

DTS 8 Monaco Release 10 – Release notes (v1.0)

Softing Automotive Electronics GmbH

Mirko Braun – Project leader DTS 8 Monaco for Daimler
24. November 2015, Haar

New functionalities

- General issues
- HMI Controls
- SWT
- OTX-Studio/OTX-Runtime

Bugs

- Fixed bugs
- Known bugs

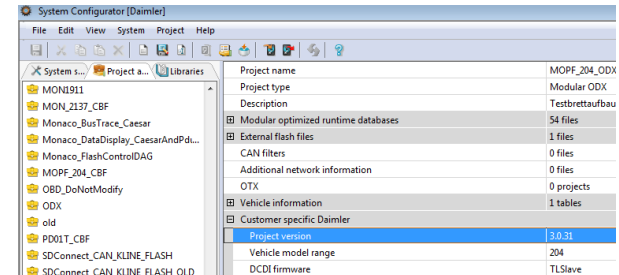
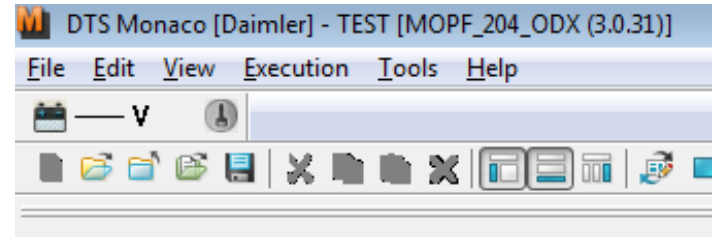
Update of DTS-Projects via DiagCon

Use Case:

- Support an update of a DTSPProject from Daimler portal via the tool DiagCon (MON-2220)

Implementation(I):

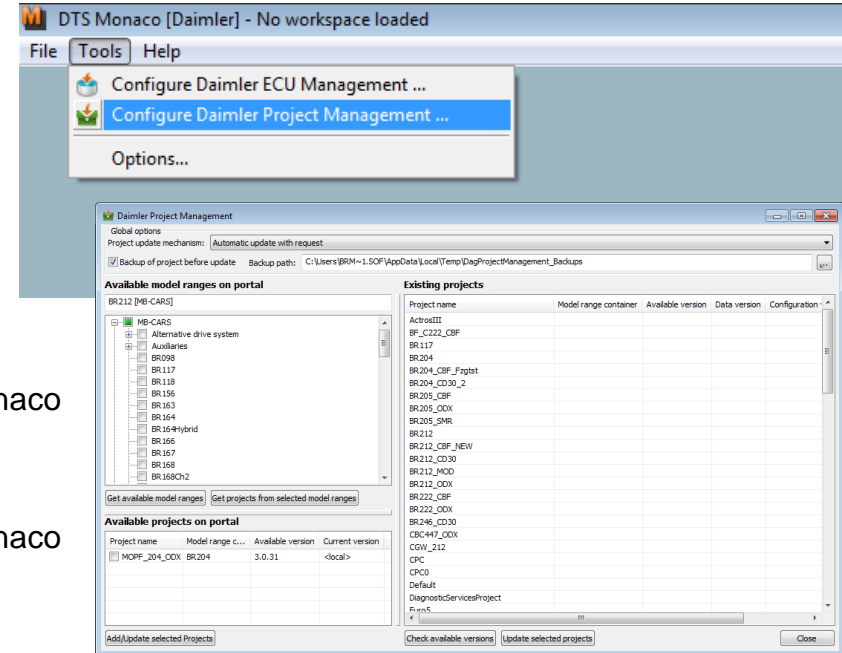
- Which DTSPProjects can be updated:
 - “Online“ DTSPProjects
 - DTSPProjects which were downloaded from the Daimler portal initially.
Those projects contain a version number (shown in DTS 8 Monaco and System Configurator).
 - “Local” DTSPProjects
 - DTSPProject which were **not** downloaded from the Daimler portal initially but created locally on the users pc.
 - Precondition: DTSPProject must have the same name as a project on the Daimler portal.



Update of DTS-Projects via DiagCon

Implementation (II):

- 3 update mechanism available:
 - Manual update
 - Update via a dialog “Daimler Project Management”, accessible via menu
“Tools” → “Configure Daimler Project Management”
 - Dialog provides possibility to change the update mechanism
 - Automatic update with request
 - Before a selected DTS-Project is loaded in DTS 8 Monaco the user is asked if the update should be done.
 - Automatic update without request
 - Before a selected DTS-Project is loaded in DTS 8 Monaco the user is **not** asked if the update should be done.
The update is done before the DTSPProject is loaded automatically.

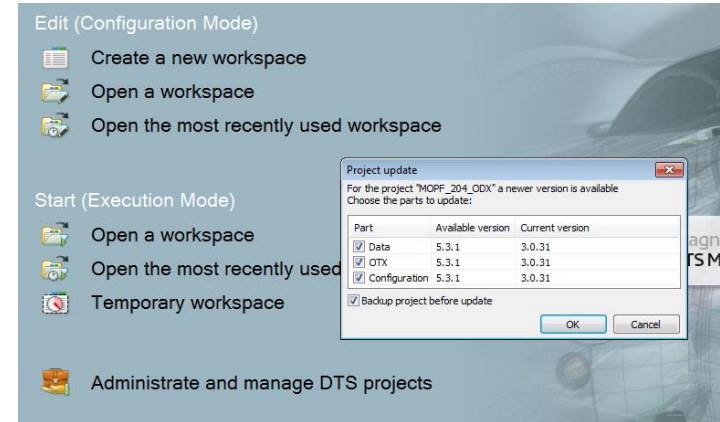


Update of DTS-Projects via DiagCon

Implementation (III):

- Separate update of 3 areas of a DTS-Project possible:
 - Area 1: “Data”
 - Content of the subfolders “cbf“, “dbr” and “db” of a DTS-Project
 - Area 2: “OTX”
 - Content of the subfolder “otx” of a DTS-Project
 - Area 3: “Configuration”
 - All subfolders of a DTS-Project and their content which are not covered by the first two areas.

- Update of areas possible by manual update of one DTS-Project in the dialog “Daimler Project Management” or automatic update with request.



Update of DTS-Projects via DiagCon

Implementation (IV):

- General aspects
 - **Important:** During an update a DTS-Project is replaced by the one from the Daimler portal completely!
 - Possibility to create an backup of the DTS-Project before the update is done
 - Option available e.g. in the dialog “Daimler Project Management” for a manual update

Import/Export of Layouts

Use Cases:

- Give service configuration (e.g. from an HMI Control Data Display) to a colleague
- Add a Layout during a testing drive with a car
- A user creates a Layout with many Graphical Instruments and wants to give it to a colleague who also needs this Layout. (MON-2191)

Implementation(I):

- General aspects:
 - Exchange format:
 - XML file with extension *.mnlx (**M**onaco **L**ayout **C**onfiguration **E**xchange file)
 - *.mnlx file contains configuration data from:
 - Layouts of a workspace which should be exchanged.
 - HMI Controls which are part of those Layouts
 - the data server which is used by HMI Controls Graphical Instruments which are part of those Layouts

Import/Export of Layouts

Implementation (II):

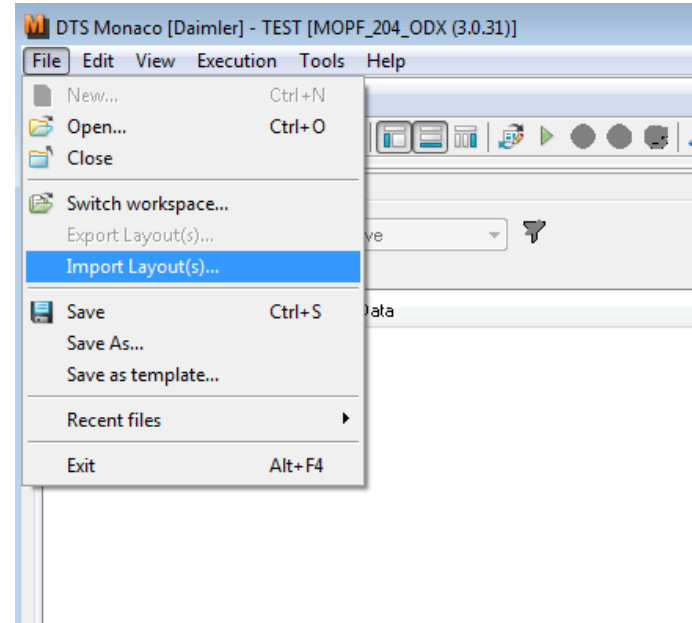
■ General aspects

- *.mnlx file **does not** contain the following information:
 - OTX scripts which are used by HMI Controls
 - Sequence files of HMI Controls (e.g. Tool and SWT Quick Test)
 - CAN filter files
 - XSL files (e.g. for the report in the HMI Control ECU Identification)
 - CFG files for HexServices and ProtocolParameterSets
 - Pictures, RTF files etc. which are referenced by HMI Control Annotation
- I.e. this information is not part of an Export and therefore this information is not available during an Import of Layout(s) anymore. The user have to make sure that those information is already available in the target workspace/DTS-Project.

Import/Export of Layouts

Implementation (III):

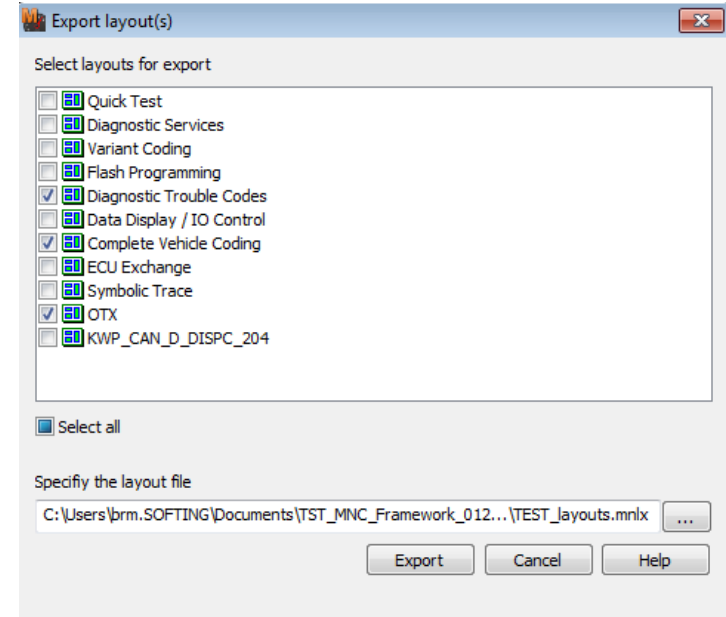
- Import Export of Layouts functionality accessible by
 - Main menu “File” > “Export Layout(s) ...” and “File” > “Import Layout(s) ...”
- Export possible in Configuration Mode
- Import possible in Configuration Mode and passive Execution Mode (Execution is stopped)



Import/Export of Layouts

Implementation (VI):

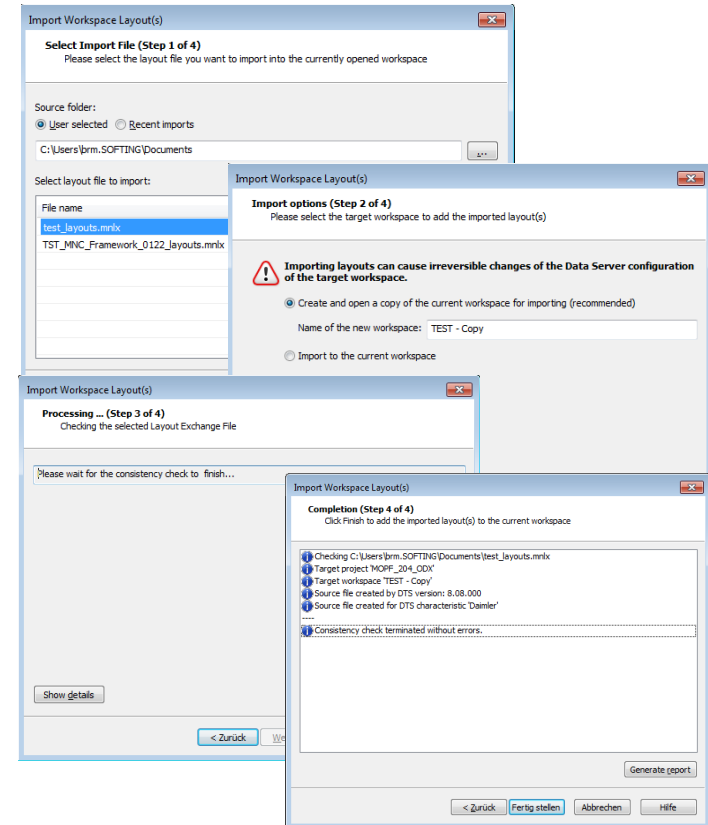
- Dialog “Export layout(s)” provides
 - a list all Layouts contained in a workspace
 - Defaults values for *.mnlx file
 - File name: <name of workspace>_Layouts
 - File path: Windows „Documents“
 - Export functionality by button [Export]



Import/Export of Layouts

Implementation (VII):

- Import of Layouts:
 - 4 step Wizard “Import Workspace Layout(s)” provides
 - in step 1 a possibility to select a *.mnlx file
 - in step 2 a possibility to import the Layout(s)
 - the currently used workspace
 - in a copy of the currently used workspace
 - in step 3 a check of the content of the Layouts which shall be imported against the content of the target workspace by a consistency check
 - in step 4
 - a presentation of the results of the consistency check
 - triggering the import functionality including a backup mechanism of the *.mnlx file to a global path
 - Generate a simple report



Import/Export of Layouts

Implementation (VIII):

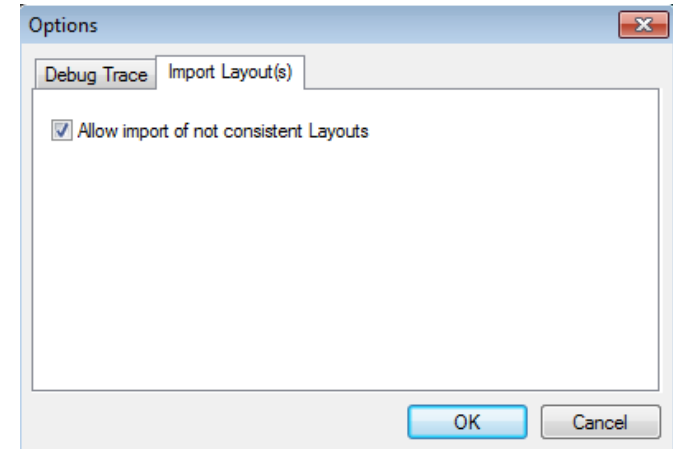
■ Consistency Check

- Checks if the content of the Layout(s) which should be imported are fit to the target workspace/DTS-Project
 - Checks if Services and their Parameters are available
 - Checks if OTX scripts are available
 - Checks if referenced files from the HMI Control Annotation are available

Import/Export of Layouts

Implementation (VIII):

- Handling of not consistent Layouts
 - Allow the import of a not consistent Layouts by a configuration option
 - Main menu “Tools” > “Options”
 - Not consistent important Layouts
 - Indicated by an “!” as a postfix of the Layout name
 - Make a not consistent Layout consistent
 - Save the workspace after the Layout was customized



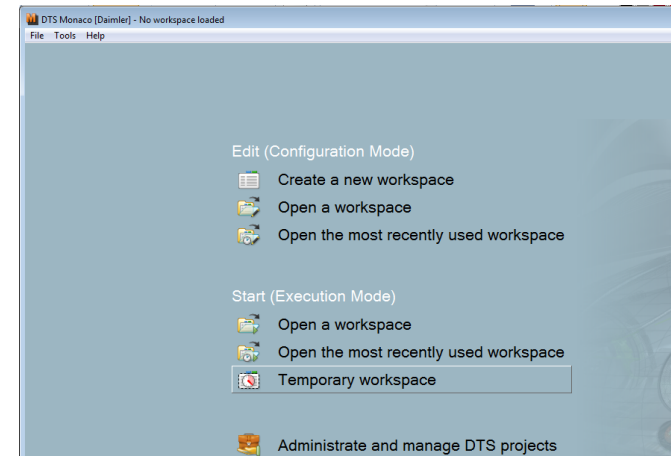
Temporary DTS-Project

Use Case:

- Use SMR-D and CBF files directly with DTS 8 Monaco without the need for creating a DTS-Project (MON-2206)

Implementation (I):

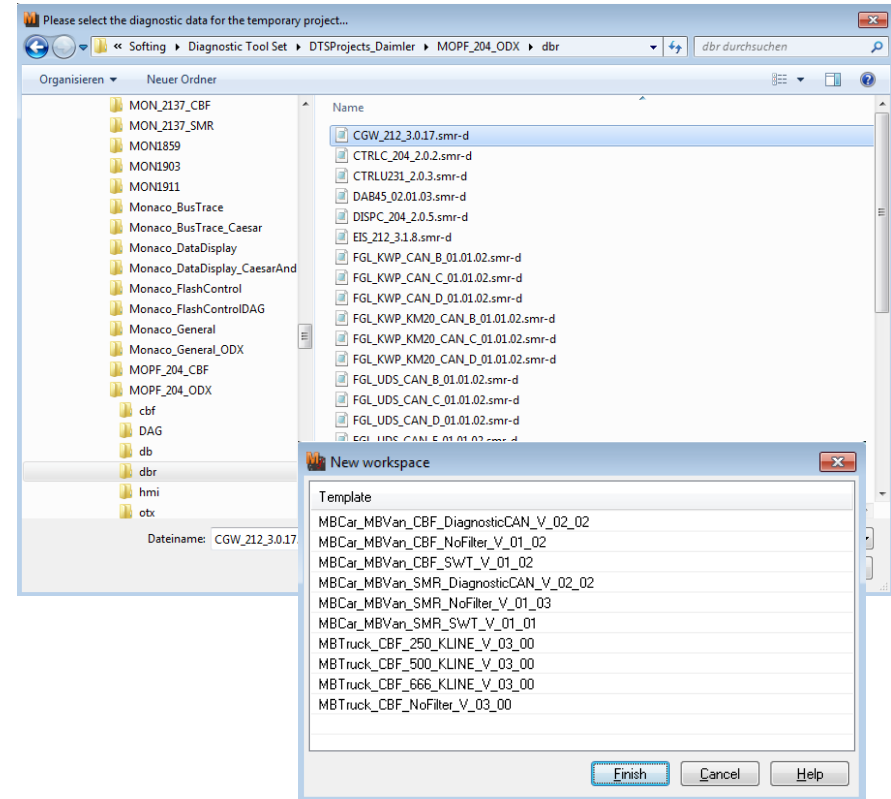
- SMR-D and CBF files can be used with DTS 8 Monaco directly by creating an temporary DTS-Project
- Access to the functionality
 - Start page entry “Temporary workspace”
 - Main menu “File” > “Temporary (start) ...”



Temporary DTS-Project

Implementation (II):

- Selection SMR-D or CBF files
 - Windows standard dialog
 - Selection of a Monaco Template
- **Important:** DTS-Project is temporary.
That means saving the workspace is not possible.



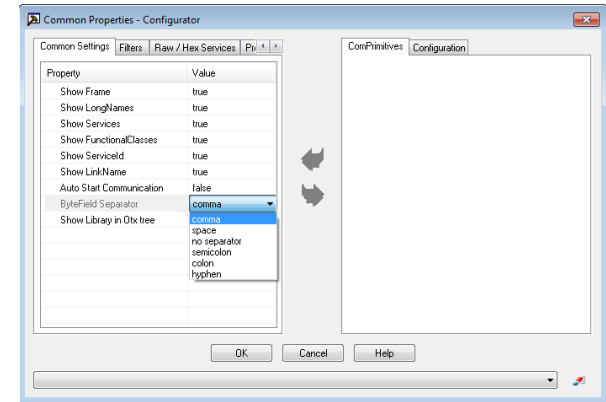
Separator for hex values

Use Case:

- Configuration of a separator for hex values shall be possible. In this way the user can copy hex values out of DTS 8 Monaco and paste them into other tools which accept only a certain separator for hex values. (MON-2210)

Implementation (I):

- A common property “Bytefield Separator” is introduced
 - Values
 - comma, space, no separator, semi colon, colon, hyphen
 - Default: comma
 - Access to the functionality
 - Configuration Mode: Main menu “Configuration” → “Configure common properties...”
 - Effect
 - In DTS 8 Monaco hex values are displayed with the separator selected by the configuration option
 - Exception: HMI Control Bus Trace, here a space is used as a separator



Keyboard control

Use Case:

- Use DTS 8 Monaco during a testing run with a vehicle because in this case using a mouse or mouse pad is not very comfortable (MON-2214, MON-2044, MON-417, MON-1551)

Implementation (I):

- Important short cuts:
 - Switching between areas
 - Constant Layout: <STRG> + <1>
 - Status message area: <STRG> + <2>
 - Layout area: <STRG> + <3>
 -

Keyboard control

Implementation (II)

- Switching between Layouts: arrow keys
- Switching between HMI Controls: <STRG> + <TAB>
- Within an HMI Control:
 - Switching between control elements: <TAB>
 - Icon buttons: Short cut is displayed in a tooltip
 - Execution of buttons and check boxes:
 - Short cut (e.g. button [Configure...]: <ALT>+<C>)
 - <space> (if the element has already the focus)
 - Context menu: <Shift>+<F10>

Keyboard control

Implementation (III)

- Restrictions:
 - Keyboard control is not available in Configuration Mode and in the Configurators of the HMI Controls.

Stop Communication by double click

Use Case:

- In the HMI Controls Logical Link List, Diagnostic Services and Flash Programming it is currently possible to start the communication to an ECU with a double click. Due to be consistent with this behavior it shall also be possible to stop the communication in the same way in those HMI Controls. (MON-2215)

Implementation:



- Double Click closes the communication to an ECU in the case the ECU is in state “communication”.
- The functionality is provided for the HMI Controls Logical Link List, Diagnostic Services and Flash Programming.

ISO DoIP - ISO 13400-2

Use Case:

- New ECUs support only ISO DoIP (ISO 13400-2). Therefore DTS 8 Monaco shall support it too to ensure that it is possible to communicate with those ECUs.

Implementation:


- Integration of Softing D-PDU-API which supports ISO-DoIP in DTS 8 Monaco
- Provide two buttons for broadcast and ethernet activation
 -  Execution of Ethernet activation and broadcast
 -  Execution of broadcast

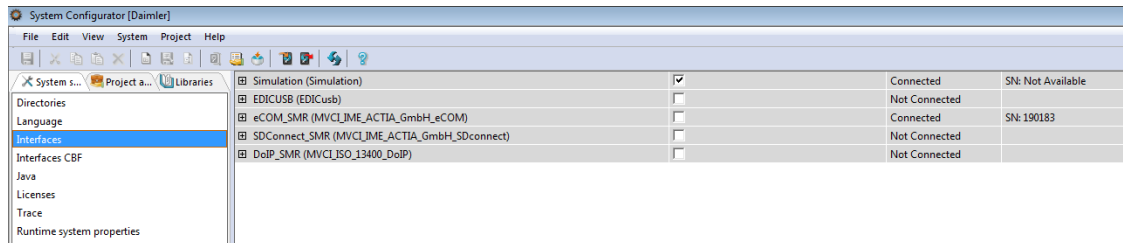
Detect all configured and available VCIs

Use Case:

- A possibility shall be provided to check all configured VCIs towards their availability. This functionality is needed for the case it is not clear which VCI is part of a test equipment. (MON-2213)

Implementation:

- System Configurator provides the desired functionality
- Access to the functionality
 -  Tool bar icon button “Update interface status”



Handling of trace path

Use Case:

- The user shall be able to write a trace into the DTS Projects directory in an easy way. In this way traces are part of the backup/restore mechanism of a DTS Project.

Implementation:

- System Configurator provides the desired functionality
- Paths for trace in HMI Controls Bus Trace and Symbolic Trace are affected
 - Default: [DTSPProjectpath]/traces
 - System trace path can overwrite the DTSPProject path by the check box „Use system trace path to overwrite project trace paths“ in System Configurator → tab „System Settings → section „Trace“

New communication components

Use Case:

- Profit from the improvements of the new version of the communication components in Monaco

Implementation:

- Integration of Actia I+ME GmbH communication components
 - CAESAR: v3.2.16.0
 - D-PDU-API: v1.12.4.0
 - SDConnect Toolkit: v2.1.10.1
 - DoIP PDU-API: v05.01.00

New functionalities

- General issues
- HMI Controls
- SWT
- OTX-Studio/OTX-Runtime

Bugs

- Fixed bugs
- Known bugs

HMI Control Diagnostic Services

Use Case:

- Display only certain information to reduce the content of the process value trace to the important parts (MON-2280)

Implementation(I):

- Introduction of a new configuration option
 - Name: "Show non only constant parameters"
 - Values: true/false (default: false)
- Only non constant parameters, the service name and error message are shown.

Current selection

Transmit

☐ Cyclic
 [ms] 2000

Read ID

Read DTC

Clear DTC

Process Values

ShortName	Value	Unit	RangeInfo
manufacturing_data_container_Read	eALL_FAILED		
The D-PDU API call has failed	0xD090		
ActiveDiagnosticInformation_Read	eALL_POSITIVE		
Version	81		Value OK
Variante	1		Value OK
Aktive SG Software	Anwendung aktiv		Value OK
Gateway	wahr		Value OK
reserviert	0		Value OK
Identifikation	0x020151		Value OK
Session Type	Extended		Value OK
reserviert	0		Value OK
HardwareVersion_Read	eALL_POSITIVE		
Patch Level	1		Value OK
Kalenderwoche	43		Value OK
Jahr	8		Value OK

HMI Control DTC

Use Case:

- Persistency of execution settings (MON-2217)

Implementation:

- Execution settings are saved in the workspace now

DTC

ECU: V CGW_212 (UDS_CAN_D).CGW_212_172_E01A

Execution Settings

☒ Environment Data ☐ Test not completed

☐ Supported DTCs ☐ Mirror Memory

Display Filter

☐ MIL On ☐ Stored

☐ Active ☐ DTC Filter

Read ☐ Cyclic (ms) 1000 Clear all DTCs Clear selected DTCs

DTC count: 2 of 2

DTC	DTC (hex)	Fault Symptom	Stat
<input checked="" type="checkbox"/> P061000	0x061000	Die Steuergerätecodierung ist fal...	0x21

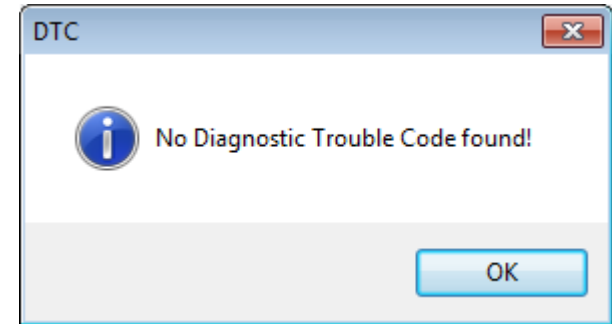
HMI Control DTC

Use Case:

- Indication for the user that no DTC could be read (MON -2347)

Implementation:

- If no DTCs could be read from an ECU a info dialog pops up with an appropriate information for the user



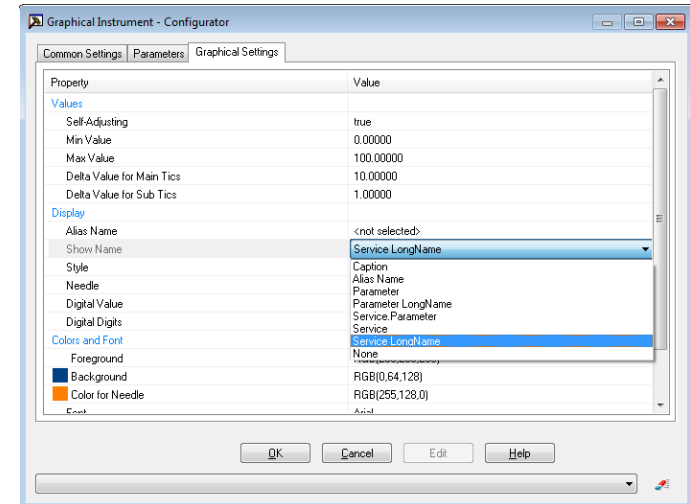
HMI Control Graphical Instruments

Use Case:

- For some reasons the name of a service is often more significant than the one of a parameter in the diagnostic data of Daimler. Therefore the possible values for the label of a Graphical Instruments shall be enhanced by the name of the service. (MON-2212)

Implementation:

- Enhancement of the possible values for the configuration option “Show name” as follows:
 - “Service” (i.e. ShortName)
 - “Service.LongName”



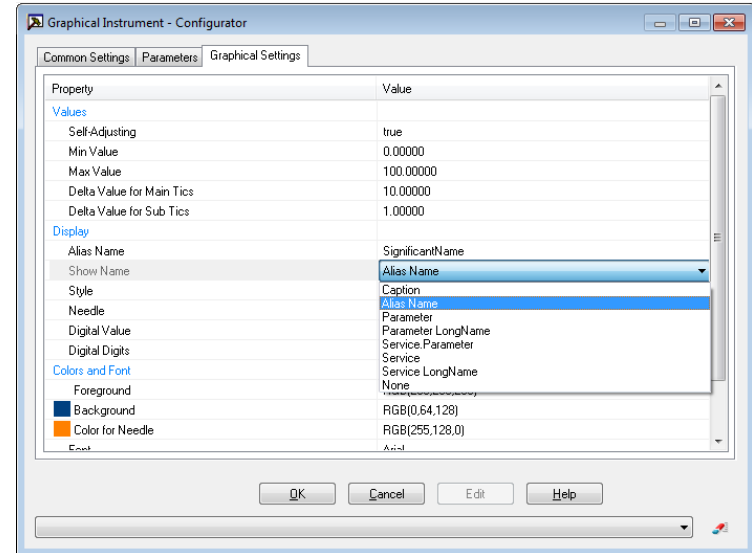
HMI Control Graphical Instruments

Use Case:

- For some reasons the name of a service or parameter is often not significant in the diagnostic data of Daimler. Therefore the possibility to enter an alias name shall be introduced. (MON-2218)

Implementation:

- Enhancement of the possible values for the configuration option “Show name” as follows:
 - “Alias Name” (if this value is selected the alias name is shown as a label)
- Introduction of a new configuration option „Alias Name“:
 - Enter the alias name



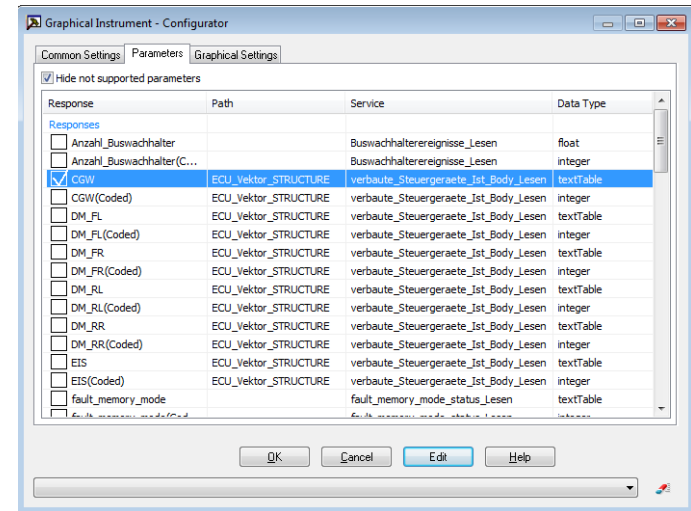
HMI Control Graphical Instruments

Use Case:

- Currently all parameters which belongs to the services selected in the data server are shown in the tab “Parameters” of the Configurator of a Graphical Instrument. The parameter list can be very long. There for a mechanism shall be introduced to reduce the list of parameter to the supported ones. (MON-2216)

Implementation:

- A checkbox “Hide not supported parameters” is introduced:
 - Default: checked
 - Status of the checkbox is saved in the workspace



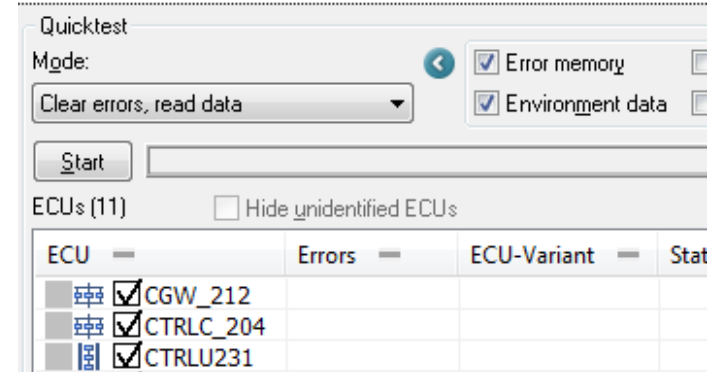
HMI Control Tool Quick Test

Use Case:

- The following workflow of the user shall be met: Read DTCs, generate a FiNAS report and clear all DTCs afterwards. This sequence shall be done automatically. (MON-2190)

Implementation:

- Introduce a combo box “Mode”
 - Replaces the checkboxes „Read data“ and „Clear errors”
 - Configuration of the button [Start] by the following values:
 - Read ECU Identification only
 - Read data (default)
 - Clear errors
 - Clear errors, read data
 - Read data, generate FiNAS, clear errors
 - Value of the combo box is stored in the workspace



HMI Control Tool Quick Test

Use Case:

- For an exchange of the FiNAS report with the user DTS-Project a possibility shall be provided to select a path for the report in the HMI Control Tool Quick Test. (MON-2219)

Implementation:

- Enhance the dialog “FiNAS report”:
 - Possibility to select a (local) path for the FiNAS report
 - (Local) path is saved in the workspace
 - Use the FiNAS (global) path which was selected in the System Configurator by the checkbox “Use default path”. Default is checked.
 - Status of checkbox “Use default path” is saved in the workspace

The screenshot shows a dialog box titled "FiNAS report". It contains the following fields and controls:

- Model range: 204 (dropdown)
- FiNAS Category: - (dropdown)
- FiNAS Number: 12345 (text input)
- Report Type: in (dropdown)
- FiNAS Path: C:\ProgramD...\Finas (text input with browse button)
- ☒ Use default path (checkbox)
- File Format: VRX (dropdown)
- FiNAS File Name: 204-12345_in (text input)
- ☐ Generate automatically (checkbox)
- OK button
- Cancel button

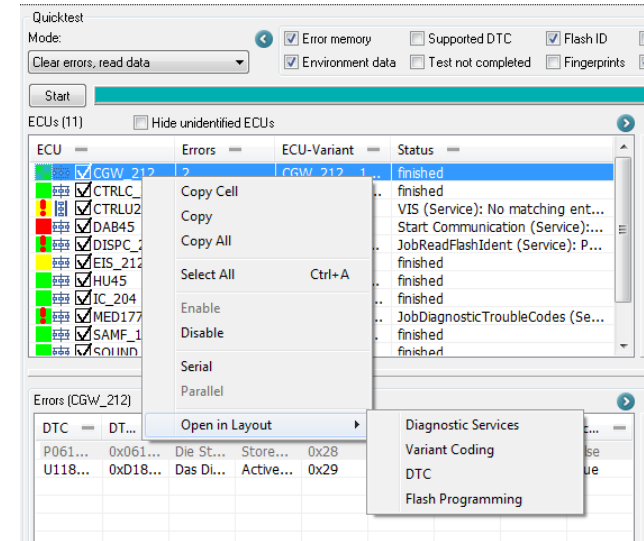
HMI Control Tool Quick Test

Use Case:

- The following workflow of the user shall be met: Execute a Quick Test and open a certain ECU in a DTC or Flash Programming HMI Controls afterwards. (MON-2211)

Implementation:

- Enhance the context menu entry “Open Layout in” of an ECU with the following values:
 - DTC
 - Opens the selected ECU in an temporary Layout which contains a DTC HMI Control. Communication is started for this ECU in this HMI Control.
 - Flash Programming
 - Opens the selected ECU in an temporary Layout which contains a Flash Programming HMI Control. Communication is started for this ECU in this HMI Control.



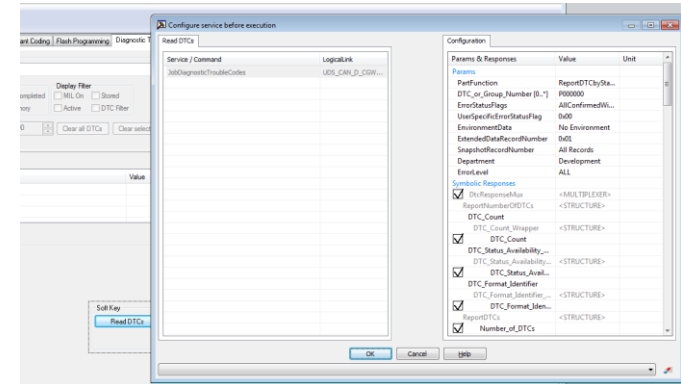
HMI Control Soft Key

Use Case:

- Some users have the use case that values of request parameters of a service shall be changed every time before the service is executed. Therefore a comfortable way shall be provided to enable the configuration of the request parameters. (MON-2207)

Implementation:

- Introduce new configuration option
“Configure service before execution”:
 - Possible values: true, false (default)
 - If this option is set to „true“ the Configurator is opened when the Soft Key is pressed. By the configurator the user can set the values of the request parameters. After closing the Configurator the sequence/service(s) is executed with the changed values.



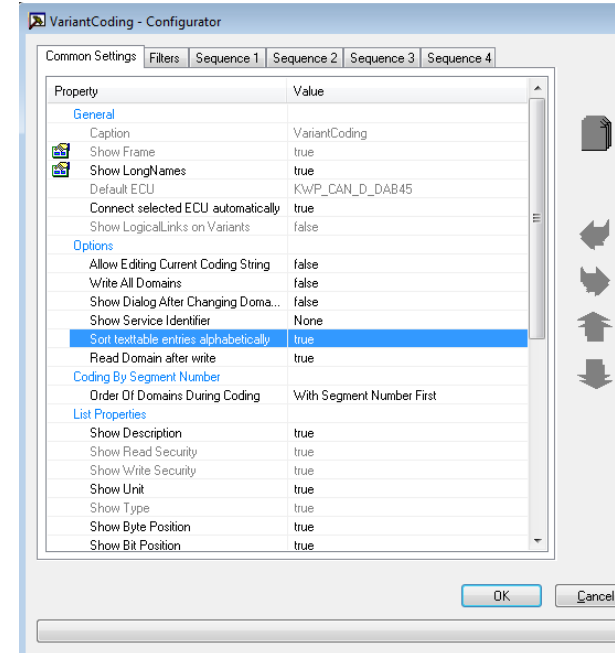
HMI Control Variant Coding

Use Case:

- Make it easier to find a Text table entry for a Meaning by an alphabetical sorting (MON-2221)

Implementation:

- Introduce new configuration option
“Sort texttable entries alphabetically”:
 - Possible values: true, false (default)
 - If this option is set to „true“ possible values for a meaning of type texttable are shown in alphabetical order in the appropriate combo box.



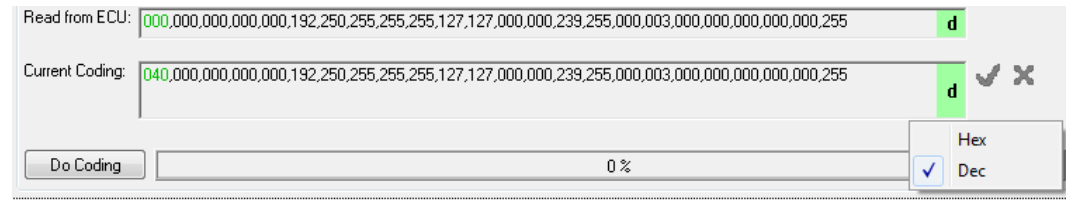
HMI Control Variant Coding

Use Case:

- In the tool chain of Daimler there are tools which accept coding strings only in dec presentation. Therefore a possibility shall be provided for the user to determine if the original and the current coding string shall be displayed in a hexadecimal or decimal format. In this way the user is able to copy the coding string from the original or current coding string edit control in the HMI Control Variant Coding and paste it in an appropriate place of another tool which requires a decimal format for the coding string. (MON-2209)

Implementation:

- Switching between dec and hex is possible a icon button integrated in the edit controls of the Original and Current Coding String by a context menu
- Representation is saved in the workspace



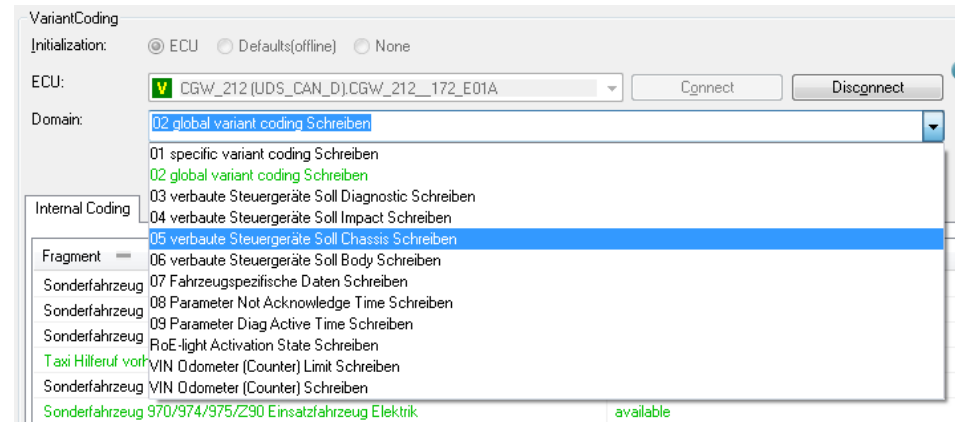
HMI Control Variant Coding

Use Case:

- User needs an indication which Domains are changed. In this way the user can recognize which Domains may shall be written to the ECU. (MON-2112)

Implementation:

- Domains for which the Original and Current Coding String are different are shown written in green color in the combo box “Domains”.



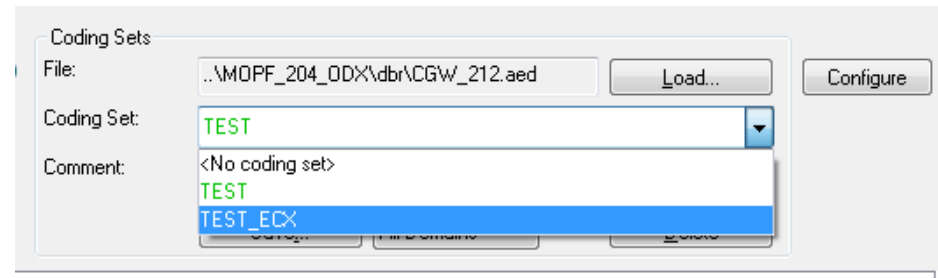
HMI Control Variant Coding

Use Case:

- User needs an indication if a Coding Set is changed. In this way the user can save the changed Coding Set in an AED or ECX file to reuse those changes later. (MON-2113)

Implementation:

- Coding Set for which the Current Coding String is different to the coding string provided by the Coding Set are displayed in green color in the combo box “Coding Sets”.



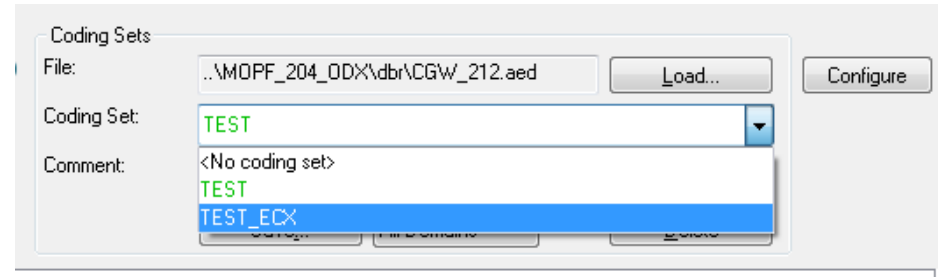
HMI Control Variant Coding

Use Case:

- A user changes a Coding Set (i.e. one of the Current Coding String is different from the appropriate one stored in the Coding Set). If the user selects another Coding Set the changes made for the previous Coding Set are lost. Therefore a mechanism shall be implemented by which the user is informed that done changes are lost. In this way the user can decide how to go on. (MON-2214)

Implementation:

- A dialog is introduced which ask the user if
 - Changes shall be discarded. I.e. the selected Coding Set is loaded and the done changes are lost.
 - Action is cancelled. I.e. the selected Coding Set is not loaded. Changes are not lost.



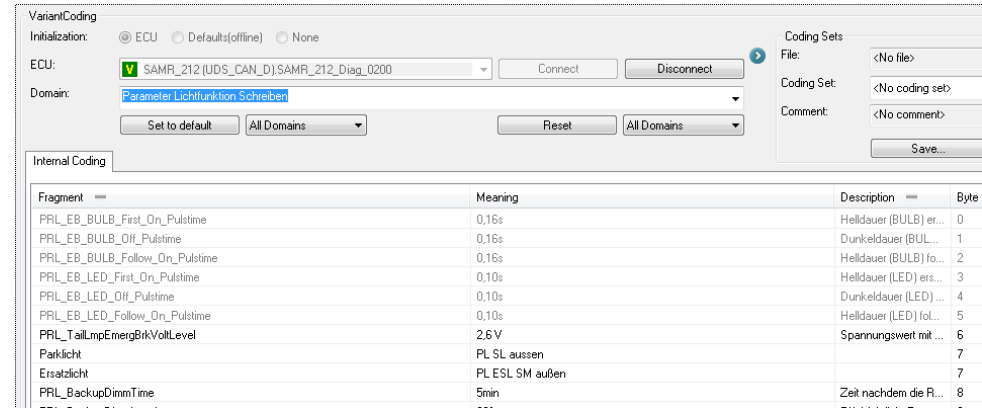
HMI Control Variant Coding

Use Case:

- Make it for an user obvious that a Fragment is constant or reserved and therefore cannot be changed (MON-2429).

Implementation:

- Fragments of type and constant or reserved are displayed disabled (with grey color).



HMI Control Flash Programming

Use Case:

- Make assignment between ECU and Flash information clear (MON-2443).

Implementation:

- Name of the selected ECU is part of the label of the Session Control window

[illegible]

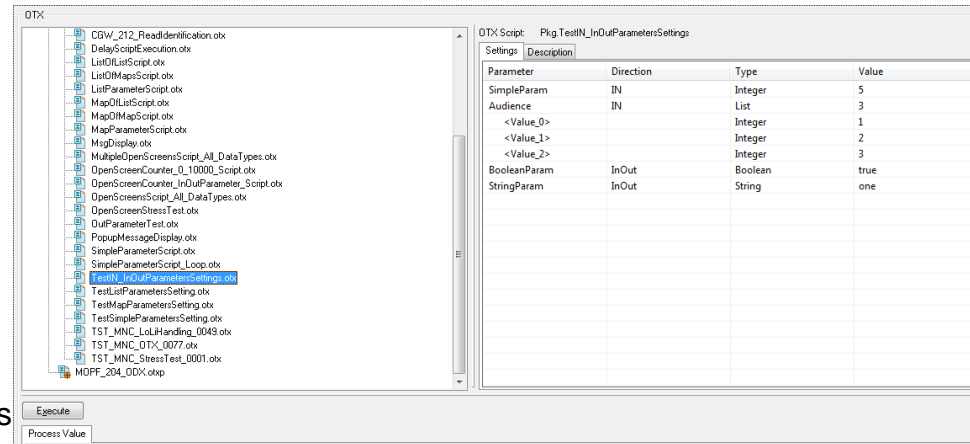
HMI Control OTX

Use Case:

- Users needs a possibility to set input parameter of an OTX script to be able to test OTX scripts fully in the first place.

Implementation:

- Tab “Settings” is added
 - Provide the name, direction, type and value of an OTX parameter
 - Possibility to set the value for a OTX Parameter
 - Supported types of OTX parameters:
 - Integer, Boolean, Float, String, Bytefield, Lists and Maps



HMI Control OTX

Use Case:

- The users want to reduce the presentation of OTX Scripts in the OTX Script Selector to the amount of practically relevant scripts. Therefore OTX Scripts which are not executable shall not be shown any more in the HMI Control OTX.

Implementation:

- OTX Scripts which are not executable (i.e. have no main procedure) are not displayed in the OTX HMI Control any more.

New functionalities

- General issues
- HMI Controls
- SWT
- OTX-Studio/OTX-Runtime

Bugs

- Fixed bugs
- Known bugs

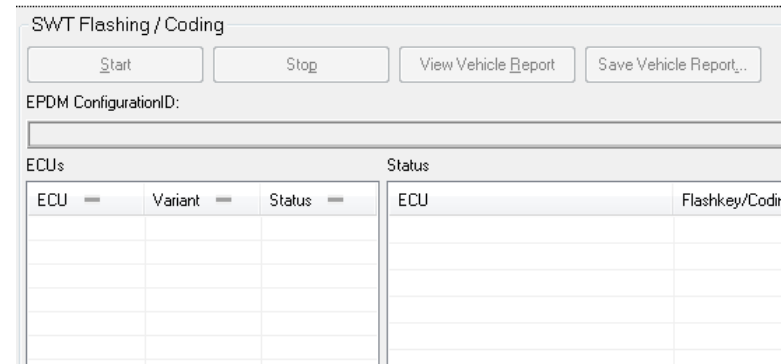
SWT: FiNAS number and target release

Use Case:

- The property EPDM-Configuration represents an unique identifier for an vehicle update package. Since the release 9 this information is displayed in the report of the HMI Control Vehicle Update. For release 10 the EPDM-Configuration shall also be displayed in the GUI of the HMI Control. In this way the user can find this information more easily. (MON-2256)

Implementation:

- The property EPDM-Configuration is shown in the GUI of the HMI Control between the button [Start] and the progress bar.



SWT Flashing / Coding

Start Stop View Vehicle Report Save Vehicle Report...

EPDM ConfigurationID:

ECUs			Status	
ECU	Variant	Status	ECU	Flashkey/Codin

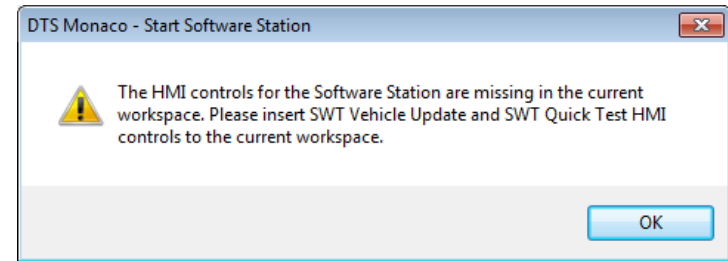
SWT: FiNAS number and target release

Use Case:

- SWT application shall not be started any more by DTS 8 Monaco in the case the used workspace does not contain all SWT HMI Controls (SWT Quick Test and SWT Vehicle Update). In this way the SWT application can instantly trigger Monaco without the need to switch to Monaco to load another Monaco workspace. (MON-2257)

Implementation:

- In the case the SWT application is started by Monaco and the workspace does not contain all SWT HMI Controls an appropriate error dialog occurs.



New functionalities

- General issues
- HMI Controls
- SWT
- OTX-Studio/OTX-Runtime

Bugs

- Fixed bugs
- Known bugs

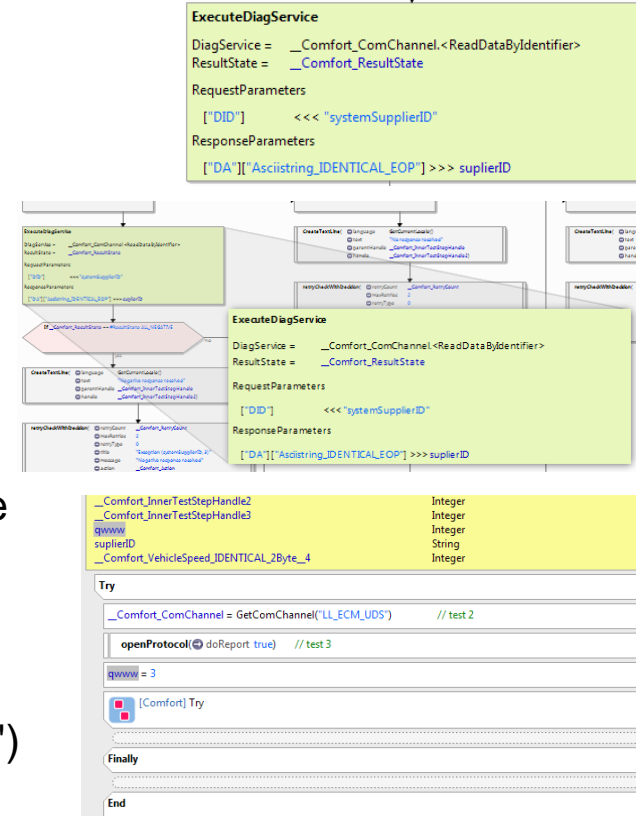
OTX: Improved layout of sequences

Use Case:

- Editing or viewing OTX sequences in OTX editor window

Implementation:

- Bold layout of action names
- More compact vertical layout for parameter lines
- Free mode: Colored blocks for different action types, magnifier for lower zoom stages ($\leq 72\%$) by pressing the CTRL key
- ExecuteDiagService hides irritating information
- Clicking on a variable highlights all other occurrences
- Collapsed blocks indicated by one element ("ghost block")



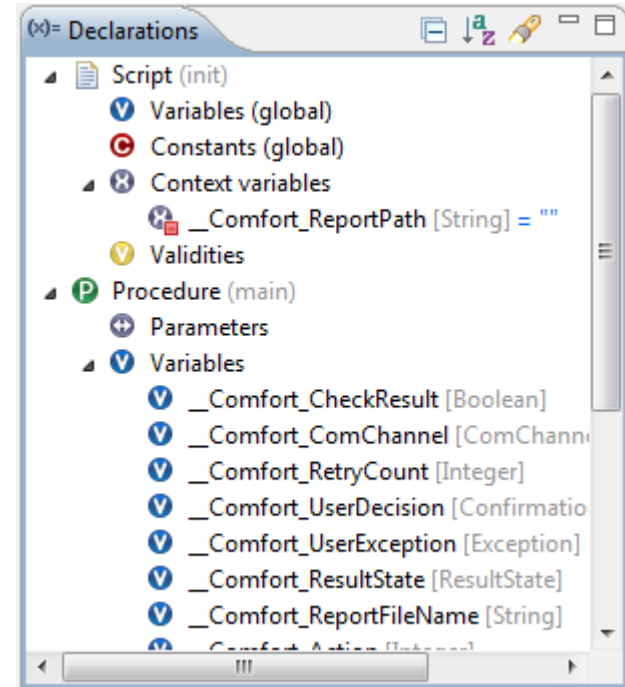
OTX: Declaration view

Use Case:

- Viewing and creating variables or parameters of OTX sequences

Implementation:

- Local or global variables, parameters etc. of a script or procedure are handled by an individual view ("Declarations")
- A context menu is provide to the user for creating, deleting, changing entries etc.



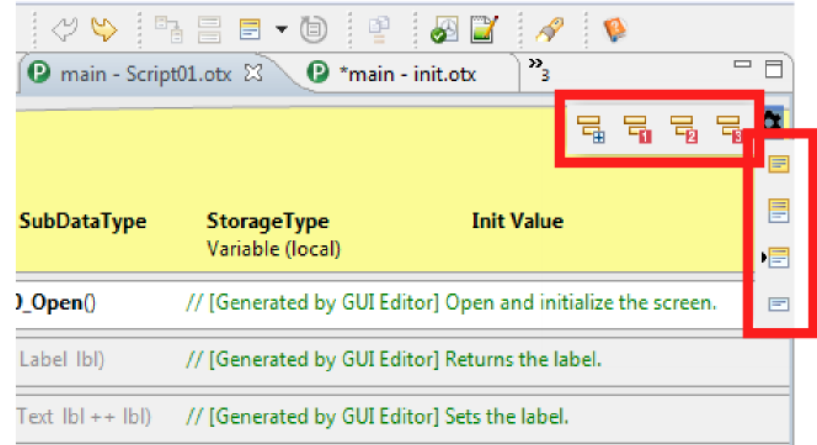
OTX: Floating toolbar

Use Case:

- Fast access to display options of the sequence editor

Implementation:

- The amount of displayed details can be selected with the buttons in the vertical bar. Further the specification mode is accessible.
- The number of displayed hierarchy levels of a sequence can be selected with the button in the horizontal bar.



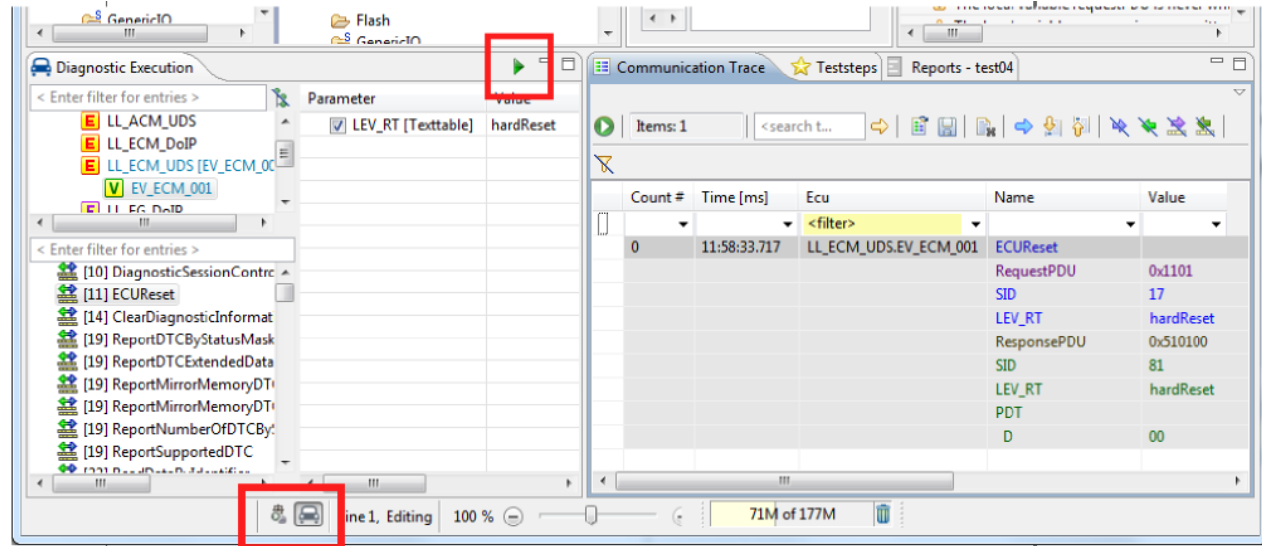
OTX: Diagnostic service execution

Use Case:

- Execute a diagnostic service within OTX Studio (no sequence necessary)

Implementation:

- New view 'Diagnostic Execution' in combination with 'Communication Trace'
- Activation of both views is done via a button in the status bar
- Execution of service via a dedicated button



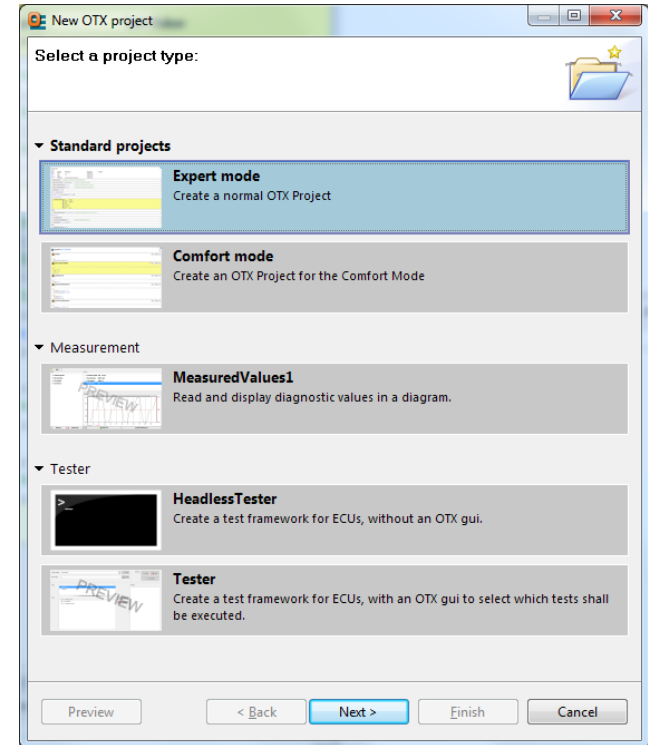
OTX: Redesigned 'Project New' wizard

Use Case:

- Create a new OTX project

Implementation:

- First page of wizard provides a selection of the project type
 - Expert mode ("standard mode")
 - Comfort mode ("newly available for DAG ODX data")
 - Measurement (applicable for SOFTING TDX)
 - Tester (licensed addon). Editor for developing diagnostic ECU tests.



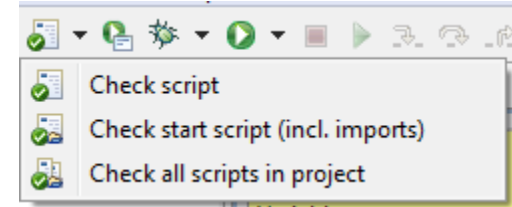
OTX: Extended "Check script"

Use Case:

- Check different parts of an OTX project

Implementation:

- Three different options available
 - Check all script of project
 - Check the open script (including imported scripts)
 - Check the start script (including imported scripts)



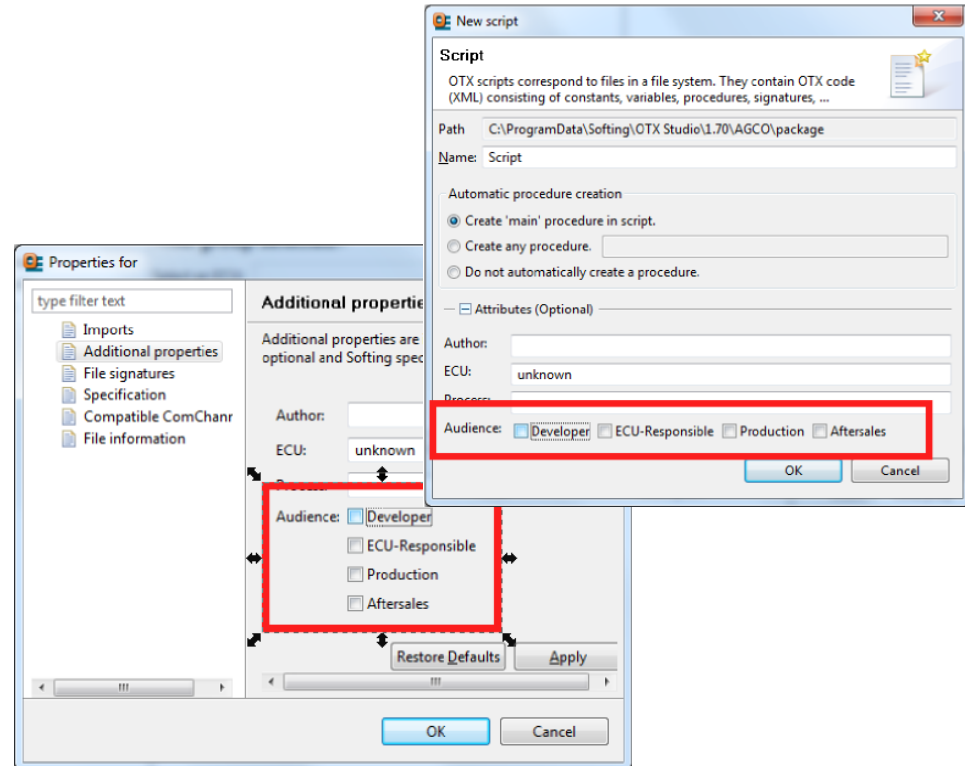
OTX: Audience property of OTX scripts

Use Case:

- Define an audience for an OTX sequence
- In future it can be used as an display filter option in MONACO

Implementation:

- The 'create OTX script' and the 'Property' dialog of a script provide defining the audience of a script



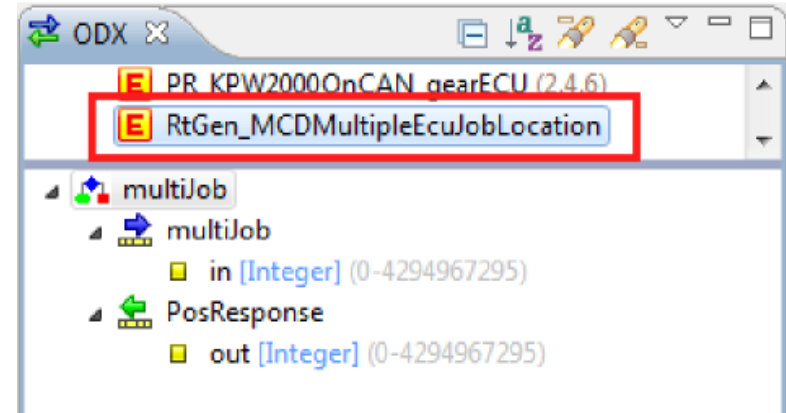
OTX: Using multi ECU jobs

Use Case:

- Execute a multi ECU job

Implementation:

- All multi ECU jobs in ODX data are assigned to a automatically generated logical link. Multi ECU jobs are handled the same way like diagnostic services or ECU related JAVA jobs.



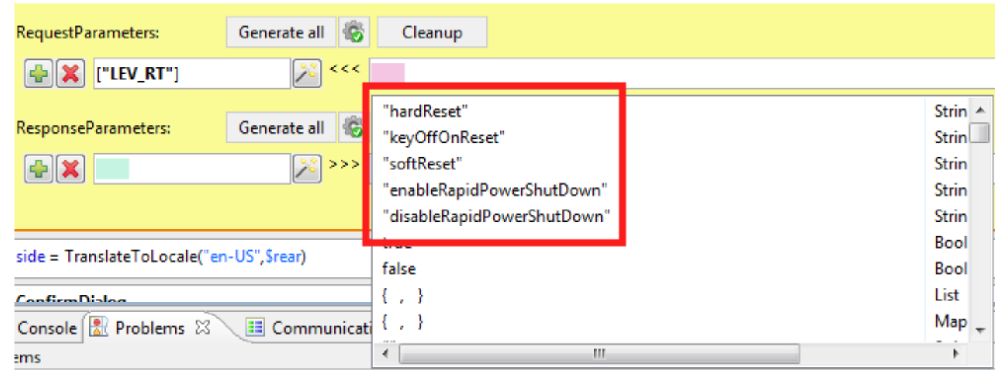
OTX: Text table entries

Use Case:

- Fast and easy parametrisation of request parameters of data type TextTable

Implementation:

- Intellisense (Auto completion list) shows all valid test table entries



OTX: Minor improvements

Use Case:

- Improvements of user experience

Implementation:

- Special character '\t' (Tabulator) can be used now in STRING terms
- Copy & Paste of packages into a different project
- Copy & Paste of entire projects
- PTX export wizard can be called via context menu in 'Script explorer'

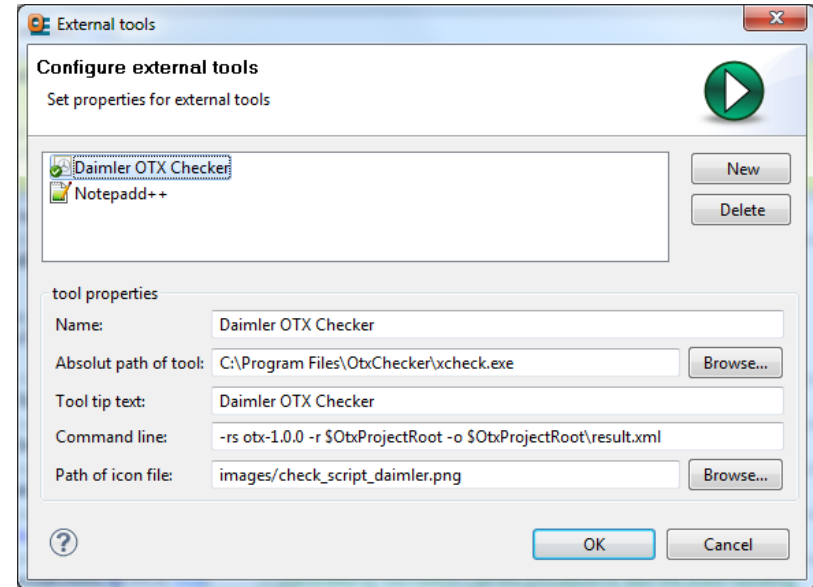
OTX: Daimler specific OTX checker

Use Case:

- External DAG specific OTX checker can be used

Implementation:

- If available on the file system the DAG specific OTX checker is automatically integrated as an external tool
- Checker can be started via toolbar button
- Error or warning messages are displayed in the 'Problems view'



OTX: OTX runtime improvements

Implementation:

- New actions and terms in extension "SoftingDiagnostics" :
 - **Actions:** StartBusMonitor, StopBusMonitor, GetBusMonitorFrames
 - **Terms:** GetPhysicalVehicleLink, GetDataBaseVersion, GetByteLength, GetBytePosition, GetIntervalOfTextTableElements
- New terms in extension "File"
 - **Terms:** GetParentDirectory, GetDTSTracePath
- New MAP types introduced:
 - **SortedMap, OrderedMap**
- Bug-Fix: UNICODE => UTF8 conversion in case of string truncation for displaying purposes now works correctly
- Bug fix: Runtime crash for MapGetKeyList() and MapGetValueList() when used in a certain condition

OTX: OTX runtime changes

Implementation:

- **IMPORTANT:** All file path ends from now on with '\'
 - Relevant for:
 - GetCurrentScriptDirectory**
 - GetDTSTracePath**
 - GetPackageRoot**
 - GetParentDirectory**
 - GetRuntimePath**
 - FileDialog**
 - FolderDialog**
- OTX scripts have to be adapted to the new behavior !

New functionalities

- General issues
- HMI Controls
- SWT
- OTX-Studio/OTX-Runtime

Bugs

- Fixed bugs
- Known bugs

Monaco: Fixed bugs (I)

General: (I)

- MON-2231: System Configurator tooltip for the "Delete" icon does not work!
- MON-2262: Project does not update if changed via EDDA opened for another project
- MON- 2383: ConsistencyCheck failed for „generic“ JavaJobs
- MON-2393: Layout Export - File name of Layout Exchange File in the Edit and Browse field is not initially set as expected
- MON-2404: ODX-V is not completely supported
- MON-2414: Restore function does not work correctly
- MON-2471: Keyboard operation: Switch between layouts are interrupted
- MON-2472: Keyboard operation: <STR> +2 do not display a status View
- MON-2469: Project Management: Monaco crash in Project Update attempt at a local project

Monaco: Fixed bugs (II)

General: (II)

- MON-2470: Project Management: Spelling Error on Project Update dialog
- MON-2497: Crash on project management before displaying the result dialogue

HMI Control VRX-Differ:

- MON-2500: VRX Differ - Difference between Variant Coding and VRX Differ

Monaco: Fixed bugs (III)

HMI Control Tool Quick Test:

- MON-2124: Monaco crashes while running a Quicktest
- MON-2269: ECU's in quicktest not listed in alphabetical order in the report files
- MON-2290: 2 Ecu's of an alternative sequence group were displayed in FINAS report

HMI Control Toggle Button:

- MON-2400: Monaco crashes

HMI Control Soft Key:

- MON-2057 : Process values are shown in case of Auto Start is set to "false"

Monaco: Fixed bugs (IV)

HMI Control Variant Coding (I):

- MON-2420: Meaning alternating with doubleclick does not work correctly!
- MON-2422: Monaco crashes when coding of HVAC222
- MON-2423: Green markers are not reset under Internal coding after coding!
- MON-2438: An error occurred while loading an aed file
- MON-2439: Incompatible aed file
- MON-2449: Crash with Runtime Exception while doCoding
- MON-2438: Error while loading an aed file
- MON-2255: Scrollbar in codingfield covers commas
- MON-2274: Copy all function does not work correctly
- MON-2388: Default sorting of the Fragments
- MON-2387: Set to default for current domain and changed Fragments
- MON-2385: Combo box for save coding sets is too small

Monaco: Fixed bugs (V)

HMI Control Variant Coding (II):

- MON-2392: Configurable separator for HEX format
- MON-2394: SCN Number in AED File
- MON-2395: Coding Set Handling
- MON-2386: Coding Set changes do not appear immediately in concerned domains
- MON-2406: Variant coding with two system parameters does not work!
- MON-2410: An error occurred while loading an aed file, although the length machted correctly
- MON-2225: Scrollbar in codingfield covers commas
- MON-2420: Meaning alternating with doubleclick does not work correctly!
- MON-2481: Column Handling Error in VariantCoding HMIControl
- TTP 20367: Sequence Button does not work any more after a disconnect
- TTP 20221: Monaco crashes when a different ECU is selected while coding an ECU

Monaco: Fixed bugs (VI)

HMI Control Variant Coding (III):

- MON-2513: Error of wrong length
- MON-2509: Coding data disappear
- MON-2506: Quicktest - Check "Hide unidentified ECU's" leads to an error
- MON-2507: A yellow frame appaer
- MON-2510: Monaco crashes when saving ECX file
- MON-2512: Wrong default file name
- TTP 20489: The column "Partnumber" is not sortable
- TTP 20461: Ethernet logical link - icon for online mode after an unsuccessful "Execute broadcast"
- TTP 20494: Crash when a Meaning for an external coding is set
- TTP 20501: Default Coding Strings are not set
- TTP 20502: Delete Button is not disabled after a Reset for all Domains

Monaco: Fixed bugs (VII)

HMI Control Variant Coding (VI):

- TTP 20490: The column "[Partnumber]Meaning" from External Coding tab is not sortable
- TTP 20500: Domain "Öldaten variable Werte ASSYST Service" is displayed randomly with black and green color in the "Domain:" combobox
- TTP 20497: Sorting of BytePos does not work in External Coding correctly
- TTP 20516: Fragment is not displayed with green color
- TTP 20496: Column widths are minimized
- TTP 20448: Usage of Ethernet links

Monaco: Fixed bugs (VIII)

OTX Studio / OTX Runtime:

- OTX-362: Comfort Mode: There was a problem if long names contained other characters than _a-zA-Z0-9 and Variables were generated from them
- OTX-360: A long file path is reduced (with ellipsis '...') in an error message to fit in the problems view

HMI Control OTX:

- MON-2382: If OTX libraries are used OpenScreen variables are not identified in OTX scripts

HMI Control Complete Vehicle Coding:

- MON-2413: Wrong XML tag was saved in vcx file

Monaco: Fixed bugs (IV)

HMI Control Flash Programming:

- MON-2259: Wrong handling of deactivating "Run" button
- MON-2399: Handling of flash data do not work correctly

HMI Control Symbolic Trace:

- MON-1064: Symbolic Trace: Interpretation of negative response code missing in symbolic trace

HMI Control ECU Exchange:

- MON-2482: The setting of "Selected response parameters" for the respective write service are not saved
- MON-2483: After executing red-write service in ECU Exchange, the entries appear in the "Data Item View" twice

Monaco: Fixed bugs (X)

HMI Control Diagnostic Services:

- MON-2440: Update description does not work correctly
- MON-2232: When you run the service "Read ECU identification" the wrong CAN ID were transmitted when ECU is INS2010
- MON-2389: Read error memory - no display of Extended dataRecords!
- MON-2396: With Read ECU Identification no output of values
- MON-2397: No output executing Read fingerprint at UDS ECU's
- MON-2398: On the issue of job Diagnostic Trouble Codes, JobEculIdentification and JabReadFlashID Range Info Error div. Values.
- MON-2401: Global search does not work properly when an ECU is set to diagnosis!

Monaco: Fixed bugs (XI)

HMI Control Graphical Instruments:

- MON-2426: New values for configuration option "Show name" are not compatible with old workspaces
- MON-2283: Inputfield overwrites title
- MON-2391: Name under Graphical Settings must always be displayed

HMI Control DTC:

- MON-2454: Execution Settings change at runtime

HMI Control ECU Exchange:

- MON-2053: Service contains incorrect datatypes
- MON-2295: Monaco crashes when open any xml file

SWT: Fixed bugs (XII)

General:

- MON-2265: Incorrect request parameter lenght
- MON-2263: Manual Entry of model range for multiple model ranges
- MON-2314: SWT Final Quicktest is partially stored in Monaco Finas folder

New functionalities

- General issues
- HMI Controls
- SWT
- OTX-Studio/OTX-Runtime

Bugs

- Fixed bugs
- Known bugs

Monaco: Know bugs (I)

General

- Trafo: If the Trafo is installed standalone TrafoCoApp will not work because the expat.dll can not be found.
- HMI Control Variant Coding and DTC: Combo box “Logical Link” and the search mechanism. If the combo box “ECU” is used for searching an ECU by entering a letter the combo box is opened showing the possible ECUs. If you change the layout the content of the combo box is still visible.

Monaco: Know bugs (II)

HMI Control VRX-Differ:

- TTP 16729, MON-1660: Monaco crashes when interpretation of VRX is active
This only occurs in with DTS-Projects that use a CBF database. Huge DTS-Projects consume a lot of memory. Active interpretation in the VRX-Differ increases the memory usage. If the memory usage exceeds a limit (round about ~1,9 GB) Monaco can crash or behave in an unnormal way.

HMI Control OBD:

- Automatic baudrate detection is not working with interface eCOM

HMI Control Quick Test:

- A Quick Test via KLine is not supported.

Monaco: Know bugs (III)

HMI Control Diagnostic Services:

- MON-2080: Read after Write fails with TesterID in VarCoding Service
This occurs in case the Read Service contains an additional Response Parameter "TesterID".

HMI Control Data Display / IO Control:

- Due to a Caesar problem the Configurator provides no services when the communication to an ECU was started by another HMI Control in DTS8 Monaco in Execution Mode.

Monaco: Know bugs (IV)

HMI Control Variant Coding:

- TTP 20523: When a domain is selected again or a set to default is done after performing "Do Coding", the progress bar is set to 0% and colored in grey
- TTP 20524: The previously selected logical link shall remain selected when the drop down list is closed by <Esc>
- TTP 20525: For the functionality "Set to original" the physical value is used instead of the PDU of the original coding string
- TTP 20527/20515: GUI shall be corrected (alignment issues)
- TTP 20509: Functionality described by REQ_MNC_CMP_VC_078 does not work
- TTP 20503: In the external coding tab <invalid> is shown instead of <not available> for a Meaning
- TTP 20536: Reserved Fragments are not updated in the Fragment lists
- TTP 20499/20504: Fragments with a linear calculation and of type Signal are not handled correctly

MIRKO BRAUN

Thank you.