters, it is possible to import some file types originally designed for other programs.

In this way you can for instance import price update disks from Nordisk Solar and Louis Poulsen. Both file types contains various types of records, and you therefore have to make special settings for each of these.

This chapter is an example on how to import a file of this type.

The next time you import a file having the same file name, you will not have to specify these settings again, because **PC|SCHEMATIC Database** memorizes the used settings.

MAKE A COPY OF YOUR TABLE BEFORE IMPORTING

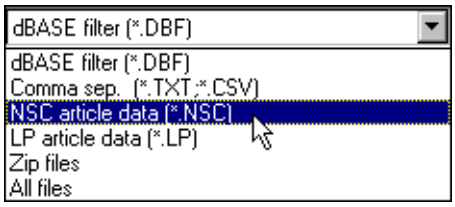
BEFORE IMPORTING ANYTHING INTO A TABLE IN YOUR DATABASE, IT IS A GOOD IDEA TO MAKE A COPY OF THE TABLE.

IF YOU REGRET THE IMPORT, YOU CAN THEN APPLY THE COPY OF THE DATABASE.

OPENING THE FILE

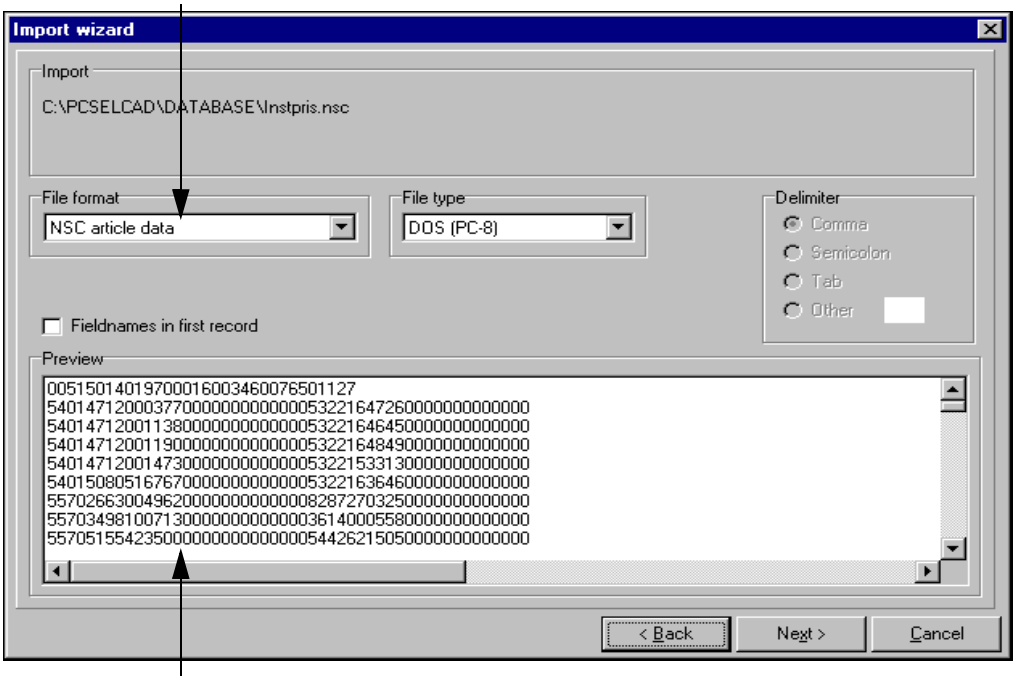
To open the file you choose **File => Import**, and click on the desired file type in the lower part of the dialog box — for instance ***NSC article data***.

Double-click on the file you wish to import.



You now enter this dialog box, where you just have to check whether the file format is set properly

Check that the file format is correct



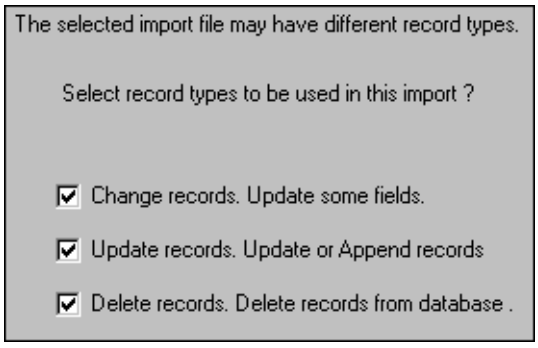
The contents of the file

Because the file is meant for another system, you cannot see the meaning of the contents in the lower part of the dialog box.

Click **Next**.

Which changes shall be made?

Because the file contains different types of records, you have to specify whether you wish to load all the changes/additions the file contains:



The record type ***Change records*** only contains minor changes (in this example price changes). Here you only change existing records in your component table; if the file contains corrections for a record you do not have in your component table, the component is not loaded.

If you enable ***Update records***, all records of this type are loaded into your database. If a record already exists for a component, the contents of this record is replaced by the contents of the update record.

Enabling ***Delete records*** will delete the specified records.

Records are only delete-marked in dBase

If you are working in a dBase table, the specified records only becomes delete-marked. Read more about delete marking in “Deleted records” on page 30.

Make your choices and click **Next**.

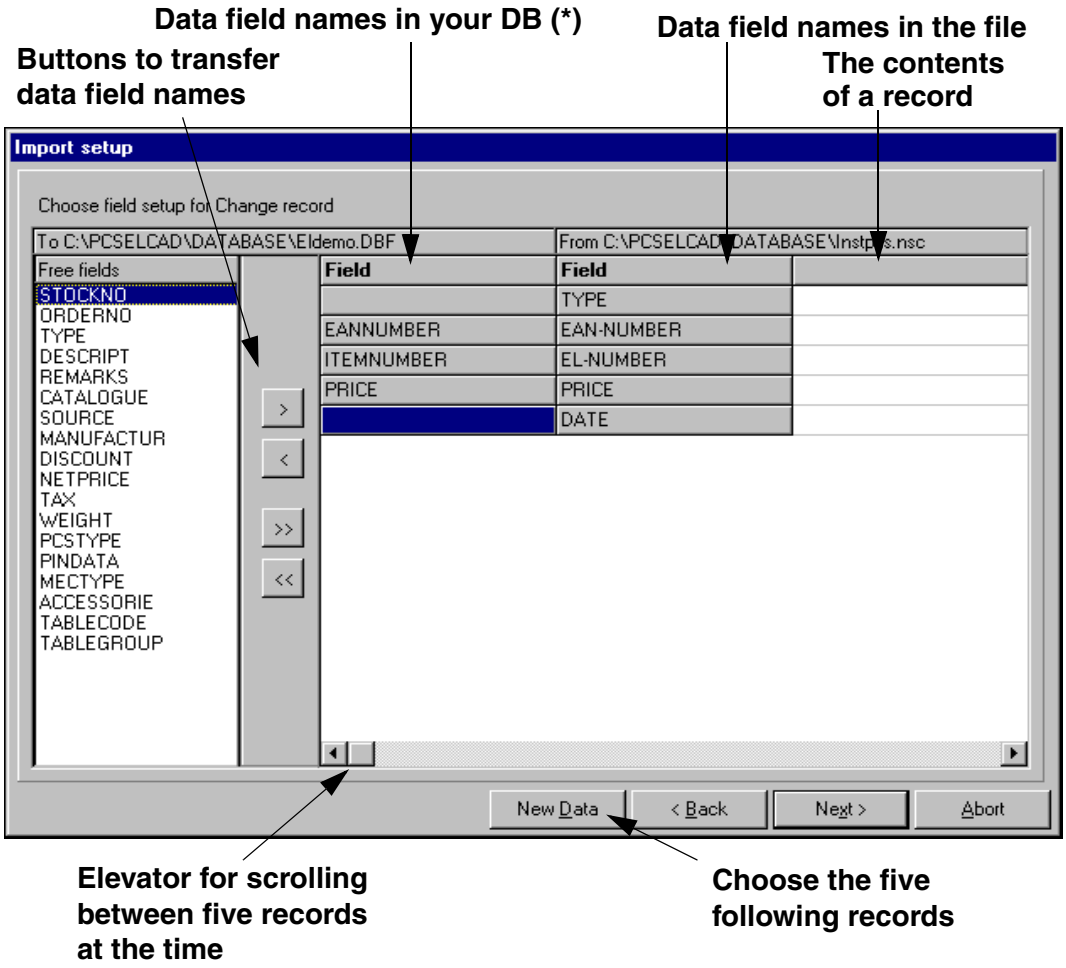
Choosing field names for Change records

Now you have to specify which fields in your database, that matches which fields in the Change records in the file.

Therefore transfer all the relevant fields from the list of ***Free fields*** to the column with the field names of your database (*). This is thoroughly dealt with in “Transfering field names” on page 65.

DO *NOT* COPY TYPE FIELDS

THE FIELD ***TYPE*** IN THE FILE SPECIFIES WHICH TYPE OF RECORD IT IS IN THE *FILE* ITSELF (HERE A CHANGE RECORD). THIS IS *NOT* THE SAME AS THE ***TYPE*** DATA FIELD IN FOR INSTANCE THE DEMO DATABASE.



Click **Next**.

Now specify which search key shall be applied during the load of this record type, and click **Next**.

You may read more about search keys in “Search key” on page 66.

Choosing data field names for Update/Append records

In a similar way you now have to specify which data fields in the Add records of the file, corresponds to which data fields in your database.

Please note, that you do not have to fill out all the fields.

Click **Next**.

Now specify which search key shall be applied for this record type, and click **Next**.

Choosing data field names for Delete records

Same procedure is repeated for the Delete records of the file.

Field	Field	Record 2
	TYPE	5
EANNUMBER	EAN-NUMBER	4014712000377
	REFERENCE-EAN-NUMBER	0
ITEMNUMBER	ITEM-NUMBER	5322164726
	REPLACEMENTNUMBER	0

Transfer the relevant data fields, click **Next**, specify search key, and click **Next** once again.

Now click **Start** to import the file.

Click **OK** when the import is completed.

IMPORTING MICROSOFT EXCEL DOCUMENTS

THIS CHAPTER DESCRIBES

— RESERVATIONS FOR IMPORT OF MICROSOFT EXCEL DOCUMENTS .....(P. 78)
— IMPORTING A MICROSOFT EXCEL DOCUMENT .....(P. 79)

RESERVATIONS FOR IMPORTING MICROSOFT EXCEL DOCUMENTS

The program ***Microsoft Excel*** is a spreadsheet program, and not a database program. However, it is possible to import Microsoft Excel documents in **PC|SCHEMATIC Database**, that opens the Microsoft Excel documents as if they contained a database. No 100% guarantee is given that this will take place without errors, but it normally works quite well.

However, there are a few things which can give errors during the import, because these things are outside the control of **PC|SCHEMATIC Database**. These things are described in the following.

MICROSOFT EXCEL DOES *NOT* HAVE TO BE INSTALLED

IT IS POSSIBLE TO IMPORT MICROSOFT EXCEL DOCUMENTS *WITHOUT* HAVING INSTALLED MICROSOFT EXCEL ON YOUR PC.

Defines the types of the fields by examining the first 12 rows

When you import data from a Microsoft Excel document the first 12 rows in the document are examined to identify the type of the fields in the Microsoft Excel document.

When other datatypes are used in succeeding rows in the Microsoft Excel document, this can result in a data error.

Understands the first row as field names

When importing the Microsoft Excel document, the program understands the first row as field names. If this is not the case in the document, this may result in an error.

Create new fields before the import

If the Microsoft Excel document contains field names, which are not defined in your table in **PC|SCHEMATIC Database**, you must create these fields *before* importing the Microsoft Excel document. This is because you can not create new field names in your table during the import.

Blank fields can cause errors

It is important to apply the same datatype all the way down for all fields in a column in the Microsoft Excel document. If for instance some of the last fields are blank, this can cause an error if the first fields contains numbers.

Importing a Microsoft Excel document

To import a Microsoft Excel document into a table in **PC|SCHEMATIC Database**, do the following:

1) Open the table you wish to import the Excel document into

2) If the Microsoft Excel table contains fields that does not exist in your own table, you must create these fields before importing the Microsoft Excel table — if you wish to import any of these fields

3) Choose **File => Import**

4) You now enter the dialog box **Open**: Select the file type ***Microsoft Excel files (*.xls)***, click on the desired Microsoft Excel document, and click on **Open**

5) Click on the spreadsheet (the “table") you wish to import and click **Next**

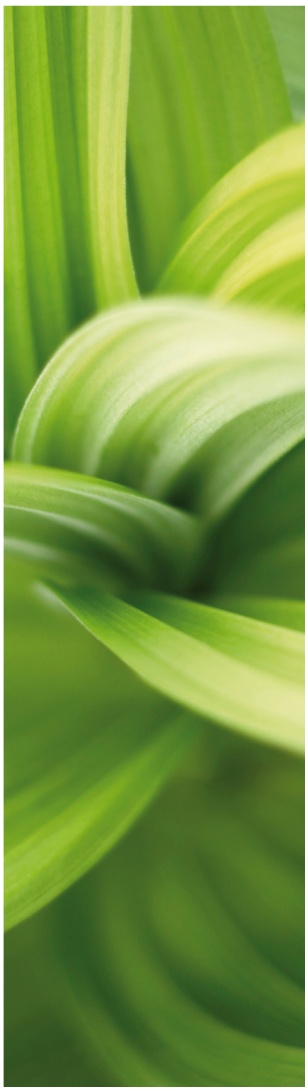
6) The import then continues as described in “Importing the component vendor database into your own database” on page 64

Do *not* copy ID fields

If you import from a table with an ID field — or another field-name that matches the field you have set to be primary key in your table and has the datatype ***AutoInc*** — you shall *not* import this field.

This field will be filled out automatically.

— Please read “Reservations for importing Microsoft Excel documents” on page 78.



Exporting data

In this section you can see how to export a table in a database from PC|SCHEMATIC Database to a file.

This can be done by:

- Creating a comma separated file .....(p. 83)
- Exporting to Microsoft Excel.....(p. 87)
- Giving a complete database away .....(p. 91)
- Direct loading into another program .....(p. 94)

EXPORTING COMMA SEPARATED FILES

THIS CHAPTER DESCRIBES

— THE TWO WAYS OF PASSING ON DATABASE INFORMATION ..... (P. 83)
— WHAT IS A COMMA SEPARATED FILE? ..... (P. 83)
— WHICH PART OF THE DATABASE IS EXPORTED ..... (P. 84)
— CREATING A COMMA SEPARATED FILE ..... (P. 85)

TWO WAYS OF PASSING ON DATABASE INFORMATION

If you wish to export information from a database, you can do this in two ways:

1) You can create a so-called comma separated file, which can be loaded into many different systems. This is controlled from **PC|SCHEMATIC Database**.

2) You can copy the entire database including its table(s) and table menu(s) and all other connected files. This is done from Windows or DOS.

This chapter deals with how to create a comma separated file.

— If you wish to create a comma separated file for Microsoft Excel, please read “Exporting to Microsoft Excel” on page 87.

— If you wish to copy your entire database, you can see how to do this in “Giving away databases” on page 91. Doing this requires further knowledge of Windows or DOS.

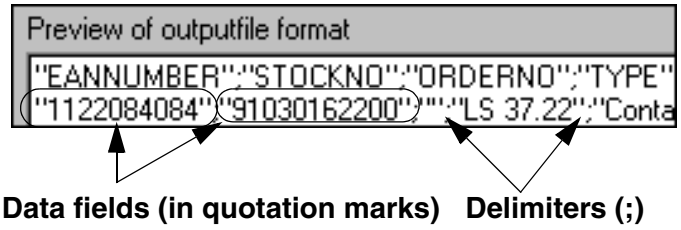
WHAT IS A COMMA SEPARATED FILE?

As explained in “What is a database?” on page 13, all information in a database is arranged in tables, where the various data fields have their own field names.

The internal information about the construction of these tables, are organized in different ways depending on the database program. If you need to use the information from your PC|SCHEMATIC database in another system, you therefore need to be able to save the information in a more generally recognizable way.

Nearly all systems are able to load comma separated files, as it is close to being the simplest way possible to save information.

In a comma separated file the contents of all the data fields are listed immediately following one another, only separated (!) by a single sign — for instance a *comma* or a semicolon.



To make sense of the file, the receiver therefore need to know that the first data field in the comma separated file is for instance of the type ITEMNUMBER — and so on.

ONLY THE CONTENTS OF THE DATA FIELDS ARE PASSED ON

ONLY THE *CONTENTS* OF THE FIELDS ARE PASSED ON.

INFORMATION ABOUT *DATA FIELD NAMES* ARE *NOT NECESSARILY* IN THE FILE!

WHICH PART OF THE DATABASE IS EXPORTED

When you export a comma separated file, you have to be aware that you only export the part of the database, which is displayed on the screen for the moment.

Thus, when you are applying the data filter, you will only export the filtered part of the database — read more about this in “The Data filter” on page 26.

It also has influence whether you see the entire database, or just the contents of a single menupoint. Read more about this in “Editing records” on page 28.

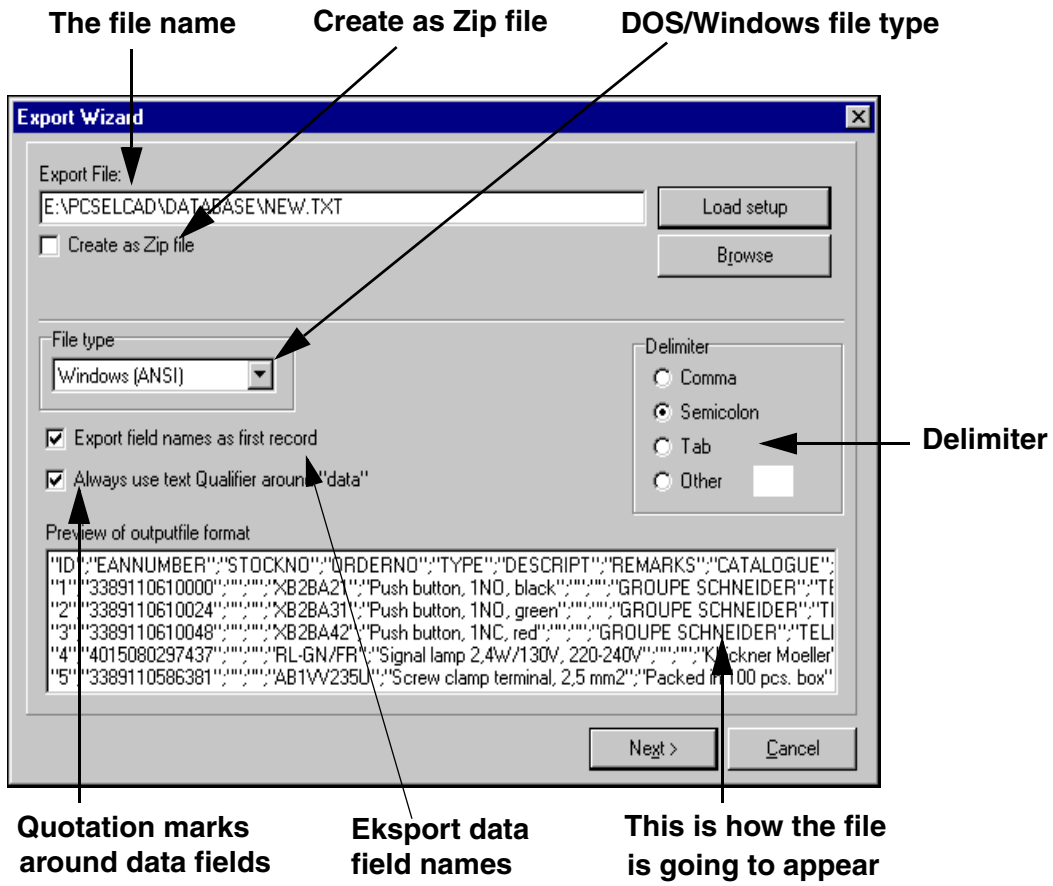
If you are familiar with SQL, you can also use SQL queries to select what to export. — Read more about this in “SQL queries” on page 35.

You can also export the entire database as a comma separated file. To do this you have to neither apply the table menu, the data filter nor SQL queries when you specify that you wish to export to a file.

Creating a comma separated file

To create a comma separated file for export choose **File => Export**.

This leads you into this dialog box, where you specify the file settings:



Please note, that you in this dialog box specify whether the file should be packed as a zip file, by clicking on ***Create as Zip file***. — If you create a zip file, only the zip file is created, and *not* the ordinary text file, whose name is specified in the ***Export file*** field.

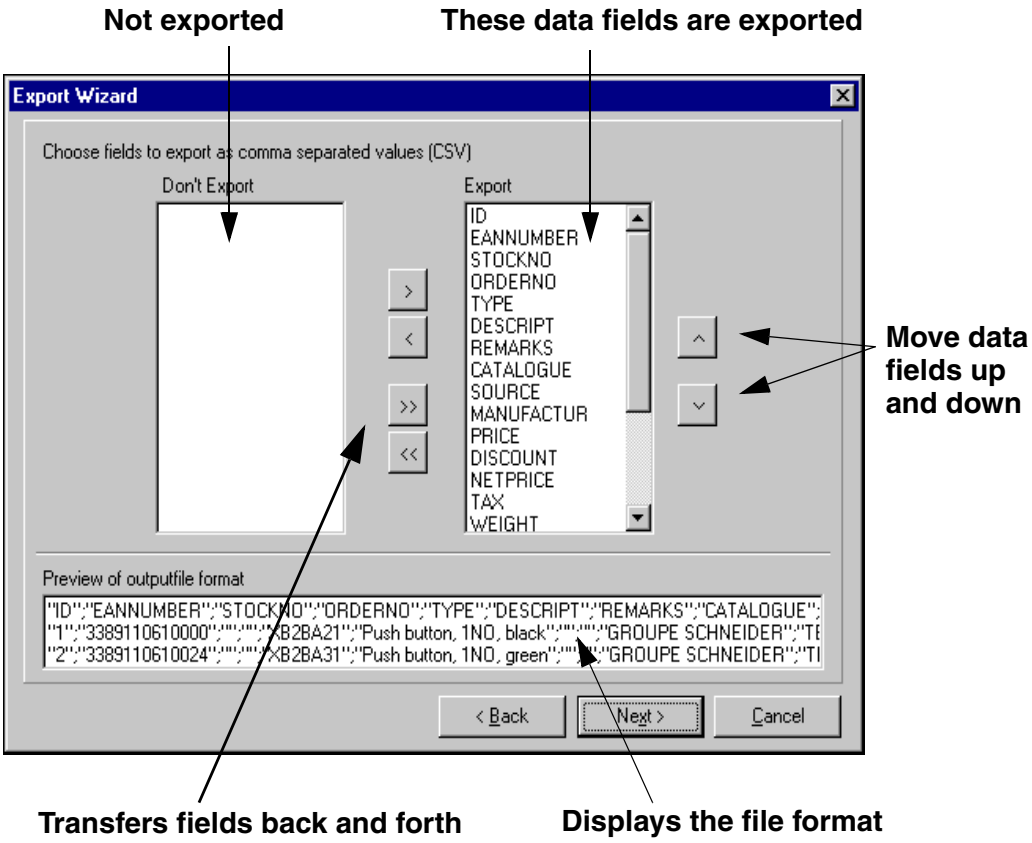
If you enable ***Export field names as first record***, the first record will contain the names of the chosen data fields.

Because DOS and Windows does not use the same character sets, you will have to specify whether the file is going to be exported to a DOS or a Windows program. This is important when working in languages using special characters.

In the lower part of the dialog box, you will see what your file is going to look like.

When you have typed in the name of the file (and path) in the ***Export File*** field, you click **Next**.

You then enter the following dialog box:



Specifying which data fields to export

Here you specify which data fields you wish to export, by transferring the desired data fields to the list in the right-hand side of the dialog box.

The data fields are transferred back and forth by clicking on the buttons, as explained in “Importing the component vendor database into your own database” on page 64.

Specifying the order of the data fields

In the export file the data fields are arranged in the order in which they are listed on the list on the right-hand side of the dialog box.

By selecting one or more fields and applying the **Up** and **Down** buttons in the right-hand side of the dialog box, you can change the order in which the data fields are going to be placed in the export file.

You select more data fields at a time by clicking on one field in the list, then press down [Shift] and click on another data field. Hereby you select the two data fields, as well as all intermediate data fields. You can now move the selected data fields as described above.

In the window below ***Preview output file format***, you can see how the file is going to appear.

Click **Next** when you have determined which data fields to export, and the file is created.

Exporting to Microsoft Excel

This chapter describes how to export a table form **PC|SCHEMATIC Database** and import it in Microsoft Excel.

This chapter describes

— Export from **PC|SCHEMATIC Database** ..... (p. 87)
— Import in Microsoft Excel ..... (p. 88)
— Printing from Microsoft Excel ..... (p. 90)

Export to Microsoft Excel only via comma separated file

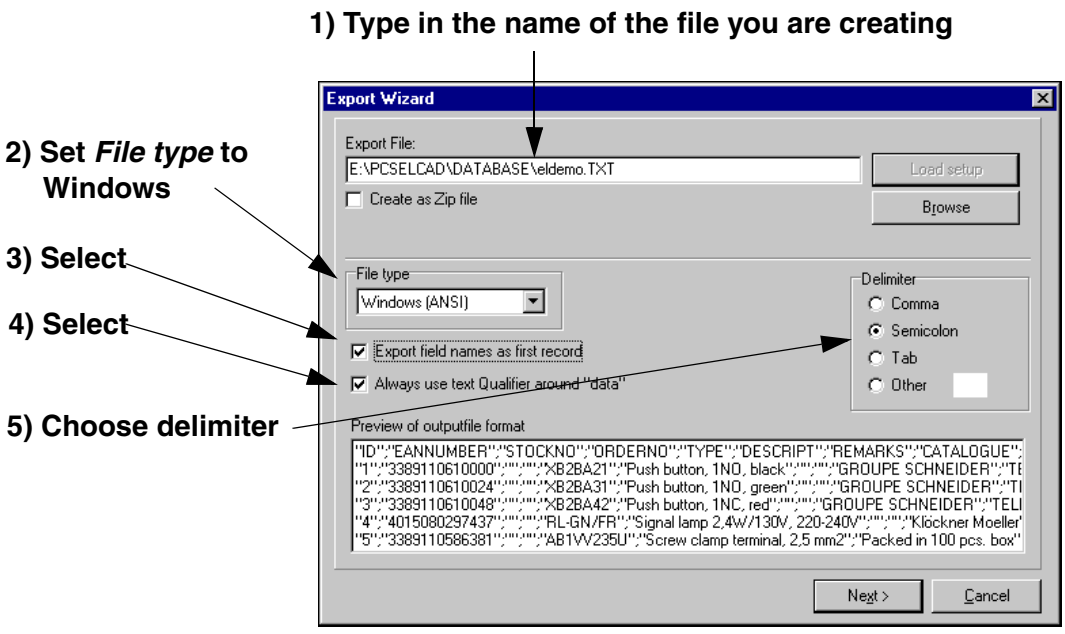
You can not export directly to Microsoft Excel, because Microsoft Excel is spreadsheet and not a database program.

However, you can export a comma separated file from **PC|SCHEMATIC Database**, which you then import in Microsoft Excel.

Export from PC|SCHEMATIC Database

In order to export from **PC|SCHEMATIC Database** you must first open the table you wish to export from. If you only wish to export part of the table to Microsoft Excel, please read “Which part of the database is exported” on page 84.

Then choose **File => Export**, and you enter this dialog box:



Here you do the following:

1) Type in the name of the comma separated file you create

2) Set ***File type*** to *Windows*

3) Select ***Export field names as first record***, to transfer the field names to Microsoft Excel

4) Select ***Always use text Qualifier around "data"*** to make sure that you can always import the contents of all the data fields in Microsoft Excel

5) Select a delimiter — for instance *Semicolon* — and remember this character when you import the file in Microsoft Excel

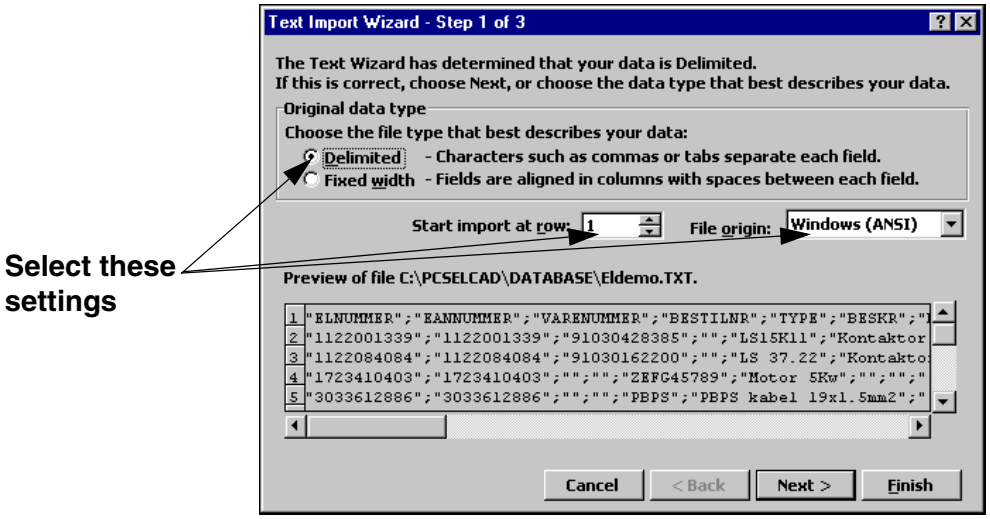
Then click on **Next**, and follow the procedure for exporting comma separated files, as described in “Specifying which data fields to export” on page 86.

Importing a comma separated file in Microsoft Excel

Now open Microsoft Excel in order to import the comma separated file, and choose **File => Open** in Microsoft Excel.

Choose the comma separated file, and click on **Open**.

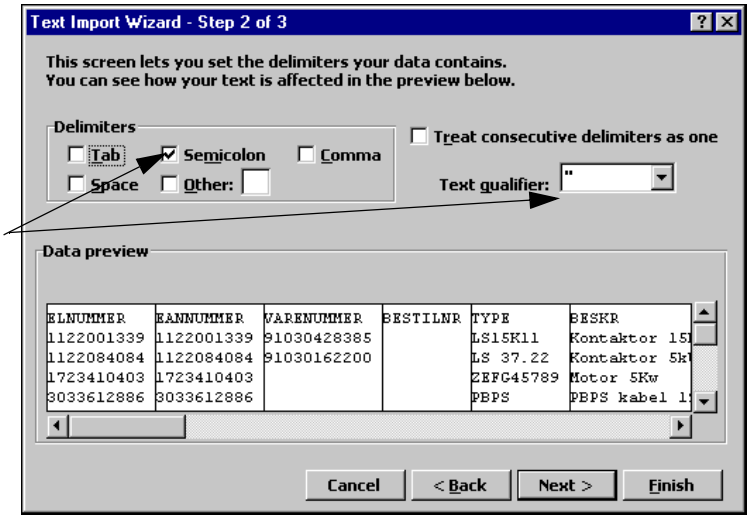
You now enter the following dialog box:



Select the displayed settings — if Microsoft Excel does not do this automatically — because they match the settings specified during the export from **PC|SCHEMATIC Database**.

Click on **Next**, and you enter this dialog box:

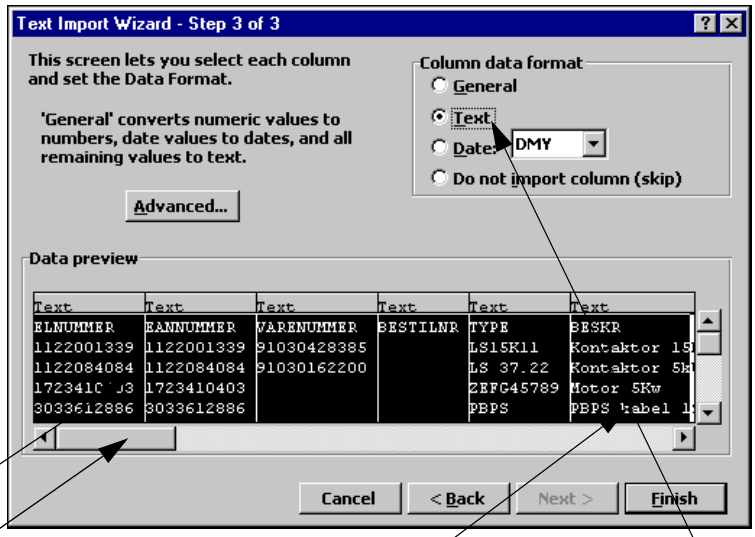
Select these settings so that they match what you exported from PC|SCHEMATIC Database



Here you specify how the table fields are separated in the comma separated file.

Therefore set ***Delimiters*** to *Semicolon* and ***Text qualifier*** to ", so that it matches the settings in the comma separated file.

Click on **Next**, and you enter this dialog box:



1) Click here
2) *Drag* the elevator to the last column
3) Press down <Shift> and click on the last column
4) Click on *Text*

Here you specify which format the fields are going to have in Microsoft Excel. In order to let these match the fields in **PC|SCHEMATIC Database** they must all be set to the format ***Text***. Do that the following way:

1) Click on the first column in the displayed part of the table

2) *Drag* the elevator to the last column of the table

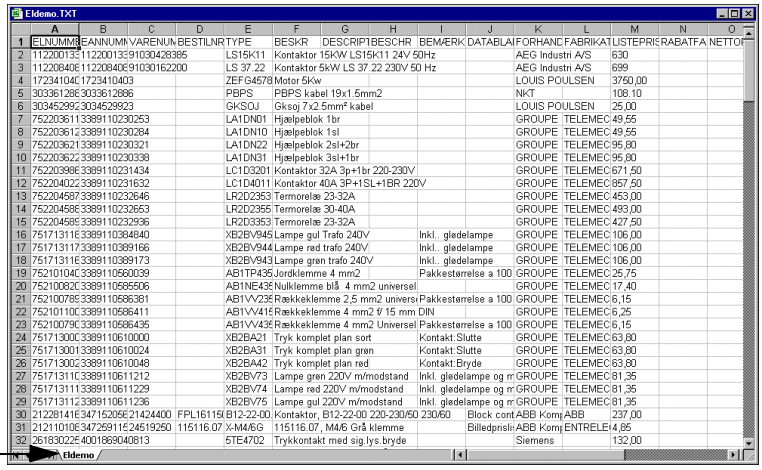
3) Press down <Shift> and click on the last column

4) All of the columns are now selected: Click on *Text* under ***Column data format***

Click on **Finish**.

The table is now imported in Microsoft Excel:

Field names



The name of the file

You can now apply all of the Microsoft Excel functions for adjusting the widths of the columns, changing style and type of field names etc.

Printing tables from Microsoft Excel

Because **PC|SCHEMATIC Database** has no printing function of its own, you can do the following when you wish to print a table:

1) Import the table in Microsoft Excel as described above

2) Adjust the appearance of the tables in Microsoft Excel to your liking

3) Print the table from Microsoft Excel

Giving away databases

This chapter describes the options you have for giving complete databases away by just copying these from Windows (or DOS) — and *not* by exporting from **PC|SCHEMATIC Database**.

This chapter describes

— On giving away databases....(p. 91)
— Giving away a Microsoft Access database....(p. 91)
— Giving away a dBase table/database....(p. 92)
— Remember symbols and connection name files....(p. 93)
— UDL files and BDE alias....(p. 93)
— Execute the copying....(p. 93)

Technical chapter

This chapter is of a bit more technical character than some of the others in the manual, and it is necessary that you know how to copy files in Windows (or DOS).

On giving away databases

The advantage of copying a database as described in the following, is that you can pass on the database exactly as it is; including the table menu, specification of which fields are index fields, as well as the memorized information from the import/export dialog boxes.

If you wish to give a copy of your entire database to someone that also use **PC|SCHEMATIC Database**, it is more than just the database you have to pass on.

— Please note, that you can also export all the records in the database, as described in “Exporting comma separated files” on page 83.

Giving away a Microsoft Access database

A Microsoft Access database is placed in a so-called ***mdb*** file, i.e. a file with the extension ***.mdb***.

Apart from the database file there can exist an attached ***idb*** file for each individual table in the database. These ***idb*** files contains information about which menu is connected to the table, as well as the applied import/export settings for the table.

The name of the ***idb*** files begins with the name of the database, followed by an underscore character and the name of the table. Therefore the ***idb*** file for the FAFGE table from the database **eldemo.mdb** will have the name **eldemo_FAFGE.idb**.

If for instance you wish to copy all relevant files for the database **eldemo.mdb**, you must therefore just copy all files beginning with **eldemo** in the folder in which the database is placed.

Giving away a dBase table/database

A dBase table is placed in a so-called ***dbf*** file, which means it has the extension ***.dbf***. — Read more about dBase tables/databases in “Databases and the dBase format” on page 41.

Apart from the ***dbf*** file, two more files can be attached to the database:

1) A ***mdx*** file, containing information about the index fields of the database.

2) An ***idb*** file, containing information about to which table menu the database is attached, as well as the memorized settings from the import/export dialog boxes.

Both has the same names as the database itself — only the extensions differ.

If the table is called ***test.dbf***, the names of the other files are ***test.mdx*** and ***test.idb*** respectively.

The idb file

It is not strictly necessary to copy the ***idb*** file. If you do not pass it on, this has two consequences:

1) The database will have no menu attached. This gives no error message, and you can work on the database without difficulties. The new user can subsequently attach another table menu.

2) The new user has to set up the import/export settings from scratch, the first time an import is made from a component vendor database.

If you wish to pass on the table menu, you also have to remember to copy the ***idb*** and the ***mdx*** file of the *menu file*. If you are using the standard FAFGE menu, there is no reason to copy the menu, because the new user has the same menu in his PC|SCHEMATIC program.

Remember symbols and connection name files

When you give away a database, you must remember the following:

1. User-defined symbols

If you have defined some symbols, which you are using in the database, you also have to copy these, and place them in the proper symbol libraries.

This goes for electrical as well as mechanical symbols.

2. Pindata files

If you are using pindata files for some of the symbols in your database, these must also be included. — Please refer to the **PC|SCHEMATIC Automation** manual.

UDL files and BDE alias

UDL files can be copied freely and be distributed to the necessary workstations. — See “MDAC and UDL files” on page 99.

However, you have to create a new BDE alias on each individual workstation. — See “Creating BDE database aliases” on page 103.

Copying the files

The copying itself is carried out in for instance Windows, where you can place the relevant files on a diskette/CD, which you pass on to the new user.

LOADING DATABASES INTO OTHER PROGRAMS

If you want to document/print out the database, or want to maintain the database using another database program, you can load the database directly into another program. — See “Exporting to Microsoft Excel” on page 87 if you wish to import tables in Microsoft Excel.

Because the files in **PC|SCHEMATIC Database** are saved in standard Microsoft Access or dBase format, this requires that the other program can open Microsoft Access or dBase files.

APPLY THE ACCESS FORMAT WITH OTHER MICROSOFT PRODUCT

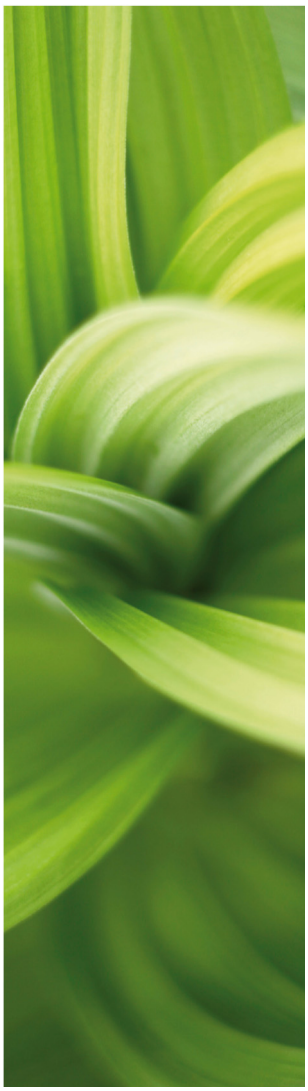
IF YOU WISH TO USE THE DATABASE IN CONJUNCTION WITH OTHER MICROSOFT PROGRAMS, IT IS RECOMMENDED TO USE THE MICROSOFT ACCESS FORMAT, BECAUSE THIS FORMAT IS SUPPORTED THE BEST BY OTHER MICROSOFT PRODUCTS.

DO NOT SAVE IN THE ORIGINAL DATABASE FILE FROM OTHER PROGRAMS

IF YOU WISH TO *SAVE* THE DATABASE FILE FROM ANOTHER PROGRAM, YOU HAVE TO BE CAREFUL *NOT* TO SAVE IT IN THE ORIGINAL DATABASE FILE — UNLESS THE PROGRAM CAN MAINTAIN THE INDEX FILE (***MDX*** FILE) OF THE DATABASE. YOU MAY READ MORE ABOUT THIS FILE TYPE IN “GIVING AWAY A DBASE TABLE/DATABASE” ON PAGE 92.

As a precaution, please make a copy of the database *before* you try whether using the database in another program has any side effects. Read more about this in “Maintaining databases” on page 37.

— Please note, that you can also transfer the database to another program, by exporting it as a comma separated file. Read more about this in “Exporting comma separated files” on page 83.



Other database formats

In this section you can see how to support other database formats than Microsoft Access and dBase:

- Supporting other database formats ..... (p. 97)
- MDAC and UDL files ..... (p. 99)
- Creating a BDE database alias ..... (p. 103)

SUPPORTING OTHER DATABASE FORMATS

This chapter describes how other database formats than Microsoft Access and dBase are supported.

THIS CHAPTER DESCRIBES

— WHAT DOES PC|SCHEMATIC SUPPORT? ..... (P. 97)
— HOW TO SUPPORT OTHER DATABASE FORMATS THE BEST ..... (P. 98)

ONLY FOR THE DATABASE ADMINISTRATOR OF THE COMPANY

TO HANDLE SUPPORT OF OTHER DATABASE FORMATS IN **PC|SCHEMATIC Database** REQUIRES DETAILED KNOWLEDGE ABOUT THE DATABASE FORMATS IN QUESTION.

HANDLING THIS IS A TASK FOR THE DATABASE ADMINISTRATOR OF THE COMPANY.

WHAT DOES PC|SCHEMATIC SUPPORT?

PC|SCHEMATIC supports dBase and Microsoft Access databases directly. When **PC|SCHEMATIC Automation** is installed for the first time, the program has an attached Microsoft Access database.

Furthermore, PC|SCHEMATIC is open to other database formats supported by either BDE, MDAC or ODBC.

Therefore you can also apply other databases, which are created and maintained in other database programs. You can for instance use MS-SQL, Oracle and practically any commonly used database program together with PC|SCHEMATIC.

INSTALL MDAC TO WORK IN THE MICROSOFT ACCESS FORMAT

MDAC *MUST* BE INSTALLED IF YOU WANT TO WORK DIRECTLY IN THE MICROSOFT ACCESS DATABASE FORMAT. — SEE HOW TO INSTALL MDAC IN “INSTALLING MICROSOFT DATA ACCESS COMPONENTS” ON PAGE 102.

INSTALL BDE TO WORK IN THE DBASE FORMAT

BDE *MUST* BE INSTALLED ON YOUR PC IF YOU WANT PC|SCHEMATIC TO SUPPORT THE DBASE DATABASE FORMAT DIRECTLY. MDAC IS NOT NECESSARY IF YOU ONLY WISH NO APPLY DBASE AND BDE ALIASES. — SEE HOW TO INSTALL BDE IN “INSTALLING BORLAND DATABASE ENGINE” ON PAGE 107.

How to support other database formats the best

To support another database format the best, the options should be examined in the following order of priority:

1) Can MDAC be used: does an OLE DB provider exist for the database format? — See “MDAC and UDL files” on page 99

2) If not: Is there a native driver in BDE that can be used? (If BDE is installed). — See “Creating BDE database aliases” on page 103

3) If not: Is there an ODBC driver in MDAC?

4) If not: Is there an ODBC driver in BDE?

The client software must be installed

In order to communicate with another database format, the client software for this other database must be installed.

This is not necessary for Microsoft Access, dBase and MS-SQL.

— If you have any difficulties, please contact your local dealer.

MDAC AND UDL FILES

This chapter gives a brief introduction to creating UDL files, which makes it possible to work with other database formats than Microsoft Access and dBase in PC|SCHEMATIC.

THIS CHAPTER DESCRIBES

— WHAT IS MDAC? ..... (P. 99)
— WHAT ARE UDL FILES? ..... (P. 100)
— CREATING AN UDL FILE ..... (P. 100)
— USING ANOTHER DATABASE FORMAT VIA AN UDL FILE ..... (P. 101)
— INSTALLING MDAC ..... (P. 102)

— It is also possible to work with other database formats via ***Borland Database Engine***. Read more about this in “Creating BDE database aliases” on page 103.

WHAT IS MDAC?

MDAC is a database engine, which makes it possible to work with different database formats in PC|SCHEMATIC. When you wish to use MDAC, you have to create an UDL file, as described in “What are UDL files?” on page 100 and “Creating an UDL file” on page 100.

GET MORE INFORMATION ABOUT MDAC ETC.

MDAC IS SHORT FOR MICROSOFT DATA ACCESS COMPONENTS.

YOU CAN GET MORE INFORMATION ON MDAC ON MICROSOFTS HOMEPAGE AT WWW.MICROSOFT.COM/DATA/PRODINFO.HTM.

THERE YOU CAN ALSO READ ABOUT ADO, ODBC AND MUCH MORE.

— If MDAC is not installed on your PC, please refer to “Installing Microsoft Data Access Components” on page 102.

What are UDL files?

When MDAC is installed on your PC — see “What is MDAC?” on page 99 — you have the option of creating the so-called UDL files, which enables you to work in other database formats than Microsoft Access and dBase from PC|SCHEMATIC.

The UDL file contains information to MDAC about how the connection to the database format in question is handled.

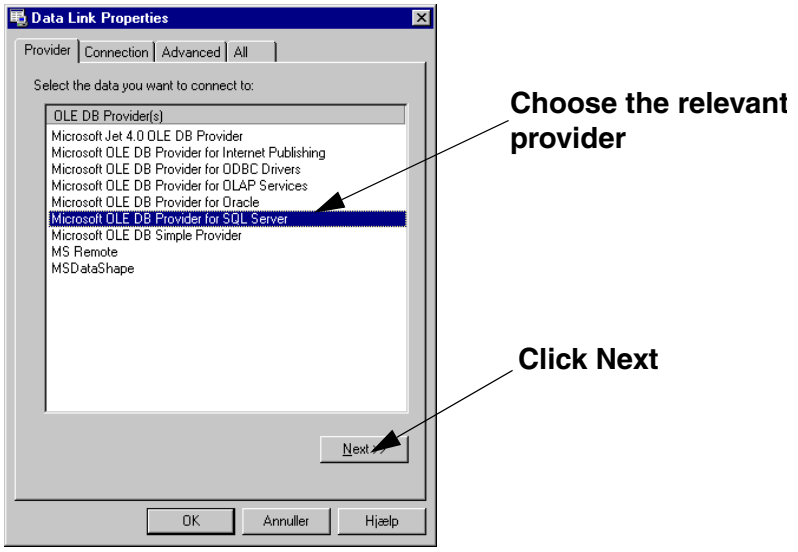
What does UDL stand for?

UDL is short for Universal Data Link (a Microsoft format). UDL files can be created from the **PC|SCHEMATIC Database** program, as described in “Creating an UDL file” on page 100.

Creating an UDL file

To create an UDL file, first choose **File => New**, and you enter the following dialog box:

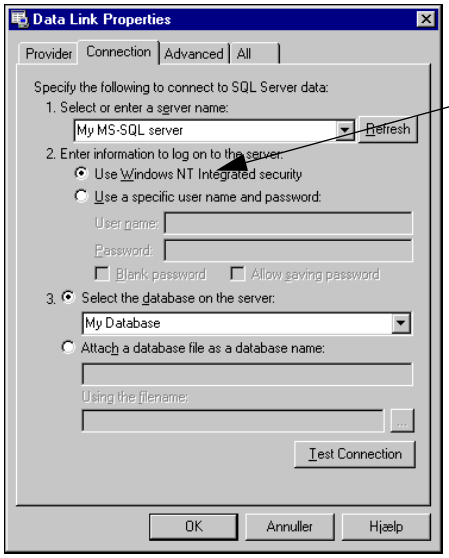
Choose ***Microsoft Data Link (udl file)*** and click **Next**. You then enter this dialog box:



Here you choose the relevant provider. If no OLE DB provider exists for the database format, please refer to “How to support other database formats the best” on page 98.

— Please note that the creation of UDL files is handled in Microsoft dialog boxes. The contents and language in these dialog boxes therefore depends on the installations on the individual PC's.

Click **Next** when you have selected a provider:



Click here to apply your ordinary Windows password

On the **Connection** tab you enter the necessary information. If you need to specify other information on the other tabs, do this also.

Click **OK**. You then enter the dialog box **Save**, where you assign a name to the UDL file and click on **Save**.

Editing the properties of an UDL file

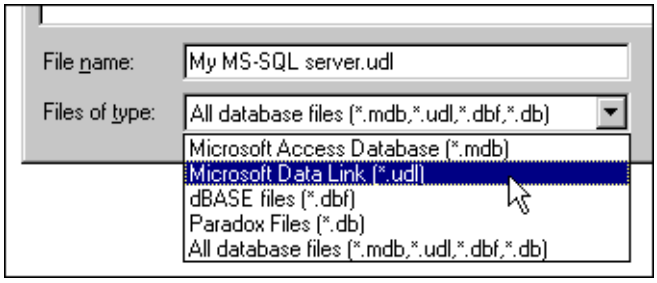
If you later need to change the properties of a created UDL file, just choose **File => UDL properties**. You then get the option to choose which UDL file to edit.

UDL FILES CAN BE COPIED TO OTHER WORKSTATIONS

UDL FILES CAN BE COPIED TO OTHER WORKSTATIONS — AS OPPOSED TO BDE FILES, WHICH MUST BE CONFIGURATED FOR EACH PC INDIVIDUALLY.

USING ANOTHER DATABASE FORMAT VIA AN UDL FILE

To use another database format via an UDL file, just choose **File => Open**:



Now choose the file type ***Microsoft Data Link (*.udl)***, click on the relevant UDL file, and click on **Open**.

You then enter the ordinary **Select table** dialog box, where you open the table you wish to work with.

Installing Microsoft Data Access Components

If you have installed Windows 2000/ME (or later), Microsoft Office 2000 (or later) or Internet Explorer 5.0 (or later), ***Microsoft Data Access components*** (MDAC) is probably already installed on your PC.

If MDAC is not installed on your PC, you get the option to do so during the installation of **PC|SCHEMATIC Automation**.

If MDAC is not installed on your PC, do the following:

1) Insert the **PC|SCHEMATIC Automation** CD, and wait a few seconds

2) The Installation menu is displayed: Choose **Support programs and drivers**

3) Choose **Install Microsoft Data Access Components**

4) MDAC is being installed

5) Restart your PC

MDAC is now installed on your PC.

— If you have any difficulties, please contact your local dealer.

CREATING BDE DATABASE ALIASES

THIS CHAPTER DESCRIBES

— WHAT IS A BDE DATABASE ALIAS? ..... (P. 103)
— CREATING A BDE ALIAS ..... (P. 103)
— OPENING A TABLE VIA A BDE DATABASE ALIAS ..... (P. 105)
— INSTALLING BORLAND DATABASE ENGINE ..... (P. 107)

PREFER MDAC IF POSSIBLE

IN MANY CASES IT IS BETTER TO USE MDAC RATHER THAN BDE FOR WORKING IN OTHER DATABASE FORMATS.

— READ "HOW TO SUPPORT OTHER DATABASE FORMATS THE BEST" ON PAGE 98 TO SEE WHAT IS MOST APPROPRIATE IN YOUR SITUATION.

WHAT IS A BDE DATABASE ALIAS?

A BDE alias can be used when you wish to work in a database created in other database programs than Microsoft Access or dBase. However, this requires detailed knowledge about the settings of this other database program.

The use of another database can take place via a so-called BDE database *alias*, which makes it possible to collect data from other database systems.

If no appropriate BDE alias is created, you have to create one yourself.

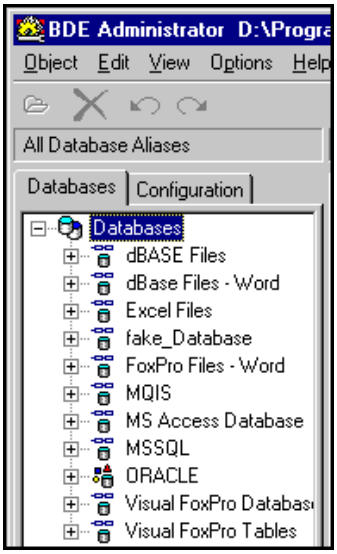
CREATING A BDE ALIAS

To create a BDE alias, you must first start the program **BDE Administrator**.

This is done by choosing **File => Open Database Alias**, and then click on **BDE Administrator**.

— If the program is not installed on your PC, please read "Installing Borland Database Engine" on page 107.

When the program is started, click on the **Databases** tab:



Here you can see which database aliases have been created.

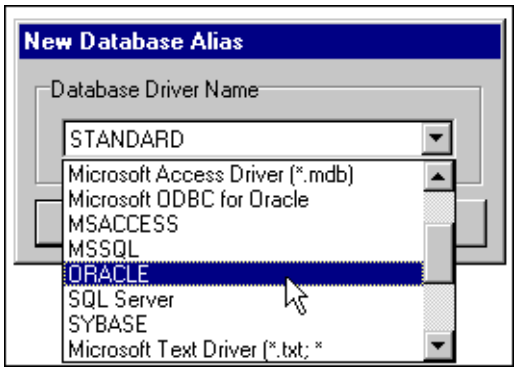
In front of each alias a small symbol is placed, which displays whether the alias is created with a *native* driver or with an *ODBC* driver.

Native driver

ODBC driver

To create a new database alias, you must choose **Object => New**.

You then enter this dialog box:



Here you choose a driver that matches the database program you wish to use. Please note that native drivers are written in upper-case letters on the displayed list.

CHOOSE A NATIVE DRIVER IF POSSIBLE

IF THERE EXISTS A NATIVE DRIVER FOR THE DATABASE PROGRAM YOU WISH TO APPLY, YOU SHOULD SELECT THIS NATIVE DRIVER. THIS IS BECAUSE IT WORKS FASTER THAN A CORRESPONDING ODBC INTERFACE.

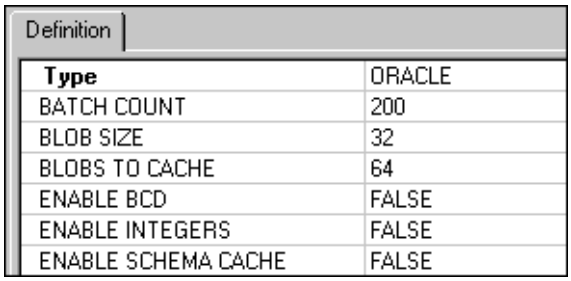
Click **OK**.

Hereby a new database alias is created.

To change the name you must double-click on the new alias in the left-hand side of the screen, change the name, and press <Enter>. This name will be displayed on the alias list in the **PC|SCHEMATIC Database** program.

Changing the driver settings

When you have chosen an alias in the left-hand side of the screen, you get the option of correcting the various settings for the chosen driver.



When you double-click on an alias, you are informed whether its settings can be applied or not.

DETAILED KNOWLEDGE ABOUT THE DATABASE SYSTEM REQUIRED

TO MAKE THESE CHANGES REQUIRES DETAILED KNOWLEDGE ABOUT THE CHOSEN DATABASE SYSTEM.

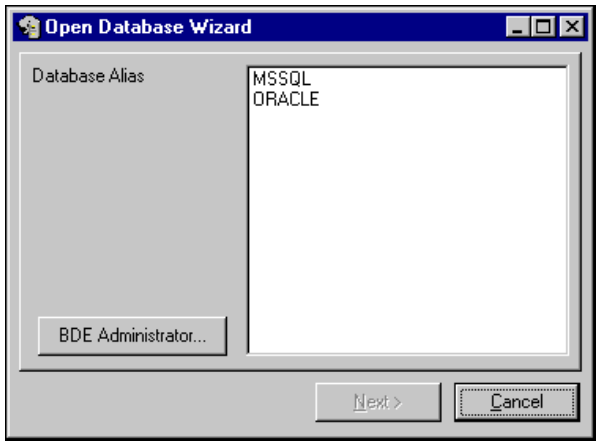
PLEASE CONTACT YOUR LOCAL HOT-LINE IF YOU HAVE ANY DIFFICULTIES.

To use the alias, click on it and choose **Object => Apply**.

Now close the **BDE Administrator** program by choosing **Object => Exit**.

OPENING A DATABASE VIA A BDE DATABASE ALIAS

To open a database via a BDE database alias, choose **File => Open Database Alias**, and you enter this dialog box:

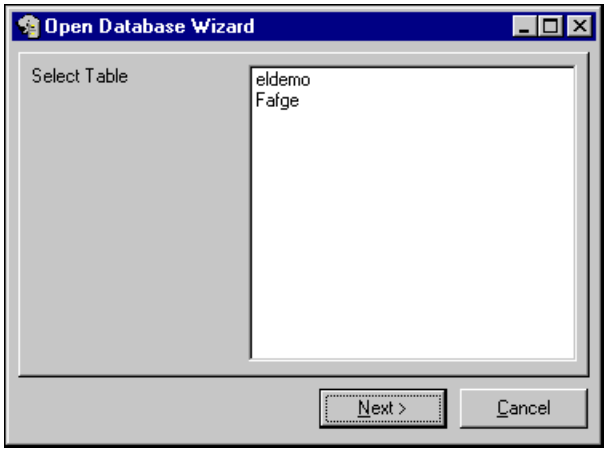


The options in this dialog box depends on which database aliases have been created on your PC.

BDE DATABASE ALIASES CAN *NOT* BE COPIED

BDE DATABASE ALIASES MUST BE CREATED ON EACH INDIVIDUAL PC USING THE ALIASES.

Choose the alias you wish to use, and choose **Next**:



Now choose the desired table, and click **Next**.

The table in the database have now been opened via the chosen BDE alias.

Installing Borland Database Engine

If ***Borland Database Engine*** is not installed on your PC, do the following:

1) Insert the **PC|SCHEMATIC Automation** CD, and wait a few seconds

2) The Installation menu is displayed: Choose **Support programs and drivers**

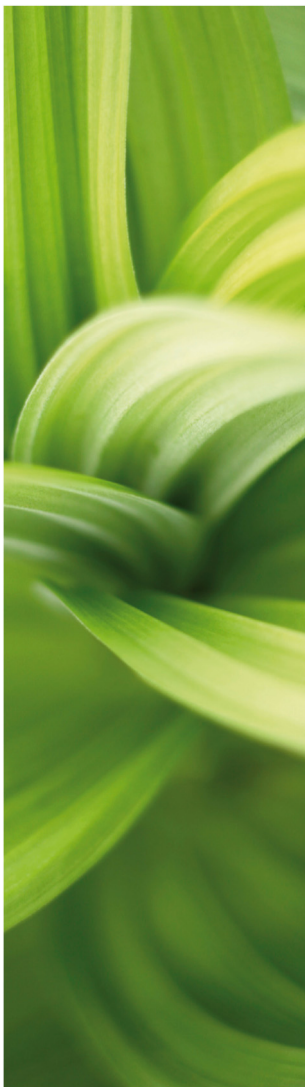
3) Choose **Install Borland Database Engine**

4) Follow the installation procedure until ***Borland Database Engine*** is installed

5) Restart your PC

Borland Database Engine is now installed on your PC.

— If you have any difficulties, please contact your local dealer.



Index

PC SCHEMATIC

INDEX

A

Access
- choose version . . . 40
- create Microsoft Access database . . . 39
- create Microsoft Access table . . . 42

Access denied . . . 49, 54
Activate, button . . . 26
Add . . . 49
- data field to database . . . 49
- menuponit . . . 57

Add new records . . . 64
ADO . . . 99
Append
- data field to database . . . 49

Automatic update . . . 30
Automation . . . 7

B

BDE
- BDE Administrator . . . 103
- BDE database alias
  - can not be copied . . . 106
- create new BDE database alias . . . 103
- install Borland Database Engine . . . 107
- open database via BDE alias . . . 105

Begins with, operator . . . 26
Bottom, button . . . 34
Buttons
- appearance . . . 20

C

Change
- data fields . . . 48
- records . . . 28
- table menus . . . 53

Change records, filtered import . . . 74
Character set
- DOS (PC-8) . . . 70
- Windows (ANSI) . . . 70

Client software . . . 98
Comma sep. . . . 70
Comma separated file . . . 83
- import . . . 69
  - import and field names . . . 71
- import with field names . . . 71
- open . . . 69
- open zip file . . . 70
- what is a comma separated file . . . 83

Component data
- edit . . . 28

Component vendor
- import database . . . 64
- import database from . . . 63
- separate by . . . 27

Connect table menu to table . . . 59
Contents . . . 19
Copy
- data field or record . . . 29
- record during RecordView . . . 29
- table menu . . . 58

Copy Table . . . 43, 45
Copy Table Structure . . . 43, 44
Copy, button . . . 29
Create
- BDE alias . . . 103
- database . . . 39
- export file . . . 85
- table . . . 42
- table menu . . . 51
- UDL file . . . 100

Create as Zip file . . . 85
Cut, button . . . 30

D

Data base menu . . . 19
Data fields
- about data fields . . . 13
- change . . . 48
- copy . . . 29
- delete . . . 49
- index . . . 50
- naming . . . 40
- remarks . . . 30
- replace contents during search . . . 33
- search . . . 32
- select . . . 28
- transfer . . . 66
- transfer all . . . 66
- update property changes . . . 49

Data filter . . . 26
- disable . . . 27

Data filter, button . . . 26, 27
Data record(s), tab . . . 36
Database
- about databases . . . 13
- change driver settings . . . 105
- choose table in active database . . . 19
- copy table from existing database . . . 41
- create . . . 39
- create table menu . . . 51
- dBase format and databases . . . 41
- denied access . . . 54
- in a network . . . 15, 30
- menu . . . 14
- menu for table . . . 24
- menu structure . . . 24
- naming . . . 40
- one at the time . . . 15

open . . . 16
open via BDE alias . . . 105
search in . . . 32
which records are displayed . . . 23
the consequence of displaying records 23
Database formats
what does PCschematic support . . . 97
best support method . . . 98
Database menu
now called Table menu . . . 51
dBase
create dBase table . . . 47
dBase format and databases . . . 41
dBase Table (dbf file) . . . 40
dbf, file type . . . 92
Define New Table . . . 43
Define new table . . . 43
Delete . . . 19
data field . . . 49
menupoint . . . 57
record . . . 29
table . . . 47
Delete records, filtered import . . . 75
Deleted records . . . 30
Delete-mark
remove . . . 29
Delimiter
importing a comma spearated file . . . 71
Denied access . . . 49, 54
Description . . . 55
Direct loading into another program . . . 94
Disable
data filter . . . 27
Displaying the contents of the database . . . 23
Documentation of key- and power control circuits 7
DOS (PC-8) . . . 70
DOS, file type . . . 85

E

Edit
menupoint . . . 55
records . . . 28
table menus . . . 53
Edit =>
Sort by . . . 25
Electronic circuits . . . 7
Electronics . . . 7
Elevator . . . 19
Empty table . . . 47
Excel
export Microsoft Excel documents . . . 87
import Microsoft Excel documents . . . 78
Execute SQL button . . . 36
Exit and Return SQL button . . . 36
Export
create export file . . . 85
direct loading into another program . . . 94
give away a database . . . 91
dBase tables . . . 92
Microsoft Access database . . . 91
Microsoft Excel documents . . . 87
to other programs . . . 83
Export field names as first record . . . 85

F

FAFGE groups . . . 60
Field name . . . 49
about field names . . . 13
sort by . . . 25
Field names
importing a comma separated file . . . 71
File
quotation marks in . . . 85
File =>
Delete . . . 47
Empty . . . 47
Exit . . . 58
Export . . . 23, 85
Import . . . 69, 73
New . . . 39
Open . . . 16
Open Database Alias . . . 103, 105
Pack . . . 30
Properties . . . 48
File format . . . 70
File type
dbf . . . 92
DOS (PC-8) . . . 70
DOS/Windows . . . 85
idb . . . 91, 92
mdb . . . 91
mdx . . . 92
udl . . . 93
Windows (ANSI) . . . 70
Filtered import . . . 73
Filtrering . . . 26
Find and replace . . . 33
Find, button . . . 32
First record contains field names . . . 71
Flat buttons . . . 20
Format, file . . . 70
Formats
which database formats PCschematic supports 97
best support method . . . 98
Functions =>
Data Filter . . . 26, 27
SQL query . . . 36

G

Give away a database . . . 91
dBase tables . . . 92
Microsoft Access database . . . 91

I

idb, file type . . . 91, 92
Import . . . 67

comma separated file . . . 69
component vendor database . . . 64
database from component vendor . . . 63
filtered import . . . 73
from Louis Poulsen . . . 73
from Nordisk Solar . . . 73
Microsoft Excel documents . . . 78
Increase search rate . . . 50
Index field . . . 50
Index file . . . 50
Indexing . . . 25
Insert . . . 19, 49
menuponit . . . 57
new record . . . 29
Insert =>
Tablename . . . 36
Insert new record, button . . . 29
Insert, button . . . 29
Is repeated, operator . . . 27

K

Key diagrams . . . 7

L

Length . . . 49
Louis Poulsen, import from . . . 73
LP varedata, file type . . . 73

M

Manual
survey . . . 8
typography in . . . 9
Match Case . . . 32
Match whole field . . . 33
MDAC
installing Microsoft Data Access Components 102
more information . . . 99
what is MDAC . . . 99
mdb, file type . . . 91
mdx, file type . . . 92
Menu . . . 19, 24, 25
create table menu . . . 51
table menu . . . 14
Menu =>
Edit . . . 53, 55
Enabled . . . 17, 24
Properties . . . 59
Restore . . . 25
Show hidden records . . . 25
Show used . . . 25
Menu bar . . . 18
Menu editor . . . 55
Menu settings . . . 60
Menu structure . . . 24
Menulink . . . 55
Menupoint
add/insert . . . 57
delete . . . 57
edit . . . 55
place in menu . . . 55
Message, tab . . . 36
Microsoft Access
choose version . . . 40
create Microsoft Access database . . . 39
create Microsoft Access table . . . 42
Microsoft Access Database (mdb file) . . . 39, 40
Microsoft Access Table . . . 39
Microsoft Data Link (udl file) . . . 40, 100
Microsoft Excel
export Microsoft Excel documents . . . 87
import Microsoft Excel documents . . . 78
Modules . . . 7
More databases open . . . 15
MS-SQL . . . 97

N

Native driver . . . 104
Network
database in . . . 15, 30
New SQL button . . . 36
Next, button . . . 34
Nordisk Solar, import from . . . 73
NSC varedata, file type . . . 73

O

Object =>
Apply . . . 105
Exit . . . 105
New . . . 104
ODBC . . . 97, 99
ODBC interface . . . 104
One database at the time . . . 15
Open . . . 16
a database . . . 16
comma sepoarated file . . . 69
zip file . . . 70
Open menu . . . 59
Open SQL file button . . . 36
Operators . . . 26
Options =>
Flat Buttons . . . 20
Quote Tablenames . . . 36
Oracle . . . 97
Other programs . . . 94
export to . . . 83

P

Pack . . . 30
PC|SCHEMATIC A/S . . . 7
PCB layout . . . 7
Power control diagrams . . . 7
Preface . . . 7
Preview output format . . . 86
Previous, button . . . 34
Primary key
check table . . . 44
Primary keys . . . 50

What is a primary key? 50
when creating Access tables form dBase tables 50
Program buttons
appearance 20
Program tool bar 19
Programs, others 94

Q

Quotation marks in files 85

R

Records
about records 13
copy 29
delete 29
deleted 30
edit 28
go to record no. 34
insert new 29
remove deleted 30
scrolling between 34
select 28
that does not fit in menu structure 25
update 64
which records are displayed 23
the consequence of displaying records 23
RecordView 24
copy record 29
RecordView, button 19
Refresh 19
Remarks, data field 30
Remove deleted records 30
Remove delete-mark 29
Replace by 33
Replace text 33
Return to the component table 58
Run 64

S

Save SQL file button 36
Scrolling 19
between records 34
Search
in database 32
on data fields 32
speed up 50
Search =>
Find 32
Find next 33
Go to record no. 34
Replace 33
Search direction 32
Searchfield 32
Sections
in the manual 8
Select
records or data fields 28
Separate by component vendor 27
Separator 85
importing a comma spearated file 71
Settings =>
Database 49
Several databases open 15
Sort
data fields 50
Sort by field name 25
Speed up searching 50
SQL 35
execute SQL queries 36
options 35
SQL query
misinterpretations 36
save or open 36
what is SQL? 35
Start 64
Start filtrering 26
Start PCschematic Database 16
Survey of the manual 8

T

Table
about tables 13
access denied 54
choose table in active database 19
connect table menu to 59
copy 45
check for primary key 44
copy table structure 44
create 42
create dBase table 47
create Microsoft Access table 42
define new table 43
delete data field in 49
empty 47
in a network 15, 30
naming 40
properties
denied access to 49
search in 32
table menu 24
update property changes 49
which records are displayed 23
the consequence of displaying records 23
working in a table 15
Table menu 14, 24
connect table to 59
copy 58
create 51
edit 53
no access 54
previously called Database menus 51
Table Menu Editor 55
add/insert menupoint 57
delete menupoint 57
edit menupoint 55
TABLEGROUP 60

TableView 24
Text file
    import 69
    import and field names 71
    open zip file 70
Tool bar 19
Tools =>
    Database in PCschematic 16
Top, button 34
Transfer
    all data fields 66
    data for PCB layout 7
    one data field 66
Typography in the manual 9

U

UDL files
    can be copied to other workstations 101
    create 100
    editing the properties of 101
    what are UDL files 100
    What does UDL stand for 100
udl, file type 93
Update 19, 30
    automatic 30
    table property changes 49
Update existing records 64
Update records, filtered import 75
User-defined modules 7

V

Vendor
    separate by 27
View =>
    Deleted records 30
    RecordView 24
    TableView 24

W

What is a comma separated file 83
Windows
    file type 85
Windows (ANSI) 70
Work area 18
Working in a table 15

Z

Zip file
    export 85
    open 70

Å

Aa, button 26