Elektrobit

EB tresos® AutoCore Generic 8 Diagnostic Stack documentation

release notes update for the Dem module

product release 8.8.7

Elektrobit Automotive GmbH
Am Wolfsmantel 46
91058 Erlangen, Germany
Phone: +49 9131 7701 0
Fax: +49 9131 7701 6333
Email: info.automotive@elektrobit.com

Technical support

https://www.elektrobit.com/support

Legal disclaimer

Confidential information.

Table of Contents

1. Overview

This document provides you with the release notes to accompany an update to the `Dem` module. Refer to the changelog Section 2.1, “Change log” for details of changes made for this update.

Release notes details

- EB tresos AutoCore release version: 8.8.7
- EB tresos Studio release version: 29.2.0
- AUTOSAR R4.0 Rev 3
- Build number: B577598

2. Dem module release notes

- AUTOSAR R4.0 Rev 3
- AUTOSAR SWS document version: 4.2.0
- Module version: 6.4.7.B577598
- Supplier: Elektrobit Automotive GmbH

2.1. Change log

This chapter lists the changes between different versions.

Module version 6.4.7

2022-11-09

- ASCDEM-5162 Fixed known issue: Possible Out-of-bound acces after recalibration of DevFF class
- ASCDEM-5163 Fixed known issue: Dem gets stuck during initialization if more than 2040 events are configured
- ASCDEM-5206 Fixed known issue: The Dem module fails to compile/link with an undeclared/unresolved Dem_EventIndicatorClassIdx error
- ASCDEM-5211 Fixed known issue: Dem with DEM_TRIGGER_ON_FDC_THRESHOLD is writing continuously in NvM when DemImmediateNvStorage is TRUE

Module version 6.4.6

2022-09-21

- ASCDEM-5015 Fixed known issue: A monitor report in the PreInit phase is not processed if DemAvailabilitySupport is enabled
- Added configuration parameter `DemObdVariant` to allow OBD variant selection *Please note, the current OBDonUDS functionality of the Dem might be subjected to changes in future releases that are not backwards compatible*
- ASCDEM-5036 Fixed known issue: The overflow indication is not set for the permanent event memory
- ASCDEM-5078 Fixed known issue: Dem always reports the readiness group "A/C system refrigerant monitoring" as "not supported" in J1939 Diagnostic Readiness 1 (DM5)

- ASCDEM-5017 Fixed known issue: Dem sets the Confirmed DTC status bits for combined events that did not allocate the event memory entry
- Improved the description and flattened the array of the OBD Readiness Group calibration `Dem_OBDReadinessGroup`
- Added backwards compatibility for older versions not using the Gasoline Particulate Filter (GPF) readiness group
- Improved documentation regarding supported OBD ECU kinds
- ASCDEM-5161 Fixed known issue: A Dem configuration with more than 255 events assigned to any MIL group can cause a compilation error

Module version 6.4.5

2022-07-04

- ASCDEM-4868 Fixed known issue: Dem_J1939DcmGetNextFreezeFrame() and Dem_J1939DcmGetNextFilteredDTC() may provide inconsistent output data
- Improved combined DTC status handling and `DemInitMonitorForEvent` callbacks triggering for combined events
- Added configuration check to ensure that at least one `DemJ1939DTCValue` exists, when `DemJ1939Support` is enabled
- Added warning message for parameter `DemExtendedDataRecordNumber` to be displayed when the configured value is within the range reserved for OBD extended data records (0x90 - 0xEF) and OBD support is disabled
- Added support in accordance with AUTOSAR R20-11 for the following RTE interfaces/operations:
  - Interface CallbackEventUdsStatusChanged with the associated operation CallbackEventUdsStatusChanged.
  - Interface GeneralDiagnosticInfo with the associated operation GetEventUdsStatus.
  - Interface GeneralCallbackEventUdsStatusChanged with the associated operation GeneralCallbackEventUdsStatusChanged.
  - Interface CallbackDTCStatusChange with the associated operation DTCStatusChanged.
- Added support for J1939 Diagnostic Readiness 1 (DM5) for non-OBD ECU's
- Implemented API `Dem_SetEventFailedWithSyncFreezeFrame()` to support the Security Event Memory (Sem)
- Implemented optional callback `DemCallbackEventSyncStorageProcessed()` for security events
- Implemented support for the Gasoline Particulate Filter (GPF) readiness group in PID$01 and PID$41 and also extended the calibration support in the array `Dem_OBDReadinessGroupMap` with `DEM_OBD_RDY_GPF` for spark ignition vehicles

Module version 6.4.4

2022-03-09

- ASCDEM-4716 Fixed known issue: Dem generates invalid internal J1939 freeze frame data structures if a DemSPNClass references a data element of type DemExternalSRDataElementClass
- ASCDEM-4828 Fixed known issue: InitMonitorForEvent with reason DEM_INIT_MONITOR_CLEAR is not called if the UDS status is already cleared (0x50)
- ASCDEM-4824 Fixed known issue: Enabling aging via calibration can lead to out-of-bounds access.
- Removed redundant code and optimized `EnCondGrpIdx` and `OpCycleIdx`
- Implemented support for grouping of association of events for OBD purpose (MIL grouping)
- ASCDEM-4847 Fixed known issue: The MIL might be activated and reported for an event with suppressed OBD DTC
- Improved run-time performance of `Dem_J1939DcmGetNumberOfFilteredDTC()` and `Dem_J1939DcmGetNextFilteredDTC()`
- ASCDEM-4867 Fixed known issue: EventDataChanged is triggered for internal data element "aging counter" even if no entry exists for an event

Module version 6.4.3

2021-12-07

- Added support for intermediate DTC status storage

Module version 6.4.2

2021-10-27

- Changed development error ID in J1939 APIs from DEM_E_PARAM_DATA to DEM_E_WRONG_CONFIGURATION, if the requested Nm node is not configured.
- Improved Filter mask attributes set by `Dem_J1939DcmSetFreezeFrameFilter()` to be reset during Dem initialization.
- Implemented stubs for the APIs `Dem_J1939DcmFirstDTCwithLampStatus()`, `Dem_J1939DcmGetNextDTCwithLampStatus()` and `Dem_J1939DcmGetNextSPNInFreezeFrame()`, according to AUTOSAR R4.2.2
- Added debug macro in internal function Dem_GetNumberJ1939DTCs() and improved documentation of API Dem_J1939DcmGetNumberOfFilteredDTC()

- Improved development error handling if external data elements cannot be read: moved error reporting from Dem internal API to `Dem_MainFunction()`, `Dem_SetEventStatus()` and `Dem_PrestoreFreeze-Frame()`
- ASCDEM-4568 Fixed known issue: `Dem_J1939DcmSetDTCFilter()` rejects certain J1939 DTC filter criteria if OBD is disabled and DET is enabled
- Changed invocation of J1939 callback `J1939Dcm_DemTriggerOnDTCStatus()` to outside of critical section
- Added configuration constraint if the total number of DIDs contained in a freeze frame record, considering both the common part as well as the event specific part, exceeds 255
- Updated Dem internal J1939 event configuration to reduce ROM consumption
- ASCDEM-4678 Fixed known issue: Memory sections conflict between declaration and definition
- Updated Dem internal J1939 freeze frame configuration to reduce ROM consumption
- ASCDEM-4697 Fixed known issue: Permanent faults are not stored in NVRAM at the end of ignition cycle but only at the subsequent DCY start under certain circumstances

Module version 6.4.1

2021-06-25

- Updated the generated configuration data structure to support larger Dem configurations, when event combination is used
- Updated internal function to improve runtime performance by checking MIL OnOff status
- ASCDEM-4374 Fixed known issue: Dem sets the UDS status bits CDTC and TFSLC for an aged event under certain circumstances
- ASCDEM-4407 Fixed known issue: The Dem module wrongly sets the TF, TFTOC, PDTC, and TFSLC bits during event memory entry storage
- ASCDEM-4415 Fixed known issue: Dem sets the CDTC bit after aging under certain circumstances
- ASCDEM-4426 Fixed known issue: Concurrent write access to a NvM block leads to a failed DTC clearing operation
- Added an optional configuration parameter DemStatusBitStorageTestFailedPerEvent to provide a more granular control of the test failed status bit storage.
- ASCDEM-4406 Fixed known issue: Dem might not compile due to the wrong memory mapping of an internal variable
- Improved event clearing algorithm to clear combined events within a minimal number of Dem_MainFunction() cycles
  - The number of required cycles to clear combined events is depending on the value of configuration parameter `DemMaxNumberClearEventsPerCycle`.

- ASCDEM-4502 Fixed known issue: EventDataChanged callback not called when only the J1939 freeze frame is stored/updated
- ASCDEM-4492 Fixed known issue: The API Dem_SetWIRStatus() does not accept the WIR status value FALSE consecutive to an ECU startup

Module version 6.4.0

2021-03-05

- Improved `Dem_J1939DcmClearDTC()` API
- ASCDEM-4315 Fixed known issue: Dem fills unavailable J1939 freeze frame data with padding value 0x00 instead of 0xFF
- ASCDEM-4322 Fixed known issue: PID$21 is reset to zero even when the MIL status does not change from deactivated to activated
- Updated the internal functions to improve runtime performance when checking whether a newly reported event should displace an existing event memory entry
- Updated the arrangement of configuration parameters within Tresos Studio GUI to improve user experience
- ASCDEM-4284 Fixed known issue: Dem J1939 read functions also consider permanent DTCs that are only stored volatile
- Updated the memory mapping keywords used by the Dem service interfaces, when RTE is used
- Updated NvM header inclusion and access to NvM symbolic name values
- Updated the internal functions to improve runtime performance when checking the status of an indicator
- Updated header inclusion scheme
  - If a DemNvRamBlockId is configured in Dem, then NvM must be configured to include Dem.h.
- Added configuration constraint to restrict J1939DTCs to primary memory only

Module version 6.3.4

2021-02-12

- Implemented stubs for the APIs `Dem_DcmGetAvailableOBDMIDs()`, `Dem_DcmGetNumTIDsOfOBDMID()` and `Dem_DcmGetDTRData()` from AUTOSAR R4.3.1
- Improved documentation for `DemCallbackEventStatusChanged` and `DemCallbackEventDataChanged`
- Updated the stubs for the Dem APIs used to read infotype $08 and $0B, according to AUTOSAR R4.3.1

- Implemented stub for the API `Dem_DcmReadDataOfPID91()` from AUTOSAR R4.3.1
- ASCDEM-4163 Fixed known issue: Following a ClearDTC, a permanent DTC is erased before the permanent fault code cycle condition is satisfied
- Updated validation checks for configuration parameters
- ASCDEM-4290 Fixed known issue: Completion status of a disabled readiness group is wrongly shown as completed in PID$41

Module version 6.3.3

2021-01-22

- Removed definition of obsolete data types
- ASCDEM-4105 Fixed known issue: J1939 lamp flash status is returned wrongly if the lamp is available and OFF
- Corrected the debug macro name used in Dem_J1939DcmReadDiagnosticReadiness2()

Module version 6.3.2

2020-12-18

- ASCDEM-4221 Fixed known issue: Dem module generates a BSWMD file with non-unique short-names for packages
- Updated configuration check to consider that DemDataElementClass can also be referenced by DemSPNDataElementClassRef
- ASCDEM-4217 Fixed known issue: A DCY event might not be confirmed at DCY qualification under certain conditions
- Improved description of Dem APIs and configuration parameters

Module version 6.3.1

2020-11-20

- Optimized the validation check regarding DemUserDefMemoryId
- Updated default value for the unsupported configuration parameter `DemOccurrenceCounterProcessing` to align with the implemented behavior
- ASCDEM-4153 Fixed known issue: No EventStatusChanged notification if DEM_ONLY_THIS_CYCLE_AND_READINESS status bits are reset

Module version 6.3.0

2020-11-11

- Updated support for NvM explicit synchronization block callback calculation in service needs
- ASCDEM-4125 Fixed known issue: The APIs Dem_GetNextFreezeFrameData() and Dem_GetNextExtendedDataRecord() silently skip records that cannot fit within the provided buffer
- ASCDEM-4122 Fixed known issue: Under certain conditions, the API Dem_GetNextExtendedDataRecord() does not return DEM_NO_SUCH_ELEMENT after all relevant records are returned
- Removed duplicated template file Dem_Customization.h from generate\include\ folder
- Updated Dem to provide the C/S interface PowerTakeOff and the corresponding port only if OBD support is enabled
- Corrected missing APIs and hyperlinks for module reference
- ASCDEM-4031 Fixed known issue: A DCY event can be confirmed without qualification of the DCY under certain conditions
- ASCDEM-4196 Fixed known issue: Reset of OCC5 counter and setting of SI30.DEM_SI30_STATUS_WIR_SLC bit are not stored immediately
- Optimized run-time performance for NvM write trigger by reducing critical sections calls
- ASCDEM-4098 Fixed known issue: J1939 freeze frame is not stored correctly in all cases
- Added clear limit for ClearThisCycleAndReadinessStatus
- Removed all SERVER-CALL-POINTS from runnable entity CddIfClearDTC due to asynchronous ClearDTC processing
- ASCDEM-4013 Fixed known issue: Condition for erasing permanent DTCs after ClearDTC is stricter than required
  - Migration advice: With this change, the structure of the data stored in the NvM block for permanent memory is changed. Therefore, it is recommended to manually erase the NvM block while doing the SW update, if it is not configured to be handled automatically (i.e., if `DemNvDataConfigSignatureUsed` == false or `DemNvDataConfigSignatureCalcSet` != DEM_NVM_SIGNATURE_ALL).
- Refined descriptions of Dem calibration data in Dem_Bswmd.arxml file
- ASCDEM-4149 Fixed known issue: Configuration parameter DemJ1939DTCValue is not named according to AUTOSAR 4.2.2 in the schema
  - The configuration parameter DemJ1939DTC is updated to DemJ1939DTCValue. During integration, the parameter name will have to be updated.
- Updated the AUTOSAR R4.3.1 compliant Dem APIs used by Dcm from prototype to mass production quality

Module version 6.2.0

2020-09-25

- Optimized out not required NvM_WriteAll attribute for DEM_NVM_BLOCK_ID_PERMANENT
- Introduced discarding of event memory entries during Dem_Init() when NvM_SetDataIndex() fails
- Improved usage of DBG macros
- Added REQUIRES-INDEX attribute at sorted lists in configuration
- Optimized ROM consumption for event failure class index
- Updated category BITFIELD_TEXTABLE for CompuMethods EventStatusExtendedType and DebouncingStateType
- Improved handling of negative return values from NvM interfaces and inconsistent NvM data
- Limited the number of processed events per `Dem_MainFunction()` cycle for ClearDTC
- ASCDEM-4078 Fixed known issue: The Dem does not call the SW-C API <Module>_SetClearDTC for ClearDTC behavior DEM_CLRRESP_NONVOLATILE_FINISH

Module version 6.1.1

2020-08-28

- ASCDEM-3890 Fixed known issue: The Nv storage or clearing of an event memory entry can be omitted under specific race conditions if no Dem_Shutdown is executed
- Updated qualification concept for DCY and corresponding CDTC and WIR/MIL handling
- ASCDEM-4294 Fixed known issue: DCY event is not confirmed if no fault confirmation is configured and `DemResetConfirmedBitOnOverflow` is set to true
- ASCDEM-3922 Fixed known issue: A compiler error occurs when permanent storage with an NvM with conditional legacy symbolic names support is used, and no immediate storage/clear is activated
- Implemented the configuration parameter `DemMaxHandledInitMonitorReenabledPerScheduling` that limits the number of `DemInitMonitorForEvent` callbacks triggered in each call of `Dem_MainFunction()`
- ASCDEM-3991 Fixed known issue: OCC counters and SI30 are not stored immediately
- ASCDEM-3993 Fixed known issue: CSLF and CSFF are not restored as immediate data
- ASCDEM-3877 Fixed known issue: DTC status of combined events is not updated correctly in Dem_SetAgingCycleState() under certain conditions
- Changed optional configuration references of events to disabled by default

Module version 6.1.0

2020-07-20

- ASCDEM-3959 Fixed known issue: Dem_ClearDTC may wrongly return DEM_CLEAR_BUSY or the clearing is executed multiple times
- Extended J1939 support to work independently from OBD support

Module version 6.0.0

2020-06-26

- Changed all NO_INIT memory sections to CLEARED
- Added J1939 support for Dem
- Added support for parallel processing of OBD and UDS protocols by updating the Dem-Dcm APIs according to ASR R431; relevant changes are:
  - Added possibility to configure diagnostic clients via `DemClientId`
  - Feature *OBD powertrain data* updated the APIs `Dem_DcmGetDTCOfOBDFreezeFrame()` and `Dem_DcmReadDataOfOBDFreezeFrame()`, also to support UDS service ReadDTCInformation (0x19) sub-function 0x05
  - Feature *DTC filtering* updated the APIs `Dem_SetDTCFilter()`, `Dem_GetNumberOfFilteredDTC()`, `Dem_GetNextFilteredDTC`, `Dem_GetNextFilteredDTCAndFDC()`, `Dem_GetNextFilteredDTCAndSeverity()`, `Dem_GetTranslationType()` and `Dem_GetDTCStatusAvailabilityMask()`
  - Feature *DTC selection mechanism* introduced the new APIs `Dem_SelectDTC()` and `Dem_GetDTCSelectionResult()`
  - Feature *control DTC setting* updated the APIs `Dem_DisableDTCSetting()` and `Dem_EnableDTCSetting()`
  - Feature *DTC record update* updated the APIs `Dem_DisableDTCRecordUpdate()` and `Dem_EnableDTCRecordUpdate()`
  - Feature *DTC status* updated the API `Dem_GetStatusOfDTC()`
  - In prototype quality, the feature *DTC attribute* updated the APIs `Dem_GetSeverityOfDTC()`, `Dem_GetFunctionalUnitOfDTC()` and `Dem_GetDTCByOccurrenceTime()`
  - In prototype quality, the feature *clear DTC* updated the API `Dem_ClearDTC()` and introduced the new API `Dem_GetDTCSelectionResultForClearDTC()`
  - In prototype quality, the feature *read event related data* updated the APIs `Dem_GetSizeOfExtendedDataRecordSelection()`, `Dem_GetSizeOfFreezeFrameSelection()`, `Dem_GetNextExtendedDataRecord()` and `Dem_GetNextFreezeFrameData()`, and introduced the new APIs `Dem_SelectExtendedDataRecord()` and `Dem_SelectFreezeFrameData()`

- In prototype quality, the feature *stored DTC filter* updated the APIs `Dem_DcmSetStoredDTCFilter()` and `Dem_DcmGetNextFilteredStoredDTC()`
- In prototype quality, the feature *freeze frame record filter* updated the APIs `Dem_SetFreezeFrameRecordFilter()` and `Dem_GetNextFilteredRecord()`
- In prototype quality, the feature *user-defined memory identifier* updated the type `Dem_DTCOriginType` impacting the input parameter `DTCOrigin` of the APIs `Dem_DcmSetStoredDTCFilter()`, `Dem_SetDTCFilter()`, `Dem_SelectDTC()`, `Dem_GetEventMemoryOverflow()`, `Dem_GetNumberOfEventMemoryEntries()` and the operation `ClearDTC` of the C/S-interface `CddIf`.

  The configured user defined memory Id is internally mapped to the secondary memory

  Additional notes:
  - All clients have access to all diagnostic data (no support for `DemEventMemorySet`)
  - The same client shall not call multiple Dem-Dcm APIs in parallel
  - `Dem_SelectDTC()` and its dependent APIs do not return `DEM_BUSY`
  - The following APIs have their functionality restricted to one client at a time: `Dem_DisableDTCSetting()`, `Dem_EnableDTCSetting()`, `Dem_DisableDTCRecordUpdate()`, `Dem_EnableDTCRecordUpdate()`, `Dem_DcmSetStoredDTCFilter()`, `Dem_DcmGetNextFilteredStoredDTC()`, `Dem_SetFreezeFrameRecordFilter()`, `Dem_GetNextFilteredRecord()`
  - While clearing is being processed for one client, a new clear request from a different client will return `DEM_CLEAR_BUSY`
  - `Dem_GetDTCSelectionResult()` returns `DEM_WRONG_DTC` for a selection of a single OBD DTC
  - The service Id of APIs `Dem_DcmSetStoredDTCFilter()` and `Dem_DcmGetNextFilteredStoredDTC()` is changed from `0x90`, respectively `0x91` to `0xD0`, respectively `0xD1`
- Implemented asynchronous triggering of DemInitMonitorForEvent callbacks when ControlDTCSetting is re-enabled by triggering at most 50 configured callbacks in each call of `Dem_MainFunction()`
- Implemented asynchronous processing of `Dem_ClearDTC()` requests in `Dem_MainFunction()` context

Module version 5.20.0

2020-05-22

- ASCDEM-3795 Fixed known issue: NvM blocks are not generated by the Calculate Service Needs wizard
- Added Calibration support for Dem time-based debouncing
- Updated calibration interface of counter based debouncing: removed unused `DemDebounceCounterBased` and optimized size of `Dem_DebounceCounterClassIdx`

Module version 5.19.9

2020-04-24

- ASCDEM-3751 Fixed known issue: Dem_GetNextDTCAndSeverity() does not consider DemDtcStatusAvailabilityMask when reporting DTCStatus
- Improved calibration short names for OBD DTC with combination
- Extended calculation of DemNvMConfigSignature
- ASCDEM-3357 Fixed known issue: DTC record update cannot ensure event entry data consistency for ReadDTCInformation

Module version 5.19.8

2020-03-25

- Added support for replacement DTCs via Multi-Event Triggering limited to Event Reports

Module version 5.19.6

2020-02-21

- ASCDEM-3566 Fixed known issue: Callback functions Dem_NvMWriteFinishedPermanentMemory(), Dem_NvMWriteCopyPermanentMemory() and Dem_NvMReadCopyPermanentMemory() are not available, if immediate storage is disabled
- Added dedicated memory sections for calibratable data introduced with Event Combination with OBD
- Added configuration switch for Dem_RepIUMPR* functions
- ASCDEM-3676 Fixed known issue: Miscalculation of total freeze frame sizes used for validation in generator
- ASCDEM-3306 Fixed known issue: Dem_ResetEventDebounceStatus() API causes segmentation fault by calling for events with DemDebounceMonitorInternal
- ASCDEM-3680 Fixed known issue: Dem_ClearDTC() gets blocked by NvM_WriteBlock() that is triggered sporadically by event reports
- Note that module version 5.19.5 is skipped and used for a special extra delivery

Module version 5.19.4

2019-10-11

- Support for Event Combination together with OBD

- ASCDEM-3452 Fixed known issue: Dem_DcmReadDataOfPID01() does not report confirmed OBD DTCs that are not assigned to a readiness group
- ASCDEM-3441 Fixed known issue: OCC6 is wrongly increased more than once if a BSW event is reported as FAILED multiple times in the same operation cycle
- Dem-callouts moved into designated memory sections
- Removed warning generated by DemIndicatorFailureCycleRef even when the parameter is disabled, or not editable

Module version 5.19.3

2019-09-06

- ASCDEM-3342 Fixed known issue: Cleared DTCs could reappear when entering DEM_OPCYC_OBD_-DCY
- ASCDEM-3380 Fixed known issue: Wrong value for 'DemInternalDataElement' set to 'DEM_EVENT_ID' for combined events
- Added dedicated memory sections for calibratable data

Module version 5.19.2

2019-08-09

- ASCDEM-3295 Fixed known issue: Using the DTC value '0' when calling an API for the first time might lead to out-of-bounds access. Root cause already fixed with ASCDEM-3264

Module version 5.19.1

2019-07-12

- Corrected behavior where update of UDS status byte was not protected against concurrent access that may lead to inconsistent status
- Removed nested critical section for permanent memory processing
- Improved handling and documentation of consecutive calls of Dem_Init() and Dem_PreInit()
- ASCDEM-3279 Fixed known issue: ConfirmedDTC bit is wrongly reset on the next Dem_MainFunction() for the Combined Event DTC status

Module version 5.19.0

2019-06-14

- ASCDEM-2910 Fixed known issue: Dem might calculate a wrong UDS status if event processing is interrupted by Dem_ClearDTC or Dem_ResetEventStatus
- Optimized MemMap mapping of pointer arrays
- ASCDEM-3081 Fixed known issue: Dem does not generate A2L files
- Optimized calibratable data. A2L files should be regenerated regardless of configuration has changed or not
- Support for Multiple OBD freeze frame: Core Functionality (for record number 0x00)
- ASCDEM-3270 Fixed known issue: A compiler error occurs when RTE is used, calibration is enabled and no indicators are configured
- ASCDEM-3264 Fixed known issue: Wrong handling for DTC value 0 for several APIs
- ASCDEM-3284 Fixed known issue: Dem wrongly reports DET errors in case of ReportErrorStatus from satellite core

Module version 5.18.16

2019-05-17

- Internal module improvement. This module version update does not affect module functionality

Module version 5.18.15

2019-04-18

- Support for BSW event reporting from multiple cores - protect APIs for incorrect accessing
- Optimized EventClass configuration description table
- ASCDEM-3183 Fixed known issue: False warning on unused DataElementClass in a specific case

Module version 5.18.14

2019-04-11

- Implemented support for ASR 4.3 compatibility for selected interfaces

Module version 5.18.13

2019-03-22

- Internal module improvement. This module version update does not affect module functionality

Module version 5.18.12

2019-03-21

- ASCDEM-3144 Fixed known issue: Incorrect handling of OBD Freeze Frame in specific race conditions
- Implemented support for dynamic event availability via API

Module version 5.18.11

2019-02-20

- ASCDEM-3133 Fixed known issue: Callout function declarations of DemCallbackMemStackMainFuncTrigger and DemCalloutDynamicDTCFnc are not inside a CODE section
- ASCDEM-3134 Fixed known issue: Wrong compiler abstraction used for Read DataElement operations of the SenderReceiver interface
- Optimized implementation of Dem internal search function for the case when no event memory entry exists
- Implemented support for reset of Dem NvM data after incompatible configuration update
- Optimized implementation of Dem_SetOperationCycleState() - Init Monitor Callbacks reworked
- ASCDEM-3079 Fixed known issue: Confirmed emission-related BSW events are not stored immediately in permanent memory
- Clarified integration code hint for configuration parameter DemCallbackMemStackMainFuncTrigger
- Optimized implementation of Dem_SetOperationCycleState() - Number of critical section activations reduced
- Optimized implementation of Dem_SetOperationCycleState() - Disable via configuration parameter DemCallbackEventStatusChangedOpCycStartSupport

Module version 5.18.10

2019-01-23

- ASCDEM-2924 Fixed known issue: Memory cannot be cleared after system reinitialization
- ASCDEM-2931 Fixed known issue: Possible null pointer access or wrong data retrieved if no events have aging configured
- ASCDEM-3083 Fixed known issue: Warning when import Dem_swc_internal.arxml file

- Extended event displacement algorithm with criteria for OBD and TNCTOC
- ASCDEM-2678 Fixed known issue: OBD freeze frame might store incorrect data for large configurations
- ASCDEM-3072 Fixed known issue: OCC counters are not stored at Dem_Shutdown
- ASCDEM-2589 Fixed known issue: No NvM storage of event-related data on update of OCC5 value occurs if immediate storage is enabled under particular conditions
- ASCDEM-3109 Fixed known issue: Dem_DTCFilter: Memory section conflict between declaration and definition
- ASCDEM-3108 Fixed known issue: Dem module does not generate if queue length for DemEventInfoPort operations is calculated by RTE

Module version 5.18.9

2018-11-23

- Internal module improvement. This module version update does not affect module functionality

Module version 5.18.8

2018-10-26

- Refined the mapping in the memory of variables and constants for optimized memory consumption
- ASCDEM-2677 Fixed known issue: Dem does not compile if DemAgingAllowedSeperateFlag has a wrong value
- Support for DataService interface with EventID

Module version 5.18.7

2018-10-09

- Support for BSW event reporting from multiple cores
- ASCDEM-2492 Fixed known issue: Dem module does not fully support BSW Distribution for multi-core systems

Module version 5.18.6

2018-09-28

- Support identical OBD DTC feature for ACG-8

Module version 5.18.5

2018-08-24

- Changed signatures of the `DemCallbackMemStackMainFuncTrigger` and `DemCalloutDynamicDTCFnc` callout functions by removal of the const qualifier of the parameters

Module version 5.18.4

2018-07-27

- ASCDEM-2896 Fixed known issue: Dem Module does not compile if SenderReceiver interface is enabled and when no DemExtendedDataRecordClass or CS DataElements are configured
- ASCDEM-2886 Fixed known issue: Dem might return wrong data when Operation Cycles non-referenced from any DemOperationCycleRef or unused Aging Cycles are configured

Module version 5.18.3

2018-06-22

- Added support for Dem_SetDTCSuppression() API according to AUTOSAR 4.3.0
- ASCDEM-2749 Fixed known issue: Dem module wrongly reports status change notifications
- Extended list of supported PIDs with the OBD-specific Dcm <-> Dem services with *Internal calculation of PID $21 handling - distance traveled while MIL is activated*

Module version 5.18.2

2018-05-25

- ASCDEM-2743 Fixed known issue: Dem_SetWIRStatus does not control the WIR-bit of ECU internal events
- Enhance checks and description of side allocation feature (config parameter DemCalloutDynamicDTCFnc)
- Implemented support for project specific Dem_ReportErrorStatus implementation

Module version 5.18.1

2018-04-20

- Internal module improvement. This module version update does not affect module functionality

Module version 5.18.0

2018-03-16

- Implemented missing memory sections in the Dem software component description causing incompilable code

Module version 5.17.29

2018-02-28

- Optimized code to eliminate dead assignments
- Optimized code for ordered DTC reporting
- ASCDEM-2675 Fixed known issue: Dem_GetSizeOfFreezeFrameByDTC() might return wrong value for large development freeze frames
- Implement support for SenderReceiver interface
- Changed Primitive Implementation Data Types to Redefinition Implementation Data Types for unspecified Implementation Data Types
- Implemented compliance to MISRA-C:2012

Module version 5.17.28

2017-12-23

- Implemented new configuration option for configurable queue sizes for Dem client server operations
- ASCDEM-2636 Fixed known issue: Dem_DcmCancelOperation() cancels ClearDTC requests of another user
- ASCDEM-2635 Fixed known issue: Dem_DcmCancelOperation() might not cancel a ClearDTC request
- Implement Reset/freeze of event debouncing via Dem_ResetEventDebounceStatus() API
- ASCDEM-2638 Fixed known issue: The DTC clearing process is not locked for other users in case of an NvM memory write error
- Implemented optimization for long Dem code generation time
- Support for Reset/freeze of event debouncing via non-fulfilled enable conditions and disabled ControlDTCSetting

- Support for interface DiagnosticInfo according to ASR 4.2

Module version 5.17.27

2017-10-13

- ASCDEM-2587 Fixed known issue: Data structures for DemExternalCSDataElement are not correctly generated for specific configuration
- ASCDEM-2581 Fixed known issue: Incomplete syntax in configuration check for DemFreezeFrameClassRef causes a generation error
- Optimized speed for configuration verifier checks
- Support for Operation Cycle Counters 'cycle since first failed' and 'cycle since first last failed' according to R4.2.1
- Implemented support for Dem_ClearDTC processing burst reduction
- Removed AUTOSAR 3.x compliant symbolic name value macros and updated the logic to only provide AUTOSAR 4.0.2 compliant macros if macro DEM_PROVIDE_LEGACY_SYMBOLIC_NAMES is defined

Module version 5.17.26

2017-07-24

- Implemented R4.2.1 clarification for EventDataChanged triggering-mechanism

Module version 5.17.25

2017-07-13

- Implemented support for healing completion based aging
- ASCDEM-2574 Fixed known issue: Incorrect handling of event combination data array causes out of bunds access
- ASCDEM-2454 Fixed known issue: Warning indicator calibration for BSW events does not work properly

Module version 5.17.24

2017-06-26

- Implemented support for immediate storage of internal events

- Implemented support for considering untested cycles for aging behavior
- Corrected misleading DET reports in store functionality
- ASCDEM-2506 Fixed known issue: Data structures for enable condition are not correctly generated for specific configuration
- ASCDEM-2491 Fixed known issue: Combined DTC status byte is not updated when the DTC status of combined event is modified

Module version 5.17.23

2017-04-28

- ASCDEM-2484 Fixed known issue: Generation error caused by missing invalidation for DemEvtCmbCommonParamMaster
- ASCDEM-2463 Fixed known issue: Absent filter handling for DTCStatusAvailabilityMask may impact reported DTCs by status mask for ReadDTCInformation (0x19) service
- ASCDEM-2482 Fixed known issue: Linker errors if no DemIndicatorAttribute exists for any event

Module version 5.17.22

2017-03-21

- ASCDEM-2387 Fixed known issue: Calibration of aging allowed flag can lead to out of bound access
- ASCDEM-2399 Fixed known issue: Compile error when legacy symbolic names are disabled via macro DEM_DONT_PROVIDE_LEGACY_SYMBOLIC_NAMES
- ASCDEM-2336 Fixed known issue: Warning indicator processing may be skipped for aged BSW events
- ASCDEM-2337 Fixed known issue: Warning indicator processing may be skipped for BSW events which are displaced
- ASCDEM-2372 Fixed known issue: StatusChanged notifications missing in API Dem_ResetReadiness()
- DTC significance as Internal Data Element with output as extended data
- ASCDEM-2410 Fixed known issue: Possible overflow for aging cycle symbolic name values
- Event id as internal data element with output as extended data
- ASCDEM-2430 Fixed known issue: Code generator does not correctly handle DemCallbackClearEventAllowed parameter
- ASCDEM-2427 Fixed known issue: Linker error: undefined reference to Dem_UpdateAgingCounter
- ASCDEM-2423 Fixed known issue: BSW Events: OCC7 counter is incremented more than once per operation cycle

- ASCDEM-2424 Fixed known issue: BSW Events: Fault confirmation counter is incremented more than once per operation cycle
- ASCDEM-2394 Fixed known issue: OBD freeze frame is not cleared in specific cases
- ASCDEM-2407 Fixed known issue: If DTC record update is disabled and event is aged, combined DTC status is not updated in case of ECU sleep and wakeup
- ASCDEM-2413 Fixed known issue: An event with immediate storage ON may not be stored immediately
- Corrected invalid XPath expression due to invalid DemMILIndicatorRef
- User controlled WarningIndicatorRequested-bit
- ASCDEM-2451 Fixed known issue: Synchronous server call points for FreezeFrameData interfaces are not generated in the Dem_MainFunction runnable
- ASCDEM-1426 Fixed known issue: Invalid OCC2 increment for multiple passed BSW events in one Dem_MainFunction cycle
- ASCDEM-2436 Fixed known issue: DemMaxNumberEventEntry[Permanent|Primary|Secondary|Mirror] configured to 255 may lead to an endless loop

Module version 5.17.21

2016-12-16

- Provided Handle-Id Wizards for all platforms

Module version 5.17.20

2016-11-04

- Implemented support for Configuration Signature
- ASCDEM-2240 Fixed known issue: Wrong calculation of freeze frame size if calibration is supported
- Implemented support of trigger on TestFailed for DemExtendedDataCapture and DemFrezeFrameCapture
- ASCDEM-2327 Fixed known issue: Code generator fails with combined events, common freeze frame enabled, and configured freeze frame record numbers
- Implemented support for API Dem_GetDTCByOccurenceTime()
- ASCDEM-2331 Fixed known issue: XPath-expression error if combined slave event uses the MIL
- ASCDEM-2346 Fixed known issue: Compiler error occurs if event combination is enabled and fault confirmation is not used
- Implemented optional support for Monitor Re-initialization callback DemInitMonitorForEvent with InitMonitorReason DEM_INIT_MONITOR_REENABLED.

- ASCDEM-2340 Fixed known issue: Readiness group cannot be calibrated
- ASCDEM-2339 Fixed known issue: Indicator of combined events with DemEvtCmbCommonParamMaster = false cannot use events failure cycle and threshold
- ASCDEM-2342 Fixed known issue: ConfirmedDTC bit might not be set for events assigned to OBD driving cycle
- ASCDEM-2373 Fixed known issue: Permanent event memory entry is only removed if all assigned indicators are healed
- Implemented Configurable clear event allowed behavior.
- ASCDEM-2378 Fixed known issue: Incorrect reset of warning indicator failure counter

Module version 5.17.0

2016-07-22

- ASCDEM-2241 Fixed known issue: Dem compilation fail due to missing MIL reference
- Updated Dem_GetSizeOfExtendedDataRecordByDTC() API, in case DemGeneral/DemOBDSupport is disabled, to handle the DTCExtDataRecordNumber of 0xFE with DEM_GET_SIZEOFEDRBYDTC_W_RNUM NRC.
- ASCDEM-2247 Fixed known issue: OBD extended data records are wrongfully checked against 90U (decimal value)
- ASCDEM-2258 Fixed known issue: Dem_GetSizeOfExtendedDataRecordByDTC() API exits with incorrect return value
- Updated Fault Confirmation and WIR bit setting behavior:
  - *Updated Fault Confirmation handling for setting the ConfirmedDTC status bit according to the number of TestFailedThisOperationCycle transition configured via* `DemEventFailureCycleCounterThreshold` *(refer to RfC #70785).*
  - *Updated Warning Indicator handling for switching on the Warning Indicator status and setting the WarningIndicatorRequested status bit according to the number of TestFailedThisOperationCycle transition configured via* `DemIndicatorFailureCycleCounterThreshold` *(refer to RfC #71313).*
  - *Multi-indicator behavior has been updated for the Warning Indicator ON status transition. All indicators assigned to the same event are handled uniformly with the new threshold behavior.*
- ASCDEM-2249 Fixed known issue: Parameter 'Common part of Development Freeze Frame A Class' could be accessed with invalid reference

Module version 5.16.6

2016-06-15

- Implemented support for Event Combination - Combination On Storage
- ASCDEM-2229 Fixed known issue: File Dem_Bswmd.arxml has no unique short name for Dem_DebounceCounterClassIdx
- ASCDEM-2230 Fixed known issue: File Dem_Bswmd.arxml has wrong type reference for Dem_DebounceCounterClassIdx
- Implemented separate memory section `SEC_VAR_SAVED_ZONE` for non-volatile data variables. New memory sections will have to be set accordingly during integration(e.g. map it along with SEC_VAR_NOINIT)

Module version 5.16.5

2016-05-25

- ASCDEM-2197 Fixed known issue: Dem does not compile when legacy symbolic names are disabled via macro DEM_DONT_PROVIDE_LEGACY_SYMBOLIC_NAMES
- Implemented support for additional internal data element priority
- ASCDEM-2198 Fixed known issue: Dem_ReportErrorStatus() may report an unjustified Det error during pre-initialization phase
- Implemented support for additional internal data element "Aging Down Counter"

Module version 5.16.4

2016-04-29

- Implemented support for reading the number of event memory entries in API `Dem_GetNumberOfEventMemoryEntries`
- Implemented internal source code improvements
- Implemented support for 0x19-04 positive response with calibrated unused freeze frame class
- ASCDEM-2175 Fixed known issue: Service 0x19 03 loses development and OBD freeze frame records in `Dem_GetNextFilteredRecord`
- Added configuration check to limit the usage of FDC_THRESHOLD for DemFreezeFrameCapture and DemExtendedDataCapture
- Implemented support for clearing OBD freeze frame when event gets aged or if PDTC bit is cleared before CDTC bit was set

Module version 5.16.3

2016-04-01

- ASCDEM-2090 Fixed known issue: `Dem_DcmReadDataOfPID41()` may return wrong value if it is interrupted by `Dem_SetOperationCycleState()` or `Dem_SetEventDisabled()`

Module version 5.16.2

2016-03-04

- ASCDEM-2106 Fixed known issue: Dem_DcmReadDataOfPID31 may return wrong value if it is interrupted by Dem_MainFunction or ClearDTC
- Implemented calibration of event availability and event storage handling for DTC value zero
- Improved event memory handling in `Dem_MainFunction()` regarding run-time and critical section length
- ASCDEM-2084 Fixed known issue: Data of Dem permanent event memory cannot be restored at startup
- Implemented freeze frame "online" calibration (FF class and DIDs in FF)

Module version 5.16.1

2016-02-15

- Implemented ClearDTC for disabled events and events without DTCs
- ASCDEM-2060 Fixed known issue: Debounce class is not generated correctly without license DIAG_VCC or EB_DIAG_FP1
- Implemented support common freeze frame for events without individual freeze frame
- ASCDEM-2074 Fixed known issue: DemCallbackMemStackMainFuncTrigger is not editable if DemClearDTCBehavior == NONVOLATILE and immediate storage is disabled
- Added support for Debug & Trace with custom header file configurable via parameter `BaseDbgHeaderFile`
- Added development freeze frame replacement when confirmed event is aged
- ASCDEM-2092 Fixed known issue: Event/DTCStatusChanged callback not called for PDTC clear in case of aged event replacement with DEM_TRIGGER_ON_FDC_THRESHOLD

Module version 5.16.0

2016-01-15

- Provision of ExtendedDataRecords with always available data via 0x19 also if event is NOT stored
- Implemented support for PID1C in API `Dem_DcmReadDataOfPID1C`
- Added support for convenient calibration of aging allowed
- Optimized enable condition data structures and processing

- Implemented calibration support for `DemEnableConditionGroupRef`
- ASCDEM-1982 Fixed known issue: Initialization of indicators write out of bound in case of calibration support is enabled
- ASCDEM-1966 Fixed known issue: OBD freeze frame is saved on unconfirmedDTC timing, but not on testFailed
- Adapted data structure according advanced event displacement (`DemAdvDisplacementPassiveTableLine`) based on vendor specific requirements
- Added support for calibration of DemEventOBDReadinessGroup with multiple readiness groups per event
- Implemented measurement support for Dem_FFSegIdx<Name>
- Implemented support for PID30 in API `Dem_DcmReadDataOfPID30`
- ASCDEM-1983 Fixed known issue: ConfirmedDTC bit is set for aged events if DemExtendedDataCapture == DEM_TRIGGER_ON_FDC_THRESHOLD
- ASCDEM-1967 Fixed known issue: Counter-based debouncing does not perform jump behavior in initial state. Hint: Please check your NvM configuration after applying this fix, since the size of the Dem default NV-block may change
- Extended FreezeFrameRecordFilter (0x19-03) with OBD- and Development freeze frames
- Implemented support for PID31 in API `Dem_DcmReadDataOfPID31`
- Implemented selectable PendingDTC bit behavior
- Implemented unused value for DemPidIdentifier
- Implemented support for reporting order of event memory entries

Module version 5.15.4

2015-11-06

- Improved usability in tresos Studio GUI by introducing a new tab for related OBD parameters
- ASCDEM-1907 Fixed known issue: MIL indicator is activated, even if DemObdDTC was calibrated to "0"
- ASCDEM-1919 Fixed known issue: The API "Dem_GetSizeOfFreezeFrameByDTC" calculates a wrong size for all freezeframes (0xFF) if an OBD-freezeframe is configured
- ASCDEM-1902 Fixed known issue: Calibration of configuration parameter DemEventFailureCycleCounterThreshold does not change qualification of UDS status bit 7 (warningIndicatorRequested)
- ASCDEM-1903 Fixed known issue: Service 0x0a Permanent faults are not removed if DemExtendedDataRecordTrigger is set to DEM_TRIGGER_ON_FDC_THRESHOLD
- Added support for Common OperationCycleRef in EventClass with calibration
- ASCDEM-1703 Fixed known issue: Dem might use uninitialized data when processing event reports during start-up

Module version 5.15.3

2015-10-02

- ASCDEM-1839 Fixed known issue: OBD Aging can not be disabled

Module version 5.15.2

2015-09-07

- ASCDEM-1755 Fixed known issue: If permanent memory is enabled and immediate storage is disabled, the Dem might use wrong NvM blocks
- Implemented support for freeze frame classes and extended data records greater than 255 byte.
- Removed compiler warning concerning zero-size DevFFAData[] array
- ASCDEM-1616 Fixed known issue: P-Ports DemOperationCycle may use wrong port-defined argument values or port / macro may not been generated

Module version 5.15.1

2015-08-28

- Implemented support for record number 0x00 in APIs `Dem_GetFreezeFrameDataByDTC()` and `Dem_GetSizeOfFreezeFrameByDTC()`
- ASCDEM-1843 Fixed known issue: PID41 DataB bit2 is not forced to enabled.
- ASCDEM-1816 Fixed known issue: Dem_GetFreezeFrameDataByDTC() does not return `DEM_GET_FF_DATABYDTC_WRONG_RECORDNUMBER` when RecordNumber 0x00 is requested
- ASCDEM-1824 Fixed known issue: If OCC2-based aged event reoccurs, previous data are not deleted
- ASCDEM-1845 Fixed known issue: OCC5 is not initialized when an event is stored in memory for the first time
- ASCDEM-1812 Fixed known issue: ClearDTC and aging do not correctly reset warning indicator-related status
- Implement support for calibration of PID41 DataB bit2.
- Implemented support for Indicator class combination.
- ASCDEM-1855 Fixed known issue: Compile error if development freeze frame is configured but DET is disabled
- Implemented support for DTC disabling via calibration
- Implemented support for deleting internal events via `Dem_ClearDTC()`

Module version 5.15.0

2015-07-10

- Implemented common freeze frame support
- Added support for calibration of event classes.
- Added support for a minimum number of event classes.
- Added support for Indicator classes.
- Added support for a minimum number of Indicator classes.
- ASCDEM-1800 Fixed known issue: Wrong OBDFreezeFrame data stored if internal data elements are configured for a DemPidClass
- ASCDEM-1771 Fixed known issue: Dem_GetSizeOfFreezeFrameByDTC() and Dem_GetSizeOfExtendedDataRecordByDTC() do not return error if called with wrong record number for non-stored DTC

Module version 5.14.1

2015-06-20

- ASCDEM-1725 Fixed known issue: NvM_ReadAll() does not load the Dem status block if nonvolatile behavior is used for Dem_ClearDTC()

Module version 5.14.0

2015-06-19

- Implemented support for enhanced OBD readiness groups
- Implemented support for advanced event displacement
- ASCDEM-1727 Fixed known issue: A wrong configuration check for the NvMWriteRamBlockToNvCallback name breaks the OBD permanent memory feature
- Implemented support for filtering of stored DTCs via Dcm (vendor specific)
- Extended upper multiplicity of DemFreezeFrameClass to 65535 (RFC #61653)
- ASCPD-179 Added additional calibration parameter
- ASCDEM-1701 Fixed known issue: Using nonvolatile behavior for Dem_ClearDTC() may block it or lead to incomplete data in NvM
- Changed default value of the DemExtendedDataCapture and DemFreezeFrameCpature configuration parameters to DEM_TRIGGER_EVENT_MEMORY_STORAGE
- Dem_SetSI30Symptom is not available if Trigger FiM Reports = false
- Support for OCC5 based aging of OBD events

- ASCPD-201 Implemented feature "event burst reduction"/"processor load reduction"
- Implemented DTC-group `DEM_DTC_GROUP_EMISSION_REL_DTCS` as a optional group
- ASCDEM-1698 Fixed known issue: NVM_BLOCK_DEM_PERMANENT is not generated correctly by the Calcualte Service Needs assistant under particular conditions
- ASCPD-199 Implemented additional freeze frame support for development purpose
- ASCPD-179 Added additional calibration parameter
- Modified the optional DTC-group `DEM_DTC_GROUP_EMISSION_REL_DTCS` to statically contain all DTCs with kind "emission-related" (i.e. the contained DTCs are not configurable any more) and updated `Dem_ClearDTC()` accordingly

Module version 5.13.0

2015-04-24

- ASCDEM-1526 Fixed known issue: Compile error for undeclared Dem_FFRecNumerationClass[] occurs if no freeze frames are configured, but configured freeze frame record numbers are enabled
- Implemented support for OCC5 (operation cycle counter 5)
- ASCDEM-1634 Fixed known issue: Linker error for extended FiM if `DemSupportFiMFDCThreshold` is enabled and `DemExtendedDataCapture` is not set to DEM_TRIGGER_ON_FDC_THRESHOLD
- ASCDEM-1538 Fixed known issue: Configuration check missing: NvM is triggered with BlockId 0

Module version 5.12.2

2015-03-27

- Implemented support for OBD Mode $04
- ASCDEM-1511 Fixed known issue: The precompiler defines DEM_OPCYC_OBD_DCY_ID and/or DEM_OPCYC_IGNITION_ID are not generated under certain conditions
- ASCDEM-1540 Fixed known issue: Dem module triggers a DET error during NvM_WriteAll
- ASCPD-166 Implemented support of OBD readiness groups in Pid 0x01 and Pid 0x41
- Added support for further RTE interfaces according to AUTOSAR 4.2.1: `DiagnosticInfo:GetEventFreezeFrameData`, `DiagnosticInfo:GetDTCOfEvent`, `CallbackEventDataChanged`, `GeneralCallbackEventDataChanged`, `CallbackEventStatusChange`, `GeneralCallbackEventStatusChange`, `DataServices`
- Added vendor specific configuration switch `DemOBDIumprFunctionsEnabled` to disable the IUMPR functions: `Dem_RepIUMPRFaultDetect()`, `Dem_RepIUMPRDenLock()` and `Dem_RepIUMPRDenRelease()`

- Implemented support for OBD driving cycle

Module version 5.12.1

2015-03-06

- Implemented support for OBD DTC
- Implemented support for OBD Freeze Frame
- ASCDEM-1609 Fixed known issue: Event qualification with PREFAILED/PREPASSED report doesn't work if the first PREFAILED/PREPASSED report shall qualify the event as FAILED/PASSED
- Improved calling strategy for `EventDataChanged` callback on overflow indication
- ASCDEM-1458 Fixed known issue: Unnecessary trigger of EventStatusChanged / DTCStatusChanged callbacks
- ASCDEM-1510 Fixed known issue: "EventStatusChanged"/"DTCStatusChanged" callbacks reports a wrong status for BSW events

Module version 5.12.0

2015-02-17

- ASCPD-179 Added first calibration parameter
- ASCPD-176 Added OBD variant
- ASCPD-181 Added extended FiM support
- ASCDEM-1483 Fixed known issue: "EventStatusChanged" / "DTCStatusChanged" callbacks report a wrong status if event memory is already full
- ASCDEM-1460 Fixed known issue: If `DemIndicatorFailureCycleSource` is `DEM_FAILURE_CYCLE_INDICATOR`, the configuration parameters `DemIndicatorFailureCycleRef` and `DemIndicatorFailureCycleCounterThreshold` are erroneously not editable
- ASCDEM-1508 Fixed known issue: The data structures used for permanent memory are mapped to the incorrect memory map section
- Optimized implementation of `Dem_GetExtendedDataRecordByDTC()` and `Dem_GetEventExtendedDataRecord()`
- Added licence-based feature-set for:
  - ASCPD-225 unconfirmed threshold mechanism (supporting additional extended data and freeze frame capture, DTC status indicators, fault detection counters and operation cycle counters)
  - ASCPD-226 extended displacement mechanism (configurable)

- ASCPD-227 extended aging mechanism (configurable), not removing aged event entries unless space is needed

- ASCDEM-1501 Fixed known issue: If 'immediate storage' is disabled, the NvM callback functions '`Dem_NvMWriteFinishedPermanentMemory`', '`Dem_NvMWriteCopyPermanentMemory`', and '`Dem_NvMReadCopyPermanentMemory`' are not available
- ASCDEM-1534 Fixed known issue: If OBD support and RTE usage are enabled, the Dem module generates invalid content
- ASCDEM-1478 Fixed known issue: EventFailureCycleCounter is incremented under wrong conditions
- ASCDEM-1503 Fixed known issue: If permanent memory is enabled, the `Dem_GetNextFilteredDTCAndFDC` and `Dem_GetNextFilteredDTCAndSeverity` functions are returning DEM_E_WRONG_CONDITION if DTC origin filter is set to DEM_DTC_ORIGIN_PERMANENT_MEMORY
- ASCDEM-1617 Fixed known issue: The generation of Dem module fails if `DemMILIndicatorRef` parameter is enabled with invalid reference and not editable
- ASCDEM-1518 Fixed known issue: `Dem_ClearDTC` might not delete an event memory entry consistently if events are reported/qualified in parallel
- Added support for multiple `Dem_<...>ClearDTC()` interfaces according to AUTOSAR 4.2.1
- Extended upper multiplicity of DemDidClass and DemDataElementClass to 65535 (RFC #55110)
- ASCDEM-1756 Fixed known issue: Wrong internal data element values may be returned and data might get corrupted if more than 242 DemDataElementClass elements are configured

Module version 5.9.100

2014-10-22

- Removed writing of data initialization to NvRAM for empty memory entries if *immediate storage* is enabled to reduce NvRAM aging
- Updated setting of the CDTC status bit which is now triggered by an event report and does not need a restart of the operation cycle
- ASCDEM-1220 Fixed known issue: Status bit CDTC might get set incorrectly if *fault confirmation* is enabled and *immediate storage* is in use
- Replaced simple semaphore mechanism in Dcm interface functions by critical sections
- ASCDEM-1286 Fixed known issue: CDTC bit is not set in specific configurations with multiple events with EventFailureCycleCounterThreshold enabled
- ASCDEM-1317 Fixed known issue: BSW event status bit WIR will be kept on if set and reset in the same power cycle
- ASCDEM-1234 Fixed known issue: ClearDTC and aging do not correctly reset warning indicator-related cycle counters

- Implemented storing the event status block immediately if DemClearDTCBehavior is set to a DEM_CLR-RESP_NONVOLATILE option
- ASCDEM-1422 Fixed known issue: The status of the warning indicator may not be updated if the update of event data APIs are called concurrently (i.e. tasks are interrupting each other)
- Implementation of OBD Permanent Memory support

Module version 5.9.1

2014-04-25

- ASCDEM-1142 Fixed known issue: `Dem_InternalSetEventStatus() != E_OK` in `Dem_Debounce-TimeTimerTick` is wrong and has to be `== DEM_E_DET_REPORT`
- ASCDEM-1135 Fixed known issue: Extended data of a new event is not written in case an event has been displaced before
- Extended the *warning indicator handling* to support that WIR bits can be switched on immediately on `FAILED` reports
- ASCDEM-1167 Fixed known issue: `Dem_GetFreezeFrameDataByDTC`/`Dem_GetEventExtended-DataRecord` does not provide correct data if more than two DIDs/Extended Data records are configured per event
- ASCDEM-1163 Fixed known issue: If indicators are configured, but none is linked to any event, a compiler error occurs
- ASCDEM-1172 Fixed known issue: Constraint definitions for `IndicatorIdType` and `RatioIdType` might be incorrectly generated
- ASCDEM-1153 Fixed known issue: Time-based debouncing gets blocked internally if event qualification is locked due to an enable condition
- Corrected several warnings during importing the file `Dem_Bswmd.arxml` into the system model
- ASCDEM-1202 Fixed known issue: Event-related data of BSW events might get updated even if the reported event status does not change
- ASCDEM-1224 Fixed known issue: For *warning indicator handling* the Dem might ignore cycles with *PASSED* results but count untested cycles for counting failure cycles

Module version 5.9.0

2013-11-21

- Extended *event displacement* not ignoring the occurance order even if all events have neither extended data nor freeze frames configured

- Reworked memory sections for internal variables used in configurations with *immediate storage* feature enabled
- Added non-volatile storage support for operation cycle states and FDCs for counter-based debounced events
- Implemented support for `ClearDTC` behavior configuration
- Corrected several warnings during importing the file `Dem_Bswmd.arxml` into the system model

Module version 5.8.0

2013-10-11

- ASCDEM-1019 Fixed known issue: `DemCallbackEventDataChanged` is incorrectly triggered on aging for events with any internal data element except aging counter configured to extended data
- ASCDEM-1028 Fixed known issue: Undefined identifier errors are reported for certain Dem configurations with Rte usage enabled
- Implemented HandleId Wizard support for calculation of `DemEventId`, `DemEnableConditionId` and `DemIndicatorID`
- Updated default value of configuration parameter `DemDebounceAlgorithmClass` to `DemDebounceMonitorInternal`
- Added warning message to `DemExtendedDataRecordClass` if the user maps any internal data elements to an ED segment where `DemExtendedDataRecordUpdate` parameter is set to false
- Implemented support for *Event Displacement*
- Updated NvM service needs calculation - disabled permanent RAM block CRC calculation and removed static block length
- Updated `DTCRecordUpdate` functionality according to AUTOSAR 4.0 Rev 3
- ASCDEM-1091 Fixed known issue: Configuration parameter `DemImmediateNvStorageLimit` set to 255 does not limit the write to NV memory
- Updated `ResetEventStatus` functionality from AUTOSAR 4.0 Rev 2 to AUTOSAR 4.0 Rev 3
- Added support for function tracing via AUTOSAR Debugging
- Updated clearing of event memory on aging for an event with DTC record update disabled

Module version 5.6.0

2013-06-14

- ASCDEM-976 Fixed known issue: Implementation Data Types of category `TYPE_REFERENCE` refer a `DataConstr` when it is not allowed

- Added internal Dcm/Dem support functions to save the event ID of the last requested DTC for subsequent use (run-time optimization)
- Added support for NvM single block callback in Service Needs Assistant
- ASCDEM-959 Fixed known issue: Events without DTC-class are incorrectly assigned to DTC-group `DEM_DTC_GROUP_EMISSION_REL_DTCS`
- ASCDEM-991 Fixed known issue: If *immediate storage* is enabled and no freeze frame classes are configured, unresolved symbol error occurs
- Updated *immediate storage* to support immediate deletion of event memory entry from NVRAM on *event displacement*
- Updated handling of valid, but non-stored records by `Dem_GetEventFreezeFrameData` and `Dem_GetEventExtendedDataRecord`
- Updated return values of `Dem_GetFreezeFrameDataByDTC()` in case of valid but not stored entries to match requirement Dem630
- Implemented support for *fault confirmation*
- Implemented support for *Event healing - Warning indicator handling*
- Added support for dynamic DTC values as part of *side allocation* feature

Module version 5.5.0

2013-02-08

- Updated support for *BSW error-queue handling*
- Updated implementation of `Dem_GetSizeOfExtendedDataRecordByDTC()` based on configuration parameter `DemGetSizeOfExtendedDataRecordByDTCOptimization`, considering both AUTOSAR 4.0 Rev 3 and Bugzilla issue http://www.autosar.org/bugzilla/show_bug.cgi?id=52426
- Implemented support for *configured freeze frame record numbers*

Module version 5.4.0

2012-10-12

- ASCDEM-882 Fixed known issue: Dem does not compile when legacy symbolic names are disabled via macro `DEM_DONT_PROVIDE_LEGACY_SYMBOLIC_NAMES`
- ASCDEM-885 Fixed known issue: Aging related internal function is called but not defined in some configurations and aging disabled
- ASCDEM-898 Fixed known issue: `Dem_SetAgingCycleState()` will not be available if more than one aging cycle is configured

- ASCDEM-899 Fixed known issue: If an event configures only one freeze frame record slot, this record will be updated if the event re-occurs
- Implemented support for `DemImmediateNvStorageLimit`
- Updated configuration and API to AUTOSAR 4.0.3:
  - Added the following configuration parameters: `DemAgingCycle`, `DemStatusBitHandlingTestFailedSinceLastClear`, `DemObdDTC`, `DemUdsDTC`, `DemSecondaryFunctionIdRef`
  - Updated the following configuration parameters: `DemExtendedDataCapture`, `DemFreezeFrameCapture`, `DemAgingCycleRef`
  - Removed the following configuration parameters: `DemDTC`, `DemDTCKind`
  - Renamed the DET error code for invalid address from `DEM_PARAM_ADDRESS` to `DEM_E_PARAM_POINTER`
  - Adapted the following types: `Dem_ReturnGetStatusOfDTCType`, `Dem_ReturnGetNextFilteredDTCType`, `Dem_ReturnClearDTCType`, `Dem_ReturnGetFreezeFrameDataByRecordType`, `Dem_ReturnGetExtendedDataRecordByDTCType`, `Dem_ReturnGetDTCByOccurrenceTimeType`, `Dem_ReturnGetFreezeFrameDataByDTCType`, `Dem_ReturnGetSizeOfExtendedDataRecordByDTCType`, `Dem_ReturnGetSizeOfFreezeFrameByDTCType`, `Dem_ReturnGetSeverityOfDTCType`
  - Added the following types: `Dem_ReturnDisableDTCRecordUpdateType` and `Dem_DTCFormatType`
  - Renamed the return value from `E_NO_DTC_AVAILABLE` to `DEM_E_NO_DTC_AVAILABLE`
  - Updated the argument from `DTCKind` to `DTCFormat` in the following APIs: `Dem_GetDTCOfEvent()`, `Dem_ClearDTC()`
  - Removed the argument `BuffSize` from the following APIs: `Dem_GetEventExtendedDataRecord()`, `Dem_GetEventFreezeFrameData()`
  - Added the argument `DTCFormat` in the following APIs: `Dem_SetDTCSuppression()`, `Dem_SetDTCFilter()` (retained kind), `Dem_SetFreezeFrameRecordFilter()`
  - Removed the argument `DTCKind` from the following APIs: `Dem_GetFreezeFrameDataByRecord()`, `Dem_GetFreezeFrameDataByDTC()`, `Dem_GetSizeOfFreezeFrameByDTC()`, `Dem_GetExtendedDataRecordByDTC()`, `Dem_GetSizeOfExtendedDataRecordByDTC()`
  - In the port `GetEventExtendedDataRecord` the output parameter name changed from `ExtData` to `DestBuffer`
  - In the port `GetEventFreezeFrameData` the following parameter names changed: `ReportFullRecord` to `ReportTotalRecord`, `DID` to `DataId`, `FFData` to `DestBuffer`
  - Renamed the `PortArgument` from `OverflowIndPersistentMemory` to `OverflowIndPermanentMemory`
  - Updated the following Operations/Dataelements: `GetDTCOfEvent`, `ClearDTC`: updated the parameter from `DTCKind` to `DTCFormat`. `DTCSuppression`: added the parameter `DTCFormat`

- The top-level structure of the software-component description in the ARXML files changed from `/AUTOSAR/Dem` to `/AUTOSAR_Dem`

Module version 5.3.0

2012-06-25

- Added configuration parameter `DemDcmUsage` which enables or disables the usage of Dcm by Dem
- Introduced Unreachable code assert and Precondition assert using `DemDefensiveProgramming` container
- ASCDEM-862 Fixed known issue: `SwcBswMappings` is located at the wrong location in BSWMD
- Implemented support for Reset PDTC bit as per upcoming ISO 14229-1 and Bugzilla issue http://www.autosar.org/bugzilla/show_bug.cgi?id=45036
- Updated counter based debouncing jump behavior (as per AUTOSAR 4.1 Rev 1 and Bugzilla issue http://www.autosar.org/bugzilla/show_bug.cgi?id=52969)
- ASCDEM-883 Fixed known issue: Definition of Implementation Data Type `Dem_DTCOriginType` is defined in `Dem_swc_internal.arxml` only

Module version 5.2.0

2012-03-23

- ASCDEM-692 Fixed known issue: `DATA-CONSTR-REF` at wrong position in arxml-files of Dem
- ASCDEM-694 Fixed known issue: Recommended configuration schema cannot be found while creating Dem project in EB tresos Studio
- ASCDEM-695 Fixed known issue: `Dem_GetSizeOfFreezeFrameByDTC()` uses old return-type
- ASCDEM-697 Fixed known issue: `Dem_MaxDataValueType` type definition missing in SW-CD
- ASCDEM-696 Fixed known issue: Missing calls to `GeneralCBStatusEvt` port
- Introduced BSW-event debouncing during pre-initialized phase (between `Dem_PreInit()` and `Dem_Init()`) in `Dem_ReportErrorStatus()`
- Implemented support for *Clear event allowed* callbacks (via RTE and C-function)
- Updated naming scheme for macros of symbolic name values to AUTOSAR 4.0 Rev 3 naming scheme
- Implemented support for *Event memory overflow indication*
- Implemented support for *Occurrence counter* and internal data element `DEM_OCCCTR`
- Updated Dem-Dcm interaction to AUTOSAR R4.0:
  - Adapted `Dem_GetNextFilteredRecord()` by returning relative addressed FF record.

- Implemented support for *Functional unit* via API `Dem_GetFunctionalUnitOfDTC()`.
- Removed record number from the output data of `DestBuffer` in `Dem_GetExtendedDataRecordByDTC()`
- Updated output data of `SizeOfExtendedDataRecord` to include size of extended record number(s) in `Dem_GetSizeOfExtendedDataRecordByDTC()`
- Implemented support for `ExtendedDataNumber` 0xFE via API `Dem_GetSizeOfExtendedDataRecordByDTC()`
- Introduced check for `DTCSettingAllowed` in `Dem_ReportErrorStatus()` and `Dem_SetEventStatus()`

- Updated naming of `Fim_DemTriggerOnEventStatus()` to `FiM_DemTriggerOnEventStatus()` (prefix changed) as per AUTOSAR 4.0 Rev 3
- Implemented support for *Immediate storage*
- ASCDEM-832 Fixed known issue: `DTCStatusChanged` and `Dem_DcmTriggerOnEventStatus()` callbacks are not triggered for some configurations

Module version 5.1.0

2011-09-07

- Implemented support for internal data element `DEM_AGINGCTR` and updated aging processing according to AUTOSAR R4.0
- Implemented support for separate aging counters via API `Dem_SetAgingCycleState()`
- Initial AUTOSAR R4.0 version

2.2. New features

- No new features have been added since last release.

2.3. Elektrobit-specific enhancements

This chapter lists the enhancements provided by the module.

- Extended buffer mechanism for the API `Dem_ReportErrorStatus()`

  Description:

The buffer mechanism for the API `Dem_ReportErrorStatus()` (configured via `DemBswErrorBufferSize`) is extended to the ECU sleep and wake-up phases (i.e. after `Dem_Shutdown()`).

- Dependency on Dcm can be switched off

  Description:

  The inclusion of `Dcm_Types.h`, which defines `Dcm_OpStatusType` for the Dcm function calls `Dem_DcmGetInfoTypeValue08` and `Dem_DcmGetInfoTypeValue0B`, can be controlled via the configuration parameter `DemDcmUsage`. This allows a simplified integration of the Dem module and ECUs without a Dcm module.

- Optimized resource usage according to HIS recommendations

  Description:

  The Dem generates ports and interface functions only for those events, which actually need them by implementing the HIS requirements HisDem0001, HisDem0002, HisDem0003, HisDem0004, HisDem0005, HisDem0006, HisDem0007, HisDem0008, HisDem0009.

- Dem behavior when occurrence counter exceeds `DemImmediateNvStorageLimit`

  Description:

  Following modifications are adapted:

  - Dem module continues to update the Dem internal event memory as against Dem552. However, it will not trigger the storage in non-volatile memory when the occurrence counter has exceeded the value defined by the configuration parameter `DemImmediateNvStorageLimit`.
  - If the occurrence counter of an event (with `DemImmediateNvStorage` enabled) exceeds `DemImmediateNvStorageLimit`, the event memory entry and its event related data will be stored persistently during the shutdown phase.

- Updated return values of APIs `Dem_GetEventFreezeFrameData` and `Dem_GetEventExtendedDataRecord`

  Description:

  The return values of the APIs `Dem_GetEventFreezeFrameData` and `Dem_GetEventExtendedDataRecord` are updated to `E_NOT_OK` for valid, but non-stored FF records and ED records.

- Support for *side allocation*

  Description:

  Statically configured DTCs of events can be changed by application components during runtime. The Dem supports configuring a callout function to be called every time the Dem needs to retrieve a DTC for an event.

- Function tracing support via AUTOSAR Debugging

Description:

The module Dem supports tracing of function entry and exit via the EB Dbg module.

Function tracing records following parameters for each function:

- function name
- values of the function arguments
- point in time of function invocation
- point in time of function termination
- return value of the function

- Support for non-volatile storage of FDCs

Description:

FDC is stored non-volatile for events with counter based debouncing if `DemDebounceCounterStorage` is enabled.

- Support for frequency-based debouncing of diagnostic events

Description:

The frequency-based debouncing algorithm feature from R3.1 (Dem SWS chapter 7.2.11.3.3) is still supported and it can be configured by setting container `DemDebounceAlgorithmClass` to `DemDebounceFrequencyBased`.

- Enhanced multiplicity of `DemFreezeFrameClass` and `DemDidClass` and `DemDataElementClass`

Description:

The upper multiplicity of `DemFreezeFrameClass` and `DemDidClass` and `DemDataElementClass` configuration container has been extended from 255 to 65535 according to RFC 61653 and RFC 55110. This provides the possibility to configure ECUs, which can contain several thousand of different FF class, DIDs (and data elements), e.g., in the powertrain domain.

- Support for configurable displacement strategy according to R4.2

Description:

The displacement strategy can be selected via the R4.2 configuration parameter `DemEventDisplacementStrategy`, i.e., the consideration of the active/passive status can be disabled by choosing the configuration option `DEM_DISPLACEMENT_PRIO_OCC`.

- Support for additional specific features (ASCPD-225, ASCPD-226, ASCPD-227)

Description:

New parameters, counters and status bits are added or extended to allow processing events also on "unconfirmed" threshold during debouncing, which can be lower than the failed threshold:

- Added configuration parameter `DemDebounceCounterUnconfirmedThreshold`. This is the unconfirmed threshold on which the event is processed.
- Extended configuration parameters `DemExtendedDataCapture` and `DemFreezeFrameCapture` by the option `DEM_TRIGGER_ON_FDC_THRESHOLD` to allow processing on unconfirmed threshold
- Added new internal data elements, which can be mapped to extended data records:

Data element	Description
`DEM_CURRENT_FDC`	Fault detection counter
`DEM_FDC12`	Maximum fault detection counter since last clear
`DEM_OCC1`	Operation cycles since last unconfirmedDTC counter
`DEM_OCC2`	unconfirmedDTC aging counter
`DEM_OCC3`	Operation cycles since first unconfirmedDTC counter
`DEM_OCC4`	unconfirmedDTC operation cycle counter
`DEM_OCC5`	Number of warm-up cycles since the DTC commanded the MIL to switch off
`DEM_OCC6`	Consecutive failed operation cycle counter
`DEM_OCC7`	Qualified/unconfirmedDTC operation cycle counter
`DEM_SI30`	DTC status indicators

- Extended displacement mechanism to be configurable based on unconfirmedDTC aging counter (OCC2).
- Extended healing and aging algorithms to be configurable based on unconfirmedDTC aging counter (OCC2).

- Support of additional enumerations in DemEventOBDReadinessGroup

  Description:

  In the configuration parameter `DemEventOBDReadinessGroup` the additional enumeration value `DEM_OBD_RDY_FLSYS_NONCONT` is supported according to AUTOSAR R4.2.1.

- Support for operation cycle automatic end feature for all operation cycle types

  Description:

AUTOSAR defines the automatic end of operation cycles feature only for the OBD driving cycle in SWS_Dem_00697, ECUC_Dem_00837. Since the feature of operation cycles automatic end may also be useful for other operation cycles than OBD driving cycle, the Dem supports this for every configured operation cycle which has the configuration parameter `DemOperationCycleAutomaticEnd` configured to `TRUE`. Every operation cycle, that has `DemOperationCycleAutomaticEnd` configured to `TRUE` is ended automatically by Dem during `Dem_Shutdown()`.

- Partial support of AUTOSAR R4.2.1 RTE interfaces

  Description:

  The following AUTOSAR R4.2.1 RTE interfaces are supported:

  - `DiagnosticInfo:GetEventFreezeFrameData`
  - `DiagnosticInfo:GetDTCOfEvent`
  - `CallbackEventDataChanged`
  - `GeneralCallbackEventDataChanged`
  - `CallbackEventStatusChange`
  - `GeneralCallbackEventStatusChange`
  - `DataServices`

- Support for advanced event displacement

  Description:

  The advanced event displacement algorithm uses the static and dynamic event priority. Static event priority is used for displacement configured by `DemEventPriority`. Dynamic event priority is calculated in consideration of the extended data items `SI30 (Bit4, 5, 6)`, `OCC1 (DemAdvDisplacementOcc1Limit)` and the configured passive table. The passive table consists of passive table lines holding references to reported events `DemReportedEventRef` and references to passive events `DemPassiveEventRef`. Every passive table line holds one reported event and several passive events, which can be displaced by that reported event under certain conditions.

  The following configuration parameters were introduced:

Parameter	Description
`DemAdvDisplacementOcc1Limit`	The number of OCC1 counts used for selecting entry by the advanced displacement algorithm.
`DemAdvDisplacementPassiveTableLine`	Describes a line in the passive table used in the advanced displacement algorithm.

Parameter	Description
DemReportedEventRef	Reference to a DemEventParameter corresponding to the event currently reported and trying to get an event memory entry.
DemPassiveEventRefs	List of references to DemEventParameter corresponding to an event currently stored in an event memory entry and subject for displacement.
DemPassiveEventRef	Reference to a DemEventParameter corresponding to the event currently stored in an event memory entry and subject for displacement.

The advanced event displacement is enabled under the following conditions:

- DemEventDisplacementSupport is set to TRUE
- DemEventDisplacementStrategy is set to DEM_DISPLACEMENT_FULL
- DemExtendedDataCapture is set to DEM_TRIGGER_ON_FDC_THRESHOLD
- Internal data elements DEM_SI30, DEM_OCC1, DEM_OCC2, DEM_CURRENT_FDC must be configured.

- Support for filtering of stored DTCs via Dcm

  Description:

  The Dem module is extended with the following vendor specific API functions:

  - Dem_DcmSetStoredDTCFilter()
  - Dem_DcmGetNextFilteredStoredDTC()

  These new API functions are also based on the SetDTCFilter concept of AUTOSAR.

- Support additional OCC2 and OCC5 based event aging

  Description:

  The Dem module was extended by an algorithm for event aging. This algorithm takes the OCC2 counter for aging of UDS events and the OCC5 counter for aging of OBD events into account.

  The Dem module is extended with the following vendor specific API configuration parameter:

  - DemOBDAgingCycleCounterThreshold

- Event Burst Reduction/Processor Load Reduction

  Description:

  To prevent an excessive workload of the Dem_MainFunction the user can specify a maximum count of failed and passed error queue entries to be processed each schedule of the Dem_MainFunction.

For that the Dem module is extended with the following vendor specific configuration parameters:

- `DemMaxHandledPassedEventsPerScheduling`
- `DemMaxHandledFailedEventsPerScheduling`

- Support for development freeze frames

Description:

Development freeze frames are additional freeze frames which are fetched together with the first freeze frame record. The development freeze frames can be requested with a specific record number by the Dcm. The Dem provides two kinds of development freeze frames: A and B. For each kind the number of events which can store a development freeze frame can be configured to save resources. If no space is available, development freeze frames are displaced based on the same criteria as regular event memory entries. The development freeze frame A provides a common freeze frame part. This part is equal in all development freeze frame A.

The development freeze frames are not supported if event combination is enabled. In addition it is required that event displacement support is enabled.

The development freeze frames can only be used for the events with primary memory destination. Immediate storage functionality does not affect the development freeze frames, i.e. they are never stored immediately.

The development freeze frames are not prestored, i.e., they are not supported by the API `Dem_PrestoreFreezeFrame()`, and they are only accessible to Dcm, i.e., they cannot be read by the API `Dem_GetEventFreezeFrameData()`.

The Dem module is extended with the following vendor specific configuration parameters:

- `DemDevFreezeFrameSupport`
- `DemCommonFFDataDevAClassRef`
- `DemMaxNumberDevFreezeFrameAEntry`
- `DemMaxSizeDevFreezeFrameAEntry`
- `DemMaxNumberDevFreezeFrameBEntry`
- `DemMaxSizeDevFreezeFrameBEntry`
- `DemDevFreezeFrameAClassRef`
- `DemDevFreezeFrameARecNum`
- `DemDevFreezeFrameBClassRef`
- `DemDevFreezeFrameBRecNum`

- Support for common freeze frame

Description:

The common freeze frame is an additional freeze frame which is captured with the first freeze frame record configured for a specific event. The content of a standard freeze frame (common + individual part of freeze frame) can be requested by standard API `Dem_GetNextFreezeFrameData()` where the common part is appended to the individual part (event specific) of freeze frame. If the configured event does not reference to a freeze frame class (no individual freeze frame) only the common freeze frame will be stored.

The Dem module is extended with the following vendor specific configuration parameters:

- `DemCommonFFDataClassRef`

- Calibration of Common and Development Freeze Frames

Description:

Support for calibration of enabling, disabling, and modification of the: regular Common Freeze Frame class, individual Development Freeze Frame A class, Common Development Freeze Frame A class, and individual Development Freeze Frame B class.

- Implemented support for DTC disabling via calibration

Description:

The Dem module was extended with support to completely disable a DTC by calibration (both storage and visibility towards tester). Calibration is done via separate bit-array which has one bit for each configured event. This feature is only available when the configuration parameter `DemCalibrationSupport` is configured to `TRUE`.

- Implemented support to also clear disabled DTCs

Description:

The Dem module was extended with support to also clear disabled DTCs in the API request Dem_ClearDTC, if all DTCs are requested per DTC group. This feature can be selected with the configuration parameter `DemClearDTCOfDisabledEvents`, with value `TRUE`.

- Support of AUTOSAR R4.3.0 Dem_SetDTCSuppression() API

Description:

The Dem module was extended with support to dynamically change the DTC suppression status via `Dem_SetDTCSuppression()` API. A suppressed DTC is not visible for an external tester via any service request but does not stop the event processing of the corresponding event. This feature can be enabled with configuration parameter `DemSuppressionSupport`.

- Support of AUTOSAR R4.3.0 Dem_SetEventAvailable() API

Description:

The Dem module was extended with support to dynamically change the event availability via `Dem_SetEventAvailable()` API. If an event is set to unavailable, it is treated as if it is not configured in the system. This feature can be enabled with configuration parameter `DemAvailabilitySupport`.

- Implemented support for always available Extended Data Records

  Description:

  The Dem module configuration has a new parameter allowing the user to configure an Extended Data Record to be always available, regardless of the existence of an event memory entry.

  This feature can be used for all extended data records which consist of data elements for which the Dem module can always provide or calculate the value.

- Support of common operation cycle

  Description:

  The common operation cycle provides the ability to configure one operation cycle (per event) that serves as the following event specific operation cycle references.

  - `DemAgingCycleRef`
  - `DemEventFailureCycleRef`
  - `DemIndicatorFailureCycleRef`
  - `DemIndicatorHealingCycleRef`

  If the common operation cycle support is enabled, these different operation cycle references are referencing the common operation cycle reference `DemOperationCycleRef`. This approach makes it easier to configure and calibrate the different operation cycles because only one operation cycle reference has to be configured respectively calibrated.

  The Dem module is extended with the following vendor specific configuration parameters:

  - `DemCommonOperationCycleSupport`

- Support of different occurrence order (OccOrder) processing

  Description:

  The Dem supports an additional behavior for the processing of the occurrence order (OccOrder) of an event.

  Under consideration of the configuration parameter `DemUpdateOccOrderOnEventEntryUpdate` the `OccOrder` is processed in the following ways:

  - Parameter `DemUpdateOccOrderOnEventEntryUpdate` is set to `TRUE`: Update by highest value + 1 when a new entry is added into event memory or an entry is updated due to an event failure report.

- Parameter: `DemUpdateOccOrderOnEventEntryUpdate` is set to `TRUE`: Update by highest value + 1 only when a new entry is added into event memory for the first time.

The Dem module is extended with the following vendor specific configuration parameter:

- `DemUpdateOccOrderOnEventEntryUpdate`. The default value is `TRUE`.

- Event storage handling for DTC value zero

  Description:

  An event with the configured/calibrated DTC value 0 will not be stored or updated in event memory (i.e. no entry, no freeze frame data, no extended data) dependent on the value of the EB-specific configuration parameter `DemStoreInternalEvents`.

- Record numbers without Freeze Frame Class

  Description:

  Support of non-storable freeze frame records, i.e., valid freeze frame record numbers without freeze frame data/class. This results into positive response (without freeze frame data) in diagnostic service (e.g. 0x19 04) in case of requesting a valid record number without freeze frame class.

- Read number of event memory entries

  Description:

  Support of the AUTOSAR R4.2.1 API `Dem_GetNumberOfEventMemoryEntries()` and RTE interface `EvMemOverflowIndication` extension with operation `GetNumberOfEventMemoryEntries` to return the number of event memory entries currently stored in the event memory.

- Extended behavior of clearing OBD freeze frames

  Description:

  OBD freeze frames are cleared under the following conditions:

  - The event is displaced by an event with higher priority.
  - Aging conditions are fulfilled, and the event becomes aged.
  - Pending DTC status is cleared (PDTC status bit changes 1 -> 0) before the DTC was confirmed.

- Internal data element event priority

  Description:

  The event priority can be used as internal data element and mapped to extended data.

- Internal data element event id

  Description:

The event id can be used as internal data element and mapped to extended data.

- Internal data element Cycles since last failed

  Description:

  An internal data element DEM_CYCLES_SINCE_LAST_FAILED can be mapped to extended data. The implementation is according to AUTOSAR R4.1/R4.2/R4.3 specification.

- Internal data element Cycles since first failed

  Description:

  An internal data element DEM_CYCLES_SINCE_FIRST_FAILED can be mapped to extended data. The implementation is according to AUTOSAR R4.1/R4.2/R4.3 specification.

- Monitor re-initialization with `InitMonitorReason DEM_INIT_MONITOR_REENABLED`

  Description:

  If the configuration parameter `DemGeneral/DemCallbackInitMForEReenabledSupport` is set to 'true' Dem offers support for callback `DemInitMonitorForEvent` with `InitMonitorReason DEM_INIT_MONITOR_REENABLED` as described in AUTOSAR R4.2.2 specification. Range of `Dem_InitMonitorReasonType` is extended with `DEM_INIT_MONITOR_REENABLED` and `DEM_INIT_MONITOR_STORAGE_REENABLED`.

- Dem_ClearDTC burst reduction

  Description:

  Dem module was extended with support to asynchronous processing of `Dem_ClearDTC()` API requests from the `Dem_MainFunction()` context. The maximum number of processed events per each `Dem_MainFunction()` call can be adjusted with the configuration parameter `DemMaxNumberClearEventsPerCycle`.

- Support for configurable queue sizes generated by RTE for Dem

  Description:

  To prevent an overflow of the RTE queue for Dem CS operation calls, the Dem module was extended with a configurable queue size for all RTE operations. New configuration parameters have been introduced for setting each queue size value per operation.

- Reset/freeze event debounce status for frequency-based debouncing via Dem_ResetEventDebounceStatus() API

  Description:

If an event is configured with frequency-based debouncing, `Dem_ResetEventDebounceStatus()` API can be used in order to control the internal debounce frequency timer via the following behaviors: FREEZE and RESET.

Since the current AUTOSAR specification (R4.0.3) does not describe the frequency-based event debouncing, internal EB requirements are defined in order to specify the functionality for the reset/freeze behavior for event debouncing via `Dem_ResetEventDebounceStatus()`. Frequency-based event debouncing is a type of debouncing that is supplier-specific, ported from R3.1.

- Reset/freeze event debounce status for frequency-based debouncing via non-fulfilled enable conditions and disabled ControlDTCSettings

  Description:

  If an event is configured with frequency-based debouncing, the internal debounce frequency timer can be freezed/reset/freeze via non-fulfilled enable conditions and disabled ControlDTCSetting, depending on configuration parameter `DemDebounceBehavior`.

  Since current AUTOSAR specifications (ASR4.0.3) do not describe the frequency-based event debouncing (this type of debouncing is supplier-specific, ported from R3.1), internal EB requirements are defined in order to specify the functionality for the reset/freeze behavior for event debouncing via non-fulfilled enable conditions and disabled ControlDTCSetting.

- Support project specific API `Dem_ReportErrorStatus()`

  Description:

  If all Dem events are configured as SWC, Dem will not provide the declaration and definition of the API `Dem_ReportErrorStatus()`. Support project specific API `Dem_ReportErrorStatus()` implementation through inclusion of separate module external header-files into provided Dem template header file. Inclusion can be controlled via the configuration parameter `DemIncludeProjectSpecificCustomizationSupport`.

- Implemented support for DTC DataService interface with EventID

  Description:

  The Dem module was extended with support for extended Callback interface for CS External Data Elements. If the configuration parameter `DemDataElementInterfaceWithEventParameter` of the Data Element is enabled, the prototype of the Callback will be extended with another parameter, EventID.

- Support for event displacement according to ASR4.3.1

  Description:

  The event displacement algorithm is extended by the features:

- protection of memory entries of emission related events to be displaced by new reported events with the same priority and
- displacement of memory entries of events which are not tested the current operation cycle.

New configuration parameters:

Parameter	Description
`DemOBDEventDisplacement`	Enables/disables a different displacement behavior for OBD events. If enabled, an OBD event that triggers the MIL indicator or holds the OBD Freeze Frame or has the Pending-DTC status bit set will not be displaced by a new event with the same priority.
`DemEventDisplacementCriterionTNCTOC`	Enables/disables the consideration of the TNCTOC event status bit. If enabled, event entries with TNCTOC bit == 1 will be considered for displacing in accordance with ASR4.3.1. If disabled, the TNCTOC bit will not be considered (ASR4.0.3 compliant behavior).

- Support for disabling the trigger for status changed notification callbacks on start of operation cycle

  Description:

  To reduce execution time of the `Dem_SetOperationCycleState()` with state `DEM_CYCLE_STATE_START` the user can disable the trigger point for EventStatusChanged/DTCStatusChanged callbacks. Such an option is of interest, for projects where the application does not need to know the evolution of the DTC status for events over operation cycle start/re-start. This feature can be controlled via the configuration parameter `DemCallbackEventStatusChangedOpCycStartSupport`.

- Optimized calibratable data

  Description:

  A2L files should be regenerated regardless of whether configuration has changed or not.

- Support for parallel processing of multiple diagnostic requests

  Description:

  The EB Dem solution provides the possibility of defining multiple diagnostic clients. A diagnostic client in the EB Dem solution is defined as a BSW (e.g. Dcm) module that accesses the Dem module in order to delegate diagnostic services, which are related to the event memory. There are multiple clients that can request access to the fault memory. A ClientId defines a unique identifier for a Dem client. This number is used by a client in the ClientId parameter in all APIs used for diagnostic services. With this new feature,

the parallel processing on the same event memory by multiple clients (BSW/SW-C/CDD) is enabled and DTC selection process for operations used to retrieve event data, clear DTC, DTC status is optimized.

In particular, the signatures of the following Dem APIs, which are used by the Dcm module to process the OBD and UDS diagnostic requests, are implemented according to AUTOSAR R4.3.1 and support ClientId as a parameter:

- `Dem_ClearDTC()`
- `Dem_GetDTCSelectionResult()`
- `Dem_GetDTCSelectionResultForClearDTC()`
- `Dem_SelectDTC()`
- `Dem_GetTranslationType()`
- `Dem_GetDTCStatusAvailabilityMask()`
- `Dem_GetStatusOfDTC()`
- `Dem_GetSeverityOfDTC()`
- `Dem_GetFunctionalUnitOfDTC()`
- `Dem_SetDTCFilter()`
- `Dem_GetNumberOfFilteredDTC()`
- `Dem_GetNextFilteredDTC()`
- `Dem_GetNextFilteredDTCAndFDC()`
- `Dem_GetNextFilteredDTCAndSeverity()`
- `Dem_SetFreezeFrameRecordFilter()`
- `Dem_GetNextFilteredRecord()`
- `Dem_GetDTCByOccurrenceTime()`
- `Dem_DisableDTCRecordUpdate()`
- `Dem_EnableDTCRecordUpdate()`
- `Dem_GetSizeOfExtendedDataRecordSelection()`
- `Dem_GetSizeOfFreezeFrameSelection()`
- `Dem_GetNextExtendedDataRecord()`
- `Dem_GetNextFreezeFrameData()`
- `Dem_SelectExtendedDataRecord()`
- `Dem_SelectFreezeFrameData()`
- `Dem_DisableDTCSetting()`
- `Dem_EnableDTCSetting()`

- Dem_DcmReadDataOfOBDFreezeFrame()
- Dem_DcmGetDTCOfOBDFreezeFrame()

- Synchronous Dcm interface

  Description:

  Following API functions are implemented in a synchronous way and do not make use of the DEM_PENDING return values: Dem_GetNumberOfFilteredDTC(), Dem_GetNextFilteredDTC(), Dem_GetNextFilteredDTCAndFDC(), Dem_GetNextFilteredDTCAndSeverity(), Dem_GetDTCSelectionResult(), Dem_GetDTCSelectionResultForClearDTC(), Dem_GetStatusOfDTC(), Dem_DisableDTCRecordUpdate(), Dem_EnableDTCRecordUpdate(), Dem_GetSeverityOfDTC(), Dem_GetFunctionalUnitOfDTC(), Dem_GetNextFilteredRecord(), Dem_GetSizeOfExtendedDataRecordSelection(), Dem_GetSizeOfFreezeFrameSelection(), Dem_GetNextExtendedDataRecord(), Dem_GetNextFreezeFrameData(). This allows for a more compact and ROM-saving implementation.

- Synchronous J1939Dcm interface

  Description:

  Following API functions are implemented in a synchronous way and do not make use of the DEM_NUMBER_PENDING respectively DEM_FILTERED_PENDING return values: Dem_J1939DcmGetNumberOfFilteredDTC(), Dem_J1939DcmGetNextFilteredDTC(), Dem_J1939DcmGetNextFreezeFrame(). This allows for a more compact and ROM-saving implementation.

- Support of uint16 type for Dem_DTCOriginType

  Description:

  The ClearDTC operation of C/S-interface CddIf and APIs Dem_GetNumberOfEventMemoryEntries() are implemented according to AUTOSAR R4.0/R4.2, with the exception of the DTCOrigin parameter type. The Dem_DTCOriginType has been adapted from uint8 to unit16 type.

- Support of user-defined memory identifier for accessing the secondary event memory

  Description:

  The Dem module was extended with support to select the secondary event memory with a configurable user-defined memory identifier.

  This feature can be enabled with configuration parameter DemUserDefMemoryId.

  If configured, and the DTCOrigin value of a service request matches with DemUserDefMemoryId + 0x100, the DTCOrigin value of the request is mapped to DEM_DTC_ORIGIN_SECONDARY_MEMORY.

This feature cannot be used for operations on C/S interfaces, where `DTCOrigin` is used as a port defined argument (operations `GetEventMemoryOverflow` and `GetNumberOfEventMemoryEntries` of C/S interface `EvMemOverflowIndication`).

- Configuration of the test failed status bit storage for individual events

  Description:

  The Dem module was extended with an optional configuration parameter `DemStatusBitStorageTestFailedPerEvent` to provide a more granular control of the test failed status bit storage.

  This feature provides the capability to configure `DemStatusBitStorageTestFailed` per event. If configured, the value of `DemStatusBitStorageTestFailedPerEvent` is preferred over the general `DemStatusBitStorageTestFailed` to determine whether the TestFailed status for an event is stored volatile or non-volatile.

- Priority of indicator status applied by the API `Dem_GetIndicatorStatus()`

  Description:

  If an indicator is activated in multiple modes (e.g. OBD and J1939) at the same time, the API `Dem_GetIndicatorStatus()` returns the status considering the following priority:

  - DEM_INDICATOR_FAST_FLASH (highest priority)
  - DEM_INDICATOR_SLOW_FLASH
  - DEM_INDICATOR_BLINK_CONT
  - DEM_INDICATOR_BLINKING
  - DEM_INDICATOR_CONTINUOUS
  - DEM_INDICATOR_OFF (lowest priority)

- Support for intermediate DTC status storage

  Description:

  To preserve the result of the last diagnosis in case of a power failure or hard reset, the Dem module was extended with support for the intermediate DTC status storage.

  If configuration parameter `DemIntermediateNvStorageOfDTCStatus` is enabled, the Dem stores the DTC status in the NvRAM whenever the immediate storage of the event related data is triggered, in addition to the usual storage at shutdown.

- Partial support of AUTOSAR R20-11 RTE interfaces/operations.

  Description:

  The Dem module was extended to support the following AUTOSAR R20-11 RTE interfaces/operations:

- Interface CallbackEventUdsStatusChanged with the associated operation CallbackEventUdsStatusChanged.
- Interface GeneralDiagnosticInfo with the associated operation GetEventUdsStatus.
- Interface GeneralCallbackEventUdsStatusChanged with the associated operation GeneralCallbackEventUdsStatusChanged.
- Interface CallbackDTCStatusChange with the associated operation DTCStatusChanged.

These RTE interfaces/operations are available when the parameter `DemDefaultASRServiceAPI` is configured with value 'AUTOSAR_2011'.

- Support for Gasoline Particulate Filter readiness group in PID$01 and PID$41

  Description:

  PID$01 and PID$41 are extended to support the Gasoline Particulate Filter (GPF) readiness group. Accordingly, the configuration parameter `DemEventOBDReadinessGroup` is extended with the enumeration value `DEM_OBD_RDY_GPF` for spark ignition vehicles. The support/enable and completion status of the GPF group is shown in bit 4 of Byte C and Byte D of the PIDs, respectively.

- Increased usability of the OBD Readiness Group calibration `Dem_OBDReadinessGroup`

  Description:

  The OBD Readiness Groups can be calibrated with associated events using the calibration `Dem_OBDReadinessGroup` is enhanced by adding further description and renamed to the format [ReadinessGroup]_Slot_[SlotNumber]. The Slot is where events are added or removed from a specific Readiness Group.

  When no event belongs to readiness group `DEM_OBD_RDY_GPF` during configuration and calibration is enabled the readiness group `DEM_OBD_RDY_GPF` is not generated or considered to allow for backward compatibility.

2.4. Deviations

This chapter lists the deviations of the module from the AUTOSAR standard.

- `DemDataElementClass` type `DemExternalSRDataElementClass` is supported according to R4.3.0

  Description:

  The Dem does not support the configuration structure `DemDataElementClass` for type `DemExternalSRDataElementClass` according to AUTOSAR R4.0.3. `DemExternalSRDataElementClass` is implemented according to AUTOSAR R4.3.0. Therefore, R4.0.3 configuration parameter `DemDataElementInstanceRef` was removed. Additionally, an important difference is that the size of data elements

is configured in bytes instead of bits. This schema adaptation does not conform to [ecuc_sws_6007] and [ecuc_sws_6008].

Rationale:

Not fully specified in Dem SWS R4.0.3.

Requirements:

Dem615_Conf, Dem770_Conf, SWS_Dem_00903 (with respect to "unused bits")

- DTC suppression support according to R4.3.0

Description:

The DTC suppression feature (API, C/S interface, configuration) is implemented according to AUTOSAR R4.3.0. Therefore, the configuration parameter that controls the availability of this feature is `DemSuppressionSupport` of type `EcucEnumerationParamDef`, instead of R4.0.3 `DemDTCSuppressionSupport`. This schema adaptation does not conform to [ecuc_sws_6007] and [ecuc_sws_6008].

Requirements:

Dem780_Conf

- The *Event combination* functionality is implemented fully compatible according to AUTOSAR Specification of Diagnostic Event Manager R4.0.3 schema.

Description:

Range of parameter `DemEventCombinationSupport` is kept compatible with AUTOSAR Specification of Diagnostic Event Manager R4.0.3 schema. The following conditions are set by AUTOSAR Specification of Diagnostic Event Manager R4.2.2:

- There is no need for a dedicated EventId, because no combined (parent/master) event will be configured.
- No combined event (parent/master event) is configured by the user, therefore the freeze frame and extended data will be configured commonly for the events involved in combination.
- If a combined DTC is displaced, also UDS status bit 2 (PDTC) and 5 (TFSLC) of the involved events can be reset to 0.

`DemEvtCmbCommonParamMaster` configuration parameter defines the master configuration event for combined DTCs.

Requirements:

Dem024, Dem537, Dem538

- *Event combination* on retrieval is not supported

Description:

Event combination on retrieval (event combination type 2) is not supported.

Requirements:

Dem539, Dem540, Dem541, Dem542

- Limited support for occurrence counter processing

Description:

Occurrence counter triggering after successful fault confirmation (by setting the value `DEM_PROCESS_OCCTR_CDTC` for configuration parameter `DemOccurrenceCounterProcessing`) is not supported.

Requirements:

Dem580, Dem767_Conf

- No Dlt interaction (reference to product description: ASCPD-68)

Description:

The APIs `Dem_DltGetMostRecentFreezeFrameRecordData()`, `Dem_DltGetAllExtendedDataRecords()`, and the notification `Dlt_DemTriggerOnEventStatus()` are not supported.

Requirements:

Dem517, Dem632, Dem633, Dem634, Dem635, Dem636, Dem637, Dem255, Dem718_Conf

- No *Storage conditions* support (reference to product description: ASCPD-36)

Description:

The feature *Storage conditions* is not supported.

Requirements:

Dem451, Dem543, Dem453, Dem455, Dem458, Dem591, Dem459, Dem556, Dem727_Conf, Dem728_Conf, Dem730_Conf, Dem731_Conf, Dem773_Conf, Dem768_Conf, Dem769_Conf

- Extension of `DemFreezeFrameCapture` and `DemExtendedDataCapture` with additional trigger point `DEM_TRIGGER_ON_FDC_THRESHOLD`

Description:

The trigger point `DEM_TRIGGER_ON_FDC_THRESHOLD` (value for configuration parameter `DemFreezeFrameCapture` and `DemExtendedDataCapture`) triggers the collection of freeze frame/extended data for the initial event memory entry. See AUTOSAR Bugzilla RFC http://www.autosar.org/bugzilla/show_bug.cgi?id=50073.

Requirements:

Dem663_Conf, Dem672_Conf

- External aging is not supported (reference to product description: ASCPD-36)

  Description:

  External aging via `Dem_SetAgingCycleCounterValue()` (by setting the value `DEM_PROCESS_AGINGCTR_EXTERN` for configuration parameter `DemAgingCycleCounterProcessing`) is not supported.

  Rationale:

  This feature was removed in AUTOSAR R4.3.0 (refer to Bugzilla RfC #http://www.autosar.org/bugzilla/show_bug.cgi?id=59615).

  Requirements:

  Dem472, Dem644, Dem647, Dem019, Dem488, Dem491, Dem639, Dem640, Dem641, Dem642, Dem555, Dem603_Conf

- Extension of DemInternalDataElement range

  Description:

  `DemInternalDataElement` is extended with new status indicator bits, fault detection counters, operation cycle counters, event priority, aging counter up/down, event id, cycles since last failed, and cycles since first failed.

  Requirements:

  Dem616_Conf, ECUC_Dem_00616

- Multiple event destinations not supported

  Description:

  Each event can only be associated to one event memory.

  The multiplicity of the related container `DemEventDestination` is limited from 0..4 to 1..1.

  Rationale:

  The behavior is undefined as per Dem SWS (see also AUTOSAR Bugzilla RfCs http://www.autosar.org/bugzilla/show_bug.cgi?id=41418 and http://www.autosar.org/bugzilla/show_bug.cgi?id=53633).

  Requirements:

  Dem658_Conf

- No support for *Init monitor for function* callback

Description:

The related parameter `DemCallbackInitMForFFnc` is not used.

Rationale:

The behavior is undefined as per Dem SWS (see also AUTOSAR Bugzilla RfC http://www.autosar.org/bugzilla/show_bug.cgi?id=35637).

Requirements:

Dem335, Dem049, Dem258, Dem614, Dem600_Conf, Dem633_Conf

- Prioritization of `InitMonitorReason` for multiple triggers of `DemInitMonitorForEvent` occurring simultaneously is not implemented.

Description:

The simultaneous occurrence condition is not fully specified and an asynchronous `DemInitMonitorForEvent` callback triggering would be needed to implement the prioritization. This would require an additional queue for storing the events to be processed in the next `Dem_MainFunction()`.

Requirements:

SWS_Dem_01046

- No support for `DemMaxNumberPrestoredFF`

Description:

Freeze frame pre-storage is supported but the optimization regarding the configuration for the maximum number of pre-stored freeze frames is not available.

Requirements:

Dem692_Conf

- Multiplicity of `DemCallbackGetFDC`

Description:

The multiplicity of the configuration container `DemCallbackGetFDC` was extended from 1..1 to 0..1. This schema adaptation does not conform to [ecuc_sws_6008].

Rationale:

The multiplicity of 0 is required to allow monitor-internal debounced events without fault detection counter support, i.e. neither RTE nor C `GetFaultDetectionCounter` callback (see also AUTOSAR Bugzilla RfC http://www.autosar.org/bugzilla/show_bug.cgi?id=44038).

Requirements:

Dem630_Conf

- Initialization check in main function

Description:

If the main function is called while the module is not yet initialized the main function returns immediately without performing any functionality and without raising any Det error. This initialization check is always performed independent of the development error detection setting.

Rationale:

The SchM module may schedule the modules main function before the module is initialized. This would result in lots of Det errors during startup. Therefore the module's main function does not throw a Det error if the module is not yet initialized and simply returns in this case.

Requirements:

Dem124

- Tracing of variables is not supported via AUTOSAR Debugging

Description:

Dem does not provide support for tracing global variables.

Requirements:

Dem519, Dem520, Dem521, Dem522

- External operation cycle counter is not supported

Description:

- Setting of the external operation cycle counter value via API-function `Dem_SetOperationCycleCntValue()` is not supported. It is provided as stub.
- The configuration parameter `DemOperationCycleProcessing` does not support `DEM_PROCESS_OPCYC_COUNTER`.

Rationale:

This API will be removed in AUTOSAR R4.1 (refer to Bugzilla RfC #http://www.autosar.org/bugzilla/show_bug.cgi?id=48567).

Requirements:

Dem485, Dem486, Dem487, Dem019, Dem488, Dem553, Dem783_Conf

- Reset PDTC bit as per ISO 14229-1

  Description:

  The `PendingDTC` bit is reset as specified in the ISO 14229-1 (UDS) standard.

  Rationale:

  The Dem must conform to the ISO 14229-1 standard. This bug is fixed in the AUTOSAR R4.1 (see also AUTOSAR Bugzilla RfC http://www.autosar.org/bugzilla/show_bug.cgi?id=45036).

  Requirements:

  Dem390

- Updated jump behavior to allow jump up and down in place of plain jump

  Description:

  Each reporting of a pre-failed/pre-passed value while the current debounce counter value is smaller/greater than the DemDebounceCounterJumpUpValue/DemDebounceCounterJumpDownValue shall first reset the debounce counter to DemDebounceCounterJumpUpValue/DemDebounceCounterJumpDownValue before performing the pre-failed/pre-passed debounce event.

  Rationale:

  The Dem must conform to the ISO 14229-1 standard. This is clarified in the AUTOSAR R4.1 (see AUTOSAR Bugzilla RfC http://www.autosar.org/bugzilla/show_bug.cgi?id=52969). The jump behavior is implemented to conform to requirements SWS_Dem_00423 and SWS_Dem_00425 from AUTOSAR R4.2.1.

  Requirements:

  Dem423, Dem425

- No support for configuration parameter `DemTriggerMonitorInitBeforeClearOk`

  Description:

  Configuration of monitor re-initialization to be triggered before or after the Dem module returns `DEM_CLEAR_OK` is limited to perform triggering only before returning `DEM_CLEAR_OK`.

  Requirements:

  Dem573, Dem765_Conf

- Development error `DEM_E_PARAM_LENGTH` is not supported

  Description:

  Wrong values of length parameters are handled via the development error `DEM_E_PARAM_DATA`.

Requirements:

Dem173

- Multiplicity of `DemFreezeFrameClass` and `DemDataElementClass`

Description:

The multiplicity of the configuration container `DemFreezeFrameClass` and `DemDataElementClass` was extended from 0..255 to 0..65535. This schema adaptation does not conform to [ecuc_sws_6008].

Rationale:

Due to enhancement "Enhanced multiplicity of `DemFreezeFrameClass` and `DemDataElementClass`", the multiplicity of `DemFreezeFrameClass` and `DemDataElementClass` is extended to 0..65535. See AUTOSAR Bugzilla RFCs https://www.autosar.org/bugzilla/show_bug.cgi?id=61653 and https://www.autosar.org/bugzilla/show_bug.cgi?id=55110.

Requirements:

Dem673_Conf, Dem610_Conf

- Multiplicity of `DemDidClass`

Description:

The multiplicity of the configuration container `DemDidClass` was extended from 0..255 to 0..65534. This schema adaptation does not conform to [ecuc_sws_6008].

Rationale:

Due to enhancement "Enhanced multiplicity of `DemDidClass`", the multiplicity of `DemDidClass` is extended to 0..65534. Value 65535 is used for internal processing (calibration unused value). See AUTOSAR Bugzilla RFC https://www.autosar.org/bugzilla/show_bug.cgi?id=55110.

Requirements:

Dem706_Conf

- Unlimited MIL activation configuration

Description:

The MIL configuration is not limited to the dedicated activation with the CDTC transition to 1. However, warning indicators (including MIL) are activated synchronously (according to Dem504), while the CDTC bit is updated via the Dem_MainFunction() asynchronously according to Dem379.

Rationale:

The deviated requirement is removed in AUTOSAR R4.2.

Requirements:

Dem544

- Reduced set of PIDs calculated Dem internally (reference to product description: ASCPD-135)

Description:

The following Dem internal calculated PIDs are not supported:

- `PID $4D - time with MIL On (2 byte)`
- `PID $4E - time since last fault clear (2 byte)`

The parameters listed below are not supported in the current implementation:

- `DemOBDInputAcceleratorPaddleInformation`
- `DemOBDInputAmbientTemperature`
- `DemOBDInputDistanceInformation`
- `DemOBDInputEngineSpeed`
- `DemOBDInputEngineTemperature`
- `DemOBDInputProgrammingEvent`

Requirements:

Dem347, Dem304, Dem377, Dem378, Dem323, Dem324, Dem627, Dem704_Conf, Dem763_Conf, Dem762_Conf, Dem761_Conf, Dem759_Conf, Dem757_Conf, Dem772_Conf, Dem760_Conf, Dem602_Conf, Dem756_Conf, J1979-DA.AnnexB.PID4D, J1979-DA.AnnexB.PID4E, SWS_Dem_00703, SWS_Dem_00704, SWS_Dem_00347

- No support for In Use Monitoring Performance Ratio (IUMPR)

Description:

Dem does not support IUMPR, therefore the following configuration parameters are not supported:

- `DemRatioId`
- `DemIUMPRGroup`
- `DemRatioIdType`
- `DemDiagnosticEventRef`
- `DemFunctionIdRef`
- `DemSecondaryFunctionIdRef`

Also the related API-functions `Dem_DcmGetInfoTypeValue08()` and `Dem_DcmGetInfoTypeValue0B()` are not supported. They are provided as stubs and their return values are always E_NOT_OK.

Requirements:

Dem359, Dem360, Dem296, Dem361, Dem362, Dem297, Dem308, Dem299, Dem302, Dem313, Dem314, Dem315, Dem734_Conf, Dem737_Conf, Dem741_Conf, Dem735_Conf, Dem736_Conf, Dem782_Conf, SWS_Dem_01101, SWS_Dem_00316, SWS_Dem_00317, SWS_Dem_00298, SWS_Dem_00357, SWS_Dem_00358, SWS_Dem_00912, SWS_Dem_00913

- Fault Confirmation and Warning Indicator behavior

Description:

The Dem module provides a correction for Fault Confirmation and Warning Indicator handling according to OEM requirements. See ISO 14229-1:2006/2013 Figure D.9, RFC 70785, and RFC 71313. Now the following configuration parameters define the number of TestFailedThisOperationCycle status bit transitions (i.e. tested and failed cycles) for the respective DTC status bits to be activated:

- `DemEventFailureCycleCounterThreshold`
- `DemIndicatorFailureCycleCounterThreshold`

This means, the handling for setting the Fault Confirmation and Warning Indicator Requested status bits represents the number of triggering points (defined by the value of the configuration parameters mentioned above) with FAILED qualification of the event.

Rationale:

AUTOSAR R4.0.3 describes different triggering points which present the changes of the failure cycle (and not the FAILED qualification). This would require an additional cycle to the value configured and therefore the meaning of the threshold values would differ. In order to implement the OEM requirements described above the Dem module deviates slightly from the requirements Dem529 with Dem391, and Dem504 with Dem395. These requirements describe the setting behavior of Fault Confirmation and WarningIndicator according to Figure D.4 in ISO 14229-1:2013 (which contradicts Figure D.9).

Requirements:

Dem529, Dem504

- Development error `DEM_E_UNINIT` is not supported in the APIs `Dem_PreInit()` and `Dem_Init()`

Description:

The Dem module provides a correction for APIs `Dem_PreInit()` and `Dem_Init()` where reporting of development error `DEM_E_UNINIT` is replaced by `DEM_E_WRONG_CONDITION`.

Rationale:

AUTOSAR defines in requirement Dem124, if development error detection is enabled and any instance calls any Dem API, before the Dem was fully initialized (not pre-initialized in case of Dem364), the Dem module shall set the error code `DEM_E_UNINIT`. This requirement is violated in the case of APIs `Dem_PreInit()` and `Dem_Init()` which directly control initialization process. Therefore, `DEM_E_WRONG_CONDITION` (see Dem518) is a more suitable development error code for these APIs.

Requirements:

Dem124, SWS_Dem_00124, Dem364

- The Dem interfaces used by Dcm are based on AUTOSAR R4.3.1 specification

Description:

The Dem interfaces called by the Dcm module for the processing of the OBD and UDS service requests are based on the AUTOSAR R4.3.1 specification, but with the following limitations:

- The APIs `Dem_DcmGetAvailableOBDMIDs()`, `Dem_DcmGetNumTIDsOfOBDMID()` and `Dem_DcmGetDTRData()`, used for the processing of OBD service $06 request, are not supported and they are provided as stubs and will always return E_NOT_OK.
- The API `Dem_DcmReadDataOfPID91()`, used for fetching the value of PID$91 in the OBD service $01, is not supported. It is provided as stub and its return value is always E_NOT_OK.
- The API `Dem_DisableDTCRecordUpdate()` does not protect the event related data of the selected DTC within the selected DTCOrigin from being deleted.
- The result of `Dem_ClearDTC()` does not depend on the result of `DemClearEventAllowed` callback.
- The Dem module does not support a special handling of the mirror memory. `DemClearEventAllowed` callback is called for mirror memory as well.
- If the DTC is configured to use event combination, the callback `DemClearEventAllowed` is only called for the master event as representative of the event combination and regardless of its availability status.
- Development error `DEM_E_WRONG_CONFIGURATION` is not reported if the function `Dem_SetDTCFilter()` is called with DTCFormat or DTCOrigin values which are not supported by configuration.
- The configuration schema is based on AUTOSAR R4.0.3 and therefore, the full AUTOSAR R4.3.1 compliant functionality is not offered. For example:
  - Dem does not support the below AUTOSAR R4.3.1 configuration parameters related to the APIs called by Dcm:
    - `DemClearDtcNotification`
    - `DemClearDTCLimitation`
    - `DemTriggerMonitorInitBeforeClearOk`
  - Dem does not support the below configuration parameters related to `DemClient` container:

- `DemClientFunctionality`
- `DemClientUsesRte`
- `DemEventMemorySetRef`

- The configuration parameter `DemDTCFunctionalUnit` is not optional.
- The lower 5 bits of the SeverityMask, which represent the DTC class information (see [SWS_Dem_00937]), cannot be configured. For OBD variant OBDonUDS, the DTC class information of emission related DTCs is fixed set to DTCClass_1/DEM_SEVERITY_WWHOBD_CLASS_A (0x02).

- The C/S interface `ClearDTC` is not supported.

Rationale:

The APIs specified in the AUTOSAR R4.3.1 are necessary to allow parallel access to the event related data and fault memory to service multiple diagnostic requests at the same time.

Requirements:

From AUTOSAR R4.0.3: Dem327, Dem596, Dem597, Dem623, Dem624, Dem035, Dem626, Dem079, Dem242, Dem080, Dem243, Dem057, Dem061, Dem216, Dem217, Dem228, Dem229, Dem287, Dem288, Dem513, Dem208, Dem214, Dem215, Dem227, Dem281, Dem013, Dem230, Dem060, Dem213, Dem059, Dem171, Dem172, Dem270, Dem648, Dem212, Dem233, Dem234, Dem232, Dem594, Dem221, Dem218, Dem593, Dem209, Dem224, Dem595, Dem226, Dem225, Dem609, Dem547, Dem560, Dem629, Dem071, Dem576, Dem630, Dem074, Dem075, Dem631, Dem076, Dem240, Dem238, Dem239, Dem236, Dem574, Dem070, Dem235, Dem575, Dem316, Dem317, Dem298, Dem357, Dem358, Dem301

From AUTOSAR R4.1.3: SWS_Dem_00298

From AUTOSAR R4.3.1: SWS_Dem_00270, SWS_Dem_01182, SWS_Dem_01295, SWS_Dem_01202, SWS_Dem_01205, SWS_Dem_01206, SWS_Dem_01066, SWS_Dem_01067, SWS_Dem_01240, SWS_Dem_01241, SWS_Dem_00670, SWS_Dem_00573, SWS_Dem_91002, SWS_Dem_00666, SWS_Dem_01009, SWS_Dem_01294, SWS_Dem_01140, SWS_Dem_00766, SWS_Dem_00767, SWS_Dem_00768, SWS_Dem_00316, SWS_Dem_00317, SWS_Dem_01187, SWS_Dem_00760, SWS_Dem_00761, SWS_Dem_00762, SWS_Dem_00759, SWS_Dem_01301, SWS_Dem_01101, SWS_Dem_01108, SWS_Dem_00293, ECUC_Dem_00790, ECUC_Dem_00925, ECUC_Dem_00943, ECUC_Dem_00933, ECUC_Dem_00942

- Limited support for severity availability mask

Description:

The Dem module does not yet support the DTC severity availability mask calculation from the severity configuration of each DTC. For OBDonUDS, the DTC severity availability mask is fixed set to DTCClass_1 (0x02) for all emission related DTCs, as required by most OBD regulations, e.g. CCR1968.2. The Dem

module provides the DTC severity availability mask to the Dcm (via Dem_GetDTCSeverityAvailability-Mask()) only if OBD variant OBDonUDS is selected.

Requirements:

SWS_Dem_01182

- Base type of `Dem_DTCOriginType` changed to `uint16`.

Description:

The base type of `Dem_DTCOriginType` has been changed from `uint8` to `uint16` according to AUTOSAR 4.3.1. specification.

Rationale:

The Dem-Dcm interface implementation is based on AUTOSAR 4.3.1 specification. Consequently, the global `Dem_DTCOriginType` type, which is used in SW-C, CDD and Dem-Dcm APIs, is adapted from base type `unit8` to base type `uint16`. For SW-C/CDD interfaces, the Dem interpretation of `DTCOrigin` values [0x0001..0x0004] is not changed i.e., interpretation is still according to AUTOSAR 4.0/4.2. In addition, if `DemUserDefMemoryId` is configured, the secondary event memory can be accessed by a `DTCOrigin` value, equal to the configured `DemUserDefMemoryId + 0x0100`.

- Upgrade of event memory overflow indication.

Description:

The API `Dem_GetEventMemoryOverflow()` has the type for the input parameter `DTCOrigin` upgraded to an `uint16`.

Rationale:

The functionality event memory overflow indication defined by the C API `Dem_GetEventMemoryOverflow()` has the DTCOrigin information represented by the `Dem_DTCOriginType` definition according to AUTOSAR R4.3. The `Dem_DTCOriginType` has been adapted from `uint8` to `unit16` type for supporting the extended range of user defined memory, as defined by AUTOSAR R4.3. The Rte interface is not affected since the argument for `DTCOrigin` is a PortDefinedArgumentValue for `GetEventMemoryOverflow` operation and the C-level API `Dem_GetEventMemoryOverflow()` is also not impacted since a larger type is provided.

Requirements:

Dem559

- No support for the APIs `GetCycleQualified()` and `RestartOperationCycle()`.

Description:

The operations `GetCycleQualified()` and `RestartOperationCycle()` of the interface `OperationCycle` and the corresonding APIs are not implemented. The restart of a driving cycle is done with the function `Dem_SetOperationCycleState()`, which is kept for backwards compatibility.

Rationale:

These APIs were not requested by the customer.

Requirements:

SWS_Dem_00601

- J1939 sub-features

Description:

The following J1939 sub-features are not implemented: Monitor Performance Ratio (DM20), Diagnostic Readiness 2 (DM21), Diagnostic Readiness 3 (DM26), Expanded Freeze Frame (DM25), DTC to Lamp Association (DM31) and callback on every J1939 DTC status change configured in DemCallbackJ1939DTCStatusChanged.

The following APIs are not supported (they are provided as stubs only):

- `Dem_J1939DcmSetRatioFilter()`
- `Dem_J1939DcmGetNextFilteredRatio()`
- `Dem_J1939DcmReadDiagnosticReadiness2()`
- `Dem_J1939DcmReadDiagnosticReadiness3()`
- `Dem_J1939DcmFirstDTCwithLampStatus()`
- `Dem_J1939DcmGetNextDTCwithLampStatus()`
- `Dem_J1939DcmGetNextSPNInFreezeFrame()`

Requirements:

SWS_Dem_00880, SWS_Dem_00881, SWS_Dem_00882, SWS_Dem_00883, SWS_Dem_00884, SWS_Dem_00885, SWS_Dem_00886, SWS_Dem_00887, SWS_Dem_00888, SWS_Dem_00889, SWS_Dem_00890, SWS_Dem_00891, SWS_Dem_00892, SWS_Dem_00893, SWS_Dem_00894, SWS_Dem_00895, SWS_Dem_00896, SWS_Dem_00897, SWS_Dem_00898, SWS_Dem_00904, SWS_Dem_00905, SWS_Dem_00906, SWS_Dem_00907, SWS_Dem_00910, SWS_Dem_00911, SWS_Dem_00912, SWS_Dem_00913, SWS_Dem_00974, SWS_Dem_00975, ECUC_Dem_00834, SWS_Dem_00983, SWS_Dem_00770, SWS_Dem_00979, SWS_Dem_00980, SWS_Dem_00981, SWS_Dem_00987

- Dem configuration structure for J1939 extension is deviating from AUTOSAR R4.2.2

Description:

The J1939 Dem extension is implemented as per AUTOSAR R4.2.2 whereas the Dem configuration schema is originally based on AUTOSAR R4.0.3. The following deviations are applied due to compatibility with AUTOSAR R4.0.3 and to simplify the configuration:

- The Dem J1939 feature is enabled via the configuration switch `DemJ1939Support`. The container `DemGeneralJ1939` is always present. This adaptation does not conform to [SWS_Dem_00845] and [ECUC_Dem_00864].
- Parameters `DemAmberWarningLampIndicatorRef [ECUC_Dem_00821]`, `DemProtectLampIndicatorRef` [ECUC_Dem_00822] and `DemRedStopLampIndicatorRef` [ECUC_Dem_00820] are placed in container `DemGeneral` instead of `DemGeneralJ1939`.
- Container `DemJ1939Node` [ECUC_Dem_00817] and reference `DemJ1939DTC_J1939NodeRef` [ECUC_Dem_00819] are not implemented. Instead parameter `DemJ1939NodeRef` is available in container `DemDTCClass` providing the functionality of the AUTOSAR parameter `DemJ1939NmNodeRef` [ECUC_Dem_00818]. This implementation allows to assign one DTC to one `J1939NmNode`.
- `DemJ1939DTCValue` [ECUC_Dem_00892] is part of container `DemDTCClass` due to compatibility with AUTOSAR R4.0.3.
- `DemJ1939FreezeFrameClassRef` [ECUC_Dem_00835] is part of container `DemDTCClass` due to compatibility with AUTOSAR R4.0.3.

Requirements:

SWS_Dem_00845, ECUC_Dem_00864, ECUC_Dem_00817, ECUC_Dem_00818, ECUC_Dem_00819, ECUC_Dem_00820, ECUC_Dem_00821, ECUC_Dem_00822, ECUC_Dem_00892, ECUC_Dem_00835

- Configuration parameters specific to AUTOSAR R20-11 RTE interfaces/ports are not supported.

Description:

The configuration schema is based on AUTOSAR R4.0.3 and therefore, the following AUTOSAR R20-11 configuration parameters are not supported:

- `DemGeneralInterfaceSupport`
- `DemCallbackEventUdsStatusChanged` and `DemCallbackEventUdsStatusChangedFnc`
- `DemCallbackOBDDTCStatusChanged` and `DemCallbackJ1939DTCStatusChanged`

Rationale:

The current implementation of the AUTOSAR R20-11 RTE interfaces/ports is meant to be used on ACG-8 configuration schema.

Requirements:

From AUTOSAR R20-11: ECUC_Dem_00880, ECUC_Dem_00628, ECUC_Dem_00629, SWS_Dem_-01031, SWS_Dem_01032, SWS_Dem_01008, SWS_Dem_00986, SWS_Dem_00987, SWS_Dem_-00617, ECUC_Dem_00823, ECUC_Dem_00824, ECUC_Dem_00825, ECUC_Dem_00826

- Limited support for the RTE interfaces/operations specific to AUTOSAR R20-11.

  Description:

  The RTE interfaces/operations specific to AUTOSAR R20-11 are supported with the following limitations:

  - `GeneralDiagnosticInfo` interface:

    The parameters names and types are not updated as per AUTOSAR R20-11 for the `GetEventUdsS-tatus` operation.

    Possible return errors and parameters types are not updated as per AUTOSAR R20-11 for `GetEven-tExtendedDataRecordEx` and `GetEventFreezeFrameDataEx` operations.

    Possible return errors are not updated as per AUTOSAR R20-11 for `GetFaultDetectionCounter` operation.

    Operation `GetMonitorStatus` is not supported.

  Rationale:

  The current implementation of the AUTOSAR R20-11 RTE interfaces/operations is meant to be used on ACG-8 Diagnostic Stack, which is based on AUTOSAR R4.0.3.

  Requirements:

  From AUTOSAR R20-11: SWS_Dem_00600

- The AUTOSAR R20-11 prototype of the API `Dem_GetEventUdsStatus()` is not supported.

  Description:

  The prototype of the API `Dem_GetEventUdsStatus()` is not updated as per AUTOSAR R20-11 as it has no impact on Dem functionality. The prototype `Dem_GetEventStatus()` provided by AUTOSAR R4.0.3 is used instead, for which the backward compatibility with SW-Cs or other BSW modules is ensured.

  Rationale:

  The current implementation of the AUTOSAR R20-11 interfaces is meant to be used on ACG-8 Diagnostic Stack, which is based on AUTOSAR R4.0.3.

  Requirements:

  From AUTOSAR R20-11: SWS_Dem_00051, SWS_Dem_91008

- Dem support for API `Dem_SetEventFailedWithSyncFreezeFrame()` is deviating from AUTOSAR R20-11

  Description:

  To support the Security Event Memory (Sem) the API `Dem_SetEventFailedWithSyncFreeze-Frame()` and the corresponding configuration structure was introduced in Autosar R20-11. But the Dem configuration schema is originally based on AUTOSAR R4.0.3. For compatibility and simplicity reasons the following deviations are applied:

  - The configuration parameter `DemEventReportingType` is not supported. Instead the parameter `DemEventKind` was extended by the value `DEM_EVENT_KIND_SEV` to configure security events that must be reported with API `Dem_SetEventFailedWithSyncFreezeFrame()`.
  - The API `Dem_SetEventFailedWithSyncFreezeFrame()` must not be called before Dem is fully initialized.
  - No StatusChange callback is called for the events with `DemEventKind` set to `DEM_EVENT_KIND_-SEV`.
  - The UDS snapshot record data may not always be captured synchronous within the call of API `Dem_-SetEventFailedWithSyncFreezeFrame()`.
  - DemMultiEventTriggering is not supported for events with `DemEventKind` set to `DEM_EVENT_-KIND_SEV`.
  - Event combination is not supported for events with `DemEventKind` set to `DEM_EVENT_KIND_SEV`.
  - Events with `DemEventKind` set to `DEM_EVENT_KIND_SEV` must not configure a `DemIndicatorAt-tribute`.
  - Freeze frames configured for events with `DemEventKind` set to `DEM_EVENT_KIND_SEV` must contain only one DID with only a single DataElement.
  - DemCallbackEventStatusChanged is not supported for events with `DemEventKind` set to `DEM_-EVENT_KIND_SEV`.

  Requirements:

  SWS_Dem_01353, SWS_Dem_01250, SWS_Dem_00124, ECUC_Dem_00970, Dem040, Dem510, Dem642_Conf, Dem628_Conf

- Limited support for emission related diagnostic information specific to AUTOSAR R21-11.

  Description:

  The following emission related diagnostic information is not supported by the Dem:

  - Activation Mode
  - Continuous-MI Counter
  - Highest ECU B1 Counter

- B1 Counter

Rationale:

Full support for WWH-OBD was not requested by the customer.

Requirements:

From AUTOSAR R21-11: SWS_Dem_01178

- The Dem interfaces used by Dcm for OBDonUDS services are based on AUTOSAR R21-11 specification

Description:

The Dem interfaces called by the Dcm module for the processing of OBDonUDS service requests are based on the AUTOSAR R21-11 specification, but with the following limitations:

- The configuration schema is based on AUTOSAR R4.0.3. Additional OBDonUDS related configuration parameters are added to corresponding R4.0.3 containers.

- Dem does not support centralized PID $21/$31 handling.

Description:

For PID $21/$31, only localized calculation is supported. If the configuration parameter DemOBDCentralizedPID21Handling/DemOBDCentralizedPID31Handling is enabled, Dem_DcmReadDataOfPID21() will always return 0.

Rationale:

Centralized handling is wrong specified in AUTOSAR.

Requirements:

SWS_Dem_00703, SWS_Dem_00704, Dem.ASR431.SWS_Dem_01099, SWS_Dem_00377, SWS_Dem_00378, J1979-DA.PID21, J1979-DA.PID31

2.5. Limitations

This chapter lists the limitations of the module. Refer to the module references chapter *Integration notes*, subsection *Integration requirements* for requirements on integrating this module.

- No support for link time and post-build time configurations (reference to product description: ASCPD-77)

Description:

The Dem module can only be configured at pre-compile time. Link time and post-build time configurations are not supported.

Rationale:

Source code can be optimized with respect to code size and execution speed more aggressively if only pre-compile time configuration must be supported.

Requirements:

Dem108, Dem268

- Implementation-specific parameter range and container multiplicity limitations

Description:

The configuration-scheme as specified in `AUTOSAR_EcucParamDef.arxml` version 3.1.2 was changed as follows:

- Parameter `DemBswErrorBufferSize`: range limited and corrected from 0..255 to 1..255, because the error queue cannot be switched off completely.
- Parameter `DemImmediateNvStorageLimit`: range limited and corrected from 1..255 to 1..254.
- Parameter `DemTaskTime`: range limited and corrected from 0..0.1 to 0.001..0.1, because 0 is no valid task time (refer to RfC #50351).
- Container `DemOperationCycle`: multiplicity limited and corrected from 1..256 to 1..255, because the API `Dem_SetOperationCycleState()` accepts only 8-bit wide IDs.
- Parameter `DemEnableConditionID`: range limited and corrected from 0..255 to 0..254, because the parent container `DemEnableCondition` multiplicity is limited to 255. Also it is not possible to configure a value of 255 for the `DemEnableConditionID` due to the additional zero-based and consecutive constraint.
- Parameter `DemIndicatorID`: range limited and corrected from 0..255 to 0..254, because the parent container `DemIndicator` multiplicity is limited to 255. Also it is not possible to configure a value of 255 for the `DemIndicatorID` due to the additional zero-based and consecutive constraint.
- Parameter `DemPidIdentifier`: range limited and corrected from 0..255 to 0..254, because the value 255 is used to disable PIDs in context of calibration.
- Container `DemGroupOfDTC`: multiplicity limited from 0..255 to 1..30.
- Container `DemConfigSet`: multiplicity limited to 1..1, because only pre-compile time configuration is supported.
- Container `DemEventParameter`: multiplicity limited from 1..65535 to 1..8190.
- Parameter `DemEventId`: range limited from 1..65535 to 1..8190.
- Parameter `DemMaxNumberFreezeFrameRecords`: range limited and corrected from 0..255 to 1..255, because the existence of FF data is controlled via `DemFreezeFrameClassRef` multiplicity.
- Parameter `DemAgingCycleCounterThreshold`: range limited from 1..256 to 1..255.

- Parameter `DemAgingCycleCounterThreshold`: multiplicity limited and corrected from 0..1 to 1..1, due to the dependency to the parameter `DemAgingAllowed`.
- Parameter `DemEventFailureCycleCounterThreshold`: range limited from 1..256 to 1..254
- Parameter `DemIndicatorFailureCycleCounterThreshold`: range limited from 0..255 to 1..255 (refer to the deviation "Fault Confirmation and Warning Indicator behavior" and RfC #71313)
- Parameter `DemDebounceCounterDecrementStepSize`: range limited and corrected from 0..32768 to 1..32768, because step-size zero does not lead to any change (refer to RfC #50351).
- Parameter `DemExtendedDataCapture`: multiplicity limited and corrected from 0..1 to 1..1 (refer to RfC #59376)
- Parameter `DemFreezeFrameCapture`: multiplicity limited and corrected from 0..1 to 1..1 (refer to RfC #59376)
- Parameter `DemDebounceCounterFailedThreshold`: range limited and corrected from 0..32767 to 1..32767, because debounce counter start-value is not allowed as qualification threshold (refer to RfC #50351).
- Parameter `DemDebounceCounterIncrementStepSize`: range limited and corrected from 0..32767 to 1..32767, because step-size zero does not lead to any change (refer to RfC #50351).
- Parameter `DemDebounceCounterPassedThreshold`: range limited and corrected from -32768..0 to -32768..-1, because debounce counter start-value is not allowed as qualification threshold (refer to RfC #50351).
- Parameters `DemDebounceTimeFailedThreshold` and `DemDebounceTimePassedThreshold`: range limited and corrected from 0 to 0.001, because 0 is no valid threshold for time (refer to RfC #50351).
- Container `DemExtendedDataClass`: multiplicity limited from 0..unbound to 0..256.
- Parameter `DemExtendedDataRecordNumber`: range limited and corrected from 0..253 to 1..239, because 0x00 and 0xF0-0xFF is reserved according to ISO 14229-1.
- Parameter `DemEnableConditionGroup`: Multiplicity limited and corrected from 0..255 to 0..254, because the value of 255 is used for internal processing.
- Parameter `DemEventPriority`: range limited and corrected from 1..256 to 1..255, because the priority can be mapped to extended data.

Rationale:

Parameter limitations allow for more efficient implementation and solve SWS issues.

Requirements:

Dem625_Conf, Dem715_Conf, Dem701_Conf, Dem654_Conf, Dem683_Conf, Dem679_Conf, Dem634_Conf, Dem661_Conf, Dem659_Conf, Dem605_Conf, Dem623_Conf, Dem750_Conf,1, Dem753_Conf,

Dem635_Conf, Dem618_Conf, Dem637_Conf, Dem636_Conf, Dem716_Conf, Dem717_Conf, Dem664_Conf, Dem666_Conf, Dem662_Conf

- Limitation on usage of Dem configuration parameter `DemPidDataElement` and API `Dem_DcmReadDataOfOBDFreezeFrame()`

  Description:

  If the order of data elements that are configured in `DemPidDataElement` differs from the configured order of data elements inside a PID (see configuration parameter `DcmDspPidDataPos`), the API `Dem_DcmReadDataOfOBDFreezeFrame()` returns wrong PID data elements.

  Rationale:

  The current implementation of API `Dem_DcmReadDataOfOBDFreezeFrame()` does not take into account the PID data element order that is configured in Dcm (DcmDspPidDataPos). Therefore, the order of data elements inside a PID of Dem that are configured in `DemPidDataElement` must accord with the configured order of data elements inside a PID of Dcm in `DcmDspPidDataPos`.

  Requirements:

  SWS_Dem_00597

- For AUTOSAR R4.2 service API: Function `Dem_ClearDTC()` is not re-entrant

  Description:

  According to AUTOSAR R4.2.1, requirement `SWS_Dem_00665`, `Dem_ClearDTC()` shall be re-entrant. The current implementation of this API is defined as "non re-entrant" according to AUTOSAR R4.0.3.

  Rationale:

  The behavior of this interface is not clearly specified for this case. RFC 64149 was created for clarification of this topic.

  Requirements:

  SWS_Dem_00665

- Usage of internal data elements is limited only for extended data records

  Description:

  The configuration parameters `DemPidDataElementClassRef`, `DemDidDataElementClassRef` and `DemSPNDataElementClassRef` do not support referencing to `DemInternalDataElementClass`. The `DemInternalDataElementClass` elements can only be referenced by an `DemExtendedDataRecordClass`.

  Rationale:

There are no known use cases where PIDs, DIDs or SPNs need to contain data defined via `DemInternalDataElementClass` elements. Newer AUTOSAR versions also specify this constraint explicitly.

Requirements:

Dem617_Conf, Dem733_Conf, ECUC_Dem_00832 and Dem469

- For AUTOSAR R4.2 service API: Function `Dem_<...>ClearDTC()` supports only `DEM_DTC_GROUP_ALL_DTCS` and `DEM_DTC_GROUP_EMISSION_REL_DTCS` of the DTC groups if the DTC format is set to `DEM_DTC_FORMAT_OBD`

Description:

According to AUTOSAR R4.2.1, requirement `SWS_Dem_00665`, `Dem_ClearDTC()` shall clear groups of DTCs for different DTC formats. The current implementation of this API prohibits the clearing of DTC groups with `DEM_DTC_FORMAT_OBD`, except for DTC group `DEM_DTC_GROUP_ALL_DTCS` (0xffffff) and `DEM_DTC_GROUP_EMISSION_REL_DTCS` (0x000000).

Rationale:

The behavior of this interface is not clearly specified for this case.

Requirements:

SWS_Dem_00665

- Functions `Dem_GetNextFreezeFrameData()`, `Dem_SetFreezeFrameRecordFilter()` and `Dem_GetNextFilteredRecord()` consider the global OBD freeze frame only for the DTC that triggered the storage

Description:

Only for the emission related DTC that stored the OBD freeze frame, the API `Dem_GetNextFreezeFrameData()` responds with the currently stored OBD freeze frame data, when record number 0x00 is requested.

For all other emission related DTCs, `Dem_GetNextFreezeFrameData()` returns E_OK with a size of 0, when record number 0x00 is requested. `Dem_GetNextFreezeFrameData()` is used, e.g., to respond to diagnostic service 0x19 0x04.

Similar behavior applies also to the APIs `Dem_SetFreezeFrameRecordFilter` and `Dem_GetNextFilteredRecord()` which are used to respond to diagnostic service 0x19 0x03.

Rationale:

If Dem is configured to support only one global OBD freeze frame via `DemFreezeFrameCapture`, then only one DTC can return the OBD freeze frame in function `Dem_GetNextFreezeFrameData()`. Even

if Dem is configured to support multiple OBD freeze frames, `Dem_GetNextFreezeFrameData()` will return the OBD freeze frame only for one DTC to have the same behavior.

In order to have a consistent response for service 0x19 subfunctions 0x03 and 0x04, the APIs `Dem_SetFreezeFrameRecordFilter()` and `Dem_GetNextFilteredRecord()` have the same limitation.

Note: Pids may be different between OBD service 0x02 and UDS service 0x19 0x04.

- OBD service 0x02: The Dcm arranges the DemPidDataElements inside the Pid.
- UDS service 0x19 0x04 record number 0x00: The Dcm passes the data directly from the Dem. The Dem cannot arrange the DemPidDataElements because the information is not available inside the Dem.

Requirements:

SWS_Dem_00576, SWS_Dem_00209

- Processing of unconfirmedDTC status during Dem pre-initialization

Description:

The unconfirmedDTC status for events reported via `Dem_ReportErrorStatus()` is not processed if the Dem module is pre-initialized, i.e., `Dem_Init()` was not called yet. The unconfirmedDTC status can be reached via debouncing.

Rationale:

At this time the event memory entries are not restored from NVRAM yet and therefore the related counters and status data (stored in the memory entries) cannot be processed together with the unconfirmedDTC status.

Requirements:

VCC_DEM_Table_12, VCC_DEM_Table_13

- Capture of `DemExtendedDataCapture` and/or `DemFreezeFrameCapture` on `DEM_TRIGGER_TESTFAILED`

Description:

Synchronous data capturing (`DemExtendedDataCapture` and/or `DemFreezeFrameCapture` equals `DEM_TRIGGER_TESTFAILED`) is possible only for events reported via `Dem_SetEventStatus()`. The data capturing for events reported via `Dem_ReportErrorStatus()` is always done asynchronously in the main function regardless of configured `DemExtendedDataCapture` and `DemFreezeFrameCapture`.

Rationale:

The data for events reported via `Dem_ReportErrorStatus()` can only be captured asynchronously during the processing of the event queue in the main function.

Requirements:

Dem461, Dem467

- Function `Dem_<...>ClearDTC()` clears only the first event matching the requested single OBD DTC to be cleared, in case when more than one event are referring to same OBD DTC value

Description:

Any call of function `Dem_<...>ClearDTC()` with `DTC` which is used by several DTC classes and `DEM_DTC_FORMAT_OBD` as input parameters, only the first matching event to the DTC will be cleared.

Rationale:

As long as the behavior of this interface is not clearly specified, in case when identical OBD DTC is supported, the current implementation of this API prohibits the clearing of all events assigned to the same OBD DTC value.

Requirements:

Dem077

- Implementation-specific parameter range and container multiplicity limitations

Description:

The configuration-scheme as specified in `AUTOSAR_EcucParamDef.arxml` version 4.3.1 was changed as follows:

- Parameter `DemClientId`: range limited from 0..255 to 0..253.

Rationale:

Parameter limitations allow for more efficient implementation.

Requirements:

ECUC_Dem_00932

- Only one client at a time can control DTC setting

Description:

`Dem_EnableDTCSetting()` and `Dem_DisableDTCSetting()` cannot control the DTC setting if it is already disabled by a different client.

Rationale:

This is because the DemEventMemorySet functionality is missing which causes a call of `Dem_EnableDTCSetting()` or `Dem_DisableDTCSetting()` to affect all events associated with a DTC.

Requirements:

SWS_Dem_00242, SWS_Dem_00243

- All clients have access to the same event memory set.

Description:

Dem supports only one global event memory set. It is accessible by all configured clients and the Dem APIs do not work differently based on the ClientId.

Rationale:

The feature of multiple `DemEventMemorySet`s and restricting access to a single `DemEventMemorySet` for a given client, as specified by AUTOSAR R4.3.1, is not supported currently.

Requirements:

SWS_Dem_00242, SWS_Dem_00243, SWS_Dem_00219, SWS_Dem_01263, SWS_Dem_00231, ECUC_Dem_00940, ECUC_Dem_00939

- Only events with UDS DTC are impacted by control DTC setting

Description:

`Dem_EnableDTCSetting()` and `Dem_DisableDTCSetting()` do not affect events configured with DTCs which are OBD only.

Rationale:

The current Dem implementation does not consider configurations with events which have only OBD DTCs assigned without having configured also UDS DTCs.

Requirements:

SWS_Dem_01290

- No support for the parallel call to the DTC selection related APIs for the same client

Description:

The Dem implementation does not support parallel call to the DTC selection related APIs for the same client. The relevant APIs are `Dem_SelectDTC()` and all the APIs that require a `Dem_SelectDTC()` according to SWS_Dem_01253. Consequently, these APIs do not support the return value DEM_BUSY as well.

Rationale:

Explicit exclusion of parallel call to the DTC selection related APIs for the same diagnostic client allows for a more efficient implementation. When considered together with the fact that the DTC selection related APIs are implemented as synchronous, such an exclusion does not lead to any loss of functionality in practice and becomes only a theoretical consideration.

Requirements:

SWS_Dem_01305, SWS_Dem_01306

- Impact of DTC suppression and event availability on DTC selection

Description:

If any operation based on the current DTC selection has already started, then any changes in the event available status and DTC suppression status have an impact on the functionality of the following APIs only after a new DTC selection. The impacted APIs are:

- `Dem_GetStatusOfDTC`

  `Dem_GetSeverityOfDTC`

  `Dem_GetFunctionalUnitOfDTC`

  `Dem_SelectFreezeFrameData`

  `Dem_SelectExtendedDataRecord`

Rationale:

This behavior is necessary to ensure that coherent results are returned by the above mentioned APIs, even if the event availability/DTC suppression status is changed in-between.

Requirements:

SWS_Dem_01100, SWS_Dem_01101

- Only one client at a time can control DTC record update

Description:

`Dem_DisableDTCRecordUpdate()` and `Dem_EnableDTCRecordUpdate()` cannot control the DTC record update if it is already disabled by a different client.

Rationale:

Parallel processing of UDS service 0x19, sub-functions 0x04, 0x06 is not supported and a diagnostic client can select only a single DTC. Therefore, there is no use case for locking multiple DTCs/event memory entries at the same time.

Requirements:

SWS_Dem_00233, SWS_Dem_00234

- No support for user defined memories

Description:

The user defined memories as specified by AUTOSAR versions R4.3.1 and newer are not supported. However, a possibility is provided to access the secondary memory as a user defined memory with the help of the configuration parameter `DemUserDefMemoryId`.

Rationale:

The event memory implementation is based on AUTOSAR R4.0.3 which does not support user defined memories.

Requirements:

SWS_Dem_00548

- Each event is assigned to exactly one event memory only

Description:

Each event is assigned to exactly one event memory only. Therefore, events without DTC event memory reference are not applicable in the EB solution.

Rationale:

There is no use case yet for supporting events which are not assigned to any event memory.

Requirements:

SWS_Dem_01249, ECUC_Dem_00941

- Clearing operation is locked until the result is returned to the requesting client or Dem_SelectDTC is called for the same client

Description:

The Dem does not release the locked clearing process until the result is returned to the requesting client or the same client calls Dem_SelectDTC, indicating the client is no longer interested in the result of the clearing operation.

Rationale:

Keeping the clearing process locked until the result is read or dropped allows a more efficient implementation.

Requirements:

SWS_Dem_01042

- Only one client at a time can filter stored DTCs

Description:

`Dem_DcmSetStoredDTCFilter()` and `Dem_DcmGetNextFilteredStoredDTC()` cannot be called by a client while a different client has already set the filter and has not yet finished reading all the filtered DTCs.

Rationale:

The APIs are used only by the UDS services and currently, the Dem implementation does not support UDS in parallel to UDS. Besides, these APIs are used together with the APIs related to DTC record update, which themselves do not support multiple clients in parallel. Therefore, there is no use case currently to support concurrent use of `Dem_DcmSetStoredDTCFilter()` and `Dem_DcmGetNextFilteredStoredDTC()` by multiple clients.

- MIL grouping is not supported for trigger point `DEM_TRIGGER_ON_FDC_THRESHOLD`

Description:

MIL grouping is not supported if parameters `DemFreezeFrameCapture` or `DemExtendedDataCapture` are configured to `DEM_TRIGGER_ON_FDC_THRESHOLD`.

Rationale:

This configuration is an Elektrobit specific extension. So far there is no use case defined for this configuration that requires the MIL grouping feature.

Requirements:

ECUC_Dem_00839, SWS_Dem_00965, SWS_Dem_00967, SWS_Dem_00968

- Not supported Readiness groups

Description:

The below readiness groups are currently not supported.

- `DEM_OBD_RDY_CSER`
- `DEM_OBD_RDY_DOR`
- `DEM_OBD_RDY_ECS`
- `DEM_OBD_RDY_NOXADSORB`
- `DEM_OBD_RDY_OTHER`

- DEM_OBD_RDY_PCV
- DEM_OBD_RDY_VVT

Rationale:

The current support of readiness group is based on AUTOSAR R4.1.3 except for GPF which is addressed in AUTOSAR R21-11.

Requirements:

ECUC_Dem_00755 of AUTOSAR R21-11

2.6. Open-source software

The Dem module does not use open-source software.