

## WinDarab v7.07.036 15.04.2020

### Min/Max/Average calculation refactored (ID2742)

- The new option to have NoValue values in math channels caused an error when calculating the average value over a channel range.  
While fixing the issue, we decided to refactor the calculation of statistic values and to standardize how statistic values are calculated.

- Before the following WinDarab functions used a simple calculation:

- Math functions like ave(), lapAve(), SegmentAve() ...
- Min/Max window
- Outing report

Thereby the “Quick Info” window in the oscilloscope used a weighted calculation

- Now, all functions weight the statistic values by TIME AND respect the signal kind of the input channel (Analoge or other/Digital)

The input value for the formulas is

- ➔ For DIGITAL channels:  $\text{input}(n) = \text{channel}(n)$
- ➔ For ANALOGE channels:  $\text{input}(n) = (\text{channel}(n) + \text{channel}(n+1)) / 2$

The statistic functions are using these formulas:

- ➔ Integral:  $\text{sum}(\text{input}(n) * dt)$
- ➔ Integral2:  $\text{sum}(\text{input}(n) * dt) - \text{channel}(0) * \text{sum}(dt)$
- ➔ Average:  $\text{sum}(\text{input}(n) * dt) / \text{sum}(dt)$

- The “Quick Info” has some specialized features which are not used/supported by the other WinDarab features:
  - WinDarab adds “virtual” samples at the left/right boundary of the evaluation range.  
In case of a digital channel the sample right before the boundary is used.  
In case of an analoge channel while the values are interpolated.
  - This includes the fact that the value at the right boundary is INCLUDED and evaluated. If you set the right boundary exactly on a (digital) sample, the min/max values include the value. Average and integral won't change because the value is weighted with a factor of zero -> the value doesn't have any effect on average and intergral!
  - The user can optionally switch to weight “by x-Axis”.  
Instead of using “dt” in the above formulas, WinDarab uses dt (x-axis is time) or ds (xAxis is distance).
- The other WinDarab features (math functions, outing report, min/max window, ...)
  - Evaluate only the samples inside of the evaluation range.  
E.g. LapMax() evaluated the range of Lap.TimeBegin (include) to Lap.TimeEnd (exclude) without virtual samples at the boundaries. Evaluation starts with the first sample AT/BEHIND TimeBegin and Ends with the last sample BEFORE TimeEnd.
  - If you examine a channel which is reset at the laptrigger (measurement channels are normally NOT measured exactly at the laptrigger!) and you calculate a LapMin over the data, you probably never see an exact zero value!
  - The same is true in case you evaluate a math channel – except you advise the math channel to add additional timestamps at the laptrigger (we added such an option recently)
- Please respect the weighting if you compare statistic results calculated externally (e.g after exporting/evaluating) sample values outside of WinDarab!

### WinDarab cannot open a file – Why ??? (ID2734)

- Currently, if a file cannot be opened by WinDarab, WinDarab simply does nothing – even not

showing any message box. This is known issue but ...

To ease identifying the reason, we added a log entry to the log file, if a file couldn't be opened because of an unsupported file format.

#### **Oscilloscope: Changing y-axis display area wasn't applied to all attached channels (ID2733)**

- If the user reduced the size/height of a y-axis the change wasn't applied to all channels attached to the same y-axis
- The same occurred, if other y-axis settings were changed (e.g. scale factor, value mode, ...) through the y-axis context menu.

#### **Installer: A C# sample file wasn't published by the installer (ID2730)**

- A sample C# file in one of the sample projects was missing in the installer. We added the file.

#### **Setupsheet Editor supports drag and drop to sort the channel rows (ID2729)**

- The data grid supports selecting and dragging channel rows to order the channels.
- **Note:** Dragging columns to order the outings is not supported. The columns are sorted by outing number, if you close/reopen the setupsheet (alternatively if the setupsheet is reselected in the "Filemask" combo box)

#### **Setupsheet Editor crashed when deleting a car setup (ID2727)**

- WinDarab crashed, if the user clicked the "Delete car setup" twice.
- The bug affected all buttons except "Add car setup" in the setupsheet editor.

#### **Support to open WinDarab files using channel data compression (future)**

- Starting with WinDarab v7.8 WinDarab files can use a format extension allowing WinDarab/DataLoggerImport to compress channel data in WinDarab files. This reduced the WinDarab file size by ~ 70-90% of the uncompressed size.
- For future backward compatibility the support to open WinDarab files using channel data compression was already added to v7.07.036. This allows users/teams to migrate step by step to v7.08.

#### **C/C++ header and TLB files added for programming COM clients**

- The folder Samples/Include contains the header and TLB files with the current COM interface definition for COMApi and BMS2API client applications written in C/C++.
- **Note:** The Microsoft Visual Studio compiler support the creation of COM interface and class definitions by using the #Import preprocessor command to import TLB files

```
#Import "WinDarab.tlb"
```

Other compilers (or C-compilers) probably don't support TLB files. Here the programmer can include the provided header files instead.

## **WinDarab v7.07.035 24.03.2020**

#### **X-y-Plot: The ribbon buttons to select the plot symbols "Cross" and "Circle" were swapped (ID2726)**

- The buttons in the ribbon of an x-y-plot to select the symbol styles "Cross" and "Circle" were swapped – the settings dialog was correct.

#### **WinDarab file format v2: Issues when extracting/exporting to a new file (ID2725)**

- If a measurement file using the WinDarab File Format v2 is extracted/exported, the exported timelines were accelerated by a factor of 10. E.g. exporting a time range of 100 seconds resulted in files of 10 seconds with a sampling period which is 10x of the original.
- The issue was caused by mixing the same variable name with different meanings: One held the

timestamp resolution in Hz, the other in 100ns. In common the ECUs use a timestamp resolution of 10000kHz or 1000x 100ns. Mixing the values without unit translation lead to the time factor issue of 10x.

#### **Math channels aren't updated automatically (ID2723)**

- Math channels weren't updated any more after making a change in the math function definition if the math channel is referenced by other channels.
- This bug was a side effect of the the changed in "Math channels: Auto-create filtered channels by channel name (ID2649)"

#### **WinDarab crashed when trying to open very old files (ID2722)**

- WinDarab crashed, if the user tried to open a very old file using the MUX-channel-pattern.
- **Note:** The MUX-channel-pattern was used by ECUs in the early years of 2000.

#### **Math channels: New option "Include relevant timestamps" added (ID2721)**

- Some math channels work better if relevant timestamps are included to the math function. E.g. functions like "LapSigma" reset their result if a laptrigger is passed. Nevertheless the function is probably not calculated for the laptrigger time, if this timestamp isn't included in the selected timeline.  
This new option ensures that the timestamps of the laptriggers are included in the math channels timeline and the math channel has a value at these positions.
- **Note:** The used math functions in a math formula control the information, if relevant timestamps exist and which they are (an example is LapSigma which adds the timestamps of the laptriggers to the math channel).

#### **Calculation of lap statistics changed (ID2721)**

- The functions to calculate lap statistics were changed. They don't include interpolated values at the lap boundaries any more. WinDarab only evaluates the recorded samples within the lap boundaries ( $LapTimeBegin \leq time < Lap.TimeEnd$ )

#### **Oscilloscope: Symbol for hidden channels changed (ID2720)**

- If a channel is hidden in the oscilloscope, the oscilloscope is drawing an outlined orb using the file color without a file-color-derived filling.

#### **Oscilloscope quick info tool windows didn't honor file shifting (ID2718)**

- The quick info tool windows didn't honor the current file shifting if multiple overlays are shown in thre oscilloscope. Instead the absolute time range of the active overlay was used for all overlays in the quick info tool ...

#### **x-y-Plot crashed after an overlay was removed (ID2716)**

- x-y-Plot crashed when deleting temporary channels/conditions after an overlay was removed. The sequence how overlays are removed from anaylsis windows was changed. In addition channel notification messages are processed with a small delay to prevent interference with the internal file replacement procedure.

#### **MDF4: Several fixed (ID2714)**

- The (internal) time channels are removed from the Channels pane (it was too easy to get confused because of their naming in the MDF4)
- Correct support of special characters in the channel name and description: WinDarab uses UTF8 encoding to read the string from the MDF4 file.

#### **WinDarab crashed when assigning a new file to an overlay (ID2713)**

- The crash occurred because of a (still) unknown reason it could occur that an oscilloscope channel lost its binding to an overlay. We fixed the crash, but still it's not known how the binding can be lost
- A diagnostic log entry was added to the logfile if this issue happens and is observed by an user.

### **WinDarab crashed when closing an oscilloscope/WinDarab (ID2713)**

- WinDarab crashed if an oscilloscope (or WinDarab) is closed while the quick info tool window is opened. We weren't able to reproduce the issue, but think to successfully prevent the crash in future.

### **Oscilloscope: Cloning of oscilloscope channel when adding a new overlay fixed (ID2712)**

- If a new overlay is added in the File Explorer pane, all oscilloscope should clone their channels for the new overlay and place them directly beside the existing channels (to group the channels). But this worked only, if an oscilloscope was visible once after loading the desktop configuration. If the oscilloscope was never visible, the cloned channels were added at the bottom of each area.

### **x-y-Plot drag and drop behavior changed (ID2711)**

- If the user dragged a channel on the x-/y-/z-axis of a x-y-plot, WinDarab reset the axis boundaries of the touched plot layer to "<auto>". This could cause behavior issues, if the axis is shared between multiple plot layers, because the limit range wasn't changed for all plot layers.
- Now the axis boundaries are reset to "<auto>" only, if the axis isn't shared between multiple plot layers. And not changed, if an axis is shared.

### **x-y-Plot: Behavior of data grid at the top changed**

- The info data grid at the top of a x-y-plot view didn't handle the cursor keys. The cursor keys were always handled by the plot layer to move the cursor cross. Now the cursor keys are handled by control holding the input focus (data grid or plot control)

### **WinDarab.Net C# wrapper: Some properties weren't exposed in the wrapper DLL but defined in the COM API reference (ID2709)**

- Following properties weren't exposed by the C# classes in WinDarab.Net:
  - Channel.Application.
  - Channel.TimeLast

### **Fix: Alias channels didn't pass through the options to set/change the user defined offset/gain**

- Affected functions were "Make zero" and "User defined offset/gain"

## **WinDarab v7.07.034**

**04.03.2020**

### **Math editor: The new math editor didn't show/save the entered unit (ID2705)**

- The new math editor didn't show/save the unit entered by the user. The field was always left empty (or unchanged for existing math channels)

### **Math channel: Support of "Make Zero" and "Set user-defined offset/gain" reactivated (ID2704)**

- In a source merge process a couple of lines found their way back into the source code. This code caused WinDarab to disable the commands "Make zero" and "Set user-defined offset/gain" for math channels.

### **DataLoggerImport failed to import data, if ethernet connection is lost (ID2702)**

- If DataLoggerImport has downloaded all files from the Datalogger (via FTP) and the user disconnects the ethernet connection while DataLoggerImport is still converting data, it could happen that some files were cleaned up too early and the final files couldn't be created.

### **FlashApiService stopped to accept client connections**

- If multiple applications are using the FlashApiService (e.g. RaceCon and WinDarab) ping the DataLogger) it could happen that FlashApiService stopped to accept client connection.

### **Oscilloscope/Quickinfo tool window: The shown extrema values weren't correct**

- The quick info tool window didn't show the correct min/max values for the evaluated range or the

data position wasn't correct. Instead the sample beside the actual min/max value was shown.

## WinDarab v7.07.033 28.02.2020

### Oscilloscope: Behavior of zoom and scroll changed (ID2694)

- We changed the zoom and scroll behavior in WinDarab.
- Setting/scrolling the cursor with a mouse click is restricted to the current data area (data range of all overlays including their file shifting).
- Nevertheless, if the visible range is larger than actually necessary this additional range is also allowed for scrolling/setting the mouse cursor.
- A larger visible range can be achieved, if another (smaller) file is attached to an overlay or by manually setting the x-axis range (right click on the x-axis)
- WinDarab tries not to change the visible range unless the user requests it (e.g. by clicking "zoom lap"). In other words if you close a large file zoomed to "all" (e.g. 24h recording) and then open a single lap file, WinDarab still shows 24h on the x-axis.

We didn't change the behavior at this point because we don't want a resetted telemetry stream (e.g. with a length of 1 sec) to change the zoom, if the user has zoomed a certain working range (e.g. a typically laptime of about 120sec).

### File explorer pane: Double click to zoom lap is back (ID2694)

- Since some Microsoft update the Tree Control didn't pass double clicks in front of the label to the application. But these double clicks were used by WinDarab to select and zoom the lap. Now WinDarab is catching the double-click before it's passed to the Tree Control and decides if the click is still passed or directly send to the parent control/window.
- **Note:** A double click on the overlay symbol zooms aligns all selected laps and then zooms the lap of the active overlay (x-axis overlay) - not the selected overlay!

### Dist-Channel: Support of non-monotone distance source channels readed (ID2693)

- WinDarab can create a monotone increasing dist channel with the data of a non-monotone input channel (e.g. if the channel overflows or resets; e.g. lapdist channel)  
In v7.05 this was (implicitly) done by the distance channel interpolation because WinDarab required the xdist channel to be available for all global samples. With the architecture change implemented in v7.6 this feature wasn't necessary any more and was removed – without recognition of the side-effect that this interpolation also took care of non-monotone distance sources.
- Now WinDarab checks the monotony of the distance source channel and if it's not monotone increasing, a intermediate channel is calculated. Here each sample breaking the increasing monotony is replaced by an extrapolated value.

### Setupsheet channels weren't updated correctly (ID2692)

- Updating the setupsheet channels didn't work correctly for certain kinds of files:
  - Files containing 2 or more laps (e.g. outing files)
  - Files with manually set outing numbers (outing numbers not set by an outing channel)  
Typically these files are created by the user (COMApi or ASCII-Import)
- Another issue we observed is an unwanted updating of setupsheets channels, if the setupsheet definition contains the duplicated channel definitions.  
**Note:** Because the setupsheet editor in WinDarab doesn't allow duplicated names, we think that the user has edited his setupsheet file with an external tool.

### Outing Report: Faster processing of a cancellation request

- If the evaluation of an outing report is interrupted while internally a complex math channel is updated, it could take a longer some time until the cancellation request was processed. Now WinDarab publishes the cancellation token and inner functions can cancel the processing earlier.

### **Telemetry streams: Update events of telemetry streams aren't passed to WinDarab**

- If a telemetry stream isn't attached to any overlay, data reception notifications aren't passed to WinDarab anymore – it's unnecessary lost CPU time ...

### **Oscilloscope/Telemetry: Smooth scrolling available in program options**

- A long time ago we implemented a "smooth scrolling" feature in WinDarab. But it was only available through a manual entry in the "settings.ini".  
Now the setting is also available in the program options (tab "Measurement window").

## **WinDarab v7.07.032**

**27.02.2020**

### **Math channel: Several crashes caused by the latest feature improvements (ID2691)**

- Some crashes were caused because of the latest changes to math channels
  - ID2649 – auto-create filtered channels) and the math parser
  - ID2667 – parse subexpression even for math functions requiring an input channel)
- The file-based Min/Max functions crashed because the internal channel pointer wasn't initialized.
- WinDarab crashed when updating a math channel after it was parsed once with an unknown channel.
- Depending on the math channel dependency a crash could occur, if the math root folder was changed.

## **WinDarab v7.07.031**

**25.02.2020**

### **Telemetry/Math channels: Lap functions calculate too far (ID2688)**

- The evaluation range of math channels using lap function include the timestamp of the last laptrigger. This caused the lap function to calculate the result for the current lap immediately instead of at the end of the running lap.
- We fixed a small gap between the Lap.TimeBegin and Lap.TimeEnd values between consecutive laps.

### **Telemetry/Math channels: "Prediction" of value for digital channels changed (ID2688)**

- In case of FM40 telemetry WinDarab gets the channel data ordered in time. The enabled WinDarab to set the evaluation range for math channels up to the current time if only digital channels are used.  
In case of the new high-speed telemetry this assumption isn't correct anymore. Using HST only the channel values in the same recording group are ordered in time. This requires WinDarab to limit the calculation of math channels to the last sample of the channel whose newest sample is the oldest.
- Nevertheless, in certain cases, where the next timestamp of a digital input channel is already known (but not its value), the math channel can be calculated to this next timestamp.  
E.g. this is true for math channel using a lap-based function (e.g. LapAve) which is used as an input of another math channel (e.g. "<channel>-LapAve(<channel>"). The LapAve channel is always one sample behind the lap trigger channel, but as soon as the next lap trigger is received, the derived channel can be calculated up to the laptrigger too!

### **Oscilloscope: "Prediction" of value for digital channels changed (ID2688)**

- Like in the previous math channel issue, we assume a digital channels value to be valid up to the next timestamp on the timeline (if already known).
- In telemetry we can "see" the digital line of a lap statistic value immediately, if the next lap has begun.  
Lap statistic channels only have a single sample at the beginning of the lap. Without this improvement the line was drawn if the next lap has also finished (line was delay by a whole lap!).

- Now the line is visible for the whole lap immediately, if the next lap has begun.
- In case of offline files and digital channels, the last recorded value is continued horizontally up to the end of the file. Again this ensures that the last value (e.g. of a lap-based math function) is visibly drawn for the whole lap.

**Math function: WinDarab crashed, if certain math functions were used outside of math channels (ID2687)**

- Certain functions can be used in math channels only (e.g. not in flow chart formulas) but caused a crash, if the user was trying so.  
Now WinDarab displays an error message: "The math function is only supported in math channels!"
- Affected math function: dt, ds, HOLD and delta(...)

**WinDarab v7.07.030  
21.02.2020**

**COMApi/BMS2API: VirtualChannel.DataFile is immediately set (ID2686)**

- If a COM client creates a new virtual channel by calling `DataFile.NewVirtualChannel()`, the `DataFile` property of the returned `VirtualChannel` object is initialized with the `DataFile` and CANNOT be changed any more.
- In other words: Programmer don't have to assign something to `VirtualChannel.DataFile = ...` explicitly.  
For compatibility of existing code an assignment of the SAME `DataFile` doesn't cause an error. But trying to assign a different `DataFile` will cause a COM error!

**Instrument panel: The bar gauge caused a crash (ID2684)**

- The bar gauge in an instrument panel caused WinDarab to crash. Sorry ...

**COMApi/BMS2Api reference updated (ID2680)**

- We have updated the COMApi/BMS2API reference. Especially we tagged all methods, properties and classes which are available in COMApi only with "(COMApi only)"!

**Math channels: The "calculation mode" in the math editor didn't match the math settings (ID2675)**

- When implementing the new math editor dialog we didn't recognize that the dialog sorted the items in the "calculation mode" combobox by name.  
This led to a mismatch between the math channel settings and the UI. The "calculation mode" didn't do what the UI was showing (nevertheless the UI was wrong).  
Now the dialog shows the proper setup of the math channel.
- Note:** Especially the UI setting "of the slowest channel" didn't do anything, because internally it was "use the following channel". But because there was no channel selected, nothing was calculated!
- Note:** Users should check the settings of all math channel you touched with the new math editor!

**Math parser for statistic functions modified (ID2667)**

- Statistic functions (e.g. "Ave", "Min", "Max", "LapAve", "LapMin", ...) expect a channel as input. If the user tried to enter a subexpression, the math editor marked the text behind the first word as erroneous. This let the user think, that something is wrong on the subexpression – but not that a channel name is expected.
- Now, the parser reads the whole sub expression and marks the whole subexpression, if it's something different than a channel reference.

**Math channels: Auto-create filtered channels by channel name (ID2649)**

- If a channel name is a valid channel name with an applied math filter, the math channel creates the required intermediate math channel automatically:
  - A math filter "ave-10" with this expression is defined:

```
average({default};10)
```

2. You create a math channel with:  
 $\{vwheel\_fl-Ave-10\} + \{vwheel\_fr-Ave-10\} / 2$

In previous version the math channel only worked, if you've added the channel  $\{vwheel\_fl-Ave-10\}$  and  $\{vwheel\_fr-Ave-10\}$  somewhere in an oscilloscope.

Now the parser auto-creates these channels if they're used nowhere – and the math channel works!

## WinDarab v7.07.029 14.02.2020

### WinDarab could crashed because of an uninitialized variable (ID2674)

- With introduction of virtual channels creates by flow chart we added a new member variable to virtual channels but missed to initialize this variable properly.  
Because an uninitialized variable can contain any value, it could happen that WinDarab crashed sometimes.

### Formula editor: Declared variables weren't offered in the intellisense selection (ID2672)

- If a formula declares local/global variables (var/global keyword), the intellisense list didn't offer these variables names if he continues to type the expression.

### Formula editor: Handling return key changed (ID2671)

- If the user presses the <Return> key in the math editor dialog, the default command "Save as" is never executed.
- Instead the <Return> does some action depending on the current focus:
  - Expression textbox: A new line is inserted (as before)
  - Channels list: The selected channel is inserted at the current position in the expression textbox.
  - Functions listbox: The selected function is inserted at the current position in the expression textbox.
- After adding a selected element to the expression textbox, the input focus is always set to the expression textbox control.

### COMApi: Files cannot be closed, if they are used in a linked file at the same time (ID2647)

- As long as a file is used in a linked file, the member file cannot be "really" closed.
  - If a user closes such a file in the file explorer pane, the file is removed from the list to surrogate the user that the file was closed.
  - But if the same action is executed by a COMApi client tool, nothing happened!  
Now, the file is also removed from the file explorer pane, if the DataFile.Close is called by a COMApi client.

### Flow Chart: Issue with not-continous data streams (like highspeed telemetry)

- A missbehavior was found especially using HST-streams because HST data blocks arrive in chunks of multiple samples. Therefore the timestamps received in the are not continous over all recording blocks.  
Because of this the evaluation of online data is done up to the oldest timestamp of the last timestamp of all timelines used by the links leaving the current flow object.

## WinDarab v7.07.028 02.02.2020

### Flow Chart: Bug fixing (ID2629)

- The settings dialog of a table denied "Ok" if a virtual channel is enabled: The check for the timeline

- source column always failed ...
- The evaluation of the <auto>-Min-Max wasn't implemented yet.

## WinDarab v7.07.027

31.01.2020

### Oscilloscope: Signal drawing in deep zoom not accurate (especially telemetry)

- Another old issue is a problem when drawing the signal in the oscilloscope. In deep zooms it could happen that lines between samples were missing or a digital signal wasn't a rectangle signal.
- Again implementing the "Flow chart creates virtual channels" feature made it much easier to reproduce the issue in the oscilloscope (together with a telemetry stream). Before it was too random and not reproducible to locate the bug ...

### WinDarab crashed, when opening a linked file (ID2666)

- WinDarab crashed, if a linked file is opened with a member file created by an data extraction and a DataLoggerImport created file. The bug was caused because a hashmap holding unique channel names was case-sensitive, while DataLoggerImport and the Data Extraction Tool used different casing for the timeline channel names.

### Change signature batch dialog: The files list wasn't populated any more (ID2662)

- The files list wasn't populated any more because the threads which should enumerate the files canceled themselves ...

### WinDarab Free locked by bad license verification (ID2661)

- If WinDarab is launched but has a bad license verification, WinDarab disappears after about 10 seconds for a couple of seconds – and probably then exits because of an invalid license.

### Flow Chart: Several fixes after deep testing the flow chart (ID2629)

- We did a deep test of the flow chart evaluation after we found many issues when implementing the "Create Virtual Channel" feature. Most of the issues were relicts of the architecture change in WinDarab in v7.6.
- Sometimes samples were skipped / not evaluated, if the exit connector of a flow object was connected to multiple target objects (accessing different input channels).
- The behavior of "NextSample" wasn't good, if its exit was connected to a non-conditional flow object (e.g. "Join" or any output object)
- The evaluation of telemetry streams works well now
- Toggling a flow on/off is working well.
- The "value" output object didn't show any values.

### Flow Chart: The data of an output table can be used to create virtual channels (ID2629)

- Columns in any output table can be enabled to create/populate a virtual channel. The virtual channel can be used in any other view of WinDarab for further analysis.
- Persistent channels are also supported for single files (not linked files and not telemetry), but only if the evaluation "Whole file" is selected for the flow chart.  
**Note:** Be aware that persistent channels are replaced each time the flow is executed!
- The name of the column is used as the name of the virtual channel and the name of the table is used as the data source.  
Nevertheless to prevent name conflicts we encourage the use of „unique names“. WinDarab will resolve channel name conflicts – nevertheless depending on the execution order of Flow Charts the name conflicts aren't always resolved the same way!

## WinDarab v7.07.026

24.01.2020

### Darab File Export: WinDarab crashed if trying to export from a WDServer created file (ID2660)

- WinDarab crashed, if the user tries to export a part of a WDServer created telemetry file into a new WinDarab file.

## WinDarab v7.07.025

24.01.2020

### Oscilloscope: For cloned channels the “Local Min/Max” could be different to the original

- If a channel is cloned in the oscilloscope (e.g. if adding a new overlay) it could happen, that the original state of the “Show local Min/Max” feature was toggled.

### COMApi/BMS2Api: DataFileExport didn’t accept TimeBegin and TimeEnd (ID2660)

- The DataFileExport class didn’t accept changes of the TimeBegin and TimeEnd properties.

## WinDarab v7.07.024

23.01.2020

### Instrument panel: Previous lap time shows current lap time (ID2659)

- The “Previous lap time” element showed the current lap time.

### COMApi/BMS2Api: Channel.Timeline returned a wrong timeline (ID2658)

- WinDarab always used the COM-wrapper class of UserChannels for all other channels (except virtual channels). Because a UserChannel doesn’t require a lock to return the correct Timeline. the call didn’t work correct for other channels.

## WinDarab v7.07.023

23.01.2020

### Telemetry: Instrument panel caused a crash, if the telemetry is reset (ID2657)

- If a telemetry stream is reset, the instrument panel crashed. Because the instrument panel shortly disconnects from the configured overlay – but tries to access the unset overlay, if the instrument panel checks, if the overlay is currently hidden.

### Channel settings dialog: Filter box / auto complete works similar to the channels pane(ID2654)

- If a single word is entered in the filter textbox, WinDarab filters all rows containing the word but selects the row with the best match. Before it could happen, that another channel was selected (e.g. typing “nmot” selected “nmot\_up” instead of “nmot”)

### Channel settings dialog: Copy/Paste of settings improved (ID2653)

- The selection in columns with a drop down button can be copied and pasted.
- A single value can be pasted into all selected cells (per row).
- The selection of a drop down button cell is correctly applied to all selected cells (in the column).

### Telemetry: Exporting a WDServer telemetry file is reenabled (ID2504)

- The WinDarab support to export a telemetry stream (from WDServer) is back.
- **Note:** Export of v2 telemetry streams (only supported by the experimental WDServer.Net) is still not supported.

## WinDarab v7.07.022

22.01.2020

### x-y-Plot: The layout calculation was broken after a small change regarding Point and Rectangles

- For easier transition between our old GUI library and MFC library, we adopted the implementation of point and rectangle classes. But there were some side-effects the compiler didn’t warn and

produced wrong/not-working code.

In this case the layout calculation in the x-y-plot window was hit by such a side effect.

#### **x-y-Plot: Calculation of linear regression changed (ID2652)**

- WinDarab used a different way to calculate a linear regression line to match the data. Now the calculation is done using the same algorithm like MATLAB or Microsoft Excel.

#### **Math function editor: Several improvements of Intellisense behavior (ID2650)**

- Several fixes to improve Intellisense feature
  - Correct refreshing of the Intellisense content, if selecting another math function.
  - Correct highlighting of error, if they are detected by the math parser (and not by the intellisense parser)

#### **Oscilloscope: Local min/max not accurate in a low zoom depth (ID2647)**

- If the zoom depth is very low the min/max detection sometimes skipped a possible extrema value, if multiple samples are projected on the same (or next) x-pixel on the screen.  
In other words: If zooming deeper inside, the local min/max values changed sometimes ...

#### **Fix: Crash, if trying to open files created by DataLoggerImport using the new file format v2.**

- WinDarab crashed, if the user tries to open files using the the new file format v2 (with the new DCP content used for Highspeed-Channels and device events)

## **WinDarab v7.07.021**

**21.01.2020**

#### **Compiler selected a strange CPoint cast which caued a division by zero (ID2655)**

- The code line `CPoint(x,y) / 2` was compiled as `CPoint(x,y) / CPoint(2,0)` and caused a division-by-zero in the Instrument panel.  
All additional (and obsolete) CPoint operators were removed, because the above use case was the only code line which used those "strange" operators at all.

## **WinDarab v7.07.020**

**21.01.2020**

#### **Math function editor: Showing the function definition and description while typing (ID2650)**

- While typing a math function Intellisense shows the call syntax and brief description of the current function.

#### **Math function editor: Intellisense offers the existing lookup tables (ID2651)**

- If the user types a `lookup()` function in the math editor, Intellisense offers the list containing all defined lookup tables.

#### **Color pickers: Now using MFC controls to pick colors (ID2646)**

- After implementation of the "Distinct color" palette and a testing phase, we replaced the color picker control by the comon MFC control and removed the old-fashioned GUI controls.

#### **ASCII Import: The ASCII-Import was broken (ID2643)**

- Creating a new measurement file by importing an ASCII file was broken because of some changes in the code to load a file using the WinDarab format v1. The import process simply ended without saving anything! Now the ASCII import is back...

#### **Math function Average: The result is not correct if a math channel with NoValues is filtered (ID2548)**

- If the average() function is used to smooth a math channel using the new “NoValue” feature, the average isn’t calculated correctly.  
Now the NoValues are counted correctly when calculating the average value.
- **Note:** The average functions uses <n> Samples before and behind the timestamp for which the average is calculated. In the old implementation the first/last sample value was used to calculate the <n> first/last average values. The new implementation uses “NoValue” instead which results in a slightly different result – but from our point of view a “better” result.

#### **Oscilloscope: Option to show local min/max values (ID2547)**

- A new function was added to show the local min/max values in the current view of an oscilloscope. The values are printed below/above the min/max occurrence directly in the graph region.  
The option can be activated/deactivated in the ribbon (category Channel or by pressing the key D-E – “D”isplay – “E”xtremas)

### **WinDarab v7.07.019** **11.01.2020**

#### **WinDarab suport ASAM MDF4 (\*.mf4) files**

- WinDarab crashed, if a MDF4 file was opened with more than 32 timelines.

#### **BMS2API/COMApi: Compatibility issues with MATLab fixed**

- Some issues were fixed regarding the interoperability between MATLab and BMS2Api.

#### **BMS2API/COMApi: Periodic timelines can be used for channel in UserDataFile**

- If a 3<sup>rd</sup> party application uses UserDataFile to create a WinDarab data file, a channel is allowed to use any kind of timelines. New is, that a periodic timeline is stored in the file “by definition” and -as before - as a discrete timeline (which saves a lot of memory in the file).
- **Note:** Measurement files created with this version of WinDarab which use periodic timelines cannot be opened by previous versions of WinDarab!

### **WinDarab v7.06.033** **11.01.2020**

#### **WinDarab Free locked by bad license verification**

- Because of problems in the license management backend users of “WinDarab Free” were locked out.  
Before locking out a user completely, WinDarab continues as a “WinDarab Free”.

### **WinDarab v7.07.018** **08.01.2020**

#### **BMS2API/COMApi: DiscreteTimeline object published an incorrect IDispatch interface**

- The DiscreteTimeline object returned the IDispatch interface of the underlying ITimeline interface instead from the IDiscreteTimeline interface, This wasn’t a big deal – but MATLab wasn’t able to call any methods of IDiscreteTimeline.

#### **Math editor crashed, if trying to use the IIR filter assistant**

- If the user tried to create a new math formula using the filter assistant for lowpass, highpass, ... filters, WinDarab crashed.

#### **WinDarab suport ASAM MDF4 (\*.mf4) files (ID2639)**

- WinDarab can be used to watch ASAM MDF4 files.
- **Note:** Some MDF4 features are not available. E.g.
  - MDF4 Events

- Opening/viewing Attachments
- Some data types like CANDate/Time and String types
- Array types

#### **WinDarab could crash after interruption of the network connection (ID2636)**

- If the user has opened a file on a network share and the connection is lost (e.g. moving the notebook somewhere else while WinDarab is opened), it could happen that WinDarab crashed if the network connection is reestablished.

### **WinDarab v7.07.017 11.12.2019**

#### **Oscilloscope: Scrolling in one window didn't scroll others (ID2633)**

- If the user scrolls the data in an oscilloscope using the mouse wheel without setting the focus to this view (by activating the window), the new visible range wasn't distributed to other oscilloscopes/windows.

#### **Oscilloscope: Mouse wheel scroll direction can be inverted (ID2620)**

- Some users didn't like the direction WinDarab scrolled the data in the oscilloscope, if they were using the mouse wheel. In "Tools" / "Settings" / "Program" tab "Measurement window", you can invert the scrolling direction.

#### **All worksheet views support "Copy as image" (ID2612)**

- If you open the context menu of a worksheet view (right click on caption/border) or in the ribbon "Windows" / "Copy Image" we added two actions:
  - "Copy as image" ( ribbon "Copy image" / "Content only")  
This will copy the inner of the view as an image to the clipboard (without the frame / caption bar)
  - "Copy as image (with frame)" ( ribbon "Copy image" / "Content&Frame")  
This will copy the whole view as an image to the clipboard (including the frame / caption bar)

#### **Oscilloscope: Mouse wheel scroll direction can be inverted (ID2597)**

- We added the option to disable changing the current lap / outing in an oscilloscope, if the zoom mode of the oscilloscope is set to "Lap" or "Outing". If the user tries to drag the cursor out of the view, the cursor stops and no scrolling occurs.  
The option can be enabled in "Tools" / "Settings" / "Program" tab "Measurement window".

#### **The font used by WinDarab in the data views can be selected/changed (ID2539)**

- The font and size can be selected in "Tools" / "Settings" / "Program" tab "Miscellaneous". The font is used in all data views (like oscilloscope, x-y-plot, histogram, ...) whenever graphical data is shown (e.g. axis scales, data values, ...)

#### **Math expression: "Global" variables are declared with "var" and not "global" (ID2527)**

- We renamed the keyword "global" to "var".
- To prevent that users think "global" variables are shared between different math channels, we renamed "global" to "var":
  - "var"-declared variables keep their values between the calculation of one sample to the next. They are initialized once when evaluating the value of the first sample and can be used to "transfer" intermediate results to the next calculation cycle.
  - "local"-declared variables are always initialized when calculating a sample. They can be used to calculate intermediate values – e.g. if an intermediate result is required multiple times in the same expression.

#### **Math Editor now supports keyword highlighting and "Intellisense" (ID1644, ID2580)**

- The math editor supports several new features:

- Keyword highlighting: Words are colorized depending on their meaning (Functions, Variables, Identifiers, Values, Flags, ...)
- While typing a keyword Windows shows possible matches in a list, the user can select one or if he continues (e.g. by pressing space) the best match is automatically inserted.
- While typing a syntax check is executed.
- When moving the cursor, the most-inner bracket pair is highlighted.
- Example:
 

```
DetectEvent(Rising; {nmot} >= 7400; 100; {nmot} < 7300; 50)
```

## WinDarab v7.06.031 11.12.2019

### Multiline conditions weren't saved correctly (ID2642)

- If a predefined condition (Tools / Conditions) consist of multiple lines, the conditions wasn't saved correctly. Only the first line was restored after restarting WinDarab.

### Oscilloscope: "Make Zero" didn't work for math channels (ID2638)

- Internally WinDarab used an inconsistent way to allow the "Make Zero" action and executing the actual action. While WinDarab offered the action for math channel, the action itself wasn't executed because it's denied for "internal" channels (and a math channel is an internal channel).
- Now offering and executing of "Make zero" use the same logic – and this distinguishes correctly between math channels and internal channels.

### Analysis windows: The evaluation range didn't match the range in the oscilloscope (ID2635)

- Because of adding the x-axis shift twice for overlays (except the overlay of the x-axis) the analysis windows didn't evaluate the correct range (if they use the GetTimeBeginEnd() method like the histogram).

### Printing: If the black theme is used for WinDarab, some elements weren't visible on printouts (ID2635)

- Now WinDarab switches to the "White theme" temporarily while preparing the print output. This ensures that all elements are visible (on white paper) ...

### Timeline.End returns a timestamp 100ns behind the last sample (ID2628)

- This is for forward compatibility with WinDarab v7.7.
- In previous version of WinDarab Timeline.End returned the exact timestamp of the last sample. And methods taking a time range (like DataAccessorConfig.CreateDataAccessor) included the last timestamp if it's equal with the EndTime value. Because the behavior will be changed in v7.7 in a way that EndTime value is excluded and Timeline.End will return a time value behind the last sample, we decided to introduce this change already today (in v7.6 adding 100ns doesn't has any effect)

### Math channel: DetectEvent could crashed, if math channel uses a conditional timeline (ID2627)

- A math channel caused a crash, if the following conditions are true:
  - The DetectEvent function is used by the math channel
  - The math channel is calculated for a a linked file.
  - The math channel uses a conditional timeline
  - One or more channels in the DetectEvent function are measurement channels (not math channels)

## WinDarab v7.07.016 26.11.2019

### Math formula editor: Expressions with multiple lines weren't supported any more (ID2632)

- Some flags weren't set for expression textbox in the new math channel editor. Expressions with line breaks couldn't be entered and existing expressions were cutted at the first line break. The

bug is fixed.

#### **FlowChart: Variables couldn't be reused in flow math objects (ID2631)**

- If variables are declared in a flow chart it was possible to assign values to them. But it wasn't possible anymore to reuse them in other math object expressions. The bug was fixed.

#### **Moving the cursor position after double-clicking a value in a Min/Max-Table (ID2630)**

- If a value is double clicked in a Min/Max-Table or FlowChart table, the cursor in the oscilloscope jumps to this position. But now, the cursor position on the screen doesn't change. Instead the oscilloscope data is scroll accordingly.
- **Note:** There's a hidden option to change the behavior of the "JumpToPosition" method used by WinDarab in case a user double clicks on a value. The hidden option changes the above described behavior in a way, that the cursor changes the screen position but scrolls the oscilloscope to keep the cursor inside of a virtual border. In the settings below the virtual border is given in a percentage of the total oscillocope width (each for the left and right side)

```
settings.ini
[Settings]
; Scroll to keep relative screen position of the cursor (default:1)
JumpToKeepsRelativePosition=1
; Cursor doesn't move close than 10% to the left/right side, if
JumpToKeepsRelativePosition==0
ScrollBarPercentage=10
```

#### **Math function DetectEvent crashed (ID2627)**

- The math function DetectEvent crashed, if the math function is calculated using a calculation condition (filtered timeline) and a referenced channel belongs to a linked file.

#### **COMApi/BMS2API: Serveral property names have been changed**

- We have changed the property names for time ranges in several classes (e.g. Timeline, Channel, Lap, DataFile) and also added new ones
  - Begin -> TimeBegin -> The time of the first timestamp/value
  - End -> TimeEnd -> The end time of a range (somewhere behind TimeLast)
  - <new> -> TimeLast -> The time of the last timestamp/value
- These changes requires some work on source code to rename the properties accordingly! But we hope its easier (and more clear) which timestamp the properties return.
- If a method requires a begin/end timestamp you should use TimeBegin and TimeEnd. This will include all samples in the range – exluding the sample at the TimeEnd timestamp.

E.g.

```
DataAccessorConfig.CreateDataAccessor( DataFile.TimeBegin, DataFile.TimeEnd)
```

➔ will return all samples of the file

```
DataAccessorConfig.CreateDataAccessor( Lap.TimeBegin, Lap.TimeEnd)
```

➔ will return all samples within the lap

```
DataAccessorConfig.CreateDataAccessor( Lap.TimeBegin, Lap.TimeEnd + TimeStampEpsilon)
```

➔ will return all samples within the lap AND the first sample of the next lap

#### **COMApi/BMS2API: Create/write new measurement file (ID2610)**

- An 3<sup>rd</sup> party application/plugin can use the UserDataFile class to create new measurement files with application generated channel data.
- The support of COMApi including the C# wrapper classes in WinDarab.Net.dll is added now.
- Files created with the new API are written in WinDarab file format v2 which is supported since WinDarab v7.6.
- An brief example how to write a file can be found in the BMS2ApiSamples.Net project, see the source in CreateUserDataFile.cs

- Additional info: If a file is published or saved (UserDataFile.Publish() or UserDataFile.SaveAs) ...
  - ... the following channels are used for special purpose:
    - “laptrig” is expected to contain a SINGLE peak value (e.g. a 1-bit value) to mark the beginning of a lap. Best used with a DiscreteTimeline having two timestamps for each laptrigger (laptrig == 1 for the first timestamp and laptrig == 0 for the second timestamp)
    - “lap” contains the lap number.
    - “outing” contains the outing number
    - Recommendation: Record “lap” and “outing” using the same timeline as “laptrig” and set the signalkind for all channels to “Digital”.
    - The names of “lap” and “outing” are examples (the names DataLoggerImport uses). Nevertheless you can use different channel names and configure the global options / Special channels accordingly.
  - The distance information is calculated/selected according to the settings in the program settings / Special Channels.  
E.g. if a channel “speed” is found, WinDarab will calculate a “xdist” channel based on the speed values.
- Even the methods are available, you shouldn’t use a UserDataFile to “work” immediately with the file. E.g.
  - adding persistent virtual channels
  - or other persistent channels
  - Access math channels (before the referenced channels are filled with data)
  - ...
 Instead:
  - Save the UserDataFile to a measurement file
  - Close the UserDataFile
  - Open the saved measurement file as a “normal” DataFile.
  - Do what ever you want!

#### **COMApi/BMS2API: Application.WinDarabVersion and Application.ApiVersion added (ID2602)**

- Two properties were added:
  - ➔ WinDarabVersion: Returning the version of the installed/underlying WinDarab as a string.
  - ➔ ApiVersion: Returns a numeric / incremental version number of the COMApi/BMS2Api.
 Note: This version is only incremented, if new methods are added or the behavior is changed (actual the last shouldn’t occur)

### **WinDarab v7.07.015 19.11.2019**

#### **Oscilloscope: After loading a oscilloscope view the values of hidden channels are drawn in black (ID2626)**

- After loading a oscilloscope view the actual color of a channel is never updated unless the channel visibility or color is changed. The bug was an unrecognized side effect of the changes in the previous version regarding the extended colorpalette available for channels.

#### **Outing Report: Emphasizing the values changed due to user confusion (ID2625)**

- In 2017 we added a feature to emphasize min/max values in the outing report using different colors. A “wanted side effect” of this coloration is also the result determination of the lap/total values (lap values, if a racetrack segmentation is used to evaluate the segment min/max values).
- If a column is configured to show the minimum value, previous WinDarab versions used to show the “maximum value of the segment minimum values”, if both are emphasized by colorization. NOW the “total minimum” is used!
- If a column is configured to show something else as the minimum value, the “maximum value of all sub values” was taken – this behavior wasn’t changed.
- Another issue was the setting dialog: The default color settings default to emphasize the maximum value – even if the column function “Minimum” is selected. Because it makes more

sense to emphasize the total minimum in this case, we changed the settings dialog:  
Now the colors for Minimum and Maximum value are swapped, if no colorization is configured for the minimum value – and swapped back if the function is again changed to another function expect “Minimum”

#### **File explorer could crash if a new overlay is added (ID2624)**

- WinDarab crashed, if a not-loaded file (-> a grayed-out file) is selected in the tree view and the user tries to add a new overlay.

#### **COMApi/BMS2API: Create/write new measurement file (ID2610)**

- An external application can use the class UserDataFile to create new measurement files with application generated channel data.
- **Attention:**
  - This is an early release for testing purpose. The classes and its methods may change but not the logic/steps to write a file.
  - The required interfaces are available in COMApi and BMS2API. Nevertheless the COMApi-C#-Wrapper WinDarab.Net.Dll doesn't contain any support yet (will come soon).
- Files created with the new API are written in WinDarab file format v2 which is supported since WinDarab v7.6.
- An brief example how to write a file can be found in the BMS2ApiSamples.Net project, see the source in CreateUserDataFile.cs

### **WinDarab v7.07.014** **12.11.2019**

#### **A set of 20 distinct colors is added to colorize channels in the oscilloscope (ID2270)**

- Instead of the base colors WinDarab offers a set of 20 distinct colors to be used for channels in the oscilloscope.
- The list of colors can be changed by an entry in setting.ini (here are the default colors)  
[Options]  
DistinctColors=0xe6194b,0x3cb44b,0xffe119,0x4363d8,0xf58231,0x911eb4,  
0x46f0f0,0xf032e6,0xbc6f0c,0xfabebe,0x008080,0xe6beff,0x9a6324,0xffffac8,0x800000,  
0xaaaffc3,0x808000,0xffd8b1,0x000075,0x808080,0xffffffff,0x000000

Here's an alternative set of colors:

```
DistinctColors=0xa6cee3,0x1f78b4,0xb2df8a,0x33a02c,0xfb9a99,0xe31a1c,0xfdbf6f,0xff7f00,  
0xcab2d6,0x6a3d9a,0x8dd3c7,0xffffb3,0xebada,0xfb8072,0x80b1d3,0xfdb462,  
0xb3de69,0xfccde5,0xd9d9d9,0xbc80bd,0xffffffff,0x000000
```

### **WinDarab v7.07.013** **09.11.2019**

#### **Fix: A still used dialog resource was readded (Math channel file dialog)**

- A resource was removed which is still required by WinDarab. The resource is readded.

### **WinDarab v7.07.012** **08.11.2019**

#### **Math channels support NoValue (ID2608)**

- In math channels you can use “NoValue” to suppress any output in WinDarab (e.g. the oscilloscope signal is interrupted).
- A math function “IsValue” was added to check a input value for “NoValue”.
- Also all analysis windows (should) work correctly, if they meet a “NoValue”.
- Note: All comparisons with NoValue will return false (NoValue is internally a NaN double value)
- Here's a sample expression:

if (nmot > 6000; nmot; NoValue)  
This channel will show only a signal, if nmot is larger than 6000 – anywhere else there's nothing!

#### **New Math Channel editor implemented (ID2596)**

- The Math Channel editor was implemented for new with MFC – because of some problems regarding copy and paste of larger formula expressions – and for future implementation of syntax highlighting.

#### **x-y-Plot: The width/height of x- and y-axis can be adjusted by the user (ID2540)**

- The x-y-axes / scales can be resized by the user.

### **WinDarab v7.06.031** **08.11.2019**

#### **WinDarab could crash if closing as flowchart (ID2619)**

- Because of an incorrect assumption that the first tab in a FlowChart is always the “Output” tab, WinDarab could crash when closing the FlowChart or closing WinDarab. When closing a FlowChart WinDarab closes the tabs beginning with the first tab – the “Output” tab. When closing the “Flow” tabs a crash was possible. Nevertheless: The bug is very old but didn't seem to occur often!

#### **Oscilloscope: The scrollbar didn't show the visible range any more (ID2618)**

- The scrollbar in the oscilloscope used to visualize the visible range proportional to the total data range. This functionality got lost a long time ago (early 2018) and is now back ...

#### **Union timelines used a 1sec time resolution always (ID2614)**

- To save memory, WinDarab determines the necessary time format for “union timelines” by evaluating the timestamp format of the input timelines. Because of a bug WinDarab always used a 64-Bit-Timestamp with a 1sec resolution (instead of 32-Bit with 0.1ms commonly used by the datalogger hardware)

#### **Union timelines used a 1sec time resolution always (ID2614)**

- To save memory, WinDarab determines the necessary time format for “union timelines” by evaluating the timestamp format of the input timelines. Because of a bug WinDarab always used a 64-Bit-Timestamp with a 1sec resolution (instead of 32-Bit with 0.1ms commonly used by the datalogger hardware)

#### **Copy & Paste of text content didn't work correctly (ID2611)**

- WinDarab mixed Ansi and Unicode content when copying data to/from the clipboard. Now WinDarab copies text to clipboard using both clipboard formats (Ansi and Unicode). Also when pasting content from the clipboard WinDarab prefers Unicode content before using Ansi content.

#### **“Samples” channel removed in files using WinDarab file format v2 (ID2603/2617)**

- After introducing the new WinDarab file format v2 the “Samples” cannot be supported by these files. “Samples” will not be added for these files any more. Nevertheless the channel still exists for WinDarab files using the old v1 format.

### **WinDarab v7.07.011** **23.10.2019**

#### **x-y-plot: Height/width of the x- and y-axis can be resized with the mouse (ID2540)**

- In x-y-plot the height of x-axis and width of y-axis can be adjusted with the mouse.
- The current implementation is not finished yet. The following issues have to be implemented:
  - Auto adjust initial width/height
  - Move width/height, if axis is moved (e.g. from left to right)
  - Save the width/height in x-y-plot setup

## WinDarab v7.06.030 23.10.2019

### File extraction of WinDarab data format v2 files crashed (ID2606)

- WinDarab crashed, if data is extracted into a new file and the source file uses the WinDarab data format v2 and contains VTabs (number to text table).

### Flow charts didn't work as expected, if the "NextSample" object is used (ID2607)

- Due to the removal of the global timeline, the NextSample object didn't work as expected. Instead the flow chart evaluation stops too early – as soon as the flow state reaches a "NextSample" object.
- **Note:** If possible the use of "NextSample" should be avoided as good as possible. In most cases a conditional object should help to avoid endless loops in a flow. But if it's not possible (or too complex) to prevent endless loops, a "NextSample" object helps ...

## WinDarab v7.07.010 16.10.2019

### Instrument panel: scale values in round gauges disappear when resizing the gauge (ID2599)

- If a round gauge in the instrument panel is resized some scale values disappear sometimes. The behavior is by design, because while WinDarab is drawing the gauge the scale values are only drawn, if they don't overlap with other values (or are very close to others). We changed the layout calculation a little bit to use more of the available space:
  - The distance of the round scale numbers to the outline of the gauge is reduced (and elliptic).
  - The height calculation of the scale numbers is slightly modified.
  - The check for overlapping numbers is a little bit more precise**Nevertheless:** Still scale values may disappear if a values collides with its neighbor values on the scale.

### Racetrack/trackmap dialog work flow slightly changed (ID2561)

- After saving a change in the racetrack/trackmap editor, the "Cancel" button on the first wizard page is changed to "Finish"
- If the "Cancel" button is pressed on any other page, the wizard returns to the first page and then the user can "Cancel" (if nothing was changed yet) or "Finish" (if a prior change was already saved).

### WinDarab doesn't show "Division by zero" warnings any more (ID2223)

- If a math channel is calculated and a "Division by zero" happens, WinDarab doesn't show any error/warning messages in the status bar anymore.

### x-y-Plot: The color scale setup can be save in a "color scale" file and restored (ID2151)

- In the settings dialog of a x-y-plot the user can save and load color scale setups to/from a file.
- **Note:** If a color scale is loaded from a file, the min/max of the color scale is adjusted to the configured range of the z-channel. This includes a proportional scaling of the color breakpoints!

## WinDarab v7.06.028 16.10.2019

### DataLoggerImport didn't convert WinDarab format v2 raw files correctly (ID2600)

- The conversion of raw files didn't produce correct WinDarab files, if multiple fragment files are merged into a single output file (one file per lap / outing) if a reset of the Data Logger caused the timestamp to restart at zero. Because DataLoggerImport missed to add offsets to the timestamps, the timelines weren't monotone increasing.
- Also DataLoggerImport didn't mark the laptrigger signal in files (one file per outing). As a result WinDarab didn't show any laps because of the missing laptrigger information.

### **WinDarab crashed when opening WinDarab format v2 files as linked files (ID2600)**

- WinDarab crashed when opening WinDarab format v2 files as linked files, if statusblock channels are turned off (invisible) or a statusblock recording block doesn't contain any supported statusblock channels

## **WinDarab v7.07.009**

**12.10.2019**

### **Math function ValueAtTime/ValueAtDist shows randomly wrong values with telemetry streams (ID2598)**

- If multiple math channels using the methods ValueAtDist or ValueAtTime are shown, the results were randomly wrong because all math function instances used the same location to store their intermediate values: The saved these values in the Operator-definition instance and not in their own Operands-instance in the expression tree.
- **Note:** The bug occurred, if the referenced xdist position is the same for two or more math function instances while they are calculated one after the other. This is also the reason why the probability to see this bug for offline files is very low (nearly impossible) and much higher for telemetry.

### **x-y-Plot: Default z-axis scale with “rainbow” colors (ID2595)**

- We added a new predefined z-axis color scale with the seven rainbow colors.

### **x-y-Plot: Invalid z-axis min and max limits could “destroy” the color scale (ID2595)**

- If invalid z-axis min/max limits were applied, they “destroyed” the color scale. The user had to recreate the color scale to get back to a working configuration.  
To protect the colorscale against invalid range settings the z-axis limits are temporarily corrected to a very small range around the configured values.
- **Note:** The bug only occurred, if not both x-axis limits Min and Max were set to <auto>..

### **GPS distance didn't work anymore after a change in v7.07.008 (ID2593)**

- The calculation of xdist using GPS channels didn't work anymore after recent changes. This bug is fixed.

### **WinDarab crashed when loading a desktop config, if loading/creating a worksheet failed (ID2586)**

- If a desktop file is loaded and a worksheet couldn't be created, WinDarab crashed.
- **Note:** We fixed the crash but weren't able yet to reproduce the cause, why the worksheet couldn't be created.

### **Oscilloscope: If a channel is dragged to add a new region, the region is highlighted before while dragging the channel (ID2586)**

- If a user drags a channel over the oscilloscope to add a new region to the oscilloscope, the future region of the new area is highlighted while the channel is hovered above the region separators.

### **WinDarab couldn't load old trackmap files (ID2573)**

- WinDarab didn't load trackmap files using the old ASCII-Format – which is used by the sample trackmap files delivered with the WinDatab installer.

### **x-y-Plot: Toggle between <auto> and absolute x-y-z-axis limits (ID2560)**

- Buttons were added in the ribbon (and hotkeys X, Y and Z) to toggle the x-, y- or z-axis limits between <auto> and the absolute value (current values of <auto>).

### **Worksheet items can be arranged (ID2559)**

- In the “Windows” ribbon category we added the function “Tile controls” to tile all worksheet controls on the current worksheet.
- The function is also available, if you open the context menu of the worksheet control caption/border.

### **The math inspector windows is splitted into two panes to show the math expression (ID2556)**

- The upper pane shows the value tree.
- The lower pane shows the math expression.

### **Oscilloscope: An existing y-axis can be switched to an automatic mode (ID2542)**

- After the user added a y-axis for a channel in the oscilloscope, the y-axis context menu can be used to turn the y-axis into an “automatic y-axis” (click “Selected channel” in the context menu)
- An automatic y-axis always shows the corresponding y-scale for the selected channel in this area.
- A second click on “Selected channel” REMOVES the automatic y-axis.
- If a channel is dropped onto an automatic y-axis, the y-axis is a normal axis for the channel.
- **Note:** A automatic axis always fills the complete height of the y-axis area and cannot be changed (height or vertical position).

### **x-y-Plot: Thickness of lines can be changed (ID2508)**

- The thickness selection is also applied, if x-y-Plot uses lines to connect the samples.

### **WinDarab aligned telemetry laps even replay is off**

- If WinDarab is (re)playing a telemetry stream and the user enabled the “Align online laps” option, WinDarab continued to align the laps if a lap trigger war received even the “replay” was disabled.

### **Foreground color of worksheet tabs is white for dark colors**

- We added some code to select a different text color (white or black) depending on the worksheet tab color.  
**Note:** To decide if white and black is better suited, WinDarab calculated the brightness (gray level) of the colors. If the background color is closer than 50%, then the foreground color is switched.

### **Trackmap rotation value is saved**

- WinDarab remembers the rotation value in the track map calculation dialog.

## **WinDarab v7.07.008**

**03.10.2019**

### **Less restrictions when opening different files as a linked file (ID2581)**

- New in WinDarab is the support to open different files as a linked file – even the recording configurations are different:
  - All channels occurring in at least one of the member files is available in the linked file.
  - If a member file doesn't contain a channel, the area in the linked file doesn't contain any samples
- Following restrictions are still valid:
  - All channels have to use the same data format and bit size (e.g. unsigned 16-Bit)
  - All channels have to use the same quantization (e.g. offset/gain)
- To complete the support, the file extract functions were refactored.
  - In case of a linked file, file extract supports the creation of additional timelines which are required to export linked channels correctly, if some member channels are missing.
  - The export functions are much faster now because they use multiple threads to collect the data to export.
- **Note:** The support of missing channels in linked files is still subject for deeper testing. Especially we currently experience problems because there are now many combinations and actions possible regarding mixing of channels. E.g.:
  - A missing alias channel can be used to provide a replacement – nevertheless the data format has to match. But – and this is a problem – if a linked channel exists using a correct alias channel but the alias channel settings are changed, it can happen that the “new” alias channel doesn't match anymore. WinDarab doesn't crash but also the result will be strange.

- If a channel is recorded with different recording settings (e.g. sampling rate) the timeline will be merged – for sure, the timeline/recording information gets invalid (e.g. one part was sampled with 2ms and 5ms under condition A, the second part was samples with 5ms and 10ms under condition B). So far this should work, WinDarab only shows “combined timeline” without a closer recording information.
- Nevertheless in case of High-Speed-Channels (VCU), the timestamp frequency isn’t taken into account currently (it’s assumed to be the same!). Different timestamp frequencies can cause problems when extracting the file.

We will test these issues closer, if we have test data for these scenarios ...

### **Manual lap triggers have to be adjusted to global timeline timestamps (ID2476)**

- With introduction of ignoring hardware laptrigger we had to fix a side effect in the “Split lap dialog”. In case of splitting a lap for files using the WinDarab format v1, the added lap triggers have to be aligned to existing timestamps on the global timeline. In files using the Windarab format v2 this is not required anymore because lap triggers are stored in the files by time.

## **WinDarab v7.07.007**

**14.09.2019**

### **Math parser showed a false syntax error (ID2584)**

- The extension to support multiple command lines in math functions introduced a small parser. If the math formula didn’t contain whitespaces between keywords and operators, the parser shows a syntax error:  
e.g. the formula “nmot>8000” failed while “nmot > 8000” is correct.  
Now the first formula is also parsed as expected.

### **File Explorer: When adding a new overlay the currently selected lap is attached immediately (ID2583)**

- If a new overlay is added in the “File Explorer” pane, WinDarab attaches the selected file/lap to the overlay immediately (or the fastest lap, if the file row is selected).

### **Context menu item with textbox show the unit in front of the input box (ID2578)**

- E.g. the “Shift file by” menu item shows the unit of the x-axis now: “Shift file by [m]:”

### **The action of context menus with textboxes was executed twice (ID2577)**

- If a context menu had a textbox and the entered value was applied, the action was executed twice – which doubled the effect. In some cases (e.g. File Shift by 10s shifted the file by 20s)

### **Change password batch dialog could crash if files in selected folder are enumerated (ID2576)**

- If the user wanted to change the file password using the “Change password batch dialog”, WinDarab could crash if the selected folder structure contains many subfolders. The crash was caused by unsynchronized access to a subfolder list by multiple threads.

### **DataLoggerImport: Converted data can show small gaps (ID2575)**

- Converted data can show small gaps, if multiple fragment files are converted into a file (-> not “one file for each fragment”)
- The issue was caused by the DataLogger, if the DataLogger appends dynamic statusblock records in a fragment file after the next fragment file already has started recording. In this case the timestamp range overlaps and causes the gaps because DataLoggerImport adds a small time offset to ensure the monotony of the timeline.
- But because dynamic statusblock channels are normally invisible in WinDarab, the timespan between the last sample in a fragment and the last (invisible) statusblock recorded appears as a gap.
- Now DataLoggerImport discards all channel samples of dynamic statusblock records which are recorded after the last measurement block (or lap endtime).
- But the dynamic statusblock in the inform windows still contains this data!

- **Note:** Showing dynamic statusblock channels can be enabled in WinDarab for internal purpose with special setting.ini flag.

#### **Show/Hide overlay crashed, if previously an inform view was closed (ID2568)**

- WinDarab crashed, if the user closed an inform view and later tried to hide/show the overlay.

#### **Adding a new area to an oscilloscope (ID2557)**

- If a new area is added to an oscilloscope, the existing area into which the new one is added is divided into two halves. One half for the existing area and the other half for the new area. The height of the other areas isn't touched anymore.
- If a area is deleted, the space is added to the area above – or below, if the topmost area is deleted.

#### **Track map: The options “Show names” and “Show numbers” are changed (ID2555)**

- If “Show names” in a trackmap is selected, WinDarab shows the name of segment (or nothing). If “Show numbers” is enabled, the segment number is shown. If both are enabled, the segment number and the name is shown.  
**Note:** Before “Show names” showed the name or, if there wasn't a name, the number.

#### **Analysis windows: The x- and y-scales show the attached channel name (ID2540)**

- The x- and y-axis of analysis windows show the name of the attached channel like the scales in the oscilloscope.

#### **Instrument panel supports multi-selection (ID2519/2500)**

- Instrument panel allow to select multiple gauges by pressing the Ctrl + Left or using a rubberband.
- The location and size of all selected gauges can be adjusted in a single step.
- Copy and paste of the selected gauges is supported
- The layout grid is removed. Now the borders of the gauges act as magnetic lines making it easy to align/adjust gauges.
- If multiple gauges are selected, the arrow keys can be used to move the gauges by pixel (without snapping to the magnetic lines). Additionally the arrow keys can be used with Shift to align all gauges to the top, left, bottom or right border, together with Ctrl the size can be adjusted (Ctrl+right, Ctrl+bottom) or aligned and sized (Ctrl+left, Ctrl+top)

#### **Ignoring hardware triggers (ID2476)**

- WinDarab now allows to “ignore” / ”hide” lap triggers created by the recording hardware. To ignore a lap trigger simply use “Delete laptrigger” from the context menu / ribbon menu.
- Hidden hardware lap triggers can be restored in the “Lap trigger” dialog (-> ribbon). Hidden hardware triggers are shown in red using an italic font.
- **Note:** This feature is not yet fully implemented/tested. Especially distance normalization for linked files lacks, because WinDarab used some optimizations in the past which cannot handle deleted hardware triggers in correctly.  
But the feature can be used for single files (e.g. if data was converted with “one file per outing”).

## **WinDarab v7.07.006 30.08.2019**

#### **Hide overlay: Hiding an overlay could cause a crash in the histogram view (ID2568)**

- WinDarab crashed, if the user hides an overlay with the feature introduced in v7.07.001 while a histogram view is on the active worksheet.

#### **Oscilloscope: Sometimes a signal wasn't drawn correctly if view is scrolled (ID2543)**

- If the oscilloscope is scrolled (with a deep zoom) sometimes the signal wasn't updated correctly. The line between two samples wasn't drawn or – in case of digital signals – the samples were connected directly with a interpolated line instead with a horizontal+vertical line.

### **Oscilloscope: Number format can be adjusted in the channel / display context menu (ID2534)**

- The number format of a channel can be adjusted in the context menu of the channel (right click on a channel) or through the “display” context menu (key sequence: “D” - “N” - ...)

WinDarab v7.06.027

30.08.2019

### **Generating a track map failed if a single file/lap is used (ID2570)**

- If the user tries to generate a trackmap using “speed / transversal acceleration” algorithm in combination with a single file containing the data of a single lap, the trackmap generation failed or even crashed.

### **Track map: Using GPS coordinated to show the current car position ENABLED (ID2566)**

- Unbelievable but true: The feature to use GPS coordinates to show the exact car position on the trackmap was only enabled for Bosch users – since the function was first implemented in 2010. Now the function is enabled and we encourage every user with GPS data to test this feature!

### **Math function DetectEvent could cause WinDarab to stop responding (ID2564)**

- If the begin condition in a DetectEvent expression refers to multiple channels using different timelines, it could happen that WinDarab stopped responding if the DetectEvent is evaluated. This happened because because combined timelines (used by these conditions) are empty after creation. They are updated in the background and, when finished, they notify their referers to update. In case of DetectEvent the “empty timeline” causes the algorithm to enter an endless loop giving the timeline no chance to retrigger the calculation.

### **A default segment color was black even for the “Bosch black” theme (ID2554)**

- The alternating colors for trackmap segments which are used by default are adjusted, if the user switches between “Bosch Black” and other themes.
- **Please note:**
  - The alternating segment colors can be user-defined in “Tools” / “Options” / “Program” / “Colors”.
  - If the color settings were touched once in the past, the automatic switching is not working, because the old settings were “user-defined” by default. So you have to select the “default” color box (the big left/bottom color box) to enable the automatic switching.

### **The "Select Printer" dialog was show three times when trying to print something (ID2552)**

- If the user wanted to print something, the “Select Printer” dialog appeared three times before printing started or canceled.

### **Video view: “Auto hide” button added in video navigation control (ID2545)**

- Because some new customers didn’t recognize that the video view has a collapsed video navigation control at the bottom which can be opened by hovering the mouse over it, we added a “Auto Hide” button.
- If a new video view is opened, “Auto Hide” is disabled by default. In case a “old” desktop is loaded, the “Auto Hide” is enabled by default. If a desktop is saved, the current state of “Auto View” is written to the desktop file.

### **Telemetry: When connecting to a running stream, the historic lap triggers weren’t shown (ID2537)**

- If the user connects to a running telemetry stream it could happen, that WinDarab didn’t show the recent laps / laptriggers. The issue only happen with certain telemetry channel configurations and was caused by a recent change where received lap infos are stored until the stream initialization has completed. But in the case that some channels are considered to be min/max channels, the generated lap values weren’t stored and the delayed processing of the lap infos failed.

### **Oscilloscope: Some channel settings weren't copied, if a channel is dragged to another area (ID2535)**

- Some channel settings weren't copied when dragging a channel from one oscilloscope area to another. E.g. affected settings were line width and line style.

### **WinDarab uses the first available dist-source channel (ID2532)**

- In previous version of WinDarab, WinDarab looked through the list of the "distance" special channels and took the first available channel/method to calculate the distance information which results to a distance > 0.  
This could lead to the problem that for some files (mostly if the car didn't move) to use another distance method making it impossible to open these files together with others as a linked file.
- Now we removed to validation "distance > 0", so WinDarab always uses the same input channel to obtain the distance information. All files with the same recording setup can be opened in a linked file now.
- Note: Some other validation checks (e.g. monotone increasing distance) are still done and cause an incompatibility – but these checks should normally (never) happen ...

### **Open a linked file fails because of incompatible (virtual) channels (ID2528)**

- If files contain virtual channels (here: setupsheet channels) using a unit for which an active physical unit converter exists, opening the files as a linked file could fail because of "incompatible" channels. The issue was reproducible with the following usage sequence:
  - A file is opened as a single file
  - The active unit converter set is changed
  - A linked file is opened containing the first file as a member

### **COMApi: WinDarab could crash when closing a file (ID2511)**

- If COMApi is used to process data, especially opens/closes data files, it could happen that the delayed action to "save the comment data" wasn't executed before closing the file. Some code to responsible to execute the delayed action earlier/immediately was buggy and caused a crash.

## **WinDarab v7.07.005 22.08.2019**

### **WinDarab v7.06.026 22.08.2019**

### **WinDarab stopped responding if new imported files are opened by multiple WinDarab users at the same time (ID2544)**

- If new WinDarab file(s) (just imported by DataLoggerImport) are opened by multiple WinDarab users simultaneously for the first time in the same moment (on different computers), WinDarab could enter a deadlock scenario and stopped responding.

### **Math inspector window crashed, if a new desktop is loaded (ID2530)**

- If a new desktop is loaded while a math inspector window is opened, the math inspector let WinDarab crash.

### **DataAccessorConfig instance caused crash, if a channel is removed from a file (ID2530)**

- If a channel is removed from a file (e.g. math channel, virtual channel) while a DataAccessorConfig instance is existing (in the observed case by a x-y-plot) the DataAccessorConfig caused a crash even the Channels collection didn't contain the removed channel.

### **Open a linked file fails because of incompatible (virtual) channels (ID2528)**

- If files contain virtual channels (here: setupsheet channels) using a unit for which an active physical unit converter exists, opening the files as a linked file could fail because of "incompatible" channels.

The issue was reproducible with the following usage sequence:

- A file is opened as a single file
- The active unit converter set is changed
- A linked file is opened containing the first file as a member

## WinDarab v7.07.004

30.07.2019

### Math function supports scripting and variables (ID2527)

- Now, math function can be defined by multiple expressions which are executed like a small script. The last expression which is evaluated returns the final value of the math function.
- Additionally math function support “local” and “global” variables – actually required to take advantage of the scripting.
- An assign operator “:=” was added to assign a value to a variable.
- “Local variables”:
  - They are valid in the whole math function but their values survive only for a single evaluation of the math function.
  - With their declaration they are initialized with zero - or the result of the initialization expression.
- “Global variables”
  - They are valid in the whole math function but their last value can be used in the next function evaluation
  - With their declaration they are initialized with zero - or the result of the initialization expression.
  - BUT this initialization is only done once when evaluation the first sample of the math function!
  - In further evaluations the initialization is skipped and the latest value is still there!

Example:

```
Global wheelSpeedSum
Global wheelSpeedCount
Local wheelSpeedAve := (vwheel_fl + vwheel_fr) / 2
wheelSpeedSum := wheelSpeedSum + wheelSpeedAve
wheelSpeedCount := wheelSpeedCount + 1
wheelSpeedSum / WheelSpeedCount
```

- **Note:** You can even use script blocks within any other function argument. Simple put your expressions within “begin” and “end”

Example:

```
Global wheelSpeedSum
Global wheelSpeedCount
If (nmot >= 5000;
  begin
    Local wheelSpeedAve := (vwheel_fl + vwheel_fr) / 2
    wheelSpeedSum := wheelSpeedSum + wheelSpeedAve
    wheelSpeedCount := wheelSpeedCount + 1
  end; 0)
wheelSpeedSum / WheelSpeedCount
```

**Note:** We know, this is not as nice as a programming language – but it’s an option how to do it today !

### Math function “ValueAtDist” and “ValueAtTime”: These function were added/extended (ID2489)

- The math function “ValueAtDist” was added:

```
ValueAtDist(<channel>; <dist> [; <LapIndex>])
```

If <LapIndex> is omitted, <dist> is the absolute distance value within the file.

➔ ValueAtDist(<channel>; <xdist>-10) returns the value 10m before the current position.

If <LapIndex> is given, the <dist> argument is a lap distance value.

E.g.

➔ ValueAtDist(<channel>; 100; LapIndex) returns the value at position 100m of the current lap.

➔ ValueAtDist(<channel>; lapdist; LapIndex-1) returns the value at the lap position but in the previous lap

- The math function “ValueAtTime” was extended accordingly to ValueAtDist:

```
ValueAtTime(<channel>; <time> [; <LapIndex>])
```

If <LapIndex> is omitted, <time> is the absolute time value within the file.

➔ ValueAtTime(<channel>; <xtime>-10) returns the value 10sec before the current position.

If <LapIndex> is given, the <time> argument is a lap time value.

E.g.

➔ ValueAtDist(<channel>; 10; LapIndex) returns the value at position 10sec of the current lap.

➔ ValueAtDist(<channel>; laptime; LapIndex-1) returns the value at the same laptime position but in the previous lap

- **Note:** WinDarab optimizes the following usage pattern of ValueAtTime

```
ValueAtTime(<channel>; xtime-<delta>)
```

In this case WinDarab doesn't calculate a “real” math channel. Instead WinDarab shifts the timeline accordingly and passes the channel data through – this is almost like accessing the original channel!

### Alias channels: Channel names can contain variable parts to reduce the number of alias channel definitions (ID2495)

- An alias channel name and the names of its possible member channels can contain indexers. WinDarab automatically expands the alias channel definition (internally) to discrete alias channels:
- The indexer can be given as a list of name parts “[part1,part2]” or a range [1..4] or [a-d] or any combination [a,b-d]
- The number of indexers in the alias channel name and its member channel names have to match!

E.g.:

You want alias channels wheelspeed\_fl, wheelspeed\_fr, ...

Now you can create an alias channel definition named “wheelspeed\_[fl,fr]”

And give the following member channels:

```
vspeed_[fl,fr]
```

```
vwheel_[fl,fr]
```

```
vspeed_[0..1]
```

### Overlay coloring: The channel coloring mode can be set individually (ID2494)

- The default/global coloring mode for channels in the oscilloscope can be overridden for an overlay individually.

**WinDarab v7.06.025**  
**30.07.2019**

### Math function “Sigma” could show wrong values

- We observed an issue where the math function sigma didn't calculate correct values, because of an invalid internal state of the Sigma function.

This invalid state was caused because an outer math function continued to calculate its values even a cancellation request is active. In that case the sigma function state is more far in the calculation of the math function than the values are actually saved. If the math calculation later restarted, the internal state with the wrong values was reused.

#### **Oscilloscope: Sample dots became invisible again when zooming very deep**

- If the user zooms deep into an oscilloscope, WinDarab normally shows the samples dots automatically. In some cases it could happen that the sample dots disappeared if the user continued to zoom in.

#### **WinDarab crashed/dead-locked when opening a linked file (ID2526)**

- When opening linked files an unprotected access to a internal global list of opened XML documents caused WinDarab to crash/stop responding. Actually the global list was for test purposes (and forgotten a long time ago) – we simply removed the code lines...

#### **WinDarab crashed when connecting to a telemetry stream (ID2524)**

- This issue happened because the channel list was changed while the channels pane was drawing its content. We don't have any idea why the channel list has changed – we can see a recursive call to synchronize the channel list with the channels pane, but we have no idea how this happened ...
- Now redrawing any parts of the channel list is suppressed as soon as synchronizing the channels pane is requested (or a recursive call occurs).

#### **Editing comment fields wiped out the comment field configuration (ID2523)**

- Replacing a system function by a “secure” variant caused this issue – while a years old bug didn't take any affect: The comment field file was created and opened 5 times and then the content was written for the last created file. The “secure” function fails on the second file open and WinDarab fails to write the settings!

#### **WinDarab crashed if telemetry stream is interrupted unexpectedly (ID2522)**

- The issue happened if the disconnect occurs while retrieving the timeline from WDServer right before reading the channel values. We tried to address this issue, but we were not successful to reproduce this issue...

#### **Math function “ave” didn't work for telemetry streams (ID2518)**

- Because of an implementation bug the average function calculate obviously wrong values for telemetry streams. The bug was introduced when changing the internal WinDarab architecture.

#### **Setupsheet channels and physical unit conversion didn't work together (ID2516)**

- If persistent setupsheet channels are defined using units for which a unit converter exists, the persistent setupsheet channels mixed units and physical values.
- Additionally, in case of file groups, the single files could be opened, but not as a file group, because at least one file applied the unit converters but not all.

#### **Moving a window from one display to the other wasn't possible (ID2515)**

- In a multi-display environment where displays have different resolutions and text scaling, it could happen that a window couldn't be moved to another display.
- The issue was observed if a high resolution display is located above a lower resolution display.
- Note: We removed the code to ensure the visibility of windows in a multi-display environment, because in case of different resolutions and scalings Windows “skips” screen pixels on the whole display area and lets Windows “virtually jumping” around. And this is actually the problem, the code cannot handle ...

#### **Ave-Function didn't include the last sample (ID2510)**

- We implemented new statistic function which are used by all windows in common.
- Especially we readded the code to weight the statistics “by sample”. Weighting “by sample” was

- removed when changing the internal architecture or WinDarab.
- Note this about weighting samples in statistic calculation:
    - Info window: Uses weight “by xaxis” (xtime/xdist). Also this windows uses “virtual samples” located at the left/right border of the visible range!
    - Min/Max-Window: Uses “weight by sample” without virtual samples at lap begin/end
    - Outing-Report: Uses “weight by sample” without virtual samples at lap begin/end
    - Math function “average”: Weight “by sample” without any virtual samples

#### **WinDarab v7.06.024** **15.07.2019**

##### **Oscilloscope: Channel colors couldn't be changed sometimes (ID2514)**

- Sometimes changing channel colors in oscilloscope didn't work because of an uninitialized variable.  
Some code changes were accidentally checked-in to v7.6 and not reverted after the feature implementation was moved to v7.7.

#### **WinDarab v7.06.023** **15.07.2019**

##### **DataLoggerImport: Fails to convert VCU recordings (ID2513)**

- In some cases the conversion of VCU recordings fails and the WinDarab File cannot be used. The issue happens, if the DCP information is written to the WinDarab file and is caused by the compressed stream class which can fails if the compressed data is flushed to the output buffer/file.

##### **Ave-Function didn't include the last sample (ID2510)**

- The math function “ave” (and min/max) didn't include the last sample of the input channel.

##### **ASCII-Import didn't work any more (ID2509)**

- The ASCII-Import to create WinDarab files didn't work anymore and is fixed now.

##### **Closing a file: Some issues with processing timeline update events fixed (ID2507)**

- WinDarab could crash if a file is closed because timeline update events were processed for already freed timelines.

##### **BMS2API: Crash with “bad programmed” external code fixed (ID2501)**

- Internal COM-instance was created for DataFile after it was freed (and should be freed)
- Bug/Crash/Assert in password protecting code found and fixed

##### **Load/Save setting dialog: Caption text was wrong (ID2493)**

- The caption of the load/save settings file dialog shows the wrong text.

##### **“Adjust y-range dialog”: Textboxes show exact values when focused (ID2492)**

- The “Adjust y-range” dialog to set the view min/max for a channel in the oscilloscope was reimplemented with MFC. The value textboxes show the formatted value (according to the channel number format) but shows the actual exact value as soon as the textbox receives the input focus. The user also has the ability to enter more decimals as the number format defines.

##### **x-y-Plot: Polynomial fitting curves were drawn as a polygon (ID2481)**

- A polynomial fitting curve was drawn as a polygon – thus the leftmost and rightmost points of the curve were connected with a line. Now the curve is drawn as a polyline – without connecting the end with the beginning.

##### **x-y-Plot: Plot wasn't updated immediately after changing the overlay color (ID2482)**

- If the user changed the color of an overlay, the x-y-Plot window didn't update immediately (in simple mode the overlay color is used by the plot)

#### **Channel info window: Resizing the first column didn't behave well (ID2480)**

- If the first column (row header column) is resized, the datagrid scrolled a little bit strange...

### **WinDarab v7.06.022** **03.07.2019**

#### **Telemetry fix: WinDarab reconnects endless to WDServer after adding "lapno\_tele" to any view**

- "lapno\_tele" is an alias name which is used to resolve name conflicts between the telemetry channel "lapno" and the internal WinDarab channel "LapNo".  
But if the user tried to add "lapno\_tele" to a view (regardless which), WinDarab lost the connection to "WDServer", reconnects and loses the connection again – endless.  
The issue was triggered because WinDarab tried to retrieve the channel data for "lapno\_tele" by using this channel name and not the original WDServer channel "lapno". WDServer returned a failure which caused WinDarab to reconnect (to validate the channel names)

#### **Channels pane: Sometimes multiple channels are selected while typing a name in the filter textbox (ID2503)**

- If the user types a channel name into the filter textbox in the "Channels" pane, WinDarab sometimes selected multiple entries and not only the exact/best match.

#### **BMS2Api: Cannot open telemetry streams (ID2501)**

- A client application using the BMS2Api can open telemetry streams.

#### **Oscilloscope: When dragging a channel into the oscilloscope, the insertion marker isn't at the correct position sometimes (ID2500)**

- If the user dragged a channel into an area of an oscilloscope view and the channel value list on the right side is scrolled, the scroll offset wasn't taken into account when drawing the insertion marker.

#### **Oscilloscope: When the users adds a channel to an oscilloscope the channel is always added for all overlays (ID2496)**

If a channel is added to an oscilloscope (from the channels pane) the channel was added for all overlays but only if the channel existed in the file assigned to the overlay.

Now the channel is added for all channels always. If a file doesn't contain a channel with the given name, a dummy channel is created.

### **WinDarab v7.07.003** **03.07.2019**

#### **Show/Hide overlay: The overlay orb in the "File Explorer" pane wasn't updated immediately (ID2502)**

- If an overlay is shown/hidden using the "Show/Hide" context menu in the oscilloscope, the orbs in the "File Explorer" pane wasn't updated immediately but after moving the cursor in the oscilloscope (or hovering over the "File Explorer" pane)

#### **Setupsheets can be used for telemetry streams (ID2464)**

- Now setupsheets can also be configured for telemetry streams. The setupsheet file (setupsheet.xml) has to be located in the folder ..,\Config\WDServer".

#### **New replay option: "Auto-align running laps" in the ribbon menu of the "Replay" button (ID2475)**

- If the option "Auto-Align running laps" is enabled all running laps of telemetry streams are automatically aligned and the oscilloscope is scrolled so that the aligned laptrigger is located at the left edge of the oscilloscope.

- As soon as the next laptrigger is received, the new lap is automatically aligned to the other laps.
- **Note:** While replaying the cursor is always moved to/for the overlay which is the closest to the right side of the oscilloscope. This ensures that all values (channels pane, analysis windows) are evaluated for the current values.

## WinDarab v7.06.021 03.07.2019

### Telemetry fix: WinDarab reconnects endless to WDServer after adding “lapno\_tele” to any view

- “lapno\_tele” is an alias name which is used to resolve name conflicts between the telemetry channel “lapno” and the internal WinDarab channel “LapNo”.  
But if the user tried to add “lapno\_tele” to a view (regardless which), WinDarab lost the connection to “WDServer”, reconnects and loses the connection again – endless.  
The issue was triggered because WinDarab tried to retrieve the channel data for “lapno\_tele” by using this channel name and not the original WDServer channel “lapno”. WDServer returned a failure which caused WinDarab to reconnect (to validate the channel names)

### Channels pane: Sometimes multiple channels are selected while typing a name in the filter textbox (ID2503)

- If the user types a channel name into the filter textbox in the “Channels” pane, WinDarab sometimes selected multiple entries and not only the exact/best match.

### BMS2Api: Cannot open telemetry streams (ID2501)

- A client application using the BMS2Api can open telemetry streams.

### Oscilloscope: When dragging a channel into the oscilloscope, the insertion marker isn’t at the correct position sometimes (ID2500)

- If the user dragged a channel into an area of an oscilloscope view and the channel value list on the right side is scrolled, the scroll offset wasn’t taken into account when drawing the insertion marker.

### Oscilloscope: When the users adds a channel to an oscilloscope the channel is always added for all overlays (ID2496)

- If a channel is added to an oscilloscope (from the channels pane) the channel was added for all overlays but only if the channel existed in the file assigned to the overlay.  
Now the channel is added for all channels always. If a file doesn’t contain a channel with the given name, a dummy channel is created.

## WinDarab v7.07.002 25.06.2019

## WinDarab v7.06.021 25.06.2019

### Gerrman math functions “Ganz” and “Wert” removed (ID2490)

- Int( <value> ) is the equivalent math function for “Ganz” to cut the decimals of a double value.
- Wert( <string> ) is a text function which was provided by the underlying math expression parser but cannot be used in WinDarab (WinDarab doesn’t support string operations)

### Math inspector crashed if the inspected math formula accessed a lookup table and the lookup table id edited/saved (ID2485)

- If the user opened a math inspector to inspect a math channel accessing a lookup table and the user edited the lookup table, WinDarab crashed as soon as the changed lookup table is saved.
- **Note:** The crash only occurred for one-dimensional lookup tables.

### COMApi/BMS2Api: Crash if trying to export data into a new measurement file (ID2483)

- WinDarab crashed after exporting data to a new measurement file using DataFileExport.DoExport() because an object was released twice.

### WinDarab crashed if laptime/lapdist channels are used (ID2479/2497/2499)

- WinDarab crashed as soon as the laptime or lapdist channel is added to an oscilloscope (or somewhere else).

## WinDarab v7.07.001 12.06.2019

### New feature “Hide overlay” (ID2445)

- The visualization of an overlay can be disabled (hidden) – while the file keeps opened and attached to the overlay.
- To hide an overlay use the overlay context menu in the file explorer. An overlay is shown again, if you select the same context menu item or – simply – by clicking into the overlay column to select a lap.  
Alternatively you can use the “Hide” context menu in the oscilloscope (“H” key).
- What happens, if an overlay is hidden:
  - ➔ The oscilloscope hide all channel of hidden overlays
  - ➔ The trackmap doesn't show the position of the car.
  - ➔ All analysis windows hide the evaluation results of hidden overlays (except LapAnalysis)
  - ➔ As long as an overlay is hidden most calculations for the attached files are stopped/disabled. Only “Event rules” and “LapAnalysis” is still evaluated.

### Math function “DetectEvent” supports “Delay” times (ID2427)

- Two additional arguments “BeginDelay” and “EndDelay” were added to DetectEvent. With these arguments DetectEvent delays state changes by the given amount of time.
- The two arguments are optional and don't change the signature of the “old” DetectEvent:

```
DetectEvent(Inactive|Rising|Active|Falling; <BeginCondition>;
<BeginDuration>; <BeginDelay>; <EndCondition>; <EndDuration>; <EndDelay>)
```

- The kind of the arguments are actually controlled by the given number of arguments. Hereby the first keyword argument (Inactive/Rising|Active|Falling) is always optional and is NOT counted as an argument!

#### DetectEvent with 6 Arguments

```
DetectEvent(Inactive|Rising|Active|Falling; <BeginCondition>;
<BeginDuration>; <BeginDelay>; <EndCondition>; <EndDuration>; <EndDelay>)
```

#### DetectEvent with 5 Arguments

```
DetectEvent(Inactive|Rising|Active|Falling; <BeginCondition>;
<BeginDuration>; <BeginDelay>; <EndCondition>; <EndDuration>; <EndDelay>)
```

#### DetectEvent with 4 Arguments

```
DetectEvent(Inactive|Rising|Active|Falling; <BeginCondition>;
<BeginDuration>; <BeginDelay>; <EndCondition>; <EndDuration>; <EndDelay>)
```

#### DetectEvent with 3 Arguments

```
DetectEvent(Inactive|Rising|Active|Falling; <BeginCondition>;
<BeginDuration>; <BeginDelay>; <EndCondition>; <EndDuration>; <EndDelay>)
```

### DetectEvent with 2 Arguments

```
DetectEvent (Inactive|Rising|Active|Falling; <BeginCondition>;  
<BeginDuration>; <BeginDelay>; <EndCondition>; <EndDuration>; <EndDelay>)
```

### DetectEvent with 1 Arguments

```
DetectEvent (Inactive|Rising|Active|Falling; <BeginCondition>;  
<BeginDuration>; <BeginDelay>; <EndCondition>; <EndDuration>; <EndDelay>)
```

## WinDarab v7.06.020 12.06.2019

### Track map: Car position not updated in case of some actions

- The position of the cars on the track map wasn't updated in some cases:
  - When replaying a offline file
  - If the user clicks/moves the selected overlay/car in the track map, the other overlays/cars aren't repositioned in the track map.

### Ribbon check button: Check state wasn't "unchecked" after it was once "checked"

- Because of some changes in v7.06.017 a check button in the ribbon wasn't unchecked after the button was checked for the first time.

### Flow chart: The evaluation range "whole file" wasn't set correctly for not-active overlays

- If a flow chart is used to evaluate the "whole file" of a not-active overlay, the uses evaluation range wasn't set correctly.
- **Note:** The bug was introduced by the architecture changes between v7.5 and v7.6.

### Analysis views: The evaluation range "visible" wasn't set correctly for multiple overlays

- If an analysis window (x-y-plot, histogram, ...) is configured to evaluate the "visible range", the correct range was only evaluated for the active overlay. For other overlays the range was somewhere else.
- **Note:** The bug was introduced by the architecture changes between v7.5 and v7.6.

### WinDarab responds laggy to user interaction (ID2256)

- The AutoSave feature in WinDarab which saves the current desktop to continue with after a crash consumes much time if the desktop is very large. Because many user interactions trigger the AutoSave, the user experience a laggy behavior of WinDarab.  
We changed several aspects of the AutoSave feature:
  - ➔ The AutoSave delay was extended to 1 second after the last trigger action.
  - ➔ AutoSave is only done, if the mouse isn't moved within a second

## WinDarab v7.06.019 30.05.2019

### BMS2API: Initialization of BMS2Api returned error 0x800700=% (ACCESS\_DENIED)

- A change in the license check sequence caused BMS2Api to always return ACCESS\_DENIED.

## WinDarab v7.06.018 29.05.2019

**Note: The work on WinDarab v8.00 is officially released as v7.06. This document was changed accordingly!**

### x-y-Plot: "Very huge" grip points for user lines (ID2465)

- If the user add user lines to a x-y-plot, WinDarab draws very large grip points. The underlying rectangle structure type was changed, but still the right/bottom coordinates were passed to initialize the rectable while the new type expects the width and height...

#### **Executable description of “Darab.exe” contains full version information (ID2463)**

- Because the task manager can show the “executable file description” (but not the file version) we have added Dthe full version number of WinDarab to the executable description.

### **WinDarab v7.06.017 22.05.2019**

#### **Outing report doesn’t show values (ID2456)**

- If a channel contains negative values only, Outing Report doesn’t show lap results if “Evaluation of segmentation” is enabled. The bug is fixed.

#### **Oscilloscope: Adjust x-axis range in x-axis context menu**

- If the user opens the x-axis context menu in an oscilloscope, some menu items were added to adjust the visible x-axis range by entering values.
  - “Start”: The x-axis is scrolled and starts exactly at the entered position (without changing the x-axis range)
  - “End”: The x-axis range is modified to show exactly the current start up to the entered end value.
  - “Range”: The x-axis range is modified to zoom exactly to the entered value. The start and end position of the x-axis is choosen in a way that the current cursor screen position is NOT changed (-> start and end are adjusted accodingly)
- **Note:** The values for start and end are automatically adjusted, if they go out of allowed x-axis range (or if start > end).

#### **File Groups show 10 sec gap between member files (ID2454)**

- If a file group is opened with WinDarab v7.6 files (currently only created by VCU devices) a small bug caused WinDarab to add a gap of 10 secs between each member file, if there isn’t already a recording pause between.

#### **x-y-Plot crashed when evaluation current/previous lap with preceeding laps > 0 (ID2451)**

- If a x-y-plot should evaluate the current/previous lap including the <x> preceeding laps, WinDarab crashed, if the number of preceeding laps cannot be fullfilled:
  - x-y-plot evaluating current lap and 2 preceeding laps
  - Current lap is #3 -> works (because lap #1 and #2 are available)
  - Current lap is #2 -> crash (because there’s only lap #1)

#### **Shifting a file moves the tag marker to another position (ID2450)**

- With WinDarab v7.6 the tag marker position is internally stored as a timestamp - it’s not necessary any more to update the tag marker position if the file is shift.

#### **File shift by x-delta value (ID2448)**

- In the oscilloscopes context menu (middle or left-right click in the oscilloscopes signal area) a new menu item “Shift file by” with a textbox was added. If the user enters a value and presses the return key (or clicks on the menu item itself), the current file is shifted by the entered x-axis offset.
- With every execution of the action the current file is shifted again.
- To close the context menu, the user can press ESC or click outside of the context menu.
- To shift to the right, the x-axis offset has to be positive. If the value is negative, the file is shifted to the left.

#### **Ribbon panels renamed (also ID2446)**

- Some panels in the ribbon have been renamed (e.g. “settings” to “view settings”)

### Changing the attributes of a lap didn't work as expected (ID2443)

- If the user tried to change the attribute of a lap (e.g. In-Lap, Out-Lap, ...) for a file group, WinDarab added a manual lap trigger at the end of a member file. In the first moment nothing happened, but after reloading the file group a user-defined lap appeared with lapttime 0s which even couldn't be deleted any more.
- The bug was fixed also WinDarab ignores manual laptriggers at the end of a file (-> the accidental added user laps disappear / are deleted automatically)

### Fix: Memory leak and other issues in internal path/folder operations (also ID2442, ID2453)

- Some changes in the internal handling of filepaths caused a memory leak and in some cases even crashes. These bugs were fixed.

### Fix: WinDarab crashed when closing a file containing not-local-defined event rules

- If a file contains persistent events which are not defined locally, they are removed WinDarab, if the file is closed. But in that case WinDarab updates the UI too early and the "File event" dockable pane tries to access already freed objects.

### WinDarab crashed when loading the docking state (ID2439)

- A crash was observed during loading a docking state (which actually can happen because of some problem in the MFC libraries). WinDarab couldn't be started anymore.
- To prevent continuous crashes when loading the docking state, WinDarab tracks and disables the loading of the corrupt docking state on the next startup. This function was added a long time ago.
- Nevertheless the observed crash occurred in a phase, we didn't expect these crashes to happen. So we extended the crash detection to the additional code section.

### Math function DetectEvent: Optional flag to switch DetectEvent into a conditional function

- Normally DetectEvent returns a value identifying the current state of the data evaluation:
  - 0 - Inactive: Searching for BeginCondition to become true
  - 1 - Rising edge: BeginCondition detected, now beginning to search for EndCondition to become true
  - 2 - Active: BeginCondition was found and now searching for the EndCondition to become true
  - -1 - Falling edge: EndCondition was found, now beginning to search for the occurrence of the BeginCondition
- In most cases the user wants to search for certain states of DetectEvent. That means that the user had to compare the result of DetectEvent with those constants above. Depending on the desired states, this could be more or less difficult (e.g. `abs(DetectEvent(...)) == 1` was required to become true for each rising or falling edge)
- Now we added an optional argument to DetectEvent() which can be used to define in cleartext which states are of interest:

```
DetectEvent(Inactive|Rising|Active|Falling; <BeginCondition>;  
<BeginDuration>; <EndCondition>; <EndDuration>)
```

One or more state keyword can be combined using a pipe character '|'.  
'|'

- Examples
  - Detect event while "nmot >= 5000" for more than 1000ms until nmot < 5000 for more than 100ms.  
New: `DetectEvent(Rising|Active; nmot >= 5000; 1000; 100)`  
Old: `DetectEvent(nmot >= 5000; 1000; 100) > 0`
  - Detect a short trigger event if "nmot >= 5000" for more than 1000ms and then again if nmot < 5000 for more than 100ms.  
New: `DetectEvent(Rising|Falling; nmot >= 5000; 1000; 100)`  
Old: `abs(DetectEvent(nmot >= 5000; 1000; 100)) = 1`

### BMS2API: ChannelAccessor continued (ID2440)

- If a channel is deleted/removed or the file is closed, a ChannelAccessor disposed itself automatically.

## WinDarab v7.06.016 24.04.2019

### **BMS2API: New class ChannelAccessor introduced**

- The Channel object offers a new method AcquireAccessor() which returns an accessor wrapper for the channel.
- The ChannelAccessor loads and keeps the channel data in memory as long as some ChannelAccessors refer to a channel. This can reduce expensive recalculations.
- ChannelAccessor also gives low-level access to channel data, by getting single values by sample index or timestamp.
- The channel data is released from memory as soon as all ChannelAccessors (created directly or indirectly by the client; e.g. DataAccessor) are released.
- In Framework.Net environment we advise to use Dispose() to have a deterministic point of releasing memory. Otherwise Dispose() will be executed, if the .Net Framework GarbageCollects all objects.
- **Known issue:** A ChannelAccessor.Dispose() has to be called currently because there's not yet a solution that it's automatically disposed, if the channel is deleted or the DataFile is closed!

### **BMS2API: Threading model changed to "free" (ID2435)**

- Threading model of BMS2Api is changed to "Free". This enables multithreaded usage of BMS2API functions.
- Note: The client is responsible to dispatch events raised by BMS2API to the UI thread of the client application (if necessary)
- Parallelism support of BMS2API improved.  
Note: Opening DataFiles from worker threads seems to work, but currently this cannot be guaranteed. For production environment we recommend to open DataFiles from a single thread. But for testing purpose we also want to encourage to open DataFiles also in WorkerThreads!

### **Math/Timeline conditions slowed down WinDarab (ID2431)**

- A change of the calculation timeline of conditions caused a performance degrade of WinDarab. We returned the old behavior and need to think about how to handle this issue in future.

## WinDarab v7.06.015 12.04.2019

### **Oscilloscope support different dashed/dotted line styles (ID2343)**

- The line style of a channel signal can be changed to be dashed/dotted. For testing purpose five styles are available beneath the "simple line" style.

### **Oscilloscope: Copy the selected channel value to clipboard (ID2426)**

- The value(s) of the selected channel(s) can be copied into the clipboard (Ctrl-C or select "Copy values" in the context menu)

### **Oscilloscope: The movement distance to start dragging adjusted to display scaling (ID2426)**

- The mouse cursor movement distance to start a drag or click-and-hold operation is scaled accordingly to the current display scaling. Before the values were fixed and the caused some operation to start accidentally.

### **Union-Timelines: The algorithm to prepare a union-timeline is changed**

- The algorithm to prepare a union-timeline was changed/simplified. Because of a higher level of parallelism we hope also an improvement.  
Time measuring is very difficult because the times change a lot from test to test, because of other activities and influences within the concurrency library...

#### **Math channel: Renaming a math channel caused a dot to be appended to the name (ID 2424)**

- If the user renames a math channel, WinDarab appended an additional dot at the end of the filename.

#### **Math inspector can cause WinDarab to stop responding for a while) (ID 2423)**

- If math inspector gets a value for a math channel for the current position, but the math channel value isn't available yet, the math inspector blocked the execution of WinDarab. Now math inspector show "NaN" (like the value display in the oscilloscope). The value will be updated as soon as it becomes available,

### **WinDarab v7.06.014 09.04.2019**

#### **Math channels using average function weren't calculated correctly (ID2422)**

- If the average function was used in a math channel, the calculation wasn't correct over the whole file. Instead the calculated was correct as long as the accessed data wasn't calculated starting at the beginning of the file.  
Note: Depending on the influence of user requested data calculation and background calculation the range of the wrong values deferred – but the error always started at the beginning of the math channel.

#### **Setupsheet channels weren't initialized correctly in linked files (ID2421)**

- Setupsheet channels weren't initialized correctly for linked files. Instead only the first setupsheet channel contained data while the sample count for all other channels was set to zero samples.

#### **Fix: Possible crash fixed, if measurement file contains dynamic statusblock data**

- After a file was opened which contains a dynamic statusblock, a crash could occur if the user closes WinDarab.

#### **BMS2API Fix: Catching concurrency exception on shutdown**

- If a 3<sup>rd</sup> party application ends which has used BMS2API, the BMS2API library could crash because of an unhandled exception while shutting down the concurrency library.

### **WinDarab v7.05.068 09.04.2018**

#### **Spelling error of "hysteresis" fixed (ID2420)**

#### **Math channels using LapSigma returned bad values (ID2419)**

- If LapSigma is used in a math channel which is calculated slower than the time axis, it could happen that the value was reset to 0 some time after the laptrigger and not immediately at/after a laptrigger.

#### **Instrument panel: Alert setting "no blinking" wasn't saved in the configuration file (ID2419)**

- If the user tries to set "no blinking" on alert for an instrument panel gauge, the setting wasn't saved to the configuration file and was lost after reloading the configuration.

#### **Instrument panel: Background color setting for gauges was ignored (ID2415)**

- We added a long time ago the possibility to change the background color for the different value/alert zones of instrument panel gauges. Nevertheless some test code was added and the background color feature got disabled up to now ...

#### **Oscilloscope could crash, if last channel is removed from the bottom-most area (ID2413)**

- Under certain conditions the oscilloscope crashed, if the last channel was removed from the bottom-most area.

### COMApi: A not-release DataAccessorConfig object could cause a crash (ID2369)

- WinDarab crashed if a DataAccessorConfig was created and kept alive longer than the accessed data file.

## WinDarab v7.06.013

15.03.2019

### COMApi/BMS2Api: Time range end wasn't respected exactly

- If data is retrieved using a DataAccessor or Channel.GetPhysicalValuesRange, the end of the time range wasn't respected exactly. In most cases a sample behind the requested end was returned.
- **Note:** Now the behavior is exactly to returns all values BEHIND/AT the beginning of the range and BEFORE the end of the range. If a value is recorded EXACTLY at the end of range, the value IS NOT included in the result.  
If you want this value to be included, you can add a small amount of time. Better use the constant TimestampEpsilon which represents the internal time stamp resolution of 100ns. if you use COM it's a simple constant value 'TimestampEpsilon'. In the WinDarab.Net-Proxy-Assembly it's defined as Timeline.TimestampEpsilon.

### Math Channel: Optimization for fast shifting of a single channel on the timeline

- A special optimization was added for the math function ValueAtTime(), if the usage pattern is:
  - ValueAtSample(<channel>; xtime +/- <constant>)E.g. ValueAtSample(nmot; xtime-1) shifts the channel data by one second on the timeline to the beginning of the file
- Because WinDarab v7.6 takes advantage of architecture changes compared with WinDarab v7 a shortcut was possible to implement for this kind of math channels with the following advantages:
  - No calculation required, the math channel is immediately available
  - Very small memory footprint because WinDarab doesn't need to remember previously calculated values
  - Exact time shifting because the timestamps of the shifted channel are shifted and not the values itself.
- WinDarab uses the optimization if the following conditions are true:
  - The timestamp (second argument) of ValueAtTime is a subexpression of the format
    - "xtime + <timeshift>" or
    - "xtime - <timeshift >"
  - <timeshift> is a constant expression (scalar value or comment field)
  - <timeshift> cannot be any kind of subexpression (e.g. "100/1000" instead of "0.1")
  - <timeshift> is given in seconds
  - The math channel is calculated for the timeline of the shifted channel.  
If computing is set to "Periodic", "Use condition" or not "every 1. Step of slowest/fastest channel", ...
- This optimization can be used to shift the channel values towards the beginning or end of file and can also be used with telemetry streams.

### COMApi/BMS2API: DataFile.GetShiftedTimeLine() added

- The new method DataFile.GetShiftedTimeLine() returns a new timeline which shifts all timestamps of the base timeline by the given time shift value.

## WinDarab v7.06.012

12.03.2019

### COMApi/BMS2API clients don't have to use Channel.Wait to ensure data availability before retrieving channel values (ID2408)

- WinDarab v7.6 may start with an empty math channel, if the math channel is based on a timeline which has to be build first (e.g. a conditional timeline). This behavior was introduced to reduce

- blocking when accessing such a channel in WinDarab v7.6.
- While WinDarab v7.6 listens for the success of building the timeline and automatically update the UI-
- COMApi(BMS2API based clients don't use this strategy because it's more difficult to implement. To reduce the need to change existing code, we added the code to wait for a channel to all functions (like Channel.GetPhysicalValuesAll or DataAccessorConfig.CreateDataAccessor).
- Nevertheless clients have the opportunity to call Channel.IsReady and use the NotificationEvents to react/continue the evaluation as soon as the data becomes available!

#### **WinDarab crashed, if a math channel is shown using a periodic timeline for a linked file (ID2408)**

- WinDarab crashed, if the user opened a linked file and tried to show a math channel which is based on a periodic timeline.

#### **Setup sheet channels sometimes not populated correctly for linked files (ID2408)**

- If a linked file is opened and a new setup sheet channel is added to the setup sheet, the values of the new channel weren't populated correctly.

#### **Math channel using with ValueAtTime() showed invalid argument error (ID2406)**

- Math channels using ValueAtTime showed an invalid argument error for the second argument. The second argument is allowed to be any subexpression returning the time position a channel value is requested for but WinDarab checked for a valid channel name.

#### **WinDarab v7.6 supports the new WDServer.Net**

- We started to implement WDServer.Net (a descendant of WDServer v2). WDServer.Net uses an extended communication protocol (not supported by WinDarab v7 or prior) which mainly addresses the way how channel recording and their timelines are handled.
- These changes reflect the same architecture changes between WinDarab v7.6 and WinDarab v7. Similar to WinDarab v7.6 WDServer.Net supports independent timelines for the recording channels. There's no need of an increasing global timeline anymore!
- One of the benefits is that incoming data can be organized in packets and only the timestamps of packets containing the same channels have to be in-order. But different channels can be transmitted with out-of-order timestamps. This enables WDServer.Net and WinDarab v7.6 to support the High-Speed-Telemetry protocol (HST) beside to the previous FM40 protocol (serial/modem transmission)!

#### **Updating the readme file started**

- Beginning with v7.06.012 we will update the changelog for WinDarab v7.6...