

imc FAMOS Update Info 7.1.0

Table of contents

imc FAMOS Update Info

1.1 imc FAMOS Update-Info (Version 7.1)	2
1.1.1 Local variables	2
1.1.2 Handling physical units	3
1.1.3 New functions	3
1.1.3.1 ConvertUnit ().....	3
1.1.3.2 Measurements.....	4
1.1.3.3 Measurement uncertainty.....	4
1.1.3.4 FOR, FOREACH.....	5
1.1.3.5 Comparison operators, logic.....	5
1.1.3.6 Spectral Kit.....	5
1.1.3.7 Class-counting Kit.....	5
1.1.3.8 Curve Window Kit.....	5
1.1.4 Curve Window	6
1.1.4.1 Units.....	6
1.1.4.2 3D bars.....	7
1.1.4.3 Color legend for 3D.....	8
1.1.4.4 4D (3D + color).....	10
1.1.4.5 Split-Mode.....	11
1.1.4.6 Selection of measurement uncertainty.....	12
1.1.4.7 Channel info.....	12
1.1.4.8 Presettings.....	12
1.1.5 Panel	12
1.1.5.1 New widgets.....	12
1.1.5.2 Drag&Drop for text variables.....	13
1.1.5.3 Panel remote control.....	13
1.1.5.4 Debugging Panel-event sequences.....	13
1.1.6 Variable list	14
1.1.6.1 Filtering for the Variables list.....	14
1.1.7 Editor	14
1.1.7.1 Editor with sub-sequences.....	14
1.1.7.2 Autocomplete.....	14
1.1.7.3 Text as HTML text.....	15
1.1.7.4 Printing.....	15
1.1.7.5 Help window.....	16
1.1.8 Data sources-Browser	16
1.1.9 Display / Curve window	16
1.1.10 Import/Export	16
1.1.10.1 Excel 2016.....	16

imc FAMOS Update Info

1.1 imc FAMOS Update-Info (Version 7.1)

1.1.1 Local variables

Variables can be defined as "local" within a sequence. Such variables are only valid during the execution of the sequence; at the end of the sequence execution, such variables are deleted automatically. For this purpose, the new keyword "LOCAL" has been introduced.

The **declaration** is made directly in the assignment:

```
LOCAL temp = Ramp(0, 1, 100)
```

or by means of forwards declaration:

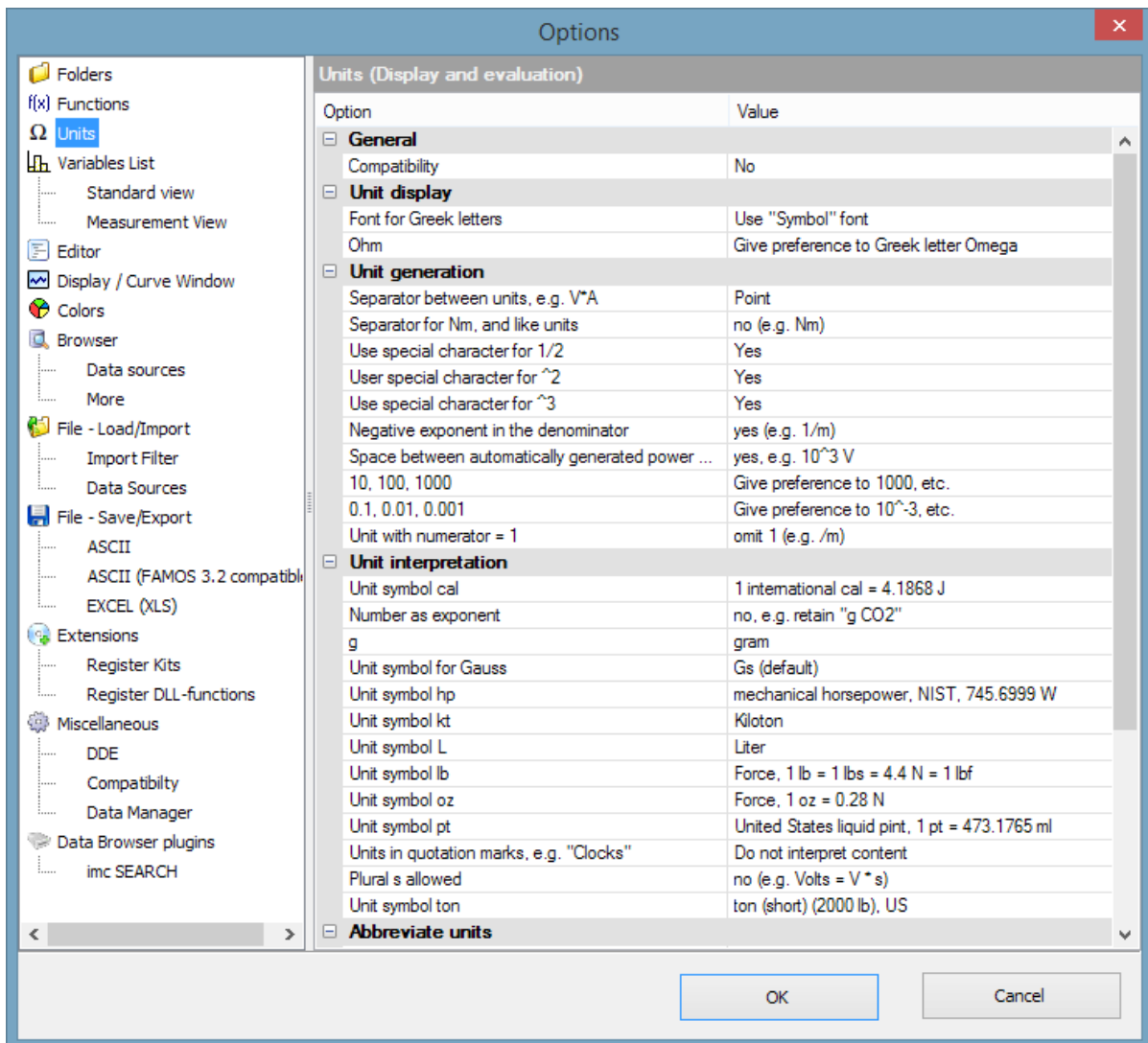
```
LOCAL temp  
temp = Ramp(0, 1, 100)
```

With the forwards declaration, it is also possible to use wildcards (*, ?):

```
LOCAL ?      ; all variables with exactly one character  
LOCAL #*    ; all variables which begin with '#'
```

1.1.2 Handling physical units

Improved display and computation of physical variables. In addition to the fundamental SI-units, many other commonly used units are now also recognized. The detailed behavior of the units manager can be specified:



1.1.3 New functions

1.1.3.1 ConvertUnit ()

This function converts a data set's physical units and changes the numerical values accordingly.

Examples:

```
Signal = ConvertUnit(Signal, "km/h", 1)
; Y-unit 'm/s' => Event: 'km/h', all Y-values are multiplied by 3.6
```

A common application is normalizing to fundamental SI-units:

```
Signal = ConvertUnit(Signal, "SI0", 1)
; Y-unit 'kV' => Event: 'V', all Y-values are multiplied by 1000
; Y-unit 'min' => Event: 's', all Y-values are multiplied by 60
; Y-unit '°C' => Event: 'K', 273.16 is added to all Y-values
; Y-unit 'V/A' => Event: 'Ohm', Y-values unchanged
```

See also the topic of units in the curve window.

1.1.3.2 Measurements

New function for working with measurements offer an alternative to manual selection of measurements/channels in the Variables list/Measurement view.

Thus, for instance, it is now possible for many measurements to be selected in succession by means of a sequence and incorporated in a report.

<code>MeasNames?</code>	Finds existing measurements
<code>MeasChanNames?</code>	Finds a measurement's existing channels
<code>SelMeasListSetName</code>	Sets an entry in the Data Selector's Measurement list
<code>SelChanListSetName</code>	Sets an entry in the Data Selector's channel list
<code>SelListControl</code>	Various Data Selector actions (emptying, updating)

Further, it is now possible to specify a measurement association immediately when creating a new variable (by assignment):

```
Developer@Measurement27 = "John Doe"
```

This principle is also applicable in combination with specifying a default measurement by means of `SelUseMeasurement()`. The measurement name then simply remains empty:

```
SelUseMeasurement(1)
MeanOfLevel@ = Mean( Level )
```

If the first selected measurement in the Data Selector has the name "Measurement27" and contains a channel "Level", then the previous notation is equivalent to:

```
MeanOfLevel@Measurementg27 = Mean( Level@Measurement27 )
```

1.1.3.3 Measurement uncertainty

The group 8> Statistics has been supplemented with functions for determining the measurement uncertainty.

These features are not available in the Standard Edition!

`UncertaintySet(Variable, TxPropName, Value)`

Sets a user-defined property in the field of measurement uncertainty.

`UncertaintyGet(Variable, TxPropName) -> Value`

Queries a user-defined property in the field of measurement uncertainty.

`UncertaintyModify(Variable) -> Result`

Adds noise to a signal for the purpose of determining the measurement uncertainty according to the Monte-Carlo method.

`UncertaintyCalc(Variable [, coverage probability] [, Reserved] [, Extended1] [, Extended2] [, Extended3] [, Extended4])`

Determines the measurement uncertainty by the Monte-Carlo method.

`UncertaintySnapshot(Variable [, Option])`

Collection of trial variants for a variable created by the Monte-Carlo method within an UNCERTAINTY_LOOP.

UNCERTAINTY_LOOP Trial count EwInit

Loop for determining the measurement uncertainty results from an algorithm using the Monte-Carlo method (MCM).

1.1.3.4 FOR, FOREACH

For the iteration variable, it is now also permitted to use the notation "Groupname:ChannelName".

1.1.3.5 Comparison operators, logic

New functions in group 20 for bit-wise operations:

BitAND Bit-wise AND operation

BitOr Bit-wise OR operation

BitNot Bit-wise inversion

BitShift Shifting of bit pattern

BitSet Setting of a bit

BitGet Querying of a bit

1.1.3.6 Spectral Kit

Savitzky-Golay filtering for smoothing signals has been added.

1.1.3.7 Class-counting Kit

Calculation of quantiles of a distribution given in histogram form.

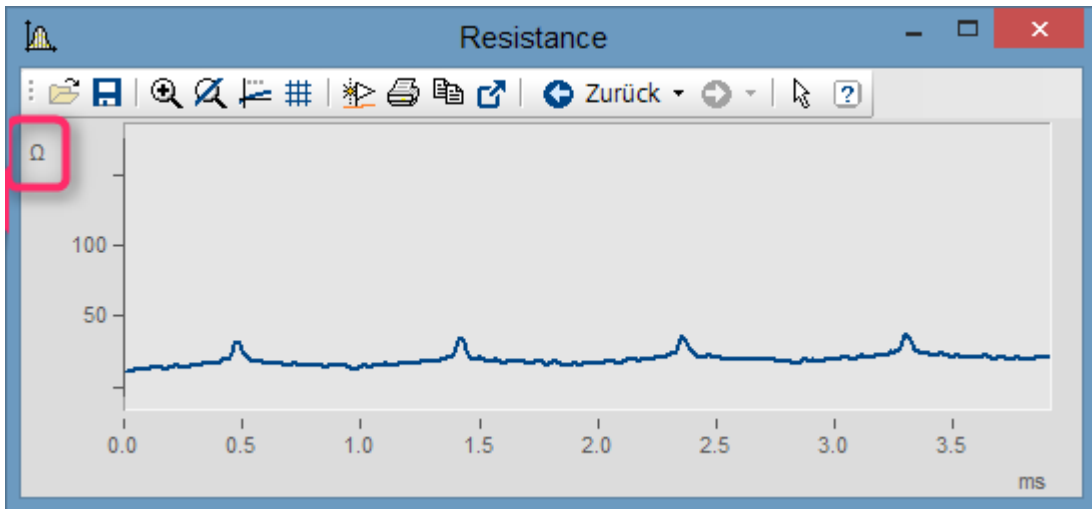
1.1.3.8 Curve Window Kit

Various properties for CwAxisSet, CwLineSet, CwMarkerSet, CwGlobalSet have been added.

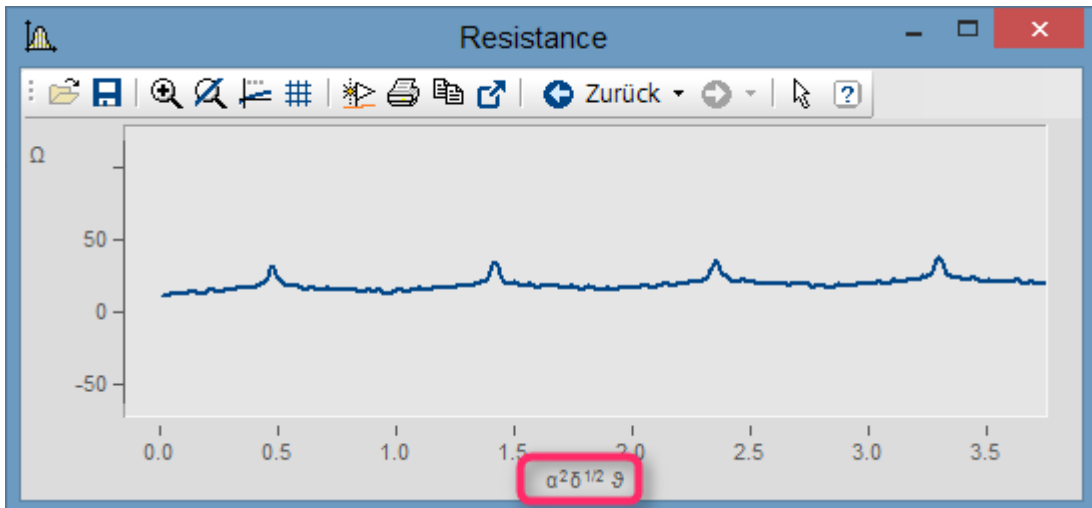
1.1.4 Curve Window

1.1.4.1 Units

- Units with Greek letters:



Ohm as a unit in the curve window



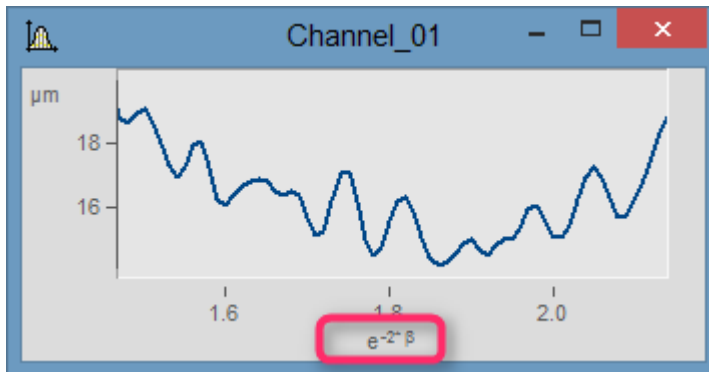
Greek letters by means of placeholders:
 $\langle g*a \rangle^2 \langle g*d \rangle^{1/2} \langle g*ta \rangle$

The screenshot shows the 'Axes' configuration dialog box with the 'Text' tab selected. The 'Axes List' on the left shows 'x1', '1', 'y1', and 'Resistance'. The 'Options description' is 'Fixed text', 'Font size in points' is 'Auto', 'Color' is 'auto', and 'Placement' is 'Auto'. The 'Text' field contains the placeholder $\langle g*a \rangle^2 \langle g*d \rangle^{1/2} \langle g*ta \rangle$.

Greek letters by means of placeholders:
 $\langle g*a \rangle^2 \langle g*d \rangle^{1/2} \langle g*ta \rangle$

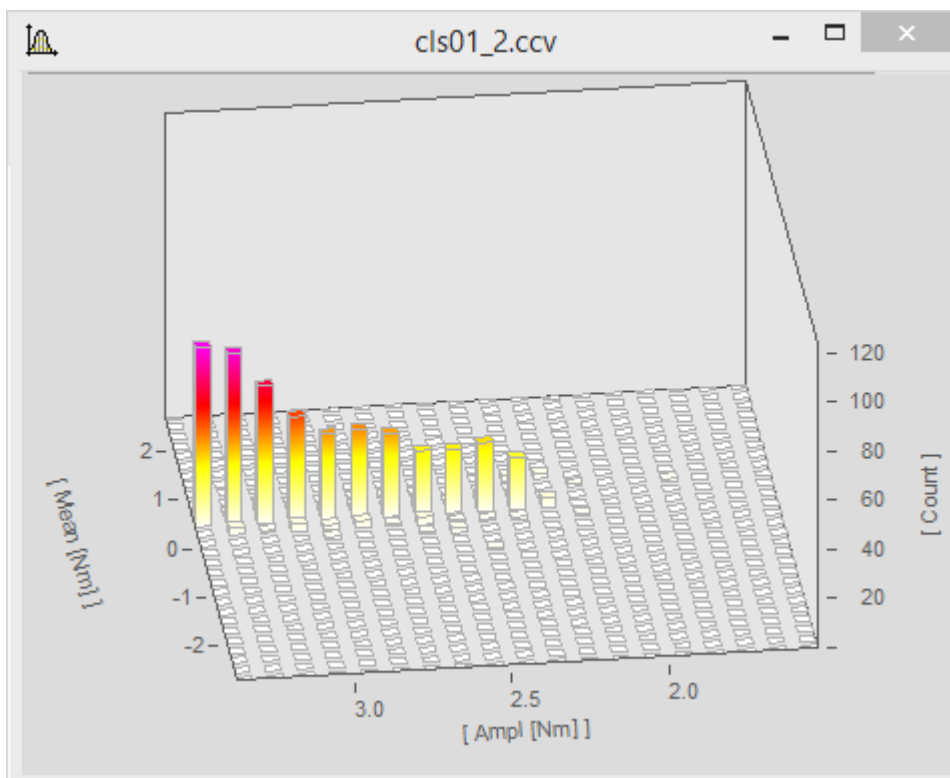
Platzhalter, Unicode, Bedeutung

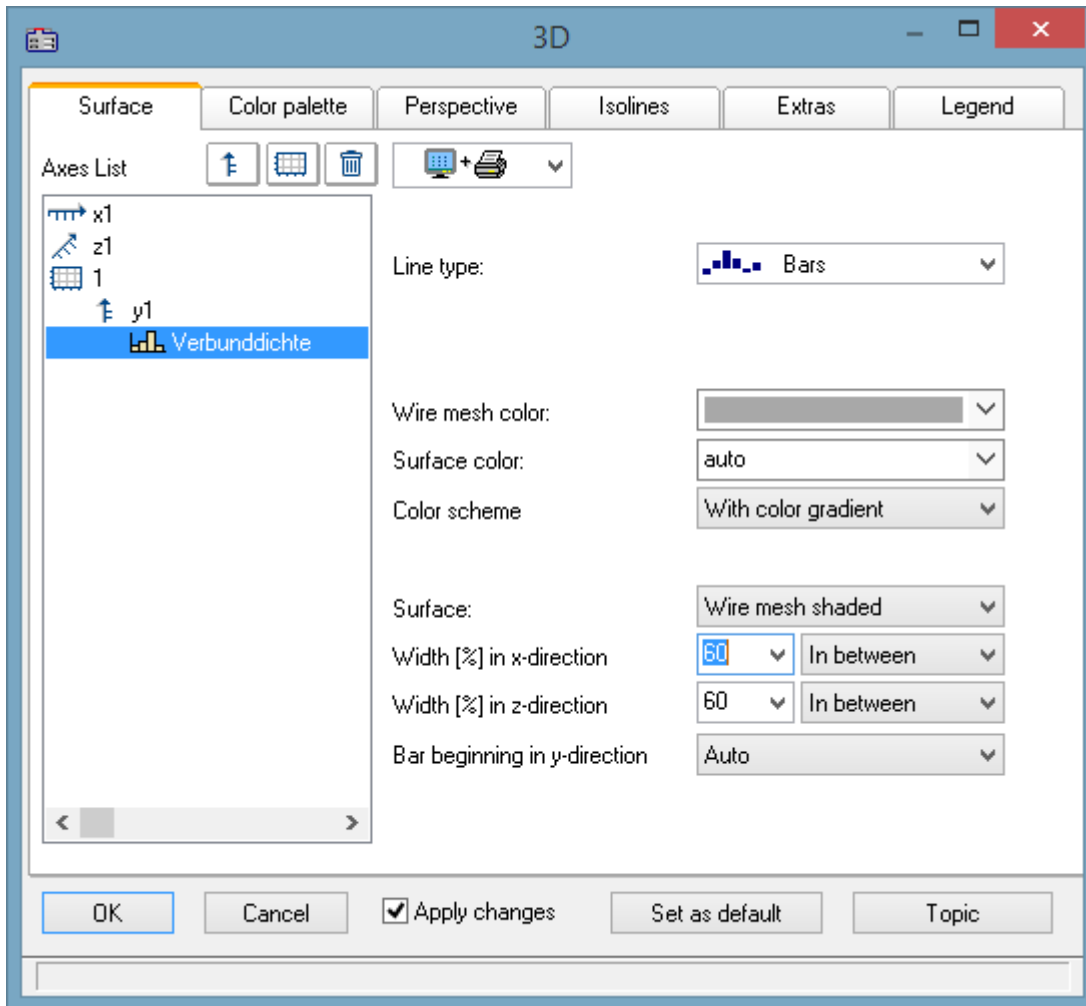
'a': 0x3b1 // Alpha	'm': 0x3bc // My
'b': 0x3b2 // Beta	'n': 0x3bd // Ny
'g': 0x3b3 // Gamma	'x': 0x3be // Xi
'G': 0x393 // Gamma groß	'X': 0x39e // Xi groß
'd': 0x3b4 // Delta	'p': 0x3c0 // Pi
'D': 0x394 // Delta groß	'ph': 0x3c6 // Phi
'e': 0x3b5 // Epsilon	'ps': 0x3c8 // Psi
'et': 0x3b7 // Eta	'P': 0x3a0 // Pi groß
'z': 0x3b6 // Zeta	'Ph': 0x3a6 // Phi groß
'th': 0x3b8 // theta	'Ps': 0x3a8 // Psi groß
'ta': 0x3d1 // theta, (gewohnte)	'r': 0x3c1 // Rho
't': 0x3c4 Schreibweise	's': 0x3c3 // Sigma
'Th': 0x398 // tau	'S': 0x3a3 // Sigma groß
'k': 0x3ba // Theta groß	'ch': 0x3c7 // Chi
'l': 0x3bb // Kappa	'Ch': 0x3a7 // Chi groß
'L': 0x39b // Lambda	'o': 0x3c9 // Omega
// Lambda groß	'O': 0x3a9 // Omega groß

Exponent of placeholder in parentheses: $e^{-2 \cdot g \cdot b}$

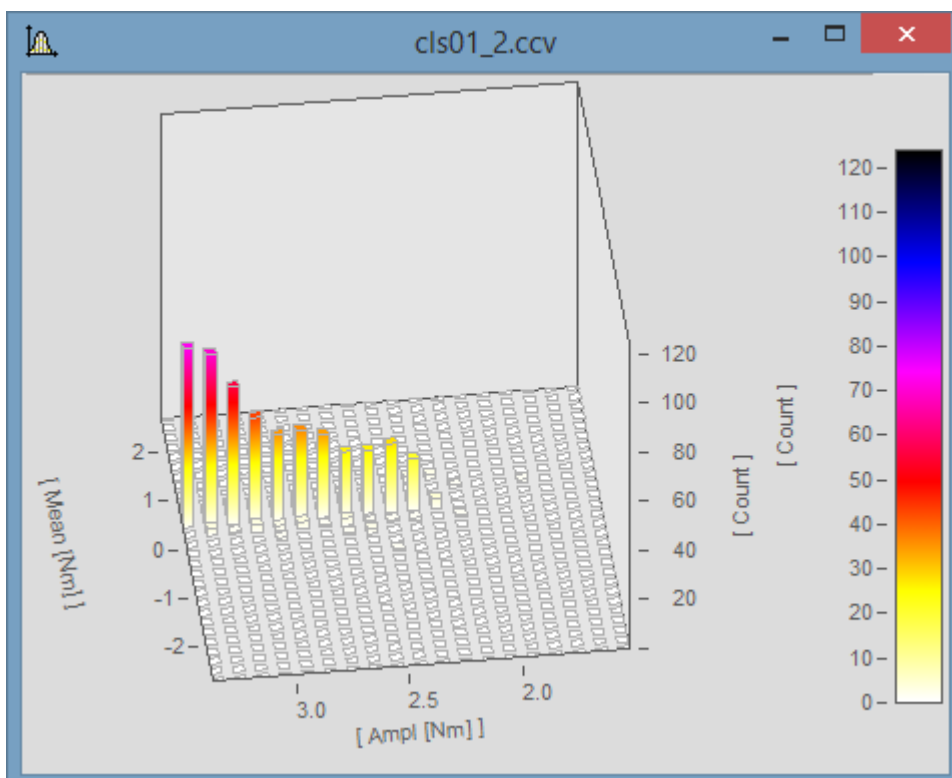
- Furthermore, the display of orders of magnitudes of units has been improved. If the unit is mm, for example, then in the curve window, the units m and μm may be displayed. Absurd unit specifications are avoided.

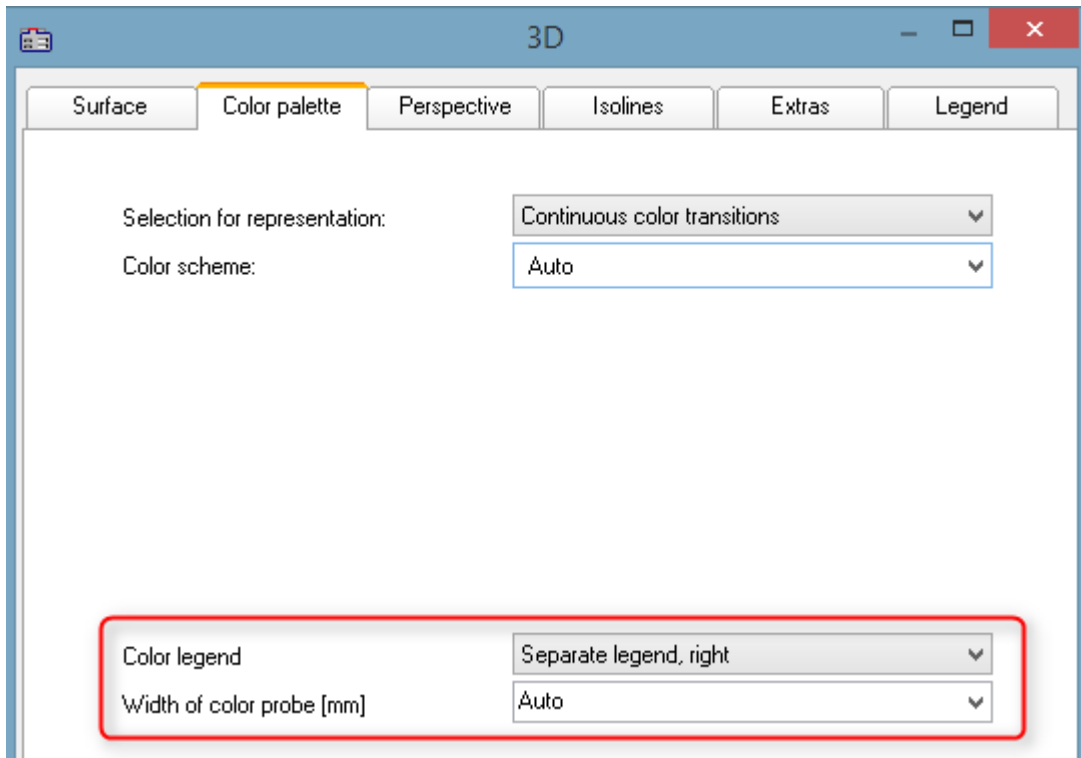
1.1.4.2 3D bars





1.1.4.3 Color legend for 3D

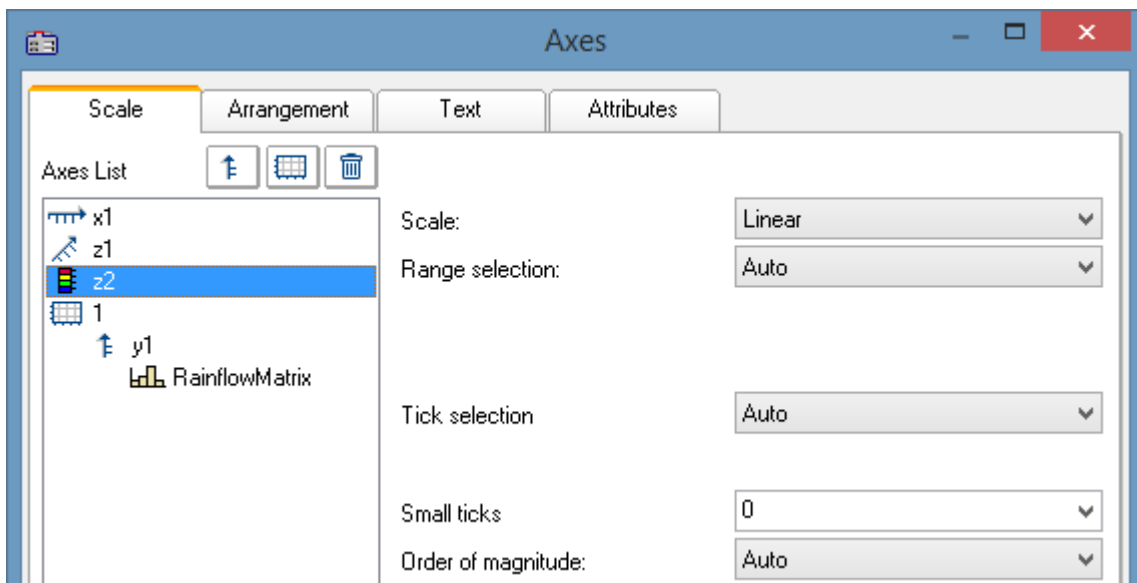




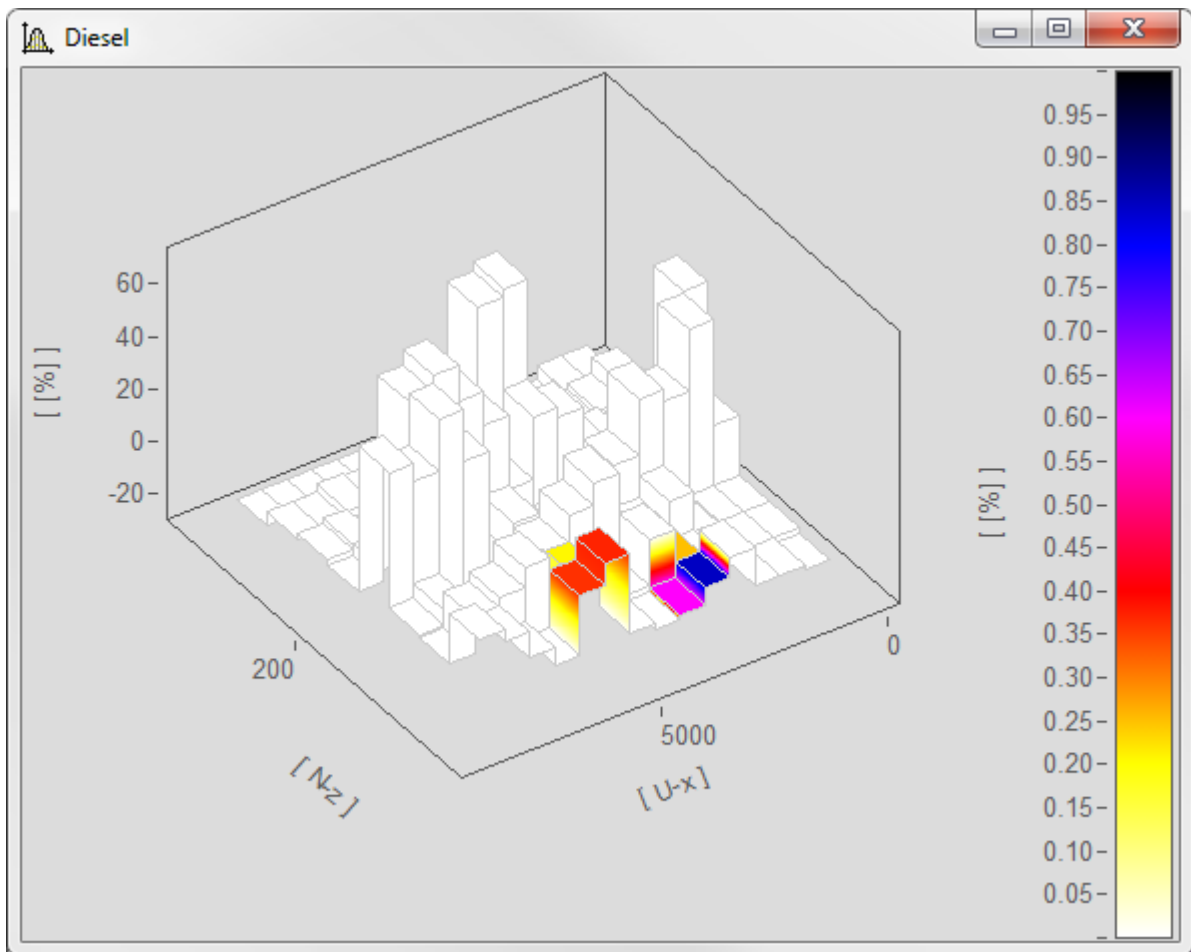
With a separate color legend, the color is specified independently of the displayed y-range.

The color legend can also adhere to the y-axis. Then the y-coordinate always corresponds to the color.

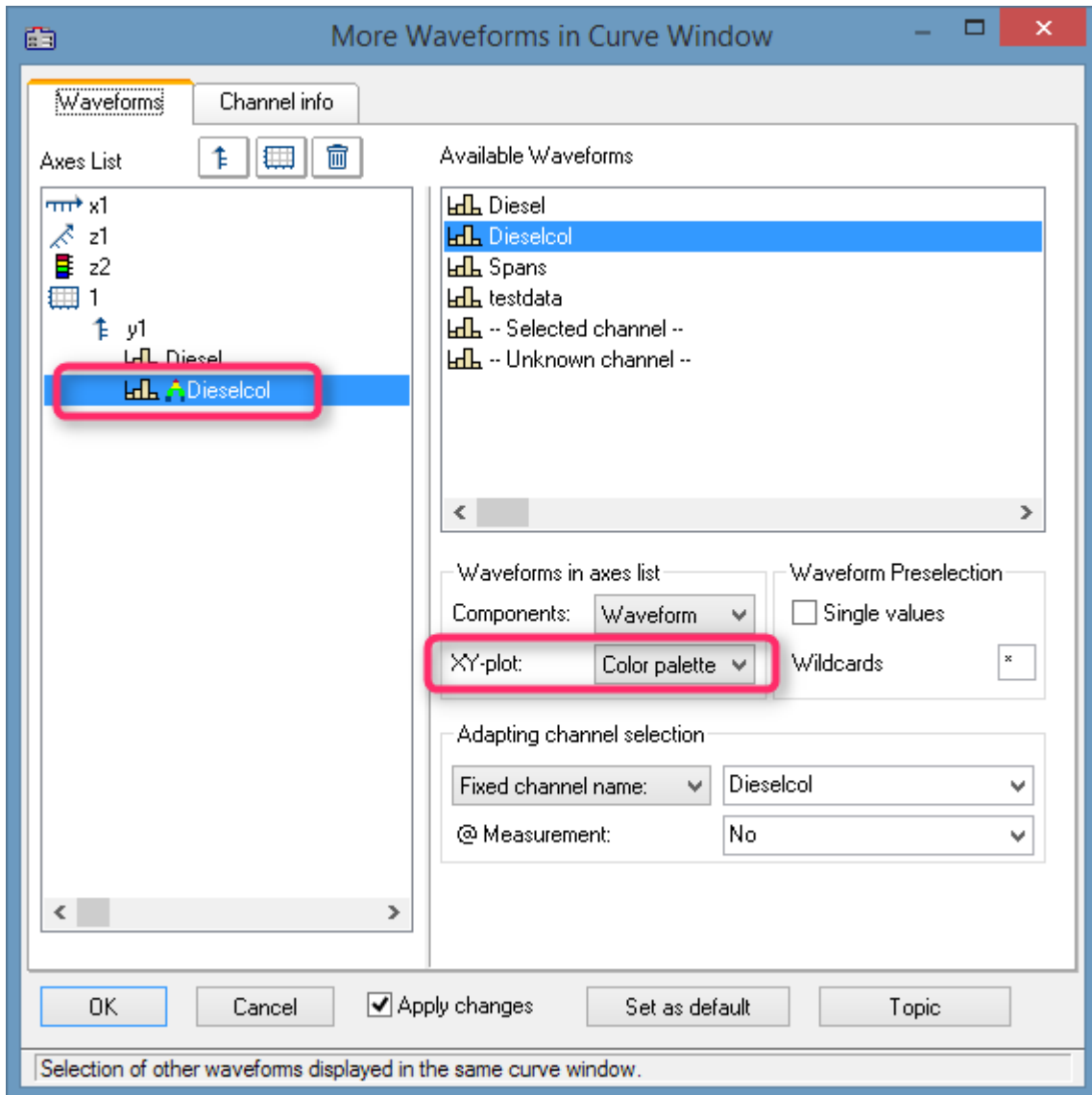
Color axis settings:



1.1.4.4 4D (3D + color)



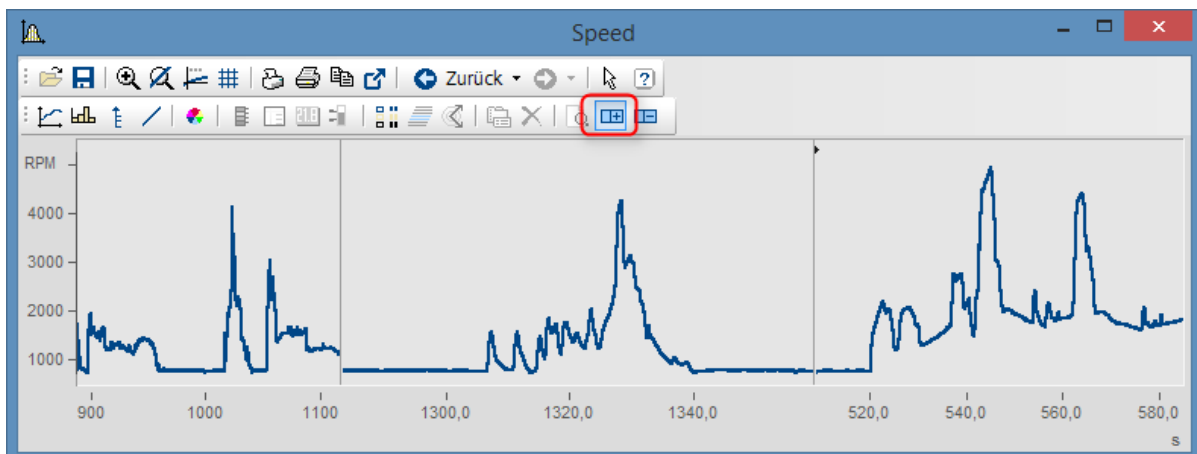
Selection of the color-determining channel (exactly same structure as channel for 3D display):



The color legend must be selected.

1.1.4.5 Split-Mode

In the *Construction* toolbox, the new Split mode has been introduced. With it, a curve window can be subdivided an arbitrary amount of times in order to focus on different regions of a signal.



1.1.4.6 Selection of measurement uncertainty

The curve window dialog *Configuration / Lines.. / Extras* has been supplemented with the entry *Selection of measurement uncertainty*.

The available options are:

- *Expanded uncertainty*: Product of a combined standard measurement uncertainty and a factor larger than the number one.
- *Standard uncertainty*: Uncertainty of the result of a measurement expressed as a standard deviation.

1.1.4.7 Channel info

The curve window dialog *Configuration / More channels* now also displays the user-defined properties on the page *Channel info*.

1.1.4.8 Presettings

The function "*Movement by means of mouse*" can now be suppressed.

1.1.5 Panel**1.1.5.1 New widgets****Datagrid**

This widget serves to display data in tabular form. In contrast to the cell-oriented widget "*Table*", this is optimized for column-wise display of data and also allows values to be changed.

Time	Channel1	Channel2
15:59:05,00	-0.0803828	-0.25438
15:59:05,09	-0.0795863	0.0349026
15:59:05,20	0.167201	0.0258671
15:59:05,29	0.535072	0.165248
15:59:05,40	0.654936	0.303725
15:59:05,50	0.786296	0.440971
15:59:05,59	0.672474	0.576611
15:59:05,70	0.929553	0.859233
15:59:05,79	1.08243	0.692321
15:59:05,90	1.20453	1.11852
15:59:06,00	1.23624	1.0945
15:59:06,00	1.67982	1.21681

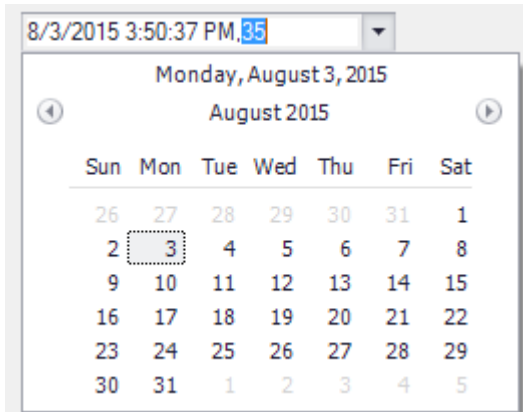
Slider

Graphical element for display/selection of a value from within a defined value range.



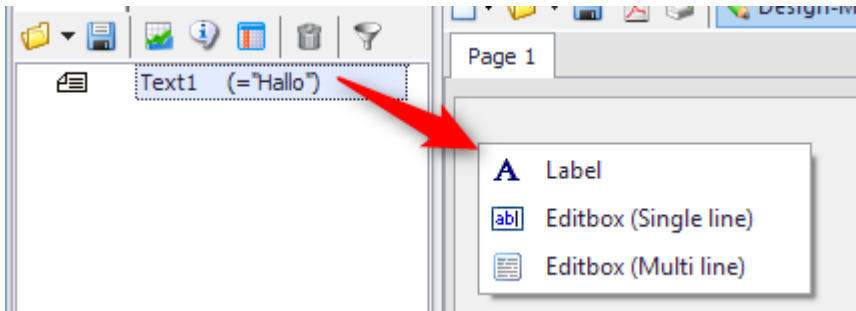
Date Time

Widget for optimized display and entry of date and time.



1.1.5.2 Drag&Drop for text variables

When applying Drag&Drop to move text variables onto a Panel in Design mode, a selection menu of available widgets (labels, input boxes, etc.) is now displayed.



1.1.5.3 Panel remote control

- New event for input boxes - "*Changed*". The condition is that the widget be linked with a variable.

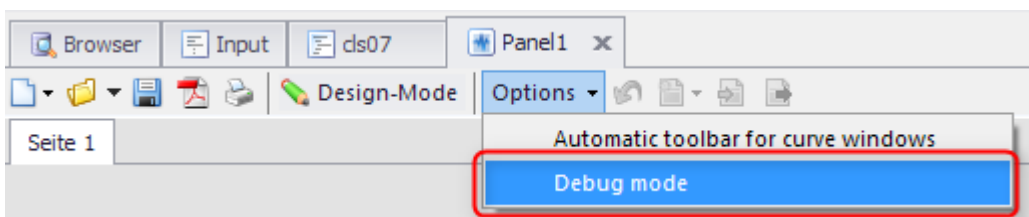
The event "*Changed*" is triggered if the value of the linked variable has changed.

- For the functions `PnTableSetColumn()` and `PnTableSetRow()`, text arrays are now also allowed as parameters.

1.1.5.4 Debugging Panel-event sequences

A Debug mode for debugging Panel event sequences has been added.

For the process of developing complex panels, this facility makes it easier to find errors. As with "normal" sequences, it is possible to define breakpoints at which execution of the sequence is interrupted. The sequence is displayed in the Editor and can be resumed step-by-step starting at the current command line. The same applies when the sequence run encounters a (syntax-) error.



1.1.6 Variable list

1.1.6.1 Filtering for the Variables list

"Inverse" filter condition added.

With a preceding "!", the pattern can be inverted.

Example:

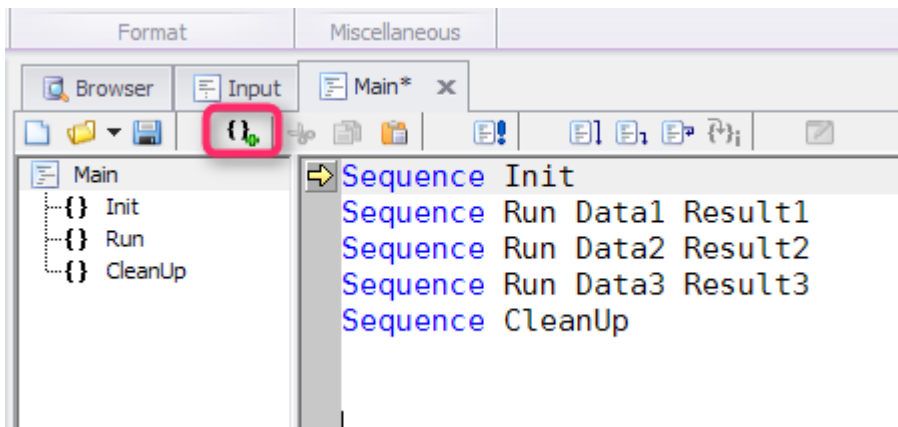
"!a*" Displays all variables which DO NOT begin with the letter "a".

"!sp*" Displays all variables which DO NOT begin with the letter "sp".

1.1.7 Editor

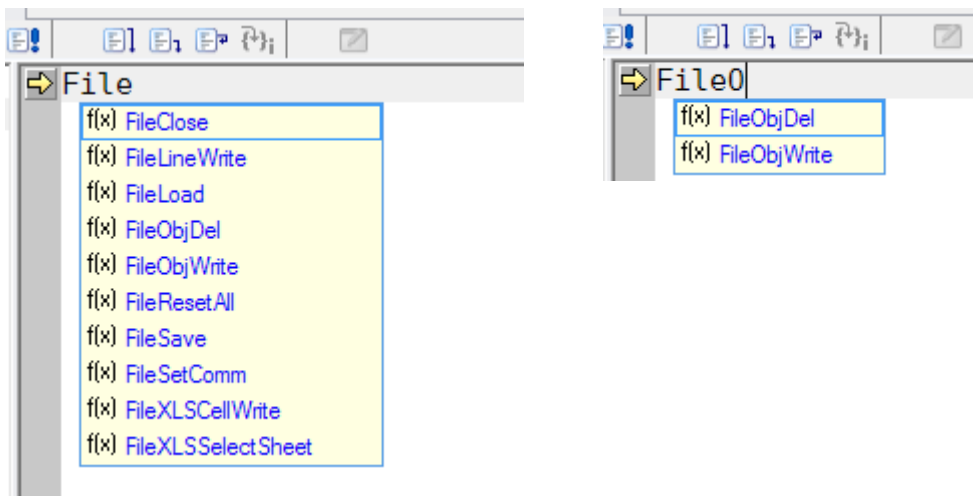
1.1.7.1 Editor with sub-sequences

Sub-sequences can now be directly embedded into the calling sequence and no longer need to be saved in a separate sequence file.



1.1.7.2 Autocomplete

When entering sequence code, a selection list is provided to the user, which contains suggestions representing sensible completions of the user's entry thus far. For example, these could be a function name, a variable's name, or lists of function parameters.

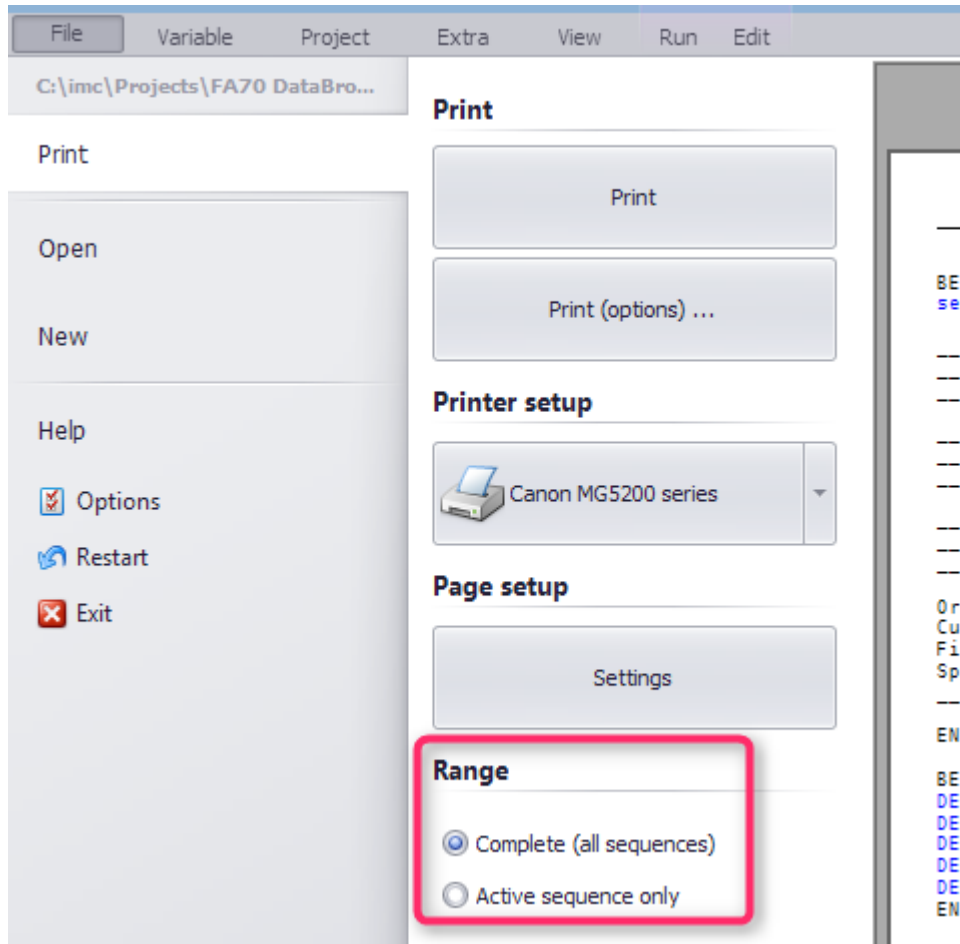


1.1.7.3 Text as HTML text

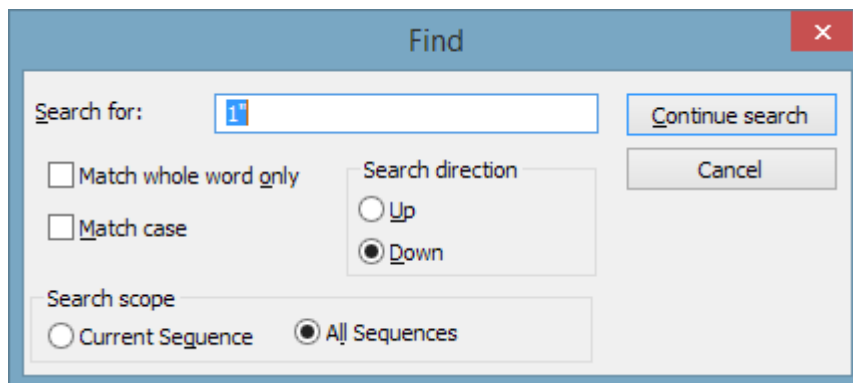
Copying to the Clipboard can now additionally be performed in HTML-format. Thus, the sequence text can now be copied from text processing programs while maintaining the color-coding for syntax.

1.1.7.4 Printing

With Panels and dialogs, it is now possible to print all the associated (event-) sequences simultaneously.

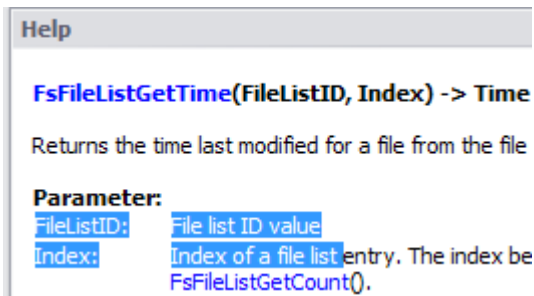


As well, global find/replace of text across all partial sequences is possible.



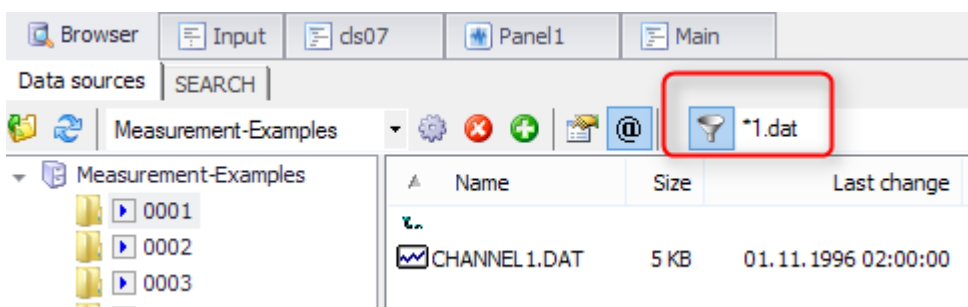
1.1.7.5 Help window

Texts in the help window can now be arbitrarily selected and copied.



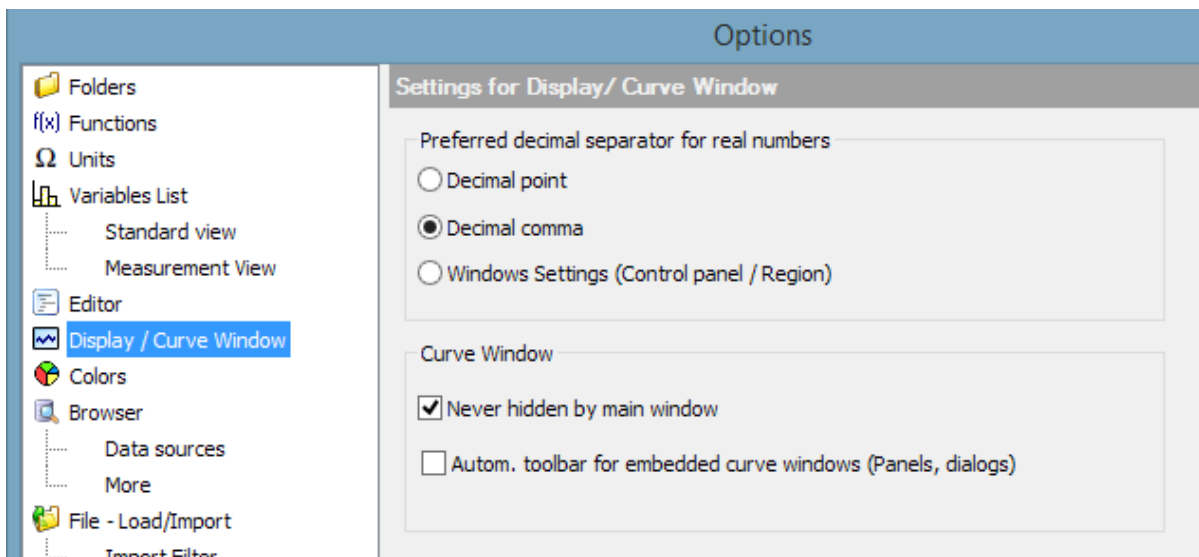
1.1.8 Data sources-Browser

A filter governing the display of data sources has been added.



1.1.9 Display / Curve window

Optionally, real numbers can now be displayed with a decimal comma (curve window, Panel, Report Generator).



1.1.10 Import/Export

1.1.10.1 Excel 2016

Excel-linkage has been extended to the EXCEL 2016 version.