

# unidb and DH

## User Guide

Version 1.0

server2/unidb/unidb.html

Hilfe Einstellungen abmelden neues Dokument  
manuelle Eingabe gel. Dok. zeigen  
Administration

Inhalt

- Administration
- Aktien
- aktuell
- Benutzerhandbuch
- Bilder
- DH & unidb
- Formulare
- Sicherungen

suchen

Formular Tabellenansicht speichern 1 / 83 Filter: Filter erstellen Sortierung

Status

Notizen\_ID 2 Aufwand in h 3 erledigt in Bearbeitung zurückgestellt offen

Priorität

Betreff Formular - Feldeigenschaften - editierbar - aktiviert - sichtbar

Datenfelder müssen die Eigenschaften editierbar, aktiviert und sichtbar bekommen. Beispiel: Notizen\_ID in diesem Formular sollte nur sichtbar, aber nicht editierbar sein.

Dokument Version Abfrage Datenbank Tabellen Abfragetyp Beschriftung

User\_Tags

User\_Akt

Abfrage

SELECT 'User\_Tags'.Point\_Path, 'User\_Tags'.Tagname, 'User\_akt'.Point\_ID, 'User\_akt'.Timestamp, 'U' 'T25' AND 'User\_akt'.Timestamp > '2020-01-01' LIMIT 6,100;

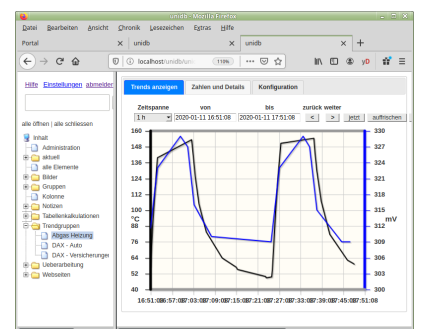
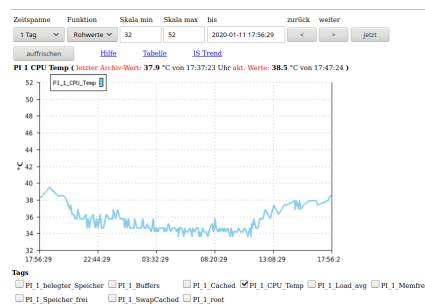
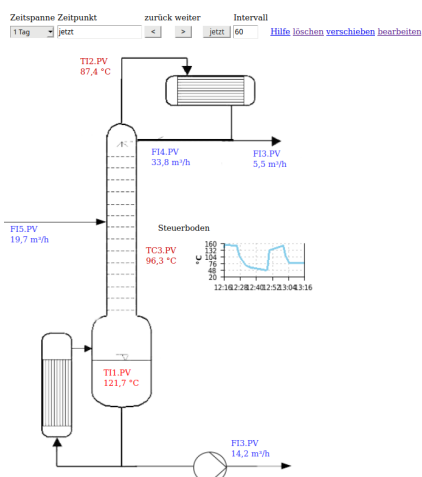
Abfrageergebnis

Field	Point_Path	Tagname	Point_ID	Timestamp	Value
Function					
Alias					
Tabelle	User_Tags	User_Tags	User_akt	User_akt	User_akt
Sortierung					
gruppieren					
anzeigen					
Kriterium	like 'T25'			>'2020-01-01'	
Kriterium					

Dokument	Spalten & Zeilen	Format	Rahmen	DH Funktionen	Hilfe
	(A)	(B)	(C)	(D)	
0					
1	/L2				
2	Rohwerte in mm	Zeitstempel	Unix Zeitstempel		
3	1406	2019-03-03 16:00:00	1551625200		
4	1397	2019-03-09 13:00:00	1552132800	83097	
5	1371	2019-03-24 11:15:22	1553422522	199181	
6	1363	2019-03-31 11:07:28	1554023248	84060	
7	1344	2019-04-20 15:45:15	1555767915	164401	
8	1327	2019-05-01 19:50:23	1556733023	67140	
9	1315	2019-05-11 14:45:58	1557578758	78518	
10	1305	2019-05-18 19:47:49	1558201669	57211	

Hilfe Gruppe bearbeiten löschen verschieben

PI_1_belegter_Speicher	benötigter RAM	2020-01-11 17:47:24	344.3	MByte
PI_1_Buffers	Puffer	2020-01-11 17:47:24	161692	KByte
PI_1_Cached	Cached	2020-01-11 17:47:24	394728	KByte
PI_1_CPU_Temp	PI 1 CPU Temp	2020-01-11 17:47:24	38.5	°C
PI_1_Load_avg	PI 1 Auslastung	2020-01-11 17:47:24	9	%
PI_1_Memfree	freier Speicher	2020-01-11 17:47:24	39324	KByte
PI_1_Speicher_frei	freier Speicher	2020-01-11 17:47:24	62.8	%
PI_1_SwapCached	Auslagerungsspeicher	2020-01-11 17:47:24	72	KByte
PI_1_root	Server Root Partition	2020-01-11 17:47:24	16.855	GByte



## table of Contents

Terms and abbreviations .....	2
Formatting used .....	2
1. unidb .....	3
1.1. Introduction .....	4
1.2. Settings .....	5
1.2.1. personal Settings .....	5
1.3. Documents .....	6
1.3.1. create new document .....	6
1.3.2. Note .....	7
1.3.3. Web site .....	8
1.3.4. Spreadsheet .....	10
1.3.5. Query .....	10
1.3.6. Form .....	14
1.3.7. Report .....	18
1.3.8. Link .....	18
1.4. Permissions .....	19
2. DH .....	21
2.1. Introduction .....	21
2.2. DH documents .....	22
2.2.1. Document type drawing .....	23
2.2.2. Group .....	25
2.2.3. Trend Group .....	27
2.2.4. export .....	33
2.2.5. Spreadsheet .....	33
2.3. Details .....	38
2.3.1. Tags, Points, Paths and permissions .....	38
2.3.2. Point and tag configuration .....	39
2.3.2. Archiving of values .....	41
2.3.3. Functions .....	41
2.3.4. Working with trends .....	42

## Terms and abbreviations

DataHistorian	Database application which continuously collects data in the format data point - time stamp - value and makes this available again in a processed form.
DH	Figreviation for DataHistorian
TimeSeriesDatabase	An alternative name for a DataHistorian.
DB	Figreviation for database.
Tag	Data point in a DataHistorian
Point	Another name for a tag / data point.
Point_ID	Unique number of a tag, comparable to a serial number.
WYSIWYG	<b>What You See Is What You Get</b> (What you see is what you get.) There are two types of HTML editors. The simplest way is the one where you can see the HTML code and have to write it yourself. The second type is the type that is easier for the user. The HTML code remains hidden and instead you can see the fully formatted text (WYSIWYG).

## Formatting used

---

<b>bold font</b>	Name of a control element (switch, text field, ...)
<i>italic</i>	Names of properties, etc.
highlighted in color	You should pay a little more attention to a text with a colored background.

# 1. unidb

The name *unidb* stands for universal *database*. Universal because it is only a database application which in turn accesses MariaDB databases. It is therefore comparable to applications such as MS Access or LibreOffice Base.

The difference is that it is web-based and no software needs to be installed on the user's computer. All you need is any browser that you can find on every computer anyway.

In addition to the range of functions provided by the database applications mentioned above, the *unidb* has the other document *types* note, Web site and spreadsheet. A tree structure, through which all documents can be easily selected, ensures order and a good overview.

The user interface is automatically adapted for mobile devices as these are recognized. An "app" is therefore superfluous. You don't have to have an "app" for everything, do you?

The *unidb* was originally developed as a user interface for the *DH* (*DataHistorian*). Therefore, it has several document types, which reproduce the data of the *DH* in different ways. You may now be wondering what a DataHistorian is. A DataHistorian is, in principle, a very simple thing. It collects all the numbers that can be provided with a time stamp and makes them available to the user in a processed form. All combinations of number and time stamp must be assigned to a data point. This *data point* represents, for example, a place where a temperature is regularly measured.

Here's an example:

Measurement (data point)	time stamp	value
Exposure to sunlight	2021-04-23 12:05:36	897
PV system power	2021-04-23 12:05:36	6.34
PV system efficiency	2021-04-23 12:06:03	13.65

The principle is simple, but the devil is in the details.

In the second chapter you will learn more about the *DH*. Here we limit ourselves to the *unidb* for the time being.

## 1.1. Introduction

As with any web-based application, you must also register at the *unidb*. When you call up the address of the application in the browser, the login page appears first.

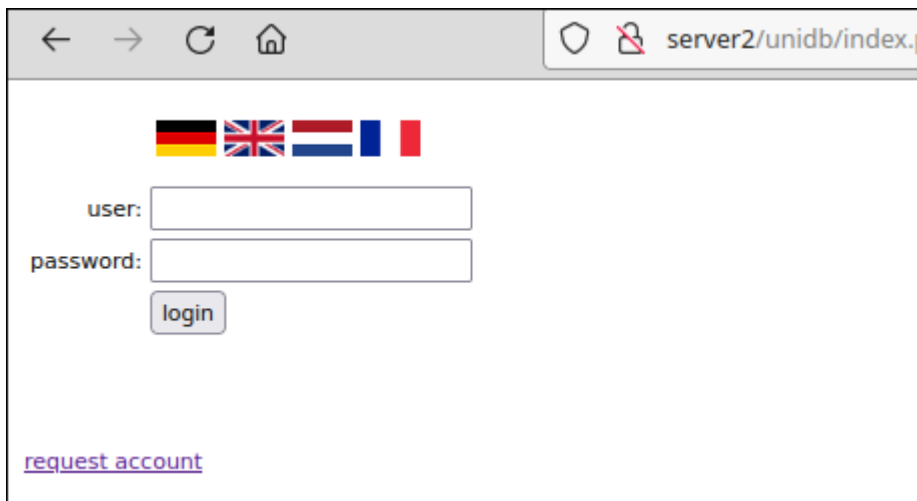


Fig 1: Registration

The application is available in German, English, Dutch and French. To switch the language, simply click on one of the flags. You can later set your preferred language permanently in your personal Settings. After you have successfully logged in, you will see a page on the screen which is divided into two areas. On the left there are various links at the top. The example below shows an administrator's menu. Most of the links should not require any explanation. **manual entry** is part of the DH and enables values to be entered manually. Show yellow **doc.** rebuilds the tree structure and shows entries that contain a deletion mark in pale font color. This makes it possible to restore accidentally deleted documents. It's like taking a piece of paper out of the trash. The only difference: the document is not created here ;-).

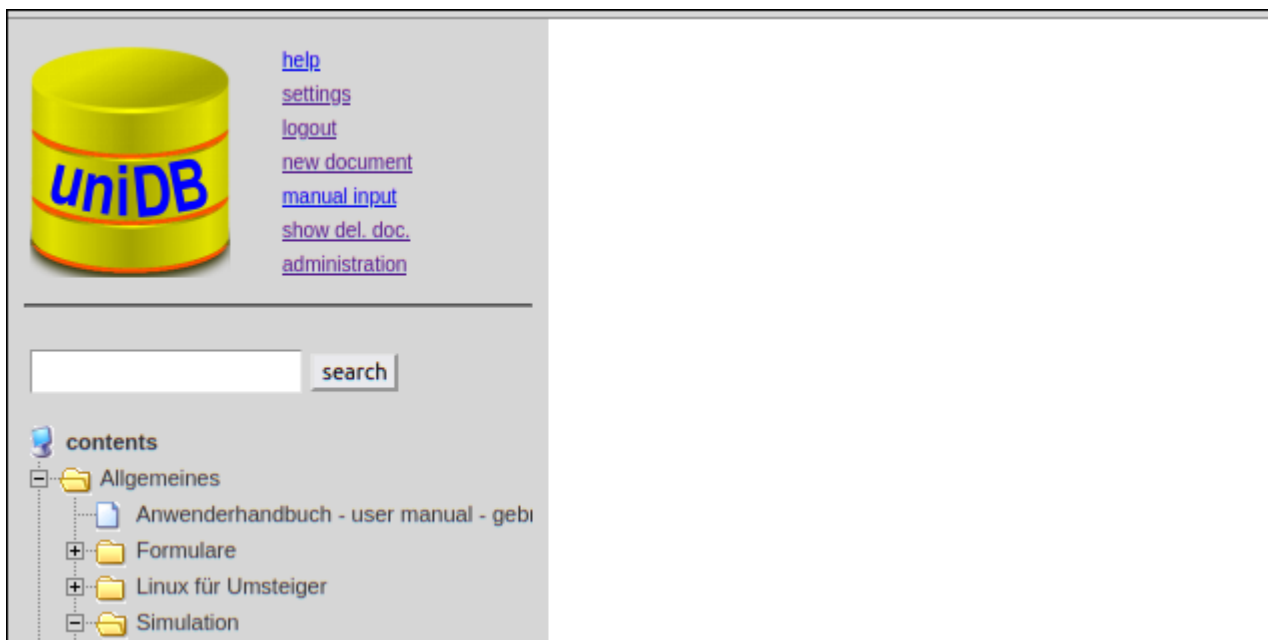


Fig 2: Tree menu on the left, document area on the right

On mobile devices, only the left side is displayed due to the small screen. If you click on an entry in the tree structure, it will be displayed in a new window (tab) instead of on the right-hand side. Each document contains a link in the view for mobile devices, which brings you back to the tab with the tree structure.

A note on the tree menu:

Each document can appear as a branch or folder. As soon as you create a new document and

subordinate it to an existing document, the existing document automatically becomes a folder. The document itself is retained in its entirety.

## 1.2. Settings

Most of the Settings relate to the DH. Only the **personal Settings** relate to the **unidb** and the **DH**.

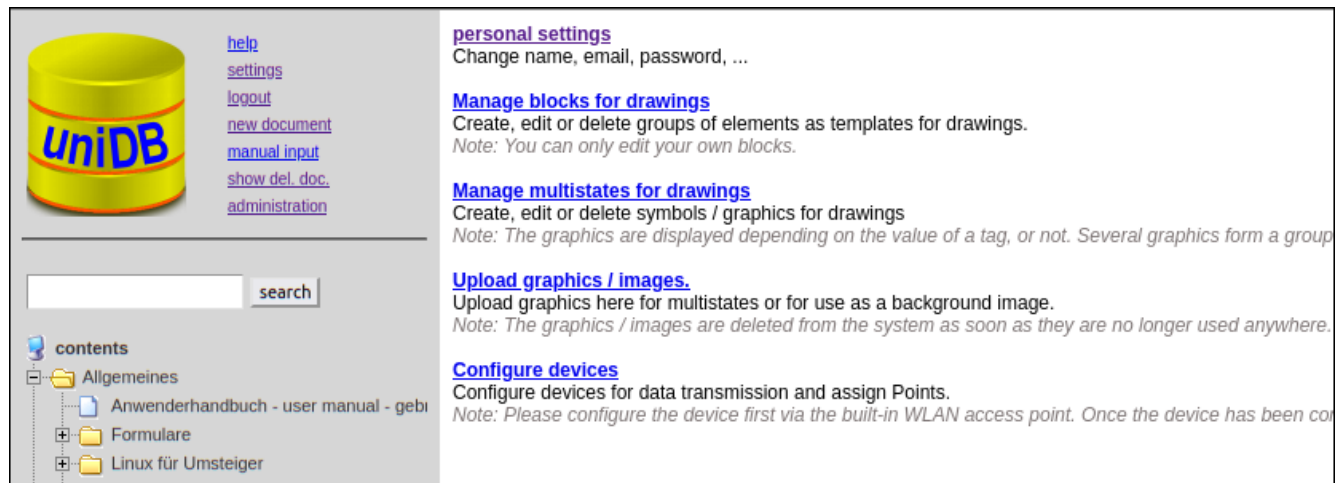


Fig. 3: Settings

### 1.2.1. personal Settings

After logging in for the first time, you may want to make a few personal Settings

You can access the form for your personal Settings via the **Settings** link and then via **personal Settings** on the right-hand side.

After you have changed the preferred language and have accepted your change using the **Save** button, you should call up the application again so that the change is also visible. Usually it is sufficient to press the F5 key.

The display on the screen can be adjusted according to your needs. Find the setting you like the most from the **Theme** drop-down menu. The setting only takes effect after you have reloaded the entire page. The quickest way to do this is with the function key F5.

**Please don't forget to click the **save** button first!**

The screenshot shows the 'personal settings' form. It has a title 'personal settings' at the top. Below it, there are several input fields and a dropdown menu. The 'user name' field contains 'Ralf'. The 'full name' field contains 'Ralf Meier'. The 'preferred language' section shows four flags: German (DE), English (EN), Dutch (NL), and French (FR). The 'EN' option is selected with a green checkmark. The 'e-mail' field contains 'Ralf@MeierSoft.de'. There are two empty fields for 'new password' and 'repeat new password'. The 'theme' dropdown menu is set to 'default'. At the bottom, there is a 'save' button. On the left side of the form, there is a sidebar with a search bar and a 'suchen' button.

Fig 4: personal Settings

## 1.3. Documents

Everything that shows on the right is a *document*.

The *Note* type provides a WYSIWYG editor that can be used to quickly and easily enter a note.

A *Web site* is a document on which any number of containers with HTML content can be accommodated. The containers can be freely positioned and each represent a *note*.

The *spreadsheet* document type is a very simple implementation and therefore only consists of a spreadsheet. This document type is also not intended to replace a conventional spreadsheet. This document type was originally developed in order to use data from the *DH* in *your* own calculations. The spreadsheet therefore also contains a number of special functions and dialogs for querying data from the *DH*.

The Query, form, and report types are document types, such as

You are probably already familiar with it from applications such as MS Access or LibreOffice Base.

That leaves the link type. This is actually not a document, just a reference to an existing document. This is a quick and easy way to create a new structure.

Every document has a header that serves as a menu. On the following pages you will always see the complete header. If you *look at* documents that were created by other users in the *unidb*, you will not be shown the options to **move**, **edit**, **rename** and **delete**. These options are of course available to you in documents that you have created yourself.

### 1.3.1.create new document

The link New Document at the top of the left frame opens a simple dialog on the right, which you can use to select the document type and give the document a name. The name will appear later as an entry in the tree menu on the left.

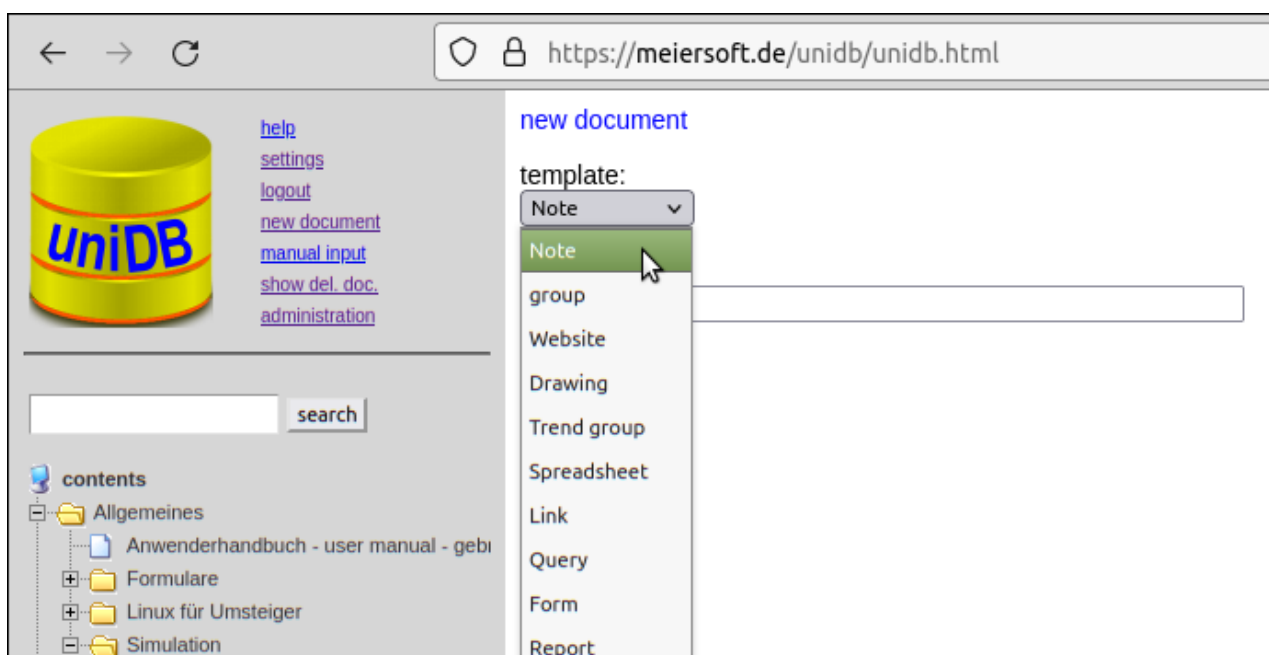


Fig. 5: create a new document - first step: select the document type



Fig. 6: create a new document - second step: name the document

The **next** button now brings us to a tree menu on the right. Here please click on the entry which should be superordinate to your document. In our example we select the branch *Revision*.

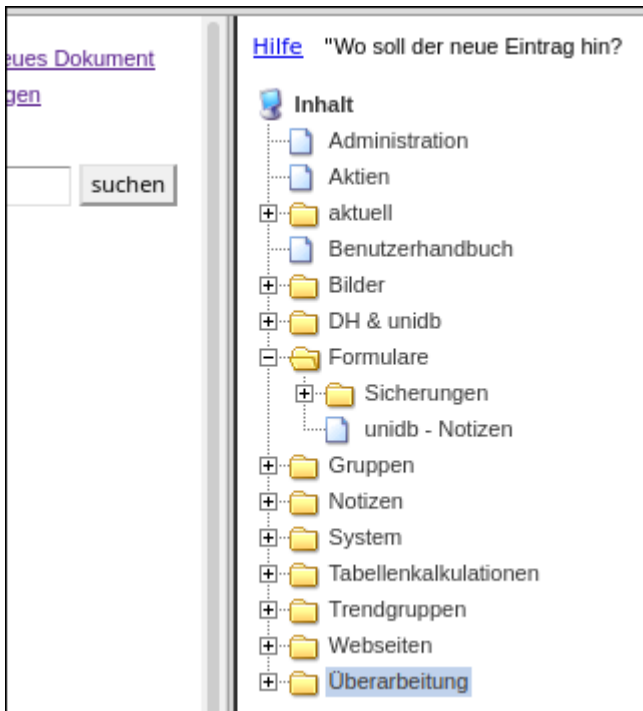


Fig 7: Select the higher-level entry

After selecting the higher-level entry, we are only asked whether the new document should be saved / created, or whether we would prefer to cancel. After that, our document appears on the left-hand side under the *Mustermann* branch.

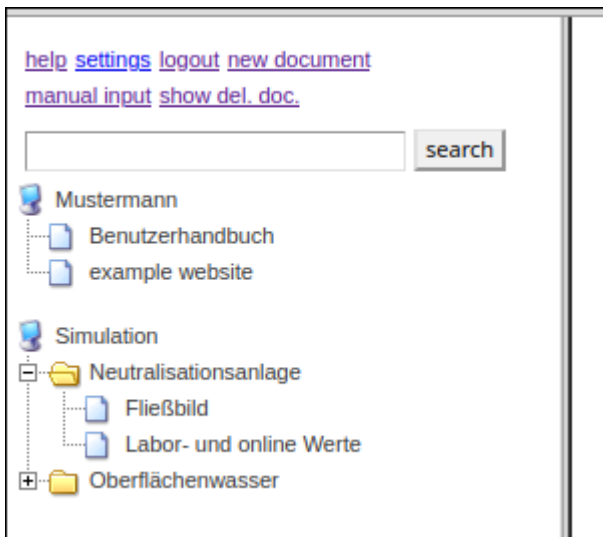


Fig 8: New document under the revision branch



### 1.3.2. Note

We create a new document of the type Note, as in 1.3.1. described. The new note looks like this:



Fig 9: newly created note

The name of the note in big, bold letters. At first there is nothing more to see.

In order to fill the note with content, we now click on the link **edit**. A WYSIWYG editor appears in which you can now write your note.

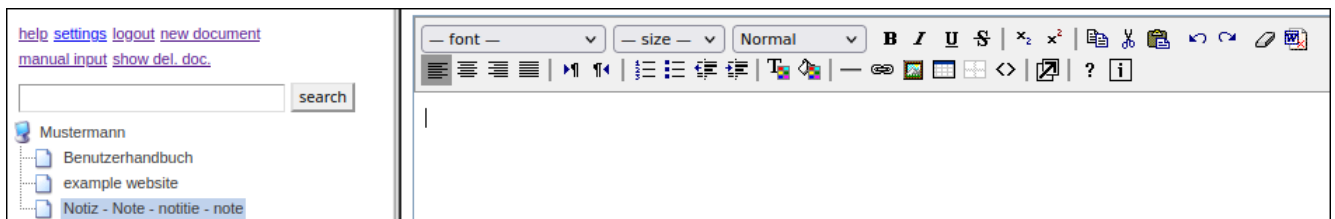


Fig 10: WYSIWYG - Editor

Please do not forget to press the **Save note** button below the editor after you have entered your note. You can also change the name of your note below the editor.

And this is what the finished note looks like when called up from the tree menu:

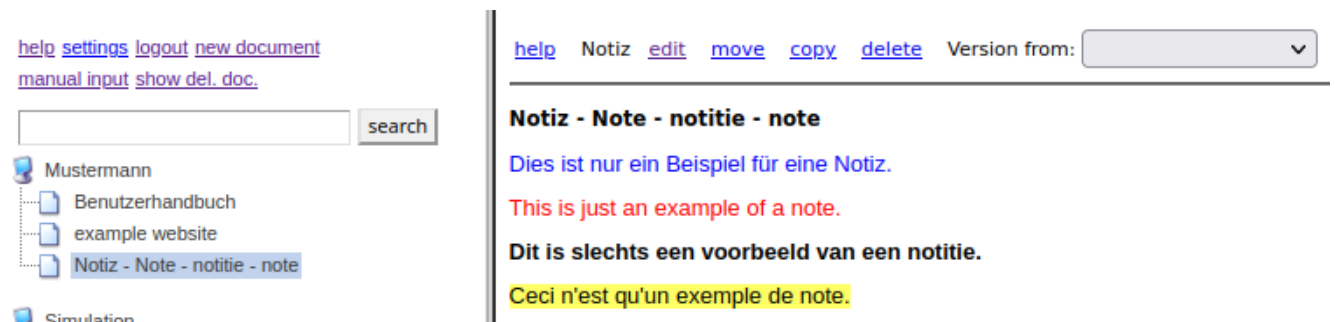


Fig 11: finished note in viewing mode

### 1.3.3. Web site

A Web site fulfills the same purpose as a page in MS-OneNote, or in KDE environments in Baskets. Any number of frames can be created, which in turn can be edited using a WYSIWYG editor. The frames can be changed in size or moved by holding down the left mouse button. They can be placed anywhere. After we have created a new document of the type Web site, it also appears as an empty document on the right-hand side.



Fig 12: new, empty Web site

To fill the new document with content, we click the **edit** link here as well. The Web site is now shown in edit mode.



Fig 13: Web site in edit mode

A new frame is created using the **new element** button. Two objects appear on the page, an empty rectangle and a WYSIWYG editor.

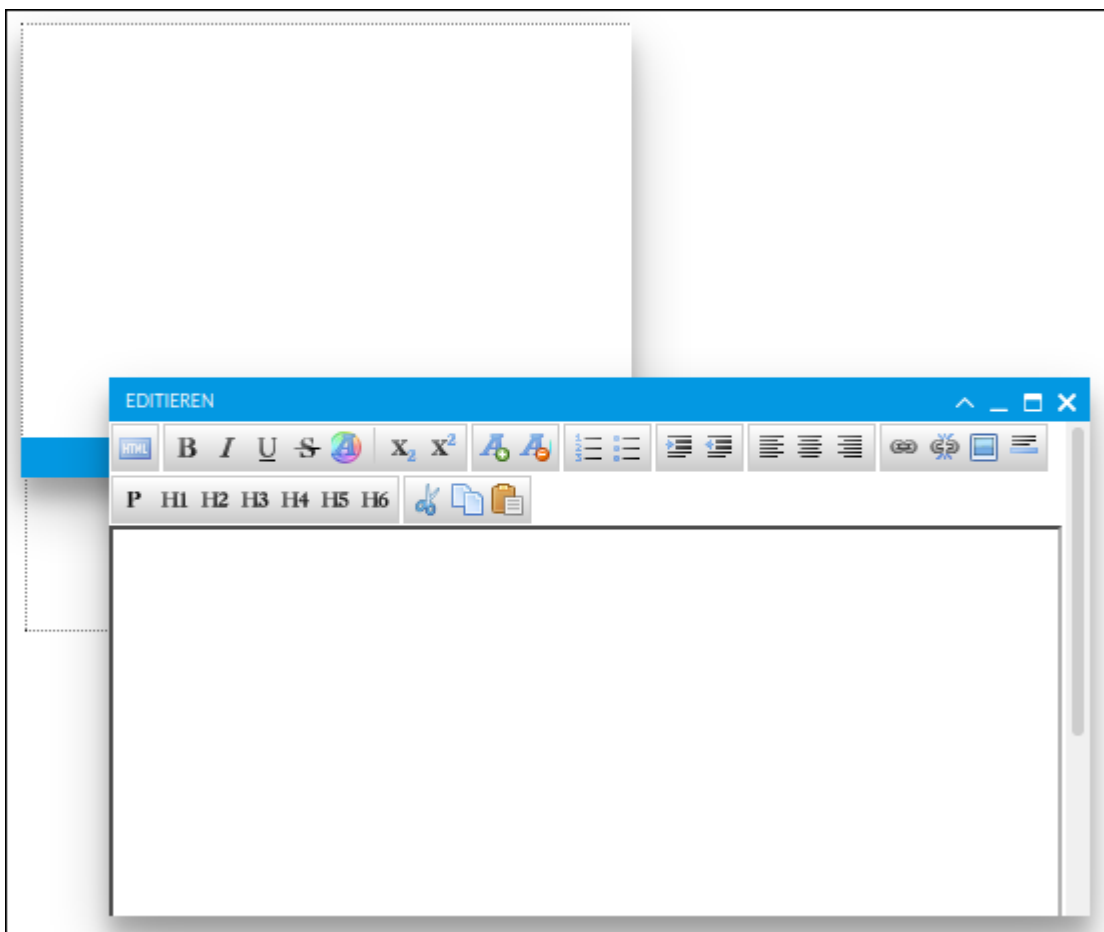


Fig 14: new element

After you have edited the content of the new element in the editor, simply close it. The content from the editor is transferred to the frame. This can now be moved or its size changed. In edit mode, the new element is shown with a dotted frame. The frame will not be visible in viewing mode.

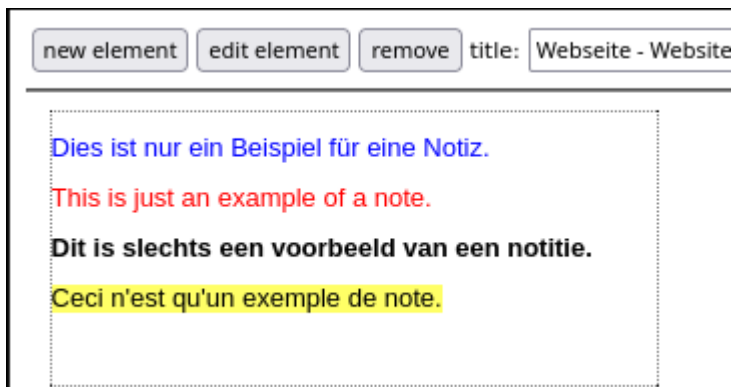


Fig 15: new element

Please do not forget to save your work with the button **Save page**.

Here is the Web site with the new element in viewing mode:

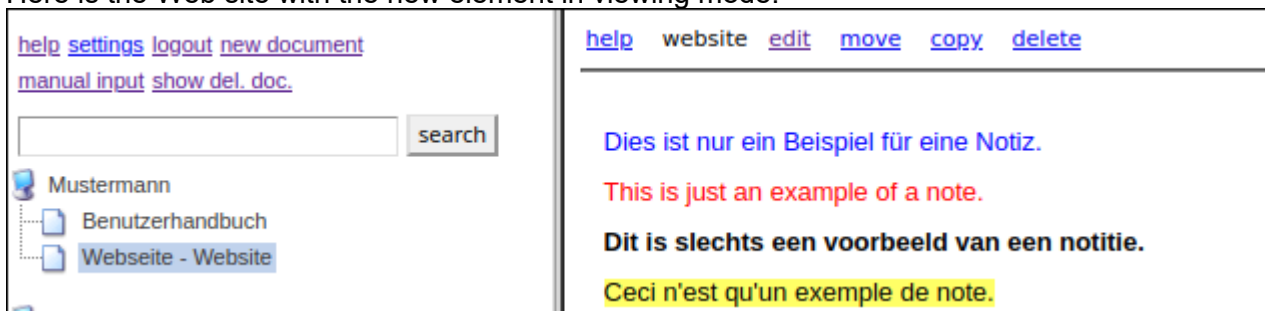


Fig 16: Web site in viewing mode

### 1.3.4. Spreadsheet

There is no design view for the spreadsheet. Even if it is much more spartan than the well-known representatives of its kind, it is still a full-fledged spreadsheet. It was developed for the *DH*. This becomes clear in the **DH functions** menu. This menu contains special functions for querying values from the *DH*. You can find out more about this in the second chapter, which deals exclusively with the *DH*.

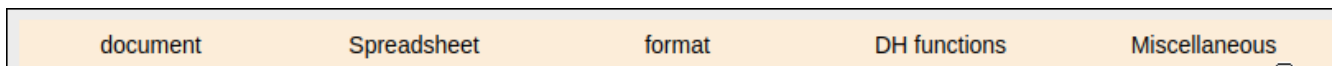


Fig 17: Spreadsheet menu bar

The **Document** menu contains everything that concerns the document itself. Here you can rename, move, copy, delete, save and recalculate the document.

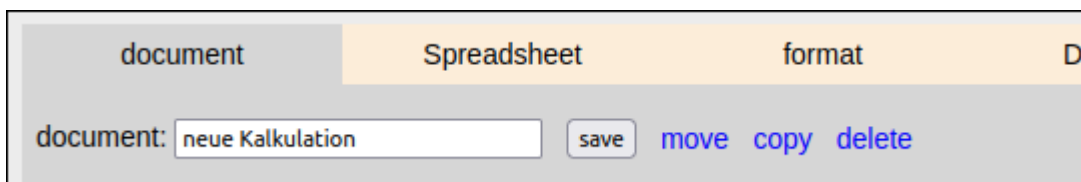


Fig 18: Document menu

The next two menus **Spreadsheet** and **format** only refer to the currently marked area. A detailed description is rather superfluous here. The cell Contents are processed in the same way as with other spreadsheets.

### 1.3.5. Query

If you've never had anything to do with database queries, then there are actually only two things that are of interest to you. First of all, this is the **Description** field in the header, which can give you an indication of what should be displayed here. The second point is the menu **search** option **run**.

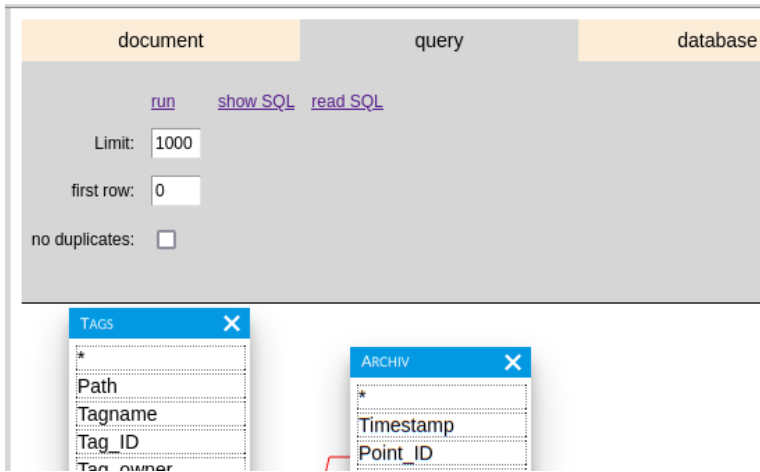


Fig 19: Header with the **Query** menu

In this case, a click on the Execute option brings up the following window:

The screenshot shows a window titled 'ERGEBNIS ABFRAGE' with a blue header bar and standard window controls. Inside the window is a table with the following data:

Point_Path	Tagname	Point_ID	Timestamp	Value
/	T25	250	2021-09-21 21:27:48	8
/	T25	250	2021-09-21 21:27:48	8
/	T25	250	2021-09-21 21:37:45	8.125
/	T25	250	2021-09-21 21:37:45	8.125
/	T25	250	2021-09-21 21:38:51	8.062
/	T25	250	2021-09-21 21:38:51	8.062

Fig 20: Result of the Query

Well, now for users who are familiar with a database Query

The interface shows a header section with the following controls:

- document Version: [dropdown]
- query: [dropdown]
- database: DH [dropdown]
- tables: User\_akt [dropdown]
- query type: SELECT [dropdown]
- description: Zeigt alle Werte des [text input]
- help: [button]

Below the header, there are two lists of fields:

- USER\_TAGS:** Path, Tagname, Tag\_ID, Tag\_owner, Pointname, Point\_Path, Point\_ID, Description, EUDESC, scan, interface, archive, step, compression, minarch, info, Property\_1, Property\_2, Property\_3, Property\_4, Property\_5, Point\_Type, Dezimalstellen.
- USER\_AKT:** Point\_ID, Timestamp, Value.

A red bracket indicates a relationship between the Point\_ID field in USER\_TAGS and the Point\_ID field in USER\_AKT.

The **SELECTION AREA** table is used to define the query:

	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-	+
field	Point_Path	Tagname	Point_ID	Timestamp	Value											
function																
alias																
table	User_Tags	User_Tags	User_akt	User_akt	User_akt											
sort																
group	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
show	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
criteria		like 'T25'		>'2020-01-01'												
criteria																

Fig 21: Complete view of the Query document type

If you have already worked with a graphical Query editor, you should find your way around here quite quickly.

In the header, select the database to which you have access. All other databases on this server will not be displayed. The tables and views for which you are also authorized are available for selection in the next list box. In the Query type field, you specify whether it should be a SELECT, UPDATE, INSERT or DELETE. It is always a good idea to do a SELECT before a UPDATE or DELETE. This may help you avoid a nasty surprise.

The tables and their fields are shown below the header. You create a JOIN by dragging a table field from one table to the field to be linked in the other table. You specify the type of JOIN by clicking the right mouse button on the connection (context menu / properties). A dialog opens in which you can choose between RIGHT, LEFT, or INNER Join.

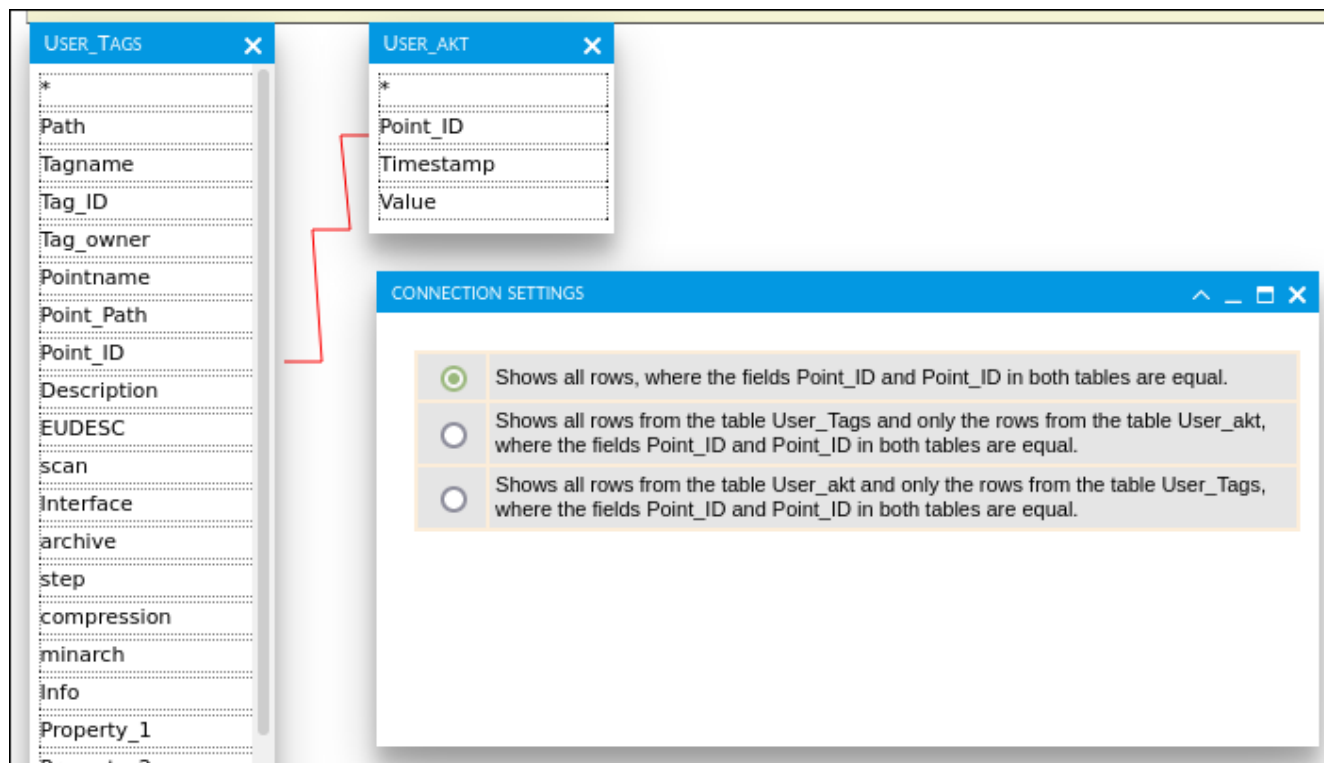


Fig 22: Define the type of connection between the two tables

SELECTION AREA										
	-	+	-	+	-	+	-	+	-	+
field	Point_Path	Tagname	Point_ID	Timestamp	Value					
function										
alias										
table	User_Tags	User_Tags	User_akt	User_akt	User_akt					
sort										
group	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>					
show	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>					
criteria		like 'T25'		>'2020-01-01'						
criteria										

Fig 23: Selection area

The conditions, the fields to be displayed, the grouping and the sorting are defined in the bottom window ( *selection area* ). To do this, either drag the required fields with the mouse from the tables into the first row, or you first look for the table in the table row and then the field in the first row. Instead of a field, you can also enter a MySQL function in the second line. In this case the field above will be removed. The *alias* is to be though, as usual, optional, but should be used when using a function.

The next three lines shouldn't really raise any questions.

The area below the horizontal line is intended for entering the Query criteria. Here everything that is in one line is logically AND linked. All lines are again logically ORed. If you want to check whether the Query is built in the way you want, you can view the SQL expression using the **Query** menu, **Show SQL** option. A new frame is displayed above the selection area, which shows the generated SQL expression. If you make corrections here, you can read them in again using the **Query** menu. To do this, select the Read in **SQL** option.

Further options in the Query menu are the definition of a limit. A limitation of 1000 lines is preset. This makes sense if you want to Query a table with several million records. The option **no duplicates** puts the

keyword DISTINCT after the SELECT.

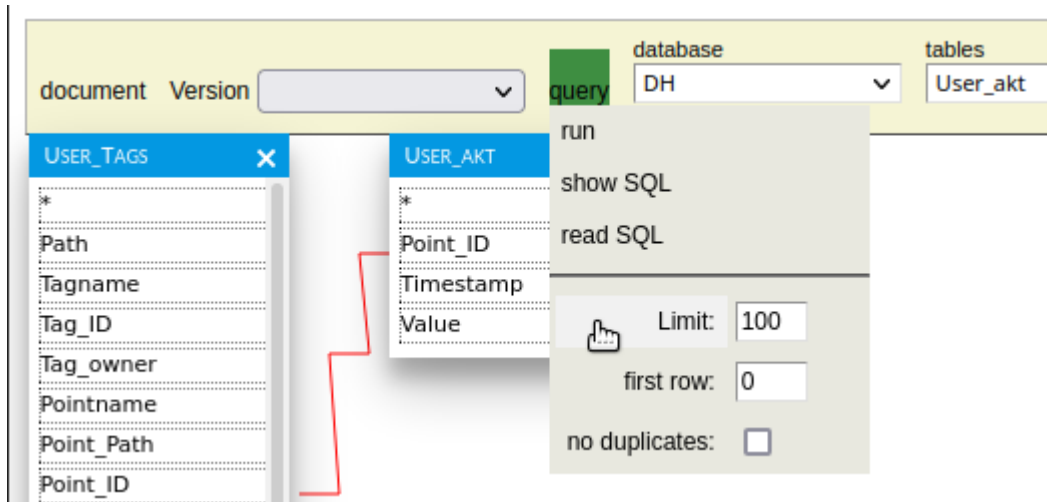


Fig 24: Menu Query

The display of the selection area depends on the selected Query type. For the Query types UPDATE and INSERT, an additional line Write is displayed. The future field content is entered in this line.

And as usual: Please don't forget to save your document before selecting another document. Wouldn't it be a shame about the work you put into it?

### 1.3.6. Form

Forms can be displayed in three different views. The "normal" form view and the table view are available for editing data. Another view, the Design View, is described in the Developer's Guide.

In the "normal" view, only one data record is displayed on the form page. In the table view, however, several data records are displayed at the same time. Data records can only be edited in the "normal" view. You can switch between the two views using the Form View or Table View button. Which button is displayed depends on the currently set view.

The data records cannot be edited in the table view, but a double-click in the relevant line takes you directly to the selected data record in the "normal" form view.

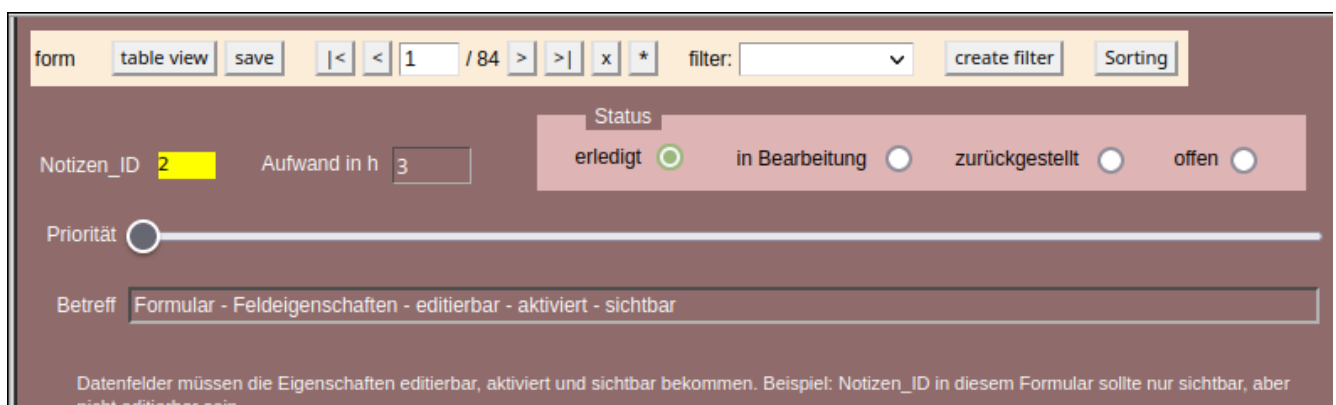


Fig 25: "normal" form view

form **Form view** |< < 1 / 84 > >| x \* filter:  create filter Sorting

Notizen_ID	Betreff	Notiz
2	Formular - Feldeigenschaften - editierbar - aktiviert - sichtbar	<div><font size="2">Datenfelder müssen die l
4	Formular - Feldeigenschaften - Rahmen	Alle Elemente sollten die Eigenschaft "Ral
5	Formular - Feldeigenschaften - editierbar - aktiviert - sichtbar	Am besten einen Dialog bauen, über den
6	Formular - Feldeigenschaften - editierbar - aktiviert - sichtbar	Testen, wie es sich bei einer Datensatzän
7	Formular - Feldeigenschaften - editierbar - aktiviert - sichtbar	
8	Formular - Ansicht umschaltbar machen	Sollte in der Tabellenansicht zum aktuellen
9	Formular - Tabellenansicht - umstellen auf eine Tabelle wie auf den Konfigurationsseiten auch.	Sortier- und Filterfunktionen einbauen. Die
10	Formular - Datenquelle - "JOIN Abfrage"	Das Problem mit dem Tabellennamen und

Fig 26: Table view

### Navigation:

The next picture shouldn't leave any questions unanswered. What you might notice here is the current record field. In order to go to a specific data record, the number of the data record can also be entered directly here. The TAB key does the rest.

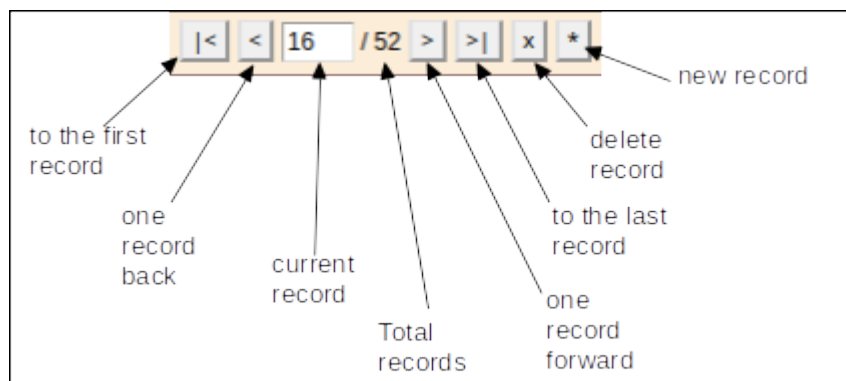


Fig 27: Navigation area in the header

### Filter records:

Any number of filters can be defined and saved for later use using the Create filter button in the header of the document. Each user only gets to see the filters that they have created themselves. The filters created in this way can be selected again and again via the filter list box.

The dialog for creating or changing a filter is shown in the background in the next picture. You can enter the filter manually in the top, large field, or you can use the much more convenient fields in the table. Regardless of whether you edit the upper field or use the table, the other area is always updated in the background.

The first of the sixth table columns is intended for the entry of one or more opening brackets. One of the available fields can be selected in the second column. Then comes the operator (<=> <= like, ...). The fourth column contains the term filter. Please make sure that you always put a text in double quotation marks. If you filter for a numeric value, the quotation marks are of course omitted. In texts, the % symbol serves as a placeholder for any number of characters. In the example from the figure below, a search is made for the term Formulaire in the Title\_FR field. It doesn't matter where the word appears in the text. If you were to leave out the first % character, only those records would be filtered out that begin with the term form.



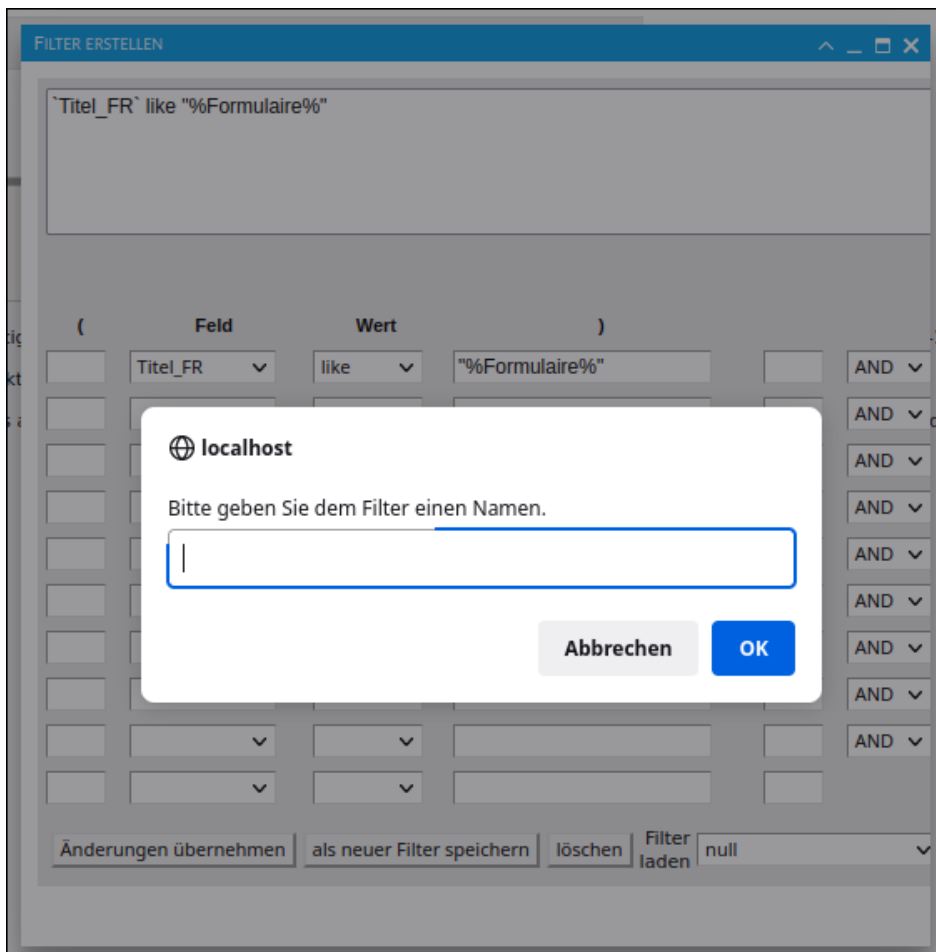


Fig 28: Dialog for creating filters

At the bottom of the dialog there are buttons with which you can save the changes to a filter or save a filter under a new name. The image above shows a field for entering the name of a new filter.

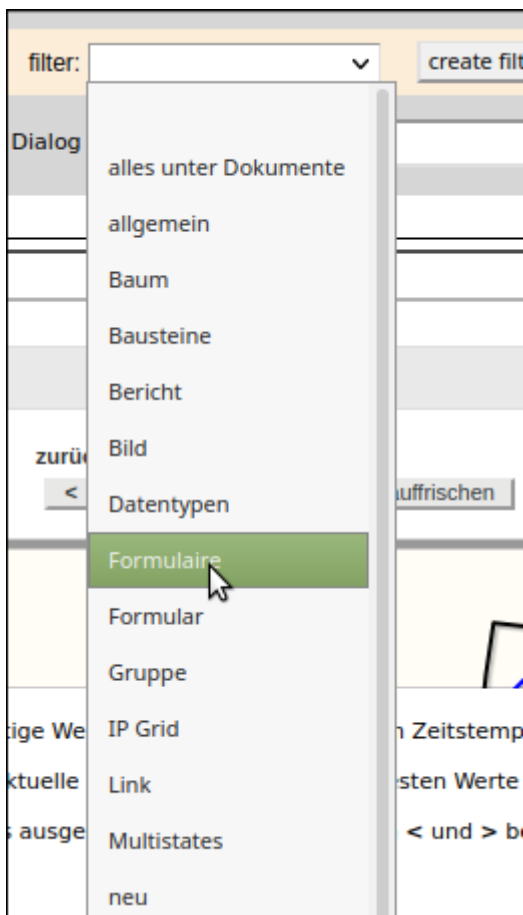


Fig 29: List of filters that have already been created

The new filter will appear in the list of filters at the latest when the form has been reloaded.

The screenshot displays a software interface with a top navigation bar containing buttons for 'form', 'table view', 'save', and pagination controls showing '1 / 2'. A 'filter:' dropdown menu is set to 'Formulaire', with 'create filter' and 'Sorting' buttons to its right. Below this, the 'Hilfe\_ID' is '31', the 'Dokument' field contains 'Formular', and the 'Dialog' field is empty. A language selection bar includes 'Deutsch', 'Englisch', 'Niederländisch', and 'Französisch'. At the bottom, the 'Überschrift' field contains 'Formulaire'.

Fig 30: new filter in action

Here the new filter form in action. There are only two records in the table that have this term in the heading. Without the filter, as can be seen in the images above, 52 records would be displayed.

### **Sorting:**

You can sort the records up or down according to each field. The sorting is not saved, however. For sorting, please use the switch on the far right in the header of the document. A very simple dialog opens which does not require any explanation.

### 1.3.7. Report

As with the form document type, the design view is described in the developer's manual.



Fig 31: Header of a report

Use the small buttons to page through the report. Alternatively, you can also enter the desired page number directly. If the report contains a cover sheet, this sheet is counted as one page in the navigation. The numbering on the pages themselves does not count the cover sheet as a page. The button Create PDF creates a PDF document of the report. Please note that the representation in the PDF document may differ slightly from the preview due to the fonts available for the PDF output of the report. In addition, it may take a while until the PDF document is ready.

### 1.3.8.Link

A link is actually not a "real" document. Rather, it is a reference to an existing document. A link is always useful when you want to group documents that are scattered somewhere in the tree structure under one branch. You can use it to arrange documents the way you want without changing their original position in the tree structure.

When you open a link, the linked document is displayed, but first, ie at the top, the options for editing the link appear (edit, delete, move). The linked document is displayed below the horizontal dividing line. If you are the admin for your area of the unidb, or if you created the linked document yourself, you will also see the same options for the linked document.

In the example shown below, a document of the type Image (DH) was linked.

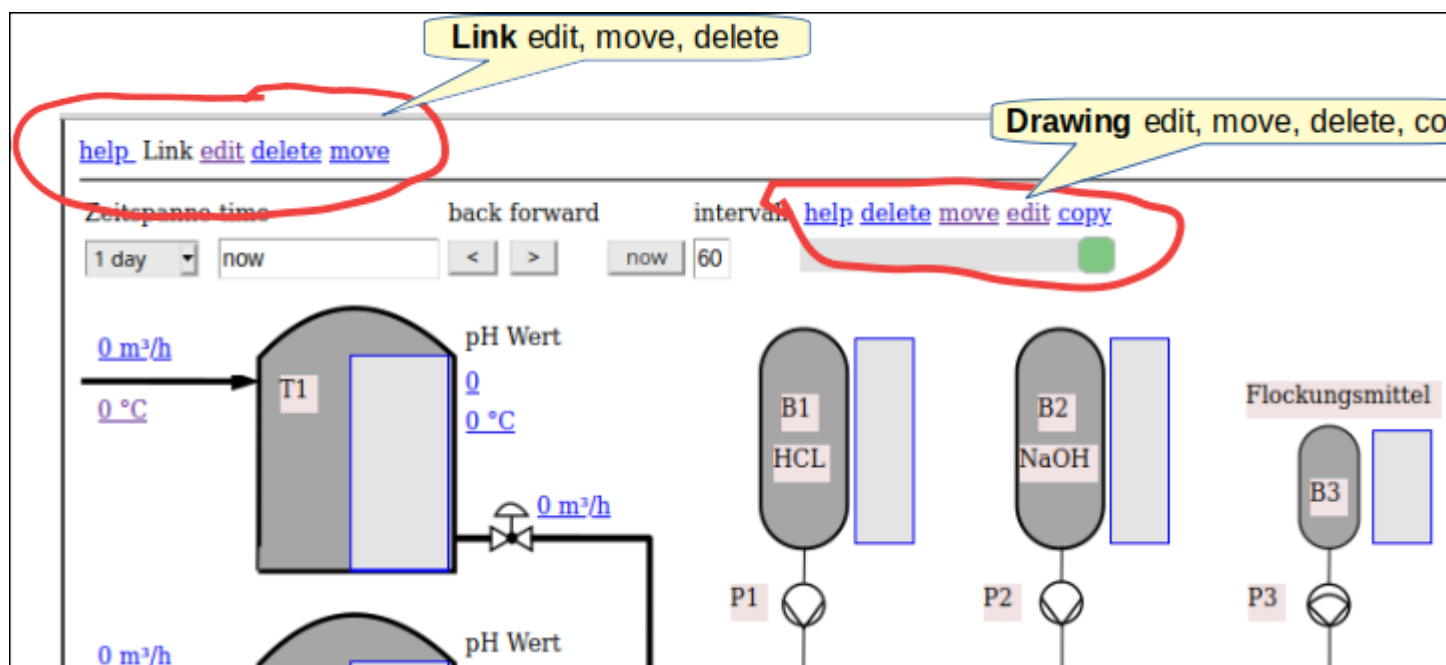


Fig 32: Headers of the link and the document

If a group (DH) has been linked, it looks something like this:

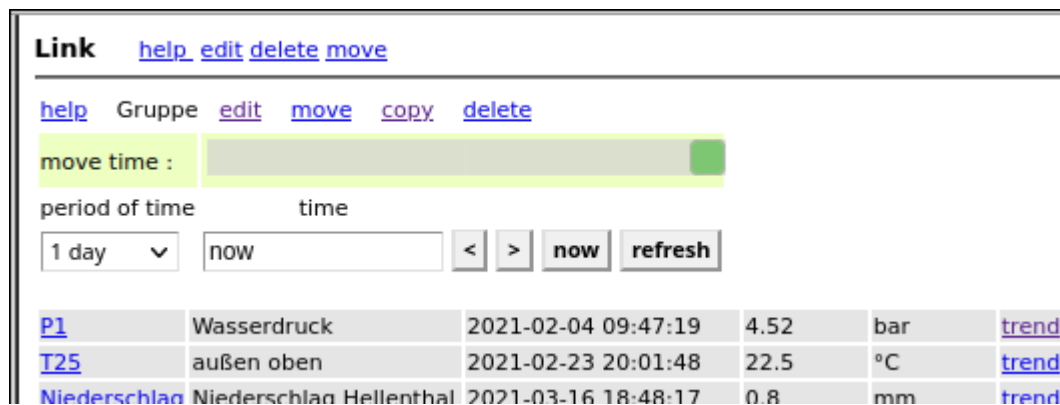


Fig 33: linked document of the group type

Here, too, the options for the link are at the top and the linked document is only displayed below the dividing line. You should therefore be careful here to avoid confusing the link and the document.

#### Caution, stumbling block:

A link is created like a real document. After selecting the higher-level document, a tree structure is displayed again. Here you select the document to which the link should point. **Therefore, please pay attention to the text above the tree structure.** In the second tree structure there is the text “Where should the link point?”.

## 1.4. Permissions

The allocation of authorizations is based on the tree structure. Your administrator determines which areas of the tree structure you have access to.

Example:

The tree structure looks like this to the user Max Mustermann:

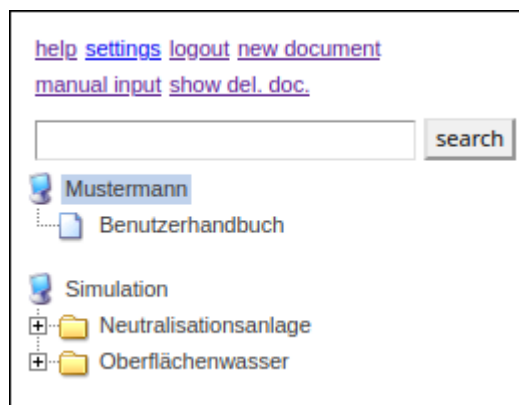


Fig 34: Tree structure as seen by the user Max Mustermann.

The administrator, however, sees the complete tree structure.

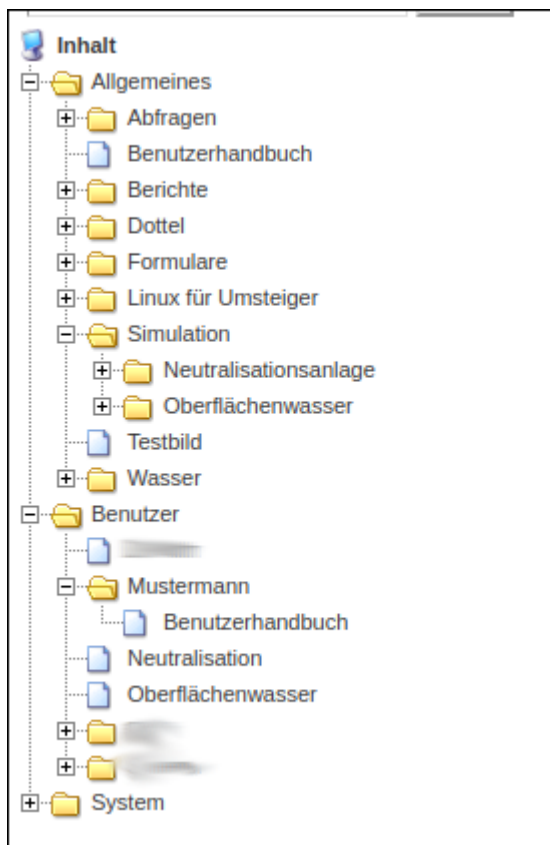


Fig 35: Tree structure from the point of view of the administrator

The reason for this is that the user Mustermann has been assigned authorizations for the branches / General / Simulation and / User / Mustermann. Since a tree structure always begins at the lowest level of a path, two tree structures are also displayed here. The last part of the path always forms the root of the tree structure shown. In the example / General / Simulation this is *simulation*.

A user can also be assigned an authorization without creating a tree structure for this path. In our example this is the authorization for the document *User Manual*. This document is a typical document that should be made available to every user. In the user's private branch, here / User / Mustermann, the administrator then creates a link that refers to / General / User Manual.

If you look at a document that was not created by you, you will see a reduced header. The options edit, delete, move and restore are not available there.

Here is the view of a document that was created by another user:

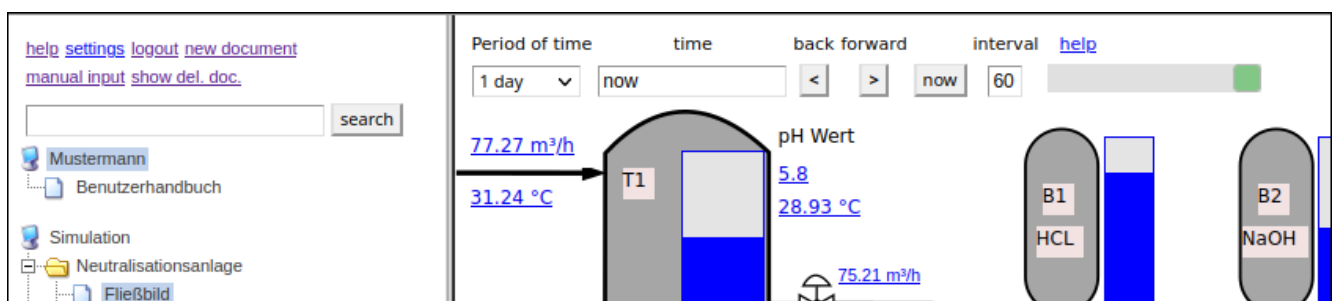


Fig 36: Document view without write authorization

And now the same document in the view of the creator:

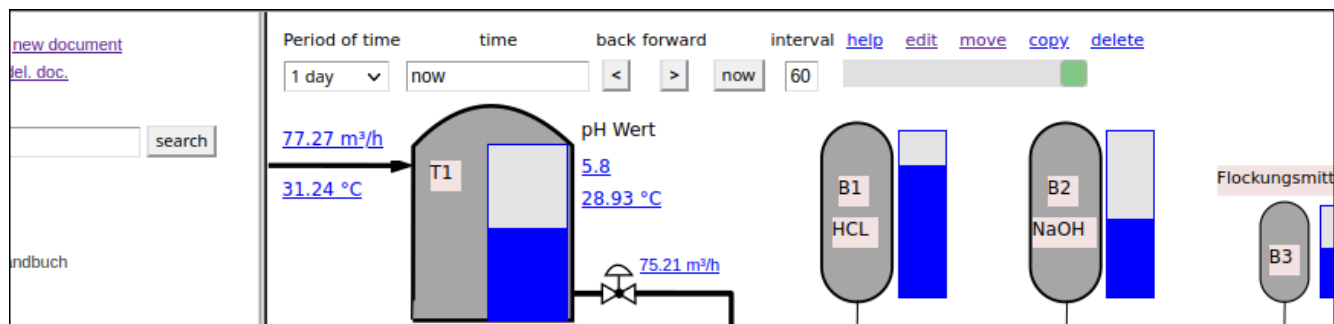


Fig 37: Document view with write authorization

## 2. DH

DH stands for DataHistorian. As the term suggests, it has something to do with data that go back a long way and that are related in time. Sounds more complicated than it is.

The *unidb* was originally developed as a user interface for the MeierSoft DataHistorian. Therefore, it also has some document types that are specially designed for this task. These documents are rarely changed. That is why we limit ourselves to viewing mode in this manual. The description of the design mode can be found in the developer's manual.

The question of how the values get into the DH is not so important here either. But when we talk about an interface, we mean a program that fetches the data from somewhere and writes them to the DH.

### 2.1. Introduction

The principle according to which a DataHistorian works is actually quite simple. It is about reading any values at regular intervals and writing them to a database together with the point in time. The collected data is then made available to the users of such a DataHistorian in a processed form. The data can be output, for example, in table form, as a diagram, embedded in drawings / images, or directly as "living" data in a spreadsheet. Of course, the data historical should also be able to deliver the data not only in raw form, but also statistically processed.

That was now short and sweet and therefore probably difficult to understand. It becomes a little easier if you first get an idea of what a data point actually is. A data point is just an object that cannot be touched. So it's a virtual thing. Like all objects, a data point also has properties that are typical for objects of its kind.

An example of an object from the non-virtual world: In the office of Mr. Doe hangs a thermometer on the wall. This thermometer has a number of properties. Important properties are, for example, the display range or the unit in which it shows the measured temperature. However, it is also important to provide information about the place where it measures the temperature.

characteristic	value	comment
Measuring range from	-10	
Measuring range up to	+40	
unit	° C	Other possible units would be K or ° F, for example.
Input medium	gas	There are also thermometers which are designed for water, organic liquids, corrosive liquids, air (ex protected), ...

Type alcohol Other types: electronic, bimetal, ...

Installation location Mustermann's office

Mr. Mustermann asks the caretaker to fix the air conditioning in his room. In order to make the urgency of his concern clear, he logs the temperature in his office and in the kitchenette every hour throughout the week. He notes the temperature in the following form:

location	Date time	Temperature in ° C
Mustermann's office	07/14/2018 12:00	22.5
Mustermann's office	07/14/2018 13:00	23.4
Mustermann's office	07/14/2018 2:00 pm	24.6
Mustermann's office	07/14/2018 3:00 pm	26.1

location	Date time	Temperature in ° C
Tea kitchen	07/14/2018 12:00	20.5
Tea kitchen	07/14/2018 13:00	21
Tea kitchen	07/14/2018 2:00 pm	21.6
Tea kitchen	07/14/2018 3:00 pm	22.4

Admittedly, Mr. Mustermann is probably, to put it mildly, a fairly thorough contemporary. But what Mr. Mustermann does is basically nothing else than what a Datahistorian also does. The indication of the place where he measures his temperatures is nothing more than a data point.

For a DataHistorian, everything is a data point that supplies values with an associated time stamp. Most of these data points are sensors. From the point of view of a DataHistorian, it does not matter what a sensor measures (temperature, fluid level, pressure, flow rate, ...). Other data points are often analysis values from a laboratory, values from other databases, manually entered measured values, or very often values that are calculated from other data points.

Simple example:

You have installed a filter in your production facility. There is a pressure measurement in the front of and behind the filter. The pressure that these measurements show is relatively uninteresting to you, but anyone who operates such a filter is interested in the difference between these two pressures. It therefore makes sense to create a data point that regularly calculates this pressure difference from the two measured pressures and writes each result of these calculations together with a time stamp in the DataHistorian. For the user of the DataHistorian it looks as if the differential pressure had been measured.

A **data point** of a DataHistorian is usually referred to as a **tag**, or also a **point**. From here on, the term *point* is used instead of the term *data point*.

## 2.2.DH documents

The documents for the DH are similar to the documents presented so far in terms of viewing and design mode. The headers are also similar. A noticeable feature of the headers of some documents is the control elements that can be used to select a point in time or a period.

This can be shown nicely using the example of a trend display for a measured value.



Fig 38: 24 hour trend of a measured value

The controls are marked from 1 to 5 in Figure 38.

The **period of time** (1) in a trend display indicates the period of time over which the trend extends, but is also required to determine the period of time by which the buttons move **back** and **forth** (4) through the data history. The button **now** (5) always takes you to the current time. The selection list **Function** (2) gives you the option of displaying the values as they were collected (raw values), or as hourly or daily averages, minimum or maximum values, ... Leave the trend of a value to yourself display a complete year, then it certainly makes little sense to use the raw values for this if the value is actually written to the DH every 60 seconds. Up to 525,600 values are recorded every year. With a trend width of 600 pixels, there are almost 1000 values per pixel. That certainly doesn't make sense. On the other hand, if you select the DMinMax function (daily maximum and minimum value), then you get the same display quality with only 2 x 365 points.

### 2.2.1. Document type drawing

This document type is very well suited for the representation of a flow diagram of a production plant. Typically one sees such flowcharts in control systems for plants from the chemical industry, water treatment, plants of energy suppliers, ...

The next figure shows such a flow diagram of a fictitious plant for wastewater treatment. Since this is a simulation, the schema also contains "living" data.



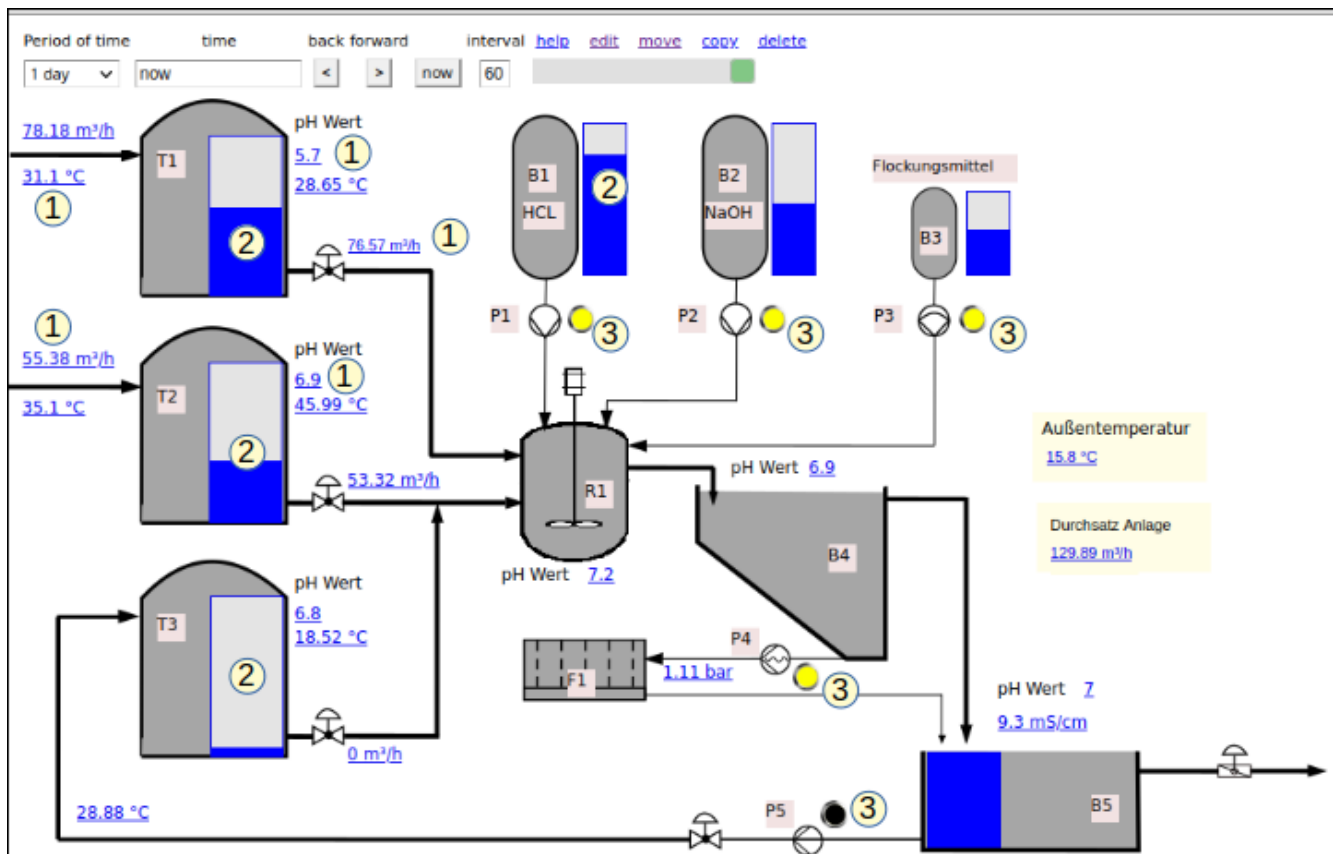


Fig 39: document type drawing with a flow chart

In the header we see the familiar control elements again, as we already know them from Chapter 1 (help, edit, move, copy, delete). In addition, there are the control elements described above, with the help of which you can move through the data in terms of time. You will also find a field called **Interval here**. The value in this field indicates the time intervals at which the document is reloaded. However, this only happens if the time is set to *now*. That makes sense, because at the time, *now* always expected the most recent value. Since many values are written to the DH once per minute, the default setting of 60 seconds is usually appropriate here very good.

The data is visualised in a picture in very different ways. The simplest representation is the display of the value. Some examples are marked with a (1) in Figure 39. The values are displayed as a hyperlink. If you click on such a value, a new tab is opened in the browser, which shows a 24 hour trend of the value. Another, simple form is the representation as a bar (2). This representation is wonderfully suitable for displaying fill levels.

Values that represent a state can be easily represented using multistates. These are graphics that change their appearance depending on the value. You will find an example of this at the points marked with (3). These are control lamps which are filled with yellow, for example if they are to show the status "switched on". If they represent the status "switched off", they are filled with black.

In order not to unnecessarily clutter an image with information, the point in time for the value is omitted. You can get an approximate indication if you look at the **time** field in the header. To get the exact point in time for a displayed value, simply position the mouse pointer over the value for a short moment. The time stamp is then displayed as a popup info.

Figure 40 shows examples of all available elements in an image.

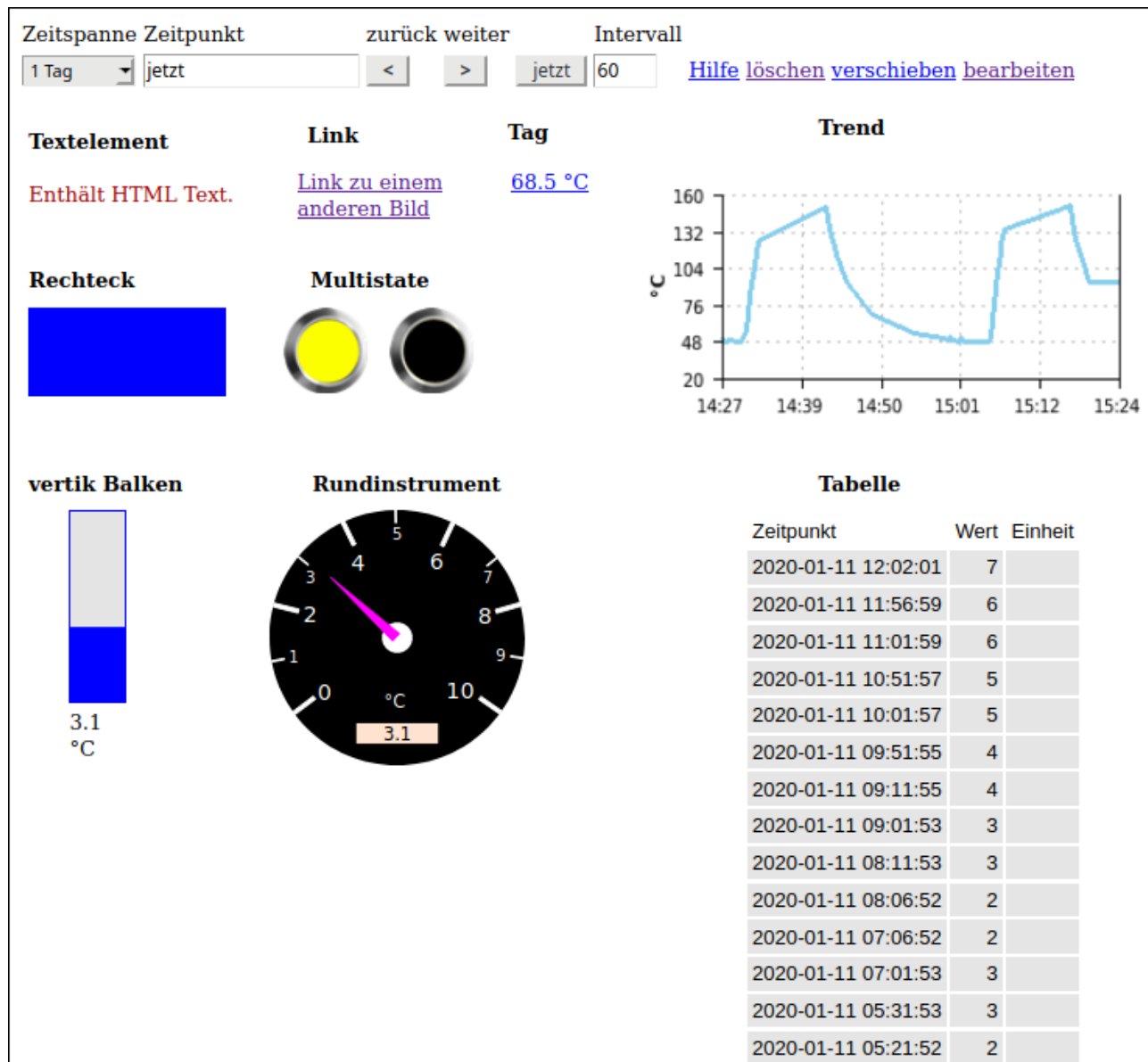


Fig 40: Examples of elements in a drawing

## 2.2.2. Group

A group is a list of tags. The tag name, a description of the tag, the time stamp, the value and the unit are listed in a table. Groups are also reloaded once a minute if the time is set to *now*.

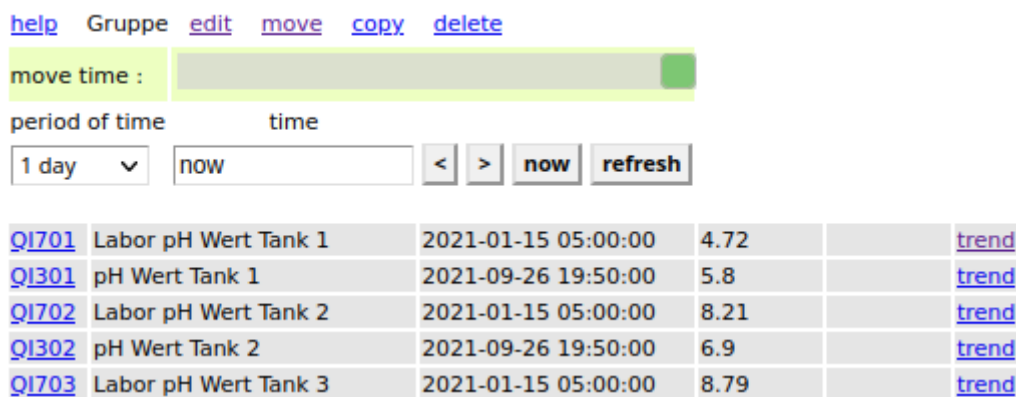


Fig 41: Partial view of a group

Here, too, you can change the time in the usual way. The first and last columns each contain a link. The

link in the last column refers to a trend as already described above.

The specialty here is that all tags from the group are listed below the trend. If you put a cross in one of the checkboxes, this tag is also shown as a trend.



Fig 42: Representation of three trends in one graphic

After you have selected one or more tags, the additional trends will be shown after you have pressed the **refresh** button.

And now back to the group. The link from the first column opens a popup window which shows the configuration of the relevant tag.

TAG DETAILS

^
\_
□
✕

Tag_ID	path	Tagname		
491	/Allgemeines/Simulation/Neutralisationsanlage/	QI701		

Point on which the Tags are based:

Point_ID	path	Pointname	description	units
503	/Allgemeines/Simulation/Neutralisationsanlage/	QI701	Labor pH Wert Tank 1	

interface	step	decimal places	scan	mean values
Simulation	1	2	1	0

Archiv	compression	minarch	scale min	scale max
1	0	0	1	14

property 1	property 2	property 3	property 4	property 5
94				

Info	Point type	intervall	first value	last change
	double	60	2018-12-31 23:00:00	2021-05-28 19:55:31

Fig 43: Configuration of a tag and the associated point.

Here we have to deal briefly with theoretical things, unfortunately there is no other way.

In many DataHistorians, the term tag stands for a data point. This is a little different with the MeierSoft DH. Here a point represents a data point and a tag is just a reference, something like a link to a point. This has the advantage that a data point can be represented using several names. Why this is a great advantage quickly becomes clear when you take a closer look at DataHistorians. At this point it is only important to know that several tags can point to a data point.

In Figure 43 we can see the configuration of the tag with the name QI701 in the table above. If there were several tags that refer to the same point, then these would also be listed here.

The configuration of the associated point, which here has the same name as the tag, is in the second, much larger table below. If not all parameters are immediately understandable here, then that is not a problem at all. These parameters are examined in more detail in Section 2.3.

### 2.2.3. Trend Group

A trend group is a document that shows up to four different values as a trend. The difference between this type of trend and the trend seen so far is that these trends can be configured in great detail. The configuration is finally saved with the document. Every tag has its own scale of values. You can also zoom into an area with the mouse. This convenience is bought with a longer charging time.

The document is divided into three tabs. The usual header controls are located on the tab that best suits them.

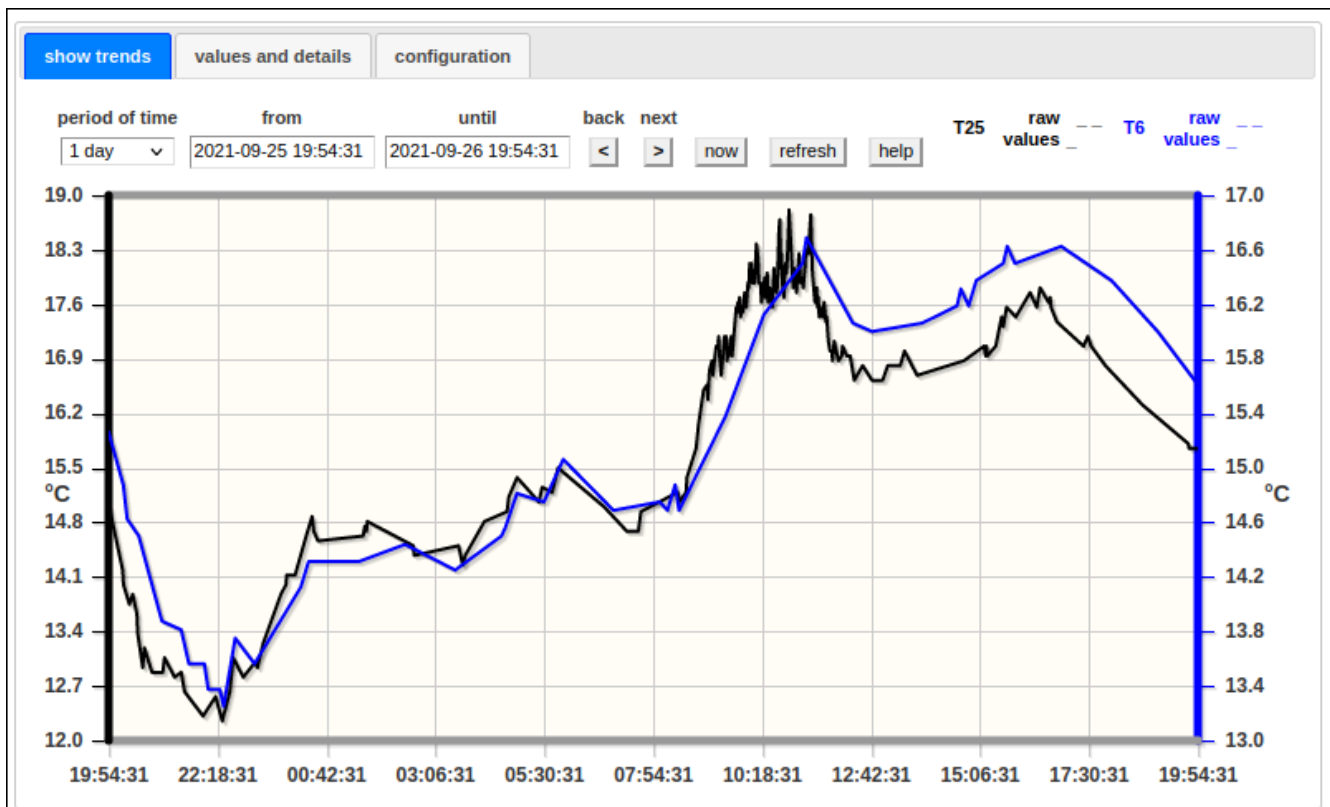


Fig 44: Trend group - first tab "Show trends"

To take a closer look at an area of the trend, mark the area with the mouse button pressed.

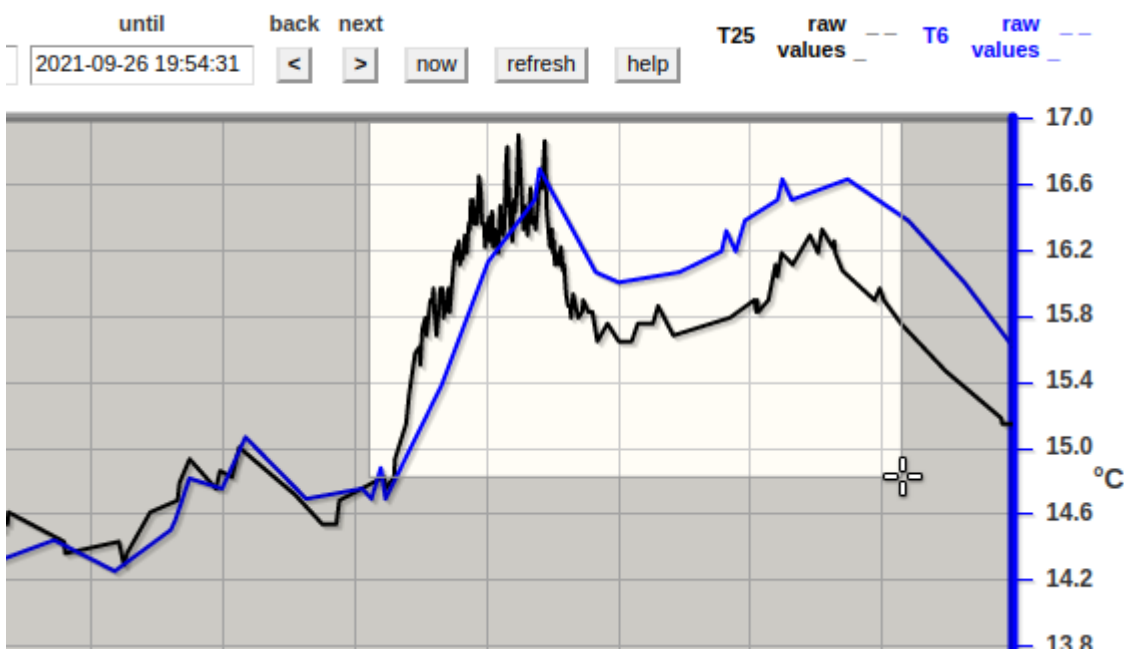


Fig 45: Zoom into an area with the mouse.

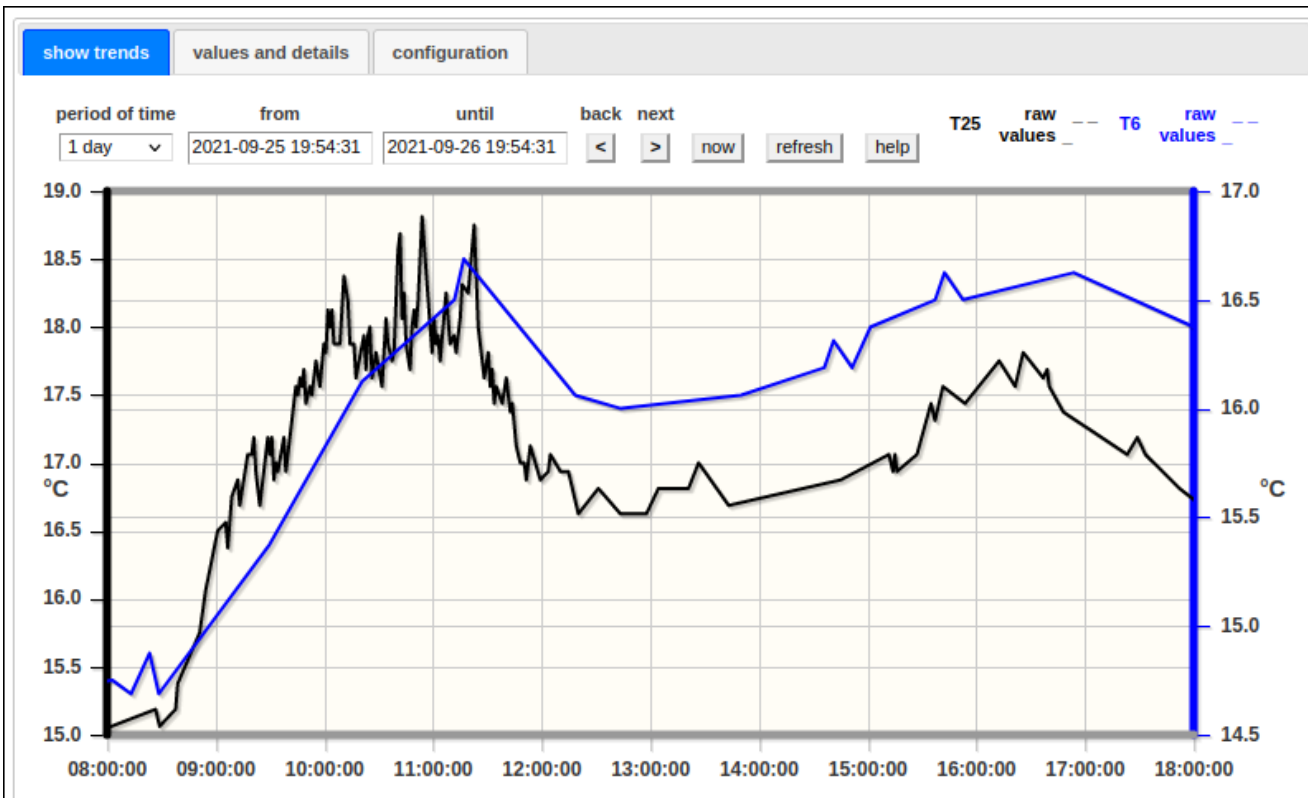


Fig 46: enlarged view of the area marked in Fig. 45

Fig. 46 shows the same document as Fig. 44. The area marked in Fig. 45 is now displayed over the entire graphic area. With a double click anywhere in the graphic, the trend is shown again as in Fig. 44. The value scale can be assigned to the appropriate trend by means of its colored marking. You can find the legend at the top right of the tab.

The second tab, numbers and details, shows the values from which the trends were generated in tabular form. A separate tab is assigned to each tag. On the last tab, Configuration, we will see that several trends can be drawn for one tag. In addition to the raw values, you can also configure, for example, the trend for the hourly mean value and / or the daily mean value for a tag. Thus, on the Numbers and Details tab, we also get several tables for each tag.

Figure 47 is a screenshot of the 'values and details' tab in the software. It shows a table of raw values for tag T25. The table has two columns: 'Timestamp' and 'value'. The data is as follows:

Timestamp	value
2021-09-26 19:53:53	15.75
2021-09-26 19:42:49	15.75
2021-09-26 19:41:42	15.812
2021-09-26 18:40:51	16.312

Fig 47: second tab - numbers and details

Above each table is the tag name and the type of value. In Fig. 47, rV is indicated there. The abbreviation rV stands for Raw Values.

The **tag configuration button** opens the dialog as shown in Figure 43.

There's a lot to talk about about the last Configuration tab.

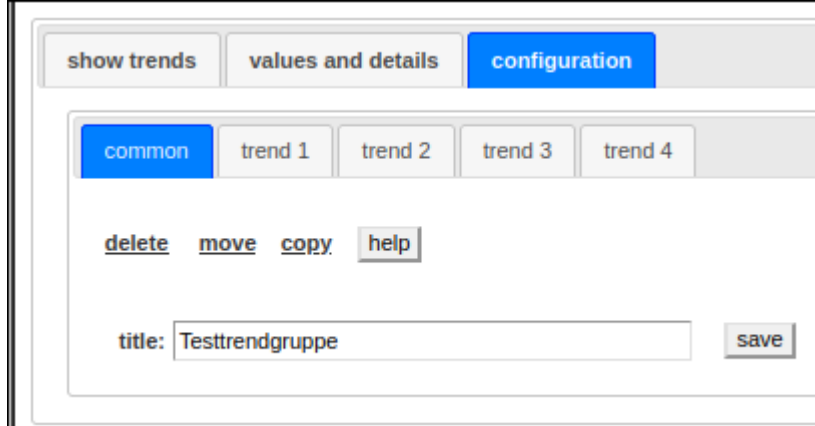
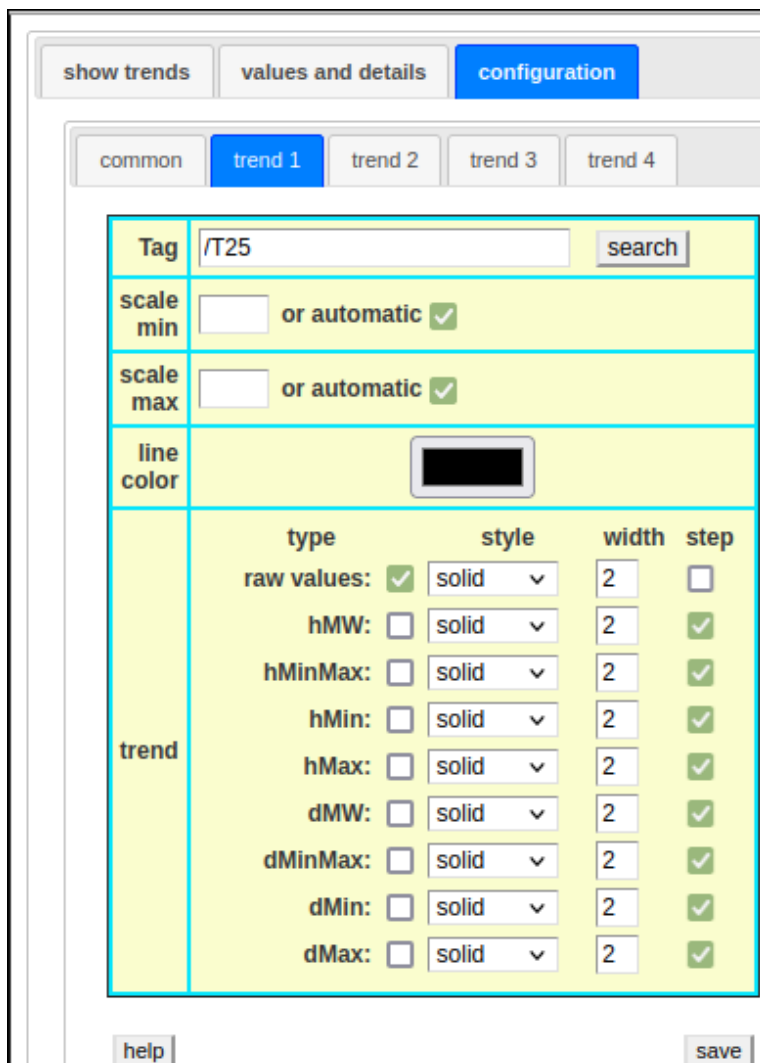


Fig 48: general configuration

This tab also contains several tabs. The first of these, called general, is the simplest. The other four cards have the same structure.

On the general tab we find the links as we see them in the header in other document types. The name of the *trend group* can also be changed here.

The configuration of the trends, on the other hand, is much more extensive.



Tag	search																																								
/T25																																									
scale min	<input type="text"/> or automatic <input checked="" type="checkbox"/>																																								
scale max	<input type="text"/> or automatic <input checked="" type="checkbox"/>																																								
line color	<div></div>																																								
trend	<table><thead><tr><th>type</th><th>style</th><th>width</th><th>step</th></tr></thead><tbody><tr><td>raw values: <input checked="" type="checkbox"/></td><td>solid</td><td>2</td><td><input type="checkbox"/></td></tr><tr><td>hMW: <input type="checkbox"/></td><td>solid</td><td>2</td><td><input checked="" type="checkbox"/></td></tr><tr><td>hMinMax: <input type="checkbox"/></td><td>solid</td><td>2</td><td><input checked="" type="checkbox"/></td></tr><tr><td>hMin: <input type="checkbox"/></td><td>solid</td><td>2</td><td><input checked="" type="checkbox"/></td></tr><tr><td>hMax: <input type="checkbox"/></td><td>solid</td><td>2</td><td><input checked="" type="checkbox"/></td></tr><tr><td>dMW: <input type="checkbox"/></td><td>solid</td><td>2</td><td><input checked="" type="checkbox"/></td></tr><tr><td>dMinMax: <input type="checkbox"/></td><td>solid</td><td>2</td><td><input checked="" type="checkbox"/></td></tr><tr><td>dMin: <input type="checkbox"/></td><td>solid</td><td>2</td><td><input checked="" type="checkbox"/></td></tr><tr><td>dMax: <input type="checkbox"/></td><td>solid</td><td>2</td><td><input checked="" type="checkbox"/></td></tr></tbody></table>	type	style	width	step	raw values: <input checked="" type="checkbox"/>	solid	2	<input type="checkbox"/>	hMW: <input type="checkbox"/>	solid	2	<input checked="" type="checkbox"/>	hMinMax: <input type="checkbox"/>	solid	2	<input checked="" type="checkbox"/>	hMin: <input type="checkbox"/>	solid	2	<input checked="" type="checkbox"/>	hMax: <input type="checkbox"/>	solid	2	<input checked="" type="checkbox"/>	dMW: <input type="checkbox"/>	solid	2	<input checked="" type="checkbox"/>	dMinMax: <input type="checkbox"/>	solid	2	<input checked="" type="checkbox"/>	dMin: <input type="checkbox"/>	solid	2	<input checked="" type="checkbox"/>	dMax: <input type="checkbox"/>	solid	2	<input checked="" type="checkbox"/>
type	style	width	step																																						
raw values: <input checked="" type="checkbox"/>	solid	2	<input type="checkbox"/>																																						
hMW: <input type="checkbox"/>	solid	2	<input checked="" type="checkbox"/>																																						
hMinMax: <input type="checkbox"/>	solid	2	<input checked="" type="checkbox"/>																																						
hMin: <input type="checkbox"/>	solid	2	<input checked="" type="checkbox"/>																																						
hMax: <input type="checkbox"/>	solid	2	<input checked="" type="checkbox"/>																																						
dMW: <input type="checkbox"/>	solid	2	<input checked="" type="checkbox"/>																																						
dMinMax: <input type="checkbox"/>	solid	2	<input checked="" type="checkbox"/>																																						
dMin: <input type="checkbox"/>	solid	2	<input checked="" type="checkbox"/>																																						
dMax: <input type="checkbox"/>	solid	2	<input checked="" type="checkbox"/>																																						

Fig 49: Configuring the trends

The tag name including the path is written in the top line of the table. Tags are organized in the same way as the *unidb* documents. This means that every tag stands at the end of a path. In the example from Figure 43, the tag with the name *QI701* is the path */ General / Simulation / Neutralization system /*. If you know the name and the path of the tag, you can enter both here. In most cases, however, it is more convenient if you only enter part of the tag name and then use the search button to display a

selection. just enter a part of the tag name and replace the missing part or the missing parts with a % character.

Example:

We are looking for the tag that shows the level of the Urfttalsperre. Since all stand measurements start with an L and we somehow still know that the tag has a number around 10, we simply write L1% in the field for the path & tag name. With this, all tags with names beginning with L1 are displayed via the search button.

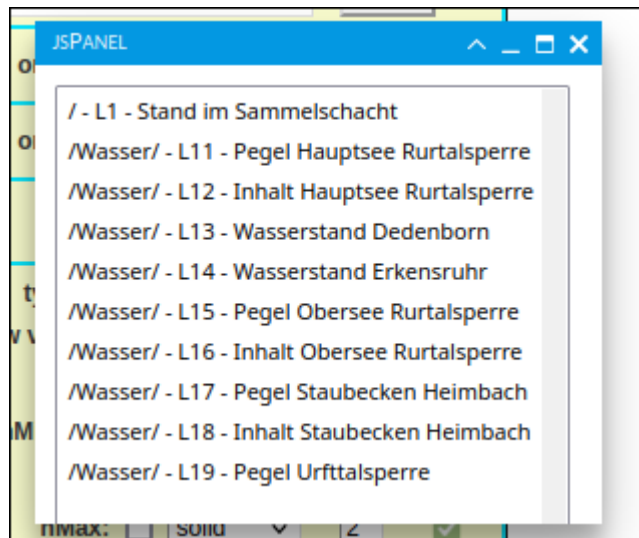


Fig 50: Searching for a tag and actually finding it

We will quickly find what we are looking for in our example. The last tag in the list is the object of desire. Just click on the tag and the unidb will fill in the field for the tag for you.

In the next two lines, Scale Min and Scale Max, define the lower and upper values of the vertical axis. Alternatively, you can also opt for automatic scaling. Both variants have their advantages and disadvantages. Ultimately, it depends on what you need the trend for. For example, if you want to check whether a value is within a certain range, you should set the scale manually.

The line line color should not need any explanation. Just click the switch in the second column and choose what you like.

In the last line you can choose which type of values you would like to have displayed as a trend. In most cases this should be the first type (raw values) in the list. Raw values are the values as they are archived in the DH. For the other types, an h stands for hour, ad for day (calendar day) and MW for mean value. hMW stands for hourly mean values, dMinMax for daily minimum and maximum values, dMin for daily minimum values, ...

Use the Style field to specify whether the trend consists of a solid, dashed or dotted line.

The width is given in pixels. You can usually find out what works best here by trying and correcting it. The last column step indicates whether the trend should have a step shape or whether the connection between two values should be made directly. Which option to prefer here depends on the tag.

two examples:

1.) We want to show the trend of the gasoline price. Here it makes sense to tick the step option, as the price is always valid until it changes.



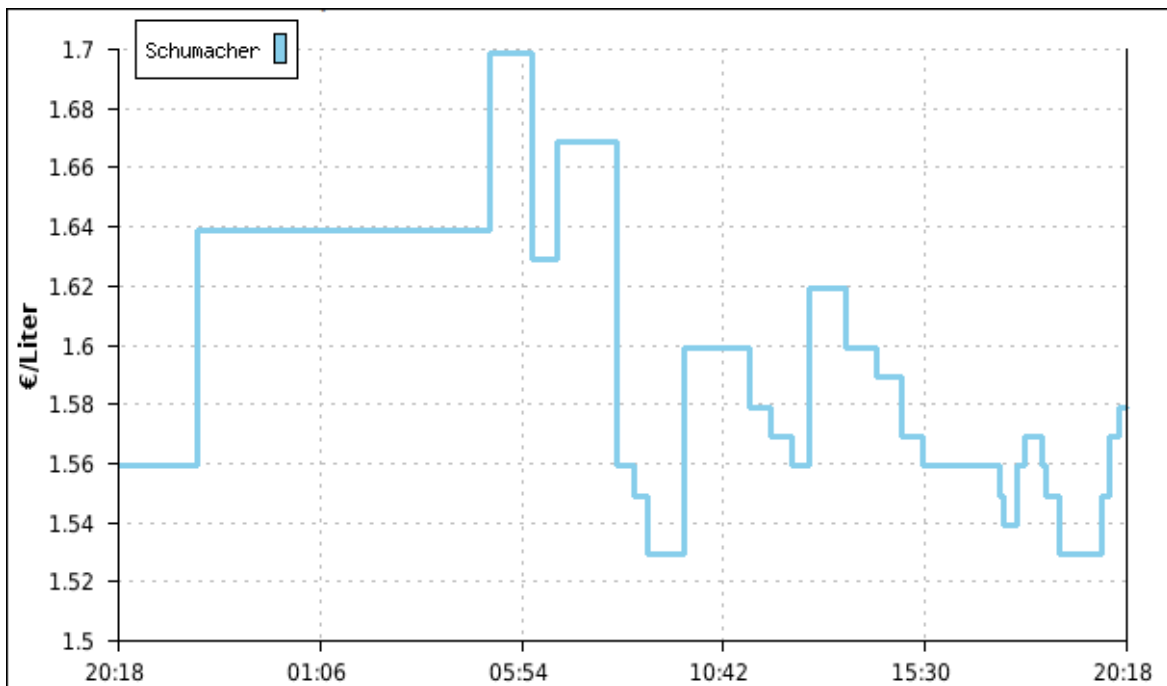


Fig 51: Trend in the price of petrol

In Figure 51 we see that the trend continues in a straight line until the price changes. Then it goes perpendicular to the new price.

2.) We show the trend of a room temperature. The temperature is written into the DH once per minute. That makes 1440 values per calendar day. Since a room temperature changes only very slowly and to a small extent, the DH compresses these values. Only a few values are needed to represent the trend. Nevertheless, the DH provides a value for any given point in time. He does this through interpolation.

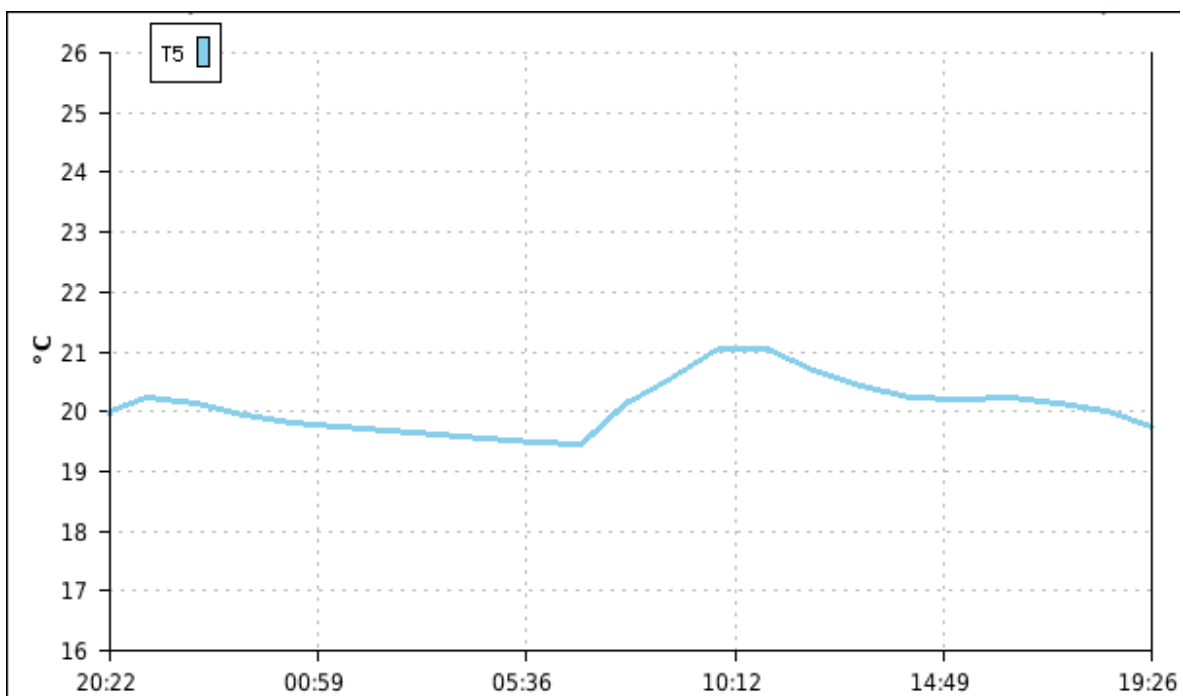


Fig 52: Trend in room temperature

The temperature does not change suddenly and occasionally, but continuously and slowly. In our example, the DH has compressed the 1440 values supplied to 22 values. The quality of the trend is not reduced. A step-by-step presentation of the trend makes absolutely no sense here. It would not adequately represent the actual temperature profile.

The structure of the trend 1 - 4 tabs is always the same.

By the way, the time period for the trend is saved with the document. This means that if you have set 8h  
page 32/45

as the period and save the document, it will later also be opened with this period as the default.  
And now the usual request:

please do not forget to press the save button before exiting the document. Otherwise you can repeat the configuration as a reward.

## 2.2.4. export

These documents are used to configure data exports from the DH.

Any number of tags can be listed, whose values can ultimately be saved in the CSV or HTML formats. The period from which the data to be exported can of course be freely selected. The type of data can be set separately for each day. Raw data is preset. The vt, vt\_interpolated and the Unix time stamp can be added to the export so that you can continue to calculate comfortably with the exported data in a spreadsheet. Details about the vt and the vt\_interpolated can be found in the spreadsheet chapter below. A detailed description of the fields and switches is available via the **Help** switch.

## 2.2.5. Spreadsheet

We already have the spreadsheet in Chapter 1.3.4. discussed. Therefore we limit ourselves at this point to the **DH functions** menu.

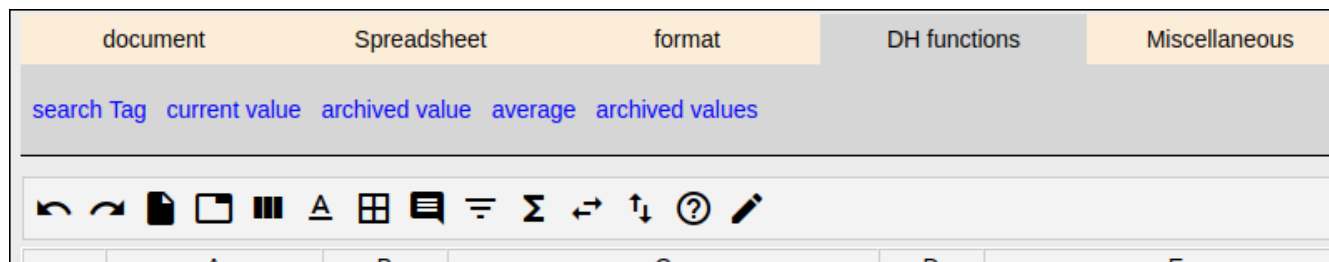


Fig 53: DH Spreadsheet functions

The options:

The first option **search tag** was already described in the previous section.

The **current value** function reads the last value for a given tag from the DH.

Archive value reads a value for any point in time from the archive of the DH.

**Average value** calculates the average value of a tag for a given period of time.

Archived values lists all archived values of a tag for a specified period.

In the following examples we always use the tag / T25 from cell A0.

Whenever a cell from the worksheet has to be specified in a dialog for a DH function, it can either be entered manually, or the cell **marking** can be set on the desired cell and the button **Accept marking**, which is always to the right of the relevant field is pressed.

## current value:

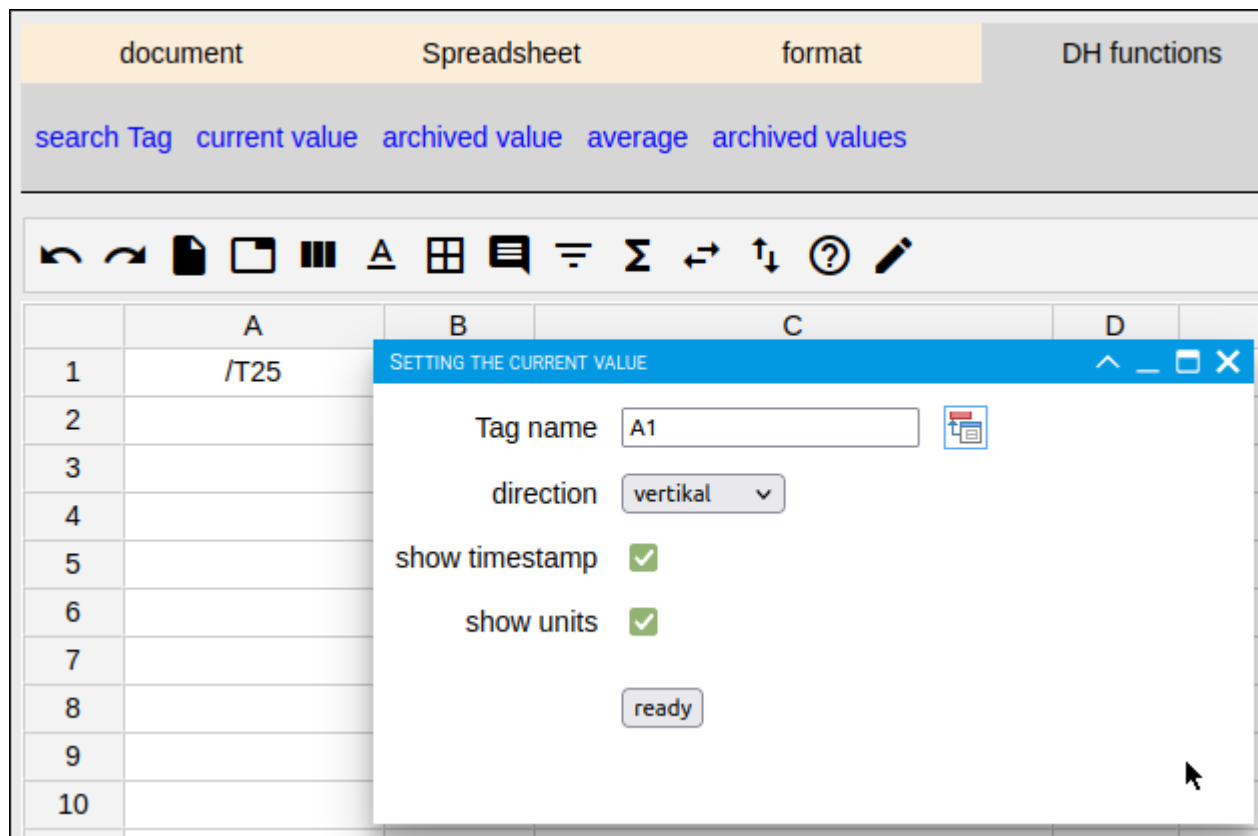


Fig 54: Dialog for the current value function

If we were to accept cell A1 in the output from field in the example from Figure 54, we would overwrite the tag with it, which again would not lead to a satisfactory result. So please be careful.

In the output from field we write the cell address from which we would like to see the result. The Tagname field should contain the cell address in which the path & name of the tag are located.

If you would like to see the time of the value in addition to the current value, please tick the corresponding checkbox below. Proceed in the same way if you want to see the unit of the value. If you have checked one of the two options, then several cells are required for the output of the data. In this case, use the **Direction** field to select whether the output should be in a row or in a column. If you just want to see the current value without time stamp and unit, you can ignore the **Direction** field.

Here is the output of the data after we have pressed the button **Done**:

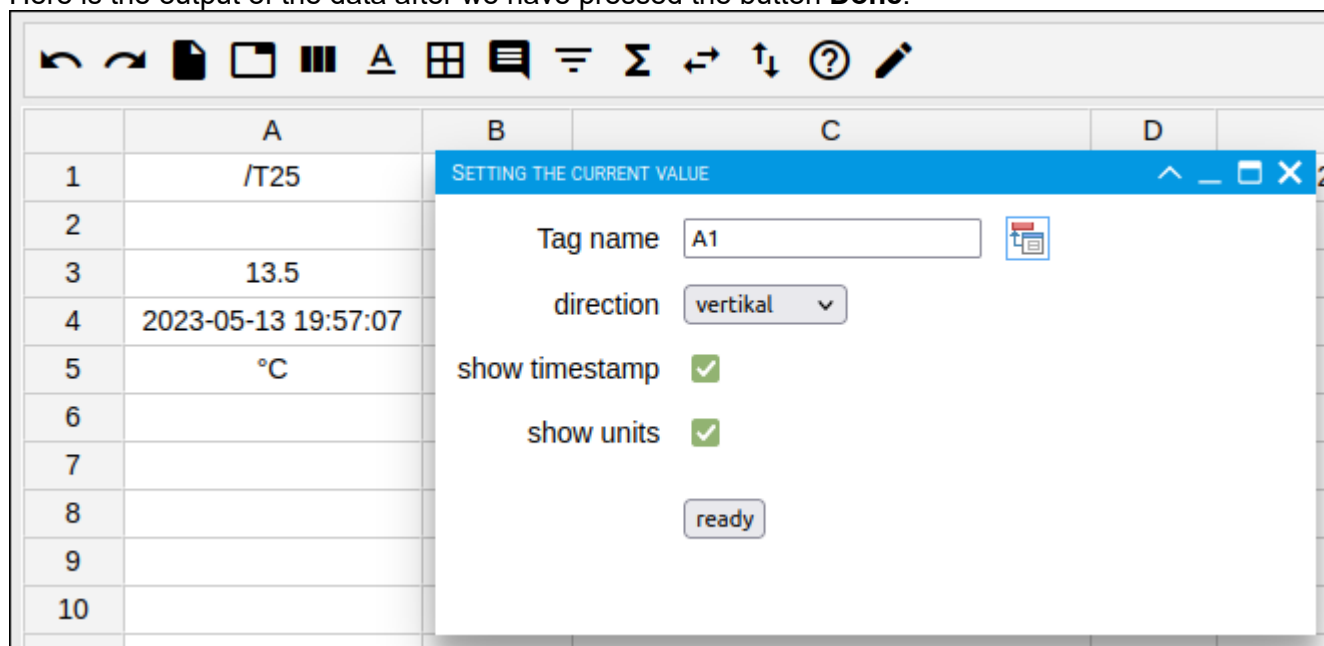
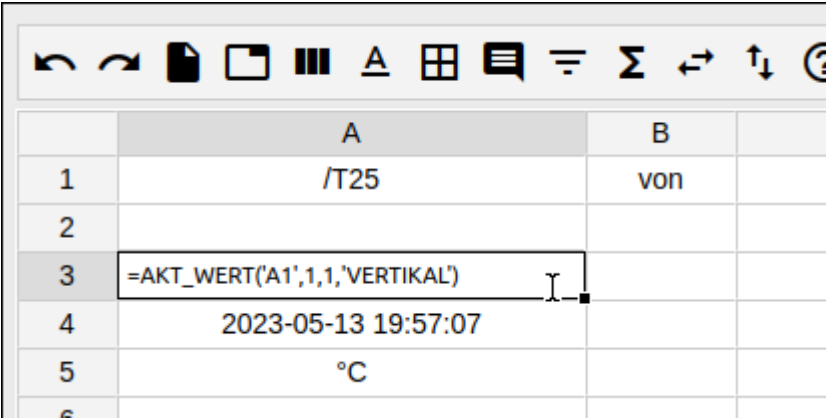


Fig 55: Output of the current value in column A.

When you close the dialog, the formula with the function is retained in the first output cell. A click in cell A1 brings up the formula in the black function area below the menu.



The screenshot shows a spreadsheet with a toolbar at the top. The data is organized in columns A and B. Column A contains the following values: /T25, an empty cell, the formula =AKT\_WERT('A1',1,1,'VERTIKAL'), the timestamp 2023-05-13 19:57:07, and °C. Column B contains the value von in the first row and is empty for the rest. The formula bar at the bottom shows the formula in cell A3.

	A	B
1	/T25	von
2		
3	=AKT_WERT('A1',1,1,'VERTIKAL')	
4	2023-05-13 19:57:07	
5	°C	

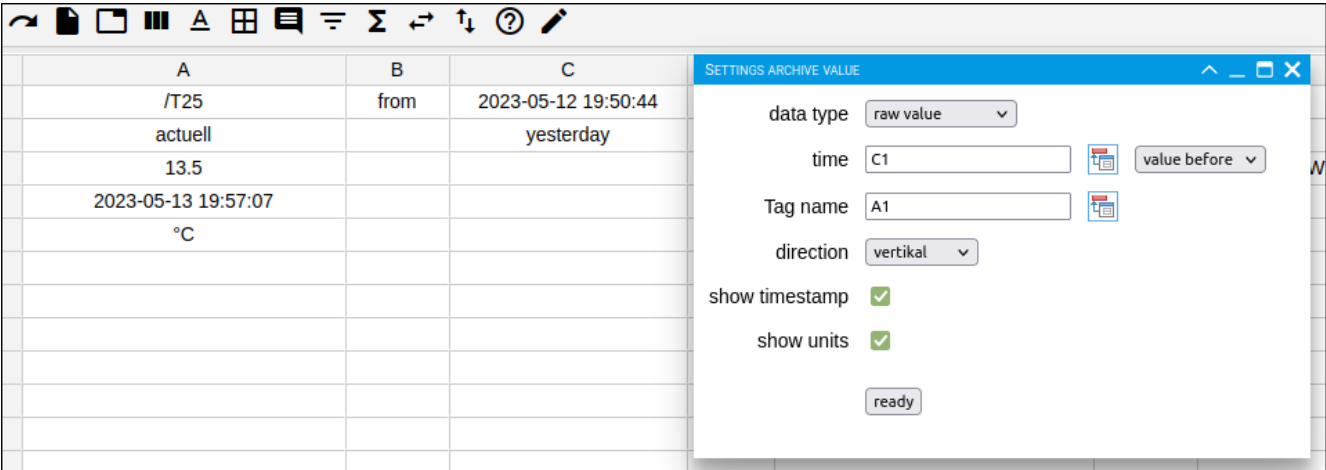
Fig 56: akt\_Wert function in cell A3

This function behaves like any other function in a spreadsheet. When the worksheet is recalculated or a parameter is changed, it is recalculated. If you write a different path & tagname in cell A1, the result of the function is recalculated.

**Archived value:**

In the dialog for the archive value function, we find three additional fields, as we need the time for the archive value and, in addition, information about the data type. The same options are available as data types as we have already seen with the trend group.

The third additional field lets us choose between *value before*, *interpolated* and *value after*. In the example based on a room temperature from Figure 52, we have seen that the DH 1440 has compressed values to 22 values. Since the DH delivers a value for a tag at any given point in time, it is rather unlikely that it actually has someone in its archive for the required point in time. Even if it were to store every delivered value, it would only have one value for every minute. He couldn't deliver for the remaining 59 seconds in between. This problem is solved by interpolation. The field next to the Time field lets us choose whether we want to see the last actual archive value before the requested time, or perhaps the next value after the specified time, or rather the most probable value at exactly the specified time.



The screenshot shows the 'SETTINGS ARCHIVE VALUE' dialog box overlaid on a spreadsheet. The spreadsheet has columns A, B, and C. Column A contains: /T25, aktuell, 13.5, 2023-05-13 19:57:07, and °C. Column B contains: from, and is empty for the rest. Column C contains: 2023-05-12 19:50:44, yesterday, and is empty for the rest. The dialog box has the following fields: data type (raw value), time (C1), Tag name (A1), direction (vertikal), show timestamp (checked), show units (checked), and a ready button. There is also a dropdown menu for 'value before'.

A	B	C
/T25	from	2023-05-12 19:50:44
aktuell		yesterday
13.5		
2023-05-13 19:57:07		
°C		

Fig 57: Dialog for the AW function (archived value)

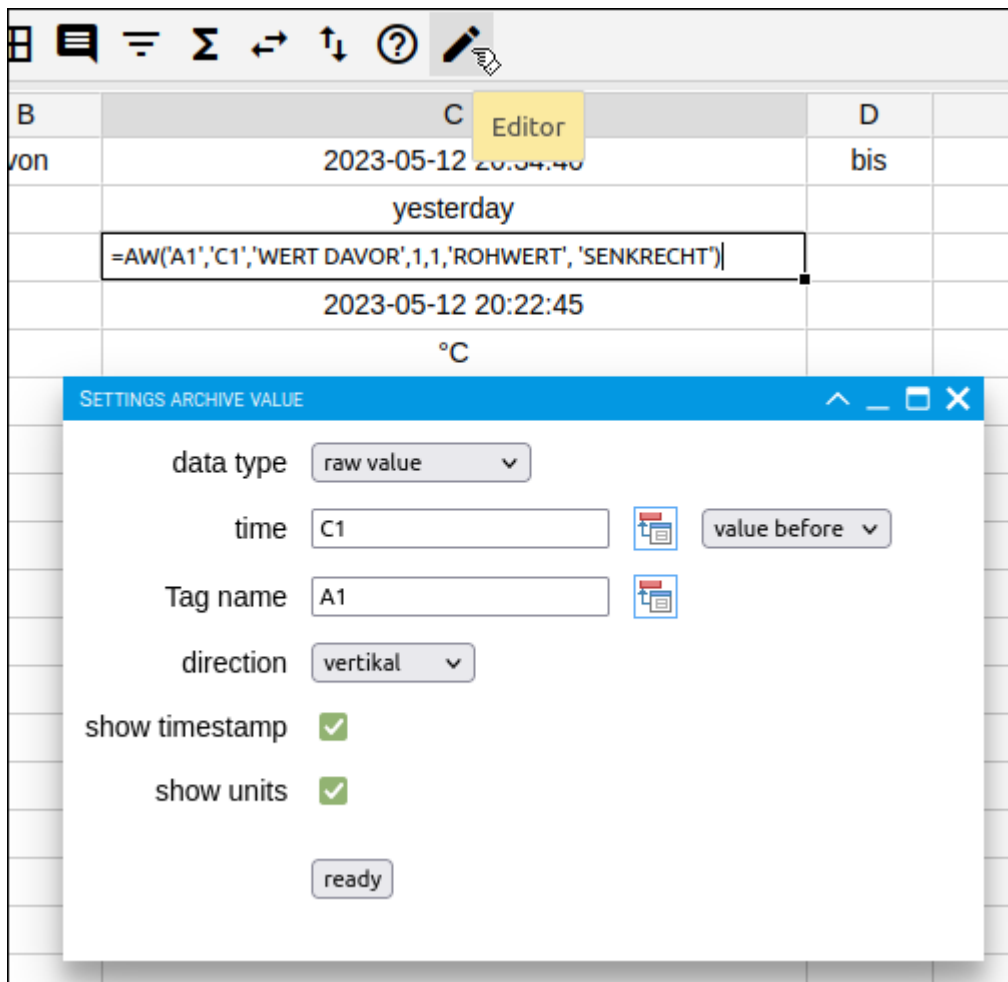


Fig 58: Cell with the function in edit mode

By the way: If you want to edit a function, you can edit it directly in the cell by double-clicking on the cell or you just click on the value and then select the editor icon in the toolbar. The latter will open the appropriate dialog for the function. This should be more convenient than editing the cell directly.

### **Average:**

In addition to the already known fields, you will find three more fields in this dialog. First of all, it is a matter of specifying the period for the mean value. In the example in Figure 58, the entire year 2020 has been selected.

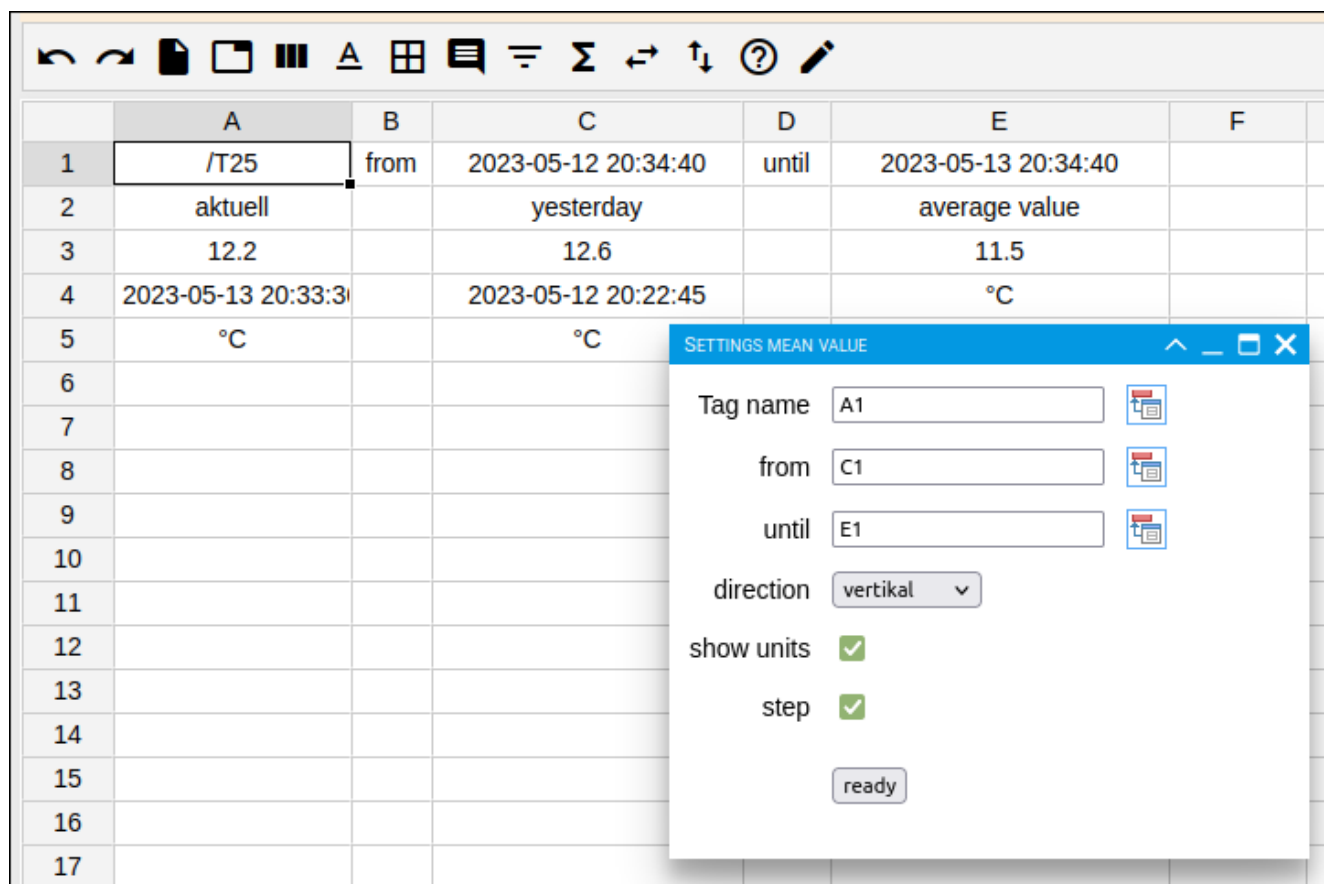


Fig 58: Mean value

Another new field is **step**. As you probably already suspect, you can choose here whether or not interpolated values are used to calculate the mean value.

### Archive values

The dialog for querying archive values over a more or less long period of time is more extensive than the dialogs presented so far. This is also where you can do the greatest nonsense. You should therefore take a closer look at the fields before you click the **done** button. In particular when choosing the time period, things can go wrong, so that the browser can even crash. Just think once how much data is queried and displayed if you enter the year 2002 as the year in the field **from** 2020 instead of 2020. The checkboxes are partly new in this dialog. The Unix time stamp indicates the seconds since 01/01/1970 00:00:00. A time difference in seconds can be calculated very easily using the Unix time stamp. The checkboxes show **vt** and show **vt interpolated** each show an additional column. The column **vt** shows the product of the previous value and the time difference in seconds. The column **vt interpolated** uses the mean value between the previous and the current value instead of the previous value. These two columns can also make further processing of the data much easier for you, since time always plays a major role when processing series of values. For example, when calculating the mean value for a tag, you cannot simply add up all the values in this period and then divide by the number of values. You would then neglect the time intervals between the values and would come to a completely unusable result. The same applies to other statistical calculations.

	A	B	C	D	E
1	/T25	from	2023-05-12 20:34:40	until	2023-05-13 20:34:40
2					
3	ROHWERTE in °C	Zeitstempel	vt	vt interpoliert	Unix Zeitstempel
4	11.7	2023-05-12 20:39:25	12155.999999999998	12625.0	1683916765
5	11.6	2023-05-12 20:50:28	7707.0435	7748.481	1683917428
6	11.5	2023-05-12 21:09:18	13030.029999999999	13065.06	1683918558
7	SETTINGS ARCHIVE VALUES				1683918690
8		from C1			1683918890
9		until E1			1683919553
10		data type raw values			1683920350
11		Tag name A1			1683920549
12		direction vertikal			1683922010
13		show timestamp	show vt		1683924668
14		show Unix timestamp	show vt interpolated		1683925332
15					1683925465
16					1683925796
17					1683925995
18					1683926792
19					1683926924
20					1683926991

Fig 59: Series of archived values

## 2.3. Details

So far, topics such as the configuration of tags or archiving have only been dealt with superficially. This is completely sufficient for everyday use of the DH. However, if you would like to read a little more background information, then you will find it here.

### 2.3.1. Tags, Points, Paths and permissions

The connection between path & tag or path & point can best be illustrated using an example.

Let's assume we operate several wind farms. Each of these wind farms has a different number of wind turbines from various manufacturers.

Employees who deal with technology work with technical drawings and circuit diagrams. Measurements are systematically named there. A measurement with the designation FC386.PV shows that this is a flow control. It is a process variable, ie the actual value of the medium flowing through. TC386.SP, on the other hand, is the default value (setpoint) of the controller.

Controlling employees are not interested. You can't do anything with the name. With the term lubricant consumption, however, you can do more. Since every wind turbine requires lubricant, it is a convenient matter for these people to know the amount of lubricant from each wind turbine with this designation. The technicians, on the other hand, are annoyed because they have to look up the correct measuring point designation every time. The solution for this is to use paths, tags as links and points.

A point always represents a data point. In the example above, this is the measurement FC386.PV or FC386.SP. When a point is generated, a tag with the same name is always generated, which points to the point as a link. So the technicians are satisfied because they can read the point about the tag of the same name. To make our colleagues from Controlling happy too, we simply create a second tag that also points to Point FC386.PV.

The combination of path and tag name must be unique. Although we set up a lubricant consumption tag for each wind turbine, this is not a problem because the path is different every tag.

Example:

[/ Windpark\\_Hintertupfingen / Windrad\\_3 / Lubricant consumption](#)  
[/ Windpark\\_Hintertupfingen / Windrad\\_4 / Lubricant consumption](#)  
[/ Windpark\\_Nordeifel / Windrad\\_1 / Lubricant consumption](#)  
[/ Windpark\\_Nordeifel / Windrad\\_2 / Lubricant consumption](#)

The assignment of rights is also based on the paths here. A user who is in the UNIDB an authority on the path [/ Windpark\\_Nordeifel](#) has can also see the tags and points whose path also with [/ / Windpark\\_Nordeifel](#) starts, but not the tags, their path with [/ Windpark\\_Hintertupfingen /](#) starts.

### 2.3.2. Point and tag configuration

Up to 28 parameters are required to configure a point. We will take a closer look at the parameters that may be of interest to you here.

The parameters **scale min**, **scale max**, **property 1 - 5**, **Point type** and **last change** are only of interest to an administrator or developer. So we won't go into it any further here.

**Point\_ID** and **Tag\_ID** are unique numbers that cannot be changed. They are not reassigned even after deleting a tag or points.

The **path** parameter should be clear by now. Since a path always starts with a /, the shortest possible path consists of a /.

The **Pointname** or **Tagname** can be changed at any time. Since the points and tags in the DH and in the unidb are recognized by their Point\_ID or Tag\_ID, they also appear in all documents with the new name after a name change.

The **description** is a supplement to the point name. Often a much longer description of a point is necessary than can be put into a point name that is as short as possible.

The **units** should not be missing from any point. This is the only way to correctly interpret the values. Example: A liquid level can be measured in%, m<sup>3</sup>, or mm. Without a unit, the measured value is of no use to us. For measurements without a unit, such as a pH value or a counter reading, this parameter naturally remains empty.

The **interface** is the program that writes the values for the point into the DH. If you want to know where the values of the point come from, then the interface name provides a good clue. Unless the developer has chosen some mysterious name. ;-)

The **step** parameter is already known. It is in the point configuration because it is not only needed to display a trend, but also to calculate mean values. The parameter is also used to preset a trend. The trend of a point for which the step parameter is set to 1 is initially always displayed in the form of a step. A database always tends to display values in the float format with an extremely large number of decimal places. This has a very annoying effect on the display in the documents. In order to display the values with a reasonable number of decimal places, these can be limited using the **decimal places** parameter.



TAG DETAILS
^ \_ □ ×

Tag_ID	path	Tagname
82	/	Wirk_gesamt_PV

Point on which the Tags are based:

Point_ID	path	Pointname	description	units
82	/	Wirk_gesamt_PV	Wirkungsgrad gesamt SC	%

interface	step	decimal places	scan	mean values
calc	0	2	1	1

Archiv	compression	minarch	scale min	scale max
1	0	0	0	20

property 1	property 2	property 3	property 4	property 5
14				

Info	Point type	intervall	first value	last change
akt(16)/akt(28)/53.76*100	calc	60	2009-10-21 10:00:00	2019-10-26 12:43:11

Fig 60: Configuration of a calculated point

If **scan** = 1, then the point is active. If **scan** is set to 0, the acquisition of new values for this point is switched off. This means that the values of a point are retained even if the measurement on which it is based no longer exists. If you are wondering why you are not seeing any new values for a tag, this parameter is the first place to look for the cause of the missing values.

**mean values** = 1 means that hourly and daily mean values are calculated and archived for this point. At the same time, the respective minimum and maximum values for the relevant period are determined and archived. For a point that is configured with **mean values** = 1, all types of values are available in the spreadsheet and in the trends. It makes no sense to switch this option on for all points. For a point, which represents a counter reading, an average value is utter nonsense.

**Archiv** = 1 means that the values for this point are archived. This is the case with almost all points. There are only a few cases in which a value is only needed for further calculation and can then be discarded.

The value in the **compression** (compression) field indicates how far a new value must change compared to the previous value in order for it to be written to the archive. There are measurements that show some noise. In other words, if you ask for a measurement 10 times within a very short period of time, you may get 7 different values, but they differ only minimally from each other. Example: A temperature sensor supplies an analog value, which is converted into a usable number via an analog-digital converter. The resolution has its limits, of course. If the A / D converter resolves to two decimal places and the analog signal constantly fluctuates by a third decimal place, then the D / A converter delivers 25.34 times and then again 25.33 and again 25.34, ... That would only be for unnecessarily many values in the archive. If you consider the fact that the measuring accuracy of the sensor is, for example, +/- 1 ° C, then you can set the compression to 0.3 with a clear conscience. This avoids unnecessary archive values and the server thanks it with a significantly better performance.

If a 0 was entered for the compression, the values that have not changed compared to the previous value are still filtered out. Imagine that we are dealing with a point that shows us whether a door is open or closed. The only possible values for the point are 0 or 1. This means that a new value is only written to the archive when the door is actually opened or closed. If the door is only opened three times in 24 hours, only 6 values are written to the archive, although the interface sends the current status to the DH every two seconds. This means that only 6 of the 43200 values are archived.

The parameter **minarch** specifies the time in seconds after which a value is definitely written to the archive. This happens regardless of whether the value has changed or not. what looks like nonsense at first makes sense. This makes it easier to calculate the mean values and improves the generation of

trend lines. In addition, you can easily see whether the point is still "alive" if fresh values are only rarely supplied.

Another interesting parameter is the **Info** field. What this parameter is used for depends on how the interface is structured. Interfaces that serve as parsers for websites often use this field to specify the URL. In the example from Figure 60, this field contains the formula for calculating the values. The interface is called `calc`, from which we can conclude that we are dealing with calculated values. The `act()` function is used twice in the formula. This function returns the current value of the point whose `Point_ID` is given in brackets. By specifying the `Point_ID` instead of the point name, it is ensured that the calculation still works properly even after changing the point name or the path.

**Interval** indicates the time **interval** at which the value is read from the interface and sent to the DH. You should not rely on this parameter, however, because the interfaces themselves also run according to a defined interval and supply all points assigned to them with values. If the interval is important to you, it is advisable to take a look at the documentation for the interface.

One last interesting parameter is the **first value**. It may be nice to know when data is being collected for the point, but it actually serves a different purpose. Hardly anything strains the patience of a user more than a database Query that comes to nothing. All DH functions therefore take this parameter into account when querying archive values. The time of the first value is also taken into account when creating trends.

### 2.3.2. Archiving of values

The interfaces always write the values in a table which only contains current, non-archived values. There they are read by a program called a compressor and, if necessary, written to the archive. After the compressor has read the values, it deletes almost all values for the relevant point from the table. A maximum of two values remain in the table for each point. Thus, the DH documents always use this table when it comes to displaying the current value of a point. The compressor decides on the basis of the compression setting which values are sent to the archive and which are discarded.

For points with the configuration **mean values** = 1, *hourly* and *daily mean values* as well as the corresponding minimum and maximum values are also written to the archive. In addition to these statistical values, the *vt* and *vt* are also archived *interpolated*.

### 2.3.3. Functions

In addition to the already described DH functions from the spreadsheet, the DH also uses other special functions for calculated points. If you intend to create your own DH documents, then these functions make it easier for you to create calculated expressions that you can use in the document instead of tags.

function	description
ZP (Point_ID, time stamp)	Returns the Unix time stamp of the value that was last written to the archive before the specified time stamp. The function accepts the time as a time stamp in the format YYYY-MM-DD hh: mm: ss. Alternatively, a relative time can also be specified. <i>Now</i> always stands for the current time. If you want to use the current time minus one hour, then write the relative time <i>now - 1h</i> . Days are given as <i>d</i> , minutes as <i>m</i> , and seconds as <i>s</i> .
ZS (Point_ID, time stamp)	Returns the same result as before, but in a readable form in the format YYYY-MM-DD hh: mm: ss.
AW (Point_ID, time stamp)	Returns the last archive value whose time stamp is less than or equal to the specified time stamp.
intp (Point_ID, relative point in time)	Returns the most likely value of a point at the given point in time.
MW (Point_ID, start time stamp, end time stamp)	Returns the mean value of the point for the specified period, taking into account the temporal relationships.
act (Point_ID)	Returns the latest value for the specified point.

## 2.3.4. Working with trends



Fig. 61 simple trend

You should already be familiar with the controls 1 - 6 from the previous chapters. Item 7 allows you to preset the y-axis for the displayed trend curves. A small window will open with a table of all the tags you are currently viewing. In the fields min and max you can make your personal presettings for the y-axis.

The 'TAG DETAILS' window is a small dialog box with a title bar containing a maximize, minimize, and close button. It contains a table with the following data:

Point_ID	Tagname	description	min	max
250	T25	außen oben	-8	12

Below the table are two buttons: 'cancel' and 'accept'.

Fig. 62 Personal preset for Tag T25.

Point 8 is really interesting. Hover your mouse pointer over the icons and brief descriptions of the icons will appear, as shown in the image below.

The best way to get familiar with it is to play around with the features. Don't worry, you can't break anything.

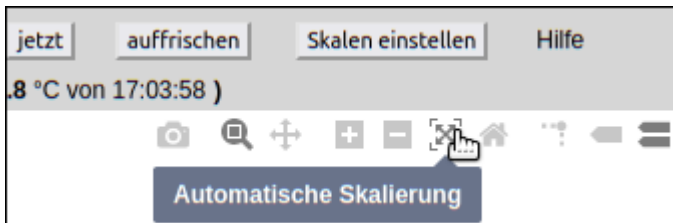


Fig. 63: Symbols at the top right of the graphic.

If you call up a trend from a group, the remaining tags from the group are offered for selection below the graphic. Here's an example:

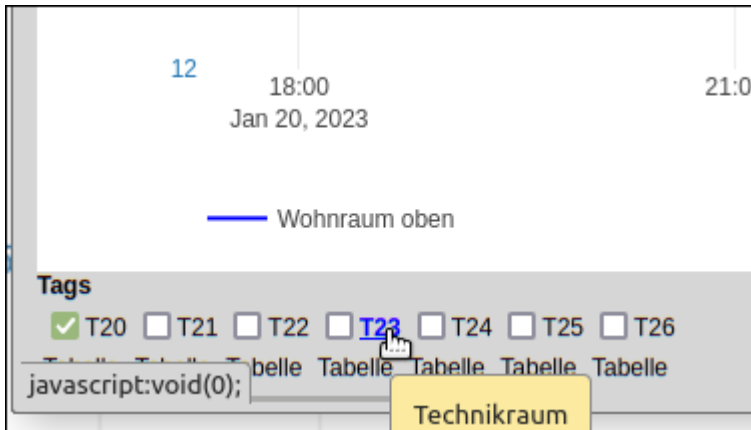


Fig. 64 additional selectable tags

If several tags were selected, they each appear in their own color and with their own y-axis in the graphic. If you move the mouse pointer over the graph, auxiliary lines are displayed for the trend that is closest to the mouse pointer. You can also see the values that the tags have at the marked time as text in the small colored areas.

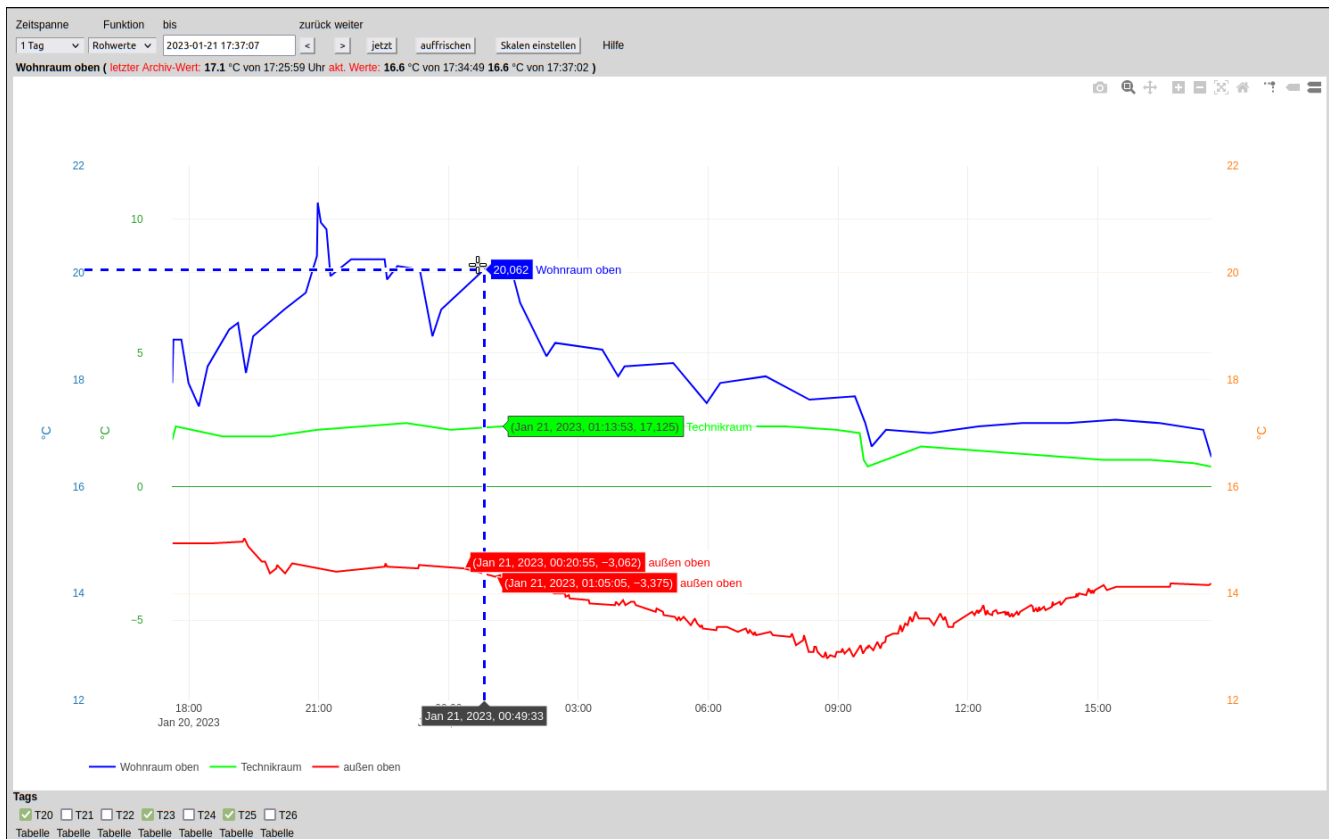


Fig. 65: Display of several trend curves.

To view an area more closely, i.e. enlarged, simply move over the desired area while holding down the left mouse button. The graphic is then focused on this area. To see the entire area again, simply double-

click anywhere in the graphic.



Fig. 66: Mark area



Fig. 67: Enlarged view of the previously marked area.

In Fig. 64 you can see that further tags are listed below the graphic, which can also be displayed as a trend. Below each check box is a Table link. Each of these links opens a pop-up window, which displays the values of the displayed trend in tabular form.

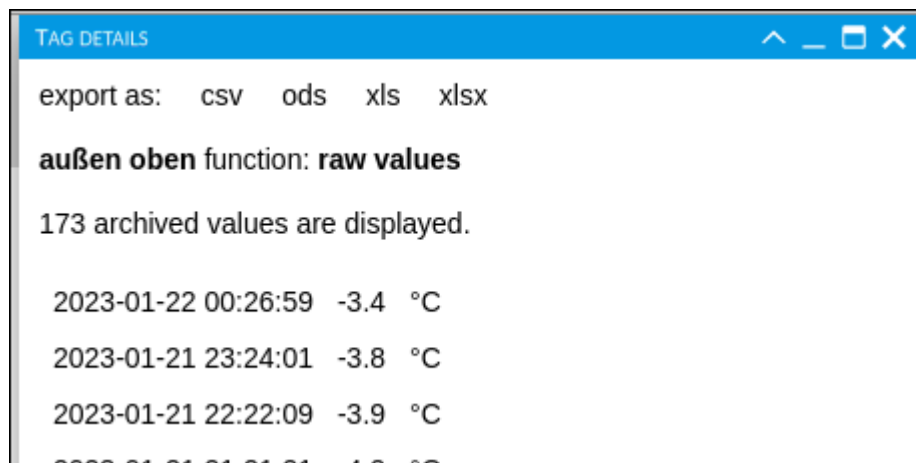


Fig. 68: Table with the values of the trend curve.

This table can be exported via the links in the first line. Please note that exporting to spreadsheet formats (ods, xls and xlsx) is very time consuming. If the third line shows you that the table contains more than 4,000 - 5,000 values, then you will have to be patient for the export. Since the values are always exported without formatting, regardless of the selected format, the export in csv format is not a disadvantage and is therefore always preferable.

A few more thoughts on setting the time period and the data type (function):  
If a value is written to the archive for one day per minute, then that is 525,600 values per year. If you have an ordinary monitor with a width of 1920 pixels, you can only display 1920 pixels side by side. This means that in our case more than 273 values are displayed on a width of one point.  
The server delivers the large amount of data within a second, but the trend is calculated on your computer using JavaScript. With large amounts of data, it can happen that your computer despairs. It is good practice to query the dMinMax values instead of the raw data. dMinMax means that two values are displayed per calendar day. For a period of one year, that is 730 values, which are distributed over your 1920 pixels. Even if you display the hMinMax values instead of dMinMax, there are only 17520 values, i.e. almost 10 values per pixel. That's still a manageable number compared to the raw data. And the nice thing about it is that the quality of the trend curve does not deteriorate. All peaks in the curve are displayed.