Document Title	SWS_PDURouter: Complete Change Documentation 4.3.0 - 4.3.1
Document Owner	AUTOSAR
Document Responsibility	AUTOSAR
Document Identification No	695

Document Status	Final
Part of AUTOSAR Standard	Classic Platform
Part of Standard Release	4.3.1

Table of Contents

1 SWS_PDURouter

1.1 Specification Item ECUC_PduR_00304

Trace References:

Content:

Name	PduRDestTxBufferRef		
Description	Reference to a buffer that is allocated in the PduRTxBufferTableR. This buffer is required for communication interface gatewaying, and for transport protocol gatewayingfor single frame routing.		
Multiplicity	0..*		
Type	Reference to [PduRTxBuffer]		
Post-Build Variant Multiplicity	true		
Post-Build Variant Value	true		
Multiplicity Configuration Class	Pre-compile time	X	VARIANT-PRE-COMPILE
	Link time	−	
	Post-build time	X	VARIANT-POST-BUILD
Value Configuration Class	Pre-compile time	X	VARIANT-PRE-COMPILE
	Link time	−	
	Post-build time	X	VARIANT-POST-BUILD
Scope / Dependency	scope: local		

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76222: [PduR] Buffer description

  Problem description:

  See proposed solution

  Agreed solution:

  "ECUC_PduR_00304 PduRDestTxBufferRef"
  "Reference to a buffer that is allocated in the PduRTxBufferTable. This buffer is required for communication interface gatewaying, and for transport protocol gatewaying for single frame routing."
  to
  "Reference to a buffer in the PduR. This buffer is required for communication interface gatewaying, and for transport protocol gatewaying."

"ECUC_PduR_00336 PduRTxBuffer"
"Specifies a buffer used for gatewaying via communication interfaces or for single frames of transport protocols."
to
"Specifies a buffer used for gatewaying via communication interfaces or transport protocols."
–Last change on issue 76222 comment 8–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.2 Specification Item ECUC_PduR_00310

Trace References:

Content:

Container Name	PduRRoutingPaths
Description	Represents one table of routing paths. This routing table allows multiple configurations that can be used to create several routing tables in the same configuration. This is mainly used for post-build (e.g. post-build selectable) but can be used by pre-compile and link-time for variant handling.
Configuration Parameters	

Included parameters:

Included Parameters	
Parameter Name	SWS Item ID
PduRConfigurationId	ECUC_PduR_00327
PduRMaxRoutingPathCnt	ECUC_PduR_00350
PduRMaxRoutingPathGroupCnt	ECUC_PduR_00348

Included containers:

Included Containers		
Container Name	Multiplicity	Scope / Dependency
PduRDestPdu	0..*	This container is a subcontainer of PduRRoutingPath and specifies one destination for the PDU to be routed.
PduRRoutingPath	0..*	This container is a subcontainer of PduRRoutingTable and specifies the routing path of a PDU.

Included Containers		
Container Name	Multiplicity	Scope / Dependency
PduRRoutingPathGroup	0..*	This container groups routing path destinations. Destinations are used instead of routing paths since a routing path can be 1:n. It is desirable to be able to enable/disable a specific bus (i.e. a destination) rather than a routing path. Of course it is possible to create groups that covers specific routing paths as well. Enabling and disabling of routing path groups are made using the PduR API
PduRSrcPdu	0..*	This container is a subcontainer of Pdu RRoutingPath and specifies the source of the PDU to be routed.
PduRTxBuffer	0..*	Specifies a buffer used for gatewaying via communication interfaces or for single frames of transport protocols.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76222: [PduR] Buffer description

Problem description:

See proposed solution

Agreed solution:

"ECUC_PduR_00304 PduRDestTxBufferRef"
"Reference to a buffer that is allocated in the PduRTxBufferTable. This buffer is required for communication interface gatewaying, and for transport protocol gatewaying for single frame routing."
to
"Reference to a buffer in the PduR. This buffer is required for communication interface gatewaying, and for transport protocol gatewaying."

"ECUC_PduR_00336 PduRTxBuffer"
"Specifies a buffer used for gatewaying via communication interfaces or for
single frames of transport protocols."
to
"Specifies a buffer used for gatewaying via communication interfaces or transport protocols."
–Last change on issue 76222 comment 8–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.3 Specification Item ECUC_PduR_00319

Trace References:

Content:

Name	PduRUseTagPduRBswModules.PduRUseTag		
Description	This parameter, if set to true, enables the usage of the tag (<up>) in the following API calls: • PduR_<Up>CancelReceive • PduR_<Up>CancelTransmit • PduR_<Up>ChangeParameter Example: If used by COM and the parameter is enabled the PduR_ComCancelTransmit is used. The background is that upper layer modules differ in usage of this tag (e.g. COM is using the tag, DCM is not).		
Multiplicity	1		
Type	EcucBooleanParamDef		
Default value	–		
Post-Build Variant Value	false		
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: local		

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

• RfC #76491: [PduR] Remove forwarding of ChangeParameter API from PduR

Problem description:

The ChangeParameter API is TP specific and there is no use case to call this API through the PduR (CDD or integration code that requires a change in the TP parameter will call the API directly in the TP module)

Agreed solution:

PDUR_SWS
Remove references from ch. 5 "Dependencies to other modules"
Remove ch. 8.4. "Change transport protocol parameter"

BSW_Model

set API PduR_<User:Up>ChangeParameter [SWS_PduR_00482] to obsolete
set optional interface from [SWS_PduR_00424] to obsolete

ECUC model
set [ECUC_PduR_00326] PduRChangeParameterApi to obsolete
remove reference in description from [ECUC_PduR_00319] PduRUseTag
–Last change on issue 76491 comment 12–

BW-C-Level:

Application	Specification	Bus
1	4	1

1.4 Specification Item ECUC_PduR_00326

Trace References:

Content:

Name	PduRChangeParameterApiPduRBswModules.PduRChangeParameterApi		
Description	This parameter, if set to true, enables the PduR_<Up>ChangeParameter Api for this Module. Tags: atp.Status=obsolete		
Multiplicity	1		
Type	EcucBooleanParamDef		
Default value	–		
Post-Build Variant Value	false		
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: local		

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76491: [PduR] Remove forwarding of ChangeParameter API from PduR

  Problem description:

  The ChangeParameter API is TP specific and there is no use case to call this API through the PduR (CDD or integration code that requires a change in the TP parameter will call the API directly in the TP module)

  Agreed solution:

PDUR_SWS
Remove references from ch. 5 "Dependencies to other modules"
Remove ch. 8.4. "Change transport protocol parameter"

BSW_Model
set API PduR_<User:Up>ChangeParameter [SWS_PduR_00482] to obsolete
set optional interface from [SWS_PduR_00424] to obsolete

ECUC model
set [ECUC_PduR_00326] PduRChangeParameterApi to obsolete
remove reference in description from [ECUC_PduR_00319] PduRUseTag
–Last change on issue 76491 comment 12–

BW-C-Level:

Application	Specification	Bus
1	4	1

1.5   Specification Item ECUC_PduR_00336

Trace References:

Content:

Container Name	PduRTxBuffer		
Description	Specifies a buffer used for gatewaying via communication interfaces or for single frames of transport protocols.		
Post-Build Variant Multiplicity	false		
Multiplicity Configuration Class	Pre-compile time	X	VARIANT-PRE-COMPILE, VARIANT-POST-BUILD
	Link time	−	
	Post-build time	−	
Configuration Parameters			

Included parameters:

Included Parameters	
Parameter Name	SWS Item ID
PduRPduMaxLength	ECUC_PduR_00324

Included containers:

No Included Containers

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76222: [PduR] Buffer description

  Problem description:

  See proposed solution

  Agreed solution:

  "ECUC_PduR_00304 PduRDestTxBufferRef"
  "Reference to a buffer that is allocated in the PduRTxBufferTable. This buffer is required for communication interface gatewaying, and for transport protocol gatewaying for single frame routing."
  to
  "Reference to a buffer in the PduR. This buffer is required for communication interface gatewaying, and for transport protocol gatewaying."

  "ECUC_PduR_00336 PduRTxBuffer"
  "Specifies a buffer used for gatewaying via communication interfaces or for single frames of transport protocols."
  to
  "Specifies a buffer used for gatewaying via communication interfaces or transport protocols."
  –Last change on issue 76222 comment 8–

  BW-C-Level:

Application	Specification	Bus
1	1	1

1.6 Specification Item SWS_PduR_00100

Trace References:

SRS_BSW_00337, SRS_BSW_00323, SRS_BSW_00452

Content:

The following Development Errors and exceptions shall be detectable by the PDU Router module depending on its build version (development/production mode):

Type or error	Related error code	Value [hex]
Invalid configuration pointer	PDUR_E_INIT_FAILED	0x00

Type or error	Related error code	Value [hex]
API service used without module initialization or PduR_Init called in any state other than PDUR_UNINIT	PDUR_E_UNINIT	0x01
Invalid PDU identifier	PDUR_E_PDU_ID_INVALID	0x02
If the routing table is invalid that is given to the PduR_EnableRouting or Pdu R_DisableRouting functions	PDUR_E_ROUTING_PATH_GROUP_ID_INVALID	0x08
Null pointer has been passed as an argument	PDUR_E_PARAM_POINTER	0x09

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #59085: Rollout of 'Runtime errors'

  Problem description:

  Inconsistencies in SWS with semantics of Default errors
  –Last change on issue 59085 comment 26–

  Agreed solution:

  solution in Column "G" of the new attachment
  https://www.autosar.org/bugzilla/attachment.cgi?id=4604

  Notes:
  - It is not enough just to migrate the error from one classification table to another. Please also check the related requirements (and background information) which is referring to that error and adapt them if needed.
  - The review task of the ITs shall be done by the WP to which the specification "belongs".

  *** BSW UML Model ***
  SWS_CanNm:
  ————-
  Chapter 8.6.1 Optional Interfaces:
  Add within SWS_CanNm_00325 the API function Det_ReportRunTimeError

  SWS_LinIf:
  ————-
  SWS_LinIf_00359: add Det_ReportRuntimeError

  SWS_UdpNm:
  ————-

Replace UDPNM_E_NO_INIT with UDPNM_E_UNINIT in description of API UdpNm_MainFunction_<Instance Id> (SWS_UdpNm_00234)

*** ECUC XML ***
Not affected. No configuration of runtime error reporting required (see SWS BSW General).
–Last change on issue 59085 comment 88–

BW-C-Level:

Application	Specification	Bus
1	4	1

1.7   Specification Item SWS_PduR_00119

Trace References:

SRS_BSW_00406

Content:

If the PDU Router module has not been initialized (state PDUR_UNINIT), all functions except PduR_Init and PduR_GetVersionInfo shall report the error PDUR_E_INVALID_REQUEST UNINIT via the DET when called, when PduRDevError Detect is enabled.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #59085: Rollout of 'Runtime errors'

  Problem description:

  Inconsistencies in SWS with semantics of Default errors
  –Last change on issue 59085 comment 26–

  Agreed solution:

  solution in Column "G" of the new attachment
  https://www.autosar.org/bugzilla/attachment.cgi?id=4604

  Notes:
  - It is not enough just to migrate the error from one classification table to another. Please also check the related requirements (and background information) which is referring to that error and adapt them if needed.
  - The review task of the ITs shall be done by the WP to which the specification

"belongs".

*** BSW UML Model ***
SWS_CanNm:
———-
Chapter 8.6.1 Optional Interfaces:
Add within SWS_CanNm_00325 the API function Det_ReportRunTimeError

SWS_LinIf:
———-
SWS_LinIf_00359: add Det_ReportRuntimeError

SWS_UdpNm:
———-
Replace UDPNM_E_NO_INIT with UDPNM_E_UNINIT in description of API
UdpNm_MainFunction_<Instance Id> (SWS_UdpNm_00234)

*** ECUC XML ***
Not affected. No configuration of runtime error reporting required (see SWS BSW
General).
–Last change on issue 59085 comment 88–

BW-C-Level:

Application	Specification	Bus
1	4	1

1.8 Specification Item SWS_PduR_00424

Trace References:

SRS_BSW_00384

Content:

API function	Description
(obsolete)<Provider:LoTp>_ChangeParameter	Request to change a specific transport protocol parameter (e.g. block size).
<Provider:Lo>_CancelTransmit	Requests cancellation of an ongoing transmission of a PDU in a lower layer communication module.
<Provider:Lo>_Transmit	Requests transmission of a PDU.

API function	Description
<Provider:LoTp>_CancelReceive	Requests cancellation of an ongoing reception of a PDU in a lower layer transport protocol module.
<Provider:LoTp>_CancelTransmit	Requests cancellation of an ongoing transmission of a PDU in a lower layer communication module.
<Provider:LoTp>_ChangeParameter	Request to change a specific transport protocol parameter (e.g. block size).
<Provider:LoTp>_Transmit	Requests transmission of a PDU.
<Provider:Up>_RxIndication	Indication of a received PDU from a lower layer communication interface module.
<Provider:Up>_TriggerTransmit	Within this API, the upper layer module (called module) shall check whether the available data fits into the buffer size reported by PduInfoPtr->SduLength. If it fits, it shall copy its data into the buffer provided by PduInfoPtr->SduDataPtr and update the length of the actual copied data in PduInfoPtr->SduLength. If not, it returns E_NOT_OK without changing PduInfoPtr.
<Provider:Up>_TxConfirmation	The lower layer communication interface module confirms the transmission of a PDU, or the failure to transmit a PDU.
<Provider:UpTp>_CopyRxData	This function is called to provide the received data of an I-PDU segment (N-PDU) to the upper layer. Each call to this function provides the next part of the I-PDU data. The size of the remaining data buffer is written to the position indicated by bufferSizePtr.
<Provider:UpTp>_CopyTxData	This function is called to acquire the transmit data of an I-PDU segment (N-PDU). Each call to this function provides the next part of the I-PDU data unless retry->TpDataState is TP_DATARETRY. In this case the function restarts to copy the data beginning at the offset from the current position indicated by retry->TxTpDataCnt. The size of the remaining data is written to the position indicated by availableDataPtr.
<Provider:UpTp>_StartOfReception	This function is called at the start of receiving an N-SDU. The N-SDU might be fragmented into multiple N-PDUs (FF with one or more following CFs) or might consist of a single N-PDU (SF). The service shall provide the currently available maximum buffer size when invoked with TpSduLength equal to 0.
<Provider:UpTp>_TpRxIndication	Called after an I-PDU has been received via the TP API, the result indicates whether the transmission was successful or not.
<Provider:UpTp>_TpTxConfirmation	This function is called after the I-PDU has been transmitted on its network, the result indicates whether the transmission was successful or not.
Det_ReportError	Service to report development errors.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76404: [Det] Clarifications on runtime errors

  Problem description:

  There are several uncertainties/problems in the SWS DET:
  1. According to SWS_Det_00180, the callouts should have the same signatures

as the corresponding DET functions, but they are void(void) (SWS_Det_00181, SWS_Det_00184, SWS_Det_00187).

2. Section 8.2.3.1 does not describe how the instance ID is passed to DET.

3. Configuration of header files for all three error type callouts are missing.

4. Why does the development error callout reside in DetNotification, while the other two callouts reside in DetGeneral?

5. The limitation in section 4.1 regarding "supervisor mode" does not really make sense. It is assumed that the DET is ignorant regarding the call context, and the software receiving DET callbacks (like DLT or the implementers of the callouts) need to take care of resolving the calling context, if necessary (e.g. in multi-core environments).

6. SWS_Det_00302 defines several runtime errors. But apart from DET_E_CANNOT_REPORT, it is unclear in which situation these errors could be reported by DET: For errors reported by BSW, the DET has no means to validate anything that could lead to such an error. And for SWCs, the modeling already takes care that DET_E_WRONG_MODULE and DET_E_WRONG_INSTANCE cannot occur, while the other two errors can also not be checked by DET without further configuration.

7. Det_ReportTransientFault (SWS_Det_01003) shall return the return value of a configured callout. But what shall happen if more than one callout exisits, and the return different values?

8. SWS_Det_00052: The only API that can result in DET_E_PARAM_POINTER is Det_GetVersionInfo (as the error description mentions correctly). Please reformulate this requirement and move it to section 8.1.3.6 "Det_GetVersionInfo".

–Last change on issue 76404 comment 13–

Agreed solution:

1.
~change SWS_Det_00181/184/187 such that signatures match the APIs
~Figures 3,5, and 7 to be corrected (return missing)

5. remove from 4.1. the sentence: "It is assumed that the whole Basic Software runs in supervisor mode or the switch to supervisor mode is done by a system call within the error reporting function of the DET module."

6. remove SWS_Det_00302 and SWS_Det_00303 and all included errors

7. change SWS_Det_01003 (Return Value-Part only): "Std_ReturnType" If no callout exists it shall return E_OK, otherwise it shall return the value of the configured callout. In case several callouts are configured the logical or (sum) of the callout return values shall be returned. Rationale: since E_OK=0, E_OK will be only returned if all are E_OK, and for multiple error codes there is a good chance to detect several of them.

8. change SWS_Det_00052 from "in case a null pointer error occurs." to "in case a null pointer error occurs in Det_GetVersionInfo." Do not move the requirement, since otherwise the section 7.7 would be empty, but add the following sentence to

8.1.3.6: "In case a null pointer is passed, DET_E_PARAM_POINTER is returned, see SWS_Det_00052."

–Last change on issue 76404 comment 30–

BW-C-Level:

Application	Specification	Bus
1	4	1

- RfC #77935: [PduR] Misleading description of CopyRxData

Problem description:

_____

Name: Martin Schlodder
Role: Member of WP-A2

_____

Description/Motivation:

The description of the CopyRxData API says: "The size of the remaining data is written to the position indicated by bufferSizePtr."

This text seems to have been copied from the CopyTxData call, where it is correct. CopyRxData should talk about "remaining buffer", not "remaining data".

Agreed solution:

In the description of the API PduR_<User:LoTp>CopyRxData (SWS_PduR_00512), replace "remaining data" by "remaining buffer".

BW-C-Level:

Application	Specification	Bus
1	1	1

1.9   Specification Item SWS_PduR_00482

Trace References:

SRS_PduR_06026,   SRS_PduR_06114,   SRS_PduR_06115,   SRS_PduR_06116, SRS_BSW_00310

Content:

Service name:	PduR_<User:Up>ChangeParameter(obsolete)PduR_<User:Up>ChangeParameter(obsolete)

Syntax:	Std_ReturnType PduR_<User:Up>ChangeParameter(obsolete)(PduIdType id, TPParameterType parameter, uint16 value)	
Service ID[hex]:	0x4b	
Sync/Async:	Synchronous	
Reentrancy:	Non Reentrant	
Parameters (in):	idPduR_<User:Up>Change Parameter(obsolete).id	Identification of the PDU which the parameter change shall affect.
	parameterPduR_<User:Up>Change Parameter(obsolete).parameter	ID of the parameter that shall be changed.
	valuePduR_<User:Up>Change Parameter(obsolete).value	The new value of the parameter.
Parameters (inout):	None	
Parameters (out):	None	
Return value:	Std_ReturnType	E_OK: The parameter was changed successfully. E_NOT_OK: The parameter change was rejected.
Description:	Request to change a specific transport protocol parameter (e.g. block size). Tags: atp.Status=obsolete	

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76491: [PduR] Remove forwarding of ChangeParameter API from PduR

  Problem description:

  The ChangeParameter API is TP specific and there is no use case to call this API through the PduR (CDD or integration code that requires a change in the TP parameter will call the API directly in the TP module)

  Agreed solution:

  PDUR_SWS
  Remove references from ch. 5 "Dependencies to other modules"
  Remove ch. 8.4. "Change transport protocol parameter"

  BSW_Model
  set API PduR_<User:Up>ChangeParameter [SWS_PduR_00482] to obsolete
  set optional interface from [SWS_PduR_00424] to obsolete

  ECUC model
  set [ECUC_PduR_00326] PduRChangeParameterApi to obsolete
  remove reference in description from [ECUC_PduR_00319] PduRUseTag
  –Last change on issue 76491 comment 12–

Document ID 695: ChangeDocumentation

BW-C-Level:

Application	Specification	Bus
1	4	1

1.10   Specification Item SWS_PduR_00512

Trace References:

SRS_PduR_06026, SRS_PduR_06121, SRS_BSW_00310

Content:

Service name:	PduR_<User:LoTp>CopyRxDataPduR_<User:LoTp>CopyRxData	
Syntax:	BufReq_ReturnType PduR_<User:LoTp>CopyRxData(PduIdType id, const PduInfoType* info, PduLengthType* bufferSizePtr)	
Service ID[hex]:	0x44	
Sync/Async:	Synchronous	
Reentrancy:	Reentrant	
Parameters (in):	idPduR_<User:LoTp>CopyRxData.id	Identification of the received I-PDU.
	infoPduR_<User:LoTp>CopyRx Data.info	Provides the source buffer (SduDataPtr) and the number of bytes to be copied (SduLength). An SduLength of 0 can be used to query the current amount of available buffer in the upper layer module. In this case, the SduDataPtr may be a NULL_PTR.
Parameters (inout):	None	
Parameters (out):	bufferSizePtrPduR_<User:LoTp>Copy RxData.bufferSizePtr	Available receive buffer after data has been copied.
Return value:	BufReq_ReturnType	BUFREQ_OK: Data copied successfully BUFREQ_E_NOT_OK: Data was not copied because an error occurred.
Description:	This function is called to provide the received data of an I-PDU segment (N-PDU) to the upper layer. Each call to this function provides the next part of the I-PDU data. The size of the remaining data buffer is written to the position indicated by bufferSize Ptr.	

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77935: [PduR] Misleading description of CopyRxData

  Problem description:

  _____

  Name: Martin Schlodder

Role: Member of WP-A2

Description/Motivation:

The description of the CopyRxData API says: "The size of the remaining data is written to the position indicated by bufferSizePtr."

This text seems to have been copied from the CopyTxData call, where it is correct. CopyRxData should talk about "remaining buffer", not "remaining data".

Agreed solution:

In the description of the API PduR_<User:LoTp>CopyRxData (SWS_PduR_00512), replace "remaining data" by "remaining buffer".

BW-C-Level:

Application	Specification	Bus
1	1	1

1.11   Specification Item SWS_PduR_00518

Trace References:

SRS_PduR_06026, SRS_PduR_06121, SRS_BSW_00310

Content:

Service name:	PduR_<User:LoTp>CopyTxDataPduR_<User:LoTp>CopyTxData
Syntax:	BufReq_ReturnType PduR_<User:LoTp>CopyTxData(PduIdType id, const PduInfoType* info, const RetryInfoType* retry, PduLengthType* availableDataPtr)
Service ID[hex]:	0x43
Sync/Async:	Synchronous
Reentrancy:	Reentrant

Parameters (in):	idPduR_<User:LoTp>CopyTxData.id	Identification of the transmitted I-PDU.
	infoPduR_<User:LoTp>CopyTxData.info	Provides the destination buffer (SduData Ptr) and the number of bytes to be copied (SduLength). If not enough transmit data is available, no data is copied by the upper layer module and BUFREQ_E_BUSY is returned. The lower layer module may retry the call. An SduLength of 0 can be used to indicate state changes in the retry parameter or to query the current amount of available data in the upper layer module. In this case, the SduDataPtr may be a NULL_PTR.
	retryPduR_<User:LoTp>CopyTx Data.retry	This parameter is used to acknowledge transmitted data or to retransmit data after transmission problems. If the retry parameter is a NULL_PTR, it indicates that the transmit data can be removed from the buffer immediately after it has been copied. Otherwise, the retry parameter must point to a valid RetryInfo Type element. If TpDataState indicates TP_CONFPENDING, the previously copied data must remain in the TP buffer to be available for error recovery. TP_DATACONF indicates that all data that has been copied before this call is confirmed and can be removed from the TP buffer. Data copied by this API call is excluded and will be confirmed later. TP_DATARETRY indicates that this API call shall copy previously copied data in order to recover from an error. In this case TxTpDataCnt specifies the offset in bytes from the current data copy position.
Parameters (inout):	None	
Parameters (out):	availableDataPtrPduR_<User:Lo Tp>CopyTxData.availableDataPtr	Indicates the remaining number of bytes that are available in the upper layer module's Tx buffer. availableDataPtr can be used by TP modules that support dynamic payload lengths (e.g. FrIsoTp) to determine the size of the following CFs.
Return value:	BufReq_ReturnType	BUFREQ_OK: Data has been copied to the transmit buffer completely as requested. BUFREQ_E_BUSY: Request could not be fulfilled, because the required amount of Tx data is not available. The lower layer module may retry this call later on. No data has been copied. BUFREQ_E_NOT_OK: Data has not been copied. Request failed.
Description:	This function is called to acquire the transmit data of an I-PDU segment (N-PDU). Each call to this function provides the next part of the I-PDU data unless retry->Tp DataState is TP_DATARETRY. In this case the function restarts to copy the data beginning at the offset from the current position indicated by retry->TxTpDataCnt. The size of the remaining data is written to the position indicated by availableDataPtr.	

Document ID 695: ChangeDocumentation

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #68035: [diverse] Introduce rules defining which input parameters shall be passed per value and which ones per const reference

Problem description:

SWS_BSW_00186 especially states that input pointer parameters shall use the const qualifier (i.e., shall be P2CONST).

In addition to that there shall be a SWS item that states that input parameters of integral and enum type shall be passed by value whereas input parameters of structure type shall be passed by reference.

The various transformer SWS documents shall be adapted accordingly.
–Last change on issue 68035 comment 4–

Agreed solution:

BSW UML model
————-

The attachment "Changed Proposal in WP-A meeting" contains a list of changes to the APIs in the model (see column H). Afterwards all related documents (included in impact list) shall update their
generated artifacts.

General Requirements on Basic Software Modules
~~~~~~

Introduce the following requirements prior to SRS_BSW_00371:

SRS_BSW_xxxxx: Input parameters of scalar and enum types shall be passed as a value.
Type: valid
Description: All input parameters of scalar or enum type shall be passed as a value.
Rationale:

Use case: For example a function named <Mip>_SomeFunction with a return type of Std_ReturnType and a single parameter named SomeParameter of type uint8 is defined with the following signature:

Std_ReturnType <Mip>_SomeFunction(uint8 SomeParameter);
Dependencies: –

Supporting Material: —

SRS_BSW_yyyyy: Input parameters of structure type shall be passed as a reference to a constant structure
Type: valid
Description: All input parameters of structure type shall be passed as a reference constant structure
Rationale: Passing input parameters of structure type by value would result in additional run-time overhead due to efforts for copying the whole structure.
Use case: For example a function named <Mip>_SomeFunction with a return type of Std_ReturnType and a single parameter named SomeParameter of type SomeStructure (where SomeStructure is a struct) is defined with the following signature:

Std_ReturnType <Mip>_SomeFunction(P2CONST(SomeStructure, AUTOMATIC, <MIP>_APPL_DATA) SomeParameter);
Dependencies: –
Supporting Material: —

SRS_BSW_zzzzz: Input parameters of array type shall be passed as a reference to the constant array base type
Type: valid
Description: All input parameters of array type shall be passed as a reference to the constant array base type
Rationale: This effectively matches the behavior specified in the ISO-C:90 namely that a "declaration of a parameter as 'array of type' shall be adjusted to 'qualified pointer to
type'".
Use case: For example a function named <Mip>_SomeFunction with a return type of Std_ReturnType and a single parameter named SomeParameter of type array of uint8 is defined with the following signature:

Std_ReturnType <Mip>_SomeFunction(P2CONST(uint8, AUTOMATIC, <MIP>_APPL_DATA) SomeParameter);
Dependencies: –
Supporting Material: —

General Specification of Transformers
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

In SWS_Xfrm_00036 change

const <type>* dataElement

to

<paramtype> dataElement

and add the following to the where clause after the API table after the bullet
"type is data type of the data element
"

<paramtype> is derived from <type> according to the parameter passing rules
rules defined by the SRS BSW General (see SRS_BSW_xxxxx, SRS_BSW_yyyyy,
and SRS_BSW_zzzzz) and SWS BSW General (see SWS_BSW_00186 and
SWS_BSW_00187).

In SWS_Xfrm_00038 change

[<type> data_1,] ...
[<type> data_n]

to

[<paramtype> data_1,] ...
[<paramtype> data_n]

and add the following to the where clause after the API table after the bullet
"type is data type of the data element
"

<paramtype> is derived from <type> according to the parameter passing rules
rules defined by the SRS BSW General (see SRS_BSW_xxxxx, SRS_BSW_yyyyy,
and SRS_BSW_zzzzz) and SWS BSW General (see SWS_BSW_00186 and
SWS_BSW_00187).

The following paragraph shall then be removed:

For the arguments of ClientServerOperation which are handed over to the
transformer as data_1, ..., data_n the requirements to API parameters stated in
chapter API Parameters of [5, SWS RTE] are valid (especially [SWS_Rte_01017],

[SWS_Rte_01018] and [SWS_Rte_05107]).

In SWS_Xfrm_00040 change

[<originalData1>, ...
<originalDataN>]

to

[<paramtype> originalData1,] ...
[<paramtype> originalDataN]

and add the following to the where clause after the API table after the bullet "type is data type of the data element
"

<paramtype> is derived from <type> according to the parameter passing rules rules defined by the SRS BSW General (see SRS_BSW_xxxxx, SRS_BSW_yyyyy, and SRS_BSW_zzzzz) and SWS BSW General (see SWS_BSW_00186 and SWS_BSW_00187).

In SWS_Xfrm_00044 change

<type> *data_1, ...
<type> *data_n

to

[<paramtype> data_1,] ...
[<paramtype> data_n]

and add the following to the where clause after the API table after the bullet "type is data type of the data element
"

<paramtype> is derived from <type> according to the parameter passing rules rules defined by the SRS BSW General (see SRS_BSW_xxxxx, SRS_BSW_yyyyy, and SRS_BSW_zzzzz) and SWS BSW General (see SWS_BSW_00186 and SWS_BSW_00187).

The following paragraph shall then be removed:

For the arguments of ClientServerOperation which are handed over to the transformer as data_1, ..., data_n the requirements to API parameters stated in chapter API Parameters of [5, SWS RTE] are valid (especially [SWS_Rte_01017], [SWS_Rte_01018] and [SWS_Rte_05107]).

Speci?cation of SOME/IP Transformer
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

In SWS_SomeIpXf_00138 change

const <type>* dataElement

to

<paramtype> dataElement

and add the following to the where clause after the API table after the bullet
"type is data type of the data element
"

<paramtype> is derived from <type> according to the parameter passing rules rules defined by the SRS BSW General (see SRS_BSW_xxxxx, SRS_BSW_yyyyy, and SRS_BSW_zzzzz) and SWS BSW General (see SWS_BSW_00186 and SWS_BSW_00187).

In SWS_SomeIpXf_00141 change

[<type> data_1,] ...
[<type> data_n]

to

[<paramtype> data_1,] ...
[<paramtype> data_n]

and add the following to the where clause after the API table after the bullet
"type is data type of the data element
"

<paramtype> is derived from <type> according to the parameter passing rules rules defined by the SRS BSW General (see SRS_BSW_xxxxx, SRS_BSW_yyyyy, and SRS_BSW_zzzzz) and SWS BSW General (see SWS_BSW_00186 and SWS_BSW_00187).

The following paragraph shall then be removed:

For the arguments of ClientServerOperation which are handed over to the transformer as data_1, ..., data_n the requirements to API parameters stated in chapter API Parameters of [5, SWS RTE] are valid (especially [SWS_Rte_01017], [SWS_Rte_01018] and [SWS_Rte_05107]).

In SWS_SomeIpXf_00145 change

<type> *data_1, ...
<type> *data_n

to

[<paramtype> data_1,] ...
[<paramtype> data_n]

and add the following to the where clause after the API table after the bullet "type is data type of the data element
"

<paramtype> is derived from <type> according to the parameter passing rules rules defined by the SRS BSW General (see SRS_BSW_xxxxx, SRS_BSW_yyyyy, and SRS_BSW_zzzzz) and SWS BSW General (see SWS_BSW_00186 and SWS_BSW_00187).

The following paragraph shall then be removed:

For the arguments of ClientServerOperation which are handed over to the transformer as data_1, ..., data_n the requirements to API parameters stated in chapter API Parameters of [5, SWS RTE] are valid (especially [SWS_Rte_01017], [SWS_Rte_01018] and [SWS_Rte_05107]).

Specification of COM Based Transformer

~~~~~~

In SWS_ComXf_00007 change

const <type>* dataElement

to

<paramtype> dataElement

and add the following to the where clause after the API table after the bullet
"type is data type of the data element
"

<paramtype> is derived from <type> according to the parameter passing rules
rules defined by the SRS BSW General (see SRS_BSW_xxxxx, SRS_BSW_yyyyy,
and SRS_BSW_zzzzz) and SWS BSW General (see SWS_BSW_00186 and
SWS_BSW_00187).

Specification of Time Sync over Ethernet
~~~~~~

In SWS_EthTSyn_00040 make the parameter DataPtr of EthTSyn_RxIndication
const.

Specification of SWS FlexRay Interface
~~~~~~

Change SWS_FrIf_05073 from
FrIf_NumOfStartupFramesPtr (IN)
to
FrIf_NumOfStartupFramesPtr (OUT)

Specification of ADC
~~~~~~
~[SWS_Adc_00419] Adc_SetupResultBuffer: change Adc_ValueGroupType* to
const Adc_ValueGroupType*
~[SWS_Adc_00369] Adc_ReadGroup: move Adc_ValueGroupType * from Parame-

ters (in) to Parameters (out)

There is no need to change parameter from IN to INOUT in Adc_SetupResultBuffer

Specification of Com
~~~~~~
Change type of parameter MetaData of Com_TriggerIPDUSendWithMetaData from uint8* to const uint8*

Specification of ComM
~~~~~~
no change required

Specification of Dem
~~~~~~
no change required

Specification of DLT
~~~~~~
no change required

Specification of DoIP
~~~~~~
From:
Std_ReturnType <User>_DoIPRoutingActivationConfirmation(boolean* Confirmed, uint8* ConfirmationReqData, uint8* ConfirmationResData)
Std_ReturnType <User>_DoIPRoutingActivationAuthentication(boolean* Authentified, uint8* AuthenticationReqData, uint8* AuthenticationResData)

To:
Std_ReturnType <User>_DoIPRoutingActivationConfirmation(boolean* Confirmed, const uint8* ConfirmationReqData, uint8* ConfirmationResData)
Std_ReturnType <User>_DoIPRoutingActivationAuthentication(boolean* Authentified, const uint8* AuthenticationReqData, uint8* AuthenticationResData)

Specification of E2ELibrary
~~~~~~

no change required

Specification of Eth
~~~~~~
no change required

Specification of EthIf
~~~~~~
no change required

Specification of EthSwitchDriver
~~~~~~
no change required

Specification of ICUDriver
~~~~~~
SWS_Icu_00201: Icu_StartTimestamp
Parameter (IN): Icu_ValueType* BufferPtr shall be changed to Parameters (out) type

Specification of LdCom
~~~~~~
[SWS_LDCOM_00027]: LdCom_CopyTxData
BufReq_ReturnType LdCom_CopyTxData(PduIdType id, const PduInfoType* info, RetryInfoType* retry, PduLengthType* availableDataPtr) shall be changed to
BufReq_ReturnType LdCom_CopyTxData(PduIdType id, const PduInfoType* info, const RetryInfoType* retry, PduLengthType* availableDataPtr)

[SWS_LDCOM_00036]: Rte_LdComCbkCopyTxData_<sn>
BufReq_ReturnType Rte_LdComCbkCopyTxData_<sn>(const PduInfoType* info, RetryInfoType* retry, PduLengthType* availableDataPtr) shall be changed to
BufReq_ReturnType Rte_LdComCbkCopyTxData_<sn>(const PduInfoType* info, const RetryInfoType* retry, PduLengthType* availableDataPtr)

Specification of Lin
~~~~~~
PduInfoPtr needs to be const in Std_ReturnType Lin_SendFrame(uint8 Channel, const Lin_PduType* PduInfoPtr)

Specification of PduR
~~~~~~
* PduR_<User:LoTp>CopyTxData
add const to "RetryInfoType* retry"

Specification of J1939Nm
~~~~~~
Change parameter 'name' of User_AddressClaimedIndication to type 'const uint8*'

Specification of SoAd
~~~~~~
=> everything already fixed with RfC 65633

Specification of SPIHandlerDriver
~~~~~~
==> nothing to change for SWS SPI

Specification of SynchronizedTimeBaseManager
~~~~~~
"StbM not affected. All issues listed in the WP-A attachment have been already
implemented by IT 69124 in context of RfC 65633"

Specification of TcpIp
~~~~~~
~[SWS_TCPIP_00040] TcpIp_DhcpReadOption: change DataPtr from (IN) to
(OUT)
~[SWS_TCPIP_00189] TcpIp_DhcpV6ReadOption: change DataPtr from (IN) to
(OUT)
=> everything else already fixed with RfC 65633

Specification of TimeSyncOverFlexRay
~~~~~~
"Change SWS_FrTSyn_00064: parameter versioninfo of type Std_VersionInfoType*
is marked wrongly as IN. Change to OUT"

Specification of EFX
~~~~~~~~~~~~~~~~~~~~~~
~ [SWS_Efx_00355] Efx_Debounce_u8_u8: Include constant for pointer Input-parameter as like below.
uint8 Efx_Debounce_u8_u8(boolean X, Efx_DebounceState_Type * State, const Efx_DebounceParam_Type * Param, sint32 dT)

~ [SWS_Efx_00376] Efx_MedianSort: The parameter <InType>* Array should be InOut instead of In parameter as like below.
Parameters (in): N Size of an array
Parameters (inout): Array Pointer to an array

~ [SWS_Efx_00309] Efx_RampCheckActivity: Include constant for pointer Input-parameter as like below.
boolean Efx_RampCheckActivity(const Efx_StateRamp_Type* State_cpst)

~ [SWS_Efx_00307] Efx_RampGetSwitchPos: Include constant for pointer Input-parameter as like below.
boolean Efx_RampGetSwitchPos(const Efx_StateRamp_Type* State_cpst)

~ [SWS_Efx_00193] Efx_Array_Average: Include constant for pointer Input-parameter as like below.
<OutType> Efx_Array_Average_<InTypeMn>_<OutTypeMn>(const <InType>* Array, uint16 Count)

Specification of MFL
~~~~~~~~~~~~~~~~~~~~~~
~ [SWS_Mfl_00192] Mfl_Debounce_u8_u8: Include constant for pointer Input-parameter as like below.
boolean Mfl_Debounce_u8_u8(boolean X, Mfl_DebounceState_Type* State, const Mfl_DebounceParam_Type* Param, float32 dT)

~ [SWS_Mfl_00266] Mfl_DebounceInit: The parameter Mfl_DebounceState_Type* State should be Out instead of In parameter as like below.
Parameters (in): X Initial value for the input state
Parameters (out): State Pointer to structure for debouncing state variables

~ [SWS_Mfl_00246] Mfl_HystDeltaRight_f32_u8: Include constant for pointer Input-parameter as like below.
boolean Mfl_HystDeltaRight_f32_u8(float32 X, float32 Delta, float32 Rsp, const uint8* State)

~ [SWS_Mfl_00285] Mfl_MedianSort_f32_f32: The parameter Array should be InOut instead of In parameter as like below.
Parameters (in): N Size of an array
Parameters (inout): Array Pointer to an array

~ [SWS_Mfl_00037] Mfl_PT1SetState: The parameter State_cpst should be Out instead of In parameter as like below.
Parameters (in): X1_f32 Initial value for input state
Y1_f32 Initial value for output state
Parameters (out): State_cpst Pointer to internal state structure

~ [SWS_Mfl_00225] Mfl_RampCheckActivity: Include constant for pointer Input-parameter as like below.
boolean Mfl_RampCheckActivity(const Mfl_StateRamp_Type* State_cpst)

~ [SWS_Mfl_00223] Mfl_RampGetSwitchPos: Include constant for pointer Input-parameter as like below.
boolean Mfl_RampGetSwitchPos(const Mfl_StateRamp_Type* State_cpst)
–Last change on issue 68035 comment 135–

BW-C-Level:

Application	Specification	Bus
1	4	1

1.12   Specification Item SWS_PduR_00615

Trace References:

SRS_PduR_06120, SRS_BSW_00310

Content:

Service name:	PduR_EnableRouting	
Syntax:	void PduR_EnableRouting(PduR_RoutingPathGroupIdType id)	
Service ID[hex]:	0xf3	
Sync/Async:	Synchronous	
Reentrancy:	Reentrant	
Parameters (in):	id	Identification of the routing path group. Routing path groups are defined in the PDU router configuration.
Parameters (inout):	None	

Parameters (out):	None
Return value:	None
Description:	Enables a routing path tablegroup.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76221: [PduR] Enable/disable routing API description

  Problem description:

  PduR_EnableRouting
  Current description "Enables a routing path table." shall be "Enables a routing path group."

  PduR_DisableRouting
  Current description "Disables a routing path table." shall be "Disables a routing path group."

  Agreed solution:

  PduR_EnableRouting
  Current description "Enables a routing path table." shall be "Enables a routing path group."

  PduR_DisableRouting
  Current description "Disables a routing path table." shall be "Disables a routing path group."

  BW-C-Level:

Application	Specification	Bus
1	1	1

1.13 Specification Item SWS_PduR_00617

Trace References:

SRS_PduR_06120, SRS_BSW_00310

Content:

Service name:	PduR_DisableRouting

Syntax:	void PduR_DisableRouting(PduR_RoutingPathGroupIdType id, boolean initialize)	
Service ID[hex]:	0xf4	
Sync/Async:	Synchronous	
Reentrancy:	Reentrant	
Parameters (in):	id	Identification of the routing path group. Routing path groups are defined in the PDU router configuration.
	initialize	true: initialize single buffers to the default value false: retain current value of single buffers
Parameters (inout):	None	
Parameters (out):	None	
Return value:	None	
Description:	Disables a routing path tablegroup.	

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76221: [PduR] Enable/disable routing API description

  Problem description:

  PduR_EnableRouting
  Current description "Enables a routing path table." shall be "Enables a routing path group."

  PduR_DisableRouting
  Current description "Disables a routing path table." shall be "Disables a routing path group."

  Agreed solution:

  PduR_EnableRouting
  Current description "Enables a routing path table." shall be "Enables a routing path group."

  PduR_DisableRouting
  Current description "Disables a routing path table." shall be "Disables a routing path group."

  BW-C-Level:

Application	Specification	Bus
1	1	1

1.14 Specification Item SWS_PduR_00646

Trace References:

SRS_PduR_06120, SRS_PduR_06104, SRS_PduR_06105

Content:

The PDU Router shall immediately disable the routing path groups. Disabling of routing path groups inhibits forwarding of the following APIs:

- <Lo>_Transmit
- <Up>_RxIndication (but NOT <UpTp>_RxIndication)
- <UpTp>_StartofReception
- <UpTp>_CopyRxData
- <UpTp>_CopyTxData

The APIs listed above with Std_ReturnType shall return E_NOT_OK if the affected routing path group is disabled, and APIs with BufReq_ReturnType shall return BUFREQ_E_NOT_OK in the same situation.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76386: APIs listed in the SWS ID SWS_PDUR_00805 are not invoked by the PduR Module

  Problem description:

  _____

  Name: Saneesh P S
  Phone: +91-9447775467
  Role:

  _____

  Description/Motivation:
  As per the SWS ID SWS_PDUR_00805, Disabling of routing path groups in case of gatewaying shall
  disable the following APIs:
  <Lo>_RxIndication (but NOT <LoTp>_RxIndication)
  <LoTp>_StartofReception
  <LoTp>_CopyRxData
  <LoTp>_CopyTxData
  <Lo>_TriggerTransmit.

  But the APIs listed above are not invoked by the PduR Module.

Could you please clarify the issue?

_____

Was there already a decision? No

_____

Agreed solution:

Original req:
[SWS_PDUR_00805] Disabling of routing path groups in case of gatewaying shall disable the following APIs:
<Lo>_RxIndication (but NOT <LoTp>_RxIndication)
<LoTp>_StartofReception
<LoTp>_CopyRxData
<LoTp>_CopyTxData
<Lo>_TriggerTransmit
The APIs listed above with Std_ReturnType shall return E_NOT_OK if the affected routing path group is disabled, and APIs with BufReq_ReturnType shall return BUFREQ_E_NOT_OK in the same situation. ()

change proposal:
[SWS_PDUR_00805]If a routing path group is disabled (by the call PduR_DisableRouting) the PduR shall directly return for following functions for this routing path group:
* PduR_<User:Up>Transmit
* PduR_<User:Lo>RxIndication
* PduR_<User:Lo>TriggerTransmit
* PduR_<User:LoTp>StartOfReception
* PduR_<User:LoTp>CopyRxData
* PduR_<User:LoTp>CopyTxData
If the function has a Std_ReturnType, it shall return E_NOT_OK. If function has a BufReq_ReturnType, it shall return BUFREQ_E_NOT_OK. ()

Note: This does not affect PduR_<User:LoTp>RxIndication

Remove requirement SWS_PduR_00646 (it is now integrated into 805)
–Last change on issue 76386 comment 24–

BW-C-Level:

Application	Specification	Bus
1	4	1

1.15   Specification Item SWS_PduR_00733

Trace References:

SRS_PduR_06026

Content:

When the PduR_<Up>ChangeParameter is called the PDU Router module shall call the <LoTp>_ChangeParameter for the destination transport protocol module of the I-PDU.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76491: [PduR] Remove forwarding of ChangeParameter API from PduR

  Problem description:

  The ChangeParameter API is TP specific and there is no use case to call this API through the PduR (CDD or integration code that requires a change in the TP parameter will call the API directly in the TP module)

  Agreed solution:

  PDUR_SWS
  Remove references from ch. 5 "Dependencies to other modules"
  Remove ch. 8.4. "Change transport protocol parameter"

  BSW_Model
  set API PduR_<User:Up>ChangeParameter [SWS_PduR_00482] to obsolete
  set optional interface from [SWS_PduR_00424] to obsolete

  ECUC model
  set [ECUC_PduR_00326] PduRChangeParameterApi to obsolete
  remove reference in description from [ECUC_PduR_00319] PduRUseTag
  –Last change on issue 76491 comment 12–

  BW-C-Level:

Application	Specification	Bus
1	4	1

1.16   Specification Item SWS_PduR_00734

Trace References:

SRS_PduR_06122, SRS_PduR_06104

Content:

The return value of the <LoTp>_ChangeParameter shall be forwarded to the upper layer module.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76491: [PduR] Remove forwarding of ChangeParameter API from PduR

   Problem description:

   The ChangeParameter API is TP specific and there is no use case to call this API through the PduR (CDD or integration code that requires a change in the TP parameter will call the API directly in the TP module)

   Agreed solution:

   PDUR_SWS
   Remove references from ch. 5 "Dependencies to other modules"
   Remove ch. 8.4. "Change transport protocol parameter"

   BSW_Model
   set API PduR_<User:Up>ChangeParameter [SWS_PduR_00482] to obsolete
   set optional interface from [SWS_PduR_00424] to obsolete

   ECUC model
   set [ECUC_PduR_00326] PduRChangeParameterApi to obsolete
   remove reference in description from [ECUC_PduR_00319] PduRUseTag
   –Last change on issue 76491 comment 12–

   BW-C-Level:

Application	Specification	Bus
1	4	1

1.17   Specification Item SWS_PduR_00747

Trace References:

SRS_PduR_06026

Content:

The I-PDU id provided in the call is Rx I-PDU ID and therefore the PDU Router module shall be able to identify this I-PDU correctly.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76491: [PduR] Remove forwarding of ChangeParameter API from PduR

  Problem description:

  The ChangeParameter API is TP specific and there is no use case to call this API through the PduR (CDD or integration code that requires a change in the TP parameter will call the API directly in the TP module)

  Agreed solution:

  PDUR_SWS
  Remove references from ch. 5 "Dependencies to other modules"
  Remove ch. 8.4. "Change transport protocol parameter"

  BSW_Model
  set API PduR_<User:Up>ChangeParameter [SWS_PduR_00482] to obsolete
  set optional interface from [SWS_PduR_00424] to obsolete

  ECUC model
  set [ECUC_PduR_00326] PduRChangeParameterApi to obsolete
  remove reference in description from [ECUC_PduR_00319] PduRUseTag
  –Last change on issue 76491 comment 12–

  BW-C-Level:

Application	Specification	Bus
1	4	1

1.18 Specification Item SWS_PduR_00802

Trace References:

SRS_BSW_00350

Content:

The PduR implementation shall include Det.h if PduRDevErrorDetect is enabled.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #59085: Rollout of 'Runtime errors'

Problem description:

Inconsistencies in SWS with semantics of Default errors
–Last change on issue 59085 comment 26–

Agreed solution:

solution in Column "G" of the new attachment https://www.autosar.org/bugzilla/attachment.cgi?id=4604

Notes:
- It is not enough just to migrate the error from one classification table to another. Please also check the related requirements (and background information) which is referring to that error and adapt them if needed.
- The review task of the ITs shall be done by the WP to which the specification "belongs".

*** BSW UML Model ***
SWS_CanNm:
—————-
Chapter 8.6.1 Optional Interfaces:
Add within SWS_CanNm_00325 the API function Det_ReportRunTimeError

SWS_LinIf:
—————-
SWS_LinIf_00359: add Det_ReportRuntimeError

SWS_UdpNm:
—————-
Replace UDPNM_E_NO_INIT with UDPNM_E_UNINIT in description of API UdpNm_MainFunction_<Instance Id> (SWS_UdpNm_00234)

*** ECUC XML ***
Not affected. No configuration of runtime error reporting required (see SWS BSW General).
–Last change on issue 59085 comment 88–

BW-C-Level:

Application	Specification	Bus
1	4	1

1.19 Specification Item SWS_PDUR_00805

Trace References:

Content:

Disabling of routing path groups in case of gatewaying shall disable the following APIsIf a routing path group is disabled (by the call PduR_DisableRouting) the PduR shall directly return for following functions for this routing path group:

<Lo>_RxIndication (but NOT <LoTp>_

- PduR_<User:Up>Transmit

- PduR_<User:Lo>RxIndication)

  <LoTp>_Startof

- PduR_<User:Lo>TriggerTransmit

- PduR_<User:LoTp>StartOfReception

  <LoTp>_

- PduR_<User:LoTp>CopyRxData

  <LoTp>_

- PduR_<User:LoTp>CopyTxData

  <Lo>_TriggerTransmit

The APIs listed above with If the function has a Std_ReturnType, it shall return E_NOT_OKif the affected routing path group is disabled, and APIs with . If function has a BufReq_ReturnType, it shall return BUFREQ_E_NOT_OKin the same situation.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76386: APIs listed in the SWS ID SWS_PDUR_00805 are not invoked by the PduR Module

  Problem description:

  _____

  Name: Saneesh P S
  Phone: +91-9447775467
  Role:

  _____

  Description/Motivation:

Document ID 695: ChangeDocumentation

As per the SWS ID SWS_PDUR_00805, Disabling of routing path groups in case of gatewaying shall
disable the following APIs:
<Lo>_RxIndication (but NOT <LoTp>_RxIndication)
<LoTp>_StartofReception
<LoTp>_CopyRxData
<LoTp>_CopyTxData
<Lo>_TriggerTransmit.

But the APIs listed above are not invoked by the PduR Module.

Could you please clarify the issue?

_____

Was there already a decision? No

_____

Agreed solution:

Original req:
[SWS_PDUR_00805] Disabling of routing path groups in case of gatewaying shall disable the following APIs:
<Lo>_RxIndication (but NOT <LoTp>_RxIndication)
<LoTp>_StartofReception
<LoTp>_CopyRxData
<LoTp>_CopyTxData
<Lo>_TriggerTransmit
The APIs listed above with Std_ReturnType shall return E_NOT_OK if the affected routing path group is disabled, and APIs with BufReq_ReturnType shall return BUFREQ_E_NOT_OK in the same situation. ()

change proposal:
[SWS_PDUR_00805]If a routing path group is disabled (by the call PduR_DisableRouting) the PduR shall directly return for following functions for this routing path group:
* PduR_<User:Up>Transmit
* PduR_<User:Lo>RxIndication
* PduR_<User:Lo>TriggerTransmit
* PduR_<User:LoTp>StartOfReception
* PduR_<User:LoTp>CopyRxData
* PduR_<User:LoTp>CopyTxData
If the function has a Std_ReturnType, it shall return E_NOT_OK. If function has a

BufReq_ReturnType, it shall return BUFREQ_E_NOT_OK. ()

Note: This does not affect PduR_<User:LoTp>RxIndication

Remove requirement SWS_PduR_00646 (it is now integrated into 805)
–Last change on issue 76386 comment 24–

BW-C-Level:

Application	Specification	Bus
1	4	1

1.20   Specification Item SWS_PDUR_00816

Trace References:

Content:

Runtime Error Types

Type of error	Related error code	Value [hex]
Loss of a PDU instance (buffer overrun in gateway operation)	PDUR_E_PDU_INSTANCES_LOST	0x0a
API service used without module initialization or PduR_Init called in any state other than PDUR_UNINIT	PDUR_E_INVALID_REQUEST	0x01
Invalid PDU identifier	PDUR_E_PDU_ID_INVALID	0x02
TP module rejects a transmit request for a valid PDU identifier	PDUR_E_TP_TX_REQ_REJECTED	0x03
If the routing table is invalid that is given to the PduR_EnableRouting or PduR_DisableRouting functions Loss of a PDU instance (buffer overrun in gateway operation)	PDUR_E_ROUTING_PATH_GROUP_ID_INVALID PDU_INSTANCES_LOST	0x08 0x0a

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #59085: Rollout of 'Runtime errors'

  Problem description:

  Inconsistencies in SWS with semantics of Default errors
  –Last change on issue 59085 comment 26–

Agreed solution:

solution in Column "G" of the new attachment
https://www.autosar.org/bugzilla/attachment.cgi?id=4604

Notes:
- It is not enough just to migrate the error from one classification table to another.
Please also check the related requirements (and background information) which is
referring to that error and adapt them if needed.
- The review task of the ITs shall be done by the WP to which the specification
"belongs".

*** BSW UML Model ***
SWS_CanNm:
————-
Chapter 8.6.1 Optional Interfaces:
Add within SWS_CanNm_00325 the API function Det_ReportRunTimeError

SWS_LinIf:
————-
SWS_LinIf_00359: add Det_ReportRuntimeError

SWS_UdpNm:
————-
Replace UDPNM_E_NO_INIT with UDPNM_E_UNINIT in description of API
UdpNm_MainFunction_<Instance Id> (SWS_UdpNm_00234)

*** ECUC XML ***
Not affected. No configuration of runtime error reporting required (see SWS BSW
General).
–Last change on issue 59085 comment 88–

BW-C-Level:

Application	Specification	Bus
1	4	1

1.21   Specification Item SWS_PduR_91001

Trace References:

SRS_BSW_00384

Content:

API function	Description
Det_ReportRuntimeError	Service to report runtime errors. If a callout has been configured then this callout shall be called.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #59085: Rollout of 'Runtime errors'

  Problem description:

  Inconsistencies in SWS with semantics of Default errors
  –Last change on issue 59085 comment 26–

  Agreed solution:

  solution in Column "G" of the new attachment
  https://www.autosar.org/bugzilla/attachment.cgi?id=4604

  Notes:
  - It is not enough just to migrate the error from one classification table to another. Please also check the related requirements (and background information) which is referring to that error and adapt them if needed.
  - The review task of the ITs shall be done by the WP to which the specification "belongs".

  *** BSW UML Model ***
  SWS_CanNm:
  ——–-
  Chapter 8.6.1 Optional Interfaces:
  Add within SWS_CanNm_00325 the API function Det_ReportRunTimeError

  SWS_LinIf:
  ——–-
  SWS_LinIf_00359: add Det_ReportRuntimeError

  SWS_UdpNm:
  ——–-
  Replace UDPNM_E_NO_INIT with UDPNM_E_UNINIT in description of API UdpNm_MainFunction_<Instance Id> (SWS_UdpNm_00234)

Document ID 695: ChangeDocumentation

*** ECUC XML ***
Not affected. No configuration of runtime error reporting required (see SWS BSW General).
–Last change on issue 59085 comment 88–

BW-C-Level:

Application	Specification	Bus
1	4	1