Elektrobit

EB tresos® AutoCore Generic 8 Diagnostic Stack documentation

release notes update for the Dcm 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 `Dcm` 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: 29.2.1
- AUTOSAR R4.0 Rev 3
- Build number: B587955

2. Dcm module release notes

- AUTOSAR R4.0 Rev 3
- AUTOSAR SWS document version: 4.2.0
- Module version: 5.0.15.B587955
- Supplier: Elektrobit Automotive GmbH

2.1. Change log

This chapter lists the changes between different versions.

Module version 5.0.15

2022-11-30

- ASCDCM-6457 Fixed known issue: The Dcm does not stop the ROE when transitioning from a non-default session to the same non-default session

Module version 5.0.14

2022-09-23

- ASCDCM-6247 Fixed known issue: Undefined behavior might occur for an OBD service 0x04 request with wrong size on confirmation of RCRRP (NRC 0x78) response transmission
- ASCDCM-6242 Fixed known issue: The Dcm triggers a Det error when a RoutineControl service (0x31) is defined with a variable length input signal and no variable length output signal
- Support for UDS service `Authentication(0x29)` - Bidirectional
- ASCDCM-6342 Fixed known issue: Dcm does not correctly process requests greater than 255 bytes for SubFunction 0x02 (defineByMemoryAddress) of UDS service DynamicallyDefineDataIdentifier (0x2C)
- ASCDCM-6353 Fixed known issue: The Dcm transmits random data for the missing DTCAndSeverityRecord in the paged-buffer response for the subservice reportDTCBySeverityMaskRecord (0x08) of UDS service ReadDTCInformation (0x19)
- Implemented subfunction reportWWHOBDDTCByMaskRecord (0x42) of UDS service ReadDTCInformation (0x19)
- Implemented support for calibration of DIDs via parameter `DcmDspDidUsed`

- Implemented OBDonUDS - Generic Dem - ReadDataStream

Module version 5.0.13

2022-06-14

- Implemented UDS service `Authentication(0x29)`
- ASCDCM-5964 Fixed known issue: The Dcm does not transmit the NRC RequestOutOfRange (0x31) for UDS services with parameter DTCExtDataRecordNumber for the unsupported data record 0xF0
- ASCDCM-5966 Fixed known issue: Out-of-bounds memory access may occur when the UDS service RoutineControl (0x31) is requested with a TID inside the OBD range
- ASCDCM-6160 Fixed known issue: An incompatible ReturnControlToEcu operation is invoked for asynchronous DID signals when DcmDefaultASRServiceAPI is set to AUTOSAR_42
- Implemented support for MemorySelection byte for the `ClearDiagnosticInformation (0x14)` service request
- ASCDCM-6163 Fixed known issue: The ECU becomes unresponsive when the UDS service RoutineControl (0x31) is requested with a TID inside the OBD range and OBD service $08 is not configured
- ASCDCM-5972 Fixed known issue: Compiler errors occur if DcmDslCallbackDCMRequestService entries are configured and DcmRteUsage is disabled
- Implemented `DcmDsdSubServiceUsed` parameter to activate or deactivate the usage of a subservice.
- ASCDCM-6185 Fixed known issue: The ECU may block or reset if the UDS service ReadDataByIdentifier (0x22) is requested for a DID containing data which need to be swapped
- ASCDCM-6205 Fixed known issue: Dcm reports ROE event onDTCStatusChange for a status bit which is not enabled in the event configuration
- Dcm processes higher priority internal ROE event instead of already accepted external UDS request that is queued for processing

Module version 5.0.11

2022-02-18

- ASCDCM-5871 Fixed known issue: The Dcm does not compile if the memory range by label list contains an entry with empty memory labels and at least one memory service is configured
- ASCDCM-5860 Fixed known issue: Dcm generation fails if data transfer services check is disabled, the memory access container is empty, and at least one data transfer service is configured
- ASCDCM-5905 Fixed known issue: Undefined Dcm behavior when service ReadScalingDataByIdentifier (0x24) and DcmModeDeclarationSupport are both enabled

- ASCDCM-5913 Fixed known issue: Dcm does not send ROOR NRC for service ResponseOnEvent (0x86) subfunction onDTCStatusChange when using finite event window and storageState equal to storeEvent
- ASCDCM-5910 Fixed known issue: The Dcm does not compile if ReadScalingDataByIdentifier (0x24) is configured and either of the following configuration items are missing DcmDspDid, DcmDspDidSignal, DcmDspDidReadScalingData or DcmDspDataScalingInfoSize
- ASCDCM-5931 Fixed known issue: The Dcm does not execute DID-related service requests correctly after processing a service 0x24 request
- ASCDCM-5939 Fixed known issue: The Dcm generates inconsistent external function declarations and operation calls for RoutineControl (0x31)
- Added the option to change the call order of Dem_SetDTCFilter() API and Dem_GetTranslationType() API for service `ReadDTCInformation (0x19)`
- ASCDCM-5928 Fixed known issue: Unreachable code assert is triggered in a pre-emption on pre-emption scenario if the transmission of a RequestCorrectlyReceivedResponsePending (0x78) negative response code is not confirmed
- ASCDCM-5929 Fixed known issue: The Dcm does not trigger a configuration error for all cases of routines with input signals larger than the reception buffer
- Updated Pdu MetaData handling: When responding to functional requests, a configurable SourceAddress is used instead of the retrieved address from EcuC

Module version 5.0.10

2021-10-08

- ASCDCM-5668 Fixed known issue: The Dcm triggers the creation of redundant NvM blocks
- ASCDCM-5693 Fixed known issue: The Dcm notifies application with wrong confirmation type for RequestCorrectlyReceivedResponsePending (0x78) negative response code transmission
- ASCDCM-5743 Fixed known issue: The DCM returns positive response with no data to a related diagnostic DID service request if the requested DID has the DcmDspDataSize configuration parameter set to maximum value
- ASCDCM-5822 Fixed known issue: The Dcm triggers the deletion of created NvM blocks
- ASCDCM-5813 Fixed known issue: The Dcm returns a positive response with incomplete data to a related diagnostic DID service request if the requested DID has a variable signal that returns a data length of greater than 8192 bytes

Module version 5.0.9

2021-06-25

- ASCDCM-5641 Fixed known issue: The Dcm declares the user callouts for service InputOutputControlByIdentifier (0x2F) with an incorrect parameter name
- ASCDCM-5584 Fixed known issue: The Dcm transmits incorrect responses to requests for the UDS service ReadDTCInformation (0x19) with subfunction reportDTCSnapshotIdentification (0x03)
- ASCDCM-5619 Fixed known issue: Service RequestTransferExit (0x37) behaves unexpectedly when DcmDataTransferServicesASRVersion is set to 'AUTOSAR_43' or 'AUTOSAR_422'
- ASCDCM-5588 Fixed known issue: The Dcm does not process a service request when replying immediately with response pending (RCRRP) transmission
- ASCDCM-5575 Fixed known issue: The DCM blocks the protocol on which a related diagnostic DID service is requested if the configuration of DCM does not contain any DID signal
- ASCDCM-5501 Fixed known issue: Generation fails if call-back functions are used for a routine with configuration parameter DcmDspRoutineTidRef enabled
- ASCDCM-5578 Fixed known issue: The Dcm may become unresponsive after confirmation of response pending (RCRRP) transmission
- ASCDCM-5700 Fixed known issue: The Dcm might use incorrect Pdu MetaData information for TYPE2 periodic transmissions

Module version 5.0.8

2021-05-21

- ASCDCM-5475 Fixed known issue: Dcm suppresses the final response for an UDS request after RCRRP (0x78) response transmission
- ASCDCM-5524 Fixed known issue: Unnecessary routine APIs are generated if port interface is used for a routine with configuration parameter DcmDspRoutineTidRef enabled
- ASCDCM-5555 Out-of-bounds memory access may occur when the UDS service InputOutputControlByIdentifier (0x2F) is requested for a DID with no signals
- ASCDCM-5568 Fixed known issue: Compiler errors might occur when memory mapping is used
- Implemented UDS service `ReadScalingDataByIdentifier(0x24)`
- ASCDCM-5559 Fixed known issue: Out-of-bounds memory access may occur in case of a TesterPresent (0x3E) request or BusyRepeatRequest (0x21) NRC transmission when DcmDslEnableTxConfirmationNotification is set to true
- ASCDCM-5545 Fixed known issue: The Dcm does not clear ApplUpdated and ResponseRequired flags after a jump from Bootloader/ECUReset
- ASCDCM-5580 Fixed known issue: Compilation error occurs if the configuration of DCM does not contain any DID signal and service WriteDataByIdentifier (0x2E) is configured

- Updated Pdu metadata support for Dcm communication interfaces based on EcuC APIs with metadata routing to the lower layer.

Module version 5.0.7

2021-03-05

- ASCDCM-5440 Fixed known issue: Incorrect validity check for service DynamicallyDefineDataIdentifier (0x2C)
- ASCDCM-5429 Fixed known issue: The Dcm transmits NRC 0x11 for the OBD service 0x00 and the OBD services between 0x0B to 0x0F if these services are not supported
- ASCDCM-5440 Fixed known issue: Incorrect validity check for service DynamicallyDefineDataIdentifier (0x2C)
- ASCDCM-5419 Fixed known issue: Dcm transmits incorrect response after maximum number of RCRRP response transmissions for OBD service $09 - Request vehicle information
- ASCDCM-5453 Fixed known issue: Dcm transmits an incorrect response for OBD service $09 - Request vehicle information
- ASCDCM-5490 Fixed known issue: The Dcm does not handle configuration with infinite number of RCRRP (0x78) response pending transmissions correctly
- ASCDCM-5378 Fixed known issue: Undefined behavior might occur if the transmission of a RequestCorrectlyReceivedResponsePending negative response code is not confirmed
- Implemented Pdu metadata support for Dcm communication interfaces based on EcuC APIs without metadata routing to the lower layer.
- ASCDCM-5509 Fixed known issue: The Dcm makes a session transition and signals an application update after a jump from ECUReset

Module version 5.0.6

2021-02-12

- ASCDCM-5391 Fixed known issue: The Dcm ignores further requests after positive response to EcuReset (0x11) even for subfunctions which should not cause an actual reset
- Performance optimization regarding Dcm Endianess Conversion.
- ASCDCM-5413 Fixed known issue: The Dcm transmits negative response code RCRRP (0x78) for OBD services not supporting RCRRP negative response code
- ASCDCM-5464 Fixed known issue: The Dcm module does not compile if a DcmDslProtocolRow has the DcmDslProtocolID set to DCM_SUPPLIER_XX and Rte usage is enabled

- ASCDCM-5403 Fixed known issue: The Dcm does not transmit a final response if at least one ManufacturerNotification/SupplierNotification operation returns DCM_E_REQUEST_NOT_ACCEPTED after a RCRRP response transmission
- ASCDCM-5408 Fixed known issue: The Dcm does not transmit a final response if the buffer for the diagnostic response transmission becomes busy after a RequestCorrectlyReceivedResponsePending response transmission

Module version 5.0.5

2021-01-22

- ASCDCM-5206 Fixed known issue: Dcm does not ensure consistent event-related data when reading freeze frames or extended data records from a non-Elektrobit Dem
- Changed multiplicity of DcmDemClientRef to 0..1

Module version 5.0.4

2020-12-17

- ASCDCM-3918 Fixed known issue: The Dcm might not send a RequestCorrectlyReceivedResponsePending (0x7F SID 0x78) NegativeResponseCode in time when Protocol Preemption occurs
- ASCDCM-5313 Fixed known issue: Undefined behavior might occur if ResponseOnEvent or periodic transmissions are used
- ASCDCM-5293 Fixed known issue: The Dcm does not postpone the processing of a ROE event on one protocol while the Dcm is executing a request on another protocol
- ASCDCM-5301 Fixed known issue: Undefined behavior might occur upon incoming requests on different connections of the same protocol
- ASCDCM-5208 Fixed known issue: The Dcm module becomes unresponsive if ResponseOnEvent (0x86) external sub-service handler returns E_NOT_OK
- Added configuration warning "The sum of the DcmDspRoutineSignalPos and the DsmDspRoutineSignalLength when aligned to the next byte boundary shall be less than the buffer size used for the reception channel of the RoutineControl service configured."
- ASCDCM-5330 Fixed known issue: The Dcm may become unresponsive when a low priority protocol is preempted by a higher priority protocol
- ASCDCM-5318 Fixed known issue: The Dcm transmits an incorrect response for the UDS service TransferData (0x36)
- ASCDCM-5212 Fixed known issue: Undefined ECU behavior might occur if the UDS service ReadDataByIdentifier (0x22) is requested for a DID containing signals with SenderReceiver access

- ASCDCM-5147 Fixed known issue: Dcm provides wrong message context data to the Dcm_Application-TransmissionConfirmation() callout function for negative response messages BusyRepeatRequest (0x21) and ConditionsNotCorrect (0x22)
- Added configuration warning: "Protocols which have different types (OBD/UDS) and which belong to different stacks should not reference service tables with mixed OBD/UDS services, or the same service table".

Module version 5.0.3

2020-10-23

- ASCDCM-5295 Fixed known issue: The Dcm might reset to the default diagnostic session while processing an application-injected request in a non-default diagnostic session

Module version 5.0.2

2020-09-25

Module version 5.0.1

2020-08-28

- ASCDCM-5164 Fixed known issue: The Dcm module might not compile if the OBD services $03, $07, and $0A are configured
- ASCDCM-5197 Fixed known issue: The ECU might reset or block when a service is requested that requires a subservice but subservices are not enabled
- ASCDCM-5155 Fixed known issue: The Dcm sends an incorrect response when two ROE events are triggered one after another
- ASCDCM-5213 Fixed known issue: Dcm does not compile with OBD service Clear/Reset emission-related diagnostic information ($04) or UDS service ClearDiagnosticInformation (0x14) enabled and postcondition assertion checks disabled
- ASCDCM-5192 Fixed known issue: Undefined Dcm behavior due to release of unlocked exclusive area

Module version 5.0.0

2020-06-26

- Implemented OBD service Request Vehicle Information ($09) according to AUTOSAR 4.4.0 specification

Module version 4.16.2

2020-05-22

- Parallel service processing - DemFacade rework
- Parallel UDS and OBD processing/ASR4.3 Dcm/Dem interface with ClientID
- ASCDCM-5067 Fixed known issue: Service TransferData (0x36) ignores the maxNumberOfBlockLength returned by the previous RequestUpload (0x35) service response
- Implemented support for 0xFF00 DID acording to ISO-14229/2013.
- Omission of DTR data in service response when Dem_DcmGetDTRData() returns E_NOT_OK
- ASCDCM-5063 Fixed known issue: Service ReadDTCInformation (0x19) related macro is incorrectly generated
- Implemented OBD service Request Vehicle Information ($09) according to AUTOSAR 4.4.0 specification

Module version 4.16.1

2020-04-24

- ASCDCM-4775 Fixed known issue: The Dcm still accepts further service requests after answering positively to service ECUReset (0x11)
- ASCDCM-4977 Fixed known issue: Session switch does not succeed after positive OBD response
- ASCDCM-5027 Fixed known issue: Dcm provide an incomplete protection against double type definition of type Dcm_DidSupportedType
- ASCDCM-4964 Fixed known issue: Dcm transmits an incorrect response for OBD service $09 - Request vehicle information

Module version 4.16.0

2020-03-25

Module version 4.15.2

2020-02-21

- ASCDCM-4742 Fixed known issue: Dcm transmits not supported NRCs requestCorrectlyReceivedResponsePending (0x78) and generalReject (0x10) for OBD services $03, $07, and $0A when its service processing needs additional time to finish

- ASCDCM-4640 Fixed known issue: Compilation error occurs when service InputOutputControlByIdentifier (0x2F) is configured and no DIDs are present
- ASCDCM-4641 Fixed known issue: The Dcm can enter a non-default session while processing an OBD request and perform another unexpected reset to the Default Session
- ASCDCM-4855 Fixed known issue: The DCM blocks the protocol on which a related diagnostic DID service is requested after a cancellation of an InputOutputControlByIdentifier (0x2F) service request

Module version 4.15.1

2020-01-24

- ASCDCM-4710 Fixed known issue: Compiler errors occur when no DcmDslSecurityLevels are configured
- ASCDCM-4709 Fixed known issue: The DCM is not able to receive requests after protocol preemption
- ASCDCM-4717 Fixed known issue: Compilation error occurs when OBD service Request Control of On-Board System, Test or Component (0x08) is configured and no Request Control configuration is present
- Restrict internal support for subservices 0x03 and 0x43 belonging to ResponseOnEvent (0x86) from configuration
- ASCDCM-4720 Fixed known issue: The Dcm does not reset the security level when starting a protocol.
- Implemented sub-services reportNumberOfDTCBySeverityMaskRecord(0x07), reportDTCBySeverityMaskRecord(0x08), reportSeverityInformationOfDTC(0x09) for UDS Service ReadDTCInformation (0x19) according to AUTOSAR 4.4.0 specifications.
- ASCDCM-4748 Fixed known issue: Dcm transmits an incorrect response for OBD service $04 Clear/reset emission-related diagnostic information
- ASCDCM-4719 Fixed known issue: The Dcm may become unresponsive as a result of protocol preemption
- ASCDCM-4777 Fixed known issue: Out of bounds memory access occurs if the services WriteMemoryByAddress (0x3D), RequestDownload (0x34), and RequestUpload (0x35) are used

Module version 4.15.0

2019-12-13

- Parallel UDS and OBD processing/ASR4.3 Dcm/Dem interface with ClientID

Module version 4.14.23

2019-12-06

- ASCDCM-4667 Fixed known issue: Compilation error occurs when service ReadDataByPeriodicIdentifier(0x2A) is configured and no DIDs are present

Module version 4.14.22

2019-10-25

- ASCDCM-4577 Fixed known issue: The Dcm transmits an incorrect response for UDS service TransferData (0x36) requests with repeated blockSequenceCounter
- ASCDCM-4626 Fixed known issue: Dcm module Generation fails when no Read Memory Ranges are defined and DcmDspMemoryRangeHighNotIncluded parameter is set to TRUE
- ASCDCM-3544 Implemented Generic End Of Line addon for Data Identifiers for Service ReadDataByIdentifier (0x22), Service WriteDataByIdentifier (0x2E) and InputOutputControlByIdentifier (0x2F)
- ASCDCM-4572 Fixed known issue: Dcm sends an incorrect positive response if paged-buffer mechanism is used
- ASCDCM-4716 Fixed known issue: Out of bounds memory access occurs for OBD services $03, $07 and $0A

Module version 4.14.21

2019-09-06

- ASCDCM-4512 Fixed known issue: Signal data corruption occurs when processing a periodic transmission in parallel with a DID read operation
- ASCDCM-4514 Fixed known issue: Null pointer dereference occurs when the ReadDataByPeriodicIdentifier (0x2A) service is executed
- ASCDCM-4509 Fixed known issue: The Dcm transmits an incorrect response when the UDS service WriteDataByIdentifier (0x2E) is requested for not supported DIDs in the OBD range
- ASCDCM-4521 Fixed known issue: The Dcm calls the ReturnControlToEcu asynchronous operation with an invalid OpStatus

Module version 4.14.20

2019-08-09

- ASCDCM-4470 Fixed known issue: Requests received shortly after a positive response caused by a WARM_START condition may be rejected

- ASCDCM-4477 Fixed known issue: RequestTransferExit behaves unexpected when DcmDataTransferServicesASRVersion is set to 'AUTOSAR_43' or 'AUTOSAR_422'
- ASCDCM-4478 Fixed known issue: Dcm does not generate if its default service API is different from Dem's default service API
- ASCDCM-4480 Fixed known issue: Dcm does not generate when DcmConfigSet default name (DcmConfigSet_0) is changed

Module version 4.14.19

2019-07-12

- ASCDCM-4149 Fixed known issue: Dcm does not generate A2L files
- ASCDCM-4469 Fixed known issue: Dem_DcmGetAvailableOBDMIDs() function is called also for non-availability OBDMIDs

Module version 4.14.18

2019-06-17

- Increased maximum number of RequestCorrectlyReceivedResponsePending (0x78) NRCs
- ASCDCM-4355 Fixed known issue: Reading DID ranges may overrun the response message buffer
- ASCDCM-4367 Fixed known issue: TYPE1 periodic responses keep the Dcm in a non-default session
- ASCDCM-4379 Fixed known issue: The Dcm stays in a non-default session indefinitely when ResponseOnEvent or PeriodicDID transmissions are used
- Implemented array type signals for the UDS service RoutineControl (0x31) according to AUTOSAR 4.3.0
- Implement Generic End Of Line handling for Routine Control service (0x31) when an unsupported (unconfigured / unused) routine is requested
- Implement Routine Info Byte handling for Routine Control service (0x31) when a byte from the application is requested to be added on the response
- ASCDCM-4401 Fixed known issue: In `Dcm_CommunicationServices.c` Incorrect response data is transmitted if buffers over 65535 bytes in length are used
- ASCDCM-4402 Fixed known issue: In `Dcm_CommunicationServices.c` Incorrect request data is received if buffers over 65535 bytes in length are used

Module version 4.14.17

2019-05-24

- Added support for configuration parameter `DcmDspDidControlMask`. Values 1, 2, 3 and 4 are supported for parameter `DcmDspDidControlMaskSize`. The value `DCM_CONTROLMASK_EXTERNAL` is supported for parameter DcmDspDidControlMask. Values configured for `DcmDspDidControlMask` and `DcmDspDidControlMaskSize` have effect only upon the interface signatures of DcmDspData elements referenced by the DcmDspDid which references the `DcmDspDidInfo` containing the previously-mentioned `DcmDspDidControlMask` and `DcmDspDidControlMaskSize` parameters
- Implemented generic interface to inject service requests from application software

Module version 4.14.16

2019-05-16

- ASCDCM-4360 Fixed known issue: Compiler abstraction in declaration of Xxx_WriteData and Xxx_ShortTermAdjustment interface functions

Module version 4.14.15

2019-04-18

- ASCDCM-4217 Fixed known issue: Requests for the ShortTermAdjustment (0x03) operation of service InputOutputControlByIdentifier (0x2F) are not handled correctly
- ASCDCM-4241 Fixed known issue: Missing AUTOSAR compiler abstraction for pointers defined in type definitions or defined as function parameters

Module version 4.14.14

2019-04-11

- ASCDCM-4274 Fixed known issue: Invalid comments in parameter documentation
- ASCDCM-4229 Fixed known issue: Rte client/server interface operation call is executed from the context of the Dcm_TpRxIndication() API

Module version 4.14.13

2019-03-22

- ASCDCM-4232 Fixed known issue: Undefined behavior when using Sender/Receiver interfaces for DcmDspData DID signals

- Project specific service callbacks interfaces ASR 4.2.2 compliant for WriteMemory (0x27) and ProcessRequestTransferExit (0x32)

Module version 4.14.12

2019-02-18

- ASCDCM-4183 Fixed known issue: Compilation fails if DcmDspMemoryRangeHighNotIncluded is set to true
- ASCDCM-4204 Fixed known issue: Compiler errors occur when a DcmDspWriteMemoryRangeByLabelInfo is configured and no DcmDspWriteMemoryRangeInfo and no DcmDspReadMemoryRangeByLabelInfo are configured
- ASCDCM-4214 Fixed known issue: Undefined behavior might occur when a request for the UDS service TransferData (0x36) is received
- ASCDCM-4216 Fixed known issue: CONST variables of RoutineControl (0x31) service implementation are not defined in an appropriate MemMap section

Module version 4.14.11

2019-01-24

- ASCDCM-4098 Fixed known issue: The Dcm sends a positive response if subfunction clearDynamicallyDefinedDataIdentifier (0x03) of service DynamicallyDefineDataIdentifier (0x2C) is requested for an unconfigured dynamically defined DID
- ASCDCM-4145 Fixed known issue: Dcm accesses wrong data when DynamicallyDefineDataIdentifier service is requested with a DID that is outside the dynamic range
- ASCDCM-4102 Fixed known issue: The Dcm may become unresponsive if the application requests a reset to the default session too often
- ASCDCM-4175 Fixed known issue: Periodic DIDs are not transmitted in the same order in which they were configured
- Enable DCM internal service processing in case of project specific service handling
- Implementation of selectable NRC behaviour for unsupported OBD services.

Module version 4.14.10

2018-11-23

- ASCDCM-4025 Fixed known issue: Compiler errors occur when a DcmDslRoeConnection is configured and referenced, and the internally-managed ResponseOnEvent (0x86) service handler is not used

- ASCDCM-4142 Fixed known issue: A code generation error occurs if DcmDspWriteMemoryRangeInfo or DcmDspWriteMemoryRangeByLabelInfo are configured and neither DcmDspReadMemoryRangeInfo nor DcmDspReadMemoryRangeByLabelInfo are configured

Module version 4.14.9

2018-10-26

- Implement support for configuration parameter DcmDspMemoryRangeHighNotIncluded to allow for more flexibility in the definition of memory ranges
- Implemented storage of output signals data between service handler calls and improved memory usage for UDS service RoutineControl (0x31)
- ASCDCM-4041 Fixed known issue: The ECU may reset during the transmission of a RequestCorrectlyReceivedResponsePending (0x78) NegativeResponseCode when the DiagnosticSessionControl (0x10) service is used with a session marked as either DCM_OEM_BOOT or DCM_SYS_BOOT
- ASCDCM-4039 Fixed known issue: Externally managed OBD service "Request on-board monitoring test results for specific monitored systems" ($06) causes compile error
- TxConfirmation callout with information for positive response suppression (ID10)

Module version 4.14.8

2018-09-28

- ASCDCM-4021 Fixed known issue: Wrong mode declaration string RTE_MODE_DcmEcuReset_KEYOFFON causes compile error
- ASCDCM-3998 Fixed known issue: Missing AdjacentLayer.properties information for TYPE2 ROE connections
- ASCDCM-4010 Fixed known issue: When DiagnosticSessionControl (0x10) service is used with a boot session the message "0x50" is transmitted incorrectly instead of a positive response
- ASCDCM-4028 Fixed known issue: Dcm does not compile if service RequestVehicleInformation ($09) is enabled without configuration data
- Notification callout on S3Timeout (ID7)

Module version 4.14.7

2018-08-24

- Add support for short response on sub-function 0x01 - onDTCStatusChange for the UDS service ResponseOnEvent (0x86)

- ASCDCM-3980 Fixed known issue: TesterPresent request with the suppressPosRspMsgIndicationBit set (0x3E 0x80) on a physical channel may result in a positive response
- ASCDCM-3950 Fixed known issue: The Dcm does not transmit a negative response for subsequent UDS service TransferData (0x36) requests which are not consistent with the UDS service RequestDownload (0x34) service requests

Module version 4.14.6

2018-07-27

- ASCDCM-3944 Fixed known issue: Undefined Dcm behavior when endianness conversion is enabled
- ASCDCM-3897 Fixed known issue: The Dcm becomes unresponsive when Dcm_Init() is called before Rte_Start()

Module version 4.14.5

2018-06-22

- ASCDCM-4188 Fixed known issue: Function declarations of TesterPresent (0x3E) service implementation are not inside a CODE section
- ASCDCM-4189 Fixed known issue: CONST variables of service request notification implementation are not declared in an appropriate MemMap section
- ASCDCM-4190 Fixed known issue: Definition of local pointers without AUTOSAR compiler abstraction
- ASCDCM-4191 Fixed known issue: Variable of RoutineControl (0x31) service implementation is not declared and defined in an appropriate MemMap section
- Implement usage of the HandleId wizard for DcmDslProtocol containers
- Support for BSW Distribution - Dcm and ComM on different partitions (cores)

Module version 4.14.4

2018-05-25

- ASCDCM-3867 Fixed known issue: Compilation errors are generated when DET is enabled and services WriteMemoryByAddress (0x3D) or RequestDownload (0x34) are configured, but ReadMemoryByAddress (0x23), DynamicallyDefineDataIdentifier (0x2C) or RequestUpload (0x35) are not configured

Module version 4.14.3

2018-05-07

- Implemented Sender/Receiver access for `DcmDspData` for UDS services ReadDataByIdentifier (0x22) and WriteDataByIdentifier (0x2E).
- Implemented endianness interpretation for `DcmDspData` with Sender/Receiver access for UDS services ReadDataByIdentifier (0x22) and WriteDataByIdentifier (0x2E) according to AUTOSAR 4.3 specifications
- ASCDCM-3665 Fixed known issue: Out-of-bounds memory access may occur when the UDS service ReadDataByIdentifier (0x22) is requested for a DID with signals with sender/receiver access

Module version 4.14.2

2018-04-20

- ASCDCM-3767 Fixed known issue: An already busy TYPE2 periodic transmission protocol accepts additional requests - processing timeout possible
- ASCDCM-3774 Fixed known issue: The TYPE2 periodic transmission protocol is blocked when the PduR rejects the periodic transmission

Module version 4.14.1

2018-03-19

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

Module version 4.14.0

2018-03-16

- ASCDCM-3693 Fixed known issue: Inconsistency between calls to ComM_DCM_ActiveDiagnostic() and ComM_DCM_InactiveDiagnostic() when more than one RxPduID of one protocol is used
- ASCDCM-3730 Fixed known issue: Memory section conflict between declaration and definition.
- ASCDCM-3686 Fixed known issue: Incorrect values may be sent for Periodic DIDs
- Memory range configuration of Services 0x23/0x3D with labels
- Added memory sections for the DCM software component description

Module version 4.13.2

2018-02-23

- ASCDCM-3715 Fixed known issue: Invalid type references generated when DID Ranges are used with Rte
- ASCDCM-3692 Fixed known issue: The ECU becomes unresponsive when the CommunicationControl (0x28) service is configured and the DiagnosticSessionControl (0x10) service is requested
- ASCDCM-3714 Fixed known issue: The Dcm might not compile if OBD services are enabled
- ASCDCM-3724 Fixed known issue: Dcm does not link if variable Dcm_BootloaderRequestToRespondTo is not assigned to RAM memory section
- ASCDCM-3694 Fixed known issue: The Dcm performs actions pertaining to a reset to the Default Session outside of normal reset-to-Default-Session circumstances
- ASCDCM-3721 Fixed known issue: Dcm wrongly resets the current communication status when the DiagnosticSessionControl (0x10) UDS service is requested
- ASCDCM-3725 Fixed known issue: DCM enable DTCSetting when the DiagnosticSessionControl (0x10) service is requested
- ASCDCM-3702 Fixed known issue: Rte generation error when asynchronous service execution is enabled for the RoutineControl (0x31) service
- ASCDCM-3706 Fixed known issue: The Dcm executes an unexpected reset to the Default Session
- ASCDCM-3708 Fixed known issue: The S3 timer is not restarted upon reception of a concurrent TesterPresent (0x3E 0x80) request
- ASCDCM-3709 Fixed known issue: A Precondition Assert is triggered in the conditions in which a BusyRepeatRequest (0x21) NegativeResponseCode is transmitted

Module version 4.13.1

2017-12-15

- ASCDCM-3618 Fixed known issue: TYPE1 Periodic DID transmissions may not be possible from a protocol that previously received a request which resulted in a suppressed transmission
- ASCDCM-3633 Fixed known issue: The XPath checker displays an incorrect warning message for the entries of DcmModeRule container
- ASCDCM-3626 Fixed known issue: Handle unexpected values returned by the Xxx_WriteDidData interface function.
- ASCDCM-3620 Fixed known issue: Return value E_NOT_OK of DID range read operation Xxx_ReadDidData() is not handled correctly
- ASCDCM-3631 Fixed known issue: Unsorted DID ranges are not correctly processed
- ASCDCM-3636 Fixed known issue: Incorrect application of the controlEnableMaskRecord in `InputOutputControlByIdentifier (0x2F)` requests
- ASCDCM-3635 Fixed known issue: DID Range Read operation Xxx_ReadDidData() is invoked with an incorrect value of the OpStatus parameter

- ASCDCM-3619 Fixed known issue: Reentrancy of the Xxx_ReadDataLength() and Xxx_ConditionCheckRead() operations if the InputOutputControlByIdentifier (0x2F) service is handled asynchronously and is requested at the same time as a periodic DID readout times out

Module version 4.13.0

2017-11-17

- ASCDCM-3540 Fixed known issue: TYPE1 Periodic DID transmissions are not possible from a protocol which has returned an NRC to a previous service request processing
- ASCDCM-3527 Fixed known issue: TYPE1 ROE transmissions on a different protocol than the configuring protocol do not change the session to the default session
- ASCDCM-3526 Fixed known issue: TYPE1 Periodic DID transmissions on a different protocol than the configuring protocol may occur and this does not change the session to the default session
- ASCDCM-1592 Fixed known issue: Endianness conversion is performed regardless of interface type for DcmDspData signals when reading, writing, or controlling DIDs. This is fulfilled by allowing the user to select or deselect endianness conversion to be performed only on Routine signals, only on DID signals, or both.
- ASCDCM-3528 Fixed known issue: A precondition assert is reached when TYPE1 periodic DID transmissions or TYPE1 ResponseOnEvent (0x86) transmissions are used and XXX_StartProtocol() operations fail
- ASCDCM-3606 Fixed known issue: The Dcm transmits a wrong response when ReadMemoryByAddress (0x23) and WriteMemoryByAddress (0x3D) are requested with memoryAddress and memorySize equal to 0

Module version 4.12.19

2017-10-04

- Removed AUTOSAR 3.x compliant symbolic name value macros and updated the logic to only provide AUTOSAR 4.0.2 compliant macros if macro DCM_PROVIDE_LEGACY_SYMBOLIC_NAMES is defined
- ASCDCM-3488 Fixed known issue: The Dcm accepts requests on RxPduIDs belonging to a MainConnection on which a BusyRepeatRequest (0x21) transmission is ongoing
- Optimized speed for configuration verifier checks for DID and PID configuration.
- ASCDCM-3503 Fixed known issue: Dcm generation fails when a DID signal has variable length and does not have the last position in the DID's DcmDspDidSignal list
- ASCDCM-3500 Fixed known issue: NegativeResponseCodes are incorrectly sent instead of periodic DIDs when the `ReadDataByPeriodicIdentifier (0x2A)` service is used
- ASCDCM-3523 Fixed known issue: DCM does not generate a link to subfunction handler for subservice 0x00 for `TesterPresent (0x3E)` service

- ASCDCM-3535 Fixed known issue: A compiler warning "unused parameter" is generated if all DcmDslPeriodicTransmissions lack DcmDslPeriodicConnection entries
- ASCDCM-3502 Fixed known issue: Dcm generates ports for DID range operations that are not required
- ASCDCM-3515 Fixed known issue: Reentrancy of the Xxx_ReadDataLength() and Xxx_ConditionCheckRead() operations if the `ReadDataByIdentifier (0x22)` service is handled asynchronously and is requested at the same time as a periodic DID readout times out
- ASCDCM-3533 Fixed known issue: Incorrect responses when EndiannessConversion is used and a DID is read

Module version 4.12.18

2017-08-25

- Provided forward-declarations in Dcm_Cbk.h for Dcm_ComM_NoComModeEntered(), Dcm_ComM_SilentComModeEntered() and Dcm_ComM_FullComModeEntered() as per AUTOSAR 4.2 requirement SWS_Dcm_01066
- ASCDCM-3470 Fixed known issue: Incorrect signature of operation RequestResults of ClientServerInterface RoutineServices_RoutineName when an AUTOSAR 4.0.3 interface is used
- ASCDCM-3492 Fixed known issue: Compilation errors and warnings occur when attempt counter is set to reset after timeout but is not configured to be handled externally
- Allow configuration of external subfunction handlers for UDS service `TesterPresent (0x3E)`. Adapted NRC sequence of UDS service `TesterPresent (0x3E)` to ISO 14229-1:2013 requirements.
- ASCDCM-3479 Fixed known issue: The Dcm may transmit an incorrect response to a SecurityAccess (0x27) request

Module version 4.12.17

2017-07-28

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

Module version 4.12.16

2017-07-21

- ASCDCM-3433 Fixed known issue: Malformed positive response to service `ECUReset (0x11)` upon reset.
- Implemented support for security access delay timers according to AUTOSAR 4.3

- ASCDCM-4773 Fixed known issue: The AttemptCounter of a security level is reset and SecurityAccess(0x27) request sequences can be made without DcmDspSecurityDelayTime time-out to pass

Module version 4.12.15

2017-06-30

- Implemented subfunctions enableRxAndDisableTxWithEnhancedAddressInformation (0x04) and enableRxAndTxWithEnhancedAddressInformation (0x05) of service `CommunicationControl(0x28)`.

Module version 4.12.14

2017-06-02

- ASCDCM-3372 Fixed known issue: DCM does not store the correct state of communication modes related to ComMChannels if service `CommunicationControl (0x28)` is called and the mode request is rejected by BswM
- ComM Channels referenced in DcmDslProtocolRx are reset to the default communication mode when the DcmDspComControlCommunicationReEnableModeRuleRef is not fulfilled anymore and no DcmDspComControlSpecificChannel and DcmDspComControlAllChannel are configured.
- Improved generation of swcd and template code when using a huge amount of DID Data. This leads to a drastic decrease of generation time.
- ASCDCM-3092 Fixed known issue: Dcm module might block or behave unexpectedly if concurrent requests are received from two or more clients under rare timing conditions.

Module version 4.12.13

2017-04-28

- ASCDCM-3295 Fixed known issue: Out-of-bounds memory access occurs when using EndiannessConversion and performing a ReadDataByIdentifier (0x22) request
- ASCDCM-3239 Fixed known issue: The ECU may block or reset if the UDS service `CommunicationControl (0x28)` is requested for a ComMChannel which is not referenced by any DcmDslProtocolRxComMChannelRef which belongs to any RxConnection
- ASCDCM-3307 Fixed known issue: Wrong responses or NvM Block data corruption when OBD services Request current powertrain diagnostic data ($01) or Request powertrain freeze frame data ($02) are used and DIDs which contain NvM BlockID signals are configured

- ASCDCM-3349 Fixed known issue: Inconsistency between advertised state and internal state of communication control when UDS service `CommunicationControl (0x28)` is processed asynchronously and the mode rule for communication control reset fails
- Allowed configuration of DIDs within the 0x0000 - 0x00FF range. Allowed the usage of services `ReadDataByIdentifier (0x22)` and `WriteDataByIdentifier (0x2E)` for DIDs in this range.

Module version 4.12.12

2017-03-31

- ASCDCM-3298 Fixed known issue: Dcm might not compile if DcmDspDids are configured but for none of them at least one signal DcmDspDidSignal is configured
- ASCDCM-3303 Fixed known issue: Undefined behavior occurs when the ComM module changes the mode of a ComM channel at the same time that a Periodic DID that was configured on an RxPduID which references the same ComM channel times out
- ASCDCM-3185 Fixed known issue: Dcm does not obey configured P2* timing behavior and ignores the maximum number of sent RequestCorrectlyReceivedResponsePending (0x78) NRCs
- Added new configuration parameter `DcmDspDiagSesRespMaxNumRespPend`. This offers more flexibility as it allows individual configuration of the maximum allowed number of RequestCorrectlyReceivedResponsePending (0x78) NRCs per request for each individual diagnostic session.

Module version 4.12.11

2017-03-22

- ASCDCM-3254 Fixed known issue: The Dcm does not compile if UDS service `RequestUpload (0x35)` is configured and internally-managed and neither UDS service `RequestDownload (0x36)` nor `WriteMemoryByAddress (0x3D)` are configured and internally-managed
- ASCDCM-3286 Fixed known issue: Dcm might not compile if the UDS service `DynamicallyDefineDataIdentifier (0x2C)` is configured and internally-managed
- ASCDCM-3290 Fixed known issue: Undefined behavior if the callout Dcm_ReadMemory() returns DCM_READ_FORCE_RCRRP when dynamically defined DIDs are read using the UDS service `ReadDataByIdentifier (0x22)`
- ASCDCM-3310 Fixed known issue: Dcm might not compile if the UDS service `RequestUpload (0x34)` is configured and internally-managed
- ASCDCM-3287 Fixed known issue: Incoming receptions are cancelled upon completion of a reception which can be processed
- ASCDCM-3294 Fixed known issue: The Dcm may become unresponsive as a result of protocol preemption

- ASCDCM-3306 Fixed known issue: Requests for the `ReadDTCInformation (0x19)` service with subfunctions reportFirstTestFailedDTC (0x0B), reportFirstConfirmedDTC (0x0C), reportMostRecentTestFailedDTC (0x0D), or reportMostRecentConfirmedDTC (0x0E) with an incorrect length are accepted

Module version 4.12.10

2017-03-03

- ASCDCM-3257 Fixed known issue: NullPointer dereference when a request for an externally-managed `DiagnosticSessionControl(0x10)` service handler which contains just the ServiceID byte is received
- Enabled the UDS service WriteDataByIdentifier (0x22) to write DIDs in the OBD DIDs interval 0xF400-0xF8FF
- ASCDCM-3282 Fixed known issue: Service `Read Generic Information (0xAF)` produces wrong NRC when request exceeds required length
- ASCDCM-3281 Fixed known issue: Dcm might not stop the S3 timer timeout even though diagnostic processing is ongoing
- ASCDCM-3256 Fixed known issue: Compiler warning or error due to signature mismatch of Dcm_RxIndication() API
- Implemented the following services according to AUTOSAR 4.3.0.
  - `RequestDownload (0x34)`
  - `RequestUpload (0x35)`
  - `TransferData (0x36)`
  - `RequestTransferExit (0x37)`
  - `ReadMemoryByAddress (0x23)`
  - `WriteMemoryByAddress (0x3D)`

Module version 4.12.9

2017-02-03

- ASCDCM-3186 Fixed known issue: A compilation error occurs due to mismatching memory mapping sections on the definition and the declaration of a data structure on the DCM
- ASCDCM-3214 Fixed known issue: Compiler errors are reported if service `ReadDataByPeriodicIdentifier (0x2A)` is configured, DcmDspDDDIDcheckPerSourceDID equals "true" and service `DynamicallyDefineDataIdentifier (0x2C)` is not configured
- ASCDCM-3226 Fixed known issue: A code generation error occurs if DcmModeDeclarationSupport is set to true and the DcmDspSession container contains more than one entry

- For periodic DID (0xF200..0xF2FF) allow the access types 'write' and 'I/O control', additional to the existing 'read'.

Module version 4.12.8

2017-01-05

- ASCDCM-3180 Fixed known issue: Dcm writes out of bounds and rejects requests after a jump from bootloader with `ResponseRequired` set to FALSE

Module version 4.12.7

2016-12-14

- ASCDCM-3129 Fixed known issue: An unreachable code assert occurs if OBD services 0x03, 0x07 or 0x0A are used
- The NRC handling for RoutineControl(0x31) service is made according with selected ISO(ISO_14229-2006/ISO_14229-2006).
- ASCDCM-3141 Fixed known issue: Events set with service `ResponseOnEvent (0x86)` to be persistently in the Stopped State with sub-service stopResponseOnEvent and the StorageState set to "storeEvent" (0x40), are erroneously re-started after an ECU reset and a Session Change.
- ASCDCM-3124 Fixed known issue: Incorrect responses to requests of service `ReadDTCInformation (0x19)` as a result of multiple calls to the Dem_GetNumberOfFilteredDTC() API
- ASCDCM-3155 Fixed known issue: Periodic transmission packets may be lost after removing or reconfiguring periodic DIDs
- ASCDCM-3164 Fixed known issue: The DCM produces redundant DET error notifications for channels which are not configured for the DCM

Module version 4.12.6

2016-11-04

- Changed all defensive programming configuration and functionality to `Dcm_DefProg_Cfg.h`
- ASCDCM-3064 Fixed known issue: Incorrectly named ports are generated for the ServiceRequestNotification ClientServerInterface
- Implemented subfunction reportFirstTestFailedDTC (0x0B) of UDS service `ReadDTCInformation (0x19)`

- Implemented subfunction reportFirstConfirmedDTC (0x0C) of UDS service `ReadDTCInformation (0x19)`
- Implemented subfunction reportMostRecentTestFailedDTC (0x0D) of UDS service `ReadDTCInformation (0x19)`
- Implemented subfunction reportMostRecentConfirmedDTC (0x0E) of UDS service `ReadDTCInformation (0x19)`
- ASCDCM-3085 Fixed known issue: If a request of service `WriteDataByIdentifier (0x2E)` is performed and DcmDslProtocolEndiannessConvEnabled is set to true, write data in the DID may be corrupted
- ASCDCM-3079 Fixed known issue: Asynchronous ReadDataLength operations are always called with a DCM_INITIAL OpStatus when an AUTOSAR 4.2.1 interface is used
- ASCDCM-3080 Fixed known issue: Compiler error if OBD service `RequestOnboardMonitoringTestResults (0x06)` is enabled and DcmOBDMIDSupportType is configured to DCM_OBDMID_SUPPORT_DEM
- Implemented TYPE2 transmission for UDS service `ResponseOnEvent(0x86)`.
- ASCDCM-3090 Fixed known issue: Unexpected behavior due to reentrancy of Dem_EnableDTCSetting
- ASCDCM-3051 Fixed known issue: The Dem_EnableDTCSetting() API is called on diagnostic session transition / referenced ModeRule failure even if DTCSetting is already enabled

Module version 4.12.5

2016-10-07

- Adapted resource file for the scheduling of main functions to the split of `IpduM_MainFunction()` into `IpduM_MainFunctionRx()` and `IpduM_MainFunctionTx()`.
- ASCDCM-3036 Fixed known issue: UDS service `ReadDataByIdentifier(0x22)` does not handle requests for OBD availability MIDs/PIDs/VehInfos correctly
- ASCDCM-3005 Fixed known issue: Compiler errors and warnings are issued when service `ReadGenericInformation (0xAF)` is configured and service `ReadDTCInformation (0x19)` is not configured.
- Implemented subfunction OnDTCStatusChange (0x01) of service `ResponseOnEvent (0x86)` and related functionality.
- ASCDCM-3054 Fixed known issue: Dcm does not respond correctly when OBD Service 0x01 is enabled and PID 0x00 is requested using `ReadDataByIdentifier (0x22)` service.

Module version 4.12.4

2016-09-09

- ASCDCM-2998 Fixed known issue: An out-of-bounds memory access occurs if the same DcmDspSubnetNumber is configured twice and is used in a request for service `CommunicationControl (0x28)`
- Implemented subfunction ClearResponseOnevent (0x06) of service `ResponseOnEvent(0x86)`
- ASCDCM-2989 Fixed known issue: Incorrect security level checks and calls to application functions are performed upon protocol preemption
- ASCDCM-3010 Fixed known issue: Service handlers may be called with a DCM_CANCEL OpStatus after completion of operation
- Improved NRC handling for `RequestDownload (0x34)` service.
- Improved NRC handling for `RequestUpload (0x35)` service.
- Improved NRC handling for `TransferData (0x36)` service.
- Improved NRC handling for `RequestTransferExit (0x37)` service.
- ASCDCM-3014 Fixed known issue: Buffer corruption and undefined behavior may occur as a consequence of the transmitting protocol being stopped prematurely in the case of TYPE2 periodic transmissions
- ASCDCM-3005 Fixed known issue: The ComMChannel on which a diagnostic request is received might not go to or remain in FullCommunicationMode

Module version 4.12.3

2016-08-05

- ASCDCM-2407 Fixed known issue: Cancellation of pending receptions is not done correctly
- Configuration Signature
- Implemented full support for eventWindowTime for service `ResponseOnEvent(0x86)`
- Diagnostic session checks are performed on subfunctions of service `DiagnosticSessionControl` too

Module version 4.12.2

2016-07-22

- ASCDCM-2876 Fixed known issue: `Dcm_Cfg.h` include many spaces in one line
- ASCDCM-2870 Fixed known issue: ROE preconfiguration data is not initialized correctly and/or memory corruption occurs when persistent ROE settings are used
- ASCDCM-2837 Fixed known issue: Compiler errors occur when the EB Dcm is integrated with other vendor's basic software modules
- ASCDCM-2867 Fixed known issue: ROE events are wrongly re-activated when re-entering the default session when they are stopped outside the default session

- ASCDCM-2959 Fixed known issue: ROE events triggered outside FullCommunicationMode are wrongly queued and processed later for transmission

Module version 4.12.1

2016-07-01

- Implemented a subset of UDS service `0x86 (ResponseOnEvent)` according to AUTOSAR 4.2 specification. The implemented subfunctions are: 0x00 stopResponseOnEvent, 0x04 reportActivatedEvents, 0x05 startResponseOnEvent for preconfigured events of OnChangeOfDataIdentifier type with Infinite(0x02) eventWindowTime.
- Implemented separate ROE Service subfunctions taking in consideration the StorageState bit (sub-function bit 6). The implementation is based on Bugzilla RfC #72061.
- ASCDCM-2717 Fixed known issue: The Dcm might not correctly respond with NRC 0x31 when reading multiple DIDs
- ASCDCM-2737 Fixed known issue: The Dcm maps the wrong security level to ReadMemoryRangeInfo and WriteMemoryRangeInfo elements in case of one of them references the same security level multiple times
- ASCDCM-2747 Fixed known issue: Unresolved possible error reference in `Dcm_swc_interfaces.arxml` lead to warnings during import
- ASCDCM-2756 Fixed known issue: Session checks could be incorrect for subservices of the service functions with an ID greater than 0x3E, if any session is configured for any subservice for the UDS service `TesterPresent (0x3E)`
- ASCDCM-2782 Fixed known issue: Final negative responses may not be produced after issuing a RequestCorrectlyReceivedResponsePending (0x78) NRC for requests received on functional RxPduIDs
- All combinations of signal configurations for all RoutineControl (0x31) operations are now available

Module version 4.12.0

2016-06-03

- The dynamically defined DID source table NvM block is marked to be stored if it has been restored to default after read block error during init.
- Add support for disabling ECU keep awake through diagnosis after ignition was turned off.
- ASCDCM-2701 Fixed known issue: ComM_DCM_ActiveDiagnostic(NetworkId) is not invoked when the diagnostic session is changed into a session different than the default session
- ASCDCM-2675 Fixed known issue: Generation errors if ClientServer interfaces are used for DcmDspRoutines, RequestResults IN-signals have variable length, and StopRoutine IN-signals have fixed length or do not exist.

- ASCDCM-2689 Fixed known issue: The Dcm might trigger a Det error when dynamically defined DIDs from NvM are initialized.
- ClientServer interfaces for RequestControlServices are now generated only when DcmDspRequestControl.DcmDspDidRangeUsePort is TRUE
- NRC 0x36 (exceededNumberOfAttempts) for `SecurityAccess service (0x27)` is transmitted one time, when SendKey subfunction is requested with an invalid key and the maximum number of failed access attempts is reached.
- ASCDCM-2674 Fixed known issue: The Dcm might trigger an Os error or block the Ecu when reading a DID with a memory source
- ASCDCM-2681 Fixed known issue: The Dcm might violate the order of calling memory callouts when multiple requests are processed
- ASCDCM-2738 Fixed known issue: OBD services Request current powertrain diagnostic data ($01) and Request powertrain freeze frame data ($02) are disabled when they are configured in more than one diagnostic services table
- ASCDCM-2743 Fixed known issue: UDS service `DynamicallyDefineDataIdentifier (0x2C)` may not transmit NRC RequestOutOfRange (0x31) for invalid (not supported) source DIDs in the OBD range

Module version 4.11.2

2016-04-29

- ASCDCM-2581 Fixed known issue: An incorrect warning message is issued when OBD DIDs are configured
- Add selectable removal from generated ClientServerInterface operations of unused IOControl operations
- Preserve data at buffer pointers provided to DID operations across asynchronous operation call sequences
- ASCDCM-2521 Fixed known issue: Signal data corruption due to race condition when processing periodic DID transmission in parallel with any DID read/write/control operation
- Implemented optimization of DID data configuration code
- ASCDCM-2439 Fixed known issue: The ECU may become unresponsive if a DTC-related OBD service is executed immediately after executing the `ReadDTCInformation (0x19)` or `ReadGenericInformation (0xAF)` service
- ASCDCM-3018 Dcm compilation fails when defining external reportDTCSnapshotRecordByDTCNumber (0x04) or reportDTCExtDataRecordByDTCNumber (0x06) sub-functions for the `ReadGenericInformation (0xAF)` service

Module version 4.11.1

2016-04-01

- Updated #defines for symbolic name values related to ROE according to AUTOSAR 4.0.3 naming schema
- ASCDCM-2508 Fixed known issue: Compiler error about exclusive areas declared implicitly in `Dcm_Dsp_MemoryServices.c`
- ASCDCM-2516 Fixed known issue: ServiceNeedsWizard generates NvM block NVM_BLOCK_DCM_DDDID independent of DcmDDDIDStorage
- ASCDCM-2551 Fixed known issue: Unconfigured dynamically defined data identifiers are not ignored by UDS service `ReadDataByPeriodicIdentifier (0x2A)`
- ASCDCM-2520 Fixed known issue: Compiler errors and warnings when no DcmDspMemoryIdInfo elements and no read or write memory ranges are configured

Module version 4.11.0

2016-03-04

- Updated #defines for symbolic name values according to AUTOSAR 4.0.3 naming schema
- ASCDCM-2432 Fixed known issue: Service `WriteMemoryByAddress (0x3D)` does not generate a valid positive response
- Added support for Debug & Trace with custom header file configurable via parameter `BaseDbgHeaderFile`
- ASCDCM-2447 Fixed known issue: The Dcm does not provide the definitions of macros which represent values of the Dcm_SesCtrlType type if the Rte is disabled
- ASCDCM-2452 Fixed known issue: NULL-pointer dereference occurs when the `InputOutputControlByIdentifier (0x2F)` service is executed if read-access for the DID to be controlled is disabled

Module version 4.10.6

2016-02-10

- Added the option of dummy configuration of Subfunction 0x00 for service `TesterPresent (0x3e)`
- ASCDCM-2331 Fixed known issue: DET errors are reported when the StopProtocol operation of the CallbackDCMRequestServices ClientServer Interface returns E_NOT_OK
- Improved transmission latency by triggering transmission of available PDID samples in the same cycle previous transmissions are confirmed
- ASCDCM-2379 Fixed known issue: Periodic UUDT transmissions occur on wrong TxPduIds when DcmDslPeriodicConnections from multiple protocols are sorted incorrectly
- ASCDCM-2384 Fixed known issue: The Dcm may become unresponsive if a page-buffered transmission for service `ReadDTCInformation (0x19)` is processed

- ASCDCM-2388 Fixed known issue: The service `RequestPowertrainFreezeFrameData (0x02)` returns wrongly a positive response if the request has incorrect length.
- ASCDCM-2403 Fixed known issue: Negative response when trying to schedule unconfigured DIDs along with existing and valid ones using UDS service `ReadDataByPeriodicIdentifier (0x2A)`
- ASCDCM-2398 Fixed known issue: The service `RequestPowertrainFreezeFrameData (0x02)` service returns a too long positive response under certain conditions
- ASCDCM-1850 Fixed known issue: The Dcm module does not compile if a DcmDslProtocolRow has the DcmDslProtocolID set to DCM_SUPPLIER_XX and Rte usage is disabled
- ASCDCM-2389 Fixed known issue: If service `ReadGenericInformation (0xAF)` is requested, an NRC 0x10 (general reject) can be wrongly generated
- The `DcmDslProtocolTransType` configuration parameter is made deprecated. The decision on which TxPduIds to transmit is now based on the presence of DcmDslPeriodicTransmissionConRef or DcmDslROEConnectionRef in the DcmDslMainConnection on which a periodic transmission is configured and on the presence of TxPduIds on the channels referenced by those two configuration parameters.
- ASCDCM-2415 Fixed known issue: The ECU resets if a DID with a smaller identifier than any configured DID is used in a request to any of the `InputOutputControlByIdentifier (0x2F)`, `ReadDataByIdentifier (0x22)`, `ReadDataByPeriodicIdentifier (0x2A)` or `WriteDataByIdentifier (0x2E)` services
- Improve Design and fix Design issues regarding service 0x2a

Module version 4.10.5

2016-01-15

- ASCDCM-2291 Fixed known issue: Generator and/or compiler errors occur when DID F186 is not configured with a signal with synchronous callout access
- ASCDCM-2314 Fixed known issue: The Dcm may become unable to transmit on a TxPduId in case of protocol cancellation
- ASCDCM-2315 Fixed known issue: Incorrect security level transitions and response messages when using `SecurityAccess (0x27)` for an invalid security level
- ASCDCM-2207 Fixed known issue: Asynchronous 'returnControlToECU' operations are always called with a DCM_INITIAL OpStatus when switching to the default session
- ASCDCM-2285 Fixed known issue: Memory corruption when service `ReadGenericInformation (0xAF)` is requested for all DTCs
- ASCDCM-2339 Fixed known issue: A compile error occurs if only start routine out signal has variable length
- ASCDCM-2306 Fixed known issue: The Dcm may become unresponsive as a result of executing a request for the `ReadDTCInformation (0x19)` service with the suppressPosRspMsgIndicationBit set

- ASCDCM-2128 Fixed known issue: Incorrect positive responses are reported for the `reportUserDefMemoryDTCExtDataRecordByDTCNumber (0x19)` subfunction of the `ReadDTCInformation (0x19)` service
- ASCDCM-2252 Fixed known issue: Periodic transmissions may be disabled if a periodic transmission event cannot be dispatched
- ASCDCM-2321 Fixed known issue: Erroneous interpretation of Dem error codes provokes unexpected behavior of the Dcm
- ASCDCM-2214 Fixed known issue: The Dcm returns a NRC when trying to read a dynamically defined DID containing an OBD DID as source
- ASCDCM-2362 Fixed known issue: Dcm might block or timeout after transmission when multiple periodic connections are configured per periodic diagnostic channel
- ASCDCM-2340 Fixed known issue: Service 0xAF with subservice 0x04 is erroneously providing a positive response under circumstances where a negative response "Conditions Not Correct" would apply
- ASCDCM-2358 Fixed known issue: Dcm module does not compile if services 0x07 or 0x0A and parameters `DcmDevErrorDetect`, `DcmDefProgEnabled`, and `DcmUnreachAssertEnabled` are enabled
- ASCDCM-2360 Fixed known issue: Service `ReadDTCInformation (0x19)` runs into timeout if used with a Dem module other than the EB Dem.
- ASCDCM-2311 Fixed known issue: UDS service `DynamicallyDefineDataIdentifier (0x2C)` allows configuring oversized dynamically defined DIDs
- ASCDCM-2334 Fixed known issue: Compiler errors occur if the `InputOutputControlByIdentifier (0x2F)` service is enabled, but no DID has control access.
- ASCDCM-1653 Fixed known issue: Wrong mode switches will be executed for ComM channels when the DcmDspComControlCommunicationReEnableModeRuleRef mode rule fails. Note: This bugfix changes the numerical values of the mode declarations.
- ASCDCM-2368 Fixed known issue: Service `ReadGenericInformation (0xAF)` sub-service 0x04 response contains too much data when no event data is present for a particular DTC.
- ASCDCM-1759 Fixed known issue: DTC Setting is reenabled on session change only when the internal `DiagnosticSessionControl (0x10)` service handler is used
- ASCDCM-2222 Fixed known issue: ComM Channels are not reset to the default communication mode when the DcmDspComControlCommunicationReEnableModeRuleRef ModeRule is not fulfilled if no DcmDspComControlSpecificChannel is configured
- ASCDCM-2375 Fixed known issue: Port interfaces for service `RoutineControl` might be defined following the wrong AUTOSAR specification version if vendor-specific parameter `DcmDspRoutineUsePortASRVersion` is disabled

Module version 4.10.4

2015-12-04

- ASCDCM-2255 Fixed known issue: The protocol on which a TYPE1 periodic request was received gets blocked in active state if the periodic request is processed on a different protocol
- ASCDCM-2242 Fixed known issue: Dcm does not correctly handle three or more parallel requests
- ASCDCM-2280 Fixed known issue: The ECU resets after it removes periodic DIDs from the scheduler with the `stopSending (0x04)` subfunction of the UDS service `ReadDataByPeriodicIdentifier (0x2A)`
- ASCDCM-2284 Fixed known issue: Incorrect answer for service `ReadGenericInformation (0xAF)` when shared reception and transmission buffers are used
- ASCDCM-2293 Fixed known issue: The Dcm becomes unresponsive if an external service handler returns an E_NOT_OK error

Module version 4.10.3

2015-11-06

- ASCDCM-2136 Fixed known issue: Compile errors occur when DcmDspRoutines have fixed length OUT signals and variable length IN signals
- ASCDCM-2187 Fixed known issue: Response contains IDs of unsupported OBD DIDs if service `ReadDataByIdentifier (0x22)` is used
- ASCDCM-2054 Fixed known issue: The service `Dynamically Define Data Identifier (0x2C)` could misbehave if DIDs from OBD Range are used as source
- ASCDCM-2206 Fixed known issue: Incorrect responses to the `RoutineControl (0x31)` service when DcmDspRoutines have fixed length OUT signals and variable length IN signals
- ASCDCM-2237 Fixed known issue: The space allocated for sampling periodic DIDs can only be used to max elements minus one
- Implemented support for reading the data identifier `0xF186 (ActiveDiagnosticSessionDataIdentifier)`

Module version 4.10.2

2015-10-22

- ASCDCM-2215 Fixed known issue: Integrating the Dcm with the EB PduR results in errors
- ASCDCM-2213 Fixed known issue: Dcm transmission may block if a lower layer module calls the `Dcm_CopyTxData()` or `Dcm_TpTxConfirmation()` APIs from an interrupt context or from the context of the `PduR_DcmTransmit()` API

Module version 4.10.1

2015-10-09

- ASCDCM-2071 Fixed known issue: The NRC set from the application is ignored if the `InputOutputControlByIdentifier (0x2F)` service is handled asynchronously and development error detection is enabled.
- ASCDCM-2186 Fixed known issue: Interfaces responsible for reading DID signals may be erroneously called with the DCM_PENDING OpStatus
- ASCDCM-2113 Fixed known issue: The ECU may block if one of the configured DIDs has just `returnControlToEcu` access enabled
- Add new vendor specific configuration parameter for user configurable Addressing of User-Defined Memory DEM_DTC_ORIGIN_SECONDARY_MEMORY
- ASCDCM-2147 Fixed known issue: If an OBD MID is requested with service `ReadDataByIdentifier (0x22)`, the response is too long

Module version 4.10.0

2015-10-02

- ASCDCM-2078 Fixed known issue: The service `DiagnosticSessionControl (0x10)` can block the ECU with incorrect configuration
- ASCDCM-2022 Fixed known issue: The service `Dynamically Define Data Identifier (0x2C)` could misbehave if DIDs from DID Ranges are used as source
- ASCDCM-2085 Fixed known issue: Wrong response if a request of service `ReadMemoryByAddress (0x23)` is made and `Dcm_DspInternalReadMemoryAddress_Execute()` returns DCM_E_READ_FORCE_RCRRP
- Changed name of file `Dcm_Dsp_SvcH_OBDMode6.c/.h` to `Dcm_Dsp_SvcH_RequestControlOfOnBoardSysTstComp.c/.h`
- ASCDCM-2083 Fixed known issue: Generator errors occur when DcmDspRoutines have fixed length OUT signals and variable length IN signals
- ASCDCM-2095 Fixed known issue: Dcm_DsdServiceUsed may deactivate several services in different DcmDsdServiceTable
- ASCDCM-2117 Fixed known issue: Incorrect NRC is sent for UDS service `InputOutputControlByIdentifier (0x2F)` with inputOutputControlParameter out of range
- Implemented UUDT transmission and AUTOSAR 4.2.1 - compatible handling for UDS service `ReadDataByPeriodicIdentifier (0x2A)`
- ASCDCM-2152 Fixed known issue: The service `ReadGenericInformation (0xAF)` runs into an endless loop under certain conditions

- ASCDCM-2155 Fixed known issue: If an OBD availability PID is requested with service `ReadDataByIdentifier (0x22)`, the response is invalid
- Added config checks according service 0x06 and OBDMID
- ASCDCM-2172 Fixed known issue: If service `ReadDataByIdentifier (0x22)` is an OBD availability VehicleInfoType e.g. by requesting DID 0xF800, SW may crash

Module version 4.9.1

2015-09-07

- ASCDCM-3700 Fixed known issue: Unexpected behavior when the CommunicationControl (0x28) service is configured and the DiagnosticSessionControl (0x10) service is requested
- ASCDCM-2060 Fixed known issue: `Dcm_DemFacade_GetEventData()` broken, leads to not working `ReportDtcSnapshotRecordByDtcNumber`
- ASCDCM-2089 Fixed known issue: MamMap include issue

Module version 4.9.0

2015-08-28

- Support of DIDs in system supplier specific range
- Introduced NRC 0x14 (responseTooLong) for UDS service ReadDataByIdentifier 0x22
- ASCDCM-2074 Fixed known issue: Compiler warnings are reported when OBD services $06 and $08 are both used
- Implemented support for protocol preemption

Module version 4.8.2

2015-07-31

- Added a limitation for the service `Dynamically Define Data Identifier (0x2C)`. The service responds with NRC 0x31 in case of adding a Dynamic DID as source.
- ASCDCM-1970 Fixed known issue: An illegal memory access may occur if the subfunction `defineByIdentifier (0x01)` of service `DynamicallyDefineDataIdentifier (0x2C)` is used
- ASCDCM-2034 Fixed known issue: Wrong data is written for a DID if service `WriteDataByIdentifier (0x2E)` is used with a DID range
- ASCDCM-2026 Fixed known issue: Description of operations of interface DataServices_DIDRange does not comply to AUTOSAR 4.2.1

- Implemented support for configuring diagnostic sessions for sub-functions of UDS service `SecurityAccess 0x27`
- ASCDCM-2001 Fixed known issue: In OBD service 0x02 (Request powertrain freeze frame data) PID 0x00 does not show PID 0x02 as supported

Module version 4.8.1

2015-07-10

- ASCDCM-1954 Fixed known issue: The ECU may block or reset if the number of signals of a DID is higher than 254
- ASCDCM-1938 Fixed known issue: If `Xxx_GetInfotypeValueData()` returns pending, an unreachable code error may be displayed (OBD service 0x09)
- ASCDCM-1933 Fixed known issue: Compiler or generator errors may occur when AUTOSAR 4.2.1 APIs are used by DcmDspRoutines which contain VARIABLE_LENGTH signals
- ASCDCM-1953 Fixed known issue: Code generation fails when a DID is configured for 'Control' access without explicit 'Read' access and uses `ClientServerInterfaces` for data access
- ASCDCM-1998 Fixed known issue: Incomplete writing of signals may occur using service `WriteDataByIdentifier(0x2E)` with DID including multiple signals with USE_BLOCK_ID interface
- ASCDCM-2002 Fixed known issue: Dcm triggers a DET error if service `ReadDataByIdentifier (0x22)` is canceled
- ASCDCM-1748 Fixed known issue: The Dcm does no longer respond to requests if the communication channel is switched off while it dispatches an ROE event
- ASCDCM-1978 Fixed known issue: Wrong response messages are sent for service `SecurityAccess (0x27)`
- ASCDCM-1936 Fixed known issue: The service `Dynamically Define Data Identifier (0x2C)` allows misconfiguration which results in incorrect run-time behavior
- ASCDCM-1641 Fixed known issue: 'Unused variable' compiler warning under certain configuration
- Implemented support for reading OBD data identifiers using service `ReadDataByIdentifier ($22)`

Module version 4.8.0

2015-06-19

- ASCDCM-1783 Fixed known issue: Service handlers act as if the response transmission was successful if the response cannot be sent before the P2 timeout expires
- ASCDCM-1913 Fixed known issue: No DTCs are reported after ClearDTC in OBD service `(0x0A)` "Request emission-related diagnostic trouble codes with permanent status"

- ASCDCM-1615 Fixed known issue: DTC setting may be enabled on every session change if DCM_ALL_SESSION_LEVEL is configured for service `ControlDTCsetting (0x85)`
- ASCDCM-949 Fixed known issue: The Dcm relies on the fact that the Dem interfaces never return a "PENDING" status
- ASCDCM-1747 Fixed known issue: Unnecessary check of the size of all extended data records for a DTC
- Implemented Calibration of Dcm Services
- Adapted EB-internal assert handling
- ASCDCM-1447 Fixed known issue: ECU resets too early if a jump to the bootloader is executed by using the service `DiagnosticSessionControl (0x10)`
- Implemented support for `DTCSettingControlOptionRecord` and `LengthCheck` for `ControlDTCSetting` service request according to AUTOSAR 4.2.1.
- ASCDCM-1879 Fixed known issue: Compiler errors are reported if the `DynamicallyDefineDataIdentifier (0x2C)` service is used and none of the `RequestDownload (0x34)`, `RequestUpload (0x35)`, `ReadMemoryByAddress (0x23)` or `WriteMemoryByAddress (0x3D)` services are used
- Implemented customer-specific diagnostic service `ReadGenericInformation (0xAF)`.
- ASCDCM-1915 Fixed known issue: Request type (reqType) in `Dcm_MsgAddInfoType` is incorrectly defined
- Implemented UDS service `ReadDTCInformation (0x19)`, subservice `ReportUserDefMemoryDTCSnapshotRecordByDTCNumber (0x18)`.

Module version 4.7.0

2015-04-29

- Implemented support for `OBD Service $06` - Request On-Board Monitoring Test-results for Specific Monitored Systems API according to both AUTOSAR 4.0.3 and AUTOSAR 4.2.1 releases.
- ASCDCM-1836 Fixed known issue: Rte interface for the `RoutineControl` service is not correctly generated
- ASCDCM-1868 Fixed known issue: If AUTOSAR 4.2.1 DataServices interfaces are used, a compile error might occur in case when RTE is not configured
- ASCDCM-1866 Fixed known issue: Compiler errors are reported when the `RoutineControl (0x31)` service is configured and the default Dcm interface version is set to Autosar 4.2.1
- ASCDCM-1876 Fixed known issue: Application call-back function `IsDidAvailable` of feature DidRanges is not using `Dcm_DidSupportedType`
- ASCDCM-1776 Fixed known issue: Compiler errors are reported if the `Dcm_ReadMemory()` or `Dcm_WriteMemory()` callbacks are not defined

- Implemented support for suppressed NRC of functional requests according to both ISO14229-2006 and ISO14229-2013 releases.
- Implemented support for the `DCM_Roe` ClientServerInterface according to AUTOSAR 4.2.1 by:
  1. having the `RoeEventId` parameter defined as a Portdefined argument value;
  2. changing the name of the generated P-port to `DCM_Roe_{RoeName}`;

  as a configuration selectable alternative to the existing `DCM_Roe` ClientServerInterface implementation according to AUTOSAR 4.0.3
- Implemented support for calibration of configuration parameters: `DcmDspPidUsed`, `DcmDspTestResultObdmidUsed`, `DcmDspTestResultTid`, `DcmDspVehInfoUsed` and `DcmDspRequestControlUsed`.
- Implemented support for the OBD service `$09` Request Vehicle Information.
- Implemented support for OBD Service `$08` Request Control of On-Board System, Test or Component
- ASCDCM-1903 Fixed known issue: Response transmission may not be possible if TYPE2 periodic responses are used and DET checks are enabled
- ASCDCM-1897 Fixed known issue: The `reportUserDefMemoryDTCByStatusMask(0x17)` sub-function of service `ReadDTCInformation (0x19)` is not configurable in Tresos.

Module version 4.6.4

2015-04-02

- Fixed usage of Dem API interfaces `Dem_GetDTCOfOBDFreezeFrame` and `Dem_ReadDataOfOBDFreezeFrame`
- ASCDCM-1865 Fixed known issue: If a DidRange is configured without any service using Dids, a compiler error occurs

Module version 4.6.3

2015-03-27

- Improved the asynchronous processing of service handlers by breaking the continuous processing within a loop inside a separate task into discreet calls to the service handler function. Each call is executed within a separate task, thus allowing tasks of a lower priority than the asynchronous service processor task to also be scheduled, as they will no longer be blocked until the service handling is completed.
- Added software units for diagnostic services `0x22`, `0x2A` and `0x2E`. Added `DidServices` software unit and improved diagnostic service handlers `0x22` and `0x2E`.
- Implemented support for the OBD services:
  - `$03` Request emission-related diagnostic trouble codes

- `$07` Request emission-related diagnostic trouble codes detected during current or last completed driving cycle
- `$0A` Request emission-related diagnostic trouble codes with permanent status

- ASCDCM-1787 Fixed known issue: The ECU becomes unresponsive or memory corruption occurs if a DID is requested which is unsupported and outside the `ReadDID` Range
- Implemented support for the `SecurityAccess` API according to AUTOSAR 4.2.1 as a configuration selectable alternative to the existing `SecurityAccess` API implementation according to AUTOSAR 4.0.3
- ASCDCM-1837 Fixed known issue: The ECU becomes unresponsive, restarts or performs faulty TYPE2 periodic response transmissions
- Implemented support for the `DataServices` API according to AUTOSAR 4.2.1 as a configuration selectable alternative to the existing `DataServices` API implementation according to AUTOSAR 4.0.3
- ASCDCM-1843 Fixed known issue: File `AdjacentLayer.properties` contains wrong x-path expressions which leads to errors during generation of PduR
- Implemented support for the OBD service: `$04` Clear/reset emission-related diagnostic information

Module version 4.6.2

2015-03-06

- ASCDCM-1771 Fixed known issue: The ECU may reset or become unresponsive if service InputOutputControlByIdentifier (0x2F) is requested
- ASCDCM-1674 Fixed known issue: Undefined identifier errors are reported when Rte usage is enabled in Dcm and Dcm internal services are not configured to use a Client/Server interface
- Added improvements to the the `DynamicallyDefineDataIdentifier` (0x2C), `ReadDataByPeriodicIdentifier` (0x2A) and `ReadDataByIdentifier` (0x22) service handlers
- Changed datatype of the index used to search the DID table in `Dcm_DspInternal_OnChangeOfDataIdentifier_SvcStart` to uint16_least in order to avoid possible problems on 8-bit architectures.
- ASCDCM-1777 Fixed known issue: Compiler errors are reported if the `DynamicallyDefineDataIdentifier` (0x2C) service is configured, but the `ReadMemoryByAddress` (0x23) service is not configured
- ASCDCM-1779 Fixed known issue: Compiler errors are reported if the `ReadMemoryByAddress` (0x23), the `WriteMemoryByAddress` (0x3D), or the `DynamicallyDefineDataIdentifier` (0x2C) service is configured, but the `DcmDspMemory` container is disabled
- ASCDCM-1671 Implemented UDS service `ReadDTCInformation (0x19)`, subservice `ReportUserDefMemoryDTCExtDataRecordByDTCNumber (0x19)`
- ASCDCM-1803 Fixed known issue: The ECU becomes unresponsive or memory corruption occurs when the `RoutineControl` (0x31) service is requested without a `routineControlType` sub-function

- ASCDCM-1498 Fixed known issue: Dcm communication may fail in the event of a paged buffering transmission timeout
- Implemented ISO14229-1:2013 Response Message Format (UUDT Identifiers) for the diagnostic service `ReadDataByPeriodicIdentifier (0x2A)` responses.
- Implemented support for TYPE2 periodic responses.
- Implemented support for the `RoutineServices` API according to AUTOSAR 4.2.1 as a per routine selectable alternative to the existing `RoutineServices` API implementation according to AUTOSAR 4.0.-3 The implementation is based on AUTOSAR Bugzilla RFC #57860

Module version 4.6.1

2015-02-13

- ASCDCM-1586 Fixed known issue: Incorrect response to UDS service 0x19 0x06 if the DTC isn't stored in error memory
- ASCDCM-1535 Fixed known issue: `ResponseOnEvent` transmissions may free buffers held by preempted normal diagnostic transmissions
- ASCDCM-1562 Fixed known issue: Corrupt responses for service `ReadDataByIdentifier (0x22)` and unreliable behavior for onChangeOfDataIdentifier-triggered ROE events when asynchronous service processing is used
- Implemented support for the `Dem_DcmClearDTC()` API according to AUTOSAR 4.2.1 as a configuration selectable alternative to the existing `Dem_ClearDTC()` AUTOSAR 4.0.3 API

Module version 4.6.0

2015-01-21

- Implemented negative response with NRC 0x10 instead of 0x72 upon `Dcm_ReadMemory` failing with `DCM_READ_FAILED`, considering both AUTOSAR 4.2 rev1 and Bugzilla issue https://www.autosar.org/bugzilla/show_bug.cgi?id=57196
- ASCDCM-1565 Fixed known issue: Executing the `EcuReset (0x11)` service may fail if a previous request to the same service timed-out
- ASCDCM-1564 Fixed known issue: Service `DiagnosticSessionControl (0x10)` may fail to execute after a timed-out jump to the bootloader
- ASCDCM-1488 Fixed known issue: The Dcm does not reset the `CommunicationControl` when switching to the default session
- ASCDCM-1340 Fixed known issue: Generated symbol names for `DcmDslProtocolRx`, `DcmDslProtocolID`, `DcmDspSecurityLevel` and `DcmDspSessionLevel` result in name clashes

- ASCDCM-1241 Fixed known issue: External subfunction handlers for the service `ReadDTCInformation (0x19)` are only called with the DCM_INITIAL OpStatus.
- Implemented UDS service `ReadDTCInformation (0x19)`, subservice `ReportDTCFaultDetectionCounter (0x14)`
- Implemented call of `SchM_Switch_Dcm_DcmDiagnosticSessionControl()` at initialization time if the initialization is a result of a warm-start condition.
- Implemented rejection of diagnostic requests after a positive response to the `EcuReset (0x11)` service, considering both AUTOSAR 4.1 rev3 and Bugzilla issue https://www.autosar.org/bugzilla/show_bug.cgi?id=53154.
- Implemented requirement Dcm768: Call of `BswM_Dcm_ApplicationUpdated` after jump from bootloader
- Implemented support of OBD DIDs via regular UDS-style DID access.
- The services `SecurityAccess(0x27)`, `CommunicationControl(0x28)`, `ControlDTCSetting(0x85)`, `ReadDataByPeriodicIdentifier(0x2A)`, and `InputOutputControlByIdentifier(0x2F)` can now be configured to be available in the default session
- Implemented support for the configuration parameter `DcmDspDataConditionCheckReadFncUsed`, as described in Autosar Bugzilla RfC #53669
- ASCDCM-1753 Fixed known issue: If the `ResponseOnEvent` (0x86) service is used with an `OnChangeOfDataIdentifier` (0x03) event for an internally managed DID which does not have Read-access enabled, the ECU may reset or become unresponsive
- Implemented support for the `DEM_NUMBER_PENDING` return value for the `Dem_GetNumberOfFilteredDTC()` API.
- Implemented support for the `DEM_FILTERED_PENDING` return value for the `Dem_GetNextFilteredDTC()` API.
- Implemented support for the `DEM_FILTERED_PENDING` return value for the `Dem_GetNextFilteredRecord()` API.

Module version 4.5.0

2014-10-22

- ASCDCM-1526 Fixed known issue: The Dcm may report errors during the generation phase if the default session's `DcmDspSession` entry is not named DCM_DEFAULT_SESSION and `DcmModeDeclarationSupport` is enabled
- ASCDCM-1529 Fixed known issue: Compiler errors are reported if the `ResponseOnEvent (0x86)` service is configured without configuring at least one `DcmDslResponseOnEvent` connection

- ASCDCM-1567 Fixed known issue: Jumping to the boot loader fails if the delay in setting the `ProgrammingConditions` causes a P2/P2* server timeout
- ASCDCM-1416 Fixed known issue: The Dcm may time out or perform the jump to the bootloader very late if the `DiagnosticSessionControl (0x10)` is configured to be run asynchronously
- ASCDCM-1552 Fixed known issue: Compiler error when a single protocol is configured and the number of `RequestCorrectlyReceivedResponsePending` NRCs that the Dcm can send is unlimited
- ASCDCM-1483 Fixed known issue: The Dcm does not enable DTC setting when transitioning to the default diagnostic session outside of an S3 timeout scenario
- ASCDCM-1566 Fixed known issue: Jumping to the bootloader may fail if a previous request to the `EcuReset (0x11)` service timed-out
- ASCPD-189 Implemented subfunction `reportDTCWithPermanentStatus (0x15)` of UDS service `ReadDTCInformation (0x19)`
- ASCPD-189 Implemented subfunction `reportEmissionsRelatedOBDDTCByStatusMask (0x13)` of UDS service `ReadDTCInformation (0x19)`
- ASCDCM-1540 Fixed known issue: The Dcm may become unresponsive when executing service `RoutineControl (0x31)`
- ASCDCM-1557 Fixed known issue: The Dcm may corrupt the memory when ComM channels are re-enabled when the `DcmDspComControlCommunicationReEnableModeRuleRef` mode rule fails
- ASCDCM-1558 Fixed known issue: The wrong ComM channels are re-enabled when the `DcmDspComControlCommunicationReEnableModeRuleRef` mode rule fails
- ASCDCM-1597 Fixed known issue: The Dcm becomes unresponsive if a bootloader jump is executed with incorrect programming conditions
- ASCDCM-1484 Fixed known issue: The Dcm does not enable DTC setting correctly after a successful `ECUReset (0x11)` request
- ASCDCM-1444 Fixed known issue: `ComM_DCM_ActiveDiagnostic()` is not always called before an NRC is sent
- ASCDCM-1585 Fixed known issue: Incorrect behavior of service `ReadDTCInformation (0x19)` with subservices `reportDTCSnapshotRecordByDTCNumber (0x04)` and `reportDTCExtendedDataRecordByDTCNumber (0x06)` if paged buffering is used
- Implemented configuration parameter support for the `DynamicallyDefineDataIdentifier (2C hex)` service
- Implemented UDS service `ReadDataByPeriodicIdentifier (2A hex)`
- Implemented UDS service `DynamicallyDefineDataIdentifier (2C hex)`
- ASCDCM-2122 Fixed known issue: Buffer corruption occurs if the maximum size of data is requested that may fit in the transmission buffer with the `ReadMemoryByAddress (0x23)` service

Module version 4.4.4

2014-08-08

- ASCDCM-1463 Fixed known issue: Out of bounds memory access when ROE events are not configured and started
- ASCDCM-1489 Fixed known issue: Compiler warnings are reported if the same symbol name is used more than once for `RoutineControl` operations
- ASCDCM-1413 Fixed known issue: Programming conditions are corrupted when the `DiagnosticSessionControl(0x10)` service and asynchronous processing are used
- ASCDCM-1412 Fixed known issue: The Dcm may attempt a jump to the bootloader while a previous call of `Dcm_SetProgConditions()` is still underway
- Implemented `DcmModeRule` support for reading or writing individual `DcmDspMemory` ranges
- Implemented `DcmModeRule` support for reading, writing, and controlling individual DIDs
- Implemented `DcmModeRule` support for re-enabling of DTC setting
- Implemented `DcmModeRule` support for resetting the state of the `CommunicationControl` service
- ASCDCM-1475 Fixed known issue: The Dcm performs service and subfunction diagnostic session and security level checks incorrectly
- ASCDCM-1491 Fixed known issue: `OnChangeOfDataIdentifier` ROE event generation fails for DIDs containing variable length signals
- ASCDCM-1556 Fixed known issue: Compiler errors are reported if Mode Declaration support is enabled and no specific channels for the CommunicationControl (0x28) service are configured
- ASCDCM-1555 Fixed known issue: Compiler warnings are reported for certain configuration options for diagnostic session handling
- ASCDCM-1438 Fixed known issue: Unexpected concurrencies and possible memory corruption occur if asynchronous service processing is used
- Added the possibility to configure input signals for the `requestRoutineResults (0x01)` sub-function of the `RoutineControl(0x31)` diagnostic service
- ASCDCM-1524 Fixed known issue: The Dcm reports warnings during the generation phase

Module version 4.4.3

2014-04-25

- ASCDCM-1311 Fixed known issue: The Dcm may send responses for service `ReadDTCInformation (0x19)` even if the communication channel is in silent communication mode
- ASCDCM-1329 Fixed known issue: The requestSeed / compareKey subfunction sequence of the `SecurityAccess (0x27)` service is not reset correctly

- ASCDCM-1339 Fixed known issue: The Dcm transmits service responses with the wrong TxPduId when only one protocol (`DcmDslProtocolRow`) is configured
- ASCDCM-1315 Fixed known issue: Duplicate CurHsmInfo variable in the `Dcm_DsdInternal_AllocateTxBuffer()` function
- Improvement to make sure all periodic DIDs are sampled and sent under the condition that more samples are generated per time unit as can be send
- ASCDCM-1336 Fixed known issue: Data may be corrupted when writing DID signals to NVRAM using the service `WriteDataByIdentifier (0x2E)`
- ASCDCM-1139 Fixed known issue: Configuration of a DID referencing other DIDs but containing no signals is not possible
- ASCDCM-1357 Fixed known issue: Service `ResponseOnEvent (0x86)` sends wrongly addressed messages, corrupts buffers and locks the Dcm
- ASCDCM-1361 Fixed known issue: The response to the `DiagnosticSessionControl (0x10)` that causes a jump/return from the bootloader is sent on an invalid txPduId
- ASCDCM-1346 Fixed known issue: Parameters `DcmDspDataUsePort` and `DcmDspSecurityUsePort` use values different from the AUTOSAR-specified ones
- ASCDCM-1366 Fixed known issue: Subfunction `ReportDtcSnapshotRecordByDtcNumber (0x04)` of service `ReadDTCInformation (0x19)` does not allow the request of specific data records with values 0x00 or 0xF0-0xFE
- ASCDCM-1396 Fixed known issue: Usage of `Dcm_GetActiveProtocol()` may cause an invalid memory access
- ASCDCM-1394 Fixed known issue: Dcm communication modes may not be set correctly by the ComM
- ASCDCM-1406 Fixed known issue: Dcm communication modes may not be interpreted correctly
- ASCDCM-1395 Fixed known issue: Use of `Dcm_ResetToDefaultSession()` may result in an invalid memory access, corrupt diagnostic session settings and communication failure
- ASCDCM-1414 Fixed known issue: Usage of `Dcm_GetSesCtrlType()` may result in an invalid memory access or the wrong session identifier being reported
- ASCDCM-1398 Fixed known issue: The Dcm may send an extra `requestCorrectlyReceivedResponsePending` NRC when executing a jump to the Bootloader via an asynchronous `DiagnosticSessionControl(0x10)` request
- ASCDCM-1401 Fixed known issue: Buffer corruption and/or blocked Dcm when ROE final responses are transmitted during a request reception
- ASCDCM-1429 Fixed known issue: If the service `ECUReset (0x11)` is configured to use an internal service handler, but only user-implemented subfunction handlers are used, a compiler warning is reported
- ASCDCM-1421 Fixed known issue: Invalid memory access when unused `DcmDspRoutineInfo` members are configured

- ASCDCM-1403 Fixed known issue: Use of the `Dcm_ResetToDefaultSession()` API function may cause inconsistent or faulty behaviour in the Dcm
- ASCDCM-1332 Fixed known issue: Jumping to the bootloader is not executed correctly if a `requestCorrectlyReceivedResponsePending(0x78)` NRC is to be sent in the same Dcm cycle
- ASCDCM-1415 Fixed known issue: If asynchronous service handlers time out, the wrong NRC may be transmitted
- ASCDCM-1220 Fixed known issue: DID data may get corrupted as a result of a DID NvM Block read/write operation taking too much time to execute
- ASCDCM-1430 Fixed known issue: ROE final responses cause the Dcm to become unresponsive
- ASCDCM-1397 Fixed known issue: The Dcm may fail to respect S3 timing requirements
- ASCDCM-1439 Fixed known issue: The Dcm may block if asynchronously processed services time out
- ASCDCM-1158 Fixed known issue: Overflows because of values assigned to `DcmDspSecurityADRSize`, `DcmDspSecurityKeySize` and `DcmDspSecuritySeedSize`
- Implemented `DcmModeRule` support for controlling individual Routines
- ASCDCM-1469 Fixed known issue:Generator reports a warning if Rte usage is enabled and externally managed DIDs are not configured

Module version 4.4.2

2013-11-27

- ASCDCM-1103 Fixed known issue: The Dcm may send corrupt responses as a result of a race condition between transmission and reception
- ASCDCM-1162 Fixed known issue: Generator errors when `DcmDspRoutineUsed` is FALSE and `DcmDspRoutineUsePort` is TRUE
- ASCDCM-1314 Fixed known issue: MemMap section for uninitialized variables of unspecified length and 32 bit variable in Dcm.c.m4 is not correct
- ASCDCM-1226 Fixed known issue: The Dcm accepts requests before being fully initialized
- ASCDCM-1217 Fixed known issue: Wrong `ProtocolID` written in `ProgConditions`
- ASCDCM-1292 Fixed known issue: The Dcm may block during processing of ROE events
- Implemented the possibility to have mixed (Variable length and Fixed length) signal types present in the `RoutineControl (0x31)` service operation interfaces
- Implemented the possibility to call `Dem_DcmCancelOperation()` in case the service `ClearDiagnosticInformation (0x14)` times out as a result of too many `REQUESTCORRECTLYRECEIVEDRESPONSEPENDING` NRCs having been sent

- ASCDCM-1168 Fixed known issue: The Dcm fails to respond to service requests received when the communication channel used is in Silent Communication Mode
- ASCDCM-1090 Fixed known issue: Incorrect service handler return values dependent on user-implemented code (callouts and Rte interface calls) may block DCM

Module version 4.4.1

2013-10-18

- ASCDCM-1182 Fixed known issue: The handling of statuses for a routine ("started" or "stopped") is not done correctly
- ASCDCM-1120 Fixed known issue: Asynchronous DID interfaces/operations are called with an OpStatus of DCM_INITIAL just for the first signal belonging to the DID
- ASCDCM-1113 Fixed known issue: A protocol processing an ROE event may corrupt the transmit buffers of another currently executing protocol
- ASCDCM-1250 Fixed known issue: DTC record updating may remain disabled if the subfunction `ReportDTCSnapshotRecordbyDTCNumber` of the service `ReadDTCInformation (0x19)` fails
- ASCDCM-1295 Fixed known issue: Asynchronous service processing of external service handlers can cause unwanted behavior

Module version 4.4.0

2013-09-17

- ASCDCM-1206 Fixed known issue: Page-buffered transmission may fail if the lower layer only inquires about the remaining amount of data in the Tx buffer when all data has been sent from the current page, instead of requesting more data
- ASCDCM-927 Fixed known issue: Asynchronous `RoutineControl` and `DataServices` service handlers are not called with OpStatus `DCM_CANCEL`
- Implemented support for multiple `DcmDslMainConnections` per `DcmDslProtocolRow`
- Introduced a configuration parameter `DcmDspRoutineVariableLengthInBytes` in the `DcmDsp` container to indicate whether the `VARIABLE_LENGTH` signals in the Routine Control configurations are expressed in bytes or bits
- ASCDCM-882 Fixed known issue: Range of values for `DcmDspSecurityDelayTime` needs to be extended
- ASCDCM-1236 Fixed known issue: The Dcm reports a DET error when `Dcm_GetProgConditions()` returns `DCM_COLD_START` at startup
- ASCDCM-1230 Fixed known issue: Compiler errors due to too large shift counts on 16-bit architectures

- ASCDCM-1009 Fixed known issue: User-configured subfunction handlers for EB-implemented services `CommunicationControl (0x28)`, `EcuReset (0x11)`, `LinkControl (0x87)` or `ControlDTCSetting (0x85)` are only called once

Module version 4.3.0

2013-08-05

- ASCDCM-1088 Fixed known issue: The ECU may crash when executing service `InputOutputControlByIdentifier (0x2F)`
- ASCDCM-1074 Fixed known issue: In case `DcmDslDiagRespOnSecondDeclinedRequest` is imported as TRUE, the response buffer may get corrupted
- ASCDCM-1075 Fixed known issue: The parameter in the function prototype for Dcm_Init() needs to be declared as type "const"
- ASCDCM-1099 Fixed known issue: Service `WriteDataByIdentifier (0x2E)` may corrupt DID data as a result of a race condition
- ASCDCM-761 Fixed known issue: Current variable sizes are insufficient for parameters dealing with timing values
- ASCDCM-1079 Fixed known issue: Compiler error when no security level is configured (no need for security level checks) and Rte usage is enabled
- ASCDCM-1112 Fixed known issue: The Dcm Dsl uses the wrong `TxPduId` and `ConfirmationTxPduId`

Module version 4.2.0

2013-06-26

- ASCDCM-1019 Fixed known issue: The Dcm may initialize its ROE persistent data structure with corrupt values
- ASCDCM-870 Fixed known issue: `ImplementationDatatype` of type `TYPE_REFERENCE` references an `InvalidValue` when it is not allowed
- ASCDCM-1068 Fixed known issue: Mode declaration groups and prototypes are generated conditionally based on service availability
- ASCDCM-1034 Fixed known issue: Undefined identifier errors are reported when Rte usage is enabled in Dcm, the `ResponseOnEvent (0x86)` service is configured and none of the Dcm internal services is configured to use a Client/Server interface
- ASCDCM-1059 Fixed known issue: Compiler errors may be reported when the `ShortTermAdjustment` operation is enabled for a DID and one of the signals configured for this DID is of variable length and uses a ClientServer interface

- ASCDCM-1018 Fixed known issue: Optional parameter `DcmDslProtocolSessionRef` causes a code generation error if it is disabled
- ASCDCM-1023 Fixed known issue: Configuration for routines is generated incorrectly
- ASCDCM-983 Fixed known issue: Compiler errors when configuring `DcmDspRoutineSignalType` of a `DcmDspRoutineInfo` to a length other than `VARIABLE_LENGTH` when `DcmDspRoutineFixedLength` is set to FALSE in a routine which references this `DcmDspRoutineInfo`
- ASCDCM-1047 Fixed known issue: The `RoutineControl (0x31)` service fails if the complete data length configured for input signals is not a multiple of 8 bits
- ASCDCM-1026 Fixed known issue: Compilation fails if the same subfunction ID `DcmDsdSidTabServiceId` is configured for the same service `DcmDsdSidTabServiceId` inside different service tables (`DcmDsdServiceTable`)
- ASCDCM-1051 Fixed known issue: The persistent ROE initialization must be modified to reinitialize the internal variables with defaults in case of failure
- ASCDCM-884 Fixed known issue: `REQUESTCORRECTLYRECEIVEDRESPONSEPENDING (0x78)` NRCs are transmitted too late
- ASCDCM-1063 Fixed known issue: `Dcm_CopyTxData()` returns `BUFREQ_E_NOT_OK` if called with a PduInfoPtr->SduLength of 0
- ASCDCM-1086 Fixed known issue: The SecurityAccess (0x27) service may fail if subfunction availability (`DcmDsdSidTabSubfuncAvail`) is disabled
- Implemented DcmModeRule support for general service and subfunction handling
- ASCDCM-1061 Fixed known issue: Asynchronous service processing can execute forever

Module version 4.1.4

2013-05-22

- ASCDCM-1017 Fixed known issue: BswM APIs are called even though the Mode Declaration Support is used
- Added support for containers `DcmDspComControlAllChannel` and `DcmDspComControlSpecificChannel`
- ASCDCM-995 Fixed known issue: The Dcm performs Diagnostic Session checks on subfunction level correctly only for the first subfunction configured in the `DcmDsdSubService` table
- ASCDCM-1005 Fixed known issue: The Dcm performs Security Level checks on subfunction level correctly only for the first subfunction configured in the `DcmDsdSubService` table
- ASCDCM-1032 Fixed known issue: Mode Declarations are generated conditionally within Mode Declaration Groups based on subfunction configuration

- Implemented `ModeDeclarationGroupPrototypes` for service `CommunicationControl (0x28)` and invoked SchM Switch APIs for mode switch notifications
- ASCDCM-994 Fixed known issue: Configuration for Diagnostic Session and Security Level permission checks on subfunction level is not correctly generated
- Updated asynchronous service execution of services and related test cases
- Implemented usage of `BswM_Dcm_CommunicationMode_CurrentState()` for notifying ComMode to BswM within the `CommunicationControl (0x28)` service

Module version 4.1.3

2013-04-12

- ASCDCM-972 Fixed known issue: Suppress Positive Response Bit is not reset correctly
- ASCDCM-854 Fixed known issue: Service `ResponseOnEvent` returns illegal Negative Response Codes
- Implemented `ModeDeclarationGroupPrototypes` for service `ControlDTCSetting(0x85)` and invoked SchM Switch APIs for mode switch notifications

Module version 4.1.2

2013-03-19

- ASCDCM-941 Fixed known issue: Compiler errors when configuring `DcmDspRoutineInfo` containers which are not referenced from any `DcmDspRoutineInfoRef`
- ASCDCM-963 Fixed known issue: Compilation fails if asynchronous service processing is enabled and Det usage is disabled
- ASCDCM-852 Fixed known issue: The ROE persistency NV block is not initialized at first start-up of the ECU, leading to usage of corrupt data stored in the non-volatile memory
- ASCDCM-944 Fixed known issue: Active security level is set to an incorrect value
- Added DataLength parameter to interface `Xxx_ShortTermAdjustment` in case the length of the DID signal is not fixed (RfC #58643)
- ASCDCM-958 Fixed known issue: The Dcm locks after a request is rejected from the Manufacturer/Supplier Indication without any NRC being sent, if the requested service is configured to be handled asynchronously
- Changed the type of the configuration parameter `DcmDslProtocolRxComMChannelRef` from `SYMBOLIC-NAME-REFERENCE` to `CHOICE-REFERENCE`
- ASCDCM-935 Fixed known issue: `DcmDspDidRoeActivateFnc` configuration parameter must always be configured if the `DcmDspDidExtRoe` container is enabled

Module version 4.1.1

2013-02-25

- ASCDCM-813 Fixed known issue: The operation `TriggerOnEvent` of Client-Server interface DCM_Roe uses the wrong `ImplementationDataType`
- ASCDCM-839 Fixed known issue: Code generation fails when Dcm is added to PduR/PduRBswModules and parameter `PduRBswModuleIsEnabled` is enabled
- ASCDCM-849 Fixed known issue: The Dcm may accept new requests while processing an ROE Type 1 event
- ASCDCM-842 Fixed known issue: Undefined identifier errors are reported when Rte usage is enabled in Dcm and none of the Dcm internal services is configured to use a Client/Server interface
- ASCDCM-848 Fixed known issue: Dcm ceases to respond to service requests if the transmission of the response to service `DiagnosticSessionControl (0x10)` fails
- ASCDCM-857 Fixed known issue: `DcmDspDataReadDataLengthFnc` always has to be configured, even if the data of the DID has a fixed length
- ASCDCM-878 Fixed known issue: Compiler error when only one protocol is used, paged buffer handling is enabled and UDS service `ReadDtcInformation (0x19)` is configured
- Added Session and Security-checks on subfunction level
- Implemented UDS service `ReadMemoryByAddress(0x23)`
- Implemented UDS service `WriteMemoryByAddress(0x3D)`
- ASCDCM-891 Fixed known issue: Warnings reported when service `ResponseOnEvent (0x86)` is not configured
- ASCDCM-892 Fixed known issue: Service `ControlDTCSetting (0x85)` can be configured without any subfunctions
- ASCDCM-872 Fixed known issue: Compiler errors may occur due to invalid code if a protocol with no `DcmDslMainConnection` is configured
- ASCDCM-907 Fixed known issue: Warnings reported when importing the Rte configuration because of an invalid reference value in `DcmIntBeh`
- ASCDCM-880 Fixed known issue: The Dcm may get blocked if it returns from the Bootloader (a Warm Start procedure) and the ComM module's Main Function is scheduled before the Dcm Main Function
- ASCDCM-916 Fixed known issue: Compiler errors due to missing declaration of the name of the multiple configuration container in `BswM_UserCallouts.c`
- Added support for container `DcmDspMemory/AddressAndLengthFormatIdentifier` as described in AUTOSAR Bugzilla RfC #53661
- Implemented `ModeDeclarationGroupPrototype`s for services `DiagnosticSessionControl (0x10)` and `EcuReset (0x11)` and invoked SchM Switch APIs for mode switch notifications

- Implemented `InputOutputControlByIdentifier (2F hex)` service
- Updated handling of service `ReadDTCInformation`, subfunction `reportDTCExtendedDataRecordByDTCNumber (0x19-0x06)` based on configuration parameter `DcmGetSizeOfExtendedDataRecordByDTCOptimization`, considering both AUTOSAR 4.0 rev3 and Bugzilla issue http://www.autosar.org/bugzilla/show_bug.cgi?id=52426

Module version 4.1.0

2012-10-17

- ASCDCM-708 Fixed known issue: The definition of `Dcm_NegativeResponseCodeType` is missing several response code types
- ASCDCM-709 Fixed known issue: Rte Editor reports an error if the Dcm has no Security Levels configured
- ASCDCM-720 Fixed known issue: Compiler errors occur when using certain values for `DcmDspSecurityLevel` and Rte usage is enabled
- ASCDCM-735 Fixed known issue: Compiler error in Dem because of undefined identifier DCM_INITIAL when Rte usage is enabled
- ASCDCM-731 Fixed known issue: Callout function Dcm_Confirmation() always needs to be provided, even if it is not used
- ASCDCM-710 Fixed known issue: Corruption of the response may occur when using the service `ReadDataByIdentifier`
- ASCDCM-732 Fixed known issue: DcmDspRoutineSignalType can be configured to VARIABLE_LENGTH even if `DcmDspRoutineFixedLength` is set to TRUE, resulting in errors during code generation
- ASCDCM-759 Fixed known issue: The parameter `DcmTaskTime` can be configured to 0 seconds
- ASCDCM-679 Fixed known issue: Positive response from UDS 0x10 is incorrect
- Implemented Migration to ASR 4.0 ComStack HandleId Policy
- ASCDCM-755 Fixed known issue: Verification of the Diagnostic Session and Security Level is also done for services `DiagnosticSessionControl (0x10)` and `SecurityAccess (0x27)`
- ASCDCM-777 Fixed known issue: Paged handling of service requests is not handled properly in case some subfunctions of `ReadDTCInformation (0x19)` are not configured
- ASCDCM-763 Fixed known issue: Wrong handling of the return status of the `Dcm_SetProgConditions()` function when returning from bootloader
- ASCDCM-780 Fixed known issue: The signature of the `Dcm_Init()` function is different from that specified in the SWS
- ASCDCM-753 Fixed known issue: If usage of the `DcmDspDataConditionCheckReadFnc()` is not enabled, the ECU may crash due to a NULL pointer access
- Implemented Asynchronous handling of diagnostic services

- The top-level structure of the software-component description in the ARXML files changed from `/AUTOSAR/Dcm` to `/AUTOSAR_Dcm`
- Updated Dcm-Dem interface to R4.0.3
- ASCDCM-811 Fixed known issue: Code generation error when external subfunction handlers are configured
- ASCDCM-778 Fixed known issue: The `DiagnosticSessionControl (0x10)` service cannot switch to Diagnostic Session Identifiers configured within the `DcmDspSession` container
- ASCDCM-794 Fixed known issue: If the configuration parameter `DcmDspDataType` is disabled, compilation fails because invalid data types are generated
- Implemented Migration to ASR 4.0 TP PDU API
- ASCDCM-819 Fixed known issue: Illegal memory access when entering Full Communication mode

Module version 4.0.15

2012-06-21

- ASCDEM-663 Fixed known issue: Dcm does not free locked NvM blocks always
- ASCDCM-683 Fixed known issue: `SwcBswMapping` is located at the wrong location in BSWMD
- ASCDCM-694 Fixed known issue: Generation of `Dcm_RoutineSesType` array not synchronized with DcmDspRoutineInfoConfig
- ASCDCM-682 Fixed known issue: Handling NvM Read operation with `NvM_GetErrorStatus` does not consider all positive cases
- ASCDCM-677 Fixed known issue: Inconsistent SWC-D generation when `DcmDspSecurityUsePort` is configured with certain options
- ASCDCM-654 Fixed known issue: `pMsgContext` can be overwritten before processing last service
- ASCDCM-692 Fixed known issue: Cannot configure `StopRoutine` without `DcmDspRoutineStopIn` and `DcmDspRoutineStopOut`
- ASCDCM-678 Fixed known issue: If only input or output signals are configured for routines, but not both, uncompilable source code is generated
- ASCDCM-643 Fixed known issue: `ClearDiagnosticInformation (0x14)` service execution malfunctions when Rte usage is enabled
- ASCDCM-486 Fixed known issue: Importing a corrected Dcm_swc_interface.arxml file into Tresos Studio may results in uncompilable code after Rte generation
- ASCDCM-707 Fixed known issue: Inconsistent SWC-D generation when `DcmDspSecurityUsePort` is configured with certain options

Module version 4.0.14

2012-05-16

- Initial AUTOSAR 4.0.3 version
- ASCDCM-689 Fixed known issue: Dcm code generation will fail with ERROR 1806 if a DID data configuration has, as Data Access Interface, `DcmDspDataUsePort = 'USE_DATA_SENDER_RECEIVER`

2.2. New features

- Implemented `DcmModeRule` support for general service and subfunction handling
- Implemented `DcmModeRule` support
- Implemented `DcmModeRule` support for controlling individual routines
- Implemented `DcmModeRule` support for reading or writing individual `DcmDspMemory` ranges
- Implemented `DcmModeRule` support for reading, writing and controlling individual DIDs
- Implemented `DcmModeRule` support for re-enabling of DTC setting
- Implemented `DcmModeRule` support for resetting the state of the `CommunicationControl` service
- Implemented configuration parameter support for the DynamicallyDefineDataIdentifier (2C hex) service
- Implemented UDS service ReadDataByPeriodicIdentifier (2A hex)
- Implemented UDS service DynamicallyDefineDataIdentifier (2C hex)
- Implemented support for the configuration parameter `DcmDspDataConditionCheckReadFncUsed`, as described in Autosar Bugzilla RfC #53669
- Implemented ISO14229-1:2013 Response Message Format (UUDT Identifiers) for the diagnostic service ReadDataByPeriodicIdentifier (0x2A) responses.
- Implemented support for TYPE2 periodic responses.
- Implemented support for the `RoutineServices` API according to AUTOSAR 4.2.1 as a per routine selectable alternative to the existing `RoutineServices` API implementation according to AUTOSAR 4.0.-3 The implementation is based on Autosar Bugzilla RFC #57860
- Implemented support for configuration parameter `DcmDspDidRange`.
- Implemented support for the OBD services `$03 (Request emission-related diagnostic trouble codes)`, `$06 (Request on-board monitoring test results for specific monitored systems)`, `$07 (Request emission-related diagnostic trouble codes detected during current or last completed driving cycle)` and `$0A (Request emission-related diagnostic trouble codes with permanent status)`.
- Implemented support for the `SecurityAccess` API according to AUTOSAR 4.2.1 as a configuration selectable alternative to the existing `SecurityAccess` API implementation according to AUTOSAR 4.0.3

- Implemented support for the `DataServices` API according to AUTOSAR 4.2.1 as a configuration selectable alternative to the existing `DataServices` API implementation according to AUTOSAR 4.0.3
- Implemented support for the OBD services `$04 (Clear/reset emission-related diagnostic information)`, according to AUTOSAR 4.4.0 specifications.
- Implemented support for the `DCM_Roe` ClientServerInterface according to AUTOSAR 4.2.1 by

  1. having the `RoeEventId` parameter defined as a Portdefined argument value

  2. changing the name of the generated P-port to `DCM_Roe_{RoeName}`

  as a configuration selectable alternative to the existing `DCM_Roe` ClientServerInterface implementation according to AUTOSAR 4.0.3
- Implemented support for suppressed NRC of functional requests according to ISO14229-2006 and ISO14229-2013 releases.
- Implemented support for the OBD service `$09 (Request Vehicle Information)`
- Implemented support for OBD Service $08 - Request Control of On-Board System, Test or Component
- Implemented Calibration of Dcm Services
- Implemented support for DTCSettingControlOptionRecord and LengthCheck for ControlDTCSetting service request according to AUTOSAR 4.2.1.
- Implemented support for the ReadGenericInformation (0xAF) service according to AUTOSAR 4.4.0, depending on the existence of a valid license.
- Implemented support for reading OBD data identifiers using service ReadDataByIdentifier ($22):

  For range F400..F4FF, the data for the DID is obtained as defined for OBD Service $01 (PID)

  For range F600..F6FF, the data for the DID is obtained as defined for OBD Service $06 (OBD MID)

  For range F800..F8FF, the data for the DID is obtained as defined for OBD Service $09 (InfoType)
- Implemented support for protocol preemption:

  Parameter DcmDslProtocolPreemptTimeout is now supported and required to be higher than 0

  Parameter DcmDslDiagRespOnSecondDeclinedRequest is now supported
- Implemented UUDT transmission and AUTOSAR 4.2.1 - compatible handling for UDS service ReadDataByPeriodicIdentifier (0x2A)
- Updated #defines for symbolic name values according to AUTOSAR 4.0.3 naming schema
- Implemented support for disabling ECU keep awake through active diagnosis after ignition was turned off by introducing the interface `Xxx_SetActiveDiagnostic()` that can be used by the application to inform the Dcm about the desired `ActiveDiagnostic` status. The status `DCM_COMM_ACTIVE` will keep the ECU awake by diagnosis while the status `DCM_COMM_NOT_ACTIVE` will prevent the ECU from being

kept awake by diagnosis. This functionality can be turned on or off by setting the configuration item `DcmDslEnableSetActiveDiagnosis` to true or false.

- Implemented optimization of DID data configuration code

  DID data configuration code now contains only the available operations function pointers

- SecurityAccess service (0x27) returns NRC 0x36 (exceededNumberOfAttempts) one time when SendKey subfunction is requested with an invalid key and the maximum number of failed access attempts is reached.

  The number of failed access attempts is reset after a correct SecurityAccess sequence (successful RequesteSeed followed by successful SendKey)

- Implemented separate ROE Service subfunctions taking in consideration the StorageState bit (sub-function bit 6). The implementation is based on Bugzilla RfC #72061
- Implemented a subset of UDS service 0x86 (ResponseOnEvent) according to AUTOSAR 4.2 specification. The implemented subfunctions are: `0x00 stopResponseOnEvent, 0x04 reportActivatedEvents, 0x05 startResponseOnEvent` for preconfigured events of `OnChangeOfDataIdentifier` type with `Infinite(0x02) eventWindowTime`.
- Implemented separate ROE Service subfunctions taking in consideration the StorageState bit (sub-function bit 6). The implementation is based on Bugzilla RfC #72061
- Implemented full support for the eventWindowTime parameter for the UDS service 0x86 (ResponseOnEvent): Infinity(0x02), CurrentCycle(0x03) and CurrentAndFollowingCycle(0x04)
- Implemented subfunction ClearResponseOnevent(0x06) of service ResponseOnEvent(0x86)
- Implemented RequestDownload (0x34) Service NRC handling according to ISO14229-1:2013.
- Implemented RequestUpload (0x35) Service NRC handling according to ISO14229-1:2013.
- Implemented TransferData (0x36) Service NRC handling according to ISO14229-1:2013.
- Implemented RequestTransferExit (0x37) Service NRC handling according to ISO14229-1:2013.
- Implemented subfunction onDTCStatusChange(0x01) of UDS service ResponseOnEvent(0x86) according to AUTOSAR 4.2 specification.
- Implemented TYPE2 transmission for UDS service ResponseOnEvent(0x86).
- Implemented services RequestDownload (0x34), RequestUpload (0x35), TransferData (0x36), RequestTransferExit (0x37), ReadMemoryByAddress (0x23) and WriteMemoryByAddress (0x3D) according to AUTOSAR 4.3.0.
- Implemented subfunctions enableRxAndDisableTxWithEnhancedAddressInformation (0x04) and enableRxAndTxWithEnhancedAddressInformation (0x05) of service CommunicationControl(0x28).
- Implemented support for security access delay timers according to AUTOSAR 4.3.0
- Allow configuration of external subfunction handlers for UDS service TesterPresent (0x3E). Adapted NRC sequence of UDS service TesterPresent (0x3E) to ISO 14229-1:2013 requirements
- Implemented the capability to configure Memory ranges of Services 0x23/0x3D to be defined with labels

- Implemented Sender/Receiver access for `DcmDspData` for UDS services ReadDataByIdentifier (0x22) and WriteDataByIdentifier (0x2E)
- Implemented endianness interpretation for `DcmDspData` with Sender/Receiver access for UDS services ReadDataByIdentifier (0x22) and WriteDataByIdentifier (0x2E) according to AUTOSAR 4.3 specifications
- Implemented support for BSW distribution, when Dcm and ComM are on different partitions (cores)
- Add support for short response on sub-function 0x01 - onDTCStatusChange for the UDS service ResponseOnEvent (0x86)

  That should provide the following format :

  byte 1: ResponseOnEvent Response SID

  byte 2: eventType

  byte 3: numberOfIdentifiedEvents

  byte 4: eventWindowTime
- Implemented support to enable notification callout for S3 timeout.
- Implemented support for configuration parameter `DcmDspMemoryRangeHighNotIncluded` to allow for more flexibility in the definition of memory ranges
- Implemented support for notification callout on each `Dcm_TpTxConfirmation` call.
- Implemented support for configuration parameter `DcmDsdDisableGenericServiceImplementation` to allow DCM internal service processing in case of project specific service handling.
- Implemented parameter `DcmDslObdNRCResponseSupression` to allow for selectable NRC behaviour when OBD services are requested but not enabled
- Implemented service callbacks interfaces according to AUTOSAR 4.2.2 for WriteMemory and ProcessRequestTransferExit
- Added generic interface to inject service requests from application software.
- Implemented array type signals for the UDS service RoutineControl (0x31) according to AUTOSAR 4.3.0
- Implemented Generic End Of Line addon for Routine Control service (0x31). If `DcmDspRoutineEnableDefaultInterfaces` configuration parameter is set to TRUE, for every routine which is not configured or not used, Dcm module shall call the Default Interfaces for Start, Stop and RequestResults subservices from application.
- Implemented Routine Info Byte handler for Routine Control service (0x31). If `DcmDspRoutineEnableRoutineInfoByte` configuration parameter is set to TRUE and if the requested routine returns E_OK, a byte can be added into the response by calling the interface `Dcm_AddRoutineInfoByte()` defined in Application.
- Implemented Generic End Of Line addon for Data Identifiers for Service ReadDataByIdentifier (0x22), Service WriteDataByIdentifier (0x2E) and InputOutputControlByIdentifier (0x2F). Behaviour to be activat-

ed via vendor-specific configuration flag. If `DcmDspDidEnableDefaultInterfaces` configuration parameter is set to TRUE, for every DID which is not configured or not used, nor in the range of a configured "DcmDspDidRange", the Dcm module shall call the Default Interfaces for the specific request.

- Implemented Parallel UDS and OBD processing/ASR4.3 Dcm/Dem interface with ClientID
- Implemented sub-services reportNumberOfDTCBySeverityMaskRecord(0x07), reportDTCBySeverityMaskRecord(0x08), reportSeverityInformationOfDTC(0x09) for UDS Service ReadDTCInformation (0x19), used to: - retrieve the number of DTCs matching a client defined severity status mask record reportNumberOfDTCBySeverityMaskRecord(0x07); - retrieve a list of DTC severity and functional unit information, which satisfy a client defined severity mask record reportDTCBySeverityMaskRecord(0x08); - retrieve severity and functional unit information for a client defined DTCMaskRecord reportSeverityInformationOfDTC(0x09). The mentioned above subservices are implemented according to AUTOSAR 4.4.0 specifications.
- Implemented sub-service reportNumberOfDTCByStatusMask(0x01) for UDS Service ReadDTCInformation (0x19), used to retrieve the number of DTCs matching a client defined status mask according to AUTOSAR 4.4.0 specifications.
- Implemented sub-service reportDTCByStatusMask(0x02) for UDS Service ReadDTCInformation (0x19), used to retrieve a list of DTCs matching a client defined status mask according to AUTOSAR 4.4.0 specifications.
- Implemented sub-service reportDTCSnapshotIdentification(0x03) for UDS Service ReadDTCInformation (0x19), used to retrieve the DTCSnapshot record identification according to AUTOSAR 4.4.0 specifications.
- Implemented sub-service reportDTCSnapshotRecordByDTCNumber(0x04) for UDS Service ReadDTCInformation (0x19), used to retrieve the DTCSnapshot record data for a client defined DTC mask according to AUTOSAR 4.4.0 specifications.
- Implemented sub-service reportDTCExtDataRecordByDTCNumber(0x06) for UDS Service ReadDTCInformation (0x19), used to retrieve the DTCExtendedData record data for a client defined DTC mask and a client defined DTCExtendedData record number according to AUTOSAR 4.4.0 specifications.
- Implemented sub-service reportSupportedDTC(0x0A) for UDS Service ReadDTCInformation (0x19), used to retrieve the status of all DTCs supported by the server according to AUTOSAR 4.4.0 specifications.
- Implemented sub-service reportFirstTestFailedDTC (0x0B) for UDS service ReadDTCInformation (0x19), used to retrieve the first failed DTC according to AUTOSAR 4.4.0 specifications.
- Implemented sub-service reportFirstConfirmedDTC (0x0C) for UDS service ReadDTCInformation (0x19), used to retrieve the first detected confirmed DTC according to AUTOSAR 4.4.0 specifications.
- Implemented sub-service reportMostRecentTestFailedDTC (0x0D) for UDS service ReadDTCInformation (0x19), used to retrieve the most recent failed DTC according to AUTOSAR 4.4.0 specifications.
- Implemented sub-service reportMostRecentConfirmedDTC (0x0E) for UDS service ReadDTCInformation (0x19), used to retrieve the most recently detected confirmed DTC according to AUTOSAR 4.4.0 specifications.

- Implemented sub-service reportEmissionsOBDDTCByStatusMask(0x13) for UDS Service ReadDTCInformation (0x19), used to retrieve the list of "only emissions-related OBD" DTCs that match a client defined status mask according to AUTOSAR 4.4.0 specifications.
- Implemented sub-service reportDTCFaultDetectionCounter(0x14) for UDS Service ReadDTCInformation (0x19), used to retrieve a list of "prefailed" DTC status according to AUTOSAR 4.4.0 specifications.
- Implemented sub-service reportDTCWithPermanentStatus(0x15) for UDS Service ReadDTCInformation (0x19), used to retrieve a list of DTCs with "permanent DTC" status according to AUTOSAR 4.4.0 specifications.
- Implemented sub-service reportUserDefMemoryDTCByStatusMask(0x17) for UDS Service ReadDTCInformation (0x19), used to retrieve the list of DTCs out of the server's user defined DTC memory that match a client defined DTC status mask according to AUTOSAR 4.4.0 specifications.
- Implemented sub-service reportUserDefMemoryDTCSnapshotRecordByDTCNumber(0x18) for UDS Service ReadDTCInformation (0x19), used to retrieve user defined memory DTCSnapshot record data for a client defined DTC mask and a client defined DTCSnapshotNumber out of the DTC user defined memory according to AUTOSAR 4.4.0 specifications.
- Implemented sub-service reportUserDefMemoryDTCExtDataRecordByDTCNumber(0x19) for UDS Service ReadDTCInformation (0x19), used to retrieve user defined memory DTCExtendedData record data for a client defined DTC mask and a client defined DTCExtendedData record number out of the DTC memory according to AUTOSAR 4.4.0 specifications.
- Implemented support for the UDS Service ClearDiagnosticInformation (0x14), used to clear diagnostic information in one or multiple server's memory according to AUTOSAR 4.4.0 specifications.
- Implemented support for the Request Vehicle Information (0x09) according to AUTOSAR 4.4.0 as a configuration selectable alternative to the existing Request Vehicle Information (0x09) implementation according to AUTOSAR 4.0.3.
- Implemented support for parallel processing of multiple diagnostic requests, the EB Dcm solution provides the possibility of configuring protocol stack number. A protocol stack number in the EB Dcm solution is used in order to define the behavior of processing two protocols. Protocols on the same stack will obey the preemption mechanism. Two protocols on different stacks may run in parallel, unless they are of the same kind (OBD/UDS). In order for two protocols to run in parallel, different stack numbers and different types for the two protocols shall be configured for them. Each protocol has a configured DcmDemClientRef defining the Dem client interacting with the Dem module. This client Id allows the Dem module to distinguish between concurrent calls of the Dcm module of the same function or set of functions to process a certain request. Parallel processing is currently limited to an OBD diagnostic request with an UDS diagnostic request.
- Implemented support for Pdu metadata. Metadata information is retrieved using the EB EcuC APIs: - EcuC_GetMetaDataSourceAddr() - EcuC_GetMetaDataTargetAddr() Metadata information is provided to the lower layer using the EB EcuC APIs: - EcuC_SetMetaDataSourceAddr() - EcuC_SetMetaDataTargetAddr() and also provided to the Xxx_Indication/Xxx_Confirmation operations of the ServiceRequestNotification interface.

- Implemented support for the UDS service `ReadScalingDataByIdentifier(0x24)` according to AUTOSAR 4.0.3 and R20-11.
- Updated the Pdu metadata handling. If the receiving DcmDslProtocolRxAddrType is configured as DCM_PHYSICAL_TYPE then EcuC_SetMetaDataSourceAddr() provides the source address set to the value of the target address of the Pdu used for receiving the diagnostic request provided by the EcuC. If the receiving DcmDslProtocolRxAddrType is configured as DCM_FUNCTIONAL_TYPE then EcuC_SetMetaDataSourceAddr() provides the source address set to the value of to the configuration parameter DcmDspProtocolEcuAddr of the protocol which contains the receive channel where the request which triggered the response was received.
- Implemented support for the UDS service `Authentication(0x29)` according to AUTOSAR R20-11 specification. Service `Authentication(0x29)` provides a means for the client to proof its identity, allowing it to access diagnostic services, which have restricted access.

  Implemented sub-service deAuthenticate (0x00), used to set the authentication state to deauthenticated.

  Implemented sub-service verifyCertificateUnidirectional(0x01), used to allow the client to identify itself as a legitimate client.

  Implemented sub-service proofOfOwnership(0x03), used to verify the client's proof of ownership data and set authentication state to authenticated.

  Implemented sub-service authenticationConfiguration(0x08)), used to retrieve the configured authentication method.
- Implemented `DcmDsdSubServiceUsed` parameter to activate or deactivate the usage of a subservice. This parameter can be used for multi-purpose ECUs. If the subservice is configured and parameter `DcmDsdSubServiceUsed` is set to false, the Dcm module shall trigger a negative response code with NRC 0x12 (subFunctionNotSupported). If the subservice is configured and parameter `DcmDsdSubServiceUsed` is set to true, the Dcm module shall prosess the subservice. If service is `Tester Present(0x3E)` with subservice of (0x00) or (0x80), the Dcm module shall prosess the subservice.
- Implemented sub-service `verifyCertificateBidirectional (0x02)` for UDS `Authentication (0x29)`, used to allow the client to identify itself as a legitimate client and to allow the server to identify itself as a legitimate server.
- Implemented support for the subfunction `reportWWHOBDDTCByMaskRecord(0x42)` for the `UDS service ReadDTCInformation(0x19)` service, according to AUTOSAR R21-11.
- Implemented support for calibration of DIDs via parameter `DcmDspDidUsed`. The configuration parameter `DcmDspDidUsed` shall be set to TRUE for any DIDs that shall be calibratable. Any DIDs which is disabled by configuration (`DcmDspDidUsed` set to FALSE) is not calibaratable. The `DcmDspDidUsed` configuration parameter is just in configuration class pre-compile time. A data identifier that is configured (the configuration parameter `DcmDspDidUsed` is set to TRUE) can be calibrated by changing the corresponding calibration memory section of the data identifier. If the corresponding bit from calibration memory section of a data identifier is set to 0, the DID is not available, in case is set to a value different from 0, the DID is available. The following UDS services are supported: `ReadDataByIdentifier (0x22)`, `Read-`

ScalingDataByIdentifier (0x24), ReadDatabyPeriodicIdentifier (0x2A), DynamicallyDefineDataIdentifier (0x2C), WriteDataByIdentifier (0x2E), InputOutputControl (0x2F).

- Implemented DcmDspDidAvailabilityCalculation parameter to activate or deactivate the auto-calculation of the supported DIDs within the following DID ranges:

  0xF400 - 0xF4FF

  0xF500 - 0xF5FF

  0xF700 - 0xF7FF

2.3. Elektrobit-specific enhancements

This chapter lists the enhancements provided by the module.

- Diagnostic application access without Rte

  Description:

  Diagnostic applications which do not use the AUTOSAR Rte interface are supported. Dcm allows the configuration of user defined handlers for diagnostic services.

  Rationale:

  Compatibility with legacy diagnostic applications.

- Addition of parameter DcmDslProtocolRxPduId

  Description:

  Parameter DcmDslProtocolRxPduId is added as a Vendor Specific parameter. It is not present in the Dcm SWS of AUTOSAR R3.0 and R3.1.

  Rationale:

  This parameter is essential for the functionality of Dcm.

- Provision for handling all session and security levels

  Description:

  The configuration of DcmDspSecurityRow with the parameter DcmDspSecurityLevel value 255 shall be used for configurations which support all security levels. The configuration of DcmDspSessionRow with the parameter DcmDspSessionLevel value 255 shall be used for configurations which can be executed in all sessions.

Rationale:

This makes the configuration more convenient for the user.

- Extension of `Dcm_NegativeResponseCodeType`

  Description:

  The type definition of `Dcm_NegativeResponseCodeType` has been extended to include these values also:

  - `DCM_E_SERVICENOTSUPPORTED (0x11)`
  - `DCM_E_SUBFUNCTIONNOTSUPPORTED (0x12)`
  - `DCM_E_INCORRECTMESSAGELENGTHORINVALIDFORMAT (0x13)`
  - `DCM_E_RESPONSETOOLONG (0x14)`
  - `DCM_E_INVALIDKEY (0x35)`
  - `DCM_E_EXCEEDNUMBEROFATTEMPTS (0x36)`
  - `DCM_E_REQUIREDTIMEDELAYNOTEXPIRED (0x37)`
  - `DCM_E_UPLOADDOWNLOADNOTACCEPTED (0x70)`
  - `DCM_E_TRANSFERDATASUSPENDED (0x71)`
  - `DCM_E_WRONGBLOCKSEQUENCECOUNTER (0x73)`
  - `DCM_E_REQUESTCORRECTLYRECEIVEDRESPONSEPENDING (0x78)`
  - `DCM_E_SERVICENOTSUPPORTEDINACTIVESESSION (0x7F)`

  Rationale:

  AUTOSAR does not specify all the NRCs possible within `Dcm_NegativeResponseCodeType`. It is not present in the Dcm SWS of AUTOSAR R3.0 and R3.1.

- Usage of `DCM_FORCE_RCRRP_OK` in external service handlers

  Description:

  As the `opStatus` parameter `DCM_FORCE_RCRRP_OK` shall not be passed to External service handlers, `DCM_PENDING` shall be passed instead.

  Rationale:

  The Dcm cannot use the `opStatus` parameter `DCM_FORCE_RCRRP_OK` in the API `<Module>_<DiagnosticService>`.

- Asynchronous handling of diagnostic services.

  Description:

The Dcm will be able to handle service execution asynchronously, within another task separate from the Dcm task.

Rationale:

The Dcm module can now execute services which contain routines that may take longer to run than the Dcm task time and may not be interruptable in the way that they can be executed in steps, each step returning a PENDING status.

- Multiple `DcmDslMainConnections` per `DcmDslProtocolRow`.

  Description:

  The Dcm will be able to handle multiple `DcmDslMainConnections` per `DcmDslProtocolRow`.

  Rationale:

  Eliminating the need to duplicate protocol information when multiple connections require the use of the same protocol

- Routine Control variable length signals.

  Description:

  Configuration parameter `DcmDspRoutineVariableLengthInBytes` is added, by which the user can specify whether the variable length signals for the Routine Control are expressed in bits or bytes. This parameter set to 'true' indicates that the variable length signals are expressed in Bytes and the value 'false' indicates signals in bits.

  Rationale:

  Bugzilla entry http://www.autosar.org/bugzilla/show_bug.cgi?id=55779.

- Mixed signal types in `RoutineControl (0x31)` operations

  Description:

  The Dcm will be able to handle mixed signal types (Variable and Fixed length) for input and output signals of the `RoutineControl (0x31)` service operations. The `DcmDspRoutineFixedLength` parameter for a routine shall only control whether the `currentDataLength INOUT` parameter is used in the interfaces generated for the `RoutineControl` operations.

  Rationale:

  Allowing more flexibility in configuring signals used in the interfaces for this service.

- `Dem_DcmCancelOperation()` is called if service `ClearDiagnosticInformation (0x14)` times out

  Description:

`Dem_DcmCancelOperation()` is now called in case the service `ClearDiagnosticInformation (0x14)` times out as a result of too many `REQUESTCORRECTLYRECEIVEDRESPONSEPENDING` NRCs having been sent, in order to cancel pending operations of the Dem.

Rationale:

Allowing an asynchronous `Dem_ClearDTC()` job to be canceled.

- Parameter `DcmDspRoePreconfRxConnectionRef` is now available for preconfiguration of ROE events

Description:

The `DcmDslProtocolRx` referenced by `DcmDspRoePreconfRxConnectionRef` shall be considered as the reception connection on which configuring messages for the preconfigured ROE event would have arrived. This is necessary as the following shall be taken from this connection:

- The `RxPduId` for the preconfigured ROE event that is used to determine the `TxPduId` for sending during Type1 transmissions.
- The ComM channel ID for this transmission.

Rationale:

Ensuring a correct and consistent ROE event preconfiguration.

- Input signals are now allowed for the `requestRoutineResults (0x03)` sub-function of the `RoutineControl(0x31)` diagnostic service

Description:

Input signals are now available for the `requestRoutineResults (0x03)` sub-function of the `RoutineControl(0x31)` diagnostic service, by configuring the `DcmDspRoutineRequestResIn` container within the `DcmDspRoutineInfo` referenced by the routine's `DcmDspRoutine` configuration. The configuration process is identical to the configuration of input signals for the other sub-functions of the service, and results in similar interface signatures.

Rationale:

Allowing greater flexibility in configuring operations for the `RoutineControl(0x31)` service.

- Availability of services in the default session

Description:

The services `SecurityAccess(0x27)`, `CommunicationControl(0x28)`, `ControlDTCSetting(0x85)`, `ReadDataByPeriodicIdentifier(0x2A)` and `InputOutputControlByIdentifier(0x2F)` can now be configured to be available in the default session.

Rationale:

Allowing this configuration in order to provide maximum flexibility. Should an ISO 14229-1 compatible behavior be required, explicitly configure these services to be unavailable in the default session.

- Only one execution at a time of the service handler function is carried out within the context of a separate task when using asynchronous service handling

Description:

Improved the asynchronous processing of service handlers by breaking the continuous processing within a loop inside a separate task into discreet calls to the service handler function. Each call is executed within a separate task instead of the entire service processing being run in one continuous loop.

Rationale:

The asynchronous service handler task is now allowed to yield after each service handler execution cycle , thus allowing tasks of a lower priority than the asynchronous service processor task to also be scheduled, as they will no longer be blocked until the service handling is completed.

- The DIDs in System Supplier Specific range are accessible by WriteDataByIdentifier (0x2E) service.

Description:

The DIDs in System Supplier Specific range (0xFD00..0xFEFF) are accessible by WriteDataByIdentifier (0x2E) service. AUTOSAR does not specify any special handling for this range.

Rationale:

AUTOSAR does not restrict the access to this DID range although the handling is not explicitly stated for the DIDs in this range.

- Transmission latency is reduced by triggering transmission of periodic DID samples as early as possible.

Description:

Improved transmission latency by triggering transmission of available PDID samples in the same cycle previous transmissions are confirmed

Rationale:

To avoid wasting bandwidth the transmission of samples that are waiting in the transmission ring buffers are triggered as soon as possible. This is done as soon as all pending confirmations of previous transmissions are received.

- The protocol for processing a periodic transmission is no longer chosen based on deprecated configuration item DcmDslProtocolTransType.

Description:

The protocol for processing periodic transmissions is now selected based on the presence of DcmDslPeriodicTransmissionConRef or DcmDslROEConnectionRef in the DcmDslMainConnection on which a periodic transmission is configured. The TxPduIds and TxConfirmationPduIds for periodic transmissions too are chosen based on those connection references and the availability of the PduIds (DcmDslPeriodicTxConfirmationPduId and DcmDslPeriodicTxPduRef) in the referenced DcmDslPeriodicTransmission connection, or the equivalent DcmDslResponseOnEvent. for ROE

Rationale:

DcmDslProtocolTransType is deprecated as it is redundant. Additionally the the new method of selecting the periodic transmission protocol provides more flexibility.

- Addition of parameter `DcmDspGenerateOnlyNeededIOControlOperations`

Description:

Parameter `DcmDspGenerateOnlyNeededIOControlOperations` allows to generate only Client-Server Interface of used IOControl operations.

- If `DcmDspGenerateOnlyNeededIOControlOperations` is set to TRUE and any DID that contains a `DcmDspDidSignal` which has client server data access interface and also references a `DcmDspData` element that in turn references a `DcmDspDidInfo` which has the `DcmDspDidControl` container enabled, only necessary Client-Server Interfaces of used IOControl operations are generated.
- If `DcmDspGenerateOnlyNeededIOControlOperations` is set to FALSE and any DID that contains a `DcmDspDidSignal` which has client server data access interface and also references a `DcmDspData` element that in turn references a `DcmDspDidInfo` which has the `DcmDspDidControl` container enabled, all Client-Server Interfaces of IOControl operations are generated.

Parameter `DcmDspGenerateOnlyNeededIOControlOperations` is not present in the Dcm SWS of AUTOSAR.

Rationale:

To avoid generating unrelated Client-Server Interfaces of IOControl operations.

- Preserve data at buffer pointers provided to DID operations across asynchronous operation call sequences

Description:

The signal buffer for input/output operations for DIDs is maintained consistent throughout the sequence of calls constituting an asynchronous operation: Every new call to the interface which is part of the same operation (subsequent to the interface having returned a DCM_E_PENDING error status), the ControlOptionRecord pointer in the case of Xxx_ShortTermAdjustment, or the Data pointer in the case of Xxx_ReadData or Xxx_WriteData, shall have the same address and point to the same data it had when the previous invocation of the operation finished. UDS services ReadDataByIdentifier (0x22), WriteDataByIdentifier (0x2E), InputOutputControlByIdentifier (0x2F) and the periodic DID sampling as a result of service ReadDataByPeriodicIdentifier (0x2A) usage are affected.

Rationale:

Autosar 4.2.2 specifies this behaviour, which is developed in RfC #62898.

- Transmit NRC 0x36 (exceededNumberOfAttempts) for SecurityAccess service (0x27) one time, when SendKey subfunction is requested with an invalid key and the maximum number of failed access attempts is reached

  Description:

  NRC 0x36 (exceededNumberOfAttempts) for SecurityAccess service (0x27) is transmitted one time, when SendKey subfunction is requested with an invalid key and the maximum number of failed access attempts is reached. The next requests will receive NRC RequiredTimeDelayNotExpired until the required delay time passes. The number of failed access attempts is reset after a correct SecurityAccess sequence (successful RequesteSeed followed by successful SendKey).

  Rationale:

  ISO 14229-1 specifies this behaviour.

- When requesting UDS service ReadDataByIdentifier (0x22), if for a DID the DcmDspDataConditionCheck-ReadFnc function or the ConditionCheckRead operation returns DCM_E_NOT_OK and a NRC equal to 0x00, the DID is treated as an unsupported DID.

  Description:

  When requesting UDS service ReadDataByIdentifier (0x22), if the DcmDspDataConditionCheckReadFnc function or the ConditionCheckRead operation for a DID returns DCM_E_NOT_OK and a NRC equal to 0x00, the DID is treated as an unsupported DID. This means that if none of the requested DIDs is supported, the DCM module will transmit a negative response with NRC 0x31 (Request out of range). If at least one DID is supported, the DCM module will transmit a positive response and the DIDs which received DCM_E_NOT_OK will not be part of the response.

  Rationale:

  This is an addition to the Autosar specifications to specify the behavior of the DCM module in case the DcmDspDataConditionCheckReadFnc function or the ConditionCheckRead operation for a DID returns DCM_E_NOT_OK and a NRC equal to 0x00.

- All combinations of signal configurations for all RoutineControl (0x31) operations are now available

  Description:

  This means that all RoutineControl (0x31) operations for any DcmDspRoutine can have any number of (including zero) signals configured for input and/or output.

  Rationale:

This is an addition to the Autosar specifications to specify the behavior of the DCM module for all combinations of signals for all RoutineControl operations, this allows maximum flexibility in configuring routines.

- Diagnostic session checks are performed on subfunctions of service DiagnosticSessionControl too

  Description:

  Although not required by AUTOSAR, to allow for more flexibility the subfunction of UDS service DiagnosticSessionControl can now be assigned diagnostic sessions and they can only be called from those diagnostic sessions.

  Rationale:

  Added flexibility for allowed session transitions.

- DIDs may contain a variable length signal as the last element even if there are more than one signals in that DID.

  Description:

  Older versions of AUTOSAR specified that a DID only may contain a variable-length signal if that signal was the only one contained in that DID. Latest AUTOSAR versions relax that restriction and allow more than one fixed-length signals before the variable-length signal which must be the last one.

  Rationale:

  Offer more flexibility to the user and support the latest Autosar specification.

- The NRC handling for RoutineControl(0x31) service is made according with selected ISO(ISO_14229-2006/ISO_14229-2006).

  Description:

  If ISO_14229-2006 is used and UDS service RoutineControl(0x31) is requested the NRC requestSequenceError(0x24) is return by DCM module according with ISO_14229-2006. If ISO_14229-2013 is used and UDS service RoutineControl(0x31) is requested the next checks needs to be handle by application "routineControlOptionRecord contains valid data for the requested RID", "Condition check" and "Request sequence respected for the RID?" to be according with ISO 14229-1:2013 and AUTOSAR 4.2.1.

  Rationale:

  Offer more flexibility to the user and support the latest AUTOSAR specification.

- The NRC handling for EcuReset(0x11) service improves to produce NRC Conditions Not Correct (0x22) in the event that the result of the switch of the DcmEcuReset mode declaration group is different than SCHM_E_OK.

  Description:

AUTOSAR 4.2.2 SWS does not consider the result of the switch of the DcmEcuReset mode declaration group before triggering the positive response. This improvement introduces the posibilty of producing a negative response whenever conditions are met to do so, and so provide information to the user about the working conditions of the system.

Rationale:

Offer more flexibility to the user and support the latest AUTOSAR specification.

- Introduced a new configuration parameter `DcmDspDiagSesRespMaxNumRespPend`. This offers more flexibility in configuring the maximum amount of `RequestCorrectlyReceivedResponsePending` (0x78) NRCs allowed in a given diagnostic session.

  Description:

  A new configuration parameter `DcmDspDiagSesRespMaxNumRespPend` is introduced for each element of the `DcmDspSessionRow` container. If the parameter is not present (is disabled) the functionality of the Dcm module is unchanged. If the parameter is enabled for a certain session and a request in that session is being processed, the maximum number of the consecutive responses with the NRC RequestCorrectlyReceivedResponsePending(0x78) will depend on the value of the new parameter, i.e. this parameter overrides the value configured in the `DcmDslDiagRespMaxNumRespPend` parameter while the ECU is in the session for which the `DcmDspDiagSesRespMaxNumRespPend` parameter is configured.

  Rationale:

  Offers more flexibility to the user.

- Allowed configuration of DIDs within the 0x0000 - 0x00FF range. Allowed the usage of services ReadDataByIdentifier (0x22) and WriteDataByIdentifier (0x2E) for DIDs in this range.

  Description:

  Some limitations placed upon the readable/writeable DID ranges were quite restrictive for the user. This is why DIDs within the 0x0000 - 0x00FF range are now allowed for configuration and read/write operations even though they are defined as reserved by ISO_14229-2006. The responsibility of making sure that the DIDs are allowed to be accessed in compliance to the ISO specification passes to the user.

  Rationale:

  Offers more flexibility to the user.

- Introduced usage of HandleId wizard for RxPduIds and TxConfirmationPduIds.

  Description:

  Allows the user to generate zero based, ascending and consecutive values for the DcmDslProtocolRxPduId configuration nodes. Allows the user to generate zero based, ascending and consecutive values

for the DcmDslTxConfirmationPduId, DcmDslPeriodicTxConfirmationPduId, DcmDslRoeTxConfirmation-PduId consfiguration nodes that are handled as a group with unique values. This could help the user to prevent entering values manually for the above mentioned configuration parameters. This is especially helpful for large configurations.

Rationale:

Simplify the module configuration for the user.

- Introduce usage of a notification callout for S3 timeout.

  Description:

  Allows the user to enable a callout, to be notified when S3 timeout occurs.

  Rationale:

  The user application is able to track S3 timeout.

- Introduce usage of a notification callout for every transmission confirmation .

  Description:

  Allows the user to enable a callout, to be notified of Dcm_TpTxConfirmation.

  Rationale:

  The user application is able to track order and variability of transmissions.

- Introduce parameter `DcmDsdDisableGenericServiceImplementation`.

  Description:

  This parameter allows to control the availability of the DCM Internal Service Handler for a Service Handler in which the External Service Handler option has been enabled.

  Rationale:

  The user can take advantage of the already available functionality in the context of developing a proprietary implementation of a particular service, by properly making use of this new parameter.

- Introduce new configuration parameter `DcmDslDiagRespMaxNumRespPendInfinityLimit`.

  Description:

  - If the configuration parameter `DcmDslDiagRespMaxNumRespPendInfinityLimit` is set to value 255, then the parameter `DcmDslDiagRespMaxNumRespPend` and `DcmDspDiagSesRespMaxNum-RespPend` can have values from 0 to 255 (inclusive), where the value 255 is treated as the infinity limit for tranmitting NRC 0x78 by the Dcm.

- If the configuration parameter `DcmDslDiagRespMaxNumRespPendInfinityLimit` is set to "65535", then the parameter `DcmDslDiagRespMaxNumRespPend` and `DcmDspDiagSesResp-MaxNumRespPend` can have values from 0 to 65535 (inclusive), where the value 65535 is treated as the infinity limit for tranmitting NRC 0x78 by the Dcm.

Rationale:

To increase the maximum number of requestCorrectlyReceived-ResponsePending (0x78) NRCs to be sent by the Dcm from 254 to 65534.

- Expected behavior of `Dem_DcmGetDTRData()` for OBD service $06 `Request On-Board Monitoring Test-results for Specific Monitored Systems`

Description:

When configuration parameter `DcmOBDMIDSupportType` is set to DCM_OBDMID_SUPPORT_DEM:

- The configuration of OBDMID, TID and UaSID of a DTR is part of Dem configuration.
- Dcm retrieves DTR data from DEM using `Dem_DcmGetDTRData()`.

`Dem_DcmGetDTRData()` shall return E_NOT_OK and Dcm shall not append the reported data to the response message, if:

- Report of DTR result fails.
- DTR status is invisible.

`Dem_DcmGetDTRData()` shall return E_OK and Dcm shall append the reported data to the response message, if report of DTR result is successful:

- If DTR data is available, Dem shall report DTR data along with TID and UaSID and test result with lower and upper limit.
- If DTR data is not available, Dem shall report DTR data along with TID and UaSID and shall set test result with lower and upper limit to 0x00.

Rationale:

ISO-15031-5: Test result with lower and upper limit set to 0x00, when no data is available.

- S3 server handling

Description:

Restrict S3 server handling to requests, which are received on the same DcmDslMainConnection as the one on which the DiagnosticSessionControl (0x10) request was received, which has changed the session to the non-default session.

Rationale:

- AR-94821: Clarify whether requests on other protocols than the one which set the default session have any effect upon the S3 timer

- AR-94194: Clarify Multiple client handling

- Dcm Endianess Conversion

  Description:

  Total signal buffer length argument in Dcm_Dsp_DidSignals_Endianness_Conversion() changed to signal data length (DCM_DATA_MAX_SIZE_BYTE_ARRAY is replaced by ReadContext->SignalLength).

  Rationale:

  - Performance optimization

- DcmDemClientRef multiplicity

  Description:

  Multiplicity of DcmDemClientRef changed from 1 to 0..1

  Rationale:

  - The Dcm should be able to generate and compile with no Dem module/stub present regardless of the availability of UDS Services 0x19, 0x14, 0x85 or OBD Services.

2.4. Deviations

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

- OBD support (reference to product description: ASCPD-33)

  Affected AUTOSAR releases:

  - R4.0 Rev 3

  Description:

  Dcm provides partial support for OBD services.

  Currently not supported OBD DID ranges:

  - 0xF500-0xF5FF
  - 0xF700-0xF7FF

  Currently NOT supported services:

  - Service 0x05 - Request oxygen sensor monitoring test results:

    This Service is not supported by the Dcm.

Note: This is no deviation of the AUTOSAR specification, because it is not required here.

- Service 0x19, Subservices 0x06, 0x10 and 0x19 - report[MirrorMemory|UserDefMemory]DTCExtDataRecordByDTCNumber:

  Mentioned Sub-Services do not support the request of all OBD specific ExtData (0x90-0xEF) using the special Record Number 0xFE.

Requirements:

Dcm474, Dcm475, Dcm476

- Parameter `DcmDslProtocolRxBufferID`

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

Parameter `DcmDslProtocolRxBufferID` is moved to the Rx connection configuration level in the schema tree.

Rationale:

The location as given by AUTOSAR specifies one Rx buffer for all Rx connections of one protocol. This makes it complex to configure the Dcm. Instead, a separate buffer can be configured for each Rx connection.

Requirements:

Dcm701_Conf

- Unsupported AUTOSAR parameters

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

Not all configuration parameters are supported by the Dcm implementation. Unsupported parameters are not editable and a note is added to their description.

Requirements:

Dcm936_Conf, Dcm825_Conf, Dcm600_Conf

- Symbolic port names used instead of numeric names

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

Rte ports are named by their symbolic short name taken from the configuration.

Rationale:

Symbolic port names do not change when ports are deleted or inserted. Also the symbolic name can be chosen to reflect the purpose which makes the port connection process easier and less error prone. But numeric names get renumbered and need to be re-connected.

Requirements:

Section 8.8

- Paged buffered transmission is supported only for service `ReadDTCInformation (0x19)` (reference to product description: ASCPD-41)

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

Paged buffer response transmissions are supported only for 0x19 service.

Requirements:

Dcm028

- No support of RTE Sender-Receiver interfaces for services `InputOutputControlByIdentifier (0x2E)` and `RequestCurrentPowertrainDiagnosticData (0x01)` (reference to product description: ASCPD-110)

Affected AUTOSAR releases:

- R4.0 Rev 3
- R4.3.0

Description:

- RTE Sender-Receiver interface for service `InputOutputControlByIdentifier (0x2F)` is not supported.
- RTE Sender-Receiver interface for service `RequestCurrentPowertrainDiagnosticData ($01)` is not supported.

Requirements:

Dcm639, Dcm638, Dcm640, Dcm718, SWS_Dcm_01121, SWS_Dcm_01122, SWS_Dcm_01123, SWS_Dcm_01124, SWS_Dcm_01125, SWS_Dcm_00687

- DID data access using ECU Signal interfaces (reference to product description: ASCPD-69)

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

Interaction with IoHwAb module for data services is not supported.

Rationale:

A workaround is to use the function callouts generated in case the configured interface is `USE_DATA_SYNCH_FNC`. In this case, the differences is:

- For service `InputOutputControlByIdentifier (0x2F)`, instead of a call to API: `void IoHwAb_Dcm_<EcuSignalName>(uint8 action, <EcuSignalDataType> signal)`, with 'action' set to the operation desired, and in case the 'action' is `shortTermAdjustment`, the 'signal' set to the desired value, a call to one of the User-defined callouts (depending on the desired operation):
  - `Std_ReturnType User_ShortTermAdjustment(uint8* ControlOptionRecord, Dcm_NegativeResponseCodeType* ErrorCode)`
  - `Std_ReturnType User_FreezeCurrentState(Dcm_NegativeResponseCodeType* ErrorCode)`
  - `Std_ReturnType User_ResetToDefault(Dcm_NegativeResponseCodeType* ErrorCode)`
  - `Std_ReturnType User_ReturnControlToEcu(Dcm_NegativeResponseCodeType* ErrorCode)`
- For service `ReadDataByIdentifier (0x22)`, instead of a call to API: `void IoHwAb_Dcm_Read<EcuSignalName>(<EcuSignalDataType>* signal)` with the current signal value to be stored at pointer 'signal', a call to the user-defined callout: `Std_ReturnType User_ReadData(uint8* Data)`

Since all these user-configurable function callouts either share the same signature with the APIs in the IoHwAb module, or the API from the IoHwAb module is refined into four callouts, depending on the 'action' desired, these mechanisms are equivalent and may be used as a valid workaround. The specification (of both Dcm and IoHwAb modules) is also inconsistent in regard to this behavior, as it is nowhere specified where the type `EcuSignalDataType` is to be obtained from.

Requirements:

Dcm578, Dcm580, Dcm626, Dcm627, Dcm628

- Values of `Dcm_StatusType` errors changed for consistency

  Affected AUTOSAR releases:

  - R4.0 Rev 3

  Description:

  The error values defined by the type `Dcm_StatusType` deviate from the ones found in the AUTOSAR R4.0 Rev 3 SWS in Section 8.2.1: Value of error `DCM_E_NOT_OK` is changed to 0x01 from 0x0B. Value of error `DCM_E_COMPARE_KEY_FAILED` is changed to 0x0B from 0x01.

  Rationale:

  To keep the error values consistent between `Dcm_StatusType` and the Rte interface.

  Requirements:

  Section 8.2.1

- Default session needs to be configured always

  Affected AUTOSAR releases:

  - R4.0 Rev 3

  Description:

  The `Default Session` needs to be always configured with a value of 1.

  Rationale:

  Every ECU is initialized to the `Default Session` after power-on. Also the default values of `P2` time and `P2*` time for the ECU are specified along with `Default Session` configuration.

  Requirements:

  Dcm769_Conf, Dcm767_Conf, Dcm765_Conf

- Renamed `DCM_DISABLE_RX_TX_NORMAL` to `DCM_DISABLE_RX_TX_NORM`

  Affected AUTOSAR releases:

  - R4.0 Rev 3

  Description:

Enum DCM_DISABLE_RX_TX_NORMAL of type `Dcm_CommunicationModeType` was renamed and exported as `DCM_DISABLE_RX_TX_NORM`.

Rationale:

To keep uniformity between the names of enums of type `Dcm_CommunicationModeType`. See also AUTOSAR Bugzilla http://www.autosar.org/bugzilla/show_bug.cgi?id=50341.

Requirements:

Section 8.2.6

- Dcm does not handle read access for `dataIdentifier` 0xF186 completely internally.

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

Dcm085 specifies that the read access of the `dataIdentifier` 0xF186 (`ActiveDiagnosticSessionDataIdentifier`) shall be handled internally within DSP. This Dcm implementation contains a recommended configuration which enables the DID 0xF186 as a normal DID and passes it to an internal function providing this information.

Rationale:

Providing this information using regular DID reading mechanisms is the most transparent and flexible solution for the user. Furthermore, there might be use cases in which `dataIdentifier` 0xF186 needs to be handled by the application using `Dcm_GetSesCtrlType()`.

In order to configure DID 0xF186 for internal read access follow the steps below:

- Add DID `0xf186`
- Add a `DcmDspData` with the following configuration:
  - Enable `DcmDspDataReadFnc` with dedicated function name `Dcm_Internalf186Readout`
  - Set `DcmDspDataUsePort` to `USE_DATA_SYNCH_FNC`
  - Configure for `DcmDspDataInfoRef` a "fix data length signal"
  - Configure `DcmDspDataSize` to "8 bit" (one byte) and `DcmDspDataType` to `UINT8`
- Configure a `DcmDspDataInfo` with `DcmDspDataFixedLength` dedicated for configured `DcmDspData`
- Configure a `DcmDspDidInfo` with `DcmDspDidRead` enabled
- Configure `DcmDspDidInfoRef` from DID `0xf186` to the above configured `DcmDspDidInfo`

- Configure only one `DcmDspDidSignal` with `DcmDspDidDataRef` to the above configured `DcmDspData`

Requirements:

Dcm085

- Mandatory configuration of parameter `DcmDspMaxDidToRead`

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

Multiplicity of parameter `DcmDspMaxDidToRead` is changed to 1, so this parameter always has to be configured if the service `ReadDataByIdentifier (0x22)` is used. Also, the range of values for this parameter has been changed to 1 .. 65535.

Rationale:

For implementation of Dcm module, the maximum number of Dids that can be received in a single request of service 0x22 has to be available. See http://www.autosar.org/bugzilla/show_bug.cgi?id=52991.

Requirements:

Dcm638_Conf

- Mandatory configuration of parameter `DcmDspMaxDidToRead`

Affected AUTOSAR releases:

- R4.2 Rev 1

Description:

Multiplicity of parameter `DcmDspMaxDidToRead` is changed to 1, so this parameter always has to be configured if the service `ReadDataByIdentifier (0x22)` is used.

Rationale:

For implementation of Dcm module, the maximum number of Dids that can be received in a single request of service 0x22 has to be available. See http://www.autosar.org/bugzilla/show_bug.cgi?id=52991.

Requirements:

ECUC_Dcm_00638

- Return value of `E_NOT_OK` for `Indication()` operation

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

If the Manufacturer/Supplier `Indication()` operation returns `E_NOT_OK`, with `ErrorCode DCM_E_OK`, the Dcm module shall send a `generalReject` (0x10) negative response code. See configuration parameters `DcmDslServiceRequestManufacturerNotification` and `DcmDslServiceRequestSupplierNotification`.

Rationale:

Requirements Dcm463 and Dcm518 specify that if `E_NOT_OK` is returned from `Indication()`, the Dcm shall send a negative response with NRC value equal to `ErrorCode` parameter value. But if an error code is not set from the function, the Dcm shall send the `generalReject` (0x10) NRC. For all interfaces having `ErrorCode` as out parameter, if interfaces return positive return values, the Dcm does not verify the `ErrorCode`.

Requirements:

Dcm463, Dcm518

- Return value of `DCM_E_REQUEST_NOT_ACCEPTED` for `Indication()` operation

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

If the Manufacturer/Supplier `Indication()` operation returns `DCM_E_REQUEST_NOT_ACCEPTED`, and at least one `RequestCorrectlyReceivedResponsePending (0x78)` NRC has been sent for the UDS service request, the Dcm module shall send a `ErrorCode ConditionsNotCorrect(0x22)` negative response code.

Rationale:

Requirements Dcm462 and Dcm517 specify that if `E_REQUEST_NOT_ACCEPTED` is returned from `Indication()`, the Dcm shall give no response. But if at least one `RequestCorrectlyReceivedResponsePending (0x78)` NRC has been sent for the service request, according with ISO 14229-1, it needs to be followed by a final response.

Requirements:

Dcm462, Dcm517

- Subfunction `reportDTCSnapShotRecordByRecordNumber (0x05)` of service `ReadDTCInformation (0x19)` (reference to product description: ASCPD-81)

  Affected AUTOSAR releases:

  - R4.0 Rev 3

  Description:

  `reportDTCSnapShotRecordByRecordNumber (0x05)` subfunction of UDS service `ReadDTCInformation (0x19)` is not supported.

  Requirements:

  Dcm632, Dcm574, Dcm388, Dcm389

- Subfunction `reportMirrorMemoryDTCByStatusMask (0x0F)` of service `ReadDTCInformation (0x19)` (reference to product description: ASCPD-81)

  Affected AUTOSAR releases:

  - R4.0 Rev 3

  Description:

  `reportMirrorMemoryDTCByStatusMask (0x0F)` subfunction of UDS service `ReadDTCInformation (0x19)` is not supported.

  Requirements:

  Dcm377, Dcm008, Dcm378

- Subfunction `reportMirrorMemoryDTCExtendedDataRecordByDTCNumber (0x10)` of service `ReadDTCInformation (0x19)` (reference to product description: ASCPD-81)

  Affected AUTOSAR releases:

  - R4.0 Rev 3

  Description:

  `reportMirrorMemoryDTCExtendedDataRecordByDTCNumber (0x10)` subfunction of UDS service `ReadDTCInformation (0x19)` is not supported.

  Requirements:

  Dcm297, Dcm295, Dcm296, Dcm478, Dcm386, Dcm382

- Subfunction `reportNumberOfMirrorMemoryDTCByStatusMask (0x11)` of service `ReadDTCInformation (0x19)` (reference to product description: ASCPD-81)

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

`reportNumberOfMirrorMemoryDTCByStatusMask (0x11)` subfunction of UDS service `ReadDTCInformation (0x19)` is not supported.

Requirements:

Dcm376, Dcm293

- Subfunction `reportNumberOfEmissionsRelatedOBDDTCByStatusMask (0x12)` of service `ReadDTCInformation (0x19)` (reference to product description: ASCPD-81)

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

`reportNumberOfEmissionsRelatedOBDDTCByStatusMask (0x12)` subfunction of UDS service `ReadDTCInformation (0x19)` is not supported.

Requirements:

Dcm376, Dcm293

- DcmDsp Container Multiplicity

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

Multiplicity of `DcmDsp` is changed from 0..1 to 1..1. `DcmDsp` must be configured explicitly.

Rationale:

The explicit configuration of `DcmDsp` allows simpler code generation.

Requirements:

Dcm712_Conf

- Behavior related to `ModeDeclarationgroupPrototype` for UDS service LinkControl (0x87) (reference to product description: ASCPD-70)

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

Features related to `ModeDeclarationgroupPrototype` for UDS service LinkControl (0x87) are not supported.

Requirements:

Dcm533

- Service `LinkControl (0x87)` functionality implemented as callout (reference to product description: ASCPD-81)

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

Dcm invokes a callout after receiving service request for Link control service. The integrator will provide the implementation part of the callout.

Rationale:

Functionality supported as a callout.

Requirements:

Dcm744, Dcm531

- BswM interface is not compliant to ASR 4.0.3

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

BswM interfaces used in Dcm module are conformant to ASR release R4.0.2

Requirements:

Section 8.7.2

- Structure of the `Dcm_ProgConditionsType` data type is changed

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

`Dcm_ProgConditionsType` has been changed to:

Name: `Dcm_ProgConditionsType`
Type: structure
Element:
`uint16 TesterSourceAddr`: Tester source address configured per protocol
`uint8 ProtocolId`: Id of the protocol on wich the request has been received
`uint8 Sid`: Service identifier of the received request
`uint8 SubFncId`: Identifier of the received subfunction
`boolean ReprogramingRequest`: Set/Clear the `ReprogramingRequest` flag on ECU reset
`boolean ApplUpdated`: Set/Clear the `ApplUpdated flag` on ECU reset
`boolean ResponseRequired`: Set/Clear the `ResponseRequired` flag on ECU reset
Description:
Used in `Dcm_SetProgConditions()` to allow the integrator to store relevant information prior to jumping to bootloader.

Rationale:

The `DcmDslProtocolRxTesterSourceAddr` is a 16-bit integer, therefore an 8-bit integer is insufficient to hold this information. This is the solution for the ongoing Bugzilla #55594 discussion.

Requirements:

Section 8.2.14

- Interface DataServices_DIDRange_{Range}

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

Interface `DataServices_DIDRange_{Range} is supported according to AUTOSAR 4.2.-1: ClientServerInterface` Operation `IsDidAvailable` receives a parameter OpStatus and can be asynchronous. `DataServices` callout function `Xxx_IsDidAvailable` receives a parameter OpStatus and can be asynchronous. `DataServices` callout functions `Xxx_ReadDidData` and `Xxx_WriteDid-Data` can be asynchronous. DID Range checks are not supported in this implementation.

Rationale:

The specification has been reworked / refined.

Requirements:

Section 8.7.3.3, Dcm769, Dcm803, Dcm804, Dcm805

- Support of Dids configured via container `DcmDspDidRange`

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

Dids configured via container `DcmDspDidRange` are not supported in all diagnostic services.

The following diagnostic services return NRC 0x31 REQUEST OUT OF RANGE if a requested Did is not configured via `DcmDspDid`:

- 0x2F IOControl
- 0x2A ReadDataByPeriodicIdentifier
- 0x2C DynamicallyDefineDataIdentifier

Rationale:

These internal service handlers do not support the asynchronous execution of the availability check, yet.

Requirements:

Section 8.7.3.3, Dcm769, Dcm803, Dcm804, Dcm805

- Multiplicity of `DcmDspMemoryIdInfo`

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

Multiplicity of the container `DcmDspMemoryIdInfo` is changed from 1..* to 0..*, as the parameter and all the sub-containers are optional.

Rationale:

To optimize the configuration, multiplicity is changed. Refer to AUTOSAR Bugzilla http://www.autosar.org/bugzilla/show_bug.cgi?id=56265.

Requirements:

Section 10.2.39, Dcm911_Conf

- Configuration parameter `DcmDslProtocolRxComMChannelRef` is implemented as `CHOICE-REFERENCE`

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

The configuration parameter `DcmDslProtocolRxComMChannelRef` is implemented as `CHOICE-REFERENCE` instead of `SYMBOLIC-NAME-REFERENCE`. This is required by the parameter's entry in AUTOSAR_MOD_ECUConfigurationParameters.arxml

Rationale:

This parameter was changed to a `CHOICE-REFERENCE` type in order to allow the ACG6.4.0 Dcm module to be used with the ACG6.3.0 ComM module. This allows the user to switch between the ACG6.3.0 or the ACG6.4.0 versions of the paths for the `ComMChannel` parameter in the ComM module. As one of these paths is always valid, the Dcm is compatible with both versions of the ComM module.

Requirements:

Dcm906_Conf

- Configuration parameters `DcmDspAllComMChannelRef` and `DcmDspSpecificComMChannelRef` and `DcmDspComControlSubNodeComMChannelRef` are implemented as `CHOICE-REFERENCE`.

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

The configuration parameters `DcmDspAllComMChannelRef` and `DcmDspSpecificComMChannelRef` and `DcmDspComControlSubNodeComMChannelRef` are implemented as `CHOICE-REFERENCE` instead of `SYMBOLIC-NAME-REFERENCE`. This is required by the parameter's entry in AUTOSAR_MOD_ECUConfigurationParameters.arxml.

Rationale:

This parameter was changed to a `CHOICE-REFERENCE` type in order to allow the ACG6.4.0 Dcm module to be used with the ACG6.3.0 ComM module. This allows the user to switch between the ACG6.3.0 or the ACG6.4.0 versions of the paths for the `ComMChannel` parameter in the ComM module. As one of these paths is always valid, the Dcm is compatible with both versions of the ComM module.

Requirements:

Dcm902_Conf, Dcm904_Conf, ECUC_Dcm_01030

- Functionality of services `ControlDTCSetting (0x85)` is executed before the response is sent

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

The AUTOSAR standard R4.0 Rev 3 defines that the functionality related to the services `ControlDTCSetting (0x85)` is executed if the send confirmation function for service responses is invoked. The current Dcm has the following implemented:

- The Dcm module does not invoke `Dem_EnableDTCSetting()` if the DSD calls the send confirmation function for the response to a request for the service `ControlDTCSetting` with `DTCSettingType = on`.
- The Dcm module does not invoke `Dem_DisableDTCSetting()` if the DSD calls the send confirmation function for the response to a request for the service `ControlDTCSetting` with `DTCSettingType = off`.
- The Dcm module does not invoke a switch of the `ModeDeclarationGroupPrototype DcmControlDTCSetting` to ModeDeclaration `ENABLEDTCSETTING` if the DSD calls the send confirmation function for the response to a request for the service `ControlDTCSetting (0x85)` with `DTCSettingType = on`.
- The Dcm module does not invoke a switch of the `ModeDeclarationGroupPrototype DcmControlDTCSetting` to ModeDeclaration `DISABLEDTCSETTING` if the DSD calls the send confirmation function for the response to a request for the service `ControlDTCSetting (0x85)` with `DTCSettingType = off`.

Instead, this functionality is executed before the responses to the requests are transmitted.

Rationale:

In this Dcm implementation, the above mentioned functionality is caused by a bug.

Requirements:

Dcm304, Dcm406, Dcm783, Dcm784

- Configuration parameter `CONTEXT` within `DcmBswModeRef` and `DcmSwcModeRef` is disabled

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

The configuration parameter CONTEXT within DcmBswModeRef and DcmSwcModeRef is omitted and not used to get the ModeDeclarationGroupPrototype.

Rationale:

Configuration parameters DcmBswModeRef and DcmSwcModeRef refer to the ModeDeclaration path within their TARGET and ModeDeclarationGroupPrototype path within CONTEXT. ModeDeclarationGroupPrototype can be obtained from the path provided by TARGET itself so that CONTEXT is not required.

Requirements:

Dcm931_Conf, Dcm930_Conf

- Dcm_Init() parameter ConfigPtr updated with const qualifier

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

A const qualifier is missing for the input parameter ConfigPtr of Dcm_Init(). This parameter is used during post-build time.

Rationale:

The missing const qualifier for the ConfigPtr is a bug in the AUTOSAR R4.0 Rev 3 Dcm SWS that has been corrected in the AUTOSAR R4.1 Rev 1 Dcm SWS. Refer to AUTOSAR Bugzilla http://www.autosar.org/bugzilla/show_bug.cgi?id=52382.

Requirements:

Dcm037

- Dcm_Init() need not be called before Dcm_GetVersionInfo()

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

If the development error detection is enabled, for a request of Dcm_GetVersionInfo, it is not necessary that the Dcm module is already initialized.

Rationale:

Version check of the module could be performed prior to a module initialization. Refer to BswM_C_207.

Requirements:

Dcm043

- Parameter `DcmTxPduId` in `Dcm_CopyTxData()` is not the same as in `PduR_DcmTransmit()`

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

`Dcm_CopyTxData()` uses parameter `DcmTxPduId` which is the Tx PduId configured within the Dcm `DcmDslTxConfirmationPduId`.

Rationale:

The PduR shall call `Dcm_CopyTxData()` through a handle ID defined by the Dcm, according to the handle ID concept in AUTOSAR_TPS_ECUConfiguration.pdf. The PduR shall not call `Dcm_CopyTxData()` through a handle ID defined by the PduR. Refer to AUTOSAR Bugzilla http://www.autosar.org/bugzilla/show_bug.cgi?id=57991.

Requirements:

Dcm350

- Internal service handlers do not call `Dcm_ExternalProcessingDone`

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

Internal service handlers do not call `Dcm_ExternalProcessingDone()`. It is called only from external service handlers.

Rationale:

This does not effect the functionality of service handlers.

Requirements:

Dcm269, Dcm733, Dcm225

- No AUTOSAR debugging support

Description:

AUTOSAR debugging support is not available.

Requirements:

Dcm484, Dcm485, Dcm486, Dcm487, Dcm506, Dcm507, Dcm508, Dcm509

- Link to service table not done during run-time at protocol initialization

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

For each configured protocol, a link to the corresponding service table shall be done statically in the code generation step.

Requirements:

Dcm035

- Endianness conversion supported for all interface types

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

Endianness conversion is performed on data that belongs to `DcmdspData` elements with `DcmDspDataUsePort` configured to `USE_DATA_SYNCH_FNC`, `USE_DATA_ASYNCH_FNC`, `USE_DATA_SYNCH_CLIENT_SERVER`, `USE_DATA_ASYNCH_CLIENT_SERVER`. Also the conversion is not only performed on data that belongs to `DcmDspData` elements with `DcmDspDataUsePort` configured to `USE_DATA_SENDER_RECEIVER` or `USE_ECU_SIGNAL`. Note: For non-support of Sender-Receiver and IoHwAb interfaces themselves, refer to the related deviation entry 'No support of RTE Sender-Receiver interface for services `ReadDataByIdentifier (0x22)` and `WriteDataByIdentifier (0x2E)`' and 'DID data access using ECU Signal interfaces'.

Rationale:

Endian conversion is valid for all interface types.

Requirements:

Dcm638, Dcm639, Dcm640, Dcm718, Dcm716

- Parameter `DcmRespondAllRequest`

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

Parameter `DcmRespondAllRequest` is not supported in this implementation.

Requirements:

Dcm084

- Return value of `E_NOT_OK` for `Confirmation()` operation that belongs to the `ServiceRequestNotification` interface

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

The `Confirmation()` operation is called after sending the NRC or a positive response. Thus it does not set any `ErrorCode`.

Rationale:

The `ErrorCode` parameter is not present in the `Xxx_Confirmation()` operation.

Requirements:

Dcm678

- Security level, session type and active protocol interfaces return `E_NOT_OK`

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

For the following interfaces the `E_NOT_OK` error code can be returned in case the DET is enabled and the interfaces are called before a Dcm initialization or are called with invalid parameters:

- `Dcm_GetSecurityLevel`
- `Dcm_GetSesCtrlType`
- `Dcm_GetActiveProtocol`

In case the Dcm is started after the application, the Det error-checking needs to be enabled to ensure runtime integrity. This applies for the production code.

Rationale:

Requires clarification from Bugzilla http://www.autosar.org/bugzilla/show_bug.cgi?id=54135

Requirements:

Dcm338, Dcm339, Dcm340

- No jump to bootloader if transmission of NRC 0x78 fails

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

If the jump to bootloader is requested with the configuration parameter `DcmSendRespPendOnTransToBoot` set to 'true' but the NRC 0x78 (Response Pending) response is not sent successfully, the Dcm shall cancel the current request. A jump to the bootloader shall not be performed.

Rationale:

The jump should only be performed when the transmission is successful and not only confirmed. Refer to AUTOSAR Bugzilla http://www.autosar.org/bugzilla/show_bug.cgi?id=57856.

Requirements:

Dcm535

- `RxBufferSizePtr` of `Dcm_StartOfReception` and `Dcm_CopyRxData()` if `SduLength` from `PduInfoPtr` equals 0

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

If `Dcm_StartOfReception` is invoked with `TpSduLength` equal to 0, the value `BUFREQ_OK` shall be returned and `RxBufferSizePtr` shall be set to the configured size of the allocated Rx buffer. If `Dcm_CopyRxData` is invoked with `SduLength` from `PduInfoPtr` equal to 0, the value `BUFREQ_OK` shall be returned and `RxBufferSizePtr` shall be filled with the remaining size of the Rx buffer.

Rationale:

Dcm642 does not mention that the size to be returned should be the remaining size of the Rx buffer and not the allocated size itself in case of `Dcm_CopyRxData` if `SduLength` equals 0. Refer to AUTOSAR Bugzilla http://www.autosar.org/bugzilla/show_bug.cgi?id=56890.

Requirements:

Dcm642

- Additional configuration parameters added to the container `DcmDspSecurityRow`

  Affected AUTOSAR releases:

  - R4.0 Rev 3

  Description:

  `DcmDspSecurityUsePort`, `DcmDspSecurityGetSeedFnc`, and `DcmDspSecurityCompareKeyFnc` are added to the `DcmDspSecurityRow` container. `DcmDspSecurityUsePort` defines the interface which shall be used to access the data, either `USE_SECURITY_ASYNCH_FNC`, `USE_SECURITY_SYNCH_FNC`, `USE_SECURITY_SYNCH_CLIENT_SERVER`, or `USE_SECURITY_ASYNCH_CLIENT_SERVER` for the service `SecurityAccess (0x27)`. Based on the value of `DcmDspSecurityUsePort` (`USE_SECURITY_ASYNCH_FNC`, `USE_SECURITY_SYNCH_FNC`), additional parameters `DcmDspSecurityGetSeedFnc` and `DcmDspSecurityCompareKeyFnc` are introduced.

  Rationale:

  Refer to AUTOSAR Bugzilla http://www.autosar.org/bugzilla/show_bug.cgi?id=53066.

  Requirements:

  Dcm759_Conf

- Additional configuration parameters added to the container `DcmDspMemory`

  Affected AUTOSAR releases:

  - R4.0 Rev 3

  Description:

  Container `DcmDspAddressAndLengthFormatIdentifier` and list `DcmDspSupportedAddressAndLengthFormatIdentifier` are added in the `DcmDspMemory` container.

  Rationale:

  Refer to AUTOSAR Bugzilla http://www.autosar.org/bugzilla/show_bug.cgi?id=53661.

  Requirements:

  Dcm784_Conf

- Function interface not specified

  Affected AUTOSAR releases:

  - R4.0 Rev 3

  Description:

The operations mentioned in the requirements Dcm793, Dcm794, Dcm796, Dcm797, Dcm799, Dcm800, Dcm801 and Dcm802 should use synchronous or asynchronous function interface operations instead of a client server interface. The signatures of operations for a client server interface is already mentioned in Dcm686.

Rationale:

This implements the solution for the issue reported in http://www.autosar.org/bugzilla/show_bug.cgi?id=61514. The isssue is introduced in http://www.autosar.org/bugzilla/show_bug.cgi?id=54767, as in the implementation Callout interfaces and ClientServer interfaces share the same signature.

Requirements:

Dcm793, Dcm794, Dcm796, Dcm797, Dcm798, Dcm799, Dcm800, Dcm801, Dcm802

- `Dcm_GetProgConditions()` called in `Dcm_Init()`

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

In order for the Dcm to know if its initialization is the consequence of a jump from the bootloader, it shall call `Dcm_GetProgConditions()` in `Dcm_Init()`. `Dcm_GetProgConditions()` shall not be called during the first call to `Dcm_MainFunction()`, as mentioned in the requirement Dcm536.

Rationale:

Refer to AUTOSAR Bugzilla http://www.autosar.org/bugzilla/show_bug.cgi?id=61670

Requirements:

Dcm536

- Support for `RoutineControl` services with fixed signals if `DcmDspRoutineFixedLength` is set to false

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

In contrast to the definitions for the `Start`, `Stop`, and `RequestResults` interfaces mentioned in the requirement Dcm690 (AUTOSAR SWS R4.0 Rev 3) it is not mandatory for their signatures to contain `VARIABLE_LENGTH` input and output signals if `DcmDspRoutineFixedLength` is set to false.

Requirements:

Dcm690

- Signature of expected interfaces for the `RequestRoutineResults` operation of service `RoutineControl (0x31)`

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

In addition to the definition for the `RequestResults` interface mentioned in the requirement Dcm690 (AUTOSAR SWS R4.0 Rev 3), the `RequestRoutineResults` operation can also accept input signals. Therefore, the signature for this interface may additionally have `IN <datatype> dataIn` or `IN uint8 dataInN[(<DcmDspRoutineSignalLength of DcmDspStartRoutineInSignal> +7)/8]` input parameters similar to the `Start` and `Stop` operations.

Requirements:

Dcm690

- If the configuration parameter `DcmDataTransferServicesASRVersion` is set to the value DATA_TRANSFER_SERVICES_AUTOSAR_403, the NRC `GeneralReject (0x10)` is triggered if `Dcm_ReadMemory()` returns DCM_READ_FAILED.

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

If the configuration parameter `DcmDataTransferServicesASRVersion` is set to the value DATA_TRANSFER_SERVICES_AUTOSAR_403 and if the call to `Dcm_ReadMemory` returns DCM_READ_FAILED, the DCM module shall trigger a negative response with NRC `GeneralReject (0x10)`, instead of `GeneralProgrammingFailure (0x72)` according with RFC #57196.

Requirements:

Dcm644

- The `Dem_SetDTCFilter()` API is not called from the context of the `Dcm_MainFunction()` when the service `ReadDTCInformation (0x19)` is requested with the subfunction `ReportDTCFaultDetectionCounter (0x14)`

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

In case asynchronous service processing is enabled for service `ReadDTCInformation (0x19)` (parameter `DcmAsyncServiceExecution` is set to 'true'), and the subfunction `ReportDTCFaultDetectionCounter (0x14)` is enabled and configured to use the EB-supplied implementation, then the `Dem_SetDTCFilter()` API is not called from the context of the `Dcm_MainFunction()`.

Requirements:

Dcm.ReportFaultDetectionCounter.Async

- Size calculation of ImplementationDataType DataArrayTypeUint8_{Data}

Affected AUTOSAR releases:

- R4.2 Rev 1

Description:

The size of the Array type is calculated wrong. The right size should be:

- ({ecuc(Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDataSize)}+7)/8 respective
- ({ecuc(Dcm/DcmConfigSet/DcmDsp/DcmDspPid/DcmDspPidData/DcmDspPidDataSize)}+7)/8

Rationale:

The type DataArrayTypeUint8_{Data} shall store a count of bits as specified in either of the following:

- 'count of bits' = {ecuc(Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDataSize)}
- 'count of bits' = {ecuc(Dcm/DcmConfigSet/DcmDsp/DcmDspPid/DcmDspPidData/DcmDspPidDataSize)}

For this a count of bytes is required. It is calculated as follows: 'count of bytes' = ('count of bits'+7)+8

Requirements:

SWS_Dcm_01121

- The parameter `DcmDspTestResultObdmidTidRef` is not supported

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

Instead a vendor specific container `DcmDspTestResultObdmidTidRefs` was introduced. See AutoCore_Generic_Diagnostic_Stack_documentation.pdf chapter DcmDspTestResultObdmidTidRefs.

Requirements:

Dcm685_Conf

- Signature of expected interfaces for the `RequestRoutineResults` operation of service `RoutineControl (0x31)`

  Affected AUTOSAR releases:

  - R4.2 Rev 1
  - CP Release 4.3.0

  Description:

  In addition to the definition for the `RequestResults` interface mentioned in the requirement SWS_Dcm_00690 (AUTOSAR SWS R4.2 Rev 1 and AUTOSAR SWS CP Release 4.3.0), the `RequestRoutineResults` operation can also accept input signals. Therefore, the signature for this interface may additionally have `IN <datatype> dataIn` or `IN uint8 dataInN[(<DcmDspRoutineSignalLength of DcmDspStartRoutineInSignal> +7)/8]` input parameters similar to the `Start` and `Stop` operations.

  Requirements:

  SWS_Dcm_00690

- Storing the dynamic length of periodicDIDs signals at the time of configuration through UDS service `ReadDataByPeriodicIdentifier (0x2A)`

  Affected AUTOSAR releases:

  - R4.2 Rev 1

  Description:

  Requirement SWS_Dcm_01099 (AUTOSAR SWS R4.2 Rev 1) states that upon configuring a periodicDID, with a configured dynamic length the Dcm module shall invoke the ReadDataLength operation (or the respective C-Function) to retrieve the length of the periodicDID and then use and consider that length valid for each ReadData operation till the periodicDID is removed from the scheduler or updated via a new request. That length shall further be used to check against the UUDT size.

  The current implementation does not support this. The length of the Data is read each time the data is retrieved and not just once at configuration time. The length is also not used to check against UUDT size.

  Requirements:

  SWS_Dcm_01099

- Value of IN-parameters of interfaces when being called with an `OpStatus` equal to `DCM_PENDING`, `DCM_CANCEL` or `DCM_FORCE_RCRRP_OK`

  Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

The Dcm does not clear all the input parameters of the following interfaces:

- `<Module>_<DiagnosticService>()`
- `<Module>_<DiagnosticService>_<SubService>()`
- `Dcm_WriteMemory()`

on calls with a `DCM_PENDING`,`DCM_CANCEL` or `DCM_FORCE_RCRRP_OK` `OpStatus`.

Rationale:

For `<Module>_<DiagnosticService>()` and `<Module>_<DiagnosticService>_<SubService>()`, this would not allow the `pMsgContext` input data (containing the response buffers) to be used by subsequent calls of the service or subfunction handler (and would require it to be memorized by the service handler in the first call).

For `Dcm_WriteMemory()`, this allows the Application to read data from the `*MemoryData` input buffer, thus making it not necessary to copy all this data into an own buffer. This improves memory usage.

The fact that these parameters are not cleared should not actually cause any problems. The Application should assume that they are cleared as per the SWS if the SWS-compliant behavior is required, and therefore ignore their values on subsequent calls with `OpStatus` values of `DCM_PENDING`, `DCM_CANCEL` or `DCM_FORCE_RCRRP_OK`.

Requirements:

Dcm763, Dcm764, Dcm540

- Updated name of AUTOSAR parameter

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

In the AUTOSAR release R4.0 Rev 3, the configuration parameter `DcmDslPeriodicTranmissionConRef` has a typo. This is changed in AUTOSAR release R4.1 Rev 1 based on Rfc 53221 into `DcmDslPeriodicTransmissionConRef`. This typo fix is included in this implementation.

Requirements:

Dcm707_Conf,

- Dcm_DemTriggerOnDTCStatus interfaces return `E_NOT_OK`

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

For the Dcm_DemTriggerOnDTCStatus interface the `E_NOT_OK` error code can be returned in case the DET is enabled and the interfaces are called before a Dcm initialization. In case the Dcm is started after the application, the Det error-checking needs to be enabled to ensure run-time integrity. This applies for the production code.

Rationale:

All DCM interfaces return E_NOT_OK if DCM module is not initialized and Det error-checking is enabled.

Requirements:

SWS_Dcm_00614

- Dcm_ResetToDefaultSession interfaces return `E_NOT_OK`

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

For the Dcm_ResetToDefaultSession interface the `E_NOT_OK` error code can be returned in case the DET is enabled and the interfaces are called before a Dcm initialization. In case the Dcm is started after the application, the Det error-checking needs to be enabled to ensure run-time integrity. This applies for the production code.

Rationale:

All DCM interfaces return E_NOT_OK if DCM module is not initialized and Det error-checking is enabled.

Requirements:

Dcm520

- Dcm_CopyTxData interfaces return `BUFREQ_E_BUSY`

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

For the Dcm_CopyTxData interface the `BUFREQ_E_BUSY` error code can be returned in case the request could not be fulfilled, because the required amount of Tx data is not available.

Requirements:

Dcm092

- Dcm_DemTriggerOnDTCStatus always returns `E_NOT_OK`

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

For the Dcm_DemTriggerOnDTCStatus interface the `E_NOT_OK` error code is always returned.

Rationale:

The onDTCStatusChange subfunction of the ResponseOnEvent service is not implemented yet.

Requirements:

SWS_Dcm_00614

- The UDS service `ResponseOnEvent(0x86)` is implemented according to AUTOSAR 4.2.1 specification

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

The UDS service `ResponseOnEvent(0x86)` is implemented according to AUTOSAR 4.2.1 specification. The ROE specification from the AUTOSAR 4.0.3 specification was significantly reworked in the new release of the document.

Requirements:

Dcm604, Dcm605, Dcm710, Dcm606, Dcm607, Dcm608, Dcm609, Dcm610, Dcm611, Dcm612, Dcm613, Dcm619, Dcm859_Conf, Dcm780_Conf, Dcm782_Conf, Dcm781_Conf, Dcm933_Conf, Dcm603, Dcm582, Dcm524, Dcm523, Dcm137, Dcm597, Dcm618, Dcm712, Dcm713, Dcm714, Dcm731, Dcm748, Dcm765, Dcm772, Dcm730, Dcm521, Dcm679, Dcm852_Conf, Dcm853_Conf, Dcm854_Conf, Dcm855_Conf, Dcm857_Conf, Dcm934_Conf, Dcm162, Dcm695

- The UDS service `ResponseOnEvent(0x86)` is not fully implemented.

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

The UDS service `ResponseOnEvent(0x86)` according to AUTOSAR 4.2.1 specification is implemented in two phases. The requirements included in this deviation will be implemented in phase 2.

Requirements:

SWS_Dcm_00873.OnChangeOfDID, SWS_Dcm_00893.OnChangeOfDID, SWS_Dcm_00907, SWS_Dcm_00912, SWS_Dcm_00913, SWS_Dcm_00918, SWS_Dcm_00892.onChangeOfDataIdentifierNonPersistent, SWS_Dcm_00892.onChangeOfDataIdentifierPersistent

- Number of fixed length signals before a dynamic length signal

Affected ISO 14229-1 releases:

- ISO14229-1:2013

Description:

Dcm620 states that the data of a DID can have dynamic datalength only if this DID contains only one data. See configuration parameter `DcmDspDataFixedLength`. The latest AUTOSAR specification relaxes that requirement to constr_6039, which says: Signals with variable datalength: Only the last signal (DcmDspDidSignal) of a DID can have variable datalength (DcmDspDataType is set to UINT8_DYN). The implementation supports more than one signal of fixed length with the last signal allowed to be of dynamic length.

Requirements:

Dcm620

- The `DCM_FORCE_RCRRP_OK` value for the parameter `OpStatus` for the fallowing interfaces is not supported due to inconsistent requirements:
  - `Dcm_ProcessRequestDownload`
  - `Dcm_ProcessRequestUpload`
  - `Dcm_ProcessRequestTransferExit`

Affected AUTOSAR releases:

- R4.0 Rev 3
- CP Release 4.3.0

Rationale:

AUTOSAR 4.0.3 specifies `DCM_FORCE_RCRRP_OK` as a valid `OpStatus` value for the interfaces mentioned above. But AUTOSAR 4.3.0 specifies it as valid only for `Dcm_ProcessRequestUpload`. Moreover, `DCM_E_FORCE_RCRRP` is not listed as a required supported return value for `Dcm_ProcessRequestDownload`, `Dcm_ProcessRequestUpload`, and `Dcm_ProcessRequestTransferExit` in neither AU-

TOSAR 4.0.3 or AUTOSAR 4.3.0. Without this specific error code there would not be any subsequent call of these interfaces with OpStatus `DCM_FORCE_RCRRP_OK`.

Dcm754, Dcm756, Dcm755

- Multiplicity of `DcmDslProtocolTx`

Affected AUTOSAR releases:

- CP Release 4.3.0

Description:

Multiplicity of the container `DcmDslProtocolTx` is changed from from 1 to 0..1, as the container is redefined to be optional.

Rationale:

To enable response suppression in case a response to a service request is not allowed or possible. See AUTOSAR Bugzilla https://www.autosar.org/bugzilla/show_bug.cgi?id=64765.

Requirements:

ECUC_Dcm_00711

- Reset of a security level's specific AttemptCounter

Affected ISO 14229-1 releases:

- ISO14229-1:2013

Description:

SWS_Dcm_01357 allows a security level's specific AttemptCounter to be reset on either a successful sendKey subfunction request or an expired SecurityDelayTimer. However, ISO14229-1:2013 says nothing about the AttemptCounter being reset when the delay timer expires. In order to be ISO compatible, SWS_Dcm_01357 is deviated so that the AttemptCounter is not reset upon an expired SecurityDelayTimer. SWS_Dcm_01357 compliance may still be enabled via the vendor-specific parameter DcmDspSecurityResetAttemptCounterOnTimeout that should be configured to TRUE.

Requirements:

SWS_Dcm_01357

- ECU path used in ImplementationDataType DataArrayType{DataType}_{Data}

Affected AUTOSAR releases:

- R4.3.0

Description:

The mentioned ECUC paths should be:

- {ecuc(Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDataByteSize)}
- {ecuc(Dcm/DcmConfigSet/DcmDsp/DcmDspPid/DcmDspPidData/DcmDspPidDataByteSize)}
- {ecuc(Dcm/DcmConfigSet/DcmDspData/DcmDspDataType)}
- {ecuc(Dcm/DcmConfigSet/DcmDsp/DcmDspPid/DcmDspPidData/DcmDspPidService01/DcmDspPidDataType)}

Furthermore, in order to provide backwards-compatibility with AUTOSAR 4.0.3 bit-size of PID and DID signals, the AUTOSAR 4.0.3 parameters:

- {ecuc(Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDataSize)}
- {ecuc(Dcm/DcmConfigSet/DcmDsp/DcmDspPid/DcmDspPidData/DcmDspPidDataSize)}

shall be used.

Furthermore, as a result of the fact that AUTOSAR 4.3 compatibility is not implemented for PID signal configuration, the parameter `DcmDspPidDataType` does not exist and therefore the necessary array type for PIDs, DataArrayTypeUint8_{Data}Type, is generated without the 'Dcm_-' prefix, in conformance to AUTOSAR 4.2.1. Furthermore, the variation of this type is:

- When the related PID is usable for service $01 and
- When the interface chosen for the related PID is DcmDspPidDataUsePort = USE_DATA_SYNCH_-CLIENT_SERVER

Rationale:

In order to provide backwards-compatibility with AUTOSAR 4.0.3 bit-size of PID and DID signals, the AUTOSAR 4.0.3 parameters `DcmDspDataSize` and `DcmDspPidDataSize` are used instead of `DcmDspDataByteSize` and `DcmDspDataType` from AUTOSAR 4.3.0. The ECUC paths are changed since the ones in the AUTOSAR 4.0.3 specification:

- {ecuc(Dcm/DcmConfigSet/DcmDsp/DcmDspData.DcmDspDataSize)}
- {ecuc(Dcm/DcmConfigSet/DcmDsp/DcmDspPid/DcmDspPidData.DcmDspPidDataSize)}
- {ecuc(Dcm/DcmConfigSet/DcmDspData.DcmDspDataType)}
- {ecuc(Dcm/DcmConfigSet/DcmDsp/DcmDspPid/DcmDspPidData/DcmDspPidService01.DcmDspPidDataType)}

are non-existent.

Requirements:

SWS_Dcm_01121, SWS_Dcm_01122, SWS_Dcm_01123, SWS_Dcm_01124, SWS_Dcm_01125, SWS_Dcm_01126

- No support for DcmDspData USE_DATA_ASYNCH_FNC_ERROR and USE_DATA_ASYNCH_CLIENT_-SERVER_ERROR interface types

  Affected AUTOSAR releases:

  - R4.3.0

  Description:

  Support for USE_DATA_ASYNCH_FNC_ERROR and USE_DATA_ASYNCH_CLIENT_SERVER_ERROR for DcmDspData elements is not available.

  Rationale:

  These interface types have been added in a version of the AUTOSAR Dcm SWS which is later than 4.0.-3. Only select functionality from later versions of AUTOSAR is available.

  Requirements:

  SWS_Dcm_00439, SWS_Dcm_00436, SWS_Dcm_00437, SWS_Dcm_00395, SWS_Dcm_91005, SWS_Dcm_91008, SWS_Dcm_91009, SWS_Dcm_91010, SWS_Dcm_91011, SWS_Dcm_00686_-ASR43, SWS_Dcm_01035, SWS_Dcm_91012, SWS_Dcm_CONSTR_6060, SWS_Dcm_01017, SWS_-Dcm_00686

- The Dcm supports sender/receiver interfaces return error codes according to AUTOSAR 4.3.0 and and not according to AUTOSAR 4.0.3

  Affected AUTOSAR releases:

  - R4.0.3

  Description:

  The Dcm supports the following return error codes for sender/receiver interfaces as specified in AUTOSAR 4.3.0 in addition to the ones specified in AUTOSAR 4.0.3:

  - RTE_E_OUT_OF_RANGE
  - RTE_E_COM_BUSY
  - RTE_E_COM_STOPPED
  - RTE_E_HARD_TRANSFORMER_ERROR
  - RTE_E_SOFT_TRANSFORMER_ERROR

  Rationale:

The sender/receiver interfaces implementation is based on AUTOSAR 4.3.0 specifications.

Requirements:

Dcm040

- DCM does not support RoutineControl checks(security level/session/mode condition) per operation

Affected AUTOSAR releases:

- CP Release 4.3.0
- R4.2 Rev 1

Description:

AUTOSAR 4.2.1 and AUTOSAR 4.3.0 specify that the RoutineControl checks(security level/session/mode condition) are per operation(DcmDspStartRoutineCommonAuthorizationRef, DcmDspStopRoutineCommonAuthorizationRef and DcmDspRequestRoutineResultsCommonAuthorizationRef) and this is not supported by DCM. DCM supports the RoutineControl checks(security level/session/mode condition) per configured Routine(DcmDspRoutineSecurityLevelRef, DcmDspRoutineSessionRef, DcmDspRoutineModeRuleRef).

Requirements:

SWS_Dcm_00570, SWS_Dcm_00571, SWS_Dcm_00570_ASR42, SWS_Dcm_00571_ASR42, SWS_Dcm_00824_ASR42, SWS_Dcm_01169, SWS_Dcm_01170, SWS_Dcm_01171, SWS_Dcm_01141, SWS_Dcm_01141_ASR42

- DCM does not support the AUTOSAR 4.3.0 configuration naming for RoutineControl

Affected AUTOSAR releases:

- CP Release 4.3.0

Description:

The AUTOSAR 4.3.0 configuration naming for RoutineControl:

- DcmDspRequestRoutineResultsFnc
- DcmDspRequestRoutineResultsOut
- DcmDspRequestRoutineResultsOutSignal
- DcmDspStopRoutineFnc
- DcmDspStopRoutineInSignal
- DcmDspStopRoutineOut
- DcmDspStopRoutineOutSignal

- DcmDspRoutineParameterSize

are changed for backward compatibility. The AUTOSAR 4.0.3 configuration naming is used.

Requirements:

ECUC_Dcm_00753, ECUC_Dcm_00831, ECUC_Dcm_00836, ECUC_Dcm_00838, ECUC_Dcm_00752, ECUC_Dcm_00832 ECUC_Dcm_00833, ECUC_Dcm_00842, ECUC_Dcm_00664, ECUC_Dcm_00847, ECUC_Dcm_00850, ECUC_Dcm_00839, ECUC_Dcm_00841, ECUC_Dcm_00844

- DCM does not support all configuration containers for RoutineControl from AUTOSAR 4.3.0

Affected AUTOSAR releases:

- CP Release 4.3.0

Description:

The AUTOSAR 4.3.0 configuration containers for RoutineControl:

- DcmDspRoutineInfoByte
- DcmDspRequestRoutineResults
- DcmDspRequestRoutineResultsCommonAuthorizationRef
- DcmDspStopRoutine
- DcmDspStopRoutineCommonAuthorizationRef
- DcmDspStartRoutine
- DcmDspStartRoutineCommonAuthorizationRef

are not supported.

Requirements:

ECUC_Dcm_01063, ECUC_Dcm_01023, ECUC_Dcm_01054, ECUC_Dcm_01022, ECUC_Dcm_01053, ECUC_Dcm_01021, ECUC_Dcm_01052

- DCM does not support confirmation operation for RoutineControl from AUTOSAR 4.3.0

Affected AUTOSAR releases:

- CP Release 4.3.0

Description:

The AUTOSAR 4.3.0 confirmation configuration containers and operations for RoutineControl:

- DcmDspRequestRoutineResultsConfirmationEnabled
- DcmDspRequestRoutineResultsConfirmationFnc
- DcmDspStopRoutineConfirmationEnabled

- DcmDspStopRoutineConfirmationFnc
- DcmDspStartRoutineConfirmationEnabled
- DcmDspStartRoutineConfirmationFnc
- Xxx_StartConfirmation
- Xxx_StopConfirmation
- Xxx_RequestResultsConfirmation

are not supported.

Requirements:

ECUC_Dcm_01091, ECUC_Dcm_01090, ECUC_Dcm_01095, ECUC_Dcm_01096, ECUC_Dcm_01093, ECUC_Dcm_01094, SWS_Dcm_91016, SWS_Dcm_91017, SWS_Dcm_91018

- DCM does not follow the NRC handling for RoutineControl service

Affected AUTOSAR releases:

- CP Release 4.3.0
- R4.2 Rev 1

Description:

AUTOSAR 4.2.1 and AUTOSAR 4.3.0 specify that the NRCs order for RoutineControl service shall follow the ISO 14229-1/2013.(see Figure 25 — NRC handling for RoutineControl service from ISO 14229-1/2013) ISO 14229-1/2013 does not specify where the ModeRule check shall be made and also the "routineControlOptionRecord contains valid data for the requested RID" check shall be made by DCM. This check is not possible because just the application know if the routineControlOptionRecord contains valid data.

Requirements:

SWS_Dcm_01139, SWS_Dcm_01139_ASR42

- `VARIABLE_LENGTH` input signals for the UDS service `RoutineControl` operations are passed to the application using pointer to const.

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

In contrast to `Dcm690`, the UDS service `RoutineControl` operations use in their signature a pointer to const for the `VARIABLE_LENGTH` input signals.

Rationale:

This is done for compliance with the RTE where it is specified that all input parameters that are passed by reference or passed as an array expression shall be declared as pointer to const.

Requirements:

Dcm690

- The Dcm supports only `DcmDspDidControlMaskSize` values from 0 to 4.

Affected AUTOSAR releases:

- CP Release 4.4.0

Description:

AUTOSAR 4.4.0 specifies that `DcmDspDidControlMaskSize` values range is from 0 to 4294967294. Currently the Dcm supports only values range from 0 to 4 as specified by AUTOSAR 4.3.0.

Requirements:

SWS_Dcm_01285, SWS_Dcm_01286, SWS_Dcm_01290, SWS_Dcm_00802, ECUC_Dcm_01060

- The Dcm provides configurable interfaces declarations via `Dcm_API_Cfg.h`

Affected AUTOSAR releases:

- CP Release 4.4.0

Description:

AUTOSAR 4.4.0 specifies that configurable interfaces declarations shall be available via `Dcm_Externals.h`. Currently the Dcm provides configurable interfaces declarations via `Dcm_API_Cfg.h`.

Rationale:

In order to export configurable interfaces declarations a generated header file is necessary while `Dcm_Externals.h` is a static file, i.e. not generated.

Requirements:

SWS_Dcm_01285, SWS_Dcm_01286, SWS_Dcm_01290, SWS_Dcm_00802

- Interpretation of the value 0xFF of the parameter `DcmDslDiagRespMaxNumRespPend` by DCM

Affected AUTOSAR releases:

- CP Release 4.3.0

Description:

AUTOSAR 4.0.3 (Dcm693_Conf) specifies that the value 0xFF of the configuration parameter `DcmDslDiagRespMaxNumRespPend` shall mean that there is no limit on the number of NRC 0x78 (requestCorrectlyReceived-ResponsePending) to be tranmitted by the DCM. However, whether an infinite amount of requestCorrectlyReceived-ResponsePending (0x78) NRCs is sent depends on the configured value of the parameter `DcmDslDiagRespMaxNumRespPendInfinityLimit`:

- If the configuration parameter `DcmDslDiagRespMaxNumRespPendInfinityLimit` is set to "255", then the parameter `DcmDslDiagRespMaxNumRespPend` can have a value from 0 to 255 (inclusive) where 255 (0xFF) is treated as the infinity limit for tranmitting NRC 0x78 by the DCM.
- If the configuration parameter `DcmDslDiagRespMaxNumRespPendInfinityLimit` is set to "65535", then the parameter `DcmDslDiagRespMaxNumRespPend` can have a value from 0 to 65535 (inclusive) where 65535 (0xFFFF) is treated as the infinity limit for tranmitting NRC 0x78 by the DCM.

The maximum number of requestCorrectlyReceived-ResponsePending (0x78) NRCs to be sent can thus be increased from 254 to 65534.

Rationale:

To allow DCM to be configured to increase the number of NRC 0x78 that can be transmitted from 0 - 254 (inclusive) to 0 - 65534 (inclusive). The new value indicating no limit on number of NRC 0x78 to be tranmitted by DCM is 65535 instead of old value 255.

Requirements:

Dcm693_Conf

- For OBD service $04, Dcm transmits NRC 0x22 instead of positive response after reaching the configured limit for number of sent RCRRPs

Affected ISO releases:

- ISO 15031-5- 2011-04-15

Description:

For OBD service $04 (Clear/Reset emission-related diagnostic information), after the Dcm reaches the configured maximum number of NRCs with the negative response code 0x78 (requestCorrectlyReceivedResponsePending), it transmits a NRC response 0x22 (conditionsNotCorrect) instead of positive response. To avoid this scenario where the NRC 0x22 (conditionsNotCorrect) response is transmitted by the Dcm instead of positive response, the maximum number of RCRRPs (requestCorrectlyReceivedResponsePending) can be configured to infinity in Tresos.

Rationale:

The 0x10 NRC (generalReject) is not supported for the OBD service $04 - Clear/Reset emission-related diagnostic information. Table 10 - Proper response from server/ECU with ISO 9141-2, ISO 14230-4 and

SAE J1850 protocol Table 11 - Proper response from server/ECU for ISO 15765-4 protocol Table 16 - Negative response code (NRC) definition

Requirements:

Dcm120

- For OBD service $04, Dcm transmits NRC 0x22 instead of NRC 0x10 after reaching the configured limit for number of sent RCRRPs

Affected AUTOSAR releases:

- CP Release 4.0.3

Description:

AUTOSAR 4.0.3 specifies that after reaching the value defined in the configuration parameter `DcmDslDiagRespMaxNumRespPend` the DCM module should stop processing of the active diagnostic request and shall send a negative response with NRC 0x10(GeneralReject). Instead of sending NRC 0x10(GeneralReject) for service 0x04 - Clear/Reset emission-related diagnostic information the Dcm will send a negative response with NRC 0x22(ConditionsNotCorrect).

Rationale:

The 0x10 NRC (generalReject) is not supported for the OBD service $04 - Clear/Reset emission-related diagnostic information according to ISO 15031-5- 2011-04-15.

Requirements:

Dcm120

- `AttemptCounter` is not incremented according to ISO14229-1:2013 requirements

Affected AUTOSAR releases:

- ISO14229-1:2013

Description:

Security `AttemptCounter` is not incremented according to ISO14229-1:2013 requirements in the following circumstances:

- A SecurityAccess (0x27) service request for the CompareKey (2*SecurityLevel) subfunction for a Security Level is received which does not occur after a successful SecurityAccess (0x27) service request for the Request (2*SecurityLevel-1) subfunction for the same Security Level.
- A SecurityAccess (0x27) service request for the CompareKey (2*SecurityLevel) subfunction for a Security Level is received which occurs after a successful SecurityAccess (0x27) service request for the Request (2*SecurityLevel-1) subfunction for the same Security Level but the request for the CompareKey (2*SecurityLevel) subfunction has an incorrect length.

Rationale:

These two scenarios are not handled by the AUTOSAR.

Requirements:

Table I.2 - State transitions - disjunctive normal form representation

- DCM does not perform check for permission to clear DTC for the UDS service `ClearDiagnosticInformation (0x14)`

Affected AUTOSAR releases:

- CP Release 4.4.0

Description:

Dcm does not check whether application allows to clear DTC.

Requirements:

SWS_Dcm_01268

- DCM does not perform check for mode condition for the UDS service `ClearDiagnosticInformation (0x14)`

Affected AUTOSAR releases:

- CP Release 4.4.0

Description:

Dcm does not check if DTC can be cleared in current mode condition.

Requirements:

SWS_Dcm_01269

- Subfunction `reportDTCBySeverityMaskRecord (0x08)` of service `ReadDTCInformation (0x19)`

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

`reportDTCBySeverityMaskRecord` (0x08) subfunction of UDS service `ReadDTCInformation (0x19)` is implemented according to AUTOSAR CP Release 4.4.0.

Requirements:

Dcm379, Dcm380

- Subfunction `reportSeverityInformationOfDTC (0x09)` of service `ReadDTCInformation (0x19)`

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

`reportSeverityInformationOfDTC (0x09)` subfunction of UDS service `ReadDTCInformation (0x19)` is implemented according to AUTOSAR CP Release 4.4.0.

Requirements:

Dcm381

- `ComM_DCM_ActiveDiagnostic()` is called in a non-default session as well

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

When in a non-default session, `ComM_DCM_ActiveDiagnostic()` is called for ComMChannels for which ActiveDiagnostic has not been requested, if a new request occurs on such a channel. This is the situation in which two different RxPduIDs reference two different ComM channels, but belong to the same DcmDslProtocol. In such a situation, if these RxPduIDs receive requests one after another, both these requests must be allowed to transmit their responses.

Requirements:

Dcm169

- `ComM_DCM_InactiveDiagnostic()` is invoked when `Dcm_TpTxConfirmation()` is called in the context of failed `RequestCorrectlyReceivedResponsePending (0x78)` NRC transmissions as well.

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

`RequestCorrectlyReceivedResponsePending (0x78)` NRC transmissions should not result in the Dcm going to Inactive Diagnostic on a ComMChannel, but in the case in which the transmission has failed, this leeds to the request being terminated. This is equivalent to a final response.

Requirements:

Dcm165

- If Dem_EnableDTCSetting() is called according to [SWS_Dcm_00751] and [SWS_Dcm_00752] and returns DEM_PENDING, the Dcm does not call this function again.

Affected AUTOSAR releases:

- CP Release 4.4.0

Description:

In case DTCSetting is disabled and shall be re-enabled (see [SWS_Dcm_00751] and [SWS_Dcm_00752]), the Dcm does not call Dem_EnableDTCSetting() again if the first call has returned DEM_PENDING. If Dem provides an asynchronous implementation of Dem_EnableDTCSetting(), i.e. E_OK is not returned with the first call, Dcm expects that Dem processes the EnableDTCSetting request to the end without any subsequent calls until E_OK is returned. During this request processing by Dem, any other EnableDTCSetting or DisableDTCSetting request with same or different ClientId shall be possible and managed by Dem.

Requirements:

SWS_Dcm_01412

- Subfunction `reportNumberOfDTCBySeverityMaskRecord (0x07)` of service `ReadDTCInformation (0x19)`

Affected AUTOSAR releases:

- R4.0 Rev 3

Description:

`reportNumberOfDTCBySeverityMaskRecord (0x07)` subfunction of UDS service `ReadDTCInformation (0x19)` is implemented according to AUTOSAR CP Release 4.4.0.

Requirements:

Dcm293

- When receiving a request for reportUserDefMemoryDTCByStatusMask (0x17), reportUserDefMemoryDTCSnapshotRecordByDTCNumber (0x18) or reportUserDefMemoryDTCExtDataRecordByDTCNumber(0x19), the Dcm does not check if the access to the selected user defined memory in parameter MemorySelection is authenticated.

Affected AUTOSAR releases:

- CP Release 4.4.0

Description:

Dcm does not support service 0x29 - Authentication.

Requirements:

SWS_Dcm_01545, SWS_Dcm_01546, SWS_Dcm_01547

- When receiving a request for UDS service ReadDTCInformation (0x19) with subservice reportDTCSnapshotIdentification (0x03), the Dcm does not call the API Dem_GetNumberOfFreezeFrameRecords().

Affected AUTOSAR releases:

- CP Release 4.4.0

Description:

In case UDS Service ReadDTCInformation is requested with subservice reportDTCSnapshotIdentification (0x03) the Dcm does not call Dem_GetNumberOfFreezeFrameRecords(). The order of API calls is Dem_SetFreezeFrameRecordFilter() followed by Dem_GetNextFilteredRecord() according to AUTOSAR 4.3.1.

Requirements:

SWS_Dcm_00298

- The Dcm-Dem interface implementation is based on AUTOSAR 4.3.1 specification with corrections from AUTOSAR 4.4.0 specification.

Affected AUTOSAR releases:

- R4.0 Rev 3
- R4.2.1
- R4.2.2

Description:

The Dcm implements Dem API calls based on AUTOSAR 4.3.1 for the following services: - UDS service 0x14 - ClearDiagnosticInformation - UDS service 0x19 - ReadDTCInformation - UDS service 0x85 - ControlDTCSetting - UDS service 0xAF - ReadGenericInformation - OBD service $04 - Clear/Reset emission-related diagnostic information - OBD service $03 - Request emission-related diagnostic trouble codes - OBD service $07 - Request emission-related diagnostic trouble codes detected during current or last completed driving cycle - OBD service $0A - Request emission-related diagnostic trouble codes with permanent status - OBD service $02 - Request powertrain freeze frame data with corrections from AUTOSAR

4.4.0, such as Dem_GetDTCSelectionResultForClearDTC() API shall be used by service 0x14 instead of Dem_GetDTCSelectionResult() API. This means that, with respect to AUTOSAR 4.0.3 and to extension from AUTOSAR 4.2, the signatures of several Dem APIs are changed, some Dem APIs are no longer supported and some new Dem APIs are introduced. These interfaces are not anymore backward compatible. AUTOSAR 4.2 extensions for example are: - new subservices 0x17, 0x18, 0x18 for service 0x19 supporting user defined memory - new error codes DEM_CLEAR_BUSY and DEM_CLEAR_MEMORY_ERROR for Dem_ClearDTC() API of services 0x14 and $04 - optional parameter DTCSettingControlOptionRecord for service 0x85

Rationale:

The Dem API calls as specified in AUTOSAR 4.3.1 with corrections from AUTOSAR 4.4.0 are necessary to support parallel OBD and UDS protocol processing, i.e. to allow parallel access to the event related data and fault memory by multiple diagnostic requests at the same time.

Requirements:

Dcm371, Dcm372, Dcm702, Dcm005, Dcm705, Dcm706, Dcm708, Dcm739, Dcm740, Dcm293, Dcm377, Dcm008, Dcm378, Dcm297, Dcm295, Dcm296, Dcm478, Dcm474, Dcm475, Dcm476, Dcm386, Dcm382, Dcm298, Dcm383, Dcm384, Dcm385, Dcm441, Dcm393, Dcm466, Dcm766, Dcm465, Dcm304, Dcm783, Dcm406, Dcm784, Dcm751, Dcm752, Dcm077, Dcm289, Dcm412, Dcm004, Dcm413, Dcm703, SWS_Dcm_00008, SWS_Dcm_00378, SWS_Dcm_00297, SWS_Dcm_00295, SWS_Dcm_00296, SWS_Dcm_00474, SWS_Dcm_00475, SWS_Dcm_00476, SWS_Dcm_00386, SWS_Dcm_00302, SWS_Dcm_01147, SWS_Dcm_01148, SWS_Dcm_01149, SWS_Dcm_00387, SWS_Dcm_00304, SWS_Dcm_01063, SWS_Dcm_01064, SWS_Dcm_00783, SWS_Dcm_00406, SWS_Dcm_00784, SWS_Dcm_00751, SWS_Dcm_00752, SWS_Dcm_00830, SWS_Dcm_00841, SWS_Dcm_00393, SWS_Dcm_00466, SWS_Dcm_00766, SWS_Dcm_00289, SWS_Dcm_00412, SWS_Dcm_00330, SWS_Dcm_00004, SWS_Dcm_00413, SWS_Dcm_00703, SWS_Dcm_01250, SWS_Dcm_01251

- Subfunction `reportWWHOBDDTCWithPermanentStatus (0x55)` of service `ReadDTCInformation (0x19)`

Affected AUTOSAR releases:

- CP Release 4.4.0

Description:

`reportWWHOBDDTCWithPermanentStatus (0x55)` subfunction of UDS service `ReadDTCInformation(0x19)` is not supported.

Requirements:

SWS_Dcm_01343, SWS_Dcm_01344, SWS_Dcm_01345, SWS_Dcm_01346

- For OBD service Request Vehicle Information 0x09, Dcm transmits NRC 0x22 instead of 0x12

Affected AUTOSAR releases:

- CP Release 4.4.0

Description:

For OBD service $09 (Request Vehicle Information), if Xxx_GetInfoTypeValueData operation returns values different than E_OK or E_PENDING, then Dcm shall return NRC 0x22 instead of 0x12.

Rationale:

According to "Table 11 — Proper response from server/ECU for ISO 15765-4 protocol" from "ISO 15031-5:2011", the NRC 0x12 is not supported by OBD Service Request Vehicle Information 0x09. (Refer to AUTOSAR Jira https://jira.autosar.org/browse/AR-96256)

Requirements:

SWS_Dcm_01191

- Subfunction checks for Session/Security/ModeRule for UDS Service Routine Control (0x31).

Affected AUTOSAR releases:

- R4.0.3

Description:

For UDS service Routine Control (0x31), subfunction checks for Session/Security/ModeRule are not performed.

Rationale:

According to "Figure 5 — General server response behaviour" from "ISO 14229-1/2013", the checks for Session/Security/ModeRule are explicitly skipped.

- Subfunction `reportMirrorMemoryDTCExtDataRecordByDTCNumber (0x10)` of service `ReadDTCInformation (0x19)`

Affected AUTOSAR releases:

- CP Release 4.4.0

Description:

`reportMirrorMemoryDTCExtDataRecordByDTCNumber (0x10)` subfunction of UDS service `ReadDTCInformation (0x19)` is not supported.

Requirements:

SWS_Dcm_00371

- UDS Service EcuReset (0x11) for subfunctions enableRapidPowerShutDown (0x04) and disableRapidPowerShutDown (0x05).

Affected AUTOSAR releases:

- R4.2.1
- R4.2.2
- R4.3.0
- R4.3.1
- R4.4.0

Description:

For UDS Service EcuReset (0x11) with subfunctions enableRapidPowerShutDown (0x04) and disableRapidPowerShutDown (0x05) further requests to the Dcm shall not be ignored.

Rationale:

Ignoring further requests after UDS Services EcuReset (0x11) with subfunctions enableRapidPowerShutDown (0x04) and disableRapidPowerShutDown (0x05) would be too restrictive as it allows no further requests after these. For example, if you enable rapid power shutdown with 0x04, you won't be able to disable it again as the 0x05 request would be rejected.

Requirements:

SWS_Dcm_00834

- Multiplicity of `DcmDemClientRef`

Affected AUTOSAR releases:

- CP Release 4.4.0

Description:

Multiplicity of `DcmDemClientRef` is changed from 1 to 0..1.

Rationale:

The `Dcm` should be able to generate and compile with no `Dem` module/stub present regardless of the availability of `UDS Services 0x19, 0x14, 0x85` or `OBD Services`.

Requirements:

ECUC_Dcm_01083

- For OBD service $09, Dcm transmits a negative response with NRC ConditionsNotCorrect (0x22) after reaching the configured limit for number of sent RCRRPs

  Affected ISO releases:

  - ISO 15031-5- 2011-04-15

  Description:

  For OBD service $09 (Request vehicle information), after the Dcm reaches the configured maximum number of NRCs with the negative response code 0x78 (requestCorrectlyReceivedResponsePending), it transmits a negative response with NRC ConditionsNotCorrect (0x22).

  Rationale:

  The NRC 0x10 (generalReject) is not supported for the OBD service $09 - Request vehicle information. Please refer the following: Table 10 - Proper response from server/ECU with ISO 9141-2, ISO 14230-4 and SAE J1850 protocol Table 11 - Proper response from server/ECU for ISO 15765-4 protocol Table 16 - Negative response code (NRC) definition

  Requirements:

  Dcm120

- For OBD service $09, Dcm transmits a negative response with NRC ConditionsNotCorrect (0x22) after reaching the configured limit for number of sent RCRRPs.

  Affected AUTOSAR releases:

  - CP Release 4.0.3

  Description:

  AUTOSAR 4.0.3 specifies that after reaching the value defined in the configuration parameter `DcmDslDiagRespMaxNumRespPend` the DCM module should stop processing of the active diagnostic request and shall send a negative response with NRC 0x10(GeneralReject). Instead of sending NRC 0x10(GeneralReject) for service 0x09 - Request vehicle information the Dcm shall sent a negative response with NRC ConditionsNotCorrect (0x22).

  Rationale:

  The 0x10 NRC (generalReject) is not supported for the OBD service $09 - Request vehicle information according to ISO 15031-5- 2011-04-15.

  Requirements:

  Dcm120

- For OBD service Request Vehicle Information 0x09, Dcm transmits NRC 0x22 instead of 0x10

Affected AUTOSAR releases:

- CP Release 4.4.0

Description:

When OBD service $09 Request vehicle information is requested and UDS service (0x22) ReadDataByIdentifier with DID inside the OBD InfoType range (F800-F8FF) is already running, the Dcm might transmit a RCRRP (0x78) before no final response or negative response with NRC CNC (0x22)

Rationale:

According to "Table 11 - Proper response from server/ECU for ISO 15765-4 protocol" from "ISO 15031-5:2011", the NRC 0x10 is not supported by OBD Service Request Vehicle Information 0x09.

Requirements:

Dcm120

- DcmDspCommonAuthorizationRef configuration parameter is not supported.

Affected AUTOSAR releases:

- CP Release 4.2.1

Description:

The Dcm module does not support the `DcmDspCommonAuthorizationRef` configuration parameter.

Rationale:

According to "Constr_6032" from CP Release 4.2.1, `DcmDspCommonAuthorizationRef` should be disabled if DcmDspRoutineTidRef is used.

Requirements:

constr_6032

- For UDS service 0x24 (ReadScalingDataByIdentifier), the Dcm does not support NRC 0x34 (authenticationRequired)

Affected ISO 14229-1 releases:

- ISO14229-1:2020

Description:

According to "Table 210 - Supported negative response codes" in ISO14229-1:2020, on receiving the UDS service 0x24 (ReadScalingDataByIdentifier) request, the Dcm shall send a response with NRC 0x34

(authenticationRequired) if the Data Identifier (DID) is secured and the client has insufficient rights based on the Authentication state for the requested DIDs.

However, the Dcm does not support role-based permission checks at the DID level.

Rationale:

Support for role-based permission checks at the DID level is not implemented.

Requirements:

Dcm.EB.ReadScalingDataByIdentifier.AuthenticationRequired.NRC0x34

- Function interface not specified

Affected AUTOSAR releases:

- ASR R20-11

Description:

The requirements SWS_Dcm_00798 and SWS_Dcm_91012 specify the synchronous and asynchronous interfaces for function as well as client-server operations respectively. However, since the requirement SWS_Dcm_00686 also specify the interfaces for synchronous and asynchronous client-server operations, the scope of the requirements SWS_Dcm_00798 and SWS_Dcm_91012 is limited only to the interface definitions for synchronous and asynchronous function operations.

Rationale:

This deviation is required to avoid duplicate requirements regarding the definition of synchronous and asynchronous interfaces for client-server operations.

Requirements:

SWS_Dcm_00798, SWS_Dcm_91012

- Signature of expected interfaces for the operation of service `RoutineControl (0x31)`

Affected AUTOSAR releases:

- R4.2 Rev 1

Description:

The ordering of a routine signal used in RoutineControl service calculated from DcmDspRoutineSignalPos bits positions in the list represents the order of the respective dataInN or dataOutN elements in the XXX_-Start/ XXX_Stop / XXX_RequestResult function call.

Requirements:

ECUC_Dcm_00836, ECUC_Dcm_00839, ECUC_Dcm_00842, ECUC_Dcm_00845, ECUC_Dcm_00848

- Whitelist based permission checks for service execution

  Affected AUTOSAR releases:

  - R20-11

  Description:

  Dcm does not support whitelist based permission checks for service execution. Consequently, the following configuration parameters are not supported:

  - DcmDspAuthenticationWhiteListDIDMaxSize
  - DcmDspAuthenticationWhiteListRIDMaxSize
  - DcmDspAuthenticationWhiteListMemorySelectionMaxSize
  - DcmDspAuthenticationWhiteListMemorySelectionElementRef
  - DcmDspAuthenticationWhiteListRIDElementRef
  - DcmDspAuthenticationWhiteListDIDElementRef
  - DcmDspAuthenticationWhiteListServicesElementRef
  - DcmDspAuthenticationWhiteListServicesMaxSize

  Requirements:

  SWS_Dcm_01539, SWS_Dcm_01541, SWS_Dcm_01542, SWS_Dcm_01562, SWS_Dcm_01545, SWS_Dcm_01546, SWS_Dcm_01547, SWS_Dcm_01548, SWS_Dcm_01549, SWS_Dcm_01524, SWS_Dcm_01525, SWS_Dcm_01526, SWS_Dcm_01527, SWS_Dcm_01552, SWS_Dcm_01553, SWS_Dcm_01496, SWS_Dcm_01554, SWS_Dcm_01555, SWS_Dcm_01556, SWS_Dcm_01557, SWS_Dcm_01528, SWS_Dcm_01532, ECUC_Dcm_01155, ECUC_Dcm_01157, ECUC_Dcm_01156, ECUC_Dcm_01154, ECUC_Dcm_01171, ECUC_Dcm_01170, ECUC_Dcm_01169, SWS_Dcm_-CONSTR_6087, ECUC_Dcm_01168, SWS_Dcm_01516, SWS_Dcm_01517, SWS_Dcm_01518, SWS_-Dcm_01531

- Role based permission checks for service execution

  Affected AUTOSAR releases:

  - R20-11

  Description:

  The following role-based permission checks are not supported:

  - Check on statically defined DIDs
  - Check on dynamically defined DIDs

- Checks on service 0x31 per sub-function
- Checks on service 0x19 parameter MemorySelection

Consequently, the following configuration parameters are not supported:

- DcmDspReadDTCInformationUserDefinedFaultMemoryRole
- DcmDspDidReadRole
- DcmDspDidWriteRole
- DcmDspDidControlRole
- DcmDspRequestRoutineResultsRole
- DcmDspStartRoutineRole
- DcmDspStopRoutineRole

Requirements:

SWS_Dcm_01540, SWS_Dcm_01545, SWS_Dcm_01546, SWS_Dcm_01547, SWS_Dcm_01548, SWS_Dcm_01549, SWS_Dcm_01552, SWS_Dcm_01553, SWS_Dcm_01496, SWS_Dcm_01554, SWS_Dcm_01555, SWS_Dcm_01556, SWS_Dcm_01557, ECUC_Dcm_01150, ECUC_Dcm_01141, ECUC_Dcm_01142, ECUC_Dcm_01143, ECUC_Dcm_01146, ECUC_Dcm_01144, ECUC_Dcm_01145

- Persistent storage for `Authentication (0x29)` service

Affected AUTOSAR releases:

- R20-11

Description:

For `Authentication ($29)` service, the persistent storage of authentication state, role and whitelist is not supported. Consequently, the configuration parameter `DcmDspAuthenticationPersistState-ModeRuleRef` is not supported.

Requirements:

ECUC_Dcm_01160, SWS_Dcm_01530, SWS_Dcm_01480, SWS_Dcm_01481, SWS_Dcm_01484

- Certificate target identification block for `Authentication (0x29)` service

Affected AUTOSAR releases:

- R20-11

Description:

Dcm does not evaluate certificate target identification blocks for certificate acceptance. Consequently, the configuration parameters `DcmDspAuthenticationTargetIdentificationModeRuleRef` and `DcmSwcSRDataElementRef` are not supported.

Requirements:

ECUC_Dcm_01172, SWS_Dcm_01491, SWS_Dcm_01490, SWS_Dcm_01492, SWS_Dcm_CONSTR_-6089

- Activation of role and white list for `Authentication (0x29)` service

Affected AUTOSAR releases:

- R20-11

Description:

Currently the Dcm considers role for access control in `DCM_DEAUTHENTICATED` and `DCM_AUTHENTICATED` state. Consequently, while in `DCM_DEAUTHENTICATED` state, the Dcm uses the role configured in `DcmDspAuthenticationDeauthenticatedRole` or the role provided by `Dcm_SetDeauthenticatedRole()` API.

Requirements:

SWS_Dcm_01531

- Access rights for services in deauthenticated state for `Authentication (0x29)` service

Affected AUTOSAR releases:

- R20-11

Description:

Currently, while in `DCM_DEAUTHENTICATED` state, the Dcm shall use the role configured in `DcmDspAuthenticationDeauthenticatedRole` as current role for all role based access verification checks. Consequently, after a successful call of `Dcm_SetDeauthenticatedRole()` API, the Dcm uses the role provided by `Dcm_SetDeauthenticatedRole()` API as the current role for all role based access verification checks.

Requirements:

SWS_Dcm_01538

- API `Dcm_CsmAsyncJobFinished()`

Affected AUTOSAR releases:

- R20-11

Description:

Prototype changed to `Dcm_CsmAsyncJobFinished (const Crypto_JobType* Job, Std_ReturnType Result)` from `Dcm_CsmAsyncJobFinished (Std_ReturnType Result)`.

Rationale:

To ensure the compatibility with Csm module.

Requirements:

SWS_Dcm_91076

- API `Dem_SelectDTC()`

Affected AUTOSAR releases:

- AUTOSAR 4.4.0

Description:

Parameter `DTCOrigin` of API `Dem_SelectDTC` supports different values than DEM_DTC_ORIGIN_PRIMARY_MEMORY.

Rationale:

The requirement `SWS_Dcm_01263` only applies when configuration parameter `DcmDspClearDTCMemorySelection` is set to FALSE, if configuration parameter `DcmDspClearDTCMemorySelection` is set to TRUE, the `DTCOrigin` parameter of API `Dem_SelectDTC` supports different values than DEM_DTC_ORIGIN_PRIMARY_MEMORY.

Requirements:

SWS_Dcm_01263

- Configuration parameter `DcmDspAuthenticationECUCertificateKeyElementRef` of `Authentication (0x29)` service

Affected AUTOSAR releases:

- R20-11

Description:

For Subfunction `verifyCertificateBidirectional (0x02)` of `Authentication (0x29)` service, the configuration parameter `DcmDspAuthenticationECUCertificateKeyElementRef` is not supported.

Rationale:

The AUTOSAR specification does not specify any use-case that requires the usage of the configuration parameter `DcmDspAuthenticationECUCertificateKeyElementRef`. Therefore, this configuration parameter is not supported. For more details, please refer to "Proposed Solution" in AUTOSAR Jira ticket https://jira.autosar.org/browse/AR-99731.

Requirements:

ECUC_Dcm_01178, SWS_Dcm_01462

- Only one TID value may be present in an UDS Service RoutineControl (0x31) request

Affected AUTOSAR releases:

- R4.0 Rev 3
- R4.2.1

Description:

AUTOSAR specifies that on reception of the `UDS Service RoutineControl (0x31)`, for every requested TID inside the OBD range (E000-E0FF), the DCM module shall get the TID value as defined for `OBD Service $08`. This means that a request can contain more then one TID, because `OBD Service $08` can request more then one TID.

Rationale:

According with ISO 14229-1 the `UDS Service RoutineControl (0x31)` can request just one routine ID.

Requirements:

Dcm701, SWS_Dcm_00701

- The OBD and UDS DTCs separation is not supported

Affected AUTOSAR releases:

- R21-11

Description:

The use of separated OBD and UDS DTCs in context of the UDS `Read DTC Information (0x19)` request with subfunction `reportWWHOBDDTCByMaskRecord (0x42)` is not supported, i.e. DTC format DEM_DTC_FORMAT_OBD_3BYTE is not supported. Consequently, the following configuration parameters are not supported:

- DcmDspReadDTCInformation
- DcmDspReadDTCInformationSupportedObdUdsDtcSeparation

Requirements:

ECUC_Dcm_01147, ECUC_Dcm_01214, SWS_Dcm_01618, SWS_Dcm_01619

- API `Dem_GetDTCStatusAvailabilityMask()` AUTOSAR CP R21-11 is not supported.

  Affected AUTOSAR releases:

  - CP Release R21 11

  Description:

  Dem API Dem_GetDTCStatusAvailabilityMask() is based on AUTOSAR CP 4.3.1.

  Rationale:

  Dcm calls the API Dem_GetDTCStatusAvailabilityMask() without the parameter DTCOrigin.

- No support for link and post build time configuration

  Affected AUTOSAR releases:

  - R21-11
  - AUTOSAR 4.0.3

  Description:

  The Dcm module can only be configured at pre-compile time. Link and post build time configuration is 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:

  Dcm805_Conf, ECUC_Dcm_00805

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.

- **Integration Note:** In addition to integration requirements documented in the current release notes, refer to the Dcm module references and to the EB tresos AutoCore Generic 8 documentation, Diagnostics/Error handler user's guide for further important information regarding the integration of the Dcm module.

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

  Description:

  The Dcm module can only be configured at pre-compile time. Link and post build time configuration is 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:

  Dcm054, Dcm172, Dcm173, VCC-DCM-007:Req22v1, VCC-DCM-007:Req23v1

- Handling of ServiceToRespond record in service `ResponseOnEvent (0x86)` (reference to product description: ASCPD-75)

  Description:

  The current implementation supports only the services 0x22 and 0x19 with subfunction 0x02 in `ServiceToRespond` record of the `ResponseOnEvent` service request.

  Rationale:

  Feature implementation has been limited to services `ReadDataByIdentifier (0x22)` and subfunction `reportDTCByStatusMask (0x02)` of service `ReadDTCInformation (0x19)` in current release.

  Requirements:

  Section 7.2.4.7

- Support for the container `DcmDsdSubService`

  Description:

  Subfunction level configuration support is provided for all UDS services except for `RoutineControl (0x31)`.

  Rationale:

  Service `RoutineControl (0x31)` configuration is done via configuration parameters in the dedicated container `DcmDspRoutine`.

  Requirements:

  Dcm802_Conf

- Limited value of `DcmDspSessionLevel` parameter

Description:

According to Dcm SWS of AUTOSAR R4.0, the value of the parameter `DcmDspSessionLevel` ranges from 0 to 255. For the current implementation, the range is limited from 1 to 255.

Rationale:

The value of the default session level is 1. So it is limited from 1 to 255.

Requirements:

Dcm765_Conf

- Limited the value of parameter `DcmModeRuleNrcValue`

Description:

According to Dcm SWS of AUTOSAR R4.0, the value of the parameter `DcmModeRuleNrcValue` ranges from 0 to 255. But for the current implementation, the range is limited from 1 to 255.

Rationale:

Requires clarification from Bugzilla http://www.autosar.org/bugzilla/show_bug.cgi?id=59277

Requirements:

Dcm949_Conf

- Limited the value of parameter `DcmTaskTime`

Description:

According to Dcm SWS of AUTOSAR R4.0 Rev 3, the value of the parameter `DcmTaskTime` ranges from 0 to 100 seconds. But for the current implementation, the range is limited from 0.001 to 1 second.

Rationale:

A minimum value of 1 ms and a maximum value of 1 s should be sufficient for most use cases. Refer to AUTOSAR Bugzilla http://www.autosar.org/bugzilla/show_bug.cgi?id=56254.

Dcm820_Conf

- Limited multiplicity of `DcmDslProtocolRow`

Description:

According to Dcm SWS of AUTOSAR R4.0 Rev 3, the container `DcmDslProtocolRow` has an infinite upper multiplicity from 1 to *. But for the current implementation, the range is limited from 1 to 256.

Rationale:

To optimize the configuration, multiplicity is changed.

Requirements:

Sections 10.2.11, 10.2.12, Dcm694_Conf, Dcm695_Conf

- Limited timing unit reference of `DcmDspSecurityDelayTime` parameter value

Description:

Dcm SWS of AUTOSAR R4.0 Rev 3 contains contradicting statements regarding the timing unit reference of the parameter `DcmDspSecurityDelayTime`. It specifies both milliseconds and seconds which affect the specified range from 0 to 20.000. If milliseconds are applicable this would mean the higher range value is 20 s. This value is less than what might be required by some applications. Thus the range is limited and corrected from 0 ms to 20.000 ms to a range of 0s to 65.335 s. This is also in accordance to Dcm SWS of AUTOSAR R4.1 Rev 1.

Rationale:

Dcm757_Conf High range value of 20.0000 ms is too less for some use cases. Refer to AUTOSAR Bugzilla http://www.autosar.org/bugzilla/show_bug.cgi?id=52073.

- Limited the value of parameters `DcmDspSecurityAdrSize`, `DcmDspSecuritySeedSize`, and `DcmDspSecurityKeySize`

Description:

According to Dcm SWS of AUTOSAR R4.0, the value of the parameters `DcmDspSecurityAdrSize`, `DcmDspSecuritySeedSize`, and `DcmDspSecurityKeySize` ranges from 1 to 4294967295. For the current implementation, the range is limited from 1 to 4294967293.

Rationale:

A security access service request or response consists of `securityAccessDataRecord`, `securitySeed`, or `securityKey` in addition to the service ID and subfunction ID. The service ID and subfunction ID occupy one byte each. This means that the payload has a maximum size equal to the maximum size of the containing buffer minus 2.

Requirements:

Dcm765_Conf Dcm725_Conf Dcm755_Conf

- Limited the range of `DcmDspRoutineSignalLength`

Description:

According to DCM SWS of AUTOSAR 4.0.3, the value of the parameter `DcmDspRoutineSignalLength` ranges from 0 to 65535. But for the current implementation, the range is limited from 1 to 65535.

Rationale:

A signal length of 0 is not useful.

Requirements:

Dcm838_Conf, Dcm841_Conf, Dcm844_Conf, Dcm847_Conf, Dcm850_Conf

- Limited multiplicity of `DcmDspPidData`

Description:

According to Dcm SWS of AUTOSAR R4.0 Rev 3, the container `DcmDspPidData` has an infinite upper multiplicity. But for the current implementation, the range is limited from 1 to 255.

Rationale:

The interface to read data from Dem (Dem_ReadDataOfOBDFreezeFrame) requires an uint8 parameter as index of PidData.

Requirements:

ECUC_Dcm_00865

- Limited type of source data identifier of a Dynamically Defined Data Identifier

Description:

According to ISO 14229-1 it is not prohibited by the UDS service DynamicallyDefineDataIdentifier (0x2C) but it is not recommended for the client to reference one dynamically defined data record from another. The deletion of the referenced record could create data consistency problems within the referencing record. EB implementation returns NRC 0x31 in this case.

Rationale:

Deletion of the referenced record could create data consistency problems.

Requirements:

Dcm.EB.DynamicallyDefineDataIdentifier.ISO

- Limited range of `DcmDslProtocolPreemptTimeout`

Description:

According to Dcm SWS of AUTOSAR R4.2.2, the parameter `DcmDslProtocolPreemptTimeout` has a lower limit of 0 and an upper limit of 1. But for the current implementation, the lower limit must be above 0 and below the value configured for `P2ServerMax` of the default session of the new protocol.

Rationale:

The handling of NRC 0x78 (RequestCorrectlyReceivedResponsePending) is bound to a protocol and a session. It is not defined how the reception layer shall handle this before starting a protocol.

Requirements:

ECUC_Dcm_00698

- Removal order of duplicated periodic DIDs in a UDS service request ReadDataByPeriodicIdentifier (0x2A)

Description:

When filtering out duplicated periodic DIDs in a ReadDataByPeriodicIdentifier request the first of a duplicated periodic DID is kept and the other duplicates are removed.

Rationale:

Reuse of an already present, tested, and working filtering function. Unlikelihood of the "keep first duplicate" approach causing problems.

- UDS services DiagnosticSessionControl (0x10), CommunicationControl (0x28), ReadDataByPeriodicIdentifier (0x2A), and ResponseOnEvent (0x86) cannot be configured for asynchronous operation

Description:

Services DiagnosticSessionControl (0x10), CommunicationControl (0x28), ReadDataByPeriodicIdentifier (0x2A), and ResponseOnEvent (0x86) must always be set up to operate synchronously, i.e. the configuration flag `DcmAsyncServiceExecution` must be set to 'false'. Setting that flag to 'true' for these services results in an error when generating the project.

Rationale:

Not allowing the services DiagnosticSessionControl (0x10), CommunicationControl (0x28), ReadDataByPeriodicIdentifier (0x2A), and ResponseOnEvent (0x86) service handler to run in the background helps avoid race conditions that would have to be taken care of. Managing those possible race conditions would degrade the performance of the service and add avoidable complexity.

- When using DynamicallyDefinedDIDs, the Dcm's `Dcm_Init()` API must be called after `NvM_ReadAll()` finishes

Description:

When using DynamicallyDefinedDIDs, the Dcm's `Dcm_Init()` API must be called after the `NvM_ReadAll()` operation returns a final status, i.e. a status which is not `NVM_REQ_PENDING`.

Rationale:

The DynamicallyDefinedDID persistency is implemented by having an NvM block maintaining the mirror of the DynamicallyDefinedDID configuration data, which is retrieved by calling the NvM API `NvM_ReadAll`.

The `NvM_ReadAll()` request is handled asynchronously which means that the operation might still be ongoing when `Dcm_Init()` is called and this leads to possibly undefined behavior.

- If the configuration parameter `DcmDataTransferServicesASRVersion` is set to the value DATA_TRANSFER_SERVICES_AUTOSAR_403, the optional parameters of Dcm_ProcessRequestTransferExit callout according to AUTOSAR 4.0.3 are not supported

  Description:

  When requesting RequestTransferExit (0x37), the following optional parameters of Dcm_ProcessRequestTransferExit callout are not supported:

  - `uint8* ParameterRecord`
  - `uint32 ParameterRecordSize`

  Rationale:

  According to AUTOSAR 4.0.3 the parameters are optional and they refer to manufacturer specific data. They are not supported in the current implementation.

- Internal ROE-triggered request shall result in a ConditionsNotCorrect (0x22) NRC

  Description

  An internal ROE-triggered request for the UDS service ReadDataByIdentifier (0x22), occurring at the same time as an external request shall result in a ConditionsNotCorrect (0x22) NRC when it is being serviced for either of the following services:

UDS service	`ReadDataByIdentifier (0x22)`
	`InputOutputControlByIdentifier (0x2F)`
	`WriteDataByIdentifier (0x2E)`
OBD service	`RequestOnboardMonitoringTestResults (0x06)`

Table 2.1. Affected services

  Rationale:

  This limitation is necessary because of the inability of the underlying services to distinguish calling contexts to the functions called for executing read-operations on DID signals in the application. In case of OBD service 6 there is a possible reentrancy issue that needs to be guarded against for the interface Xxx_GetDTRValue.

- internal ROE-triggered request results in a ConditionsNotCorrect (0x22) NRC

An internal ROE-triggered request for service `ReadDTCInformation (0x19)` occurring at the same time as an external request results in a ConditionsNotCorrect (0x22) NRC when it is being serviced for either service:

- `ClearDiagnosticInformation (0x14)`
- `ReadDTCInformation (0x19)`
- `ReadGenericInformation (0xAF)`
- `RequestPowerTrainFreezeFrameData ($02)`
- `Request emission-related diagnostic trouble codes ($03)`
- `Clear/Reset emission-related diagnostic information ($04)`
- `RequestOnboardMonitoringTestResults ($06)`
- `Request emission-related diagnostic trouble codes detected during current or last completed driving cycle ($07)`
- `Request emission-related diagnostic trouble codes with permanent status ($0A)`

Rationale:

The limitation is necessary because of the inability to distinguish calling contexts to the functions called in the Dem module for executing needed operations.

- Limited the value of parameter `DcmDspPidDataPos`

Description:

According to Dcm SWS of AUTOSAR R4.0, the value of the parameter `DcmDspPidDataPos` ranges from 0 to 2040. But for the current implementation, the range is limited from 1 to 2039.

Rationale:

A `DcmDspPidDataPos` of 2040 (starting on 0) is out of range of possible 255 bytes.

Requirements:

Dcm866_Conf

- Limited the multiplicity of container `DcmDspSession` if configuration parameter `DcmModeDeclarationSupport` is set to true

Description:

According to Dcm SWS of AUTOSAR R4.0.3, the multiplicity of the container `DcmDspSession` is 1 to *. But for the current implementation, the multiplicity is limited to 1 if configuration parameter `DcmModeDeclarationSupport` is set to true.

This implies that different DcmDslProtocolRow cannot use different DcmDslProtocolSessionRef references. If selective session availability per protocol is needed, it can be achieved in the following way:

1. Configure different entries of the UDS service DiagnosticSessionControl (0x10) in different DcmDsdServiceTable containers.
2. Configure different DcmDsdSubService support for each UDS service DiagnosticSessionControl (0x10) entry.
3. Configure each DcmDslProtocolRow to use the appropriate DcmDsdServiceTable.

Rationale:

If container DcmDspSession contains more than one entry and the configuration parameter DcmModeDeclarationSupport is set to true, then there are two default sessions and thus two references are generated to an initial mode declaration. It is not possible to have two references to an initial mode declaration. See also https://www.autosar.org/bugzilla/show_bug.cgi?id=58711.

- Use of RequestTransferExit (0x37) is limited

Description:

If the configuration parameter DcmDataTransferServicesASRVersion is set to the value DATA_TRANSFER_SERVICES_AUTOSAR_430, the processing of RequestTransferExit (0x37) service on a protocol with a shared Rx/Tx buffer is not possible. If a shared Rx/Tx buffer is desired, the user must ensure that the request data is not overwritten when Dcm transmits the response so that the request data is available for subsequent calls. This can be done by storing the request data in a local copy.

A shared Rx/Tx buffer means that for a DcmDslProtocolRow, the configuration parameter DcmDslProtocolTxBufferID is the same as DcmDslProtocolRxBufferID of the receive channel belonging to the DcmDslMainConnection on which the request is processed.

Rationale:

This limitation is necessary because in a configuration with a shared Rx/Tx buffer, transferRequestParameterRecord and transferResponseParameterRecord point to the same address. This means if the user writes data in transferResponseParameterRecord via Dcm_ProcessRequestTransferExit, the request data is not longer available for potential calls of the Svch with OpStatus DCM_PENDING.

- Limited check for NRC 0x71 (transferDataSuspended) conditions

Description

Dcm cannot perform the check for NRC 0x71 (transferDataSuspended) conditions for a TransferData (0x36) request before the check for NRC 0x73 (wrongBlockSequenceCounter) conditions as specified in Figure 28 from ISO 4229-1-2013

Rationale:

According to SWS_Dcm_01173 from AUTOSAR 4.3.0, NRC 0x71 (transferDataSuspended) is the responsibility of the callout function (`Dcm_ReadMemory` for an upload request or `Dcm_WriteMemory` for a download request). The callout function is called after the check for NRC 0x73 (wrongBlockSequenceCounter) is done.

- Limited checks for UDS services `RequestDownload (0x34)` and `RequestUpload (0x35)`

Description:

Dcm can not perform the following checks for the UDS services `RequestDownload (0x34)` and `RequestUpload (0x35)`:

- Validity of `dataFormatIdentifier` for RequestDownload (0x34) as specified in Figure 26 from ISO 4229-1-2013
- Validity of `dataFormatIdentifier` for RequestUpload (0x35) as specified in Figure 27 from ISO 4229-1-2013

Rationale:

Dcm has no available information to decide whether a `dataFormatIdentifier` received in a request is valid or not. Dcm defers this responsibility to the application, as specified in SWS_Dcm_01132 and SWS_Dcm_01133 from AUTOSAR 4.3.0.

- The feature which is implemented along with the introduction of parameter DcmDsdDisableGenericServiceImplementation has the following limitations

Description:

The NRC responses from the internal generic Service Handler shall not be changed.

The external service handler shall call the internal generic Service Handler only via the provided API.

The requests which are processed by the external service handler are bound to the same limitations as the internal generic service handlers.

Rationale:

Those limitations are determined by the method used to call internal generic Service Handlers from the external service handlers.

- The Dcm does not support the configuration values `DCM_CONTROLMASK_NO` and `DCM_CONTROLMASK_INTERNAL` for `DcmDspDidControlMask`

Description:

The Dcm supports only the following configuration values for `DcmDspDidControlMask`:

- DCM_CONTROLMASK_EXTERNAL

DCM_CONTROLMASK_NO is not supported

DCM_CONTROLMASK_INTERNAL is not supported

Rationale:

The functionality related to DCM_CONTROLMASK_NO can be obtained by disabling the DcmDspDidControlMask parameter but in this case the number of signals of the controlled DID is limited to one. The functionality related to DCM_CONTROLMASK_INTERNAL can be obtained by disabling the DcmDspDidControlMask parameter and relying on AUTOSAR 4.0.3 behaviour. This means that that DcmDspDidControlMask shall correspond in length to the number of controlled signals.

- DcmDspRoutineSignalLength shall be an even multiple of base signal bit size for array types

Description:

- If DcmDspRoutineSignalType is set to SINT8_N or UINT8_N, then DcmDspRoutineSignalLength shall be an even multiple of 8
- If DcmDspRoutineSignalType is set to SINT16_N or UINT16_N, then DcmDspRoutineSignalLength shall be an even multiple of 16
- If DcmDspRoutineSignalType is set to SINT32_N or UINT32_N, then DcmDspRoutineSignalLength shall be an even multiple of 32

Rationale:

According to AUTOSAR 4.3.0 specifications (SWS_Dcm_91041, SWS_Dcm_91043, SWS_Dcm_91045, SWS_Dcm_91047, SWS_Dcm_91049) an array type signal has DcmDspRoutineParameterSize elements. Since the Dcm uses AUTOSAR 4.0.3 configuration schema, DcmDspRoutineSignalLength is limited in order to maintain the same understanding regarding array type signals size.

- The protocol currently preempting another protocol cannot itself be preempted

Description:

ProtocolA with higher priority cannot preempt protocolB with lower priority while protocolB is preempting protocol C ProtocolA's request will be rejected even if it has higher priority than the other two protocols.

Rationale:

The AUTOSAR 4.0.3 specification requirement Dcm015 is not applicable while no protocol is running, one is waiting for a preemption to end and the second one is being cancelled as a result of preemption. The AUTOSAR specification does not give more details on this use case.

- Limited multiplicity of DcmDslCallbackDCMRequestService

Description:

According to Dcm SWS of AUTOSAR R4.2.1, the multiplicity of the container `DcmDslCallbackDCMRequestService` is 0 to *. But for the current implementation, the multiplicity is limited from 0 to 255.

Requirements:

Dcm690_Conf, ECUC_Dcm_00690

Rationale:

Autosar gives no maximum number. The value 255 is expected to be sufficient for most use cases, since in any way, one function call can be used as a dispatcher for multiple condition checks if need to be, as all these calls are executed with the same parameter values.

- An OBD request cancells a request to switch to a non-default session

Description:

If an OBD request is received while the Server is processing a UDS request for service DiagnosticSessionControl (0x10), and the request is for a non-default session, the Server shall not switch to the non-default session.

Rationale:

To a Tester, this would be indistinguishable from the case in which the OBD request arrives right after the UDS request has completed, and therefore would result in a transition to the default session immediately. The OBD request thus would therefore cancel the transition to the non-default session.

- Parallel service processing in conjunction with asynchronous service handling is limited

Description:

Protocols of more than one `DcmDslProtocolStackNumber` shall not reference service tables `DcmDslProtocolSIDTable` containing services which have asynchronous service processing enabled, i.e. `DcmAsyncServiceExecution` set to 'true'.

Rationale:

This limitation is necessary because only one single asynchronous service processor instance is available and therefore two services cannot be executed asynchronously in parallel.

- The functionality related to `TxConfirmationNotification` is not available for ResponseOnEvent and Periodic tranmissions

Description:

The Dcm does not call `Dcm_ApplicationTransmissionConfirmation()` on `ResponseOnEvent` and `Periodic` responses transmission confirmation.

Rationale:

`ResponseOnEvent` and `Periodic` transmissions are not triggered directly by external requests, they are internally generated by the Dcm.

- External sub-service handlers for service DiagnosticSessionControl (0x10) are not supported

  Description:

  Configuration of external sub-service handlers for service DiagnosticSessionControl (0x10) is not possible.

  Rationale:

  This limitation is necessary because the sub-service identifier only represents a session identifier.

- The handling of multiple "availability InfoTypes" for `OBD service Request Vehicle Information 0x09`

  Description:

  According to chapter "8.9.2.2 Request vehicle information response message definition (report supported INFOTYPE)" from ISO 15031-5:2011, states the following: "ECU(s) shall respond to all supported ranges if requested. A range is defined as a block of 32 INFOTYPEs(e.g. range #1: INFOTYPE 0x01 to 0x20). The ECU shall not respond to unsupported INFOTYPE ranges unless subsequent ranges have a supported INFOTYPE(s)." In the current implementation, on reception of an OBD Service $09 request for multiple availability InfoTypes the Dcm shall returns the availability Infotypes that fit into the response buffer.

  Rationale:

  The integrator has to make sure that the response buffer size determined by the `DcmDslBufferSize` configuration parameter is sufficiently sized to receive the data returned by the provider of the data.

- Metadata support is limited

  Description:

  Pdus which require metadata handling by the the Dcm shall have the EcuC configuration parameters `PduId` and `MetaDataTypeRef` configured.

  Rationale:

  This limitation is necessary because the `PduId` and `MetaDataTypeRef` are needed by the Dcm and EcuC modules to handle the metadata.

- Length is limited for `SOURCE_ADDRESS_16` and `TARGET_ADDRESS_16`

  Description:

  `SOURCE_ADDRESS_16` and `TARGET_ADDRESS_16` metadata items lengths for Pdus which are referenced by the Dcm shall be 2 bytes.

Rationale:

This limitation is necessary in order to match SourceAddress data type of `Xxx_Indication()` and `Xxx_Confirmation()` operations of the ServiceRequestNotification interface.

- The Dcm does not support `ReadScalingDataByIdentifier (0x24)` service request for a Dynamically-defined DID and Range DID

Description:

The Dcm does not support service `ReadScalingDataByIdentifier (0x24)` request for a Dynamically-defined Data Identifier (i.e. a DID whose individual DID configuration `DcmDspDid` references a `DcmDspDidInfo` that has the configuration parameter `DcmDspDidDynamicallyDefined` enabled) and for a Range Data Identifier (i.e. a DID that is configured via configuration parameter `DcmDspDidRange`). Consequently, the Dcm transmits a negative response with an NRC code equal to the value 0x31 (requestOutOfRange), if the service `ReadScalingDataByIdentifier (0x24)` is requested for any of the previously mentioned DIDs.

Rationale:

The Autosar specifications (4.0.3 and R20-11) as well as the ISO specifications (14229 - 2013 and 2020) does not mention anything on how the Scaling Information shall be retrieved for a Dynamically-defined DID and Range DID.

- The range for `DcmDspSupportedAddressAndLengthFormatIdentifier` is limited

Description:

For UDS services `ReadMemoryByAddress (0x23)`, `WriteMemoryByAddress(0x3D)`, `DynamicallyDefineDataIdentifier (0x2C)`, `RequestDownload (0x34)` and `RequestUpload (0x35)` the range for `DcmDspSupportedAddressAndLengthFormatIdentifier` is limited to 0x11, 0x12, 0x13, 0x14, 0x21, 0x22, 0x23, 0x24, 0x31, 0x32, 0x33, 0x34, 0x41, 0x42, 0x43, 0x44 values.

Rationale:

This limitation is necessary because the following APIs, `Dcm_WriteMemory()`, `Dcm_ReadMemory()`, `Dcm_ProcessRequestDownload()`, `Dcm_ProcessRequestUpload()` that are used by the previous mentioned services support maximum 4 Bytes for memoryAddress and memorySize.

- The range for `DcmDslProtocolRxConnectionId` is limited

Description:

The range for `DcmDslProtocolRxConnectionId` is 0 - 65534. The value 65535 is excluded.

Rationale:

This limitation is necessary because the value 65535 is used internally since `DcmDslProtocolRxConnectionId` parameter is currently used only for UDS service `Authentication (0x29)` and it is not mandatory yet.

- The range for `DcmDspAuthenticationDefaultSessionTimeOut` is limited

  Description:

  According to ECUC_Dcm_01161 of AUTOSAR specification R20-11 the range for `DcmDspAuthenticationDefaultSessionTimeOut` is [0 .. INF[seconds. But for the current implementation, the range is limited to [0.001 .. 4294967] seconds.

  Rationale:

  A minimum value of 0.001 is the minimum value for a `DcmTaskTime`. Autosar gives no maximum value. The minimum value of 0.001 second and the maximum value of 4294967 seconds is expected to be sufficient for most use cases.

- The value of `DcmDspDataUsePort` is limited in case `DcmDspDidControlMask` is configured

  Description:

  The value of `DcmDspDataUsePort` is limited to `USE_DATA_SYNCH_FNC` for all DIDs that have a `DcmDspDidInfoRef` reference with configuration parameter `DcmDspDidControlMask` set to `DCM_CONTROLMASK_EXTERNAL`.

  Rationale:

  The functionality related to `DcmDspDataUsePort` set to other value than `USE_DATA_SYNCH_FNC` in case `DcmDspDidControlMask` is configured is not yet supported.

- External sub-service handlers for UDS service `Authentication (0x29)` are not supported

  Description:

  Usage of external sub-service handlers for UDS service `Authentication (0x29)` is not possible.

  Rationale:

  This limitation is necessary because the verification of authentication permission rights for service requests are handled by Dcm.

- The auto-calculation for availability DID '0xF4E0' is limited in case `DcmDspDidAvailabilityCalculation` is set to 'False'

  Description:

  When availability DID '0xF4E0' is requested via `ReadDataByIdentifier (0x22)` UDS service, and the configuration parameter `DcmDspDidAvailabilityCalculation` is set to 'False', the availability

bit-map response shall contain the configured PIDs from '0xE1' to '0xFF' range and shall not include the availability of the configured DIDs from '0xF500' to '0xF5FF' range.

Rationale:

The auto-calculation of DIDs from '0xF500' to '0xF5FF' range is performed when `DcmDspDidAvailabilityCalculation` is set to 'True'

2.6. Open-source software

The `Dcm` module does not use open-source software.