COPADATA
do it your way

zenon manual

Reporting

v.7.00

COPA-DATA
do it your way

Contents

1. Welcome to COPA-DATA help

GENERAL HELP

If you miss any information in this help chapter or have any suggestions for additions, please feel free to contact us via e-mail: documentation@copadata.com (mailto:documentation@copadata.com).

PROJECT SUPPORT

If you have concrete questions relating to your project, please feel free to contact the support team via e-mail: support@copadata.com (mailto:support@copadata.com)

LICENSES AND MODULES

If you realize that you need additional licenses or modules, please feel free to contact the sales team via e-mail: sales@copadata.com (mailto:sales@copadata.com)

2. Reporting

The integrated reporting in zenon serves as documentation, evaluation and presentation of process data on the basis of online and archive data from zenon. Reporting is divided into:

- ▸ Report Generator (on page 6)
- ▸ Report Viewer (on page 210)

Neither reporting variant is available with Windows CE.

> **License information**
>
> ▸ In the Editor: Both must be licensed.
>
> ▸ In Runtime:
> **Report Viewer:** In the Runtime always included in the standard package.
> **Report Generator**: Integrated as standard for display of reports only. If you want to write archive values or set values with the **Report Generator**, you also need a license in the Runtime.

zenon Analyzer is available for comprehensive, platform-independent reporting of the process level through to ERP. Your COPA-DATA sales partner would be happy to inform you about this.

3. Report Generator

The Report Generator documents, evaluates and presents process data on the basis of online and archive data. The report generator works on a cell-orientated basis. Each cell can be assigned certain functions and formats.

Reports are configured using the **report** (on page 8) screen. The data is filtered, calculated, issued and displayed via report functions (on page 48). These are entered using dialogs or manually. Values can be read and - if permitted on the screen - also edited. The report generator meets the requirements of rule sheet and ATV H260.

PROJECT MANAGER CONTEXT MENU

Menu item	Action
`New report`	Creates a new report in the list in detail view.
`Import existing report...`	Opens the File Manager to select an existing file.
`Editor profile`	Opens the drop-down list that includes pre-defined editor profiles.
`Help`	Opens online help.

Info

The status bits NORM and N_NORM are not displayed in the report. The status bit NORM cannot be changed using the report.

3.1   Engineering in the Editor

To be able to use reports in Runtime, the following must be carried out in the editor:

▶   A report type screen (on page 8) is created

▶   A report (on page 10) is created and configured

▶   A switching function (on page 164) is configured in Runtime

Recommended procedure for planning the report:

▶   Create report (on page 10)

▶   Format report (on page 23)

▶   Create archive filter (on page 45)

▶   Configure report functions (on page 48)

3.1.1 Create screen of type Report Generator

In the type report screen, the reports created with the report generator are displayed in Runtime and configured with the corresponding rights. The display size depends on the template size selected. The report that is displayed can be determined in screen switching or in Runtime.

In order to create a report screen:

1. Select the `New screen` command in the `Screens` node

2. In the `Screen type` column, select the `Report` drop-down list

3. in the menu bar, select *Control elements -> Add template*

4. A report with standard settings is created

5. If necessary, add further control elements from the `Control elements` menu

6. Create a screen switch (on page 164), to select the screen in Runtime

Parameters	Description
Add template...	Opens the dialog for selecting a template for the screen type.
	Templates are shipped together with zenon and can also be created by the user.
	Templates add pre-defined control elements to pre-defined

	locations in the screen. Elements that are not necessary can also be removed individually once they have been created. Additional elements are selected from the drop-down list and dragged onto the screen. Elements can be moved on the screen and arranged individually.
Buttons	Buttons for controlling in the Runtime.
`Refresh`	Recreate build report completely (values and display).
`Print`	Print report in Runtime.
`File`	Select report file and change filter conditions.
`Export`	Export report.
`Save archive`	Write changed values to the archive.
`Save report`	Report is saved in Runtime (*.xrs). **Attention:** When saving, formulas and functions are replaced by the current values. The functions in these reports (.xrs files) are no longer available. These reports can also not be edited in the editor anymore. **Recommendation:** Set original report files to "write-protected" status and save changes with `Save as` in a new file.
`Save report as`	Report is saved under a new name during Runtime (*.xrs), formulas and functions are replaced by current values.
`Read all values`	All values are read in again from the driver.
`Read selected values`	Selected values are read in again from the driver.
`Write all values`	All values are sent to the driver.
`Write selected values`	Selected values are sent to the driver.
`Line up`	Scroll text in list element up
`Line down`	Scroll text in list element down

Column right	Scroll text in list element to the right
Column left	Scroll text in list element to the left
Page up	Scroll text in list element up
Page down	Scroll text in list element down
Page right	Scroll text in list element to the right
Page left	Scroll text in list element to the left
Report	Control elements for report display.
Report	Report window to display the report.
Filter profiles	Buttons for filter profiles.
Profile selection	Display of the available filter profile and input for new profile names.
Save	Saves current setting as a profile.
X (save)	Deletes selected profile.
Import	Exports profiles.
Export	Imports profiles.

3.1.2　Create a new report

Before a report is created, the report printer should be defined because the page formatting of a report depends on the printer that has been set. Subsequent changing of the printer can necessitate reformatting of the existing table.

In order to create a new report:

▸　Navigate to the report generator node in the project manager

▸　Select the New report command in the context menu

▸　a new report is created in the detail view

▶ An empty worksheet is created in the main window
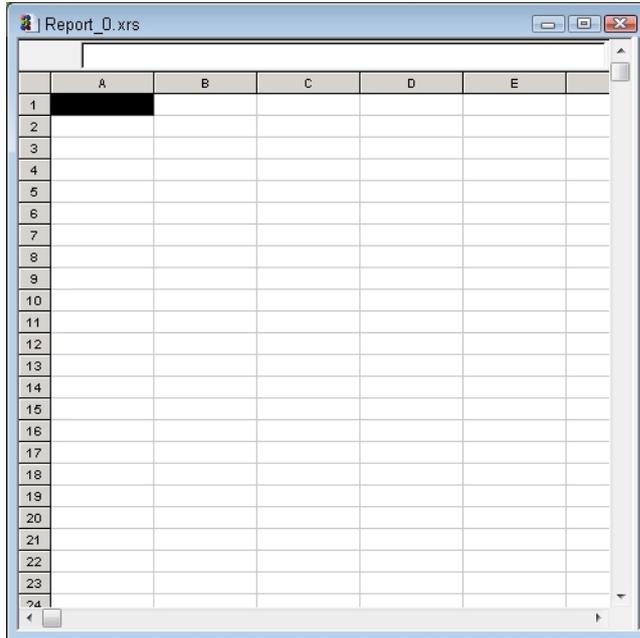
Functionalities of the worksheet:

▶ Assignment of process information using report functions (on page 48)

▶ Formatted display of content of cells, columns and rows

▶ Subtotals

▶ Assignment of editing blocks

💡 Info

For the input in cells: A decimal value can be entered with a `comma` as well as with a `period` as a separator. The decimal separator is automatically changed to a `period`.

🔍 See also

Report menu

The **Report** menu is shown when a **report** is selected:

Parameters	Description
Save report	Saves all changes in the active report. Changes in other reports are not saved.
Export report functions	Exports all report functions from the active report to a text file. Structure: Column / row =function(parameter)
Redraw report	Draws a new report.
Calculate report	Calculates a new report.
Set up page (on page 24)	Opens the dialog to set up the print page.
Print report	Prints the report using the printer defined in the standard settings..
Printing area:	Defines printed areas or cancels them: ▶ Define printing area: Defines marked areas of the table as a print area. ▶ Cancel print area: Cancels defined print area
Report - Page view	Switches from layout view to page view. The view can be zoomed using the left mouse button. Leaving page view: Press Esc key.
Report file info (on page 21)	Opens the dialog to administer the report file information.

Menu Edit

When a report is selected, the **Edit** menu offers special functions for reports:

Command	Description
`Cut`	Cuts out highlighted area and copies it to the clipboard.
`Copy`	Pastes the contents of the clipboard.
`Paste`	Copies highlighted areas to the clipboard.
`Advanced paste`	Provides advanced paste options for cell operations:
▸ `All`	Inserts everything from the clipboard.
▸ `Formulas`	Only inserts formulas
▸ `Value`	Only inserts values. The relationship to formulas is lost.
▸ `Format`	Only accepts the format.
`Fill`	Copies the content of the first cell of a marking and transfers it to all further marked cells.
▸ `Down`	Filling is carried out in a downward direction.
▸ `To the right`	Filling is carried out from left to right.
`Remove`	Removes from highlighted cells:
▸ `All`	Everything.
▸ `Contents`	Contents only.
▸ `Format`	Formats only.
`Delete column/row`	Deletes highlighted columns or rows.
`Insert column/row`	Inserts new columns or rows. Insertion is carried out above the row or to the left of the column. As many rows/columns are inserted as are highlighted in the worksheet.
`Search/replace`	Opens the dialog (on page 21) for searching and replacing in a worksheet.

Format menu

The Format menu provides properties and functions for formatting the report table:

Parameters	Description
Cells	Formatting options for cells (on page 30):
	▸ Font
	▸ Pattern
	▸ Frame
	▸ Alignment
	▸ Numbers
	▸ Lists
Row	Formatting options for rows:
	▸ Row height: Opens dialog to define the row height
	▸ Row format: Opens dialog for all row properties (on page 44)
Column	Formatting options for columns:
	▸ Column width: Opens dialog to define the column height
	▸ Column format: Opens dialog for all column properties (on page 43)
Report	Opens the dialog (on page 27) to format the report.
Function assistant	Opens assistant to create report functions (on page 48).
Filter	Opens report filter (on page 45) to select archives.

Toolbar Report Generator

A toolbar is available in the Editor for the Report Generator. If it is not displayed yet, you can load it:

1. Open the menu `Options -> Settings -> Tab: Toolbar`

2. Tick the checkbox next to Report Generator

3. confirm this with `OK`.

Symbol (from left to right)	Function
`Print report`	Prints the report on the default printer.
`Report - Page view`	Opens the report in a preview. The report can be enlarged or minimized in the preview (zoom), you can leaf through it and it is printable.
`Redraw report`	Refreshes the display.
`Recalculate functions`	Recalculates the formulas in the report.
`Function assistant`	Opens a dialog that helps with the configuration of functions.
`Font`	Opens the dialog for cell formatting - focused on the **Font** tab.
`Text alignment`	Opens the dialog for cell formatting - focused on the **Alignment** tab.
`Frame`	Opens the dialog for cell formatting, with focus on the tab **Frames**.
`Pattern`	Opens the dialog for cell formatting - focused on the **Pattern** tab.
`Symbol bar options`	Clicking on the arrow opens the submenu: `Active`: Toolbar is displayed. If the toolbar is not displayed, it can be activated using the Menu *Options -> Toolbar*.

Detail view of context menu and toolbar

Menu item	Action
New report	Creates a new report in the list in detail view.
Open report	Opens the selected report or focuses on the one that is already open.
Create standard function (on page 164)	Creates a zenon function for report generator. The action is documented in the output window.
Jump back to starting element	If you entered the list via function **linked elements**, the symbol leads back to the start element. Only available in the context menu when all linked elements are opened.
Copy	Copies selected entries to the clipboard.
Paste	Pastes the contents of the clipboard. If an entry with the same name already exists, the content is pasted as "Copy of".
Delete	Deletes selected entries.
Import existing report...	Opens the File Manager to select an existing file.
Remove all filters	Removes all filter settings.
Edit selected cell	Opens the selected cell for editing. The binocular symbol in the header shows which cell has been selected in a highlighted line.
Replace text in selected column	Opens the dialog for searching and replacing texts.
Properties	Opens the property window.
Help	Opens online help.

Creation of table window

Reports are displayed in two parts when being edited in the Editor in the main window:
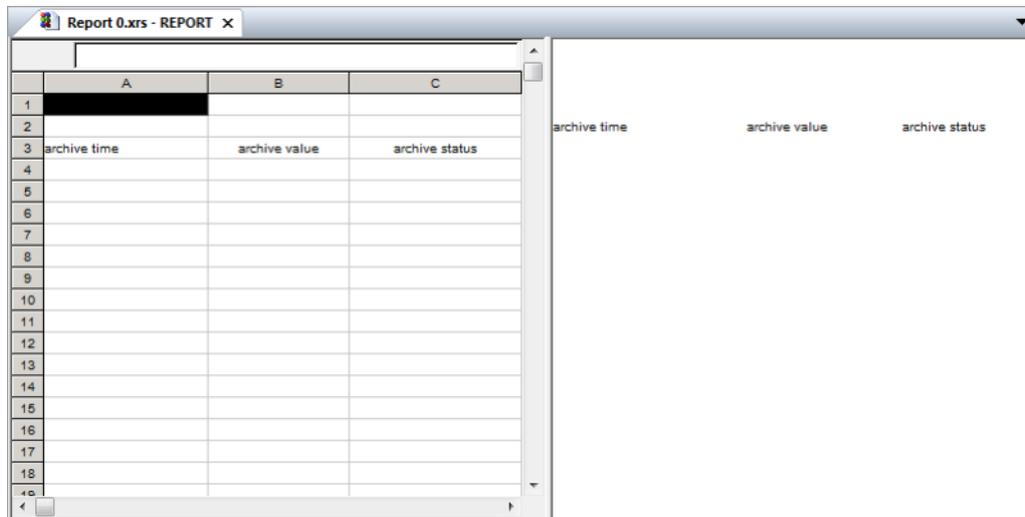
- ▶ The left area contains the table for the draft

- ▶ The right part displays a preview of the output

The size of the areas can be amended individually. To do this, move the mouse to the right edge of the table scroll bar until the mouse pointer becomes a moving tool and move the scroll bar to the desired position.

INPUT ROW

The table has an input row at the upper edge. If, in the report Properties (on page 28), the `Edit in the cell` property is activated, this row is not displayed. Input must then be made in the row itself.

Data sheet context menu

Menu item	Action
Cut	Deletes highlighted entries and places them on the clipboard.
Copy	Copies selected entries to the clipboard.
Paste	Pastes the contents of the clipboard.
Expand insert	Enables the contents of the clipboard that are to be inserted to be selected: ▷ All: transfers complete contents ▷ Formulas: Transfers only formulas ▷ Value: Transfers only the value ▷ Format: Transfers only the format
Fill	Copies the content of the first cell of a marking and transfers it to all further marked cells. It is possible to select if they are filled downwards or to the right.
Remove	Makes it possible to select the content of a cell to be deleted: ▷ All: Deletes the complete contents with format ▷ Contents: Removes the content, but leaves the format ▷ Format: Deletes only the format, but leaves the content
Insert column row	Inserts a new row above a highlighted row or a new column to the left of a highlighted column. If several rows/columns are highlighted, the number of highlighted rows/columns is inserted.
Delete column row	Deletes the highlighted rows or columns
Report...	Opens the dialog to format the report.
Cell...	Opens the dialog to format the a cell.
Column...	Opens the dialog to format the columns.
Row...	Opens the dialog to format the rows.
Functions...	Opens the assistant to create report functions (on page 48). Note: The assistant offers a selection of over 150 functions. Further functions can be entered manually into the input line of the data sheet.

3.1.3   Printer settings

The report layout (side margins, page sizes) depends on the printer configured. You can find the settings for the report printer via *File -> General configuration -> Standard -> Printer for values to be given and protocols.*

Here you define

▶   the printer and

- Paper format of the print
- Paper source of the printer
- Direction of print

> ⚠ **Attention**
>
> *The desired printer must be selected before creating the report, because the page formatting of a report depends on the printer that has been set. Subsequent changing of the printer can necessitate reformatting of the existing table.*

PRINTER CONFIGURATION
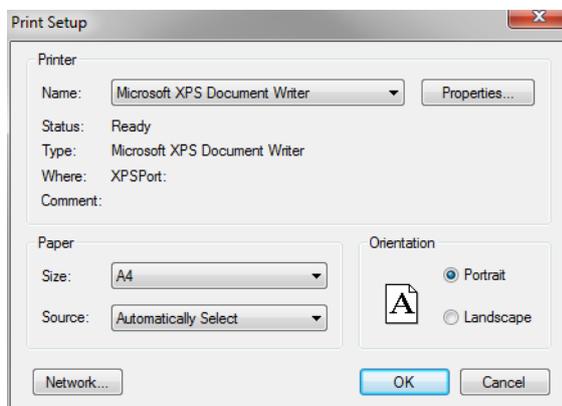
Parameters	Description
Printers	Settings for the printer.
Name:	Selection of the printer from the drop-down list. The list contains all printers configured in the operating system.
Properties...	Opens printer configuration dialog.
Status:	Display printer state. For information only.
Type:	Display printer type. For information only
Location:	Display the location of the printer if configured. For information only.
Comment:	Display comment about printer if configured. For information only.
Paper	Configuration of the printout.
Size	Select paper format from drop-down list.
Source	Select paper feed from drop-down list.
Alignment	Select paper alignment. Possible parameter: • Portrait Format • Landscape format
Network	Opens dialog for selecting a printer in the netwok.
OK	Accepts configuration and closes dialog. With this printing is started in the Runtime.
Cancel	Discards configuration and closes the dialog. In the Runtime this also cancels the printout.

3.1.4   File info

In the *Report -> Report file info* dialog, information on the respective report file can be saved.
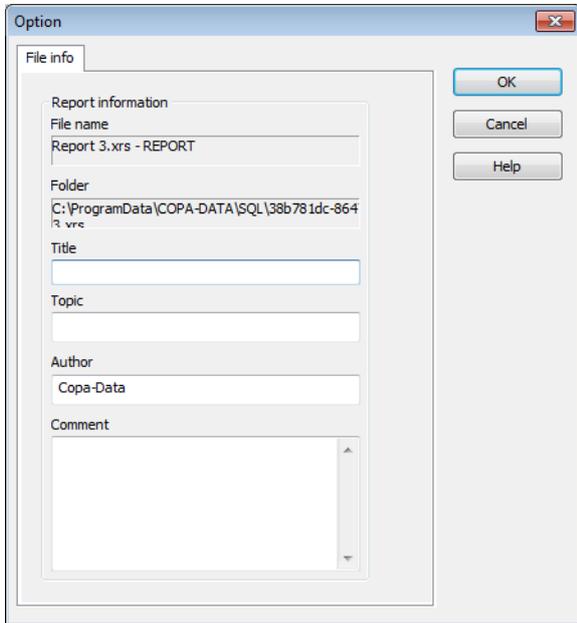
Parameters	Description
File name	Name of the report file.
	Information only; cannot be changed here. Can be changed in the detail view.
Folder	Save location of the report file.
	Information only; cannot be changed here.
Title	Free text input
Topic	Free text input
Author	Free text input
Comment	Free text input

3.1.5   Finding and replacing in the report

To find formulas, character strings or values and replace these:

1. Activate the desired report

2. Select **Find/replace** in the **Edit** menu.

3. The dialog to find and replace is opened
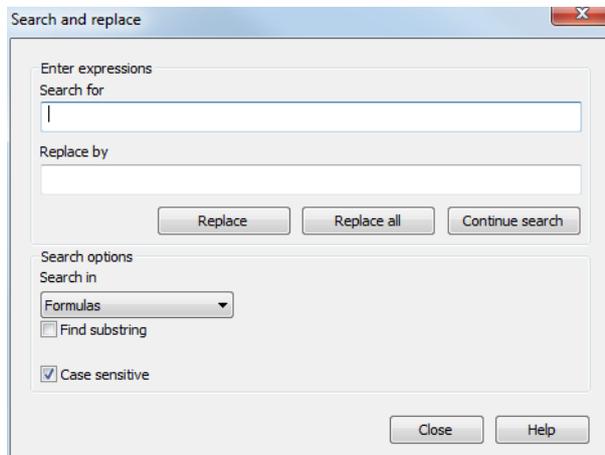
Parameters	Description
Enter terms	Input of character string for finding and replacing.
Search for	Character string that is being searched for.
replace with	Character string that replaces the one that is found.
Replace	Replaces currently displayed finding with sequence from **Replace with**.
Replacing all	Replaces all currently displayed findings with sequence from **Replace with**.
Find next	Looks for next instance of the character string being searched for.
Search options	Search options
Find in	Select from drop-down list: ▸ Formulas: Searches in formulas ▸ String: Searches in character strings ▸ Value: Searches in values
Search partial string	Also searches partial areas of cells.
Note capitalization	Note capitalization during the search.
Close	Closes the dialog.
Help	Opens online help

3.1.6   Formatting

Before creating a report, the necessary formatting should be carried out:

▸   **Page (on page 24):** Set up print properties for the page

▸   **Report (on page 27):** Set up report properties

▸   Cell (on page 30): Define the formatting of the cells (direction, border, font, pattern, type, lists)

▸   Columns and cells (on page 42): Formatting for selected columns or cells

You can find the settings in the `Report` and `Format` drop-down lists in the menu bar or in the context menu of the worksheet.
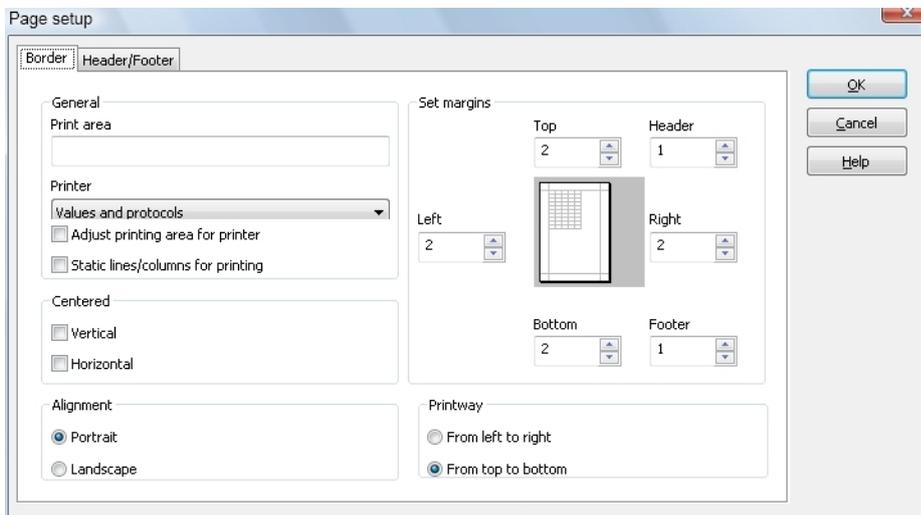
Hint: Ensure that reports with large amounts of data have sufficient lines/columns for output.

Set up page

The `Set up page...` command in the `Report` menu opens the dialog to configure the report page. This contains the two tabs:

▶ Borders (on page 24)

▶ Headers and footers (on page 26)

Borders

Parameters	Description
Set side borders	Define print margin independently of the printer setting (on page 19)
top	Distance of the report from the upper edge.
below	Distance of the report from the lower edge.
Left	Distance of the report from the left edge.
Right	Distance of the report from the right edge.
Header	Distance of the header from the text. Possible values: 0 to 20; 0 = hidden. The value selected must be large enough to be able to display the selected font size under header and footer (on page 26).
Footer	Distance of the footer from the text. Possible values: 0 to 20; 0 = hidden. The value selected must be large enough to be able to display the selected font size under header and footer (on page 26).
General	
Printing area:	Defines the print area based on the cell A1. Input in the text field: Coordinates of the end value of the range. Example: D5 defines the print area of the cells A1 to D5.
Printers	selection of the printer for the report. Clicking on the drop-down list opens the list of all defined printers available in zenon printer settings. Default: Values and protocols
Adjust printing area to printer	Active: The printing area of the report is adjusted to the settings of the selected printer.
Static lines/columns for printing	Active: Static lines and columns are also printed.
Centering	Orientation of the report on the print page

`vertical`	Centers the report vertically.
`horizontal`	Centers the report horizontally.
`Alignment`	Settings of page format
`Portrait Format`	Prints in portrait format.
`Landscape format`	Prints in landscape format.
`Direction of printing`	Sequence of the page print out for multiple pages
`From left to right`	Print out several pages from left to right.
`From top to bottom`	Print out several pages from top to bottom.

Headers and footers
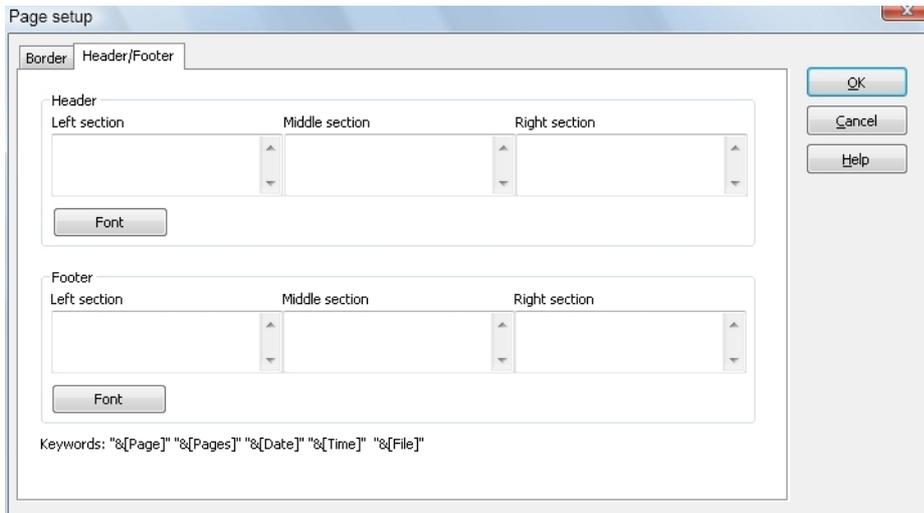
Parameters	Description
`Header`	Definition of the header area
`– Left part`	Left text for the left third of the header.
`Centre part`	Centered text for the center of the header.
`Right part`	Right text for the right third of the header.
`Font`	Opens the dialog to select the font, font size and emphasis of the footer text.
`Footer`	Definition of the footer area
`– Left part`	Left text for the left third of the footer.
`Centre part`	Centered text for the center of the footer.
`Right part`	Right text for the right third of the footer.
`Font`	Opens the dialog to select the font, font size and emphasis of the footer text. Note that the settings for the header/footer can be selected in borders (on page 24) in such a way that the font size can be displayed in full.
`Key words`	Key words for headers and footers
`&[Page]`	Page number
`&[Pages]`	Total number of pages
`&[Date]`	Printing date
`&[Time]`	Printing time
`&[File]`	File name

Format report

For formatting the whole report, there are settings for:

▶ General (on page 28): General settings for the report

▶ Columns: (on page 29) General settings for columns, such as number etc.

▶ Rows (on page 30): general settings for rows, such as number etc.

To format, select either the `Report` command in the `Format` menu or the `Format report` command in the context menu of the report sheet.

General
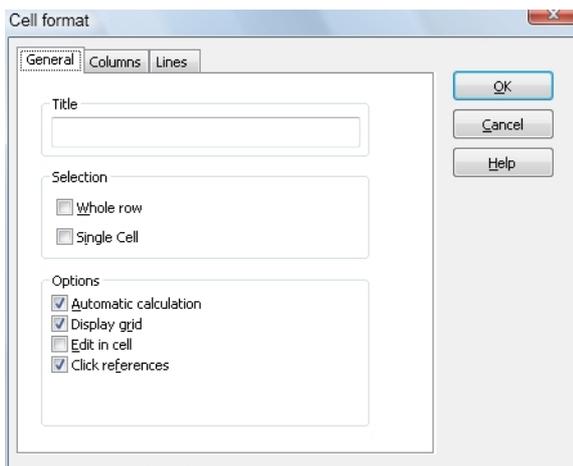
Parameters	Description
Heading	Table heading
Selection	Defines selection behavior when clicked.
whole row	`Active:` The whole row is selected.
one cell	`Active:` Only the cell that has been clicked on is selected; multiple selection of cells is not possible.
Options	
Automatic calculation	`Active:` Calculations are automatically updated when input values are changed.
Display grid	`Active:` Shows grid network lines in the editor.
Editing in the cell	`Active:` Entries are made in the cell directly; the input cell in the report window is shaded out.
Click reference	`Active:` The cell that has been clicked is accepted as the report coordinate when formulas are entered. For example: Instead of entering A12, the A12 cell is clicked.

Columns
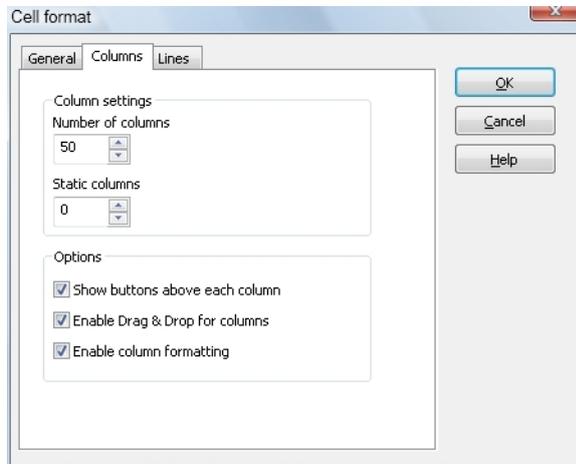
Parameters	Description
Column settings	
Number of columns	Defines the maximum number of columns. Valid value: 0 to 31000.
Static columns	Number of columns that are always shown. Scrolling only takes place after the static columns.
Show buttons above each column	Active: Shows target numbers.
Activation of the column formatting	Active: The column width can be adjusted using the mouse by dragging the borders of the header.

Rows
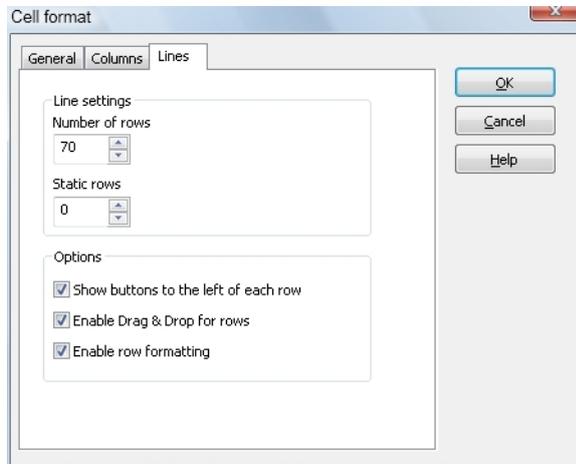
Configurable features of the rows are:

Parameters	Description
Number of rows	Defines the maximum number of rows. Valid value: 0 to 31000.
Static rows	Number of rows that are always shown. Scrolling only takes place after the static columns.
Display of the buttons in the first row	Active: Shows column header >(A, B, C, ...).
Activation of the row formatting	Active: The row width can be adjusted using the mouse by dragging the borders of the row numbering.

Format cell

For formatting the cells of the report sheet, there are settings for:

For formatting, select:

▶   The `Format cell` command in the context menu of the report sheet to open the dialog

▶   or the desired sub-menu in *Report -> Format cells,* to open the dialog with the corresponding tab

Alignment

The display of cell contents is defined in the `Orientation` tab.
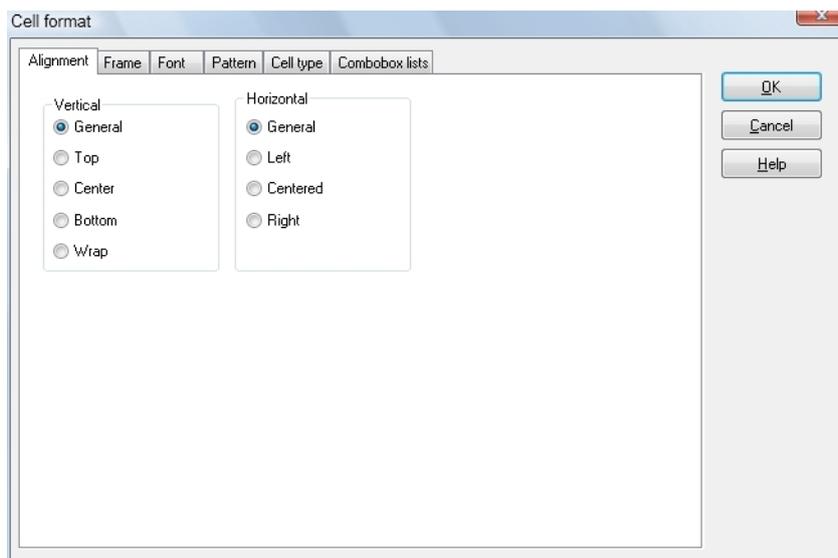
The following are available for vertical positioning:

▶   General

▶   Top

▶   Center

▶   Bottom

▶ Wrap

The following are available for horizontal positioning:

▶ General

▶ Left

▶ Center

▶ Right

> 💡 **Info**
>
> *If text is to be displayed vertically, select the* `vertical` *property in the* **Font** *tab.*

Frame

In the Frame tab, you define if a frame is displayed around a cell and if so, in what form. You can define each side of the frame (top, bottom, left, right) in terms of type and color:

1. Define the type and color.

2. Allocate these by clicking the whole frame or an individual page.
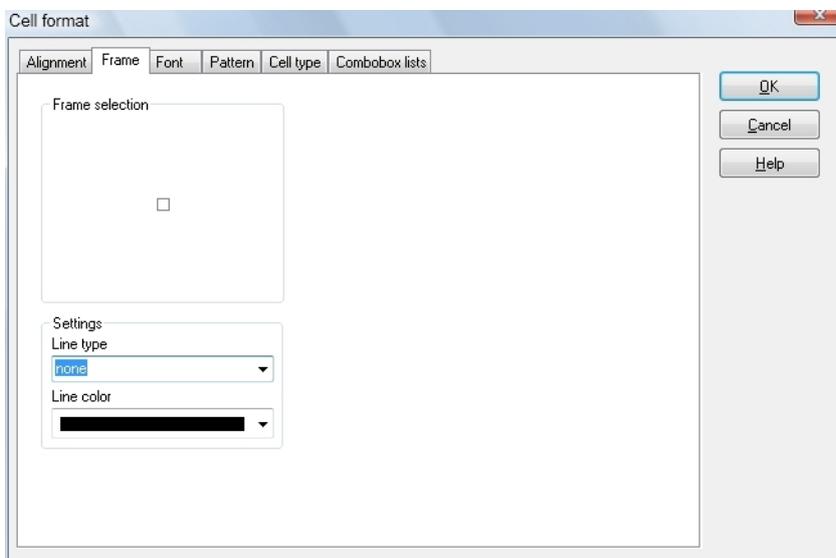
Parameters	Description
Frame selection	Clicking in the center activates all our frame sides. Clicking on the frame side activates these.
Type of line	Selection of line type from drop-down list.
Line color	Selection of line color from drop-down list.

Font

The standard Windows settings are available for fonts.
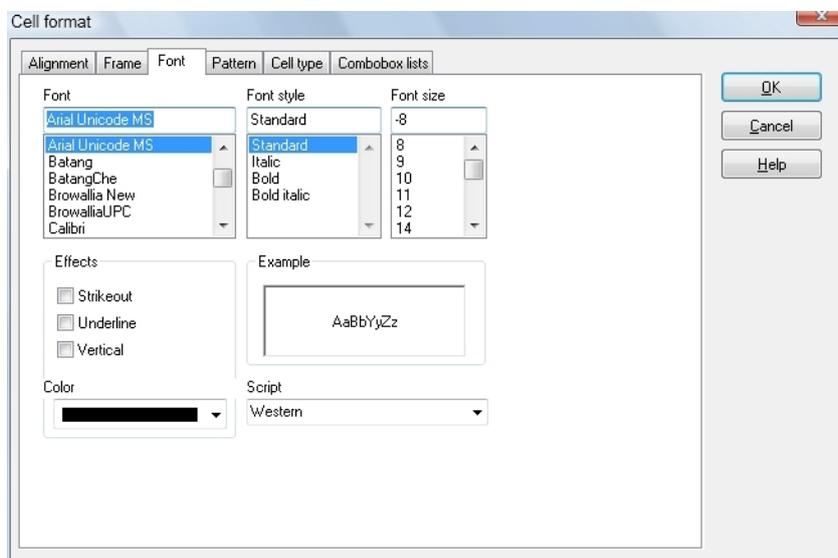
Parameters	Description
`Font`	Selection of the font.
`Font style`	Selection of the aspect.
`Size`	Selection of aspect in point.
`Effects`	
`Strike-through`	`Active`: Text is displayed as strike-through in the cell.
`Underlined`	`Active`: Text in the cell is underlined.
`Color`	Is not taken into account! The color must, for example, be set in a dynamic element when using the font
`Script`	Is not taken into account! Unicode display is used throughout.
`Example`	Shows effects of the selection on the text in cells.

Pattern

You define the color and pattern of the cell in the pattern tab.
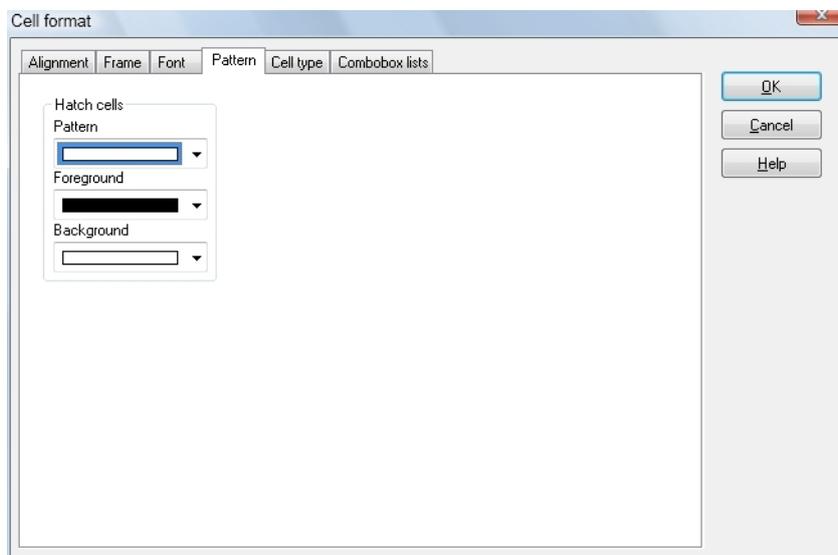
Parameters	Description
Pattern	Selection of cell pattern from the drop-down list.
Foreground	Selection of foreground color from drop-down list.
Background	Selection of background color from drop-down list.

Cell type

Define the output format in the `cell type` tab. You have the choice between:

▶ General: Output of values, data and binary data

▶ Combobox: Create lists

▶ Data type: Definition of data types

▶ Graphics file: Inserting graphics

The respective format key is described in the Formatting key (on page 160) chapter.

GENERAL
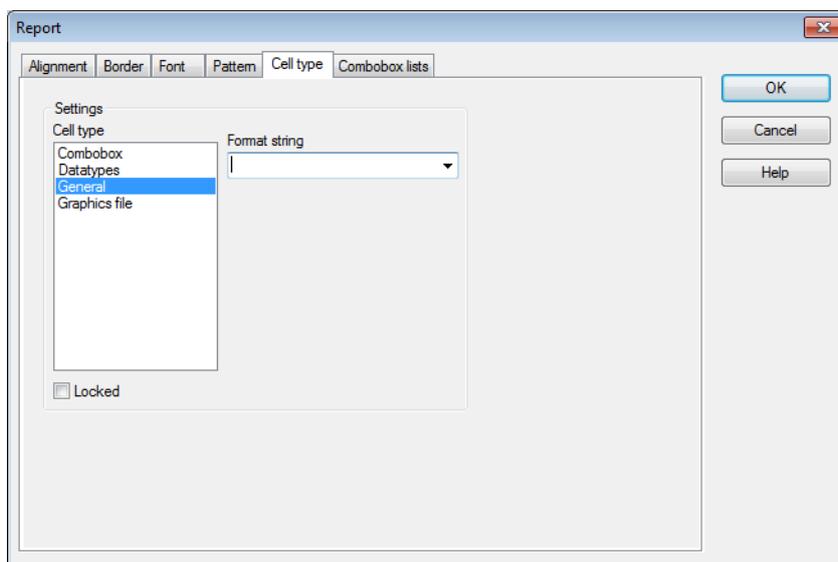
Parameters	Description
Format strings	Selection of the output format
▸ Value	Numerical value. For details, see the Numerical data types (on page 160) chapter.
▸ Date	Date information. For details, see the Data types for date and time (on page 162) chapter.
▸ Time	Time information. For details, see the Data types for date and time (on page 162) chapter.
▸ Binary value	binary value. For details, see the Logical data types (on page 161) chapter.
Locked	Active: Locks the cell for editing in the editor and locks entries in the online report.

Example

Format string for value output:

0.00 = any desired value with two decimal points

dd/mm/yy hh:mm = 01/01/10 12:30 (corresponding to the settings in the operating system

COMBOBOX

The combobox cell type offers drop-down lists to select pre-defined inputs. Before this cell type can be used, a list must be created in the combobox lists (on page 40) tab.
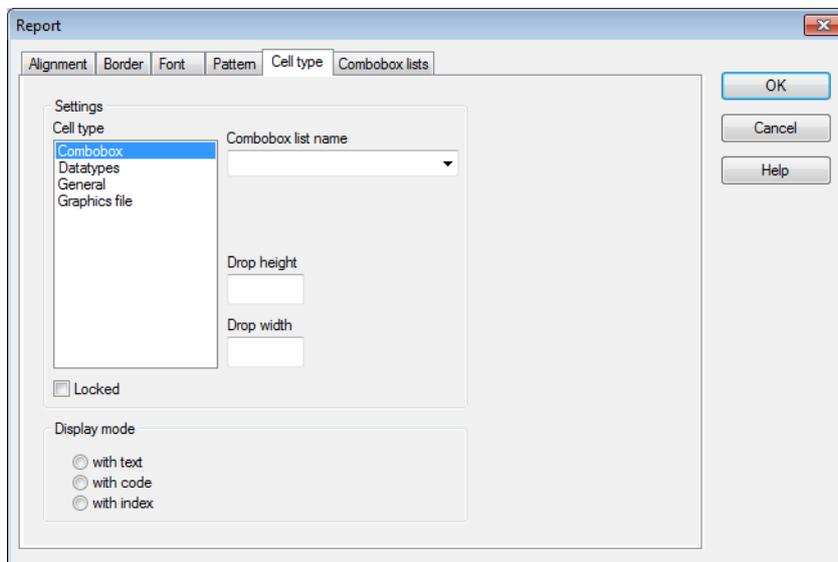
Parameters	Description
Combobox lists name	Name of the list file. This must already be defined in the Combobox lists (on page 40) tab.
Drop height	Display height
Drop width	Display width
Locked	Active: Locks the cell for editing in the editor and locks entries in the online report.
Display mode	Type of display
▸ with text	Alphabetically sorted according to text
▸ with code	Sorted according to code
▸ with index	Sorted according to the order of input of the list entries

DATA TYPE

An input template is created in the report with the data type. Existing functions are removed from this cell. The user receives instructions on how data is to be entered into the cell.
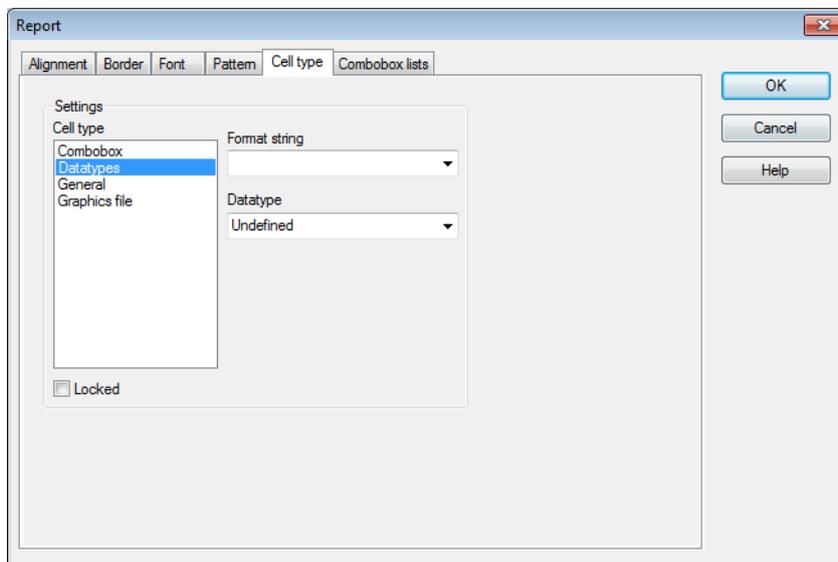
Parameters	Description
Format string	Input of format, depending on data type.
Data type	
▸ Date/time	Input of a date/time value. For format string details, see the Data types for date and time (on page 162) chapter.
▸ Float	Input of a float value. For format string details, see the Numerical data types (on page 160) chapter.
▸ Integer	Input of an integer value. For format string details, see the Numerical data types (on page 160) chapter.
▸ Logical	Input of a binary value. For format string details, see the Logical data types (on page 161) chapter. Configurable: Yes/no; right/wrong.
▸ Text	Input of text.
▸ Undefined	No data type defined.

Locked	**Active:** Locks the cell for editing in the editor and locks entries in the online report.

GRAPHICS FILE

Graphics files must be created before they are selected in the project manager: *File nodes -> Graphics -> [context menu] Add files.*
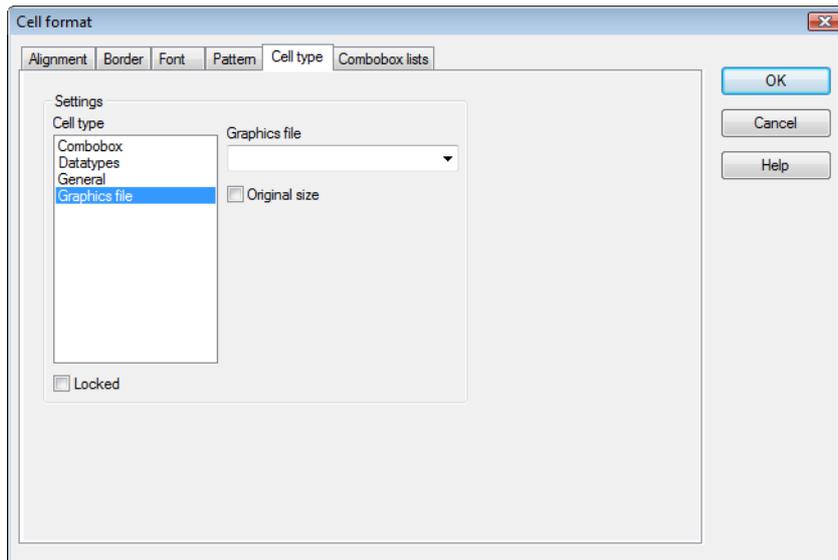
Parameters	Description
Graphics file	Selection of graphics file. BMP, JPG, GIF, PNG and WMF formats are supported.
Original size	**Active:** The graphics are displayed in original size. **Inactive:** The graphics are adapted to the size of the cell. Default: inactive.
Locked	**Active:** Locks the cell for editing in the editor and locks entries in the online report.

Combo Box lists

In the **Combobox lists** tab, lists for the cell type **Combobox** are defined.

Parameters	Description
`Combobox lists`	Drop-down list to select existing lists.
New	Opens the dialog to create a new list.
Delete	Deletes selected list.
Content of the selected lists	Shows contents of the selected lists
Current entry	Configuration of the entry.
`Text`	Input of the text, as it should appear in the list in the Runtime.
`Code`	Input of a number for sorting the lists. `0`: no sorting according to code. There is a choice of sorting lists in Runtime alphabetically or in the order of input. Minimum value: `0` Maximum value: `9999999999`
Add	The new entry is inserted depending on the settings in **Content of selected lists** that were selected when the list was created. ▶ Not: Entry is inserted at the end of the list. ▶ According to text: Alphabetic sequence. ▶ According to codes: Code number determines the sequence (increasing) in lists.
Paste	The new entry is added above the entry selected in the **Content of selected lists**:
Delete	Deletes the entry selected in **Content of selected lists**.

CREATE NEW LIST
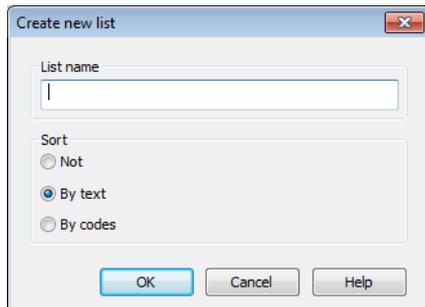
Parameters	Description
`Name lists`	Name of list
`Sort`	Sorting criteria for display in **Content of selected lists** and in Runtime
`not`	Entry is inserted at the end of the list.
`According to text` (default)	Alphabetic sequence
`According to codes`	Code number determines the sequence (increasing) in lists.

Format columns and cells

If an entire row or an entire column is highlighted, the additional commands **Format column** and/or **Format row** are available in the context menu.

As with **Format cell**, tabs are available for:

Alignment (on page 31)

Frame (on page 33)

Font (on page 33)

Pattern (on page 34)

Note: These settings always only concern the column heading and/or the row heading. They have no effect on the other cells of a column or row. These must always be formatted using the **Format cells** (on page 30) dialog.

There is also a further Column (on page 43) or Row (on page 44) tab available in this dialog.

Column

In the Column tab, properties that concern the whole column are defined:

Parameters	Description
Definition	
`Heading`	Defines column heading. $C: Letters in a rising sequence of column numbers E: The column automatically expands to the correct number of columns if an **archive***, **aml** or **cel** is present in one of the cells. If other report functions are used, "E" may not be used. "E" can be used in a report several times. `Default:` $C
`Width in pixels`	Width of the column in pixels
Options	
`Hide`	`Active:` The column is hidden. `Inactive:` The column is shown. To show a column that has been hidden: Highlight the two neighboring columns and set the `Hide` property to `inactive`. **Attention:** Functions that are in the hidden cells are not executed. To execute a function without it being shown: Set the row height or column width to `0`.
`Protection`	`Active:` The cells in the column are protected from changes of formatting. `Inactive:` The cells of this column can be formatted.

Row

This dialog allows defining the properties of the entire selected row.

Parameters	Description
Definition	
Heading	Defines row heading. $R: Numbers in reverse order of column number E: The row automatically expands to the correct number of rows if an **archive***, **aml** or **cel** is present in one of the cells. If other report functions are used, "E" may not be used. "E" can be used in a report several times. Default: $R
Width in pixels	Height of the row in pixels.
Options	
Hide	Active: The row is hidden. Inactive: The row is displayed. To show a row that has been hidden: Highlight the two neighboring rows and set the Hide property to inactive. **Attention:** Functions that are in the hidden cells are not executed. To execute a function without it being shown: Set the row height or column width to 0.
Protection	Active: The cells in the row are protected from changes of formatting. Inactive: The cells of this row can be formatted.

3.1.7   Set up archive filter

Should archive values be used in the report then these are to be defined before activating the report functions. To set up the archive filter:

▶   Select **Filter** in the **Format** menu.
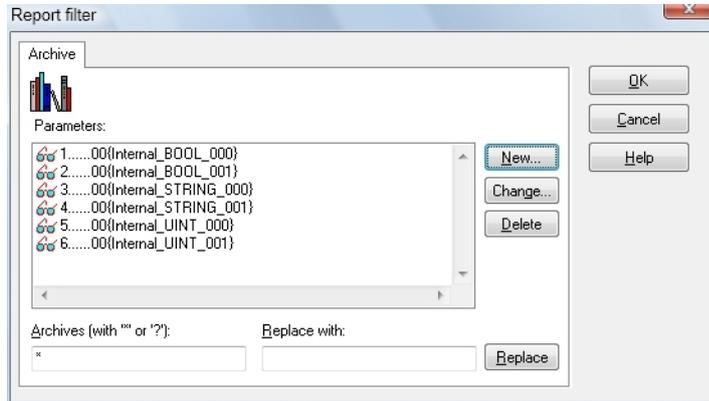
Parameters	Description
`Parameters`	List of variables selected from the archives. The numbers in front of the variables are used in the archive and protocol functions (on page 52) for classification.
`New`	Opens the dialog for selecting variables.
`Change`	Opens the dialog for selecting variables.
`Delete`	Deletes selected variable from the `Parameters` list.
`Archives`	Definition of a variable filter. Wildcards * and ? are permitted at the start of an expression.
`replace with`	Enter the expression that is to replace the string that is being searched for
`Replace`	Carries out a search and replace. The result is displayed in a message box.

"NEW FILTER" OR "CHANGE FILTER"

Clicking on the `New` or `Change` buttons opens the dialog to select the variables:
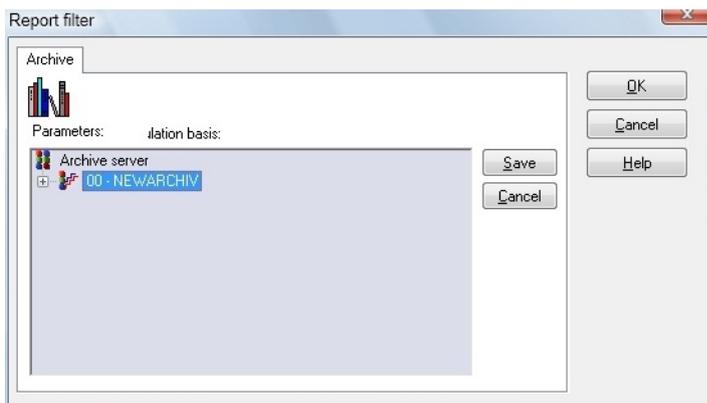
Parameters	Description
Parameters	Display of the archives present. It is possible to select a complete archive or individual variables. Each variable automatically receives an increasing number, which is addressed in the report functions for archive (on page 52). A report can therefore also be used throughout projects.
Save	Provides selected archives/variables for the report.
Cancel	Leaves dialog without selection.

Each variable automatically receives an increasing number, which is addressed in the report functions for archive (on page 52).

3.1.8   Report functions

For the formation of the report, different formatting and calculation functions are available.

▶   `Processing functions:`

Defined access to selected zenon data (online and archive values, user, computer names etc.)

▶   `Report functions`

- formatting functions
- calculating functions

The following groups of functions are available:

▶   Database functions (on page 79)

▶   Date and time functions (on page 87)

▶   Logical functions (on page 93)

▶   Archive and protocol functions (on page 52)

▶   Statistical functions (on page 119)

▶   Mathematical and trigonometric functions (on page 99)

The functions that are used most often can be configured using the function assistant (on page 51).

INPUT

The input of the functions is done in the input row of the report.

If several cells are to receive the same input/function:

1.   Highlight the starting cell and the target cell

2.   Select the `Fill` command in the context menu and the desired direction (`right` or `down`)

References to cell co-ordinates in the function are automatically adjusted.

EXAMPLE FOR INPUTS

Parameters	Description
`=archiv(1,value,down)`	All values of the archive variable 1 are displayed downwards from the cell.
`=sum(A4:D4)`	The contents of cells A4 to D4 are added together. The cell must be formatted as a value.
`=now()`	Provides the current date/the current time.

Info

Report functions can lead to error messages in the editor if not all data required to calculate the required data is properly available. To check to see that the configuration is correct, test the report in Runtime.

Direction

Options	Meaning
`Right`	Value is entered to the right (row).
	There must be sufficient cells left free for expected entries from further report creation.
	Attention: If an `archivesp` function is created with only one column, the value is entered downwards (in the column).
`below`	Value is entered downwards (column).
	There must be sufficient cells left free for expected entries from further report creation.
	Attention: If an `archivesp` function is created with only one row, the value is entered to the right (in the row).

Info

The Report Generator module has to be licensed in the development environment and in the Runtime (standalone, server/standby and clients).

If the module is not licensed in the Runtime, the following functions are not available:

▸ Save archives (for manual inputs)

▸ and writing set values

Read access is possible.

Conventions

The following conventions are valid:

Parameters	Description
Number:	(num); divider for decimal points depends on the settings in the operating system (1,0 or 1.0).
Area:	(A1:A10)
String:	Character sequence is separated by a dual apostrophe (**"**) , for example "ABC".
Radians:	The input of goniometric functions is done in radians.

CONVERSION OF DEGREES INTO RADIANS

degrees	arc measurement	radians
0,00	0	0
45,00	p/4	0,78539816339744828
90,00	p/2	1,5707963267948966
135,00	3p/4	2,3561944901923448
180,00	p	3,1415926535897931
225,00	5p/4	3,9269908169872414
270,00	3p/2	4,7123889803846897
315,00	7p/4	5,497787143782138
360,00	2p	6,2831853071795862

Function assistant

The function assistant supports the planning of report functions. It makes the most frequently used functions available. All other functions are manually entered into the input cell.

To open the function assistant:

1. Click in the desired cell

2. in the context menu, select the `Functions` command

3. The assistant is opened

4. The selection on the assistant is transferred to the input cell;

   any quotation marks that are open may need to be closed manually
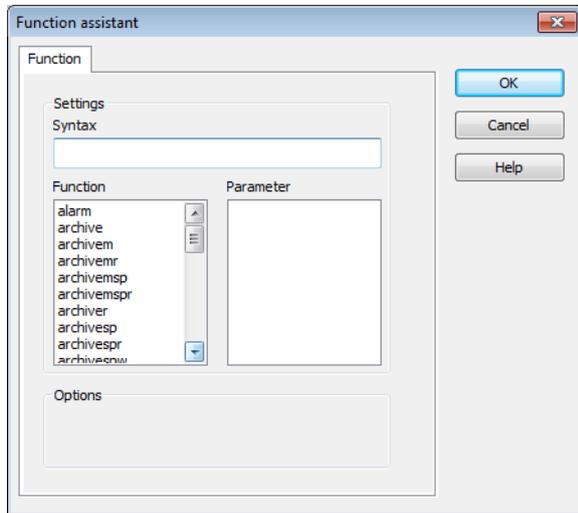
Parameters	Description
Syntax	Result of the selection of function and parameter.
Function	Selection of the function.
Parameters	Selection of parameter that is available for the selected function.

Archive and logging functions

BASICS

All archive functions build on the basic `archive(FILTERINDEX,PARAMETER,RICHTUNG)` function.

The expansions are distinguishable by additional letters in function names.

Code	Meaning	Comment
r	read only	There is only read access to the archive. Changed values cannot be saved.
sp	Columns	Provides special formatting to define cells to be used.
m	Mathematics	Only has an effect on `value` and `time`.
ex	extended	provides special treatments.
w	write	Writing is possible.

FILTER INDEX

The filter index creates a distinction between archive filter numbers (on page 45) used in the report functions and variables from the archives.

> **Example**
>
> `=archive(5,"value","downwards")`
>
> *FILTERINDEX = 5*
>
> *The entry present in the filter under no. 5 is used for the report function.*

PARAMETERS

Options	Possible value
value	[1-n]
status	[1-n]
time	[1-n]
number	[1]

Direction

Options	Meaning
`Right`	Value is entered to the right (row). There must be sufficient cells left free for expected entries from further report creation. **Attention:** If an `archivesp` function is created with only one column, the value is entered downwards (in the column).
`below`	Value is entered downwards (column). There must be sufficient cells left free for expected entries from further report creation. **Attention:** If an `archivesp` function is created with only one row, the value is entered to the right (in the row).

All archive functions can be created with the function assistant (on page 57).

On outputting the status information, the settings from the `project.ini` file in the `[STATUS]` section or the settings from the internally defined allocation are used. If the status labeling was changed in `project.ini`, the changed name is given. (Tip: makes it possible to keep the project compatible with old status labels.)

In online operation, the query time is either defined in the function query (on page 164) or given in Runtime when a report file is opened.

CHANGE ARCHIVE ENTRIES

If permitted by user authorizations, archive values can also be changed in Runtime. Changes are either saved manually via the `Save` control element or automatically saved when saving a report file. The status of an amended entry is automatically set to `manual value`.

> **Info**
>
> *Changing string fields in the report: If a text consists only of numbers, spaces are removed by default when it is saved. Spaces that are desired must be displayed by ASCII character 255 (`Alt+255`).*

Handling of archive entries

Archive entries can be displayed in a report and also be changed and saved in the archive again with sufficient user authorizations. There is a function assistant (on page 57) available to create the report functions.

READING AND WRITING ARCHIVE ENTRIES

The following contents of an archive entry can be read for display in a report:

- ▶ Value

- ▶ Status

- ▶ `Time`

The following archive entries can be changed and saved via calculations and new entries in the report:

- ▶ Value

- ▶ `Status`

Amended archive entries are saved in Runtime using the `Save archive` button. Saving changes can be stopped by:

- ▶ Activating the `Read only` in the zenon `Screen switching` function for the report screen

- ▶ Activating the `Locked` option in the format (on page 35) dialog cell

If the value of an archive entry is changed, the status is set as `MAN_VAL` when it is saved in the archive.

Exception: If the status of the archive entry is changed, only the changed status is entered. If safety regulations such as those of the FDA are to be met, no status change may be permitted as a result of the report when configuring. Archive entries with `MAN_VAL` status are displayed in `blue` in a report.

REPRESENTATION

Archive entries can be represented as archive value rows or as row-formatted archive value rows. The archive entries which lie in the time range of the opened log are represented rightwards or downwards from the cell in which the archive function is configured. If no archive entry is found then a − is output in the cell in which the archive function is defined.

ARCHIVE VALUE ROWS

The read-in archive entries are shown as a row, one after the other, without blank spaces.

ROW-FORMATTED ARCHIVE VALUE ROWS

The read-in archive entries are shown with a defined number of:

- ▶ Archive entries per row

- ▶ Blank spaces between two entries

- ▶ Empty rows between the rows.

.

CONTROL CELLS

The parameters `P1`, `P2` and `P3` set out which cells of the table can be filled:

- ▶ `P1`: Number

- ▶ `P2`: Empty cells between two value outputs, depending on `DIRECTION` (`down`, `right`) either empty rows or empty columns

- ▶ `P3`: Empty cells between two value outputs, depending on `DIRECTION` (`down`, `right`) either empty rows or empty columns

The value, status and time of an archive entry can be read in. All entries of the time range of the log are shown in the log going downwards or rightwards from the cell in which the function was given, with the row format defined by the parameters P1-P3.

WHEN SELECTING `DOWN`, THE FOLLOWING APPLIES:

- ▶ `P1` = number of rows to be filled with values

- ▶ `P2` = empty rows between the values

- ▶ `P3` = empty columns

WHEN SELECTING `RIGHT`, THE FOLLOWING APPLIES:

- ▶ `P1` = number of columns to be filled

▶   `P2` = empty rows between the values

▶   `P3` = empty rows between the values

Configuration with function assistant

You can configure report functions for the archive with the function assistant:

1. Select the desired cell in the context menu with the `Functions` command.

2. The assistant is opened.

3. Select the desired archive function.
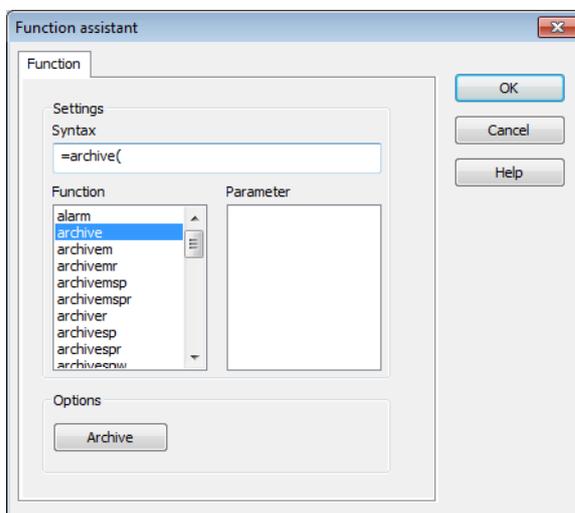
4. The archive function is entered into the syntax.

5. Clicking on archive opens the archive filter (on page 45).
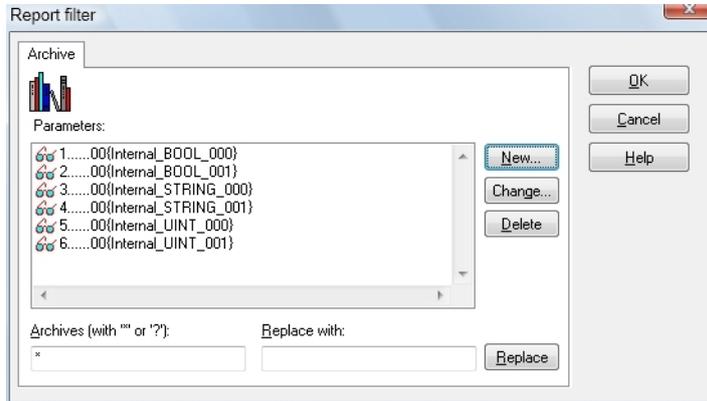
6. Select the desired entry.
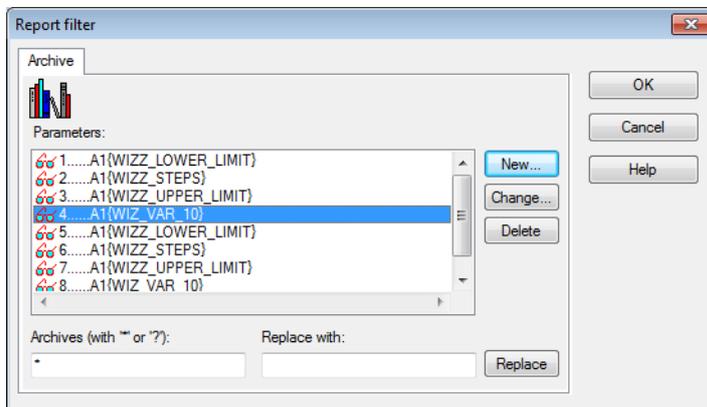
7. The reference number of the entry (for example 4) is transferred to the string of the syntax.

8. Select the desired parameter, for example value).
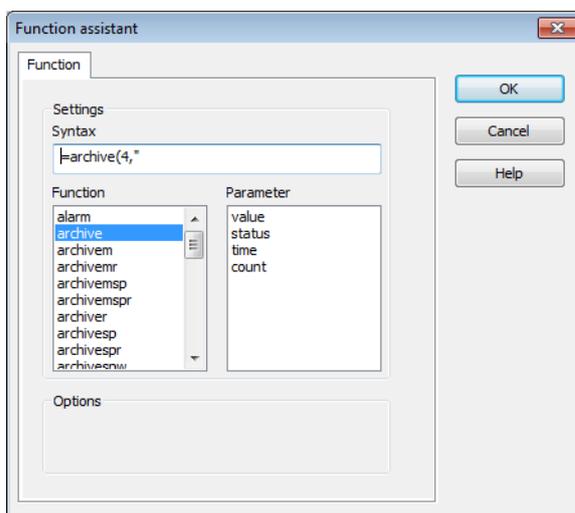
9. The parameter is transferred into the string.

10. Select the desired direction of the entry in the table:

    - `bottom`: The function fills the column

    - `right`: The function fills the row

      Attention: When using the `archivesp` function, corresponding columns/rows are envisaged! If for example, to the right is selected, but there is only one column available, then the values are entered downwards (in the only column available).
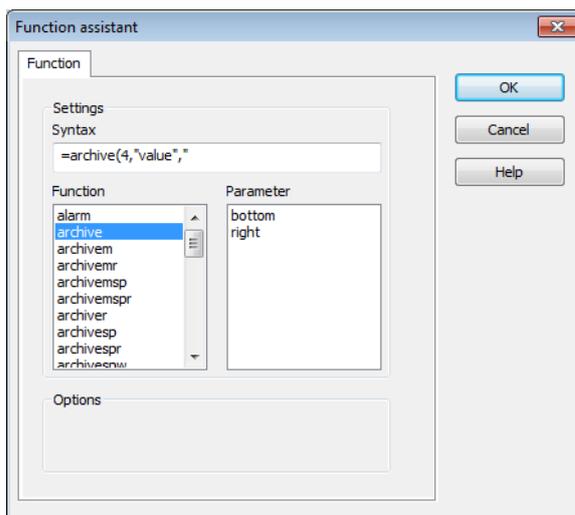
11. The direction is transferred into the string.

12. The brackets are closed.
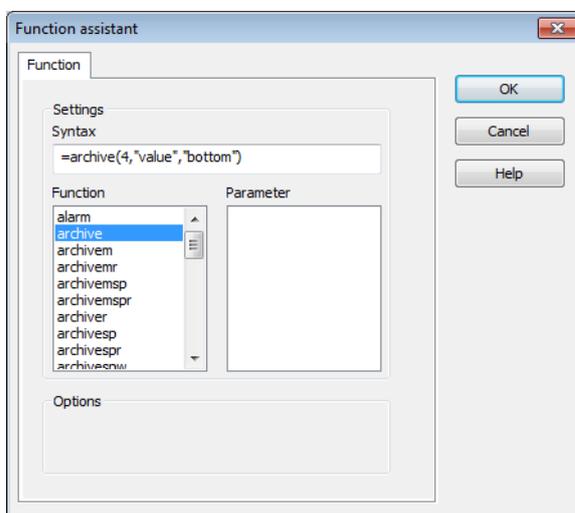
13. The function is transferred to the cell by clicking on OK.

archive

Value, Status or time of the archive entry can be read. Changes are either saved manually via the `Save` control element or automatically saved when saving a report file. The status of an amended entry is automatically set to `manual value`.

`Syntax: =archive(`FILTERINDEX,PARAMETER,DIRECTION`)`

Transfer parameters	Valid range	Comment
`FILTERINDEX`		Corresponding entry from the archive filter (on page 45).
`PARAMETER`	value, status, time, count	The selected parameter is read from the archive and displayed.
`DIRECTION`	bottom, right	Defines the direction in which filling is to take place, starting from the cell in which the function was created.

Example

In a report, all archive values from the archive variable 1 should be given downwards.

=archive(1,"value","bottom")

archiveex

`Syntax:`
`=archiveex(`FILTERINDEX,PARAMETER,DIRECTION,STATUS,ALTERNATIVETEXT,NN_TEXT,NN_ALTERNATIVEVALUE,CYCLEOFFSET`)`

Transfer parameters	Valid range	Comment
FILTERINDEX	[1-n]	Corresponding entry from the archive filter (on page 45).
PARAMETER	value, status, time, count	The selected parameter is read from the archive and displayed
DIRECTION	bottom, right	Filling in the corresponding direction.
STATUS	SPONT, MAN_VAL, INVALID	Status of the values to be displayed.
ALTERNATETEXT		Text to be displayed if the status of the archive variables does not correspond to the STATUS.
ND_TEXT		Text for the "non-detectability limit"; not case-sensitive.
ND_ALTERNATEVALUE		Cell reference or entered value are possible. If the ALTERNATETEXT is entered into the cell and then the table is saved, the NN_ALTERNATEVALUE is written to the archive.
CYCLEOFFSET		Is only taken into account for filter type: ▸ Time, day ▸ Time, month ▸ Time, year ▸ Time, week ▸ Time, 2week ▸ Time, 15min. ▸ Time, 30min. ▸ Time, 60min.

Example

=archiveex(1,"time","downwards",SPONT,MAN_VAL,"-","NN",G2,-2)

archiveexr

Syntax:
=archiveexr(FILTERINDEX,PARAMETER,DIRECTION,STATUS,ALTERNATIVETEXT,NN_TEXT,NN_ALTERNATIVEVALUE,CYCLEOFFSET)

Transfer parameters	Valid range	Comment
FILTERINDEX	[1-n]	Corresponding entry from the archive filter (on page 45).
PARAMETER	value, status, time, count	The selected parameter is read from the archive and displayed
DIRECTION	bottom, right	Filling in the corresponding direction.
STATUS	SPONT,MAN_VAL, INVALID	Status of the values to be displayed.
ALTERNATETEXT		Text to be displayed, if the status of the archive variables does not correspond to the STATUS.
ND_TEXT		Text for the "non-detectability limit"; not case-sensitive.
ND_ALTERNATEVALUE		Cell reference or entered value are possible. If the ALTERNATETEXT is entered into the cell and then the table is saved, the NN_ALTERNATEVALUE is written to the archive.
CYCLEOFFSET		Is only taken into account for filter type: ‣ Time, day ‣ Time, month ‣ Time, year ‣ Time, week ‣ Time, 2week ‣ Time, 15min. ‣ Time, 30min. ‣ Time, 60min.

archivem

This function makes it possible to automatically change values or times of an archive entry by adding a constant to a value. Changes are either saved manually via the `Save` control element or automatically saved when saving a report file. The status of an amended entry is automatically set to `manual value`.

`Syntax: =archivem`(FILTERINDEX,PARAMETER,DIRECTION,MATH)

Transfer parameters	Valid range	Comment
`FILTERINDEX`	[1-n]	Corresponding entry from the archive filter (on page 45).
`PARAMETER`	`value, status, time, count`	The selected parameter is read from the archive and displayed
`DIRECTION`	`bottom, right`	Filling in the corresponding direction.
`MATH`		▸ Constant: With sign ▸ Decimal points: Point separator (.) ▸ Correction of a time: In minutes

The value is read-in and shown in the report with the appropriate manipulation. The corrected values can be saved using **Save archive**.

> ⚠ **Attention**
>
> *Every time the archive is saved, the values are corrected by the defined constant!*

archivemr

This function makes it possible to automatically change values or times of an archive entry by adding a constant to a value.

Syntax: =archivemr(FILTERINDEX,PARAMETER,DIRECTION)

Transfer parameters	Valid range	Comment
FILTERINDEX	[1-n]	Corresponding entry from the archive filter (on page 45).
PARAMETER	value, status, time, count	The selected parameter is read from the archive and displayed
DIRECTION	bottom, right	Filling in the corresponding direction.
MATH		▸ Constant: With sign ▸ Decimal points: Point separator (.) ▸ Correction of a time: In minutes

⚠ **Attention**

Changed values cannot be saved in the archive.

The value is read-in and shown in the report with the appropriate manipulation. Only `time` and `value` are possible as parameters.=archivemr(4,value,right,6.5)

`All archive values read` are increased by `6.50` for the display in the report.

`=archivemr(4,time,right,-1)`

The time stamps of the archive entries are corrected back by `1` minute for display in the report.

Possibility for use: For example, for a calculated monthly value that is only available after the end of the month (time stamp, for example. `00:01:00`).

archivemsp

This function makes it possible to automatically change the value or times of an archive entry by adding a constant to the value as well as establishing the cells for the entries. The value is read-in and shown in the report with the appropriate manipulation. Changes are either saved manually via the `Save` control element or automatically saved when saving a report file. The status of an amended entry is automatically set to `manual value`.

```
Syntax: =archivemsp(FILTERINDEX,PARAMETER,DIRECTION)
```

Transfer parameters	Valid range	Comment
FILTERINDEX	[1-n]	Corresponding entry from the archive filter (on page 45).
PARAMETER	value, status, time, count	The selected parameter is read from the archive and displayed
DIRECTION	bottom, right	Filling in the corresponding direction.
P1		Number of archive values in the first row.
P2		Empty cells between two value outputs.
P3		Empty cells until the next value row.
MATH		▸ Constants with sign, constants: With sign ▸ Decimal points: Point separator (.) ▸ Correction of a time: In minutes

> ⚠ **Attention**
>
> *The values are corrected by the defined constant with every* **Save archive**.

CONTROL CELLS

The parameters P1, P2 and P3 set out which cells of the table can be filled:

▸ P1: Number

▸ P2: Empty cells between two value outputs, depending on DIRECTION (down, right) either empty rows or empty columns

▸ P3: Empty cells between two value outputs, depending on DIRECTION (down, right) either empty rows or empty columns

The value, status and time of an archive entry can be read in. All entries of the time range of the log are shown in the log going downwards or rightwards from the cell in which the function was given, with the row format defined by the parameters P1-P3.

WHEN SELECTING `DOWN`, THE FOLLOWING APPLIES:

▶ `P1` = number of rows to be filled with values

▶ `P2` = empty rows between the values

▶ `P3` = empty columns

WHEN SELECTING `RIGHT`, THE FOLLOWING APPLIES:

▶ `P1` = number of columns to be filled

▶ `P2` = empty rows between the values

▶ `P3` = empty rows between the values

▶

> ### 🔖 Example
>
> `=archivemsp(4,"time","right",16,1,0,-0.5)`
>
> *All time entries of the archive entry with the filter number 4 are retarded 30 seconds; 16 entries are envisaged with an empty column and no space.*

archivemspr

This function makes it possible to automatically change values or times of an archive entry for display in a report by adding a constant to a value. The value is read-in and shown in the report with the appropriate manipulation. Only `time` and `value` are possible as parameters.

> ### ⚠ Attention
>
> *Changed values cannot be saved in the archive.*

`Syntax: =archivemspr(FILTERINDEX,PARAMETER,DIRECTION)`

Transfer parameters	Valid range	Comment
FILTERINDEX	[1-n]	Corresponding entry from the archive filter (on page 45).
PARAMETER	value, status, time, count	The selected parameter is read from the archive and displayed
DIRECTION	bottom, right	Filling in the corresponding direction.
P1		Number of archive values in the first row.
P2		Empty cells between two value outputs.
P3		Empty cells until the next value row.
MATH		▸ Constants with sign, constants: With sign ▸ Decimal points: Point separator (.) ▸ Correction of a time: In minutes

CONTROL CELLS

The parameters P1, P2 and P3 set out which cells of the table can be filled:

- ▸ P1: Number

- ▸ P2: Empty cells between two value outputs, depending on DIRECTION (down, right) either empty rows or empty columns

- ▸ P3: Empty cells between two value outputs, depending on DIRECTION (down, right) either empty rows or empty columns

The value, status and time of an archive entry can be read in. All entries of the time range of the log are shown in the log going downwards or rightwards from the cell in which the function was given, with the row format defined by the parameters P1-P3.

WHEN SELECTING DOWN, THE FOLLOWING APPLIES:

- ▸ P1 = number of rows to be filled with values

- ▸ P2 = empty rows between the values

- ▸ P3 = empty columns

WHEN SELECTING `RIGHT`, THE FOLLOWING APPLIES:

▶ `P1` = number of columns to be filled

▶ `P2` = empty rows between the values

▶ `P3` = empty rows between the values

> **Example**
>
> `=archivemsp(4,"time","right",16,1,0,-0.5)`
>
> *All time entries of the archive entry with the filter number 4 are retarded 30 seconds; 16 entries are envisaged with 1 empty column and 0 spaces.*

archiver

This function makes it possible to read the value, status and time of an archive entry. No changes are possible.

`Syntax: =archiver(FILTERINDEX,PARAMETER,DIRECTION)`

Transfer parameters	Valid range	Comment
`FILTERINDEX`	[1-n]	Corresponding entry from the archive filter (on page 45).
`PARAMETER`	`value, status, time, count`	The selected parameter is read from the archive and displayed
`DIRECTION`	`bottom, right`	Filling in the corresponding direction.

This differs from the `archive` (on page 60) function in that no manual entries can be made in Runtime.

Area of application: These functions are employed, for example, if an entry in a report must be shown several times and may only be edited at one place. All entries of the time range of the log are shown in the report going downwards or rightwards from cell in which the function was given.

archivesp

Value, Status or time of the archive entry can be read. Changes are either saved manually via the `Save` control element or automatically saved when saving a report file. The status of an amended entry is automatically set to `manual value`.

Syntax: `=archivesp(FILTERINDEX,PARAMETER,DIRECTION,P1,P2,P3)`

Transfer parameters	Valid range	Comment
`FILTERINDEX`	[1-n]	Corresponding entry from the archive filter (on page 45).
`PARAMETER`	`value, status, time, count`	The selected parameter is read from the archive and displayed
`DIRECTION`	`bottom, right`	Filling in the corresponding direction
`P1`		Number of archive values in the first row
`P2`		Empty cells between two value outputs
`P3`		Empty cells until the next value row

CONTROL CELLS

The parameters `P1`, `P2` and `P3` set out which cells of the table can be filled:

- ▶ `P1`: Number

- ▶ `P2`: Empty cells between two value outputs, depending on `DIRECTION` (`down`, `right`) either empty rows or empty columns

- ▶ `P3`: Empty cells between two value outputs, depending on `DIRECTION` (`down`, `right`) either empty rows or empty columns

The value, status and time of an archive entry can be read in. All entries of the time range of the log are shown in the log going downwards or rightwards from the cell in which the function was given, with the row format defined by the parameters P1-P3.

WHEN SELECTING `DOWN`, THE FOLLOWING APPLIES:

- ▶ `P1` = number of rows to be filled with values

- ▶ `P2` = empty rows between the values

▶ `P3` = empty columns

WHEN SELECTING `RIGHT`, THE FOLLOWING APPLIES:

▶ `P1` = number of columns to be filled

▶ `P2` = empty rows between the values

▶ `P3` = empty rows between the values

When outputting the status information, the settings from the `project.ini` file in the `[STATUS]` section or the settings from the internally defined allocation are used. If the status labeling was changed in `project.ini`, the changed name is given. (Tip: makes it possible to keep the project compatible with old status labels.)

archivespr

Value, Status or time of the archive entry can be read.

Syntax: `=archivespr(FILTERINDEX,PARAMETER,DIRECTION,P1,P2,P3)`

Transfer parameters	Valid range	Comment
`FILTERINDEX`		Corresponding entry from the archive filter (on page 45).
`PARAMETER`	`value, status, time, count`	The selected parameter is read from the archive and displayed
`DIRECTION`	`bottom, right`	Filling in the corresponding direction
`P1`		Number of archive values in the first row
`P2`		Empty cells between two value outputs
`P3`		Empty cells until the next value row

CONTROL CELLS

The parameters `P1`, `P2` and `P3` set out which cells of the table can be filled:

▶ `P1`: Number

▶ `P2`: Empty cells between two value outputs, depending on `DIRECTION` (`down`, `right`) either empty rows or empty columns

▶   P3: Empty cells between two value outputs, depending on `DIRECTION` (`down`, `right`) either empty rows or empty columns

The value, status and time of an archive entry can be read in. All entries of the time range of the log are shown in the log going downwards or rightwards from the cell in which the function was given, with the row format defined by the parameters P1-P3.

WHEN SELECTING `DOWN`, THE FOLLOWING APPLIES:

▶   `P1` = number of rows to be filled with values

▶   `P2` = empty rows between the values

▶   `P3` = empty columns

WHEN SELECTING `RIGHT`, THE FOLLOWING APPLIES:

▶   `P1` = number of columns to be filled

▶   `P2` = empty rows between the values

▶   `P3` = empty rows between the values

Unlike the `archivesp` (on page 70) function, no manual presettings can be set on the cells. These functions are employed, for example, if an entry in a log must be shown several times and may only be edited at one place. All entries of the time range of the log are shown in the report going downwards or rightwards from cell in which the function was given

> **Example**
>
> *=archivespr(1,"value","bottom",12,1,3)*
>
> *All values of the archive entry with the filter number 1 are written* `downwards`*; 12 entries are envisaged with 1 empty row and 3 empty columns.*

archivespw

This function suppresses the displays of read-in archive entries and makes it possible to define the cells to be used. When saving, the archive entries are overwritten with values and statuses existing at other places (e.g. calculated).

```
Syntax:
=archivespw(FILTERINDEX,PARAMETER,DIRECTION,P1,P2,P3,SZ_VALUE,SZ_STATUS
)
```

Transfer parameters	Valid range	Comment
FILTERINDEX	[1-n]	Corresponding entry from the archive filter (on page 45).
PARAMETER	value, status, time, count	The selected parameter is read from the archive and displayed
DIRECTION	bottom, right	Filling in the corresponding direction.
P1		Number of archive values in the first row.
P2		Empty cells between two value outputs.
P3		Empty cells until the next value row.
SZ_VALUE	Start-cell values	Cell from which calculated values start to appear.
SZ_STATUS	Start cell status	Defining a fixed status text (for example MAN_VAL), or stating the starting cell from which the status text should be taken.

CONTROL CELLS

The parameters P1, P2 and P3 set out which cells of the table can be filled:

▶   P1: Number

▶   P2: Empty cells between two value outputs, depending on DIRECTION (down, right) either empty rows or empty columns

▶   P3: Empty cells between two value outputs, depending on DIRECTION (down, right) either empty rows or empty columns

The value, status and time of an archive entry can be read in. All entries of the time range of the log are shown in the log going downwards or rightwards from the cell in which the function was given, with the row format defined by the parameters P1-P3.

WHEN SELECTING DOWN, THE FOLLOWING APPLIES:

▶   P1 = number of rows to be filled with values

▶ `P2` = empty rows between the values

▶ `P3` = empty columns

WHEN SELECTING `RIGHT`, THE FOLLOWING APPLIES:

▶ `P1` = number of columns to be filled

▶ `P2` = empty rows between the values

▶ `P3` = empty rows between the values

STATUS TEXTS

The configuring is not done via the input help but directly in the cell. For this, the following applies:

▶ When entering the status texts, the defined short texts from the `project.ini` file or from the system-internal default configuration must be used. This applies regardless of whether the statues are input as fixed or read in from another cell.

▶ Several statuses are separated by a comma (`,`).

▶ The calculated values are updated regardless of whether the newly calculated value has changed in comparison to the read-in value.

▶ The status is changed to manual value (`MAN_VAL`).

archivew

This function suppresses the display of the archive entries in the report that have been read in. When saving, the archive entries are overwritten with values and statuses existing at other places (e.g. calculated). This report function can be used as an example for reduction of monthly values to annual values as an archive value.

```
Syntax:
=archivew(FILTERINDEX,PARAMETER,DIRECTION,P1,P2,P3,SZ_VALUE,SZ_STATUS)
```

Transfer parameters	Valid range	Comment
`FILTERINDEX`	[1-n]	Corresponding entry from the archive filter (on page 45).
`PARAMETER`	`value, status, time, count`	The selected parameter is read from the archive and displayed
`DIRECTION`	`bottom, right`	Filling in the corresponding direction.
`SZ_VALUE`	Start-cell values	Cell from which calculated values start to appear.
`SZ_STATUS`	Start cell status	Defining a fixed status text (for example MAN_VAL), or stating the starting cell from which the status text should be taken.

CONTROL CELLS

The parameters   `P1`, `P2` and `P3` set out which cells of the table can be filled:

- ▶ `P1`: Number

- ▶ `P2`: Empty cells between two value outputs, depending on `DIRECTION` (`down`, `right`) either empty rows or empty columns

- ▶ `P3`: Empty cells between two value outputs, depending on `DIRECTION` (`down`, `right`) either empty rows or empty columns

The value, status and time of an archive entry can be read in. All entries of the time range of the log are shown in the log going downwards or rightwards from the cell in which the function was given, with the row format defined by the parameters P1-P3.

WHEN SELECTING `DOWN`, THE FOLLOWING APPLIES:

- ▶ `P1` = number of rows to be filled with values

- ▶ `P2` = empty rows between the values

- ▶ `P3` = empty columns

WHEN SELECTING `RIGHT`, THE FOLLOWING APPLIES:

- ▶ `P1` = number of columns to be filled

- ▶ `P2` = empty rows between the values

- ▶ `P3` = empty rows between the values

STATUS TEXTS

The configuring is not done via the input help but directly in the cell. For this, the following applies:

- ▶ When entering the status texts, the defined short texts from the `project.ini` file or from the system-internal default configuration must be used. This applies regardless of whether the statues are input as fixed or read in from another cell.

- ▶ Several statuses are separated by a comma (`,`).

- ▶ The calculated values are updated regardless of whether the newly calculated value has changed in comparison to the read-in value.

- ▶ The status is changed to manual value (`MAN_VAL`).

> **Example**
>
> Reductions from monthly values to annual values as an archive value:
>
> In a monthly report with daily values, the amount of days on which a limit x was exceeded is calculated. The sum of the limit violations is written into a cell of the report.
>
> - ▸ In an archive, a value was defined, that gets one entry per month.
>
> - ▸ This entry is read in via `archivew` in the report.
>
> - ▸ `=archivew(4,"value","bottom",B12,C12)`
>
>   - • `4`: variable from archive filter
>   - • `value`: Parameter of variables that are to be displayed
>   - • `bottom`: Direction in which values are written
>   - • `B12`: Start cell, corresponds to the calculated cell
>   - • `C12`: Defines status text to be given
>
> - ▸ The sum is written to the archive value when the archive values are saved.

archivetime

The `archivetime` function supplies, depending on the time filter that has been set, the time information for the start and end of the archive interrogation in online operation.

Syntax: `=archivetime(FILTERINDEX,PARAMETER)`

Transfer parameters	Valid range	Comment
FILTERINDEX	[1-n]	Corresponding entry from the archive filter (on page 45).
PARAMETER	absolutetimefrom, absolutetimeto	Start time or end time of the archive request.

batch

This function outputs the first batch names for the archive selected in the filter area of the report.

Syntax: `=batch(FILTERINDEX)`

Transfer parameters	Valid range	Comment
FILTERINDEX	[1-n]	Corresponding entry from the archive filter (on page 45).

Example

=batch(1)

First batch name for the archive of the variables placed at position 1 of the archive filter.

batch no.

This function displays different information about batches. The output is displayed as a list which is evaluated from top to bottom or from left to right. The following are displayed:

▶   Name of the batches

▶ Start point

▶ End point

Syntax: =batchnr(FILTERINDEX,PARAMETER,DIRECTION)

Transfer parameters	Valid range	Comment
FILTERINDEX	[1-n]	Corresponding entry from the archive filter (on page 45).
PARAMETER	name, time from, time to	The selected parameter is read from the archive and displayed
DIRECTION	bottom, left	Filling in the corresponding direction.

wmz

The =wmz wertet function evaluates counter values saved in the archive. Counter overflow is not automatically checked.

Syntax:
=wmz(FILTERINDEX,ACTIVATED,OVERFLOW,STARTVALUE,ENDVALUE,COUNTERINFO,ERRORCOUNTER)

Transfer parameters	Valid range	Comment
FILTERINDEX	[1-n]	Corresponding entry from the archive filter (on page 45).
AKTIVATED	0 = deactivated 1 = activated	
OVERFLOW	Value or cell reference	With an overflow of "1000" the function automatically knows, that the difference between the start value "998" and the end value "10" is 12.
STARTVALUE	Cell reference	Start value from the archive via cell reference.
ENDVALUE	Cell reference	End value from the archive via cell reference.
COUNTERINFO	Cell reference	
ERRORCOUNTER	Cell reference	

wmzz

The `wmzz(filter index,direction,activated,overflow)` function evaluates counter values saved in the archive.

Syntax: `=wmzz(filter index,direction,activated,overflow)`

Transfer parameters	Valid range	Comment
FILTERINDEX	[1-n]	Corresponding entry from the archive filter (on page 45).
DIRECTION	bottom, left	Filling in the corresponding direction.
AKTIVATED	0 = deactivated 1 = activated	
OVERFLOW	Value or cell reference	

Database functions

This function enables data from Access or SQL databases to be read off.

Syntax: `=sqldao (DATABASE,SQL-QUERY,PARA1,PARA2)`

> ⚠ **Attention**
>
> *SQLDAO does not work with:*
>
> ‣ `ntext`
>
> ‣ `sql_variant`
>
> ‣ `nvarchar(MAX)`
>
> ‣ `nchar(MAX)`

| Transfer parameters	Comment
DATABASE	Name of the *.mdb database or the OLEDB provider.
SQL query	Standard 'SELECT', 'INSERT', 'UPDATE', 'DELETE' and 'EXECUTE' database queries can be used here. Two-integer parameters can be transferred with `%d  %d`, and two-string parameters can be transferred with `%s  %s`
PARA1	parameters for SQL query
PARA2	parameters for SQL query

CONFIGURATION WITH FUNCTION ASSISTANT
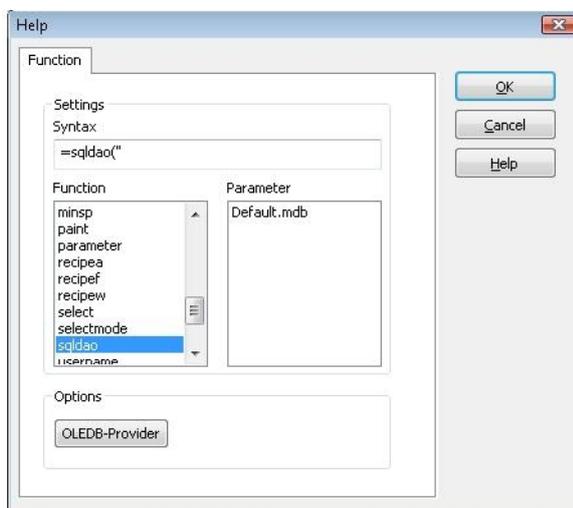
You can use MS Access databases and databases with OLEDB provider. For this, use the following syntax:

Database	Syntax
MS Access database	`=sqldao` (database,SQL query,Para1,Para2)
Databases with OLEDB provider	Replace the reference to a database with an OLEDB initialization string. This string can be ▸ entered manually or ▸ created by clicking on the **OLEDB provider** button.

DATABASES WITH OLEDB PROVIDER

Clicking on the **OLEDB provider** button opens the dialog for **data linking properties:**
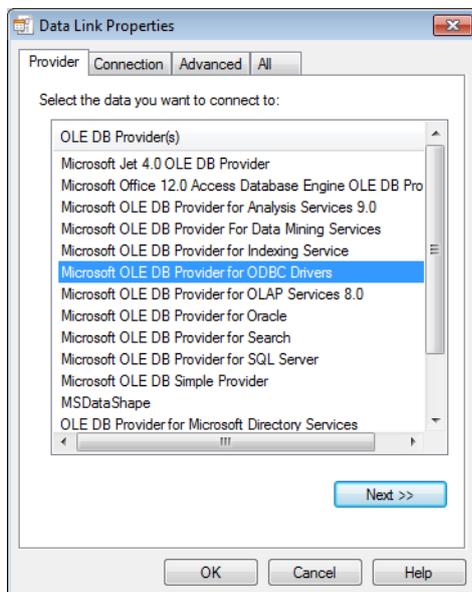
Here, you define the connection to the desired database:

▸ OLEDB provider type

▸ Data source

▸ Registration information

▸ etc.

After confirmation of the dialog by clicking on **OK**, the initialization string is generated and inserted into the `Syntax` field.

Syntax: `=sqldao (Provider,SQL query,Para1,Para2)`

> ⚠ **Attention**
>
> *The initialization string must always start with the character sequence* `Provider=`.
> *Note upper case and lower case letters, because the string is* `case sensitive`.

SETTINGS FOR MS ACCESS AND OLEDB

Transfer parameters	Description
`Database`	Name of the *.mdb database. Storage directory: Project folder which is superordinate to the Runtime folder. E. g.: Runtime storage location: Subfolder of `C:\Users\Public\Documents\zenon_Projects\Multi\TEST_PROJEKT` Save location *.mdb: saved directly in the folder`C:\Users\Public\Documents\zenon_Projects\Multi\TEST_PROJEKT` The project folder is defined in the project properties.
`Provider`	OLEDB provider; the initialization string must always begin with `Provider=` (case sensitive).
`SQL query`	You can use standard 'select' database queries here. Two-integer parameters can be transferred with `%d  %d` and two-string parameters can be transferred with `%s  %s`
`Para1, Para2`	Parameters for SQL query.

Example of access

EXAMPLES FOR MS ACCESS

Example database: `Address.mdb`

Name	First name	Street	City	Telephone	CanAddress
Müllner	Egon	Am Rheinberg 22	5020 Salzburg	0662/329354	1
Roider	Jürgen	Mozartgasse 7	5020 Salzburg	0662/329354	2
Leitner	Hans	Strubergasse 6	5020 Salzburg	0662/329354	3
Müller	Peter	Blumengasse 6	80003 München	089/234243	5
Weinberger	ww	Hafnergasse 5	7020 Klagenfurt	0222/2222	6

EXAMPLE 1

All entries from the `Addresses` table with `Name` field names are to be given.

Function: `=sqldao("Address.mdb","SELECT Address.name FROM Address ;",1,1)`

`Address.mdb`	Access database (must be in the project database path)
`SELECT`	Tells the Microsoft Jet database module to return information from the database as a group of records
`Address.name`	table.field name
`FROM`	states the table of query in which to find the fields stated in the SELECT statement
`Address`	Table

Result:

Müllner
Roider
Leitner
Müller
Weinberger

EXAMPLE 2

- ▸ Lower limit: Contents of cell `E1`
- ▸ Upper limit: Contents of cell `E2`
- ▸ Field name for evaluation: `CanAddress`

▶ Entries from the `Addresses` table with `Name` and `First name` field names are to be given.

▶ The `CanAddress` condition must be higher than contents of cell `E1` and lower than contents of cell `E2`.

Function: `=sqldao(Adress.mdb,SELECT Address.name,Address.first name FROM Address WHERE (((Adressen.CanAdresse)>%d) AND ((Address.CanAdresse)<%d)) ;,E1,E2)`

Address.mdb	Access database (must be in the project database path)
SELECT	Tells the Microsoft Jet database module to return information from the database as a group of records
Address.name	table.field name
,	SQL separator
Address.first name	table.field name
FROM	states the table of query in which to find the fields stated in the SELECT statement
Address	Table
WHERE	States which records of the table defined in the FROM section are affected by a SELECT, UPDATE or DELETE statement
(((Address.CanAddress)>%d)	Condition: table.field name
AND	connection
(((Address.CanAddress)<%d)	Condition: table.field name
;	SQL statement end
E1, E2:	Cell references (integer)

Result for E1 = 1, E2 = 6

Roider	Jürgen	2
Leitner	Hans	3
Müller	Peter	5

> ### 🔆 Info
>
> *For further information on SQL statements refer to the Online-Help of Access.*

SQL example

EXAMPLE FOR DATABASES WITH OLEDB PROVIDER

```
=sqldao("Provider=SQLNCLI.1;Integrated Security=SSPI;Persist
Security Info=False;Initial Catalog=<databasename>;Data
Source=<database instance>;Use Procedure for Prepare=1;Auto
Translate=True;Packet Size=4096;Workstation ID=<PCNAME>;Use
Encryption for Data=False;Tag with column collation when
possible=False","SELECT <tablename>.<columnname> FROM
<tablename>;",0,0)
```

Transfer parameters	Comment
DATABASE	"Provider=SQLNCLI.1;Integrated Security=SSPI;Persist Security Info=False;Initial Catalog=<databasename>;Data Source=<database instance>;Use Procedure for Prepare=1;Auto Translate=True;Packet Size=4096;Workstation ID=<PCNAME>;Use Encryption for Data=False;Tag with column collation when possible=False"
SQL query	"SELECT <tablename>.<columnname> FROM <tablename>;"
PARA1	0
PARA2	0

VALUES FOR EXAMPLE

```
Database: "AdventureWorks"

Installation: AdventureWorksDB.msi

Source: http://msftdbprodsamples.codeplex.com/releases/view/4004

Computer name: "PC1"
```

```
Authentication: "Windows NT"

SQL server instance: "SQLEXPRESS"
```

SELECT

SELECT

=sqldao("Provider=SQLOLEDB.1;Integrated Security=SSPI;Persist Security Info=False;Initial Catalog=AdventureWorks;Data Source=.\SQLEXPRESS;Use Procedure for Prepare=1;Auto Translate=True;Packet Size=4096;Workstation ID=PC1;Use Encryption for Data=False;Tag with column collation when possible=False","SELECT TOP 100 [AddressID],[AddressLine1],[AddressLine2],[City],[StateProvinceID],[PostalCode] FROM [AdventureWorks].[Person].[Address] ORDER BY AddressID;",0,0)

SELECT WITH STRING PARAMETER (CELL H2)

=sqldao("Provider=SQLOLEDB.1;Integrated Security=SSPI;Persist Security Info=False;Initial Catalog=AdventureWorks;Data Source=.\SQLEXPRESS;Use Procedure for Prepare=1;Auto Translate=True;Packet Size=4096;Workstation ID=PC1;Use Encryption for Data=False;Tag with column collation when possible=False","SELECT TOP 100 [AddressID],[AddressLine1],[AddressLine2],[City],[StateProvinceID],[PostalCode] FROM [AdventureWorks].[Person].[Address] WHERE (City = N'%s') ORDER BY AddressID;",H2,0)

SELECT WITH NUMERICAL PARAMETER (CELL H2)

=sqldao("Provider=SQLOLEDB.1;Integrated Security=SSPI;Persist Security Info=False;Initial Catalog=AdventureWorks;Data Source=.\SQLEXPRESS;Use Procedure for Prepare=1;Auto Translate=True;Packet Size=4096;Workstation ID=PC1;Use Encryption for Data=False;Tag with column collation when possible=False","SELECT TOP 100 [AddressID],[AddressLine1],[AddressLine2],[City],[StateProvinceID],[PostalCode] FROM [AdventureWorks].[Person].[Address] WHERE (StateProvinceID = %d) ORDER BY AddressID;",H2,0)

INSERT

=sqldao("Provider=SQLOLEDB.1;Integrated Security=SSPI;Persist Security Info=False;Initial Catalog=AdventureWorks;Data Source=.\SQLEXPRESS;Use Procedure for Prepare=1;Auto Translate=True;Packet Size=4096;Workstation ID=PC1;Use Encryption for Data=False;Tag with column

collation when possible=False","**INSERT INTO** [AdventureWorks].[Sales].[Currency] (CurrencyCode,Name,ModifiedDate) **VALUES** ('AAA','NewValue','');",0,0)

UPDATE

=sqldao("Provider=SQLOLEDB.1;Integrated Security=SSPI;Persist Security Info=False;Initial Catalog=AdventureWorks;Data Source=.\SQLEXPRESS;Use Procedure for Prepare=1;Auto Translate=True;Packet Size=4096;Workstation ID=PC1;Use Encryption for Data=False;Tag with column collation when possible=False","**UPDATE** Sales.Currency **SET** Name = 'ModifiedValue' **WHERE** CurrencyCode = 'AAA';",0,0)

DELETE

=sqldao("Provider=SQLOLEDB.1;Integrated Security=SSPI;Persist Security Info=False;Initial Catalog=AdventureWorks;Data Source=.\SQLEXPRESS;Use Procedure for Prepare=1;Auto Translate=True;Packet Size=4096;Workstation ID=PC1;Use Encryption for Data=False;Tag with column collation when possible=False","**DELETE** Sales.Currency **WHERE** CurrencyCode = 'AAA';",0,0)

EXECUTE

EXECUTE with numerical parameter (cell J2)

=sqldao("Provider=SQLOLEDB.1;Integrated Security=SSPI;Persist Security Info=False;Initial Catalog=AdventureWorks;Data Source=.\SQLEXPRESS;Use Procedure for Prepare=1;Auto Translate=True;Packet Size=4096;Workstation ID=PC1;Use Encryption for Data=False;Tag with column collation when possible=False","**EXECUTE** dbo.uspGetBillOfMaterials %d, '2001-05-08 12:35:29.123';",J2,0)

Date and time functions

This function gives date and time values.

You can find the formatting keys in the Data types for date and time (on page 162) chapter.

date

The `date(iYear, iMonth, iDay)` function date calculates the date from the numerical values for year (`iYear`), month (`iMonth`) and day (`iDay`).

Syntax: `=date(iYear,iMonth,iDay)`

Transfer parameters	Valid range	Comment
`iYear`	[0100-2200]	
`iMonth`	[1-12]	Values out of the valid range are set to the upper or lower limit
`iDay`	[1-(28,29,30,31)]	Values out of the valid range are set to the upper or lower limit

Result format: date

Standard format: dd/mm/yyyy

Example

`=date(2010,5,12)` is **12.05.2010**

*Cell A1=2010, cell A2=5, cell A3=12, cell A4=date(A1,A2,A3) is **12.05.2010***

datevalue

The `datevalue(date_text)` function calculates the date from the date text (`date_text`).

Syntax: `=datevalue(date_text)`

day

The `day(date)` function calculates the day from a date value.

Syntax: `=day(date)`

Result format: Integer

> **Example**
>
> *=now() with format string hh:mm:ss dd/mm/yy is 12:46:24 05.06.10*
> *Cell A2=day(A1) is 5*

hour

The hour(time) function calculates the hour from a time value.

Syntax: =hour(time)

Result format: Integer

> **Example**
>
> *=now() with format string hh:mm:ss dd/mm/yy is 12:46:24 05.06.10*
> *Cell A2=hour(A1) is 12*

minute

The minute(time) function calculates the minutes from a time value.

Syntax: =minute(time)

Result format: Integer

> **Example**
>
> *=now() with format string hh:mm:ss dd/mm/yy is 12:46:24 05.06.10*
> *cell A2=minute(A1) is 46*

month

The `month(date)` function calculates the month from the date value.

Syntax: =month(date)

Result format: Integer

> **Example**
>
> *=now() with format string hh:mm:ss dd/mm/yy is 12:46:24 05.06.10*
> *Cell A2=month(A1) is 6*

now

The `now()` function displays the current date and time.

The cell automatically gets the format string dd/mm/yy.

Syntax: =now()

> **Example**
>
> *=now() is 05.06.2010*
> *=now() with format string hh:mm:ss dd/mm/yy is 12:46:24 05.06.10*

second

The `second(time)` function calculates the seconds from a time value.

Syntax: =second(time)

Result format: Integer

time

The `time(iHour, iMinute, iSecond)` function provides the time from the numerical values for hour (`iHour`), minute (`iMinute`) and seconds (`iSecond`).

The cell automatically gets the format string hh:mm:ss.

Syntax: `=time(iHour,iMinute,iSecond)`

Transfer parameters	Valid range	Comment
`iHour`	[0-23]	Values out of the valid range are set to the upper or lower limit
`iMinute`	[0-59]	Values out of the valid range are set to the upper or lower limit
`iSecond`	[0-59]	Values out of the valid range are set to the upper or lower limit

Result format: `time`

> 📑 **Example**
>
> `=time(12,24,48)` is **12:24:48**
>
> *Celle A1=12, cell A2=24, cell A3=48, cell A4=time(A1,A2,A3) is 12:24:48*

today

The `today()` function displays the current date.

The cell automatically gets the format string dd/mm/yy.

Syntax: `=today()`

> 📋 **Example**
>
> *=today() is 05.06.2010*
> *=today() with format string hh:mm:ss dd/mm/yy is 12:00:00 AM 05.06.10*

weekday

The `weekday(date)` function calculates the weekday from the date value.

Syntax: `=weekday(date)`

value	weekday
1	Sunday
2	Monday
3	Tuesday
4	Wednesday
5	Thursday
6	Friday
7	Saturday

Result format: Integer

> 📋 **Example**
>
> *=now() with format string hh:mm:ss dd/mm/yy is 12:46:24 05.06.10*
> *Cell A2=weekday(A1) is 3*

year

The `year(date)` function calculates the year from the date value.

Syntax: `=year(date)`

Result format: Integer

> ### ⬛ Example
>
> *=now() with format string hh:mm:ss dd/mm/yy   is 12:46:24 05.06.10*
> *Cell A2=year(A1) is 2010*

Logical functions

This section summarizes all logical functions of the report generator

and

The `and(logical,logical)` function provides the `TRUE` value if both arguments are `TRUE`.

The cell automatically gets the format string `True;False`.

Syntax: `=and(logical,logical)`

EXAMPLE

Cell A1	Cell A2	Cell A3=and(A1,A2)
false	false	false
false	true	false
true	false	false
true	true	true

exact

The `exact(string1,string2)` function returns the value `TRUE`, if both text strings are identical (case-sensitive).

Syntax: `=exact(string1,string2)`

false

The `false()` function returns the value `FALSE`.

The cell automatically gets the format string `True;False`.

Syntax: `=false()`

> **Example**
>
> *The entry in cell A1=`false()` is `False`*

if

The `if(exprTest,valTrue,valFalse)` function provides the content of `valTrue` (value or string), if the argument is `exprTest` TRUE, otherwise the content of `valFalse`. It is possible to nest `if` functions, for example by implementing `and`-links.

Syntax: `=if(exprTest,valTrue,valFalse)`

isblank

The `isblank(ref)` function has the logical value TRUE, if the reference cell is empty.

Syntax: `=isblank(ref)`

EXAMPLE

ContentA1	A1	Result
Logical expression	No	False
Text	This is a text	False
empty		True
Number	123.00	False
Error	#ARG!	#ARG!

iserror

The `iserror(ref)` function has the logical value TRUE if the reference cell contains an error.

Syntax: `=iserror(ref)`

EXAMPLE

ContentA1	A1	Result
Logical expression	No	False
Text	This is a text	False
empty		False
Number	123.00	False
Error	#ARG!	True

islogical

The `islogical(ref)` function has the logical value TRUE if the reference cell contains a logical value.

Syntax: `=islogical(ref)`

Example

ContentA1	A1	Result
Logical expression	No	True
Text	This is a text	False
empty		False
Number	123.00	False
Error	#ARG!	#ARG!

isnontext

The `isnontext(ref)` function has the logical value TRUE if the reference cell does not contain a text string.

Syntax: `=isnontext(ref)`

EXAMPLE

ContentA1	A1	Result
Logical expression	No	True
Text	This is a text	False
empty		True
Number	123.00	True
Error	#ARG!	#ARG!

isnumber

The `isnumber(ref)` function has the logical value TRUE if the reference cell contains a numerical value.

Syntax: `=isnumber(ref)`

EXAMPLE

ContentA1	A1	Result
Logical Link	No	False
Text	This is a text	False
empty		False
Number	123.00	True
Error	#ARG!	#ARG!

isref

The `isref(ref)` function has the logical value `TRUE` if the reference cell contains a reference to another cell.

Syntax: `=isref(ref)`

EXAMPLE

ContentA1	A1	Result
Logical expression	No	False
Text	This is a text	False
empty		False
Number	123.00	False
Error	#ARG!	False
Reference	=B1	True

istext

The `istext(ref)` function has the logical value `TRUE` if the reference cell contains a text string.

Syntax: `=istext(ref)`

EXAMPLE

ContentA1	A1	Result
Logical Link	No	False
Text	This is a text	True
empty		False
Number	123.00	False
Error	#ARG!	#ARG!

not

The `not(logical)` function returns the logical opposite of the argument.

The cell automatically gets the format string `True;False`.

Syntax: `=not(logical)`

EXAMPLE

Cell A1	Cell A2=not(A1)
false	True
true	False

or

The `or(logical,logical)` function provides the `TRUE` logical value if both arguments are `TRUE`.

The cell automatically gets the format string `True;False`.

Syntax: `=or(logical,logical)`

EXAMPLE

Cell A1	Cell A2	Cell A3=`or(A1,A2)`
false	false	false
false	true	true
true	false	true
true	true	true

true

The `true()` function returns the logical value `TRUE`.

The cell automatically gets the format string `True;False`.

Syntax: `=true()`

> 📑 **Example**
>
> *The entry in cell A1=`true()` is `True`*

Mathematical and trigonometric functions

Common mathematical functions are available for reports. These can not only be planned using the function assistant, but can also be directly entered into the input row.

abs

The `abs (num)` function returns the absolute value of a number as a result. The absolute value of a number is the value without sign.

```
Syntax: =abs(num)
```

`num:` is the real number which is the basis for the absolute value

> **Example**
>
> =abs(2.2) is 2
>
> *=abs(-2.2) is 2*

acos

The `acos (num)` function provides the arc cosine of a number.

```
Syntax: =acos(num)
```

acosh

The `acosh (num)` function provides the inverted hyperbolic cosine of a number.

```
Syntax: =acosh(num)
```

asin

The `asin (num)` function provides the arc cosine of a number.

```
Syntax: =asin(num)
```

asinh

The `asinh (num))` function provides the inverted hyperbolic sinus of a number.

```
Syntax: =asinh(num)
```

atan

The `atan (num)` function provides the arc tangent of a number.

Syntax: `=atan(num)`

atanh

The `atanh (num)` function provides the inverted hyperbolic tangent of a number.

Syntax: `=atanh(num)`

atan2

The `atan2 (x,y)` function provides the arc tangent based on an x and y coordinate.

Syntax: `=atan2(x, y)`

ceil

The `ceil (num)` function returns the next higher whole-numbered value of the value, rounded up.

Syntax: `=ceil(num)`

> **Example**
>
> `=ceil(2.2)` is 3
>
> *=ceil(-2.2) is -3*

column

The `column (ref)` function provides the column number of a connection (for example C3=2) in the value range of 0 to n.

Syntax: `=column(ref)`

columns

The `columns (range)` function returns the numbers of columns in the stated range.

`Syntax: =columns(range)`

cos

The `cos(num)` function provides the cosine of a number.

`Syntax: =cos(num)`

EXAMPLE

`=cos(Input)`

	Input	**Result**
`degrees`	`radians()`	`cosine`
`0,00`	`0`	`1,000`
`45,00`	`0,78539816339744828`	`0,707`
`90,00`	`1,5707963267948966`	`0,000`
`135,00`	`2,3561944901923448`	`-0,707`
`180,00`	`3,1415926535897931`	`-1,000`
`225,00`	`3,9269908169872414`	`-0,707`
`270,00`	`4,7123889803846897`	`-0,000`
`315,00`	`5,497787143782138`	`0,707`
`360,00`	`6,2831853071795862`	`1,000`

cosh

The `cosh(num)` function provides the hyperbolic cosine of a number.

`Syntax: =cosh(num)`

degree

The `degree(num)` function converts radians to degrees.

`Syntax: =degree(num)`

EXAMPLE

`=degree(Input)`

Input	Result
`Radians()`	`Degree()`
0	0
0,78539816339744828	45
1,5707963267948966	90
2,3561944901923448	135
3,1415926535897931	180
3,9269908169872414	225
4,7123889803846897	270
5,497787143782138	315
6,2831853071795862	360

even

The `even(num)` function returns the next higher even whole-numbered value. Negative values are rounded from 0 away.

`Syntax: =even(num)`

> **Example**
>
> =`even(2.2)` is 4
>
> *=even(-2.2) is -4*

exp

The `exp(num)` function potentializes the basis `e` with the number stated as argument.

`Syntax: =exp(num)`

> **Example**
>
> =`exp(0)` is 1
>
> =`exp(1)` is 2.718282
>
> *=exp(10) is 22026.465795*

fact

The `fact(int)` function provides the factorial of a number.

`Syntax: =fact(int)`

> **Example**
>
> `=fact(0)` is `0` `(0)`
>
> `=fact(1)` is `1` `(1)`
>
> `=fact(2)` is `2` `(1*2)`
>
> `=fact(3)` is `6` `(1*2*3)`
>
> `=fact(4)` is `24` `(1*2*3*4)`
>
> `=fact(5)` is `120` `(1*2*3*4*5)`

factdouble

The `factdouble(num)` function provides the double factorial of a number. In doing so, only the product of each second number is formed.

`Syntax: =factdouble(num)`

> **Example**
>
> `=factdouble(0)` is `0` `(0)`
>
> `=factdouble(1)` is `1` `(1)`
>
> `=factdouble(2)` is `2` `(2)`
>
> `=factdouble(3)` is `3` `(1*3)`
>
> `=factdouble(4)` is `8` `(2*4)`
>
> `=factdouble(5)` is `15` `(1*3*5)`

floor

The `floor(num)` function returns the next lower whole-numbered value of the value, rounded down. Negative values are rounded in direction to 0.

`Syntax: =floor(num)`

> 📋 **Example**
>
> =floor(2.2) is 2
>
> *=floor(-2.2) is -2*

int

The `int(num)` function returns the next lower whole-numbered value of the value, rounded down. Negative values are rounded from 0 away.

`Syntax: =int(num)`

> 📋 **Example**
>
> =int(2.2) is 2
>
> *=int(-2.2) is -3*

ln

The `ln(num)` function provides the natural logarithm of a number.

`Syntax: =ln(num)`

> **Example**
>
> =`ln(1)` is 0
>
> =`ln(2.718282)` is 1
>
> *=ln(22026.465795) is 10*

log10

The `log10(num)` function calculates the logarithm of a value on the basis 10.

`Syntax: =log10(num)`

> **Example**
>
> =`log10(1)` is 0
>
> =`log10(10)` is 1
>
> =`log10(100)` is 2
>
> *=log10(1000) is 3*

mod

The `mod(num, divisor)` function returns the rest of the division of a value (`num`) by the divisor (`divisor`).

`Syntax: =mod(num,divisor)`

> **Example**
>
> =mod(0.3) is 0
>
> =mod(1.3) is 1
>
> =mod(2.3) is 2
>
> =mod(3.3) is 0
>
> *=mod(4.3) is 1*

odd

The function odd(num) returns the next higher odd whole-numbered value. Negative values are rounded from 0 away.

Syntax: =odd(num)

> **Example**
>
> =odd(2.2) is 3
>
> *=odd(-2.2) is -3*

pi

The pi() function pi returns the value for pi: 3.1415926535897931

Syntax: =pi()

Standard format: *0.######

> **Example**
>
> *=pi() is with standard formatting 3.141593*

product

The `product (range)` function calculates the product of the values in the stated area.

`Syntax: =product(range)`

EXAMPLE

Cells A1 to A10: any numbers

Cell `B1=product(A1:A10)`

Cell/column	A	B
1	1	362880 0
2	2	
3	3	
4	4	
5	5	
6	6	
7	7	
8	8	
9	9	
10	10	

The result for the range A1 to A10 is `3628800`.

quotient

The `quotient(num,divisor)` function returns the whole-numbered part of a division.

`Syntax: =quotient(num,divisor)`

> **Example**
>
> =quotient(0.3) is 0
>
> =quotient(1.3) is 0
>
> =quotient(2.3) is 0
>
> =quotient(3.3) is 1
>
> *=quotient(4.3) is 1*

radians

The radians(num) function converts degrees into radians.

Syntax: =radians(num)

num: input in degrees

EXAMPLE

=radians(Input)

Input	Result
0,00	0
45,00	0,78539816339744828
90,00	1,5707963267948966
135,00	2,3561944901923448
180,00	3,1415926535897931
225,00	3,9269908169872414
270,00	4,7123889803846897
315,00	5,497787143782138
360,00	6,2831853071795862

rand

The `rand()` function returns a new random value between `0.000` and `1,000` every time the report is calculated.

```
Syntax: =rand()
```

randbetween

The `randbetween(min,max)` function returns a new random value between `min` and `max` every time the report is calculated.

```
Syntax: =randbetween(min,max)
```

round

The `round(num,digits)` function rounds a value (`num`) to a stated number of decimals (`digits`).

```
Syntax: =round(num,digits)
```

row

The `row(ref)` function returns the row number of a reference (e.g. C3=3) in the value range from `0` to `n`.

Syntax: `=row(ref)`

rows

The `rows(range)` function calculates the number of empty cells in the defined range.

Syntax: `=rows(range)`

sign

The `sign(num)` function provides the value

- ▶ `1`, if the number (`num`) is positive

- ▶ `-1` if the number is negative

- ▶ `0` if the number is `0`

Syntax: `=sign(num)`

> **Example**
>
> =sign(2.2) is 1
>
> =sign(-2.2) is -1
>
> *=sign(0.0) is 0*

sin

The `sin(num)` function provides the sine of a number.

Syntax: `=sin(num)`

`num`: input in radians

EXAMPLE

```
=sin(Input)
```

	Input	Result
degrees	**radians()**	**sine**
0,00	0	0,000
45,00	0,78539816339744828	0,707
90,00	1,5707963267948966	1,000
135,00	2,3561944901923448	0,707
180,00	3,1415926535897931	0,000
225,00	3,9269908169872414	-0,707
270,00	4,7123889803846897	-1,000
315,00	5,497787143782138	-0,707
360,00	6,2831853071795862	-0,000

sinh

The `sinh(num)` function provides the hyperbolic sine of a number.

```
Syntax: =sinh(num)
```

sqrt

The `sqrt(num)` function provides the square root of a number.

```
Syntax: =sqrt(num)
```

> **Example**
>
> `=sqrt(1)` is 1
>
> `=sqrt(2)` is 1,414214
>
> `=sqrt(3)` is 1.732051
>
> *=sqrt(4) is 2*

sqrtpi

The `sqrtpi(num)` function provides the square root of a `pi` value.

`Syntax: =sqrtpi()`

> **Example**
>
> `=sqrtpi(1)` is 3.141593 = 1*pi
>
> `=sqrtpi(2)` is 4.442883 = `sqrt(2)*pi`
>
> `=sqrtpi(3)` is 5.441398 = `sqrt(3)*pi`
>
> *=sqrtpi(4) is 6.283185 = sqrt(4)*pi*

tan

The `tan(num)` function provides the tangent of a number.

`Syntax: =tan(num)`

tanh

The `tanh(num)` function provides the hyperbolic tangent of a number.

```
Syntax: =tanh(num)
```

trunc

The `trunc(num)` function returns the next lower whole-numbered value of the value, rounded down. Negative values are rounded in direction to 0.

```
Syntax: =trunc(num)
```

> ### 📑 Example
>
> =`trunc(2.2)` is 2
>
> *=trunc(-2.2) is -2*

Recipegroup Manager functions

Variables from other loaded projects that are used in recipes can also be displayed with the report functions for the recipe group manager. These functions can only be used with screens of the recipegroup manager type.

The function assistant (on page 51) is available for configuration.

recipea

The =`recipeaInhalt)` function displays general data of a recipe of the recipe group manager.

```
Syntax: =recipea(Inhalt)
```

Contents	Description
`muser`	User who last edited the recipe.
`mzeit`	Time of the last change.
`rez`	Name of the recipe.
`rezg`	Name of the recipe group to which the recipe belongs.
`reznum`	Number of the recipe
`text1 to text 8`	User comments 1 to 8 of the recipe.
`text2`	User comment 2 for the recipe.
version	Version number of the recipe.
status	Status of the recipe as text. For example: **1 - sample recipe**

📘 **Example**

=recipea(recnum)

recipef

The =recipef(content,variable) function displays variable-orientated data of a recipe of the recipe group manager. These data can freely be positioned in the report.

Syntax: =recipef(content,variable)

Contents	Description
`name`	Name of the variable.
`identification`	Identification of the variable.
`type`	Source type.
`unit`	Technical unit of the variable.
`value`	Value for the variable in the recipe.
`symbols`	Display of limit value texts from the reaction matrix or limit value as text.
`min`	Minimum of the permitted value range for the variable in this recipe.
`max`	Maximum of the permitted value range for the variable in this recipe.
`actual_value`	If the defined variable does not exist in the selected recipe, the cell is marked with −.
`Interlocking`	Display of Interlocking. Because the names are not available in Runtime, only the IDs of the interlockings selected in the recipe group manager are displayed.
`visibility`	Shows visibility of the variables.
`shadow variable`	Shows the name of the linked shadow variable. Only display in the Runtime. Configuration in the Editor.
`filter`	Filter text that was selected in the RGM.
`Keyboard`	Shows linked keyboard keyboard.
`Set value limit`	`Yes:` Takes over set value limit to the variable. Only display in the Runtime. Configuration in the Editor.

Example

=recipef("value","temperature")

recipew

The =recipew(content,direction) function displays variable-orientated data of a recipe of the recipe group manager in table form.

Syntax: =recipew(content,direction)

Contents	Description
name	Name of the variable.
identification	Identification of the variable.
type	Source type.
unit	Technical unit of the variable.
value	Value for the variable in the recipe.
symbols	Display of values as text.
min	Minimum of the permitted value range for the variable in this recipe.
max	Maximum of the permitted value range for the variable in this recipe.
actual_value	If the defined variable does not exist in the selected recipe, the cell is marked with −.
Interlocking	Display of Interlocking. Because the names are not available in Runtime, only the IDs of the interlockings selected in the recipe group manager are displayed.
visibility	Shows visibility of the variables.
shadow variable	Shows the name of the linked shadow variable. Only display in the Runtime. Configuration in the Editor.
filter	Filter text that was selected in the RGM.
Keyboard	Shows linked keyboard keyboard.
Set value limit	Yes: Takes over set value limit to the variable. Only display in the Runtime. Configuration in the Editor.

📧 **Example**

=recipew(name,bottom)

Statistical functions

Common statistical functions are available for reports. These can not only be planned using the function assistant, but can also be directly entered into the input row.

avg

The `avg (range)` function calculates the average of the values within the range.

`Syntax: =avg(range)`

EXAMPLE

Cell A4= avg(A1:A3)

Row/column	A
1	1
2	3
3	5
4	3

The average of the cells A1 to A3 is 3

count

The `count (range)` function returns the number of numerical values in the stated range.

`Syntax: =count(range)`

EXAMPLE

Cell A5= `count(A1:A4)`

Row/column	A
1	1
2	Text
3	
4	3
5	2

The number of numerical values in the cells A1 to A4 is `2`.

counta

The `counta (range)` function calculates the number of non-empty cells in a given range.

`Syntax: =counta(range)`

EXAMPLE

Cell `A5=counta(A1:A4)`

Row/column	A
1	1
2	text
3	
4	3
5	3

The number of non-empty cells in the range`A1` to `A4` is `3`.

countblank

The `countblank (range)` function calculates the number of empty cells in a given range.

`Syntax: =countblank(range)`

EXAMPLE

Cell `A5 = countblank(A1:A4)`

Row/column	A
1	1
2	Text
3	
4	3
5	1

The number of empty cells in the range A1 to A4 is `1`.

max

The `max (range)` function calculates the highest value in a range.

`Syntax: =max(range)`

EXAMPLE

Cells A1 to A10: any numbers

Cell `B1=max(A1:A10)`

Row/column	A	B
1	1	10
2	2	
3	3	
4	4	
5	5	
6	6	
7	7	
8	8	
9	9	
10	10	

The highest value in range A1 to A10 is in cell A10 and is 10.

min

The `min (range)` function calculates the lowest value in a range.

`Syntax: =min(range)`

EXAMPLE

Cells A1 to A10: any numbers

    Cell `B1=min(A1:A10)`

Row/column	A	B
1	1	1
2	2	
3	3	
4	4	
5	5	
6	6	
7	7	
8	8	
9	9	
10	10	

The lowest value in range A1 to A10 is in cell A1 and is `1`.

percentile

The `percentile (range,num)` function returns as result a threshold value, from which the values are accepted.

`Syntax: =percentile(range,num)`

`Range:` Is a matrix or a data range that describes the relative position of the data.

`Num:` Integer from `0-100`

EXAMPLE

Cells A1 to A10: any numbers

Cell B1=percentile(A1:A10,90)

Row/column	A	B
1	1	9,1
2	2	
3	3	
4	4	
5	5	
6	6	
7	7	
8	8	
9	9	
10	10	

The threshold value for the range A1 to A10 is `9.1`.

stdev

The `stdev (range)` function returns an estimation of the standard deviation based on a random sample of the values in the defined area.

Formula:

$$s = \sqrt{\frac{n\sum x^2 - \left(\sum x\right)^2}{n(n-1)}}$$

`Syntax: =stdev(range)`

EXAMPLE

Cells A1 to A10: any numbers

Cell A11=stdev(A1:A10)

Row/column	A
1	745,00

2	730,00
3	732,00
4	768,00
5	769,00
6	747,00
7	718,00
8	723,00
9	766,00
10	739,00
11	18,785633

The standard deviation for the area A1 to A10 is estimated using on the sample on `18.785633` as a basis.

stdevp

The `stdevp (range)` function returns an estimation of the standard deviation based on the total of the values in the defined area.

Formula:

$$s = \sqrt{\frac{n \sum x^2 - (\sum x)^2}{n^2}}$$

`Syntax: =stdevp(range)`

EXAMPLE

Cells A1 to A10: any numbers

Cell `A11= stdevp(A1:A10)`

Row/column	A
1	745,00
2	730,00
3	732,00
4	768,00
5	769,00
6	747,00
7	718,00
8	723,00
9	766,00
10	739,00
11	17,821616

The standard deviation for the range A1 to A10 is estimated at `17.821616`.

sum

The `sum (range)` function calculates the sum of the values in the defined area.

`Syntax: =sum(range)`

EXAMPLE

Cells A1 to A10: any numbers

Cell B1=sum(A1:A10)

Row/column	A	B
1	1	55
2	2	
3	3	
4	4	
5	5	
6	6	
7	7	
8	8	
9	9	
10	10	

The sum of all values in the A1 to A10 range is 55.

sumsq

The `sumsq (range)` function calculates the sum of the squares of the values within a range.

`Syntax: =sumsq(range)`

EXAMPLE

Cells A1 to A10: any numbers

Cell A11=sumsq(A1:A10)

Row/column	A	B
1	1	1
2	2	4
3	3	9
4	4	16
5	5	25
6	6	36
7	7	49
8	8	56
9	9	81
10	10	100
11	385	385

SUPERVISION

Cell B1 to B10: Squares of A1 to A10

Cell `B11= sum(B1:B10)`

var

The `var (range)` function returns an estimation of the variance based on a random sample of the values in the defined area.

Formula:

$$v = \frac{n\sum x^2 - (\sum x)^2}{n(n-1)}$$

`Syntax: =var(range)`

EXAMPLE

Cells A1 to A10: any numbers

Cell A11=var(A1:A10)

Row/column	A
1	745,00
2	730,00
3	732,00
4	768,00
5	769,00
6	747,00
7	718,00
8	723,00
9	766,00
10	739,00
11	352,9

The variance for the area A1 to A10 is estimated using on the sample on `352.9` as a basis.

varp

The `varp (range)` function returns an estimation of the variation based on the total of the values in the defined area.

Formula:

$$v = \frac{n \sum x^2 - (\sum x)^2}{n^2}$$

`Syntax: =varp(range)`

EXAMPLE

Cells A1 to A10: any numbers

Cell A11=varp(A1:A10)

Row/column	A
1	745,00
2	730,00
3	732,00
4	768,00
5	769,00
6	747,00
7	718,00
8	723,00
9	766,00
10	739,00
11	317,61

The variance for the range A1 to A10 is estimated at `317.61`.

Text functions

Different text functions are available for reports. These can not only be planned using the function assistant, but can also be directly entered into the input row. Exception: paint (on page 135)

char

The function `char(code)` returns the ASCII character that is defined under `code`.

`Syntax: =char(code)`

`code`: number between `1` and `255`.

> **📑 Example**

=char(88) is X

> *If the cell A1 contains the value 35, the =char(A1) function gives the character # as a result.*

To determine the ASCII code for a character, use the code (on page 131) function.

clean

The clean(string) function deletes all control characters from a text character chain.

Syntax: =clean(string)

code

The code(string) function provides the ASCII code for the first character in a text character chain.

If the string is in the brackets of the text to be coded, it is put in quotation marks. If it refers to another row, this is entered.

Syntax: =code(string)

> **📑 Example**

=code("z") is 122.

> *Cell A1 contains the text zoom:*

=code(A1) is 122.

To determine the character corresponding to an ASCII code, use the char (on page 130) function.

> **⚠️ Attention**
>
> *The code(string) function only provides the ASCII code for the first character in a text string.*

find

The `find(keystring,string,start)` function provides the position of a character sequence (`keystring`) within a text string (`string`). The position is found in the text string starting at the start character (`start`) with case-sensitivity.

Syntax: `=find(keystring,string,start)`

> **Example**
>
> Cell A1: `cd`
>
> Cell A2: `ABCDabcd`
>
> Cell A3: `1`
>
> Cell A4=`find(A1,A2,A3)`: Result =`7`

fixed

The `fixed(num,decimals)` function formats a number and converts it into a defined character string.

Syntax: `=fixed(num,decimals)`

- ▶ `num`: Numerical value or cell reference

- ▶ `decimals`: Number of decimals

> **Example**
>
> Cell A1 = `12345.672`
>
> `=fixed(A1,2)` is `12345.67`
>
> `=fixed(A1,1)` is `12345.6`
>
> The separator and the display also depend on the regional settings and formatting settings in cell (on page 30).

Note: The fixed() function cannot be combined with a format string such as the stating of units.

DIFFERENCE TO ZENON 5.50 AND EARLIER VERSIONS

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

Syntax: =fixed(num,decimals,no_seps)

▶ `no-seps:`

- 0: A separator is inserted
- 1: A separator is left out

Figures are generally shown without thousand separators in later versions.

left

The `left(string,num)` function provides the number of characters from a character string defined in (`num`), starting with the first character.

Syntax: =left(string,num)

▶ `string`: Character string in quotation marks or reference to a cell with a character string

▶ `num`: Number of characters that are to be read off, starting with the first

> **Example**
>
> Cell A1 = `abcdefgh`
>
> =left(A1, 3) is `abc`
>
> *=left(A1,5) is abcde*

To read off characters beginning with the last character of a character string, use the right (on page 137) function.

len

The `len(string)` function provides the number of characters in a character string.

Syntax: =len(string)

> **Example**
>
> Cell A1 = `abc`
> =`len(A1)` is 3
>
> *Cell A1 = `abcdefgh`*
> *=`len(A1)` is 8*

lower

The `lower(string)` function replaces all upper case letters of a text character string with lower case letters.

Syntax: =`lower(string)`

> **Example**
>
> *Celle A1 = `AbCdEf`*
> *=`lower(A1)` is `abcdef`*

To replace lower case letters with upper case letters, use the upper (on page 139) function.

mid

The `mid(string,numStart,numLen)` function extracted

▶ From the defined character string `string`

▶ A new character string

▶ With a defined number of characters through the argument `numLen`

▶ From the start position defined by `numStart`.

Syntax: =`mid(string,numStart,numLen)`

> **Example**
>
> *Cell A1 = abcdefgh*
>
> *=mid(A1,2,4) is bcde*

paint

The =paint(referenz) function continues the character string given in the referenced cell, if there is not enough space for this in the referenced cell. Thus long character strings can be well displayed by being divided into several cells.

Syntax: =paint(referenz)

> **Example**
>
> *Cell A1 contains the characters: abcdefghijklmn*
>
> *In cell A1, the following is displayed due to the width of the cell: abcdefg*
>
> *=paint(A1) in cell A2 leads to this display:*
>
> *A1: abcdefg*
>
> *A2: hijklmn*

This text function can be configured with the function assistant.

proper

The proper(string) function transforms the first character of a word into upper case letters and all subsequent characters into lower case letters. The start of the word is the first alphabetical character of a character string. If the character string contains several non-alphabetic characters, a new word is started after each non-alphabetic character.

Syntax: = proper (string)

> **Example**
>
> Cell A1 = `AbCdEf`
> `=proper(A1)` is `Abcdef`
>
> Cell A1 = `aBcDeF`
> `=proper(A1)` is `Abcdef`
>
> *Cell A1 = `1aBcD2eF`*
> *`=proper(A1)` is 1Abcd2Ef*

replace

The `replace(string,numStart,numLen,stringNew)` function

▶ replaces, in the text character string `string`,

▶ from the start position defined by`numStart`,

▶ a number of characters defined by `numLen`

▶ with characters from the new text character string, stringNew

Syntax: `=replace(string,numStart,numLen,stringNew)`

> **Example**
>
> Cell A1 = `abcdefgh`
> Cell A2 = `xyz`
> `=replace(A1,3,2,A2)` is `abxyzefgh`
>
> *In this example, the character `cd` (3 positions in the string, 2 characters) in string `abcdefgh` is replaced by `xyz`.*

rept

The `rept(string,num)` function repeats the character string `string` for as many times as is defined in `num`.

Syntax: `=rept(string,num)`

> 📋 **Example**
>
> *Cell A1 = abc*
> *Cell A2 = 3*
> *=rept(A1,A2) is abcabcabc*

right

The `right(string,num)` function provides the number of characters from a character string defined in (`num`), starting with the last character.

Syntax: `=right(string,num)`

▸ `string`: Character string in quotation marks or reference to a cell with a character string

▸ `num`: Number of characters that are to be read off, starting with the last

> 📋 **Example**
>
> Cell A1 = abcdefgh
>
> =right(A1,3) is fgh
>
> *=right(A1,5) is defgh*

To read off characters beginning with the first character of a character string, use the left (on page 133) function.

search

The `search(keystring,string,start)` function provides

▸ the position of a `keystring` search text

- ▶ within a character string `string`

- ▶ from the start character position `start`

Syntax: `=search(string,keystring,start)`

EXAMPLE

Cell `A4=search(A1,A2,A3)`

Cell/column	A
1	abcdeABCDEabcde
2	cd
3	1
4	3

The function in cell A4 finds the text being searched for `cd` in cell A1 at position `3`.

substitute

The `substitute(string,oldstring,newstring,instance)` function

- ▶ replaces, in the character string `string`,

- ▶ a character sequence `oldstring`

- ▶ with a new character sequence `newstring`

- ▶ whereby `instance` determines the position of `oldstring`

  - • `instance=0` replaces all character strings found

The function is case sensitive.

Syntax: `=substitute(string, oldstring, newstring, instance)`

> **Example**
>
> Cell A1 = `abcdef ABCDEF abcdef ABCDEF abcdef ABCDEF`
> Cell A2 = `XYZ`
>
> **Example 1:**
>
> `=substitute(A1,"ABCDEF",A2,0)` is `abcdef XYZ abcdef XYZ abcdef XYZ`
>
> In this example, the whole character sequence `ABCDEF` in the entire string `abcdef ABCDEF abcdef ABCDEF abcdef ABCDEF` is replaced by `XYZ`
>
> **Example 2:**
>
> `=substitute(A1,"abcdef",A2,2)` is `abcdef ABCDEF XYZ ABCDEF abcdef ABCDEF`
>
> *In this example, the second character sequence* `abcdef` *found in the string* `abcdef ABCDEF abcdef ABCDEF abcdef ABCDEF` *is replaced by* `XYZ`*.*

Note: Character string within the function are put into quotation marks ("character string").

trim

The `trim(string)` function allows a maximum of one space between two characters. All other spaces are removed.

`Syntax: =trim(string)`

> **Example**
>
> *Cell A1 = Too many   spaces   are reduced to one.*
> *=trim(A1) is: Too many spaces are reduced to one.*

upper

The `upper(string)` function replaces all lower case letters of a text character string with upper case letters.

```
Syntax: =upper(string)
```

> 📑 **Example**
>
> *Cell A1 = AbCdEf*
>
> *=upper(A1)* *is ABCDEF*

To replace upper case letters with lower case letters, use the lower (on page 134) function.

Other functions

Other functions group together report functions that cannot be allocated to a sub-group.

The function assistant (on page 51) is available for configuration.

Alarm

This function shows information from the alarm administration as a formatted list in a report form.

```
Syntax: =alarm(FILTER,PARAMETER,DIRECTION)
```

Transfer parameters	Valid range	Comment
FILTER		Configuration via AML filter.
PARAMETER	▸ name ▸ identification ▸ text ▸ status ▸ value ▸ time received ▸ time sent ▸ time acknowledged ▸ time reactivated ▸ status reactivated ▸ number reactivated ▸ active ▸ number_h ▸ number_t ▸ number_m ▸ user ▸ computer ▸ note ▸ class ▸ group	Selection of the filter column to be read. Hints: ▸ project: Only present in the AML filter with multi-user projects ▸ number: only for project generator ▸ number_h: only for project generator ▸ number_t: only for project generator ▸ number_m: only for project generator

	‣ project	
	‣ number	
DIRECTION	‣ below	Filling in the corresponding direction.
	‣ Left	

The time filter results from the definition in Screen switching filter report (on page 164).

Direction

Options	**Meaning**
Right	Value is entered to the right (row).
	There must be sufficient cells left free for expected entries from further report creation.
	Attention: If an `archivesp` function is created with only one column, the value is entered downwards (in the column).
below	Value is entered downwards (column).
	There must be sufficient cells left free for expected entries from further report creation.
	Attention: If an `archivesp` function is created with only one row, the value is entered to the right (in the row).

> **Info**
>
> *To ensure that the function can dynamically insert lines or columns without overwriting other lines/columns the labeling of the corresponding row (on page 44) or column (on page 43) must be set to E (for "Expand").*

cel

This function shows information from the chronological event list (CEL) as a formatted list in a report. The configuration with the function assistant works along the lines of the configuration of the `aml` (on page 144) function.

Syntax: =cel(FILTER,PARAMETER,DIRECTION)

Transfer parameters	Valid range	Comment
FILTER		Configuration via CEL filter.
PARAMETER	▸ name ▸ identification ▸ text ▸ status ▸ value ▸ time received ▸ user ▸ computer ▸ note ▸ class ▸ group ▸ project ▸ number	Selection of the filter column to be read. Hints: ▸ project: Only present in the CEL filter with multi-user projects ▸ number: only for project generator
DIRECTION	bottom, left	Filling in the corresponding direction.

The time filter results from the definition in Screen switching filter report (on page 164).

Direction

Options	Meaning
`Right`	Value is entered to the right (row).
	There must be sufficient cells left free for expected entries from further report creation.
	Attention: If an `archivesp` function is created with only one column, the value is entered downwards (in the column).
`below`	Value is entered downwards (column).
	There must be sufficient cells left free for expected entries from further report creation.
	Attention: If an `archivesp` function is created with only one row, the value is entered to the right (in the row).

Info

To ensure that the function can dynamically insert lines or columns without overwriting other lines/columns the labeling of the corresponding row (on page 44) or column (on page 43) must be set to E (for "Expand").

Configuring the alarm and CEL with the function assistant

You can configure functions for AML and CEL with the function assistant. This example is carried out with `aml` and works along the lines of `cel`:

1. Select the desired cell in the context menu with the `Report functions` command.

2. The assistant is opened.

3. Select `Alarm`.
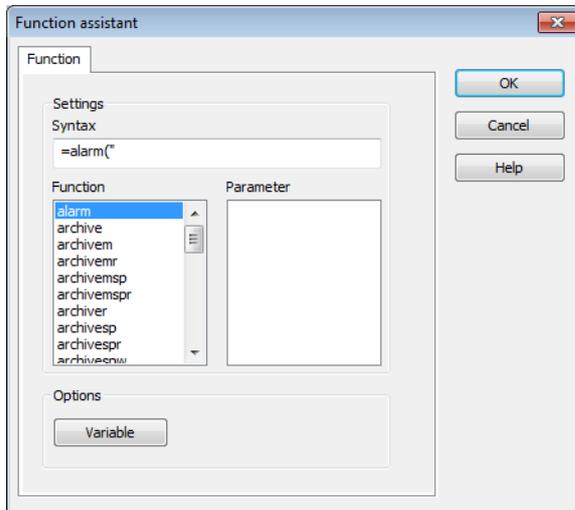
4. The `alarm` report function is entered into the syntax.

5. Clicking on `Variable` opens the Filter for AML. (cel: Filter for CEL.)

6. Configure the filter.

7. Select the desired parameter by double clicking, for example `identification`).
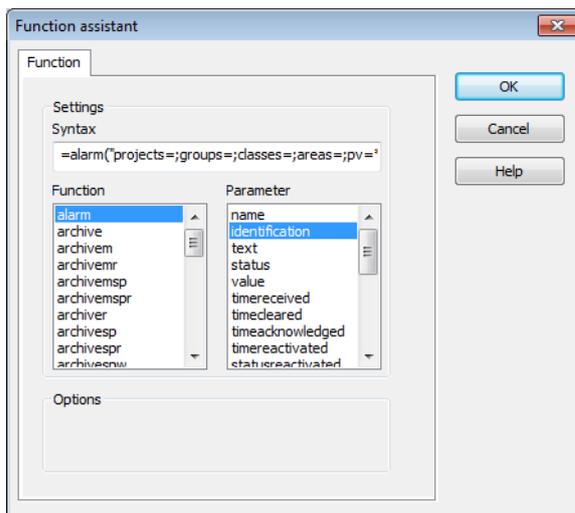
8. The parameter is transferred into the string.

9. Select the desired direction of the entry in the table by double clicking:

   - `bottom`: The function fills the column

- `right`: **The function fills the row**
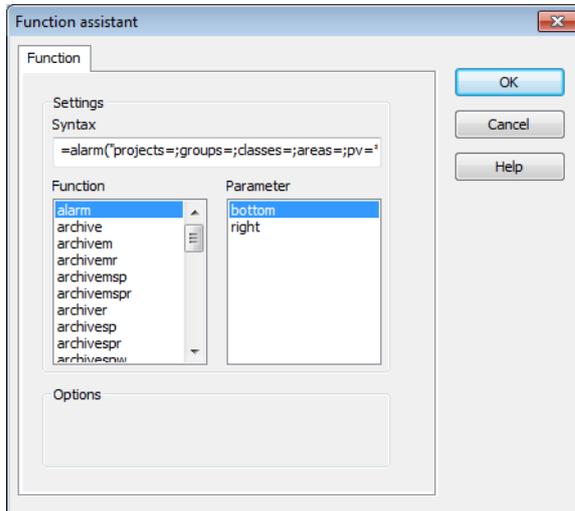
10. The direction is transferred into the string.

11. The brackets are closed.

12. The function is transferred to the cell by clicking on OK.

compare

The `compare(range,"condition")` function compares values in the stated range with a condition and returns the number of hits.

`Syntax: =compare(range,"condition")`

`Area:` cells that are used for the comparison

`Condition:` Content for which the examination is relevant (information always in quotation marks)

> ### Example
>
> *=compare(A1:A10,6>"=5")*
>
> *Compares cells 1 to 10 in column A and gives the number of values that are greater than or equal to 5.*

comparesp

The `comparesp(start cell,direction,P1,P2,P3,P4,"condition")` function compares values in a given area in a special format with a condition and returns the number of the hits from a reference area.

`Syntax: =comparesp(start cell,direction,P1,P2,P3,P4,"condition")`

Parameters	Description
`Start cell`	Cell with the first value.
`Direction`	"right" or "bottom"
`P1`	Number of values in the first row.
`P2`	Empty cells between two value outputs.
`P3`	Empty cells until the next value row.
`P4`	Maximum number of values to compare.
`condition`	condition. Information always in quotation marks.

CONTROL CELLS

The parameters `P1`, `P2` and `P3` set out which cells of the table can be filled:

▶ `P1`: Number

▶ `P2`: Empty cells between two value outputs, depending on `DIRECTION` (`down`, `right`) either empty rows or empty columns

▶ `P3`: Empty cells between two value outputs, depending on `DIRECTION` (`down`, `right`) either empty rows or empty columns

The value, status and time of an archive entry can be read in. All entries of the time range of the log are shown in the log going downwards or rightwards from the cell in which the function was given, with the row format defined by the parameters P1-P3.

WHEN SELECTING `DOWN`, THE FOLLOWING APPLIES：

▶ `P1` = number of rows to be filled with values

- ▶   `P2` = empty rows between the values

- ▶   `P3` = empty columns

WHEN SELECTING `RIGHT`, THE FOLLOWING APPLIES:

- ▶   `P1` = number of columns to be filled

- ▶   `P2` = empty rows between the values

- ▶   `P3` = empty rows between the values

Example

=comparesp(A1,bottom,12,0,0,12,"=5")

comparespsum

The `comparespsum(start cell,direction,P1,P2,P3,P4,"condition")` function compares values in a given area in a special format with a condition and returns the sum of the hits from a reference area.

Syntax: `=comparespsum(start cell,direction,P1,P2,P3,P4,condition)`

Parameters	Description
Start cell	Cell with the first value.
Direction	"right" or "bottom"
P1	Number of values in the first row.
P2	Empty cells between two value outputs.
P3	Empty cells until the next value row.
P4	Maximum number of values to compare.
condition	condition. Information always in quotation marks.

CONTROL CELLS

The parameters `P1`, `P2` and `P3` set out which cells of the table can be filled:

- ▶ `P1`: Number

- ▶ `P2`: Empty cells between two value outputs, depending on `DIRECTION` (`down`, `right`) either empty rows or empty columns

- ▶ `P3`: Empty cells between two value outputs, depending on `DIRECTION` (`down`, `right`) either empty rows or empty columns

The value, status and time of an archive entry can be read in. All entries of the time range of the log are shown in the log going downwards or rightwards from the cell in which the function was given, with the row format defined by the parameters P1-P3.

WHEN SELECTING DOWN, THE FOLLOWING APPLIES:

- ▶ `P1` = number of rows to be filled with values

- ▶ `P2` = empty rows between the values

- ▶ `P3` = empty columns

WHEN SELECTING RIGHT, THE FOLLOWING APPLIES:

- ▶ `P1` = number of columns to be filled

▶ `P2` = empty rows between the values

▶ `P3` = empty rows between the values

> ### Example
>
> `=comparespsum(A1,bottom,12,0,0,12,"=5")`

comparesum

The `comparesum( comparison range,sum reference,"condition")` function compares values in the stated area with a condition and returns the sum of the hits from a reference area.

`Syntax: =comparesum(comparison range,sum reference,"condition")`

▶ `Area:` cells that are used for the comparison

▶ `Sum reference:` value that is given if the condition is met

`Condition`: Content for which the examination is relevant (information always in quotation marks)

> ### Example
>
> `=comparesum(B2:B11,A2,"=5")`
>
> *Cells B2 to B11 are checked for =5. In the lines where this is true, the values of the cell are added to column A.*

computername

The `computername()` function provides the computer names of the computers on which the report is executed.

`Syntax: =computername()`

No transfer parameters are needed.

maxsp

The `maxsp` function determines the highest value in a range und displays it. The format can be a color or underlined.

`Syntax: =maxsp(range,num)`

Transfer parameters	Valid range	Comment
`range`	`A1-ATWH31000`	Maximum number of columns: `31000` Maximum number of rows: `31000`

`num`	`-1`
	
	`0-15`
		`-1` : underlined
		`0 ... 15`: Color according to the font color in the Create new font dialog (screens/fonts). You can find the allocation list under this table.

ALLOCATION OF FONT COLORS

The font colors are taken from the operating system. To make it easier for you, you can find a list of colors as they are used in Windows XP, Vista and 7:
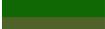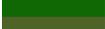
Color	Number
	0
	1
	2
	3
	4
	5
	6
	7
	8
	9
	10
	11
	12
	13
	14
	15

Note: If colors are rearranged or redefined in the operating system, the colors defined for `maxsp` also change.

Example

The highest value is to be displayed as underlined in table A1 to B4.

INPUT IN THE EDITOR：

Row/column	A	B	C
1	8	5	=maxsp(A1:B4,-1)
2	20	2	
3	12	10	
4	32	14	

RESULT IN RUNTIME

```
      A    B
01   08   05
02   12   02
03   12   10
04   32   14
```

minsp

The `minsp(range,num)` functions determine the highest value in a range und displays it. The format can be a color or underlined.

Syntax: `=minsp(range,num)`

Transfer parameters	Valid range	Comment
`range`	`A1-ATWH31000`	

`num`	`-1`
	
	`0-15`
		`-1:` underlined
		`0 ... 15:` Color according to the font color in the Create new font dialog (screens/fonts). You can find the allocation list under this table.

ALLOCATION OF FONT COLORS

The font colors are taken from the operating system. To make it easier for you, you can find a list of colors as they are used in Windows XP, Vista and 7:

Note: If colors are rearranged or redefined in the operating system, the colors defined for `maxsp` also change.

> **Example**
>
> Cell C1=`minsp(A1:B4,-1)`

parameter

Parameters make it possible to define a report that is executed via the zenon `Execute report` function and different transfer parameters used in the process perform different tasks. For example, a single report can be used to display the archives of several controls. Or a report displays values in different units of measurement.Syntax: `=parameter(number)`

Ten parameters are available:

▶   1 - 5: String type

▶   6 - 10: DWord type

CONFIGURATION

1.   Allocate the desired parameters to the appropriate cells in the report
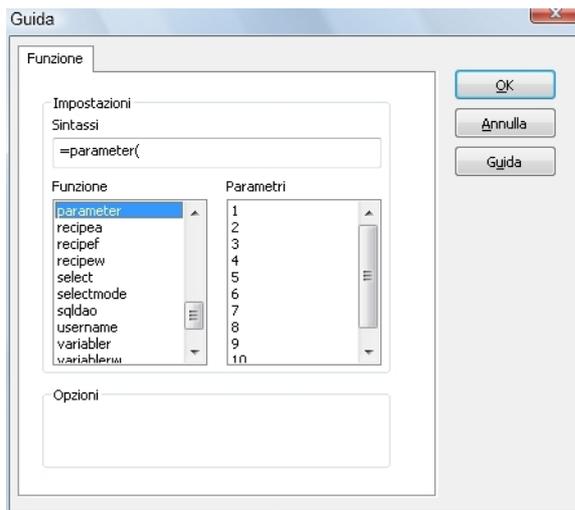
2.   Supplement the parameters with any computer operations that may be required or reference them

3. In Runtime, select values for the parameters via the `Execute report` function or the `Open file` button and the `Parameters` tab
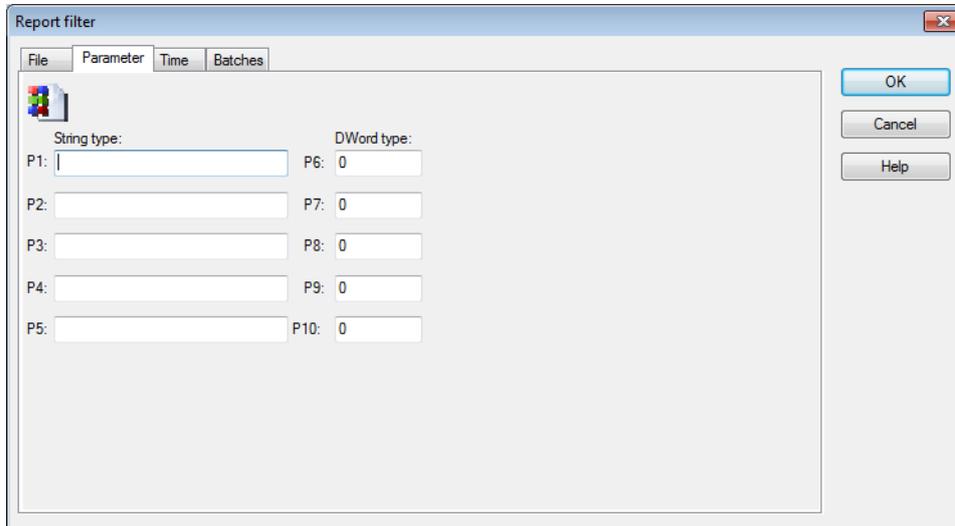
Hint: If you would like to display data from different archives in a report, for example five PLCs, then define a joint report. You control the display via five zenon functions that provide the parameters. You can simply copy these functions, however you must always only change the transfer values in the Parameters tab.

EXAMPLES OF PARAMETERS

▶   `=parameter(1):` allocates parameter (1) to the cell

▶   `=paramter(6)*A1/100:` allocates parameter (6) to the cell, multiplies the value that was set in the `Parameter` tab by the value from cell `A1` and divides this by `100`

select

The `=select(cell address)` function defines the cell that should be selected on opening the report. As a parameter the cell reference has to be stated.

`Syntax: =select(cell address)`

🖼️ **Example**

=select(B3)

selectmode

The `selectmode(selection mode)` function defines the selection mode.

`Syntax: =selectmode(selection mode)`

> **Example**
>
> *=selectmode(3)*

username

The `username` function username returns the user names that are currently logged in to zenon.

`Syntax: =username()`

No transfer parameters are needed.

variabler

The `=variabler(variable name,property)` function provides information on a variable.

`Syntax: =variabler(variable name,property)`

Parameters	Description
name	Variable name.
text	Current limit text in the event of a limit being exceeded.
value	Current value of the variable.
unit	Unit of the variable.
identification	Identification of the variable.
mmin	Minimum measuring range of the variable.
mmax	Maximum measuring range of the variable
status	Current status of the variable.
gw1	Limit 1 of the variable.
gw2	Limit 2 of the variable.
gw3	Limit 3 of the variable.
gw4	Limit 4 of the variable.

Info

The report is created more quickly if the variables called up are advised.

CONFIGURATION WITH THE FUNCTION ASSISTANT

1. Select the `Report functions` report table in the context menu

2. Select `variabler`

3. click on the `Variable` button

4. Select the desired variable:
    Several variables can be selected; these are entered in a downward direction, starting from the cell in which the report function is defined

5. select the property to be displayed

> **Example**
>
> *=variabler("WIZ_VAR_10","name")*

variablerw

The variablerw(`variablename,property,action`) function provides information on variables and writes values in variables.

Syntax: =variablerw(`variable name,property,action`)

Parameters	Description
name	Variable name.
text	Current limit text in the event of a limit being exceeded.
value	Current value of the variable.
unit	Unit of the variable. Note: may not be used together with unit conversion. Only for use with older projects. If the unit defined in the unit conversion of a variable is changed by the report function in the Runtime, you must carry out the configuration in unit conversion again.
identification	Identification of the variable
mmin	Minimum measuring range of the variable. Note: may not be used together with unit switching.
mmax	Maximum measuring range of the variable Note: may not be used together with unit switching.
status	Current status of the variable.
gw1	Limit 1 of the variable.
gw2	Limit 2 of the variable.
gw3	Limit 3 of the variable.
gw4	Limit 4 of the variable.

 Info

The report is created more quickly if the variables called up are advised.

CONFIGURATION WITH THE FUNCTION ASSISTANT

1. Select the `Report functions` report table in the context menu

2. Select `variabler`

3. click on the **`Variable`** button

4. Select the desired variable:
   Several variables can be selected; these are entered in a downward direction, starting from the cell in which the report function is defined

5. select the property to be displayed

6. Enter the desired action

> **Example**
>
> `=variablerw("WIZ_VAR_10","value","read"`

Formatting keys

The formatting keys define the formatted output of cell details depending on the cell contents (data types).

The definition is made:

▶ In the drop-down list *Format -> Cells -> Numbers -> Cell type tab -> General/format string*

or

▶ In the context menu Report table *-> Cell -> Cell type tab -> General/format string*

Numerical data types

Format keys for numerical data types are symbolic dummies that are replaced by the formatted value.

▶ The comma is used as a thousand-divider; a period is used for decimal points.

▶ Each character which does not represent a format key is transferred into the output as a text character.

▶ If no format key is given, the format *0.* is used as a standard setting.

Note: A maximum of 15 decimal digits are permitted with the standard setting.

Format key	Description
, (comma)	Inserts the thousand-separator three digits after the decimal separator.
. (period)	Inserts the decimal separator.
; (semicolon)	Separates the format key for positive and negative value ranges.
#	Is only replaced if the corresponding place is also available.
*	All places before and after the decimal point are displayed.
0 (zero)	If the position in the value does not exist then it is filled with zero.
/	The following character is treated as a text character string.

EXAMPLES

Data	Format key	Formatted text	Description
-0.2	*0.00	-0.20	Two decimal places are always output.
123	*00000	00123	The first 5 places are output with leading zeros.
104	->*<-	->104<-	Numbers are basically output between the ->...<- characters.
0.25 00	*0.*	0.25	For values between -1.0 and 1.0, a leading zero is always output.
12	+*0.00;-*0.00	+12.00	Positive values are indicated with + and negative values with -.

Logical data types

Yes/No information can be displayed in different ways with logical data types. The output text is determined using the formatting key.

► Format: `Text1;Text2`

► Output:

• Positive values including zero = `Text 1`

• Negative values = `Text 2`

Output information can be described independently of the cell formatting with the `if (on page 94)(exprTest,valTrue,valFalse)` function.

EXAMPLE

Data	Format key	Formatted text	Description
`1`		Yes	If no formatting key is given, true/false is used as a default.
`-1`	Yes;No	No	

Data types for date and time

A variety of formatting keys are available for the output of date and time details. In input mode and for output without formatting keys, the output format defined under the windows country settings is used.

> 💡 **Info**
>
> *Descriptions for days and months can be configured as language switchable in Runtime.*

Each character which does not represent a format key is transferred into the output as a text character.

Format key	Value	Description
m	1 - 12	Month numerical short
mm	01 - 12	Month numerical long
mmm	Jan - Dec	Month short
mmmm	January - December	Month long
d	1 - 31	Day short
dd	01 - 31	Day long
ddd	Sun - Sat	Weekday short
dddd	Sunday - Saturday	Weekday long
yy	00 - 99	Year short
yyyy	1700 - 2900	Year long (0x is interpreted as 190x)
/		International date separator
h	0 - 23 or 1 - 12	Hour short, either 24 h or 12 h with ap
hh	0 - 23 or 01 - 12	Hour long, either 24 h or 12 h with ap
mm	00-59	Minutes, only after h or hh
ss	00-59	Seconds
ff	000-999	Milliseconds
ap	am, pm (or international equivalent)	am or pm
: (colon)		International time separator

EXAMPLE

Data	Format key	Formatted text
22.09.95 14:05	dddd dd/mm/yyyy	Friday 22.09.1995
22.09.95 14:05	dd/mmmm yyyy	22 September 1995
22.09.95 14:05	dd/mm/yy hh:mm	22.09.95

		14:05
22.09.95 14:05	hh:mm:ff	14:05:000

3.1.9   zenon functions for report generator

zenon provides functions to control the report generator in Runtime.

The screen switching (on page 164) function switches to the report screen and makes it possible to define filters.

Special functions make it possible to print, export, and execute a report in Runtime. To create zenon functions for the report generator:

1. Navigate in the functions context menu to: *New function -> Report generator*

2. select the desired function:

    - Print report (on page 185)

    - Export report (on page 186)

    - Execute report (on page 189)

Report screen switching

To use a report in Runtime, configure screen switching to a report screen type:

1. Create a new `Report` screen type

2. Select the `New function` command in the `Functions` node

3. select the function `Screen switching` function

4. select the `Report screen`

5. The report filter (on page 165) is displayed

6. select the desired properties

7. Link the functions with a button in the screen, in order to switch in Runtime

Report filter

Displayed when creating zenon functions for configuring reports:

- ▶ File (on page 165): Selection of report file and conditions in Runtime

- ▶ Parameter (on page 166): Transfer parameters

- ▶ Time (on page 167): Time range

- ▶ Batches (on page 176): Filtering on batches

- ▶ Links (on page 178): Replace functions

- ▶ Indices (on page 182): Replace indices

File

In the File tab, you define which report is called up with screen switching or if this is to be decided in Runtime.
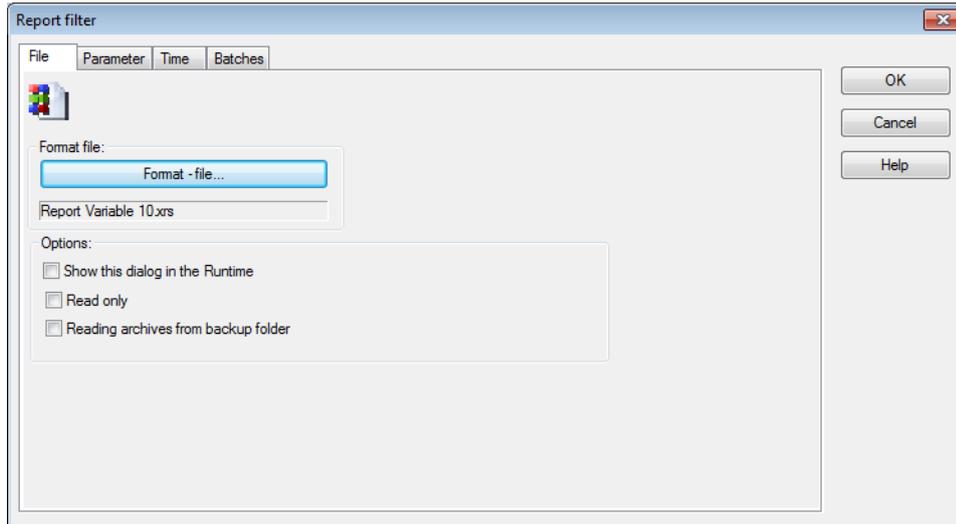
Parameters	Description
`Format file`	selection of the report file configured (on page 10) in the editor (*.xrs)
`Show this dialog in the Runtime`	`Active:` This filter dialog is offered before being called up in Runtime.
`Read only`	`Active:` Entries can only be read. It is not possible to change or save them.
`Read archives from backup folder`	`Active:` Stored archives are read in from the backup folder defined in project configuration.

The file configuration is different for:

- ▶ Print report (on page 185)

- ▶ Export report (on page 186)

- ▶ Execute report (on page 189)

Parameters

Parameters make it possible to use a report for different archives. To do this, the fields to be used flexibly are defined with the `parameter(Number)` (on page 153) report function. The parameters to be used are set in the Parameters tab of the screen switching function. For example, a single report can be used to display the archives of several controls. Or a report displays values in different units of measurement.
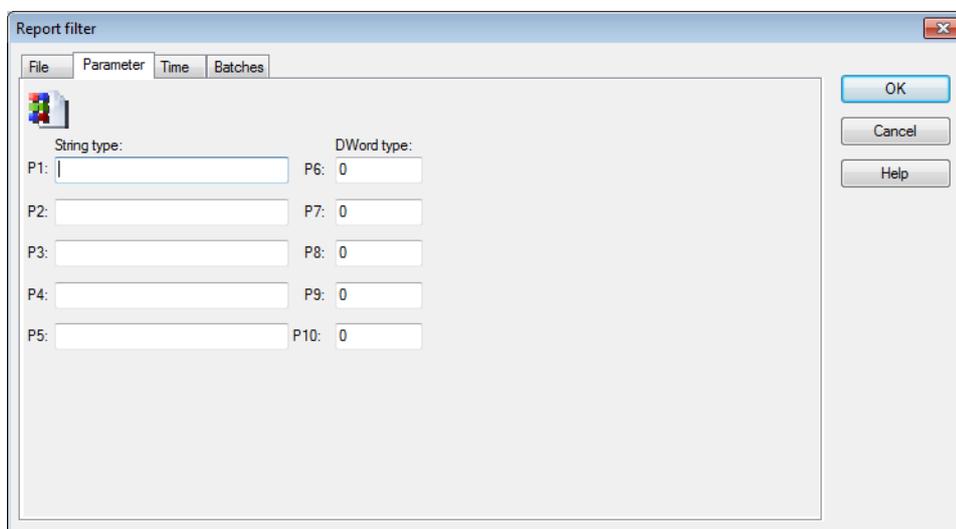
Ten parameters are available:

▶ 1 - 5: String type

▶ 6 - 10: DWord type

If, for example, the archive data of five PLCs is to be displayed, a report is created and the five parameters for the PLCs are transferred via the five switching functions.

Time

The time filters provide an easy possibility to limit the data to be displayed or exported. The time filters are very flexible to implement and can be pre-set in the editor or adjusted in Runtime.

Note: Time is saved in UTC. For details see chapter Handling of date and time in chapter Runtime.

The mechanisms described here are applicable for screen switching as well as export for:

▶ Alarm Message List

▶ Chronological Event List

▶ Archive revision

▶ Extended Trend

▶ Report Generator (on page 6)
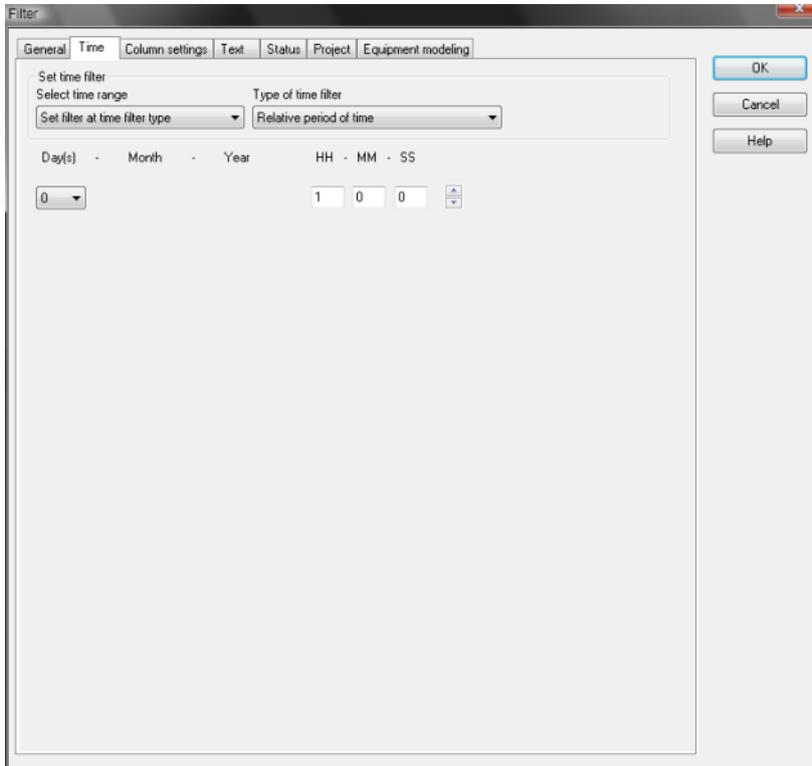
Time filtering can be carried out in two ways:

1. Define time period in the Editor
   Fixed time areas are used. A time period is given in the editor. It is only possible to filter according to this time period in Runtime. Other filters - such as filtering according to variable name, alarm/event groups and alarm/event classes etc. can no longer be amended in Runtime.

2. Time filter amendable in Runtime
   Pre-defined times are used. The time filter is defined in the Editor and can be changed in Runtime as desired.

   > 💡 **Info**
   >
   > *If a screen that does not have a time filter is referenced, the time filters are deactivated.*
   >
   > *For the setting* `Set filter for time filter type` *the option* `Display filter dialog in Runtime` *can be selected, but it is not available in runtime.*

Time filter can be modified as you will in the Runtime

With this method, you stipulate a time filter in the Editor. This can be amended in Runtime before execution. To create the filter:

1. create a picture switch function.

2. The screen must have `Filter` and `Display filter` buttons

3. Select, from the `Time period selection` drop-down list, the `Set filter for time filter type` setting

   Note: The option `Display filter dialog in Runtime` can be selected, but is not available in Runtime to set the `Set filter for time filter type` option.

4. Select the time period from the `Time filter type` drop-down list

Parameters	Options	Description
`Absolute period of time`		A fixed period of time is entered in the editor. When the function is executed, the defined absolute time period is exactly used. Example: If you want to see all alarms from January 1, 2011 to December 31, 2011, then you must enter the corresponding data at `From` and `To`. **Note:** Time is saved in UTC. For details see chapter Handling of date and time in chapter Runtime.
	`From`	Defines the start time in day, month, year, hour (HH), minute (MM), second (SS)
	`To`	Defines the end time in day, month, year, hour (HH), minute (MM), second (SS)
`Relative time period`		A relative time period is entered. **Attention!** this filter is constantly updated. It is therefore carried over. For example: You set up a relative time of `10` minutes and switch to an Alarm Message List Screen with this time filter at 12:00. Then you are shown the alarms from 11:50 to 12:00 when switching. If the Alarm Message List screen stays open, the filter is automatically updated. At 12:01, you see the alarms from 11:51-12:01 etc.
	`Time`	Defines the relative period in days, hours (HH), minutes (MM) and seconds (SS)
`from HH:MM:SS o' clock`		A time from which the filter is effective is stated. If the time is not reached on the current day, filtering takes place from the corresponding time the previous day. Example: You enter `23:00:00`. If it is then 23:30 when executing the function, then it is filtered from 23:00:00 up to the current point in time. If it is 22:30 however, then filtering takes place from 23:00:00 on the previous day to the current point in time. **Attention!** The start point of this filter is not updated automatically. Only the existing times are used when shown, even if the screen remains open and 23:00:00 is reached. The end time point is not defined with this filter, it is carried over.
	`Time`	Defines the start time period in hours (HH), minutes (MM) and seconds (SS)

Parameters	Options	Description
From day – HH:MM:SS time		A day and time for the start of the filter are entered. If the time given has not been reached in the current month, the corresponding time from the previous month is used.
		Example: You enter `day 5 - 23:00:00`. If it is the 10th of the month at 23:30, then filtering takes place from the 5th of the month from 23:00:00 to the current time point. If, however, it is the 4th of the month, then filtering takes place from the 5th of the previous month to the current time point.
		Attention! The start time of this filter is not updated automatically. Only the existing times are used when shown, even if the screen remains open and 23:00:00 is reached. The end time point is not defined with this filter, it is carried over.
	Time	Defines the start point in hours (HH), minutes (MM), seconds (SS)
From day, month – HH:MM:SS time		A month, day and time for the start of the filter are entered. If the time stated has not been reached in the current year, the corresponding time from the previous year is used.
		Example: You enter `Month October,Day5- 23:00:00`. If it is October 10th at 23:30, then filtering takes place from October 5th from 23:00:00 to the current time point. If, however, it is only October 4th, then filtering takes place from the 5th of the previous year to the current time point.
		Attention! The start point of this filter is not updated automatically. Only the existing times are used when shown, even if the screen remains open and 23:00 on October 5 is reached. The end time point is not defined with this filter, it is carried over.
	Time	Defines the start point in months, days, hours (HH), minutes (MM), and seconds (SS)
No time filter		No time filter is used.
		At the Report Viewer (on page 214) and in the Archive revision all entries since 1.1.2000 are displayed with this setting.

⚠ **Attention**

If a different `time period` than `Set filter for time filter type` is selected for the Alarm Message List or the Chronological Alarm List, the time of the screen to be called up cannot be transferred over in Runtime.

Specify time area in the Editor

With this method, you enter a fixed time period into the editor, which is applied when the function is carried out in Runtime. You can then only define the start time in Runtime, but no further filter settings.

For example: You set a 30 minute time filter. In Runtime, you can now only set when this 30 minute time period is to start. However, you cannot change the filter to a day filter.

> ⚠ **Attention**
>
> *When using this type of filter, you can also no longer amend all other filters in Runtime that are available in the* `General` *tab. It is still possible to filter for text, status and equipment.*

To create the filter:

1. create a picture switch function.

2. The screen must have the `Filter` button to start the filter in Runtime

   Hint: Activate the `Display filter dialog in Runtime` option in the `General` tab. This way you can amend the start time before the function is carried out. Do not have the filter displayed in Runtime when the function is turned on; this way the current time period is always used. If you have activated the `Use last closed time period` option, the previous time period is shown.
   For example: You have set a 30 minute filter. It is 10.45 when the function is activated. If the `Use last closed time period` option is deactivated, the filter is set to the current time period 10:30:00 to 10:59:59. If the option is activated, the filter is set to the previous time period of 10:00:00 to 10:29:59.

3. Select `Time period selection` from the drop-down list;
   Exception: `Set filter for time filter type` - this is envisaged for the Time filter can be adapted in Runtime option

4. Configure the selected time period

Parameters	Description
Set filter for filter type	Activates the possibility to Amend Runtime.
One day	A whole day is filtered: 00:00:00 to 23:59:59.
One week	A whole week is filtered: Monday 00:00:00 to Sunday 23:59:59.
Two weeks	Two whole weeks are filtered: Monday 00:00:00 of the first week to Sunday 23:59:59 of the second week.
One month	A whole month is filtered: The first of the month at 00:00:00 to the last day of the month at 23:59:59.
One year	A whole year is filtered: January 1 of the year at 00:00:00 to December 31 of the year at 23:59:59.
15 minutes	A 15 minute time period is filtered.
30 minutes	A 30 minute time period is filtered.
60 minutes	A 60 minute time period is filtered.
Relative time period	For these filters, you must set the Display filter dialog in Runtime option on the **General** tab to active. if the option is not set, the time period in Runtime is set to 0. When carrying out the function, a dialog is displayed with which a relative time period in days, hours (HH), minutes (MM) and seconds (SS) can be given. **Attention!** the selected time period relates to the activation time. The filtered time period is then fixed. Example: It is 10:23:00 when calling the function. You set a relative time period of an hour. The filter is set to 9:23:00 - 10:23:00.
Absolute period of time	For these filters, you must set the Display filter dialog in Runtime option on the **General** tab to active. if the option is not set, the time period in Runtime is set to 0. When executing the function, a dialog is displayed with which an absolute period of time can be entered. In doing so, you set the start and end time in day, month, year, hour (HH), minute (MM), second (SS). **Note:** Time is saved in UTC. For details see chapter Handling of date and time in chapter Runtime.
Batches	For these filters, you must set the Display filter dialog in

	`Runtime` option on the **General** tab to active. if the option is not set, the time period in Runtime is set to 0. When carrying out the function, a dialog is displayed with which you can select a batch. The time filter is then set to the time period of the batch. You can therefore filter according to all alarms or all CEL entries which occur in a batch, for example. **Note**: The filter only displays closed batches.
`Relative batches`	**Extended Trend** only. Display always starts from zero point. Enables several batches to be compared directly.

OPTIONS

The following settings are possible to adapt the time period for all time periods (except absolute and relative time period, as well as batches):

Parameters	Options	Description
`Use last concluded time period`		`Active`: The current time period is always used. The previous time period is displayed if the option is set. Example: You have set a 30 minute filter. It is 10.45 when the function is activated. If the option is deactivated, the filter is set to the current time period of 10:30:00 AM to 10:59:59 AM. If the option is activated, the filter is set to the previous time period of 10:00:00 to 10:29:59.
`Move time period back by`		`Active:` The whole time period is moved back by the given time. The time period remains unaffected by this setting. Example: You have selected a 60 minute filter and enter a time of 5 minutes here. In Runtime, the filter is now now started at each whole hour, but always five minutes later, for example at 10:05:00, at 11:05:00 etc. Filtering takes place in 60 minute intervals as before, i.e. to 11:04:59, to 12:04:59 etc.
	`Time`	Defines the time difference in hours (HH), minutes (MM) and seconds (SS)

Change time period by		`Active:` The filtered time period is shortened or extended. Example: You have selected a 60 minute filter and enter a time of 5 minutes here. The `Add time stated` option is set. In Runtime Filters are no longer carried out for 60 minutes but for 65 minutes, i.e. from 10:00:00 to 11:04:59.
	`Time`	Defines the time by which the time period is shortened or extended. in hours (HH), minutes (MM) and seconds (SS)
	`Add time stated`	The selected time period is extended by the time stated
	`Deduct time stated`	The selected time period is shortened by the time stated

⚠️ **Attention**

If a different `time period` *than* `Set filter for time filter type` *is selected for the Alarm Message List or the Chronological Alarm List, the time of the screen to be called up cannot be transferred over in Runtime.*

Batches

Batch filtering corresponds to filtering in the archives:

Parameters	Description
Batch filter	On the left side you can choose the desired archive from the available archives. On the right side the available batches are displayed. You can filter the batches there.
Lot name	In this column the names of the available batches are displayed. By left clicking the top part of the header, the batches are sorted alphabetically in an ascending or descending order. In the bottom part of the header you can enter a character string. Only batches matching the respective character string will be displayed.
Start date	In this column the start date of the available batches is displayed. By left clicking the top part of the header, the batches are sorted in an ascending or descending order. Batches with the same start date are sorted according to their start time. In the bottom part of the header you can enter a start date manually or use the displayed calendar.
Start time	Only available if you entered a start date. In this column the start time of the available batches is displayed. By left clicking the top part of the header, the batches are sorted in an ascending or descending order. In the bottom part of the header you can enter the start time manually. Note: '*' means 0:00:00 o' clock.
End date	In this column the end date of the available batches is displayed. By left clicking the top part of the header, the batches are sorted in an ascending or descending order. Batches with the same end date are sorted according to their end time. In the bottom part of the header you can enter an end date manually or use the displayed calendar.
End time	Only available if you entered an end date. In this column the end time of the available batches is displayed. By left clicking the top part of the header, the batches are sorted in an ascending or descending order. In the bottom part of the header you can enter the end time manually. Note: '*' means 11:59:59 PM o' clock.
Duration	This column displays the duration for each available batch. It is only for display.

Info

Still open batches are also displayed if they match the set filter criteria.

Info

The value of the batch variable is written in the index file and in the header of the ARX file at the start of the batch. These entries are adjusted with every change of the variable. When the batch is closed, the value of the batch variable at this moment is finally written in the index file and in the header.

Thus the batch name is final when the batch is closed.

Replacing links

Linked variables or functions can be permanently replaced in the Editor in a rule-based manner. The replacement is not limited to the functions or variables defined in the properties of the `Variable / function` group. All other variables used the screen can, such as variables for `Visibility`, `Flashing`, `Size and rotation dynamic` and others. Rule-based replacement also works for several elements in a screen at the same time and for elements with embedded symbols.

To replace variables or functions in a rule-based manner in the element directly:

1. Highlight the desired element and open it by right clicking the element in the context menu

2. select **Replace Links**

3. The dialog for the replacement of links opens
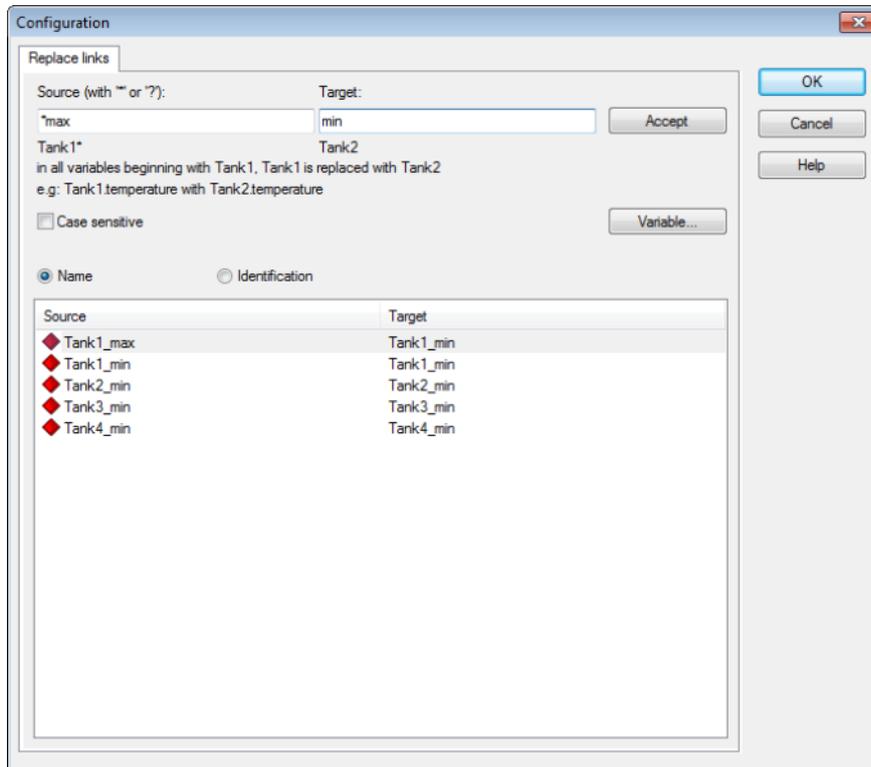
Property	Description
Source	Enter the partial string to be searched for. Place holder * and ? can be used. Placeholders are only permitted as prefix or suffix; e.g. *xxx or xxx*.
Target	Entry of the partial string
Name	Swaps information in process variable names.
Identification	Exchanges information in the identification
Note capitalization	When swapping, be sure that any capitalization is an exact match.
Accept	Swaps target strings from the `source` for those defined in the `target`.
Variable/function	Opens the selection list for variables/functions in relation to the selected line in the list. Clicking on the variable in the list defines new target variables. Alternative: Double-click on the source variable in question.

REPLACE

A) REPLACE BY MANUAL SELECTION

▶ select the element from the list that you would like as the source

▶ select a target element via the `variable/function` button

▶ the previous element is replaced by the new one

B) AUTOMATED REPLACEMENT WITH RULES

▶ In the source input field, define the parameters for the elements that you wish to replace

▶ define the parameters for the new variables/functions in the target input field

▶ specify what is to be replaced via Name/Identification

▶ Click on `Accept`

Several entries are separated by a semicolon (;).

Example:

```
Source:A;b
```

```
Target:C;d
```

> 💡 **Info**
>
> *The target variable or target function can also be in a different project as the source variable or source function.*

> ⚠ **Attention**
>
> *When replacing variables, be aware of the type and signal resolution. If you replace a variable with one of an incompatible type, this can lead to errors during execution. You will be warned when making the substitution; the substitution will however be carried out.*

> 💡 **Info**
>
> *Replace via Drag&Drop: Elements that can only contain one variable or function can be swapped by dragging & dropping. Drag the new variable or function to the element using the mouse. It automatically replaces the previous one.*

EXAMPLE

All variables with "Tank1" in the name are to be replaced with "Tank4".

1. Enter "Tank1" into the `Source` text field. You can include all variables that start with "Tank1" with the `*` character.

2. Enter "Tank4" into the target text field.

3. Click on Accept.

4. The variables Tank1_max and Tank1_min are replaced by the variables "Tank4_max and Tank4_min.

Replace indices

When switching screens in Runtime, variables can be replaced dynamically (virtually) using indexing rules. In doing so, variable values can be used as part of the replacement rule.

⚠ **Attention**

Because the names are replaced, variables and functions must always be given different names. Variables and functions are replaced if the names are the same.

Example:

If there is a variable "test" and a function "test" and the variable "test" is to be replaced, the function "test" would be replaced at the same time.

Tip: Name variables and functions with a respective prefix (such as var_ and func_). You therefore always have a overview in lists too.
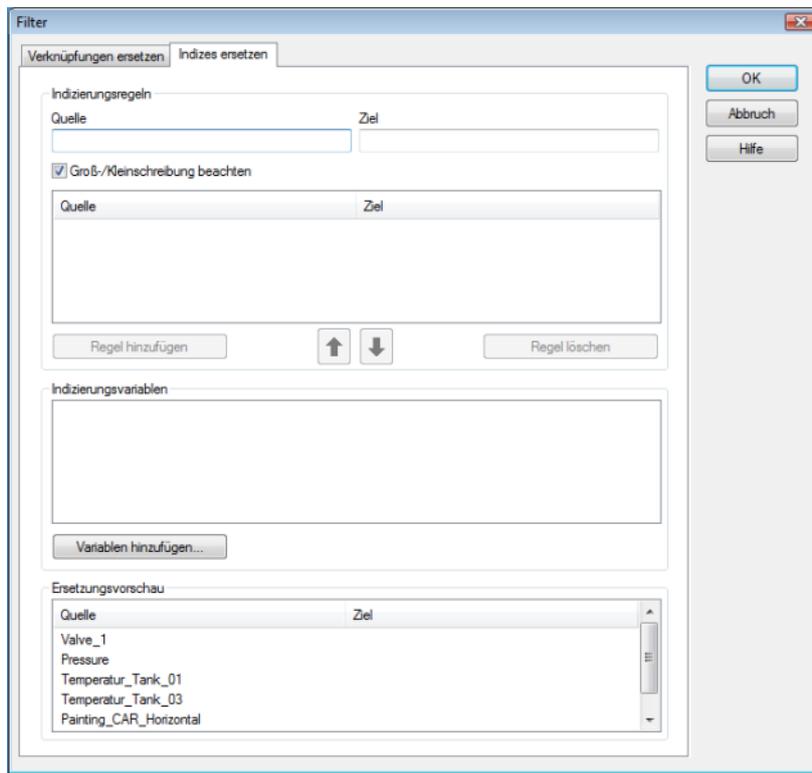
To configure the replacement of indices:

1. All variables and functions used in the screen are displayed in the `Source` field of the **`Replacement preview`**.

2. Left-click the variable or function for which you want to create an indexing rule.

3. In the **`Indexing rules`** area, the variable/function is entered in the fields **`Source`** and **`Target`**. Alternatively, you can also enter the desired variable/function directly into the **`Source`** field in the **`Indexing rules`**.

4. Define the indexing rule.
   The dynamic part of the rule is inserted in curly brackets, for example {X01}.

Example:

The source variable `Motors1.speed` is replaced by the target `Motors{X01}.speed`, the placeholder `X01` is replaced with the respective value of the index variable in the Runtime.
The replacement rule would thus be:

Source: `Motor1`

Target: `Motor{X01}`

5. Carry this over to `Add Rule` by left clicking in the indexing rules.

6. All indexing rules that have been created are displayed in this list.

7. You can change the order of the entries using the button with the arrow symbol.

   Attention: The substitutions are executed in the order in which they are displayed in the list!

8. Variables that are not present on the screen but are used in the replacement can be inserted via the `Add Variables...` button in the `Indexing Variables` window.

9. The variables which are in this list are always kept in the memory in order to minimize waiting time when the screen is loaded.

> 💡 **Info**
>
> *For the placeholder * the same rules as for Replace links are valid.*

> ⚠ **Attention**
>
> If index variables are used in a network project, the Client must first fetch the value from the Server. Then the calculation is executed. On devices with outdated hardware this may cause delay at screen switching.
> Therefore in a network project index variables should always be set to `DDE active`.

CROSS-PROJECT

If the target is in a different project to the source, this can be displayed using '#'.

> **Example**
>
> *Source: VAR_1*
>
> *Target: VAR_1 in Project_1*
>
> *You thus enter VAR_1 as the source and Project_1#VAR_1 as the target.*
>
> *Source: SUBPROJECT1#VAR_1*
>
> *Target: {X01}VAR_1*
>
> *The index variable X01 is a string variable and has the project name including # as a value, for example "SUBPROJECT4#".*

Print report

The `Print report` function makes it possible to print out a report without calling it up beforehand. To configure the function:

1. Create a new function

2. Select, in the `Report generator` section, the `Execute report` entry

3. The dialog for the report filter (on page 165) is opened

4. **Hint:** The `File` tab settings are different. The settings that are valid for `Print report` are clarified in this chapter; you can find all further settings in the report filter (on page 165) chapter.

5. select the desired properties

6. Link the functions with a button in the screen, in order to switch in Runtime

FILE TAB

Parameters	Description
Format file	selection of the report file configured (on page 10) in the editor (*.xrs)
Show this dialog in the Runtime	`Active:` This filter dialog is offered before being called up in Runtime.
With cancel dialog	`Active:` a button to cancel printing is shown whilst the report is being prepared for print.
Read archives from backup folder	`Active:` Stored archives are read in from the backup folder defined in project configuration.

> ### 💡 Info
>
> *Archive entries can only be read with the* **Print report** *function. Write actions are not possible.*

Export report

This function is used to write a table into a file in the background during online operation. The export file is stored in the \Export sub-folder of the project folder. (open from zenon: highlight project, press `ctrl+alt+D`.) Provide the file parameters, the filter and the time as the transfer parameters.

The `Export report` function makes it possible to export a report without calling it up beforehand. To configure the function:

1. Create a new function

2. Select, in the `Report generator` section, the `Export report` entry

3. The dialog for the report filter (on page 165) is opened

4. Hint: The `File` tab settings are different. The settings that are valid for `Export report` are clarified in this chapter; you can find all further settings in the report filter (on page 165) chapter.

5. select the desired properties

6. Link the functions with a button in the screen, in order to switch in Runtime

> Info
>
> *When exporting a report using the `Export report` function, the converted units are exported at units of measurement. The basic units are not a part of the export.*

FILE TAB

Parameters	Description
Format file	selection of the report file configured (on page 10) in the editor (*.xrs)
Show this dialog in the Runtime	Active: This filter dialog is offered before being called up in Runtime.
Read archives from backup folder	Active: Stored archives are read in from the backup folder defined in project configuration.
Export file	Properties of the file that is exported in the report.
Name	Name of the file. Maximum of eight alphanumeric characters. If no name is given and automatic naming is switched off, the name dummy is used. Attention: The use of special characters may lead to a loss of data.
Format	Format of the export file.
▸ TXT	Text file
▸ XRS	zenon Report file
▸ XML	XML-File
Automatic assignment	Active: The filename is created automatically. It consists of the short identifier and the coded date and the chosen file ending.

CODING NAME FOR AUTOMATIC NAMING

Code	Meaning
xx	Short identifier: 2 characters The first two characters of the name given under 'Name' or the name issued by the system, du.
JJ	Year: two digits
MM	Month: two digits
TT	Day: two digits
HH	Hour: two digits
MM	Minute: two digits
SS	Second: two digits

EXAMPLE

▶ The `Report` report file is exported on 9. 6. 2010 at 11:52 in XRS format

▶ Automatic filename: `re100506091152.xrs`

> ### 💡 Info
>
> *Archive entries can only be read with the `Export report` function. Write actions are not possible.*

Execute report

The `execute report` function makes it possible to execute a report automatically in the background in Runtime. If writing functions are configured in the report, calculated values can automatically be written to archives or to the process image.

To configure the function:

1. Create a new function

2. Select, in the `Report generator` section, the `Execute report` entry

3. The dialog for the report filter (on page 165) is opened

   Hint: The `File` tab settings are different. The settings that are valid for `Execute report` are clarified in this chapter; you can find all further settings in the report filter (on page 165) chapter.

4. select the desired properties

5. Link the functions with a button in the screen, in order to switch in Runtime

FILE TAB

Parameters	Description
Format file	selection of the report file configured (on page 10) in the editor (*.xrs)
Show this dialog in the Runtime	Active: This filter dialog is offered before being called up in Runtime.
With cancel dialog	Active: a button to cancel printing is shown whilst the report is being prepared for print.
Read archives from backup folder	Active: Stored archives are read in from the backup folder defined in project configuration.

💡 Info

With this function, all write actions are carried out in the report.

3.2 Operating during Runtime

The reports on the configured buttons are displayed, configured and administered in Runtime. The following user elements are available in Runtime:

Parameters	Description
Add template...	Opens the dialog for selecting a template for the screen type.
	Templates are shipped together with zenon and can also be created by the user.
	Templates add pre-defined control elements to pre-defined locations in the screen. Elements that are not necessary can also be removed individually once they have been created. Additional elements are selected from the drop-down list and dragged onto the screen. Elements can be moved on the screen and arranged individually.
Buttons	Buttons for controlling in the Runtime.
Refresh	Recreate build report completely (values and display).
Print	Print report in Runtime.
File	Select report file and change filter conditions.

Parameters	Description
Export	Export report.
Save archive	Write changed values to the archive.
Save report	Report is saved in Runtime (*.xrs).
	Attention: When saving, formulas and functions are replaced by the current values. The functions in these reports (.xrs files) are no longer available. These reports can also not be edited in the editor anymore.
	Recommendation: Set original report files to "write-protected" status and save changes with **Save as** in a new file.
Save report as	Report is saved under a new name during Runtime (*.xrs), formulas and functions are replaced by current values.
Read all values	All values are read in again from the driver.
Read selected values	Selected values are read in again from the driver.
Write all values	All values are sent to the driver.
Write selected values	Selected values are sent to the driver.
Line up	Scroll text in list element up
Line down	Scroll text in list element down
Column right	Scroll text in list element to the right
Column left	Scroll text in list element to the left
Page up	Scroll text in list element up
Page down	Scroll text in list element down
Page right	Scroll text in list element to the right
Page left	Scroll text in list element to the left
Report	Control elements for report display.

Parameters	Description
`Report`	Report window to display the report.
<u>Filter profiles</u>	Buttons for filter profiles.
`Profile selection`	Display of the available filter profile and input for new profile names.
`Save`	Saves current setting as a profile.
`X (save)`	Deletes selected profile.
`Import`	Exports profiles.
`Export`	Imports profiles.

💡 Info

A period is used as a decimal separator. When inputting, both a period or a comma can be used. The separator is automatically converted by the system.

Example: `10,5` is turned into `10.5`

3.2.1   Filter for screen switch

If you switch to a report type screen, the following filter is displayed.

Parameters	Description
Batch filter	On the left side you can choose the desired archive from the available archives. On the right side the available batches are displayed. You can filter the batches there.
Lot name	In this column the names of the available batches are displayed. By left clicking the top part of the header, the batches are sorted alphabetically in an ascending or descending order. In the bottom part of the header you can enter a character string. Only batches matching the respective character string will be displayed.
Start date	In this column the start date of the available batches is displayed. By left clicking the top part of the header, the batches are sorted in an ascending or descending order. Batches with the same start date are sorted according to their start time. In the bottom part of the header you can enter a start date manually or use the displayed calendar.
Start time	Only available if you entered a start date. In this column the start time of the available batches is displayed. By left clicking the top part of the header, the batches are sorted in an ascending or descending order. In the bottom part of the header you can enter the start time manually. Note: '*' means 0:00:00 o' clock.
End date	In this column the end date of the available batches is displayed. By left clicking the top part of the header, the batches are sorted in an ascending or descending order. Batches with the same end date are sorted according to their end time. In the bottom part of the header you can enter an end date manually or use the displayed calendar.
End time	Only available if you entered an end date. In this column the end time of the available batches is displayed. By left clicking the top part of the header, the batches are sorted in an ascending or descending order. In the bottom part of the header you can enter the end time manually. Note: '*' means 11:59:59 PM o' clock.
Duration	This column displays the duration for each available batch. It is only for display.

> 💡 **Info**
>
> *Still open batches are also displayed if they match the set filter criteria.*

> 💡 **Info**
>
> *The value of the batch variable is written in the index file and in the header of the ARX file at the start of the batch. These entries are adjusted with every change of the variable. When the batch is closed, the value of the batch variable at this moment is finally written in the index file and in the header.*
>
> *Thus the batch name is final when the batch is closed.*

3.3 Example of archive data in reports

In this example, you create archive data in a report. To do this:

1. Open a new report using the context menu and the `New report` entry.

2. Design the Title area (on page 196)

3. Fill the data area (on page 201)

4. Set up a report screen (on page 207) and create a function

3.3.1 Title area

The title area contains, in the first four rows:

▶ Texts (on page 197) as headings for the data

▶ Date (on page 197) and time

▶ Company logo (on page 198)

The title area is formatted in such a way that it is fixed for longer reports and does not disappear from view during scrolling.

Texts

Enter the following texts as headings for the report:

- ▶ Time

- ▶ Hall 1

  - Tank 1
  - Tank 2

- ▶ Hall 2

  - Tank 1
  - Tank 2
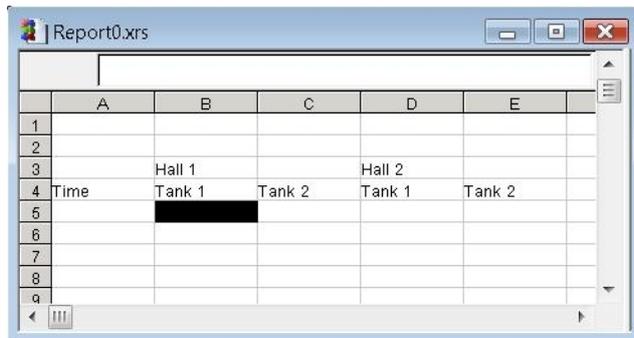
Date and time

The date and time of the report are configured for the for cells A1 and E1 with report functions. To do this:

1. Enter the function =now() into the cells A1 and E1.

2. The current date is now displayed in both cells.

3. In order to get the time in cell E1 instead, you have to change the formatting of this cell.

   a) Highlight cell E1
   b) Open the format dialog using the Format/Cells/Editing menu.

c) Select the formatting string as shown in the illustration below

4. The result:

Screens as a company logo

Screens in cells are not treated as cell contents by the Report Generator, but as formatting. Configuration is therefore carried out using cell formatting:

1. Import the desired screen in the Project Manager into the `Files/graphics` nodes

2. Select cell E2 with the left mouse button.

3. Open the context menu of the report with the right mouse button and select the `Cell` command.

4. In the dialog box, change to the `Cell type` page.

5. Change the cell type to `Graphics file`

6. Select the desired screen from the drop-down list on the right side

7. The result:

Fixed title area for multipage reports

To always have the row and column titles in view in multi-page reports, you must prevent these also being moved when the report is scrolled. To do this, define fixed title areas.

To fix the row title:

1. Select the `Report` command in the Format menu.

2. The configuration dialog is opened

3. Open the `Columns` tab
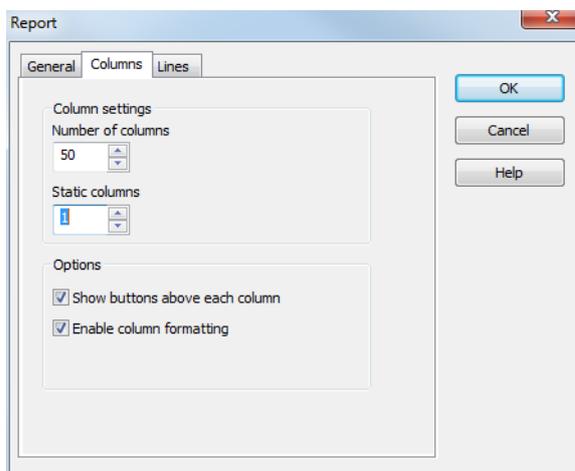
Next to the number of columns you also can define the number of `Non-scrolling columns` here. The stipulated number of fixed columns is then no longer scrolled out of the visible screen area.

4. Enter `1` for the `Non-scrolling columns`.

5. Change to the `Rows:` tab
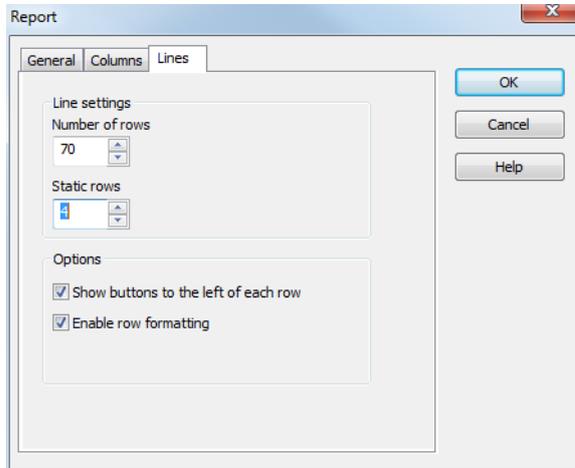
Our column headings are in rows 1 to 4 for the lines:

6. Enter `4` for the `Non-scrolling rows`.

7. Confirm all changes by clicking on the `OK` button

3.3.2 Data area

In the data area:

▶ Configure the display of time (on page 201) and values (on page 205)

▶ Have the sum and average (on page 206) displayed

Time

The time of the displayed archive values is displayed in column `A`.

> 💡 **Info**
>
> *We select a cyclical archive as a basis. The entries of all variables are therefore made at the same time and a time column is sufficient for all entries.*

To configure the time:

1. Select cell `A5`.

2. Select the `Function assistant` command in the `Format` menu

3. The function assistant is opened
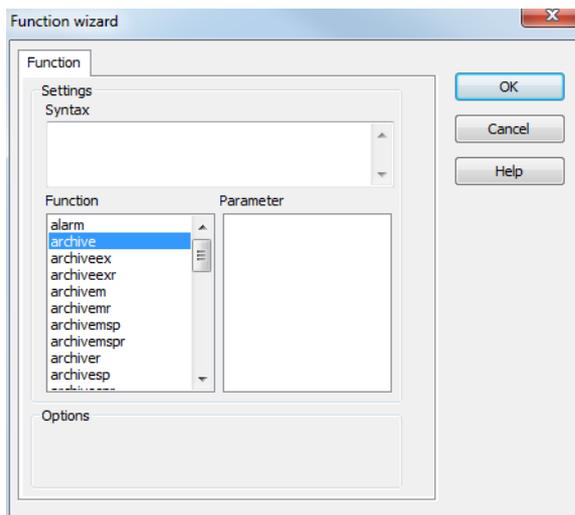
4. Select the `Archive` function with a double-click.

5. The `Archive` button is always displayed in the lower area of the dialog
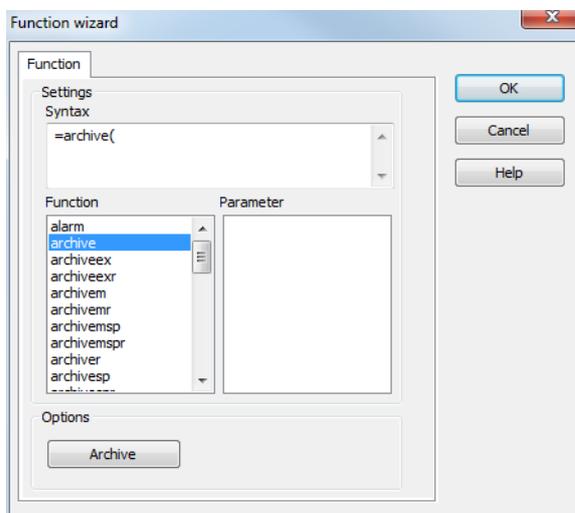
6. Click on `Archive`

7. The dialog for archive selection is opened:
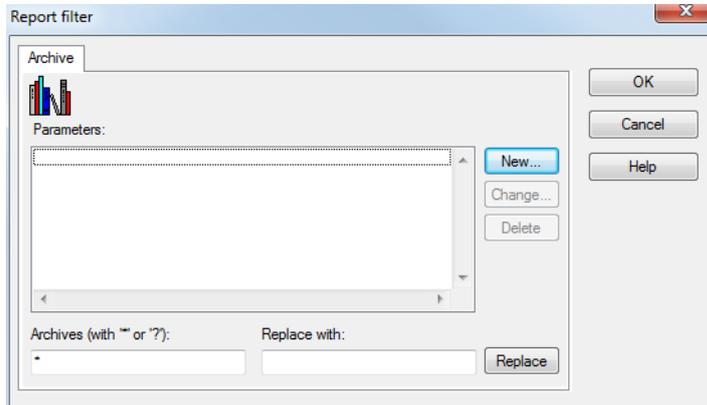
8. Click on **New**

9. The archives stored in the archive server node are displayed

10. Select the variables as shown in the screenshot
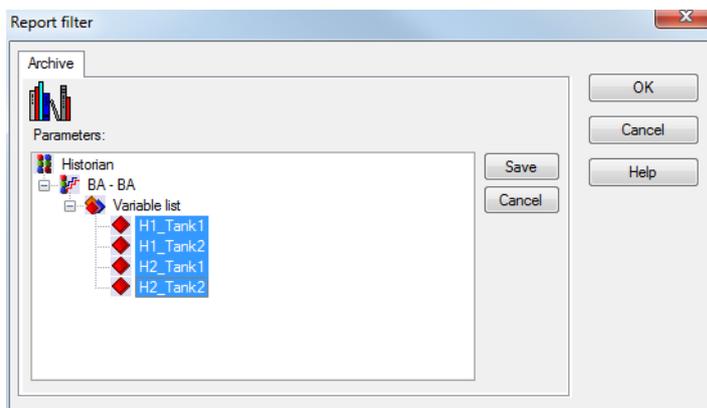
11. Click on Save

12. Select the variable "H1_Tank1"
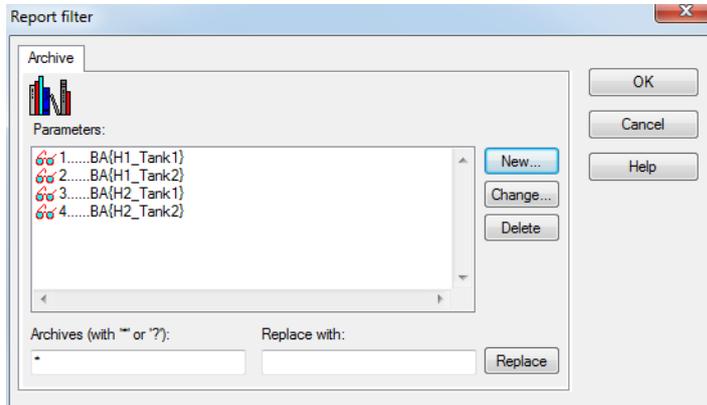
13. Confirm the selection by clicking OK

14. The function assistant is displayed again

15. Select `time` as the `parameter` with a double-click.
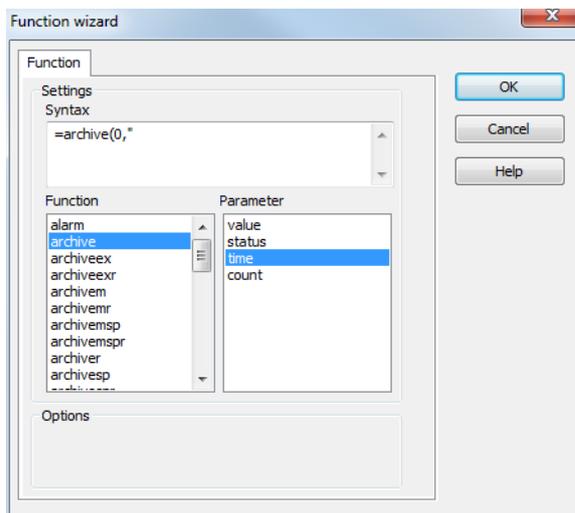
16. Select the `parameter down` with a double-click.

17. Configuration of the function is complete (`=archive(1,"time","down")`
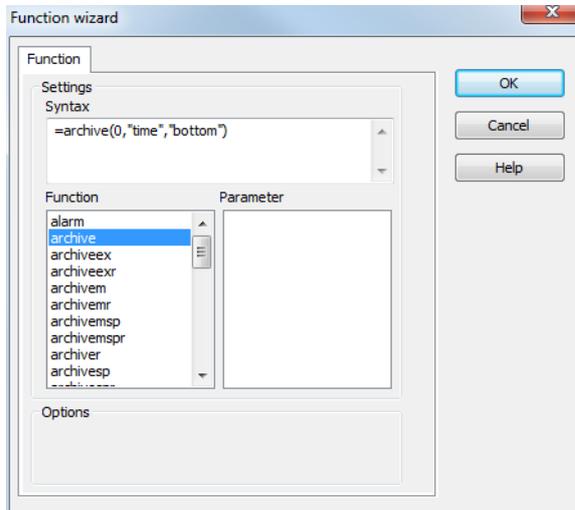
18. Close the assistant by clicking on the oĸ button

The time of variable archiving is thus continued downwards from row 5 in column A.

💡 Info

The parameters always have to be selected with a double-click. All parameters have been set up once the function in the preview is closed with a bracket and no further parameters are offered. If the oĸ button is clicked on before all parameters have been selected, the assistant is closed before all parameters for the function have been set up.

Values

You configure the values for the tanks in hall 1 and 2 in the same manner as the configuration for the time (on page 201). Pay attention to the following differences:

▸   The archive variables no longer need to be set up. They already exist and can be selected immediately. Assign each tank the corresponding variable.

▸   Instead of the `time` parameter, select the `value` parameter.

COPYING A FUNCTION

As soon as you have created a function, you can copy it to other cells and then edit it accordingly:

To do this:

1. Highlight the cell to be changed

2. Then click in the editing cell

3. In the function that is now displayed, the reference numbers refer to the corresponding archive variables as they are automatically issued during archive selection when the function is first created

4. Therefore replace the `1` as the first parameter in column `C` by a `2`
   in column `D` with a `3`
   and in column `E` with a `4`

Sum and average

For this report, you now configure the display of sum and average.

To do this, you must determine how many cells are needed for archive entries. You define these in the Report Generator. The number of entries in the list depends on the filter settings that are used to subsequently call up the report (`Execute report`, `Print report`, `Export report` functions).

In our example, we will select a time range of two minutes. As our archive records values every five seconds, this will make 24 entries for the report. Therefore:

1.   Enter the text `Sum:` into cell A29 on

2.   Enter the text `Average:` into cell A30 on

3.   Enter the function `=sum(B5:B25)` into cell B29.

4.   Enter the function `=sum(B5:B25)` into cell B30.

5.   Copy the cell B29 to C29, D29 and E29

6.   Copy the cell B30 to C30, D30 and E30

7.   Save the report under the name `Archiv.xrs`

> ### 💡 Info
>
> *When entering the functions, you can either type in cell names or click on the corresponding cell with the mouse.*
>
> *If you copy functions with cell addresses in the Report Generator, the cell addresses are automatically adapted.*

3.3.3   Displaying the report in Runtime

To be able to look at the report in Runtime, you need:

▶   A report screen

▶   A screen switching function

CREATE SCREEN OF TYPE REPORT

In order to create a `Report` screen:

1.   Create a new screen named `Report`.

2.   Select `Report` as screen type

3.   Select the frame MAIN

4.   Click in the new screen

5. Select the "Default" command from the "Control elements" menu

6. Add any other required control elements there may be

CREATING A FUNCTION

Create a screen switch function to select the screen in Runtime. To do this:

1. Add a new button to the start screen

2. Label it with `Report`

3. Change to the `Functions` node

   create a new function

4. select `Screen switching`

5. Select the `Report` screen

6. The report filter is opened
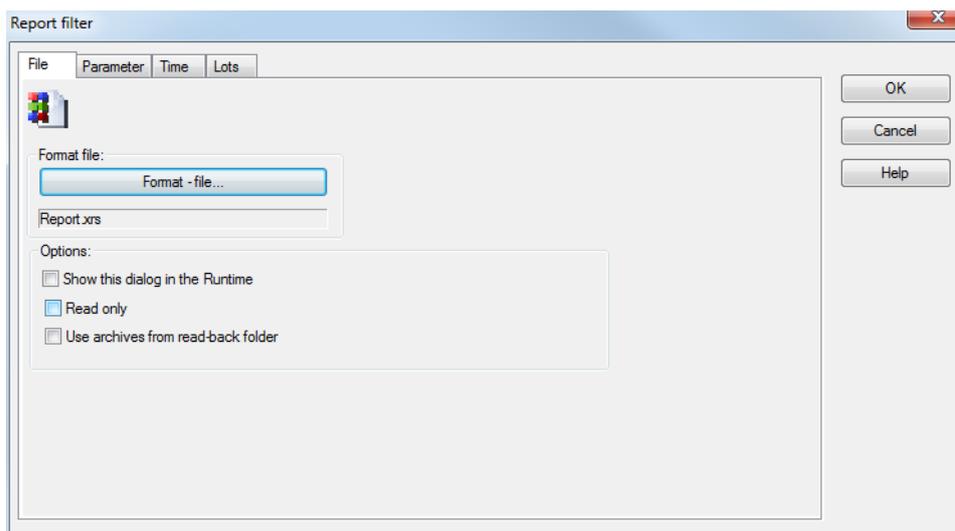
7. Click on the `Format file` button

8. Select `Archiv.xrs`

9. Open the `Time` tab
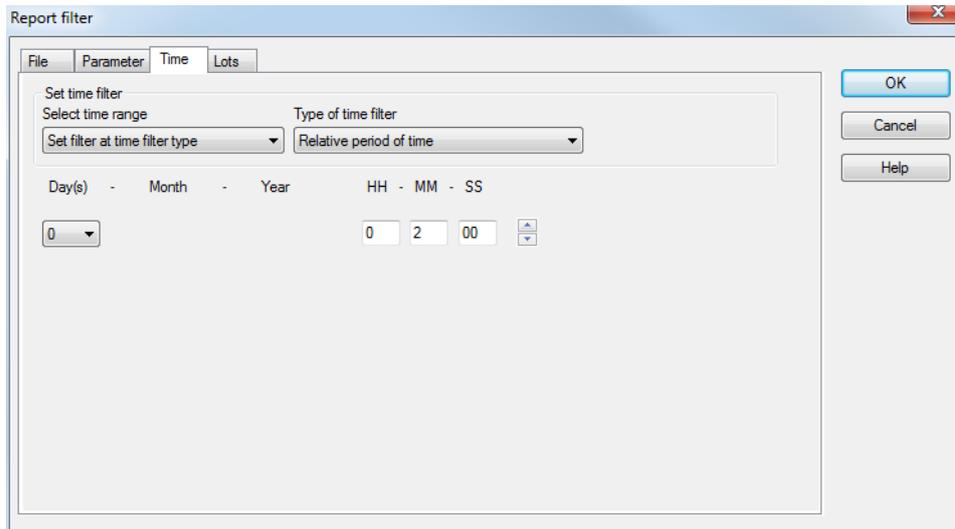
10. Set, for our example

   a) The time format to `Relative time period` and

   b) Set the time to `2 minutes`.

 You can read more about the time filter in the alarm administration manual in the time filter chapter

11. Close the filter by clicking on OK

12. Allocate a the configured button to the function

13. Start the function in Runtime

4. Report Viewer

The Report Viewer can be used to display RDL reports of archive data, AML data, and CEL data as well as online values. Two historical time ranges can also be compared using two configurable time ranges.

RDL files that display the report template for Runtime are created and edited in the Editor. There are already 7 pre-defined datasets available. Further datasets can be created (on page 228) freely; only the table scheme (on page 241) for the four data categories is stipulated.

The source data that supplies the datasets created in the RDL file in Runtime is defined in the screen switching function (on page 214) on the Report Viewer (on page 212) screen. For each dataset used in the RDL file, a dataset with the same name and data origin must be created. The source variables and filter criteria can be prescribed for these depending on their type.

TYPICAL CONFIGURATION STEPS:

1. Create a new report definition file (on page 226) or import one

2. Create Report Viewer (on page 212) screen

3. Create screen switching function (on page 214) and configure datasets (on page 218)

4. Configuring (on page 228) an RDL file

5. Call up the function in Runtime using a button or menu

4.1   Engineering in the Editor

To be able to use the Report Viewer in Runtime, the following must be carried out in the Editor:

1. Create a Report Viewer (on page 212) screen

2. Create or import an RDL file, if none is present

3. Set up a screen switching function (on page 214) on the Report Viewer screen

4. Configure the datasets in the screen switching

5. Amend the report definition file (on page 226) (RDL) to suit the datasets (on page 241) defined in zenon

6. Add a button to call up the Report Viewer in Runtime in the start screen or the menu screen

FOLDER FOR REPORT DEFINITION FILES (RDL)

RDL files are stored in the project tree in the Files/Report Viewer node. Existing RDL files can be imported and edited here or new files can be created.

REPORT TEMPLATE CONTEXT MENU

Menu item	Action
`New report definition file`	Creates a new RDL file on the basis of the supplied `default.rdl` (on page 228).
`Open report definition file`	Opens the software linked to the RFL files to edit an existing definition file. zenon first looks for MS Report Builder, then for MS Report Designer (on page 228) by default.
`Jump back to starting element`	With linked elements, jumps back to the element from which the report template was jumped to. Only present in the context menu if linked elements are present.
`Copy`	Copies selected entries to the clipboard.
`Paste`	Pastes the contents of the clipboard. If an entry with the same name already exists, the content is pasted as "Copy of".
`Delete`	Deletes selected entries.
`Import existing report definition file`	Opens dialog to import existing report definition files. These must be of RDL 2.0 type.
`Remove all filters`	Removes all filter settings.
`Edit selected cell`	Opens the selected cell for editing. The binocular symbol in the header shows which cell has been selected in a highlighted line.
`Replace text in selected column`	Opens the dialog for searching and replacing texts.
`Properties`	Opens the `Properties` window for the selected entry.
`Help`	Opens online help.

4.1.1 Create Report Viewer screen

In order to create a Report Viewer screen:

1. Create a new screen

2. Select the `Report Viewer` special screen type in the drop-down list

3. Name the screen

4. select **Add template** in the control element menu

5. The standard elements are created and placed automatically

Parameters	Description
Add template	Opens the dialog for selecting a template for the screen type. Templates are shipped together with zenon and can also be created by the user. Templates add pre-defined control elements to pre-defined locations in the screen. Elements that are not necessary can also be removed individually once they have been created. Additional elements are selected from the drop-down list and dragged onto the screen. Elements can be moved on the screen and arranged individually.
Report window	Functions and profiles for the report.
Refresh	Reloads the report definition and the data and displays the report with the updated data.
Print	Prints the report on the printer defined for values and logs.
Create PDF	Saves the report currently being displayed as a PDF file in the folder defined for export.

4.1.2   Screen switching to a Report Viewer screen

To use the Report Viewer in Runtime, configure screen switching to a Report Viewer screen type:

1. Create a new **Report Viewer** screen

2. Select the **New function** command in the **Functions** node

3. select the **Screen switching** function

4. Select the `Report Viewer` screen

5. The report filter (on page 218) is displayed

6. Select the RDL file

7. Define the time filter

8. Create the desired datasets (on page 218)

9. Link the functions with a button in the screen, in order to switch in Runtime

10. Edit the datasets in the RDL file (on page 226) according to their settings in screen switching
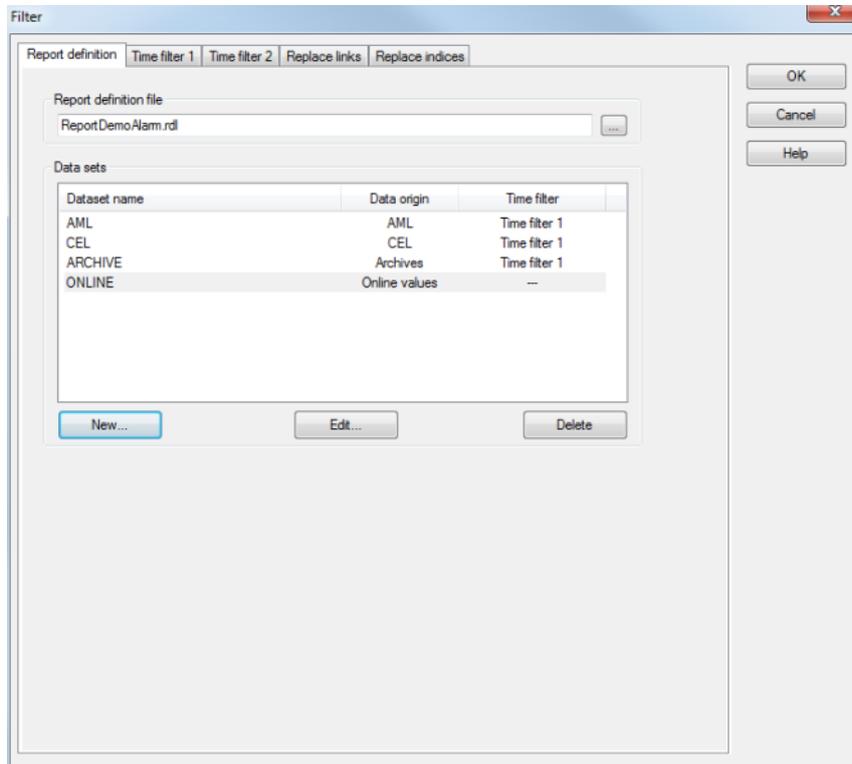
Parameters	Description
`File for report definition`	RDL file that is used for report definition. This file must be present in the **File/report templates** node. A click on the `...` button opens the file selection dialog.
Datasets	Display of defined datasets (on page 218). The list contains: ▸ Dataset name ▸ Origin of the data ▸ Time filter: - time filter 1 - time filter 2
New	Opens the dialog (on page 217) to create a new report definition.
Edit	Opens the dialog (on page 218) to edit the definition of the selected element.
Delete	Deletes selected element and its definition from the list.
OK	Applies all changes and closes dialog.
Cancel	Discards all changes and closes dialog.
Help	Opens online help.

TIME FILTER

There are two time filters available. Two historical areas can therefore be compared.

Time filtering can be carried out in two ways:

1. Specify time area in the Editor
   Two fixed time areas are used. A time period is given in the editor. It is only possible to filter according to this time period in Runtime. Other filters - such as filtering according to variable name, alarm/event groups and alarm/event classes etc. can no longer be amended in Runtime.

2. Time filter can be amended in Runtime
   Two pre-defined times are used. The time filter is defined in the Editor and can be changed in Runtime as desired.

You can read details on configuration in the Alarm administration manual in the Time chapter.

Note: If you set option `no time filter` as time filter type, all Runtime entries since 1.1.2000 are displayed.

New dataset dialog

To define a new data set:

1. Define, in the screen switching (on page 214), the RDL file on which it is based on a Report Viewer screen

2. click on the button `New`

3. The dialog to create the datasets is opened

4. Select the data origin you want to assign

5. Confirm the selection by clicking `OK`

6. the dialog (on page 218) for configuring the defined type is opened
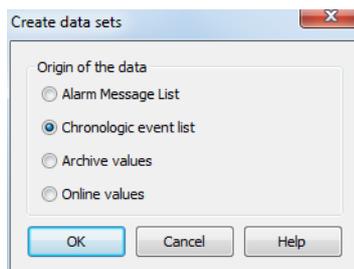
Parameters	Description
`Origin of the data`	Selection of the data origin: ▸ Alarm Message List ▸ Chronological Event List ▸ Archive values ▸ Online values
OK	Confirms selection and opens dialog (on page 218) for configuration of the type.
Cancel	Closes dialog without further configuration.
Help	Opens online help.

Datasets in the Report Viewer

The settings for the different datasets depend on the type of data origin to be configured:

▶ Online values (on page 219)

▶ AML data (on page 221)

▶ CEL data (on page 223)

▶ Archive data (on page 225)

Online values

Parameters	Description
Dataset name	Name of the data set: ▶ Must correspond exactly to the name of the data set in the RDL file (on page 226) ▶ Must not contain any spaces or special characters.
Variables	List of selected variables with: ▶ Name ▶ Identification ▶ Address ▶ Project
Add	Opens the dialog for selecting several variables.
Remove	Deletes selected variable from the list.
OK	Applies settings and closes dialog.
Cancel	Discards changes and closes dialog.
Help	Opens online help.

AML data

Parameters	Description
`Dataset name`	Name of the data set: ▸ Must correspond exactly to the name of the data set in the RDL file (on page 226) ▸ Must not contain any spaces or special characters.
`Time filter`	Selection of the time filter. Can be used to compare two periods of time based on the origin of the data: ▸ Time filter 1 ▸ Time filter 2
General tab	Settings for: ▸ Variable filter ▸ Alarm type ▸ Origin of the data ▸ Alarm/event groups, alarm/event classes and alarm areas For details see the Alarm administration manual, General chapter.
Text tab	Settings for the text filter. For details see the Alarm administration manual, Text chapter.
Status tab	Data for evaluation of status bits. For details, see the Alarm administration manual, Status chapter.
Equipment modeling tab	For details see the Equipment modeling manual, Equipment modeling dialog chapter.
OK	Applies settings and closes dialog.
Cancel	Discards changes and closes dialog.
Help	Opens online help.

CEL data

Parameters	Description
`Dataset name`	Name of the data set: ▸ Must correspond exactly to the name of the data set in the RDL file (on page 226) ▸ Must not contain any spaces or special characters.
`Time filter`	Selection of the time filter. Can be used to compare two periods of time based on the origin of the data: ▸ Time filter 1 ▸ Time filter 2
General tab	Settings for: ▸ Variable filter ▸ Origin of the data ▸ Alarm/event groups, alarm/event classes and alarm areas For details see the Chronological Event List manual, General chapter.
Text tab	Settings for the text filter. For details see the Alarm administration manual, Text chapter.
Status tab	Data for evaluation of status bits. For details, see the Alarm administration manual, Status chapter.
Equipment modeling tab	 For details see the Equipment modeling manual, Equipment modeling dialog chapter.
OK	Applies settings and closes dialog.
Cancel	Discards changes and closes dialog.
Help	Opens online help.

Archive values

Parameters	Description
Dataset name	Name of the data set: ▸ Must correspond exactly to the name of the data set in the RDL file (on page 226) ▸ Must not contain any spaces or special characters.
Time filter	Selection of the time filter. Can be used to compare two periods of time based on the origin of the data: ▸ Time filter 1 ▸ Time filter 2
Archive – variable names	List of selected variables.
Add	Opens the dialog for selecting several variables.
Remove	Deletes selected variable from the list.
OK	Applies settings and closes dialog.
Cancel	Discards changes and closes dialog.
Help	Opens online help.

4.2 Report definition files

Report definition files (RDL) must be type 2 files in order to be able to be used in the zenon Report Viewer. Type 3 RDL files can be displayed, however features that are not present in version 2 are displayed as an empty space.

To create your own datasets or edit existing ones, you need appropriate software to render the RDL files. There are three options for this:

▶ Microsoft Business Intelligence Development Studio:

Is supplied by COPA-DATA and installed with the installation of zenon.

The Business Intelligence Development Studio consists of Microsoft Visual Studio 2008 with additional project types which are only available for SQL Server Business Intelligence. The Business Intelligence Development Studio is the primary environment for Business Services projects such as reporting. Templates for creating objects necessary for the Business Intelligence solution are provided as well as designer, tools and assistants for processing these objects.

▶ Microsoft Report Builder 2.0:

Is available on the Microsoft download page (http://www.microsoft.com/download/en/details.aspx?id=24085) as a download. It provides a wizard for the configuration of datasets and is recommended for users without special knowledge for the configuration of RDL files.
Attention: Note the licensing conditions before downloading.
Users of the zenon Analyzer already have Report Builder 3 and the SQL license required for the Report Builder. However, Report Builder 2 must be installed and used for editing the datasets of the Report Viewer.

▶ Software from other suppliers:

Software from other manufacturers can also be used to edit report definition files.

If both the MS Report Builder and the Microsoft Business Intelligence Development Studio are installed, you can select the desired tool in a dialog.

> ⚠ **Attention**
>
> *Note:*
>
> ▶ RDL files should always be created and edited with the same tool.
>
> ▶ RDL versions 2 and 3 must not be mixed.
>
> ▶ If an RDL file was saved with the Microsoft Business Intelligence
>   Development Studio 2008R2, it can no longer be edited with the MS Report
>   Builder 2.0.

DATA SOURCE

The properties for the data source (superordinate to the datasets) must not be changed.

The standard settings are:
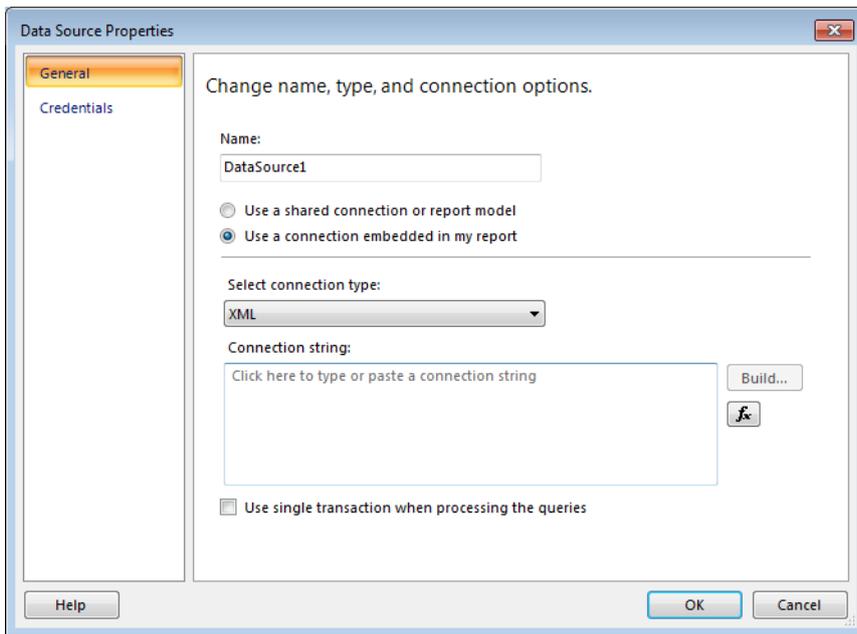
Parameters	Description
Name	DataSource1
Connection	Use a connection embedded in my report
Select connection type	XML

4.2.1 Default.rdl

Default.rdl is called up if a new RDL file is created in the Editor. There are already pre-defined datasets available in the supplied default.rdl, which can be provided with data in Runtime:

- ▶ Online values

- ▶ AML with time filter 1

- ▶ AML with time filter 2

- ▶ CEL with time filter 1

- ▶ CEL with time filter 2.

- ▶ Archive values time filter 1

- ▶ Archive values time filter 2

If more datasets are required, these can be created (on page 234) with any desired name. If fewer datasets are required, these must be deleted from the RDL file. The names of the datasets must correspond to the dataset names in the screen switching (on page 214).

Note: The table schematic (on page 241) for the 4 data categories is stipulated and must not be modified.

4.2.2 Creating and editing RDL files with MS Report Builder 2.0

MS Report Builder is available on the Microsoft download page (http://www.microsoft.com/download/en/details.aspx?id=24085) as a free download.
Attention: Note the licensing conditions before downloading.
Users of the zenon Analyzer already have Report Builder 3 and the SQL license required for the Report Builder. However, Report Builder 2 must be installed and used for editing the datasets of the Report Viewer.

Install the Report Builder and check if it is correctly linked to zenon.

CREATING AN RDL FILE

To create an RDL file:

1. Select, in the zenon Editor, in the `File/Report Viewer` node in the context menu or the `New report definition file` command in the toolbar.A new report definition file is created, based on a default file (on page 226).

2. Start the Report Builder by double clicking on the file.
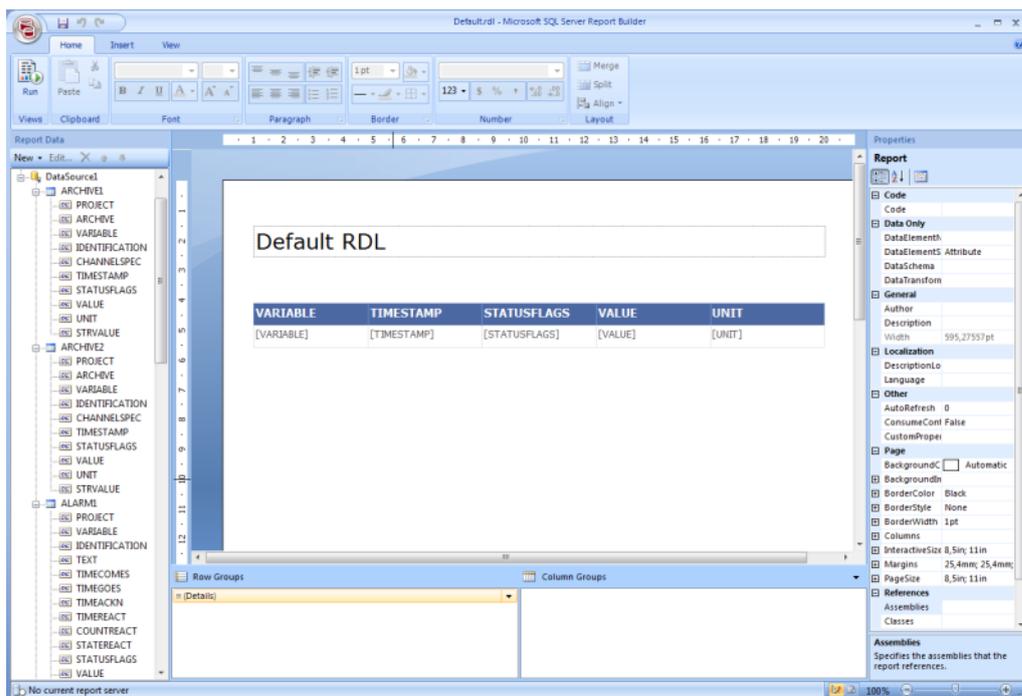
3. Adapt the datasets (on page 234) to the configuration in the screen switching function by:
   - Rename: Datasets must be identical with the dataset names in screen switching
   - Delete: Datasets that do not have any corresponding linking in the screen switching and are thus not supplied with data lead to an error message in Runtime
   - Create new: Datasets that are configured in the screen switching but do not have any corresponding entries in the RDL file lead to an error message in Runtime.

4. Select, in the `Insert` menu, the command *Table -> Table Wizard.*

5. The wizard is started.

6. Select the desired dataset.
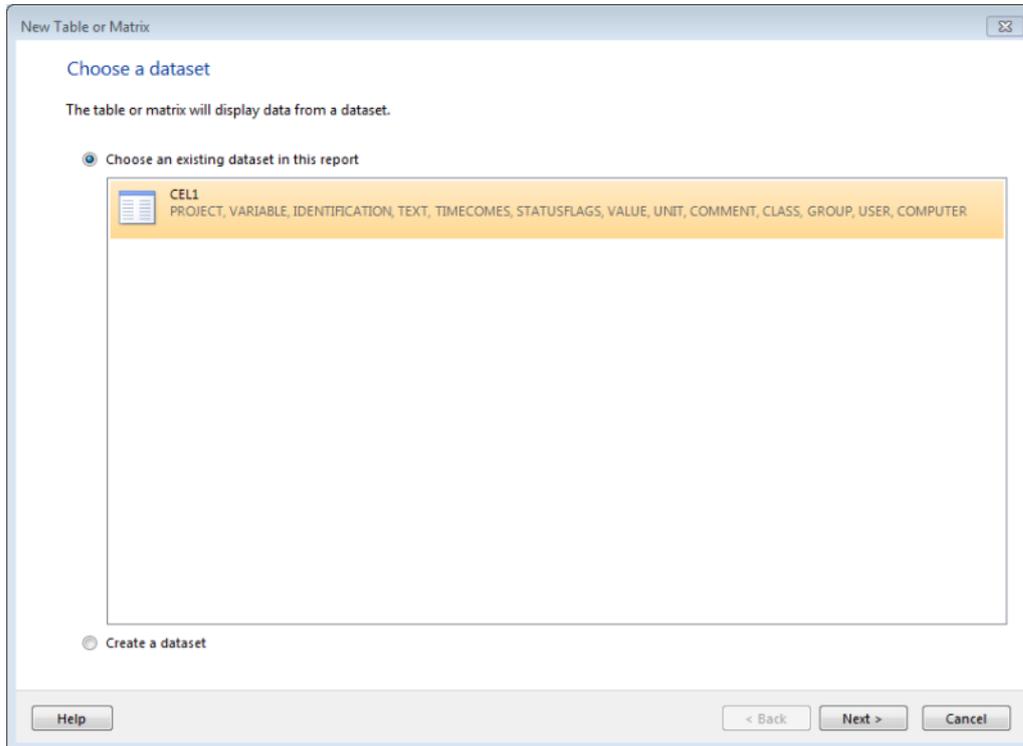
7. Locate the fields as desired.
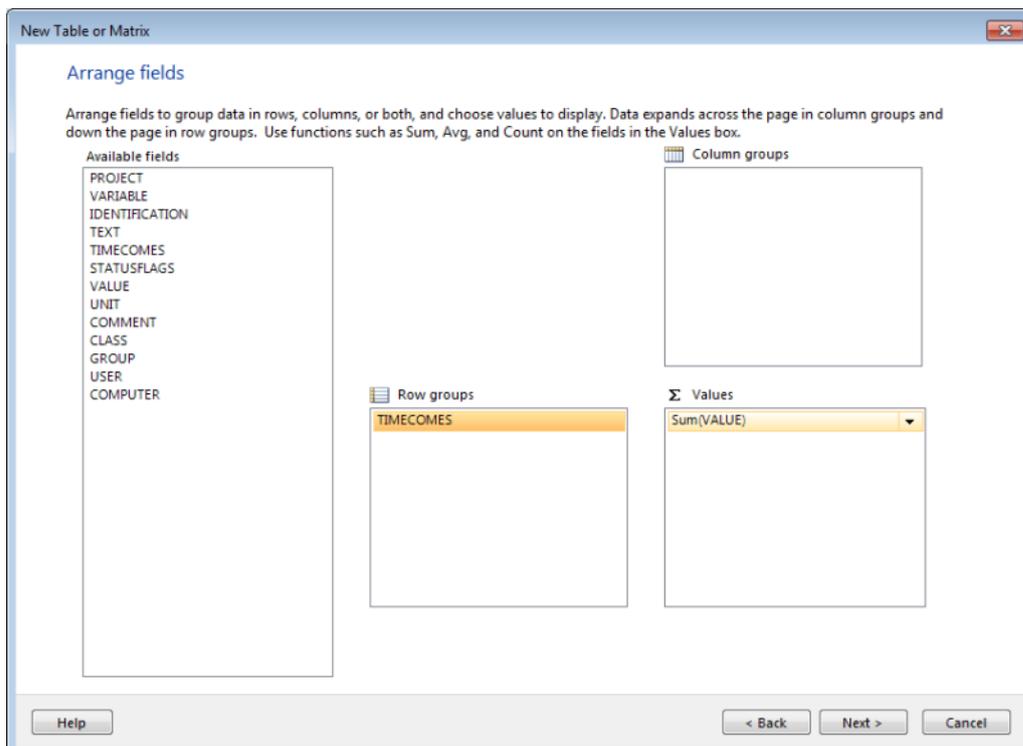
8.   Click on **Next** and select the layout.
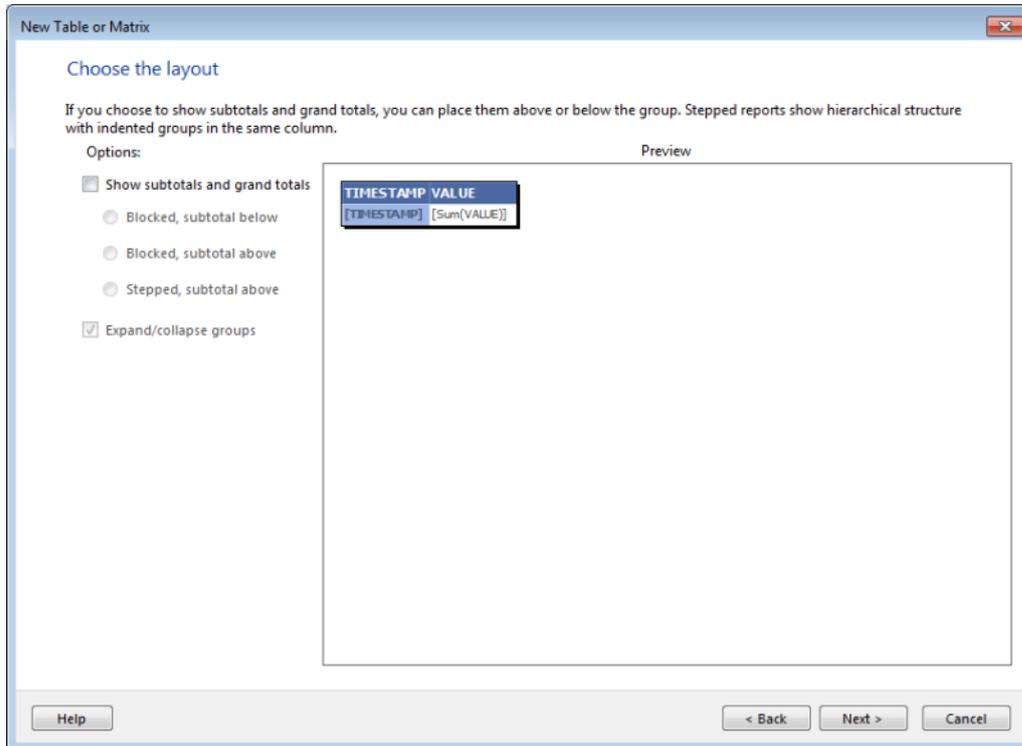
9.   Click on **Next** and select the style of the display.
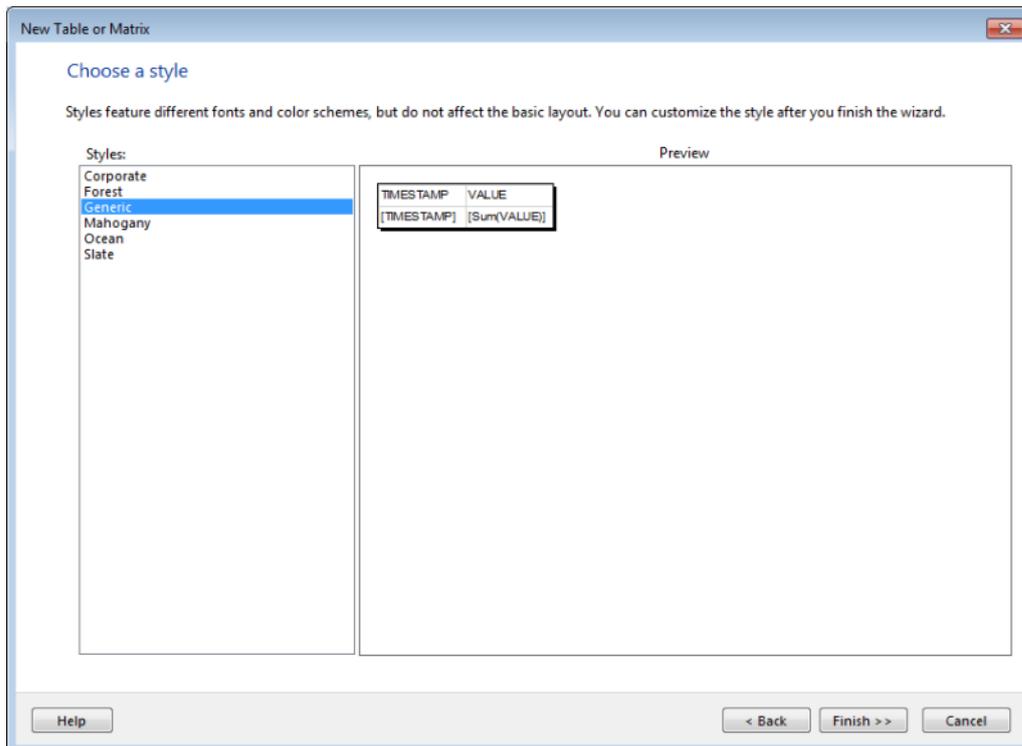
10. Click on `Finish.`

    The dataset is displayed with the finished table
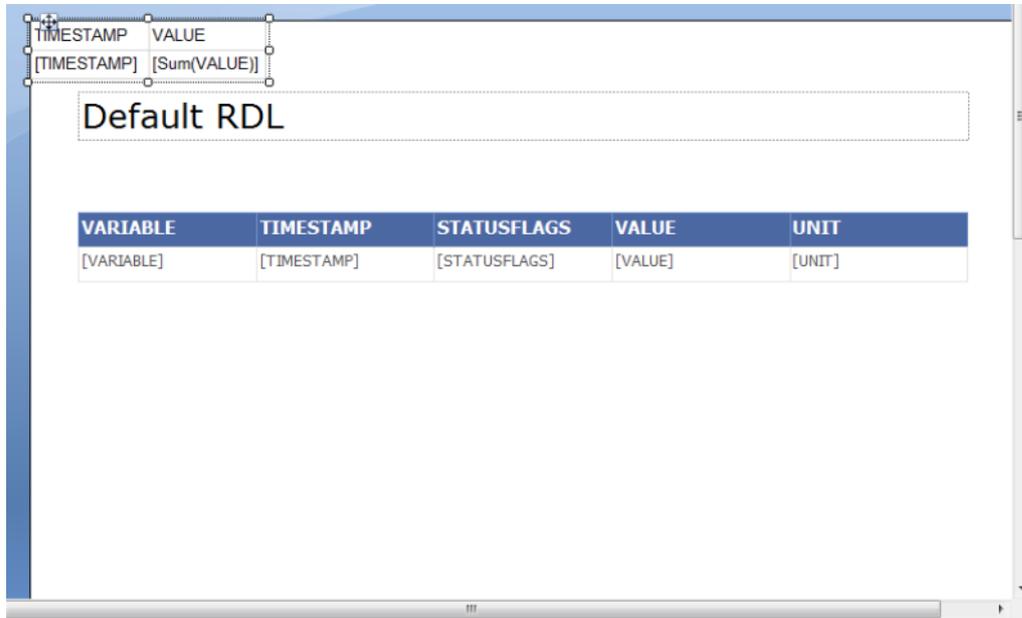
11. Delete the sample table.

12. Place the newly created table at the desired location.

13. Adapt the heading as desired.

14. Repeat the process for all datasets to be configured.

15. Save the RDL file.
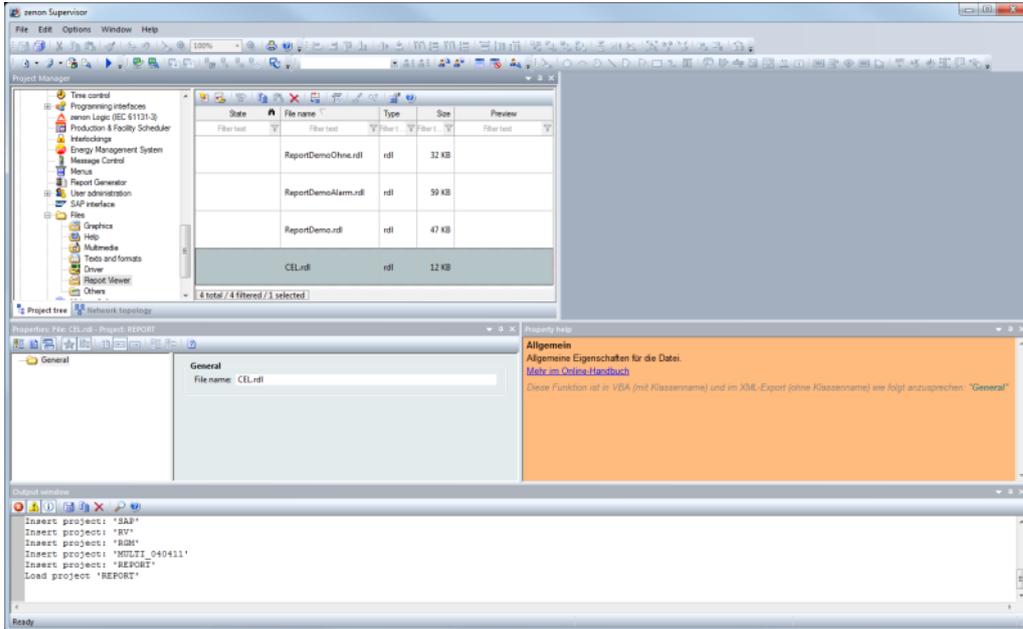
Result in Runtime:
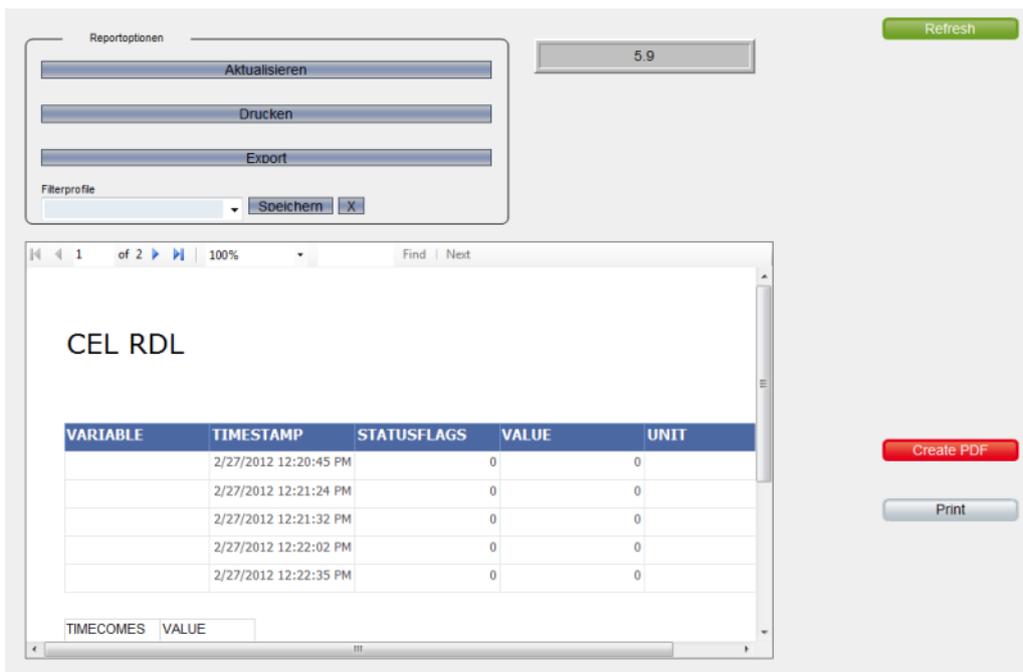
EDITING AN RDL FILE

To edit an RDL file, double click the file or select the `Open report definition file` command. Use the same process as creation for editing.
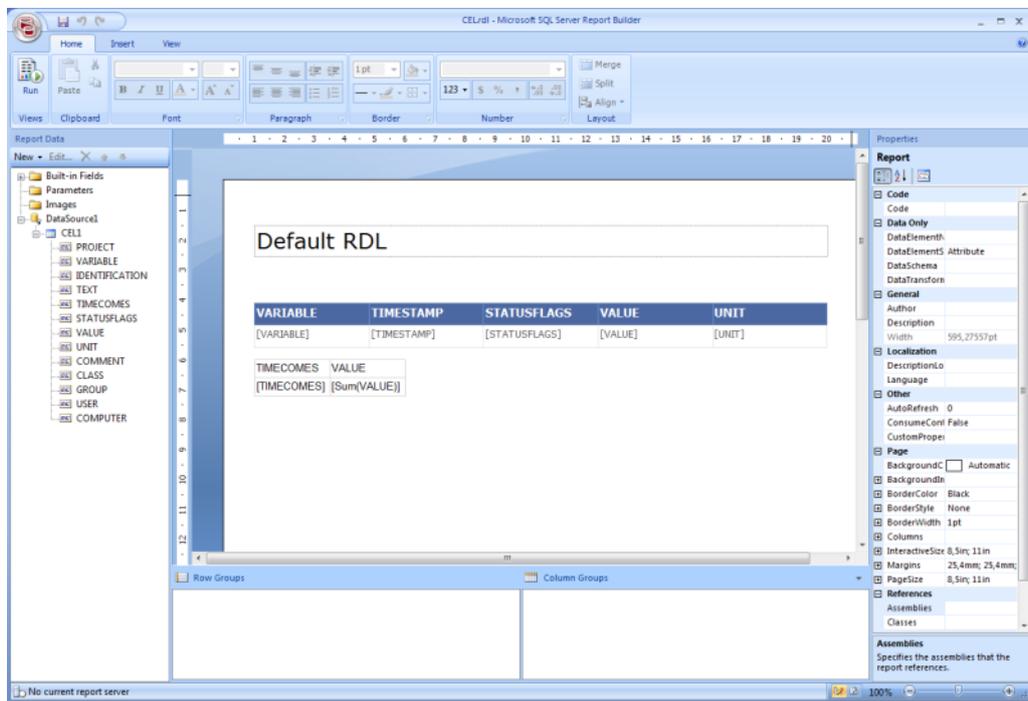
Administering datasets

You can do the following with datasets in the Report Builder:

▶ Rename

▶ Delete

▶ create new

Number and name of the dataset of an RDL file must always correspond to that of the dataset defined in screen switching.

> ⚠ **Attention**
>
> *Note:*
>
> ▸ The data source must not be edited.
>
> ▸ The data source must not be edited.

RENAMING A DATASET

To rename a data set:

1. Right-click on the data set

2. Select the **Properties** command in the context menu

3. The dialog for setting the properties of the data set is opened

4. Give it the desired name in the `Name` property

Note: the name must correspond to the name in the screen switching (on page 214). It must not contain any spaces or special characters.
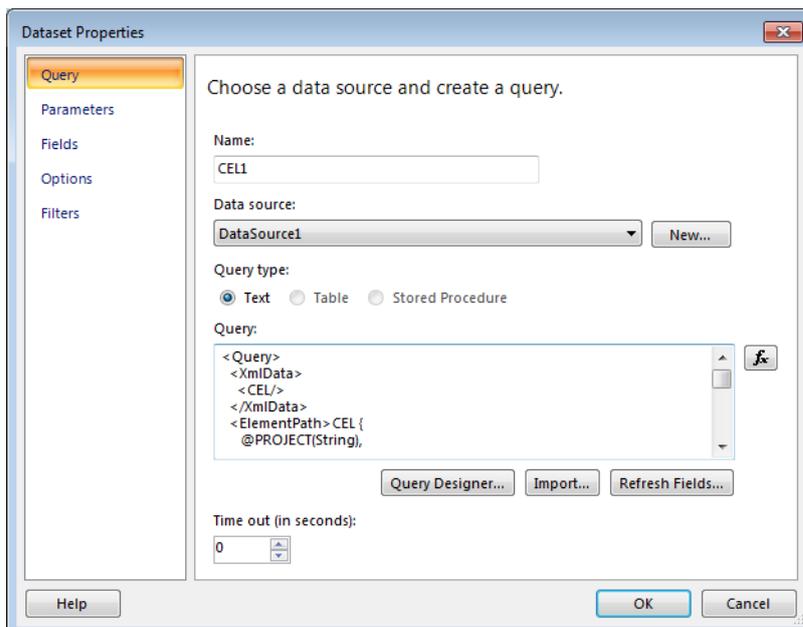
5. Leave the other settings unchanged

6. Close the dialog by clicking on OK

DELETING THE DATA SET

To delete a data set:

1. Highlight the data set to be deleted in the report data window below the data source
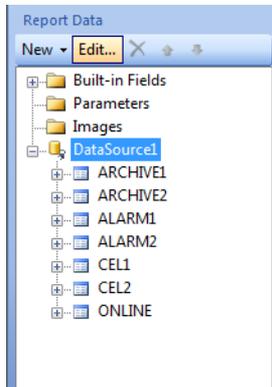
2. Select the `Delete` command in the context menu

3. the data set is deleted after confirmation is requested

CREATING A NEW DATA SET

To create a new data set:

1. Right-click on the data source

2. Select `Add dataset` in the context menu
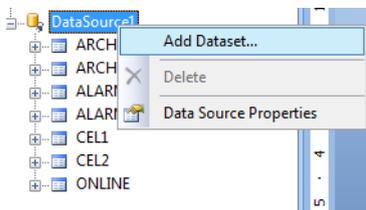
3. A new data set is created and the properties window is opened
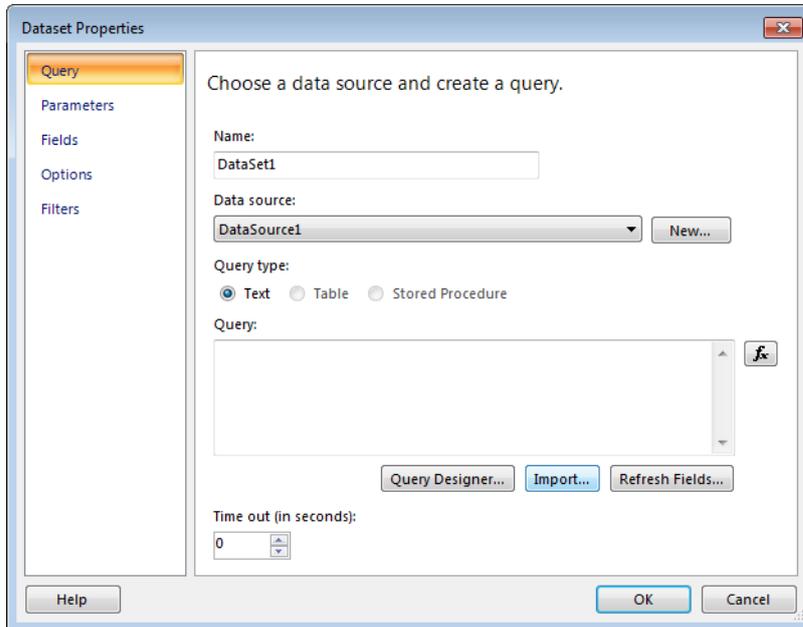
4. name it as you wish

Note: the name must correspond to the name in the screen switching (on page 214). It must not contain any spaces or special characters.

5. Insert the appropriate query; you have three options for doing this:

   a) `Query Designer`: Opens the dialog for manual input of the query

   b) `Import`: Import from an existing RDL file

   c) Copying manually from an existing dataset: First open the corresponding source dataset, copy the content of the query and paste the content of the query in the `Query` field in the new dataset

6. Close the dialog by clicking on OK

IMPORTING A QUERY

To import a query from an existing RDL file:

1. Click on the `Import` bin the dataset properties dialog

2. The file selection assistant is opened

3. Search for and open the desired RDL file

4. The import dialog is displayed
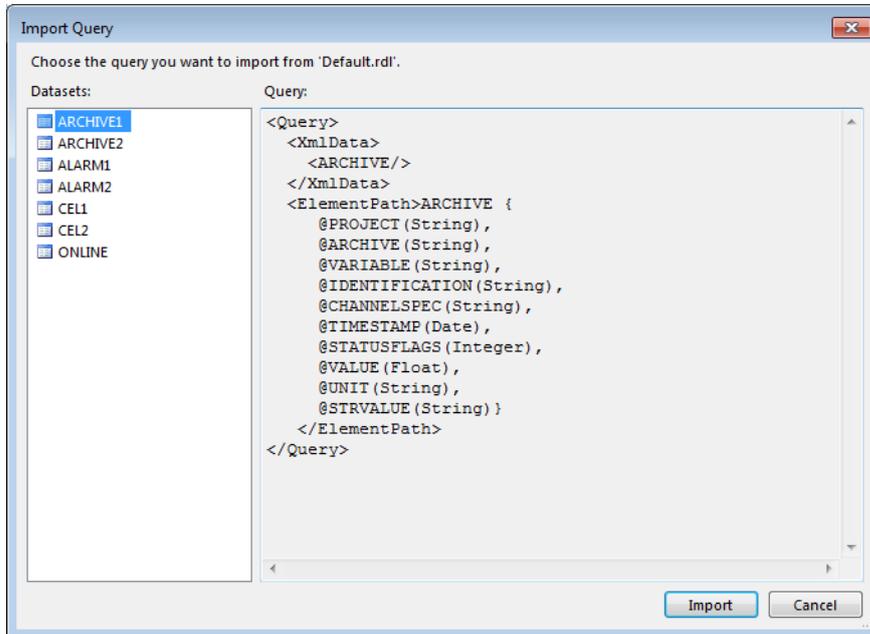
5. Highlight the desired dataset
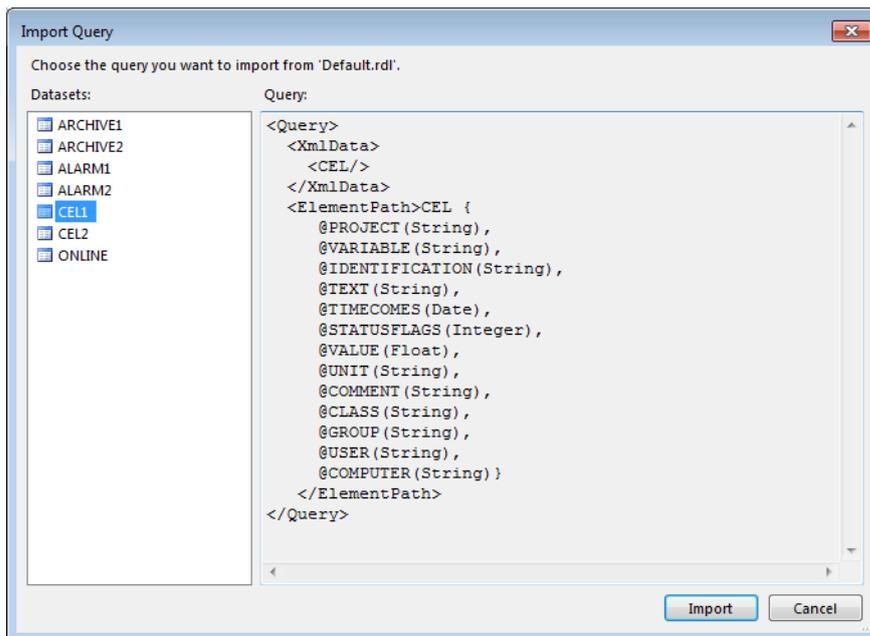
6. Click on **Import**

7.   The query of the source data set is pasted to the dataset that is currently open
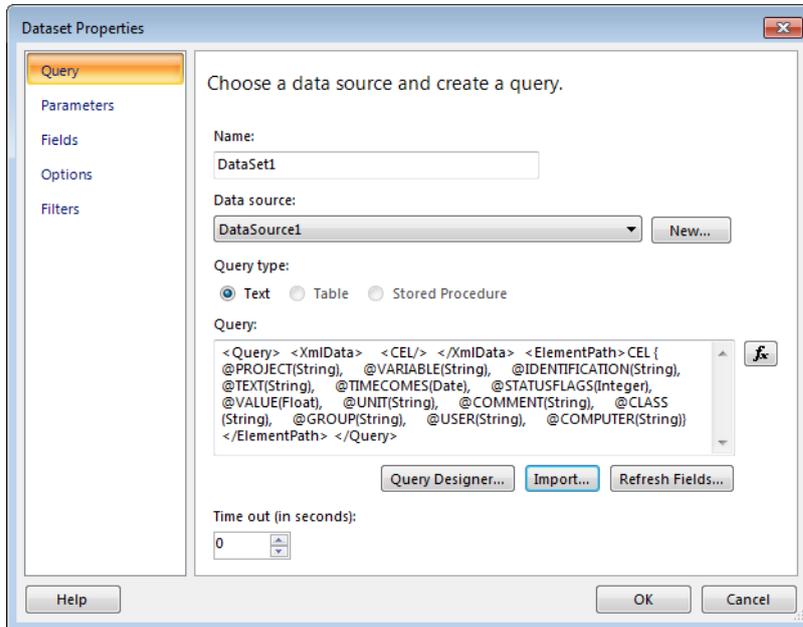
4.2.3   Create and edit RDL files with the help of the Microsoft Business Intelligence Development Studio

The Microsoft Business Intelligence Development Studio is shipped and installed together with zenon.

To create and edit RDL files with the help of the Microsoft Business Intelligence Development Studio, you basically act the same way as with the Report Builder 2.0 (on page 228).

The differences are primarily:

▶   For the report to be edited, you must first create a new project of type `Report Server Project` - Solution - in the Microsoft Business Intelligence Development Studio.
They are created automatically when you open the RDL file via double click. When closing the Business Intelligence Development Studios you can save the project and the solution in the temporary folder. If you do this, you are no longer requested to do so.
The datasets and the report can now be edited in Microsoft Business Intelligence Development Studio.

▶ There is no table wizard in the Microsoft Business Intelligence Development Studio; the tables of the datasets must be configured manually.

▶ The version is not checked:
This way, features from version 3 can be added to RDL files from version 2. These are not displayed in Runtime however.

▶ In the Microsoft Business Intelligence Development Studio the datasets are separated from the data sources. The data source must not be changed.

INSTRUCTION

For a comprehensive guide about the use of the Microsoft Business Intelligence Development Studios see the Microsoft website in several languages:

Language	Link
`English:`	http://msdn.microsoft.com/en-en/library/ms173767.aspx (http://msdn.microsoft.com/en-en/library/ms173767.aspx)
`German:`	http://msdn.microsoft.com/de-de/library/ms173767.aspx (http://msdn.microsoft.com/de-de/library/ms173767.aspx)
`French:`	http://msdn.microsoft.com/fr-fr/library/ms173767.aspx (http://msdn.microsoft.com/fr-fr/library/ms173767.aspx)
`Italian:`	http://msdn.microsoft.com/it-it/library/ms173767.aspx (http://msdn.microsoft.com/it-it/library/ms173767.aspx)

DELETE DATASETS

Unused datasets must be deleted in the Microsoft Business Intelligence Development Studio as they may cause errors in the Runtime.

To delete datasets:

1. click on `View`
2. select `Report Data`
3. navigate to `Datasets`
4. delete all unused datasets

4.2.4   Definition of data sets

The columns of the different dataset types are stipulated.

Note: If zenon Analyzer reports are to be reused in the Report Viewer, the following applies:

▶   Language-switchable texts have already had the language switched

▶   Unit-switchable values are given in a secondary unit

▶   Time stamps correspond to the local time of the executing computer

DEFINITION OF AML DATASET

Name of the column	Data type	Meaning
PROJECT	String	Name of the project.
VARIABLE	String	Name of the variable.
IDENTIFICATION	String	Identification of the variable.
TEXT	String	Entry text.
TIMECOMES	Date	Time when the alarm occurs.
TIMEGOES	Date	Time.
TIMEACKN	Date	Time of acknowledgement.
TIMEREACT	Date	Time of reactivation.
COUNTREACT	Integer	Number of reactivations.
STATEREACT	Integer	Status of reactivation.
STATUSFLAGS	Integer	Status bits.
VALUE	Float	Value of variables.
UNIT	String	Unit.
COMMENT	String	User comments.
CLASS	String	Alarm class.
GROUP	String	Alarm group.
USER	String	name of the user.
COMPUTER	String	Name of the computer.

DEFINITION OF CEL DATASET

Name of the column	Data type	Meaning
PROJECT	String	Name of the project.
VARIABLE	String	Name of the variable.
IDENTIFICATION	String	Identification of the variable.

TEXT	String	Entry text.
TIMECOMES	String	Time when the event occurs.
STATUSFLAGS	Integer	Status bits.
VALUE	Float	Value of variables.
UNIT	String	Unit.
COMMENT	String	User comments.
CLASS	String	Alarm class.
GROUP	String	Alarm group.
USER	String	name of the user.
COMPUTER	String	Name of the computer.

DEFINITION OF ARCHIVE DATA SET

Name of the column	Data type	Meaning
PROJECT	String	Name of the project.
ARCHIVE	String	Name of the archive.
VARIABLE	String	Name of the variable.
IDENTIFICATION	String	Identification of the variable.
CHANNELSPEC	String	Variable including type of data reduction.
TIMESTAMP	String	Time stamp.
STATUSFLAGS	Integer	Status bits.
VALUE	Float	Value of variables.
UNIT	String	Unit.
STRVALUE:	String	Value as string.

DEFINITION OF ONLINE DEFINITION

Name of the column	Data type	Meaning

PROJECT	String	Name of the project.
VARIABLE	String	Name of the variable.
IDENTIFICATION	String	Identification of the variable.
TIMESTAMP	String	Time stamp.
STATUSFLAGS	Integer	Status bits.
VALUE	Float	Value of variables.
UNIT	String	Unit.
STRVALUE:	String	Value as string.

4.3 Operating during Runtime

the Report Viewer screen is switched to in Runtime with a screen switching function. Depending on the configured report definition, reports for AML, CEL, archives and online values can be displayed. You can also do the following with these reports, if the corresponding functions are configured:

▸ Update: Reload report definition and data and display report

▸ Filter: Defining, saving and deleting filters

▸ Print: on the printer defined for values and logs

▶ Export: as a PDF file in the defined export folder

Parameters	Description
`Refresh`	Reloads the report definition and the data and displays the report with the updated data. After the report file in the Editor has been changed, the screen must be called up again; a reload is not sufficient.
`Print`	Prints the report on the printer defined for values and logs.
`Export`	Saves the report currently being displayed as a PDF file in the folder defined for export.
`Filter profiles`	Filter settings that can be saved by the user in Runtime.
`Profile selection`	Select profile from list.
`Save`	Saves current setting as a profile.
`X (save)`	Deletes selected profile.
`Report list`	Displays the reports defined in the screen switching function (on page 214). The list can be searched. Long lists are broken up into several pages.

NAVIGATION, ZOOM AND SEARCH

The report header offers elements for:

- ▶ Navigation

- ▶ Zoom

- ▶ Search

Parameters	Description
Page navigation	Displays number of pages and current page. Enables navigation in the document by means of ▸ Cursor keys ▸ Selection of a particular page
Zoom	Selection of a zoom level from a drop-down list: ▸ Page Width ▸ Whole Page ▸ 500% ▸ 200% ▸ 150% ▸ 100% (default) ▸ 75 % ▸ 50 % ▸ 25%
Search	Input of a search term in the search field. ▸ **Find**: Displays the first place of finding ▸ **Next**: Navigates to further places of finding Wildcards cannot be used.

4.4 Troubleshooting

ERROR MESSAGES IN POP-UPS

Error message	Meaning

`DataTable missing for DataSet:`
 	
`<DatasetName>`
	A dataset is used in the report definition, but there is no table with this name in the configuration. The report cannot be displayed.

`DataSet not used for DataTable:`
 	
`<TableName>`
	A table was defined, but the report definition does not use this name for a dataset. This message is only given after a missing table has been established.