

Document Title	SWS_ECUStateManager: 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_ECUStateManager	3
1.1	Specification Item ECUC_EcuM_00103	3
1.2	Specification Item ECUC_EcuM_00109	5
1.3	Specification Item ECUC_EcuM_00112	7
1.4	Specification Item ECUC_EcuM_00113	9
1.5	Specification Item ECUC_EcuM_00125	11
1.6	Specification Item ECUC_EcuM_00126	14
1.7	Specification Item ECUC_EcuM_00127	16
1.8	Specification Item ECUC_EcuM_00129	18
1.9	Specification Item ECUC_EcuM_00135	20
1.10	Specification Item ECUC_EcuM_00141	22
1.11	Specification Item ECUC_EcuM_00143	24
1.12	Specification Item ECUC_EcuM_00144	26
1.13	Specification Item ECUC_EcuM_00145	28
1.14	Specification Item ECUC_EcuM_00147	30
1.15	Specification Item ECUC_EcuM_00165	32
1.16	Specification Item ECUC_EcuM_00166	35
1.17	Specification Item ECUC_EcuM_00189	37
1.18	Specification Item ECUC_EcuM_00191	39
1.19	Specification Item ECUC_EcuM_00202	42
1.20	Specification Item ECUC_EcuM_00225	44
1.21	Specification Item SWS_EcuM_02337	46
1.22	Specification Item SWS_EcuM_02563	49
1.23	Specification Item SWS_EcuM_02788	50
1.24	Specification Item SWS_EcuM_02810	53
1.25	Specification Item SWS_EcuM_02859	57
1.26	Specification Item SWS_EcuM_02907	61
1.27	Specification Item SWS_EcuM_02923	62
1.28	Specification Item SWS_EcuM_03009	63
1.29	Specification Item SWS_EcuM_04094	67
1.30	Specification Item SWS_EcuM_04095	68

1 SWS_ECUStateManager

1.1 Specification Item ECUC_EcuM_00103

Trace References:

none

Content:

Container Name	EcuMConfigurationEcuMConfiguration
Description	This container contains the configuration (parameters) of the ECU State Manager.
Configuration Parameters	

Included parameters:

No Included Parameters

Included containers:

Included Containers		
Container Name	Multiplicity	Scope / Dependency
EcuMCommonConfiguration	1	This container contains the common configuration (parameters) of the ECU State Manager.
EcuMFixedConfiguration	0..1	This container contains the configuration (parameters) of the EcuMFixed. Only applicable if EcuMFixed is implemented. Tags: atp.Status=obsolete
EcuMFlexConfiguration	0..1	This container contains the configuration (parameters) of the EcuMFlex. Only applicable if EcuMFlex is implemented.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77051: [EcuMfixed] Set SWS_EcuStateManagerFixed to obsolete

Problem description:

Description/Motivation:

The need for having 2 Ecu Statemanager modules has vanished.

The specification of EcuMfix can be removed from the standard.

Was there already a decision?

No, but afaik Daimler is the last remaining stakeholder.

Agreed solution:

Set EcuMfix to obsolete to be able to remove it for the next minor release.

==TR ClassicPlatformReleaseOverview==

==Release Overview==

add Release Note to chapter 3.1.2.3 Obsolete Specifications:

"The following specification is set to status obsolete in this release:

Specification of ECU State Manager with fixed state machine (UID 444, SWS)"

==EcuMfixed==

+add sentence to 1 Scope of Document

"This specification is obsolete and will be removed from the standard in an upcoming release."

==ECUC XML Model==

< _____

< set atp.Status of EcuMFixedGeneral container and its content to obsolete (lower-Multiplicity = 0):

< The following parameters are affected:

< - EcuMTTIIEnabled

< - EcuMTTIIWakeupSourceRef

< - EcuMIncludeComM

<

< set atp.Status of EcuMFixedConfiguration container and its content to obsolete (lowerMultiplicity = 0):

< The following parameters are affected:

< - EcuMRunMinimumDuration

< - EcuMNvramReadallTimeout

< - EcuMNvramWriteallTimeout

< - EcuMNormalMcuModeRef

< - EcuMComMCommunicationAllowedList

< - EcuMFixedUser

< - EcuMDivisor

< - EcuMSleepModeRef
< - EcuMSuccessorRef
<
< The following subcontainers are affected:
< - EcuMDriverInitListTwo
< - EcuMDriverInitListThree
< - EcuMTTII
< - EcuMFixedUserConfig
–Last change on issue 77051 comment 36–

BW-C-Level:

Application	Specification	Bus
3	5	1

1.2 Specification Item ECUC_EcuM_00109

Trace References:

none

Content:

Name	EcuMDivisorEcuMTTII.EcuMDivisor		
Parent Container	EcuMTTII		
Description	This parameter defines the divisor preload value. Tags: atp.Status=obsolete		
Multiplicity	1 0..1		
Type	EcucIntegerParamDef		
Range	0 .. 18446744073709551615		
Default value	–		
Post-Build Variant Value	false		
Value Configuration Class	Pre-compile time	X	VARIANT-POST-BUILD
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: local		

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77051: [EcuMfixed] Set SWS_EcuStateManagerFixed to obsolete

Problem description:

Description/Motivation:

The need for having 2 Ecu Statemanager modules has vanished.
The specification of EcuMfix can be removed from the standard.

Was there already a decision?

No, but afaik Daimler is the last remaining stakeholder.

Agreed solution:

Set EcuMfix to obsolete to be able to remove it for the next minor release.

==TR ClassicPlatformReleaseOverview==

==Release Overview==

add Release Note to chapter 3.1.2.3 Obsolete Specifications:

"The following specification is set to status obsolete in this release:

Specification of ECU State Manager with fixed state machine (UID 444, SWS)"

==EcuMfixed==

+add sentence to 1 Scope of Document

"This specification is obsolete and will be removed from the standard in an upcoming release."

==ECUC XML Model==

<

< set atp.Status of EcuMFixedGeneral container and its content to obsolete (lower-Multiplicity = 0):

< The following parameters are affected:

< - EcuMTTIIEnabled

< - EcuMTTIIWakeupSourceRef

< - EcuMIncludeComM

<

< set atp.Status of EcuMFixedConfiguration container and its content to obsolete (lowerMultiplicity = 0):

< The following parameters are affected:

< - EcuMRunMinimumDuration

< - EcuMNvramReadallTimeout

< - EcuMNvramWriteallTimeout

- < - EcuMNormalMcuModeRef
- < - EcuMComMCommunicationAllowedList
- < - EcuMFixedUser
- < - EcuMDivisor
- < - EcuMSleepModeRef
- < - EcuMSuccessorRef
- <
- < The following subcontainers are affected:
- < - EcuMDriverInitListTwo
- < - EcuMDriverInitListThree
- < - EcuMTTII
- < - EcuMFixedUserConfig
- Last change on issue 77051 comment 36–

BW-C-Level:

Application	Specification	Bus
3	5	1

1.3 Specification Item ECUC_EcuM_00112

Trace References:

none

Content:

Container Name	EcuMDriverInitListThreeEcuMDriverInitListThree
Description	<p>Container for Init Block III.</p> <p>This container holds a list of modules to be initialized. Each module in the list will be called for initialization in the list order.</p> <p>All modules in this list are initialized after the OS is started and so these modules may use OS support. These modules may also rely on the Nvram ReadAll job to have provided all data.</p> <p>Tags: atp.Status=obsolete</p>
Configuration Parameters	

Included parameters:

No Included Parameters

Included containers:

Included Containers		
Container Name	Multiplicity	Scope / Dependency

Included Containers		
Container Name	Multiplicity	Scope / Dependency
EcuMDriverInitItem	1..*	These containers describe the entries in a driver init list.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77051: [EcuMfixed] Set SWS_EcuStateManagerFixed to obsolete

Problem description:

Description/Motivation:

The need for having 2 Ecu Statemanager modules has vanished.
The specification of EcuMfix can be removed from the standard.

Was there already a decision?

No, but afaik Daimler is the last remaining stakeholder.

Agreed solution:

Set EcuMfix to obsolete to be able to remove it for the next minor release.

==TR ClassicPlatformReleaseOverview==

==Release Overview==

add Release Note to chapter 3.1.2.3 Obsolete Specifications:

"The following specification is set to status obsolete in this release:

Specification of ECU State Manager with fixed state machine (UID 444, SWS)"

==EcuMfixed==

+add sentence to 1 Scope of Document

"This specification is obsolete and will be removed from the standard in an upcoming release."

==ECUC XML Model==

< _____

< set atp.Status of EcuMFixedGeneral container and its content to obsolete (lower-Multiplicity = 0):

< The following parameters are affected:

< - EcuMTTIIEnabled
< - EcuMTTIIWakeupSourceRef
< - EcuMIncludeComM
<
< set atp.Status of EcuMFixedConfiguration container and its content to obsolete (lowerMultiplicity = 0):
< The following parameters are affected:
< - EcuMRunMinimumDuration
< - EcuMNvramReadallTimeout
< - EcuMNvramWriteallTimeout
< - EcuMNormalMcuModeRef
< - EcuMComMCommunicationAllowedList
< - EcuMFixedUser
< - EcuMDivisor
< - EcuMSleepModeRef
< - EcuMSuccessorRef
<
< The following subcontainers are affected:
< - EcuMDriverInitListTwo
< - EcuMDriverInitListThree
< - EcuMTTII
< - EcuMFixedUserConfig
–Last change on issue 77051 comment 36–

BW-C-Level:

Application	Specification	Bus
3	5	1

1.4 Specification Item ECUC_EcuM_00113

Trace References:

none

Content:

Container Name	EcuMDriverInitListTwoEcuMDriverInitListTwo
Description	<p>Container for Init Block II.</p> <p>This container holds a list of modules to be initialized. Each module in the list will be called for initialization in the list order.</p> <p>All modules in this list are initialized after the OS is started and so these modules may use OS support. These modules may not rely on the Nvram ReadAll job to have provided all data.</p> <p>Tags: atp.Status=obsolete</p>

Configuration Parameters

Included parameters:

No Included Parameters

Included containers:

Included Containers		
Container Name	Multiplicity	Scope / Dependency
EcuMDriverInitItem	1..*	These containers describe the entries in a driver init list.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77051: [EcuMfixed] Set SWS_EcuStateManagerFixed to obsolete

Problem description:

Description/Motivation:

The need for having 2 Ecu Statemanager modules has vanished.
The specification of EcuMfix can be removed from the standard.

Was there already a decision?

No, but afaik Daimler is the last remaining stakeholder.

Agreed solution:

Set EcuMfix to obsolete to be able to remove it for the next minor release.

==TR ClassicPlatformReleaseOverview==

==Release Overview==

add Release Note to chapter 3.1.2.3 Obsolete Specifications:

"The following specification is set to status obsolete in this release:

Specification of ECU State Manager with fixed state machine (UID 444, SWS)"

==EcuMfixed==

+add sentence to 1 Scope of Document

"This specification is obsolete and will be removed from the standard in an upcoming release."

==ECUC XML Model==

< _____
< set atp.Status of EcuMFixedGeneral container and its content to obsolete (lower-Multiplicity = 0):
< The following parameters are affected:
< - EcuMTTIIEnabled
< - EcuMTTIIWakeupSourceRef
< - EcuMIncludeComM
<
< set atp.Status of EcuMFixedConfiguration container and its content to obsolete (lowerMultiplicity = 0):
< The following parameters are affected:
< - EcuMRunMinimumDuration
< - EcuMNvramReadallTimeout
< - EcuMNvramWriteallTimeout
< - EcuMNormalMcuModeRef
< - EcuMComMCommunicationAllowedList
< - EcuMFixedUser
< - EcuMDivisor
< - EcuMSleepModeRef
< - EcuMSuccessorRef
<
< The following subcontainers are affected:
< - EcuMDriverInitListTwo
< - EcuMDriverInitListThree
< - EcuMTTII
< - EcuMFixedUserConfig
–Last change on issue 77051 comment 36–

BW-C-Level:

Application	Specification	Bus
3	5	1

1.5 Specification Item ECUC_EcuM_00125

Trace References:

none

Content:

Name	EcuMNormalMcuModeRefEcuMFixedConfiguration.EcuMNormalMcuModeRef		
Parent Container	EcuMFixedConfiguration		
Description	This parameter is a reference to the normal MCU mode to be restored after a sleep. Tags: atp.Status=obsolete		
Multiplicity	1 0..1		
Type	Symbolic name reference to [McuModeSettingConf]		
Post-Build Variant Value	false		
Value Configuration Class	Pre-compile time	X	VARIANT-POST-BUILD
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: local		

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77051: [EcuMfixed] Set SWS_EcuStateManagerFixed to obsolete

Problem description:

Description/Motivation:

The need for having 2 Ecu Statemanager modules has vanished.
The specification of EcuMfix can be removed from the standard.

Was there already a decision?

No, but afaik Daimler is the last remaining stakeholder.

Agreed solution:

Set EcuMfix to obsolete to be able to remove it for the next minor release.

==TR ClassicPlatformReleaseOverview==

==Release Overview==

add Release Note to chapter 3.1.2.3 Obsolete Specifications:

"The following specification is set to status obsolete in this release:

Specification of ECU State Manager with fixed state machine (UID 444, SWS)"

==EcuMfixed==

+add sentence to 1 Scope of Document

"This specification is obsolete and will be removed from the standard in an upcoming release."

==ECUC XML Model==

< _____

< set atp.Status of EcuMFixedGeneral container and its content to obsolete (lower-Multiplicity = 0):

< The following parameters are affected:

< - EcuMTTIIEnabled

< - EcuMTTIIWakeupSourceRef

< - EcuMIncludeComM

<

< set atp.Status of EcuMFixedConfiguration container and its content to obsolete (lowerMultiplicity = 0):

< The following parameters are affected:

< - EcuMRunMinimumDuration

< - EcuMNvramReadallTimeout

< - EcuMNvramWriteallTimeout

< - EcuMNormalMcuModeRef

< - EcuMComMCommunicationAllowedList

< - EcuMFixedUser

< - EcuMDivisor

< - EcuMSleepModeRef

< - EcuMSuccessorRef

<

< The following subcontainers are affected:

< - EcuMDriverInitListTwo

< - EcuMDriverInitListThree

< - EcuMTTII

< - EcuMFixedUserConfig

—Last change on issue 77051 comment 36—

BW-C-Level:

Application	Specification	Bus
3	5	1

1.6 Specification Item ECUC_EcuM_00126

Trace References:

none

Content:

Name	EcuMNvramReadallTimeoutEcuMFixedConfiguration.EcuMNvramReadallTimeout		
Parent Container	EcuMFixedConfiguration		
Description	Period given in seconds for which the ECU State Manager will wait until it considers a ReadAll job of the NVRAM Manager as failed. Tags: atp.Status=obsolete		
Multiplicity	1		
Type	EcucFloatParamDef		
Range	[0 .. INF]		
Default value	-		
Post-Build Variant Value	true		
Value Configuration Class	Pre-compile time	-	
	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 #77051: [EcuMfixed] Set SWS_EcuStateManagerFixed to obsolete

Problem description:

Description/Motivation:

The need for having 2 Ecu Statemanager modules has vanished.
The specification of EcuMfix can be removed from the standard.

Was there already a decision?

No, but afaik Daimler is the last remaining stakeholder.

Agreed solution:

Set EcuMfix to obsolete to be able to remove it for the next minor release.

==TR ClassicPlatformReleaseOverview==

==Release Overview==

add Release Note to chapter 3.1.2.3 Obsolete Specifications:

"The following specification is set to status obsolete in this release:

Specification of ECU State Manager with fixed state machine (UID 444, SWS)"

==EcuMfixed==

+add sentence to 1 Scope of Document

"This specification is obsolete and will be removed from the standard in an upcoming release."

==ECUC XML Model==

< _____

< set atp.Status of EcuMFixedGeneral container and its content to obsolete (lower-Multiplicity = 0):

< The following parameters are affected:

< - EcuMTTIIEnabled

< - EcuMTTIIWakeupSourceRef

< - EcuMIncludeComM

<

< set atp.Status of EcuMFixedConfiguration container and its content to obsolete (lowerMultiplicity = 0):

< The following parameters are affected:

< - EcuMRunMinimumDuration

< - EcuMNvramReadallTimeout

< - EcuMNvramWriteallTimeout

< - EcuMNormalMcuModeRef

< - EcuMComMCommunicationAllowedList

< - EcuMFixedUser

< - EcuMDivisor

< - EcuMSleepModeRef

< - EcuMSuccessorRef

<

< The following subcontainers are affected:

< - EcuMDriverInitListTwo

< - EcuMDriverInitListThree

< - EcuMTTII

< - EcuMFixedUserConfig

–Last change on issue 77051 comment 36–

BW-C-Level:

Application	Specification	Bus
3	5	1

1.7 Specification Item ECUC_EcuM_00127

Trace References:

none

Content:

Name	EcuMNvramWriteallTimeoutEcuMFixedConfiguration.EcuMNvramWriteallTimeout		
Parent Container	EcuMFixedConfiguration		
Description	Period given in seconds for which the ECU State Manager will wait until it considers a WriteAll job of the NVRAM Manager as failed. Tags: atp.Status=obsolete		
Multiplicity	1		
Type	EcucFloatParamDef		
Range	[0 .. INF]		
Default value	-		
Post-Build Variant Value	true		
Value Configuration Class	Pre-compile time	-	
	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 #77051: [EcuMfixed] Set SWS_EcuStateManagerFixed to obsolete

Problem description:

Description/Motivation:

The need for having 2 Ecu Statemanager modules has vanished.
The specification of EcuMfix can be removed from the standard.

Was there already a decision?

No, but afaik Daimler is the last remaining stakeholder.

Agreed solution:

Set EcuMfix to obsolete to be able to remove it for the next minor release.

==TR ClassicPlatformReleaseOverview==

==Release Overview==

add Release Note to chapter 3.1.2.3 Obsolete Specifications:

"The following specification is set to status obsolete in this release:

Specification of ECU State Manager with fixed state machine (UID 444, SWS)"

==EcuMfixed==

+add sentence to 1 Scope of Document

"This specification is obsolete and will be removed from the standard in an upcoming release."

==ECUC XML Model==

< _____

< set atp.Status of EcuMFixedGeneral container and its content to obsolete (lower-Multiplicity = 0):

< The following parameters are affected:

< - EcuMTTIIEnabled

< - EcuMTTIIWakeupSourceRef

< - EcuMIncludeComM

<

< set atp.Status of EcuMFixedConfiguration container and its content to obsolete (lowerMultiplicity = 0):

< The following parameters are affected:

< - EcuMRunMinimumDuration

< - EcuMNvramReadallTimeout

< - EcuMNvramWriteallTimeout

< - EcuMNormalMcuModeRef

< - EcuMComMCommunicationAllowedList

< - EcuMFixedUser

< - EcuMDivisor

< - EcuMSleepModeRef

< - EcuMSuccessorRef

<

< The following subcontainers are affected:

< - EcuMDriverInitListTwo

< - EcuMDriverInitListThree

< - EcuMTTII

< - EcuMFixedUserConfig
–Last change on issue 77051 comment 36–

BW-C-Level:

Application	Specification	Bus
3	5	1

1.8 Specification Item ECUC_EcuM_00129

Trace References:

none

Content:

Name	EcuMRunMinimumDurationEcuMFixedConfiguration.EcuMRunMinimumDuration		
Parent Container	EcuMFixedConfiguration		
Description	Duration given in seconds for which the ECU State Manager will stay in RUN state even when no one requests RUN. This duration should be long at least as long as a SW-Cs needs to request RUN. Tags: atp.Status=obsolete		
Multiplicity	1		
Type	EcucFloatParamDef		
Range	[0 .. INF]		
Default value	-		
Post-Build Variant Value	true		
Value Configuration Class	Pre-compile time	-	
	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 #77051: [EcuMfixed] Set SWS_EcuStateManagerFixed to obsolete

Problem description:

Description/Motivation:

The need for having 2 Ecu Statemanager modules has vanished.
The specification of EcuMfix can be removed from the standard.

Was there already a decision?

No, but afaik Daimler is the last remaining stakeholder.

Agreed solution:

Set EcuMfix to obsolete to be able to remove it for the next minor release.

==TR ClassicPlatformReleaseOverview==

==Release Overview==

add Release Note to chapter 3.1.2.3 Obsolete Specifications:

"The following specification is set to status obsolete in this release:

Specification of ECU State Manager with fixed state machine (UID 444, SWS)"

==EcuMfixed==

+add sentence to 1 Scope of Document

"This specification is obsolete and will be removed from the standard in an upcoming release."

==ECUC XML Model==

<

< set atp.Status of EcuMFixedGeneral container and its content to obsolete (lower-Multiplicity = 0):

< The following parameters are affected:

< - EcuMTTIIEnabled

< - EcuMTTIIWakeupSourceRef

< - EcuMIncludeComM

<

< set atp.Status of EcuMFixedConfiguration container and its content to obsolete (lowerMultiplicity = 0):

< The following parameters are affected:

< - EcuMRunMinimumDuration

< - EcuMNvramReadallTimeout

< - EcuMNvramWriteallTimeout

< - EcuMNormalMcuModeRef

< - EcuMComMCommunicationAllowedList

< - EcuMFixedUser

< - EcuMDivisor

< - EcuMSleepModeRef

< - EcuMSuccessorRef

<

< The following subcontainers are affected:
< - EcuMDriverInitListTwo
< - EcuMDriverInitListThree
< - EcuMTTII
< - EcuMFixedUserConfig
–Last change on issue 77051 comment 36–

BW-C-Level:

Application	Specification	Bus
3	5	1

1.9 Specification Item ECUC_EcuM_00135

Trace References:

none

Content:

Name	EcuMSleepModeRefEcuMTTII.EcuMSleepModeRef		
Parent Container	EcuMTTII		
Description	This configuration parameter is a reference to a configured sleep mode that is used for TTI. Tags: atp.Status=obsolete		
Multiplicity	1 0..1		
Type	Symbolic name reference to [EcuMSleepMode]		
Post-Build Variant Value	false		
Value Configuration Class	Pre-compile time	X	VARIANT-POST-BUILD
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: local		

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77051: [EcuMfixed] Set SWS_EcuStateManagerFixed to obsolete

Problem description:

Description/Motivation:

The need for having 2 Ecu Statemanager modules has vanished.
The specification of EcuMfix can be removed from the standard.

Was there already a decision?
No, but afaik Daimler is the last remaining stakeholder.

Agreed solution:

Set EcuMfix to obsolete to be able to remove it for the next minor release.

==TR ClassicPlatformReleaseOverview==

==Release Overview==

add Release Note to chapter 3.1.2.3 Obsolete Specifications:

"The following specification is set to status obsolete in this release:

Specification of ECU State Manager with fixed state machine (UID 444, SWS)"

==EcuMfixed==

+add sentence to 1 Scope of Document

"This specification is obsolete and will be removed from the standard in an upcoming release."

==ECUC XML Model==

<

< set atp.Status of EcuMFixedGeneral container and its content to obsolete (lower-Multiplicity = 0):

< The following parameters are affected:

< - EcuMTTIIEnabled

< - EcuMTTIIWakeupSourceRef

< - EcuMIncludeComM

<

< set atp.Status of EcuMFixedConfiguration container and its content to obsolete (lowerMultiplicity = 0):

< The following parameters are affected:

< - EcuMRunMinimumDuration

< - EcuMNvramReadallTimeout

< - EcuMNvramWriteallTimeout

< - EcuMNormalMcuModeRef

< - EcuMComMCommunicationAllowedList

< - EcuMFixedUser

< - EcuMDivisor

< - EcuMSleepModeRef

< - EcuMSuccessorRef

<
< The following subcontainers are affected:
< - EcuMDriverInitListTwo
< - EcuMDriverInitListThree
< - EcuMTTII
< - EcuMFixedUserConfig
–Last change on issue 77051 comment 36–

BW-C-Level:

Application	Specification	Bus
3	5	1

1.10 Specification Item ECUC_EcuM_00141

Trace References:

none

Content:

Name	EcuMSuccessorRefEcuMTTII.EcuMSuccessorRef		
Parent Container	EcuMTTII		
Description	This parameter is a reference to the next sleep mode in the TTI protocol. Tags: atp.Status=obsolete		
Multiplicity	0..1		
Type	Symbolic name reference to [EcuMSleepMode]		
Post-Build Variant Multiplicity	false		
Post-Build Variant Value	false		
Multiplicity Configuration Class	Pre-compile time	X	VARIANT-POST-BUILD
	Link time	–	
	Post-build time	–	
Value Configuration Class	Pre-compile time	X	VARIANT-POST-BUILD
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: local		

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77051: [EcuMfixed] Set SWS_EcuStateManagerFixed to obsolete

Problem description:

Description/Motivation:

The need for having 2 Ecu Statemanager modules has vanished.
The specification of EcuMfix can be removed from the standard.

Was there already a decision?

No, but afaik Daimler is the last remaining stakeholder.

Agreed solution:

Set EcuMfix to obsolete to be able to remove it for the next minor release.

==TR ClassicPlatformReleaseOverview==

==Release Overview==

add Release Note to chapter 3.1.2.3 Obsolete Specifications:

"The following specification is set to status obsolete in this release:

Specification of ECU State Manager with fixed state machine (UID 444, SWS)"

==EcuMfixed==

+add sentence to 1 Scope of Document

"This specification is obsolete and will be removed from the standard in an upcoming release."

==ECUC XML Model==

< _____

< set atp.Status of EcuMFixedGeneral container and its content to obsolete (lower-Multiplicity = 0):

< The following parameters are affected:

< - EcuMTTIIEnabled

< - EcuMTTIIWakeupSourceRef

< - EcuMIncludeComM

<

< set atp.Status of EcuMFixedConfiguration container and its content to obsolete (lowerMultiplicity = 0):

< The following parameters are affected:

< - EcuMRunMinimumDuration

< - EcuMNvramReadallTimeout

< - EcuMNvramWriteallTimeout

< - EcuMNormalMcuModeRef

< - EcuMComMCommunicationAllowedList

< - EcuMFixedUser
< - EcuMDivisor
< - EcuMSleepModeRef
< - EcuMSuccessorRef
<
< The following subcontainers are affected:
< - EcuMDriverInitListTwo
< - EcuMDriverInitListThree
< - EcuMTTII
< - EcuMFixedUserConfig
–Last change on issue 77051 comment 36–

BW-C-Level:

Application	Specification	Bus
3	5	1

1.11 Specification Item ECUC_EcuM_00143

Trace References:

none

Content:

Container Name	EcuMTTII
Description	These containers describe the structures and the following configuration items describe its elements. These structures are concatenated to build a list as indicated by Figure 27 - Configuration Container Diagram. The list must contain at least one element when ECUM_TTII_ENABLED is set to true. Tags: atp.Status=obsolete
Configuration Parameters	

Included parameters:

Included Parameters	
Parameter Name	SWS Item ID
EcuMDivisor	ECUC_EcuM_00109
EcuMSleepModeRef	ECUC_EcuM_00135
EcuMSuccessorRef	ECUC_EcuM_00141

Included containers:

No Included Containers

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77051: [EcuMfixed] Set SWS_EcuStateManagerFixed to obsolete

Problem description:

Description/Motivation:

The need for having 2 Ecu Statemanager modules has vanished.
The specification of EcuMfix can be removed from the standard.

Was there already a decision?

No, but afaik Daimler is the last remaining stakeholder.

Agreed solution:

Set EcuMfix to obsolete to be able to remove it for the next minor release.

==TR ClassicPlatformReleaseOverview==

==Release Overview==

add Release Note to chapter 3.1.2.3 Obsolete Specifications:

"The following specification is set to status obsolete in this release:

Specification of ECU State Manager with fixed state machine (UID 444, SWS)"

==EcuMfixed==

+add sentence to 1 Scope of Document

"This specification is obsolete and will be removed from the standard in an upcoming release."

==ECUC XML Model==

<

< set atp.Status of EcuMFixedGeneral container and its content to obsolete (lower-Multiplicity = 0):

< The following parameters are affected:

< - EcuMTTIIEnabled

< - EcuMTTIIWakeupSourceRef

< - EcuMIncludeComM

<

< set atp.Status of EcuMFixedConfiguration container and its content to obsolete

(lowerMultiplicity = 0):

< The following parameters are affected:

- < - EcuMRunMinimumDuration
- < - EcuMNvramReadallTimeout
- < - EcuMNvramWriteallTimeout
- < - EcuMNormalMcuModeRef
- < - EcuMComMCommunicationAllowedList
- < - EcuMFixedUser
- < - EcuMDivisor
- < - EcuMSleepModeRef
- < - EcuMSuccessorRef

<

< The following subcontainers are affected:

- < - EcuMDriverInitListTwo
- < - EcuMDriverInitListThree
- < - EcuMTTII
- < - EcuMFixedUserConfig

–Last change on issue 77051 comment 36–

BW-C-Level:

Application	Specification	Bus
3	5	1

1.12 Specification Item ECUC_EcuM_00144

Trace References:

none

Content:

Name	EcuMTTIIEnabledEcuMFixedGeneral.EcuMTTIIEnabled		
Parent Container	EcuMFixedGeneral		
Description	Boolean switch to enable / disable TTII Tags: atp.Status=obsolete		
Multiplicity	1 0..1		
Type	EcucBooleanParamDef		
Default value	–		
Post-Build Variant Value	false		
Value Configuration Class	Pre-compile time	X	VARIANT-POST-BUILD
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: local		

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77051: [EcuMfixed] Set SWS_EcuStateManagerFixed to obsolete

Problem description:

Description/Motivation:

The need for having 2 Ecu Statemanager modules has vanished.
The specification of EcuMfix can be removed from the standard.

Was there already a decision?

No, but afaik Daimler is the last remaining stakeholder.

Agreed solution:

Set EcuMfix to obsolete to be able to remove it for the next minor release.

==TR ClassicPlatformReleaseOverview==

==Release Overview==

add Release Note to chapter 3.1.2.3 Obsolete Specifications:

"The following specification is set to status obsolete in this release:

Specification of ECU State Manager with fixed state machine (UID 444, SWS)"

==EcuMfixed==

+add sentence to 1 Scope of Document

"This specification is obsolete and will be removed from the standard in an upcoming release."

==ECUC XML Model==

<

< set atp.Status of EcuMFixedGeneral container and its content to obsolete (lower-Multiplicity = 0):

< The following parameters are affected:

< - EcuMTTIIEnabled

< - EcuMTTIIWakeupSourceRef

< - EcuMIncludeComM

<

< set atp.Status of EcuMFixedConfiguration container and its content to obsolete

(lowerMultiplicity = 0):

< The following parameters are affected:

- < - EcuMRunMinimumDuration
- < - EcuMNvramReadallTimeout
- < - EcuMNvramWriteallTimeout
- < - EcuMNormalMcuModeRef
- < - EcuMComMCommunicationAllowedList
- < - EcuMFixedUser
- < - EcuMDivisor
- < - EcuMSleepModeRef
- < - EcuMSuccessorRef

<

< The following subcontainers are affected:

- < - EcuMDriverInitListTwo
- < - EcuMDriverInitListThree
- < - EcuMTTII
- < - EcuMFixedUserConfig

–Last change on issue 77051 comment 36–

BW-C-Level:

Application	Specification	Bus
3	5	1

1.13 Specification Item ECUC_EcuM_00145

Trace References:

none

Content:

Name	EcuMTTIIWakeupSourceRefEcuMFixedGeneral.EcuMTTIIWakeupSourceRef		
Parent Container	EcuMFixedGeneral		
Description	This configuration parameter references the initial sleep mode to be used by TTII when TTII is activated after a RUN mode. Tags: atp.Status=obsolete		
Multiplicity	1 0..1		
Type	Symbolic name reference to [EcuMWakeupSource]		
Post-Build Variant Value	false		
Value Configuration Class	Pre-compile time	X	VARIANT-POST-BUILD
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: local		

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77051: [EcuMfixed] Set SWS_EcuStateManagerFixed to obsolete

Problem description:

Description/Motivation:

The need for having 2 Ecu Statemanager modules has vanished.
The specification of EcuMfix can be removed from the standard.

Was there already a decision?

No, but afaik Daimler is the last remaining stakeholder.

Agreed solution:

Set EcuMfix to obsolete to be able to remove it for the next minor release.

==TR ClassicPlatformReleaseOverview==

==Release Overview==

add Release Note to chapter 3.1.2.3 Obsolete Specifications:

"The following specification is set to status obsolete in this release:

Specification of ECU State Manager with fixed state machine (UID 444, SWS)"

==EcuMfixed==

+add sentence to 1 Scope of Document

"This specification is obsolete and will be removed from the standard in an upcoming release."

==ECUC XML Model==

<

< set atp.Status of EcuMFixedGeneral container and its content to obsolete (lower-Multiplicity = 0):

< The following parameters are affected:

< - EcuMTTIIEnabled

< - EcuMTTIIWakeupSourceRef

< - EcuMIncludeComM

<

< set atp.Status of EcuMFixedConfiguration container and its content to obsolete

(lowerMultiplicity = 0):

< The following parameters are affected:

- < - EcuMRunMinimumDuration
- < - EcuMNvramReadallTimeout
- < - EcuMNvramWriteallTimeout
- < - EcuMNormalMcuModeRef
- < - EcuMComMCommunicationAllowedList
- < - EcuMFixedUser
- < - EcuMDivisor
- < - EcuMSleepModeRef
- < - EcuMSuccessorRef

<

< The following subcontainers are affected:

- < - EcuMDriverInitListTwo
- < - EcuMDriverInitListThree
- < - EcuMTTII
- < - EcuMFixedUserConfig

–Last change on issue 77051 comment 36–

BW-C-Level:

Application	Specification	Bus
3	5	1

1.14 Specification Item ECUC_EcuM_00147

Trace References:

none

Content:

Container Name	EcuMFixedUserConfigEcuMFixedUserConfig
Description	<p>These containers describe the identifiers that are needed to refer to a software component or another appropriate entity in the system which is designated to request the RUN state. Application requestors refer to entities above RTE, system requestors to entities below RTE (e.g. Communication Manager).</p> <p>Tags: atp.Status=obsolete</p>
Configuration Parameters	

Included parameters:

Included Parameters	
Parameter Name	SWS Item ID
EcuMFixedUser	ECUC_EcuM_00202

Included containers:

No Included Containers

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77051: [EcuMfixed] Set SWS_EcuStateManagerFixed to obsolete

Problem description:

Description/Motivation:

The need for having 2 Ecu Statemanager modules has vanished.
The specification of EcuMfix can be removed from the standard.

Was there already a decision?

No, but afaik Daimler is the last remaining stakeholder.

Agreed solution:

Set EcuMfix to obsolete to be able to remove it for the next minor release.

==TR ClassicPlatformReleaseOverview==

==Release Overview==

add Release Note to chapter 3.1.2.3 Obsolete Specifications:

"The following specification is set to status obsolete in this release:

Specification of ECU State Manager with fixed state machine (UID 444, SWS)"

==EcuMfixed==

+add sentence to 1 Scope of Document

"This specification is obsolete and will be removed from the standard in an upcoming release."

==ECUC XML Model==

<

< set atp.Status of EcuMFixedGeneral container and its content to obsolete (lower-Multiplicity = 0):

< The following parameters are affected:

- < - EcuMTTIIEnabled
- < - EcuMTTIIWakeupSourceRef
- < - EcuMIncludeComM
- <
- < set atp.Status of EcuMFixedConfiguration container and its content to obsolete (lowerMultiplicity = 0):
- < The following parameters are affected:
- < - EcuMRunMinimumDuration
- < - EcuMNvramReadallTimeout
- < - EcuMNvramWriteallTimeout
- < - EcuMNormalMcuModeRef
- < - EcuMComMCommunicationAllowedList
- < - EcuMFixedUser
- < - EcuMDivisor
- < - EcuMSleepModeRef
- < - EcuMSuccessorRef
- <
- < The following subcontainers are affected:
- < - EcuMDriverInitListTwo
- < - EcuMDriverInitListThree
- < - EcuMTTII
- < - EcuMFixedUserConfig
- Last change on issue 77051 comment 36–

BW-C-Level:

Application	Specification	Bus
3	5	1

1.15 Specification Item ECUC_EcuM_00165

Trace References:

none

Content:

Container Name	EcuMFixedConfigurationEcuMFixedConfiguration
Description	This container contains the configuration (parameters) of the EcuMFixed. Only applicable if EcuMFixed is implemented. Tags: atp.Status=obsolete
Configuration Parameters	

Included parameters:

Included Parameters	
Parameter Name	SWS Item ID
EcuMNvramReadallTimeout	ECUC_EcuM_00126
EcuMNvramWriteallTimeout	ECUC_EcuM_00127
EcuMRunMinimumDuration	ECUC_EcuM_00129
EcuMComMCommunicationAllowedList	ECUC_EcuM_00191
EcuMNormalMcuModeRef	ECUC_EcuM_00125

Included containers:

Included Containers		
Container Name	Multiplicity	Scope / Dependency
EcuMDriverInitListThree	0..1	<p>Container for Init Block III.</p> <p>This container holds a list of modules to be initialized. Each module in the list will be called for initialization in the list order.</p> <p>All modules in this list are initialized after the OS is started and so these modules may use OS support. These modules may also rely on the Nvram ReadAll job to have provided all data.</p> <p>Tags: atp.Status=obsolete</p>
EcuMDriverInitListTwo	0..1	<p>Container for Init Block II.</p> <p>This container holds a list of modules to be initialized. Each module in the list will be called for initialization in the list order.</p> <p>All modules in this list are initialized after the OS is started and so these modules may use OS support. These modules may not rely on the Nvram ReadAll job to have provided all data.</p> <p>Tags: atp.Status=obsolete</p>
EcuMFixedUserConfig	1.0..*	<p>These containers describe the identifiers that are needed to refer to a software component or another appropriate entity in the system which is designated to request the RUN state. Application requestors refer to entities above RTE, system requestors refer to entities below RTE (e.g. Communication Manager).</p> <p>Tags: atp.Status=obsolete</p>
EcuMTTII	0..*	<p>These containers describe the structures and the following configuration items describe its elements. These structures are concatenated to build a list as indicated by Figure 27 - Configuration Container Diagram.</p> <p>The list must contain at least one element when ECUM_TTII_ENABLED is set to true.</p> <p>Tags: atp.Status=obsolete</p>

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77051: [EcuMfixed] Set SWS_EcuStateManagerFixed to obsolete

Problem description:

Description/Motivation:

The need for having 2 Ecu Statemanager modules has vanished.
The specification of EcuMfix can be removed from the standard.

Was there already a decision?

No, but afaik Daimler is the last remaining stakeholder.

Agreed solution:

Set EcuMfix to obsolete to be able to remove it for the next minor release.

==TR ClassicPlatformReleaseOverview==

==Release Overview==

add Release Note to chapter 3.1.2.3 Obsolete Specifications:

"The following specification is set to status obsolete in this release:

Specification of ECU State Manager with fixed state machine (UID 444, SWS)"

==EcuMfixed==

+add sentence to 1 Scope of Document

"This specification is obsolete and will be removed from the standard in an upcoming release."

==ECUC XML Model==

<

< set atp.Status of EcuMFixedGeneral container and its content to obsolete (lower-Multiplicity = 0):

< The following parameters are affected:

< - EcuMTTIIEnabled

< - EcuMTTIIWakeupSourceRef

< - EcuMIncludeComM

<

< set atp.Status of EcuMFixedConfiguration container and its content to obsolete

(lowerMultiplicity = 0):

< The following parameters are affected:

- < - EcuMRunMinimumDuration
- < - EcuMNvramReadallTimeout
- < - EcuMNvramWriteallTimeout
- < - EcuMNormalMcuModeRef
- < - EcuMComMCommunicationAllowedList
- < - EcuMFixedUser
- < - EcuMDivisor
- < - EcuMSleepModeRef
- < - EcuMSuccessorRef

<

< The following subcontainers are affected:

- < - EcuMDriverInitListTwo
- < - EcuMDriverInitListThree
- < - EcuMTTII
- < - EcuMFixedUserConfig

–Last change on issue 77051 comment 36–

BW-C-Level:

Application	Specification	Bus
3	5	1

1.16 Specification Item ECUC_EcuM_00166

Trace References:

none

Content:

Container Name	EcuMFixedGeneralEcuMFixedGeneral
Description	This container holds the general, pre-compile configuration parameters for the EcuMFixed. Only applicable if EcuMFixed is implemented. Tags: atp.Status=obsolete
Configuration Parameters	

Included parameters:

Included Parameters	
Parameter Name	SWS Item ID
EcuMIncludeComM	ECUC_EcuM_00189
EcuMTTIIEnabled	ECUC_EcuM_00144

Included Parameters	
Parameter Name	SWS Item ID
EcuMTTIIWakeupSourceRef	ECUC_EcuM_00145

Included containers:

No Included Containers

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77051: [EcuMfixed] Set SWS_EcuStateManagerFixed to obsolete

Problem description:

Description/Motivation:

The need for having 2 Ecu Statemanager modules has vanished.
The specification of EcuMfix can be removed from the standard.

Was there already a decision?

No, but afaik Daimler is the last remaining stakeholder.

Agreed solution:

Set EcuMfix to obsolete to be able to remove it for the next minor release.

==TR ClassicPlatformReleaseOverview==

==Release Overview==

add Release Note to chapter 3.1.2.3 Obsolete Specifications:

"The following specification is set to status obsolete in this release:

Specification of ECU State Manager with fixed state machine (UID 444, SWS)"

==EcuMfixed==

+add sentence to 1 Scope of Document

"This specification is obsolete and will be removed from the standard in an upcoming release."

==ECUC XML Model==

< _____
< set atp.Status of EcuMFixedGeneral container and its content to obsolete (lower-Multiplicity = 0):
< The following parameters are affected:
< - EcuMTTIIEnabled
< - EcuMTTIIWakeupSourceRef
< - EcuMIncludeComM
<
< set atp.Status of EcuMFixedConfiguration container and its content to obsolete (lowerMultiplicity = 0):
< The following parameters are affected:
< - EcuMRunMinimumDuration
< - EcuMNvramReadallTimeout
< - EcuMNvramWriteallTimeout
< - EcuMNormalMcuModeRef
< - EcuMComMCommunicationAllowedList
< - EcuMFixedUser
< - EcuMDivisor
< - EcuMSleepModeRef
< - EcuMSuccessorRef
<
< The following subcontainers are affected:
< - EcuMDriverInitListTwo
< - EcuMDriverInitListThree
< - EcuMTTII
< - EcuMFixedUserConfig
-Last change on issue 77051 comment 36-

BW-C-Level:

Application	Specification	Bus
3	5	1

1.17 Specification Item ECUC_EcuM_00189

Trace References:

none

Content:

Name	EcuMIncludeComMEcuMFixedGeneral.EcuMIncludeComM
Parent Container	EcuMFixedGeneral

Description	This configuration parameter defines whether the communication manager is supported by Ecu M. This feature is presented for development purpose to compile out the communication manager in the early debugging phase. Tags: atp.Status=obsolete		
Multiplicity	1 0..1		
Type	EcucBooleanParamDef		
Default value	-		
Post-Build Variant Value	false		
Value Configuration Class	Pre-compile time	X	VARIANT-POST-BUILD
	Link time	-	
	Post-build time	-	
Scope / Dependency	scope: local		

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77051: [EcuMfixed] Set SWS_EcuStateManagerFixed to obsolete

Problem description:

Description/Motivation:

The need for having 2 Ecu Statemanager modules has vanished.
The specification of EcuMfix can be removed from the standard.

Was there already a decision?

No, but afaik Daimler is the last remaining stakeholder.

Agreed solution:

Set EcuMfix to obsolete to be able to remove it for the next minor release.

==TR ClassicPlatformReleaseOverview==

==Release Overview==

add Release Note to chapter 3.1.2.3 Obsolete Specifications:

"The following specification is set to status obsolete in this release:

Specification of ECU State Manager with fixed state machine (UID 444, SWS)"

==EcuMfixed==

+add sentence to 1 Scope of Document

"This specification is obsolete and will be removed from the standard in an upcoming

release."

==ECUC XML Model==

< _____
< set atp.Status of EcuMFixedGeneral container and its content to obsolete (lower-Multiplicity = 0):
< The following parameters are affected:
< - EcuMTTIIEnabled
< - EcuMTTIIWakeupSourceRef
< - EcuMIncludeComM
<
< set atp.Status of EcuMFixedConfiguration container and its content to obsolete (lowerMultiplicity = 0):
< The following parameters are affected:
< - EcuMRunMinimumDuration
< - EcuMNvramReadallTimeout
< - EcuMNvramWriteallTimeout
< - EcuMNormalMcuModeRef
< - EcuMComMCommunicationAllowedList
< - EcuMFixedUser
< - EcuMDivisor
< - EcuMSleepModeRef
< - EcuMSuccessorRef
<
< The following subcontainers are affected:
< - EcuMDriverInitListTwo
< - EcuMDriverInitListThree
< - EcuMTTII
< - EcuMFixedUserConfig
-Last change on issue 77051 comment 36-

BW-C-Level:

Application	Specification	Bus
3	5	1

1.18 Specification Item ECUC_EcuM_00191

Trace References:

none

Content:

Name	EcuMComMCommunicationAllowedListEcuMFixedConfiguration.EcuMComMCommunicationAllowedList		
Parent Container	EcuMFixedConfiguration		
Description	These parameters contain references to the ComMChannels for which EcuM has to call ComM_CommunicationAllowed. Tags: atp.Status=obsolete		
Multiplicity	0..*		
Type	Symbolic name reference to [ComMChannel]		
Post-Build Variant Multiplicity	false		
Post-Build Variant Value	false		
Multiplicity Configuration Class	Pre-compile time	X	VARIANT-POST-BUILD
	Link time	–	
	Post-build time	–	
Value Configuration Class	Pre-compile time	X	VARIANT-POST-BUILD
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: local		

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77051: [EcuMfixed] Set SWS_EcuStateManagerFixed to obsolete

Problem description:

Description/Motivation:

The need for having 2 Ecu Statemanager modules has vanished.
The specification of EcuMfix can be removed from the standard.

Was there already a decision?

No, but afaik Daimler is the last remaining stakeholder.

Agreed solution:

Set EcuMfix to obsolete to be able to remove it for the next minor release.

==TR ClassicPlatformReleaseOverview==

==Release Overview==

add Release Note to chapter 3.1.2.3 Obsolete Specifications:

"The following specification is set to status obsolete in this release:

Specification of ECU State Manager with fixed state machine (UID 444, SWS)"

==EcuMfixed==

+add sentence to 1 Scope of Document

"This specification is obsolete and will be removed from the standard in an upcoming release."

==ECUC XML Model==

< _____

< set atp.Status of EcuMFixedGeneral container and its content to obsolete (lower-Multiplicity = 0):

< The following parameters are affected:

< - EcuMTTIIEnabled

< - EcuMTTIIWakeupSourceRef

< - EcuMIncludeComM

<

< set atp.Status of EcuMFixedConfiguration container and its content to obsolete (lowerMultiplicity = 0):

< The following parameters are affected:

< - EcuMRunMinimumDuration

< - EcuMNvramReadallTimeout

< - EcuMNvramWriteallTimeout

< - EcuMNormalMcuModeRef

< - EcuMComMCommunicationAllowedList

< - EcuMFixedUser

< - EcuMDivisor

< - EcuMSleepModeRef

< - EcuMSuccessorRef

<

< The following subcontainers are affected:

< - EcuMDriverInitListTwo

< - EcuMDriverInitListThree

< - EcuMTTII

< - EcuMFixedUserConfig

–Last change on issue 77051 comment 36–

BW-C-Level:

Application	Specification	Bus
3	5	1

1.19 Specification Item ECUC_EcuM_00202

Trace References:

none

Content:

Name	EcuMFixedUserEcuMFixedUserConfig.EcuMFixedUser		
Parent Container	EcuMFixedUserConfig		
Description	Parameter used to identify one user. Tags: atp.Status=obsolete		
Multiplicity	1 0..1		
Type	EcucIntegerParamDef (Symbolic Name generated for this parameter)		
Range	0 .. 255		
Default value	-		
Post-Build Variant Value	false		
Value Configuration Class	Pre-compile time	X	VARIANT-POST-BUILD
	Link time	-	
	Post-build time	-	
Scope / Dependency	scope: local		

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77051: [EcuMfixed] Set SWS_EcuStateManagerFixed to obsolete

Problem description:

Description/Motivation:

The need for having 2 Ecu Statemanager modules has vanished.
The specification of EcuMfix can be removed from the standard.

Was there already a decision?

No, but afaik Daimler is the last remaining stakeholder.

Agreed solution:

Set EcuMfix to obsolete to be able to remove it for the next minor release.

==TR ClassicPlatformReleaseOverview==
==Release Overview==

add Release Note to chapter 3.1.2.3 Obsolete Specifications:

"The following specification is set to status obsolete in this release:

Specification of ECU State Manager with fixed state machine (UID 444, SWS)"

==EcuMfixed==

+add sentence to 1 Scope of Document

"This specification is obsolete and will be removed from the standard in an upcoming release."

==ECUC XML Model==

< _____

< set atp.Status of EcuMFixedGeneral container and its content to obsolete (lower-Multiplicity = 0):

< The following parameters are affected:

< - EcuMTTIIEnabled

< - EcuMTTIIWakeupSourceRef

< - EcuMIncludeComM

<

< set atp.Status of EcuMFixedConfiguration container and its content to obsolete (lowerMultiplicity = 0):

< The following parameters are affected:

< - EcuMRunMinimumDuration

< - EcuMNvramReadallTimeout

< - EcuMNvramWriteallTimeout

< - EcuMNormalMcuModeRef

< - EcuMComMCommunicationAllowedList

< - EcuMFixedUser

< - EcuMDivisor

< - EcuMSleepModeRef

< - EcuMSuccessorRef

<

< The following subcontainers are affected:

< - EcuMDriverInitListTwo

< - EcuMDriverInitListThree

< - EcuMTTII

< - EcuMFixedUserConfig

–Last change on issue 77051 comment 36–

BW-C-Level:

Application	Specification	Bus
3	5	1

1.20 Specification Item ECUC_EcuM_00225

Trace References:

none

Content:

Module Name	EcuMEcuM
Module Description	Configuration of the EcuM (ECU State Manager) module.
Post-Build Variant Support	true
Supported Config Variants	VARIANT-POST-BUILD

Included containers:

Included Containers		
Container Name	Multiplicity	Scope / Dependency
EcuMConfiguration	1	This container contains the configuration (parameters) of the ECU State Manager.
EcuMFixedGeneral	0..1	This container holds the general, pre-compile configuration parameters for the EcuMFixed. Only applicable if EcuMFixed is implemented. Tags: atp.Status=obsolete
EcuMFlexGeneral	0..1	This container holds the general, pre-compile configuration parameters for the EcuMFlex. Only applicable if EcuMFlex is implemented.
EcuMGeneral	1	This container holds the general, pre-compile configuration parameters.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77051: [EcuMfixed] Set SWS_EcuStateManagerFixed to obsolete

Problem description:

Description/Motivation:

The need for having 2 Ecu Statemanager modules has vanished.

The specification of EcuMfix can be removed from the standard.

Was there already a decision?

No, but afaik Daimler is the last remaining stakeholder.

Agreed solution:

Set EcuMfix to obsolete to be able to remove it for the next minor release.

==TR ClassicPlatformReleaseOverview==

==Release Overview==

add Release Note to chapter 3.1.2.3 Obsolete Specifications:

"The following specification is set to status obsolete in this release:

Specification of ECU State Manager with fixed state machine (UID 444, SWS)"

==EcuMfixed==

+add sentence to 1 Scope of Document

"This specification is obsolete and will be removed from the standard in an upcoming release."

==ECUC XML Model==

< _____

< set atp.Status of EcuMFixedGeneral container and its content to obsolete (lower-Multiplicity = 0):

< The following parameters are affected:

< - EcuMTTIIEnabled

< - EcuMTTIIWakeupSourceRef

< - EcuMIncludeComM

<

< set atp.Status of EcuMFixedConfiguration container and its content to obsolete (lowerMultiplicity = 0):

< The following parameters are affected:

< - EcuMRunMinimumDuration

< - EcuMNvramReadallTimeout

< - EcuMNvramWriteallTimeout

< - EcuMNormalMcuModeRef

< - EcuMComMCommunicationAllowedList

< - EcuMFixedUser

< - EcuMDivisor

< - EcuMSleepModeRef
< - EcuMSuccessorRef
<
< The following subcontainers are affected:
< - EcuMDriverInitListTwo
< - EcuMDriverInitListThree
< - EcuMTTII
< - EcuMFixedUserConfig
–Last change on issue 77051 comment 36–

BW-C-Level:

Application	Specification	Bus
3	5	1

1.21 Specification Item SWS_EcuM_02337

Trace References:

none

Content:

If the pointer to the shutdownMode parameter is NULL, EcuM_GetLastShutdownTarget shall simply ignore the shutdownMode parameter and return the last shutdown target regardless of whether it was SLEEP or not. If **Default Development** Error Detection is activated, EcuM_GetShutdownTarget shall send the ECUM_E_PARAM_POINTER development error to the DET module.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #73570: No "default error" in AUTOSAR

Problem description:

The DET was renamed from development error tracer to default error tracer.

This change was most of the time done automatically and unfortunately renamed "development error" to "default error".

"default error" should always be followed by "tracer", otherwise, "development error" is probably the right term.

This could increase the impact (compared to my selection of impacted document, but formally, the configuration parameters *DevErrorDetect are not using the

correct description:

"Switches the Default Error Tracer (Det) detection and notification..."

The parameter switches on/off the development error detection. The DET does not need to be detected and can be present even when the parameter is set to false.

Agreed solution:

Rename "default error" to "development error" in all impacted documents, but not in an automated way (Do not change "default error tracer" to "development error tracer"!)

Blueprint/Example:

- sub chapter is now called "7.x Default errors"

- "[SWS_xxx_yyyyy]

In case default error detection is enabled for the xxxx module: The xxxx module shall check API parameters for validity and report detected errors to the DET. ()"

- "[SWS_xxx_yyyyy]

If default error detection is enabled: the function shall check that the service xxx_Init was previously called. If the check fails, the function shall raise the default error XXX_E_NOT_INITIALIZED otherwise (if DET is disabled) return E_NOT_OK. ()"

- "In case default errors are enabled,..."

- "module raises the Default error XXX_E_TRANSITION"

- "The DET provides services to store default errors"

...

The correct text would be:

- sub chapter is called "7.x Development errors"

- "[SWS_xxx_yyyyy]

In case development error detection is enabled for the xxxx module: The xxxx module shall check API parameters for validity and report detected development errors to the DET. ()"

- "[SWS_xxx_yyyyy]

If development error detection is enabled: the function shall check that the service xxx_Init was previously called. If the check fails, the function shall raise the development error XXX_E_NOT_INITIALIZED otherwise (if DET is disabled) return E_NOT_OK. ()"

- "In case development errors are enabled,..."

- "module raises the development error XXX_E_TRANSITION"

- "The DET provides services to store development errors"

Solution for SWS_RTE:

– SWS_RTE –

- Change 4.8 Default errors to 4.8 Development errors
- Change "Errors which can occur at runtime in the RTE are classified as default errors" to "Errors which can occur at runtime in the RTE are classified as development errors"
- Remove [SWS_Rte_07676]
- Change [SWS_RTE_06611] "If a violation is detected the RTE shall report a default error to the DET." to "If a violation is detected the RTE shall report a development error to the DET."
- Change [SWS_Rte_06631]
[SWS_Rte_06631] d The RTE shall use the OS Application Identifier as the Instance Id to enable the developer to identify in which runtime section of the RTE the error occurs. This Instance ID is even unique across multi cores and so implicitly allows the development error to be traced to a specific core. c(SRS_BSW_00337)

SRS_Libraries:

- In chapter "3 Acronyms and abbreviations": Rename "Development Error Tracer" to "Default Error Tracer"

SRS_SPALGeneral:

- In chapter "6.1.1.3.1 [SRS_SPAL_00157] ...": Rename "Development Error Tracer" to "Default Error Tracer"
- In chapter "6.1.1.4.2 [SRS_SPAL_12448] ...": Rename "Development Error Tracer" to "Default Error Tracer"

SRS_FlashTest:

- In chapter "6.1 Functional Requirements": Rename "Development Error Tracer" to "Default Error Tracer"
- In chapter "7 References":
Rename "Development Error Tracer" to "Default Error Tracer"
Rename "AUTOSAR_SWS_DevelopmentErrorTracer" to "AUTOSAR_SWS_DefaultErrorTracer"

SWS_MFXLibrary:

- In chapter "2 Acronyms and abbreviations": Rename "Development Error Tracer" to "Default Error Tracer"

SWS_MemoryAbstractionInterface:

- In chapter "3.1 Input documents":

Rename "Development Error Tracer" to "Default Error Tracer"

Rename "AUTOSAR_SWS_DevelopmentErrorTracer" to "AUTOSAR_SWS_DefaultErrorTracer"

SWS_FlexRayNetworkManagement:

- In chapter "3.3 Related AUTOSAR documents":

Rename "Development Error Tracer" to "Default Error Tracer"

Rename "AUTOSAR_SWS_DevelopmentErrorTracer" to "AUTOSAR_SWS_DefaultErrorTracer"

SWS_CANStateManager:

- In chapter "3.1 Input documents": Rename "AUTOSAR_SWS_DevelopmentErrorTracer" to "AUTOSAR_SWS_DefaultErrorTracer"

SWS_PDURouter:

- In chapter "3.1 Input documents": Rename "AUTOSAR_SWS_DevelopmentErrorTracer" to "AUTOSAR_SWS_DefaultErrorTracer"

SWS_EEPROMDriver:

- In chapter "3.1 Input documents": Rename "AUTOSAR_SWS_DevelopmentErrorTracer" to "AUTOSAR_SWS_DefaultErrorTracer"

–Last change on issue 73570 comment 47–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.22 Specification Item SWS_EcuM_02563

Trace References:

none

Content:

When hardware has been put into a sleep mode during SHUTDOWN then this hardware must be restarted by its driver. The ECU Manager module shall invoke the Wakeup Restart Sequence (see Section REFActivities in the WakeupRestart Sequence).

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76706: [EcuM] SWS_EcuM_02563 is not relevant to EcuM flex

Problem description:

Following requirement assumes state transition with "SHUTDOWN" mode before entering sleep mode. However, EcuM flex doesn't require it.

In short: [SWS_EcuM_02563] is for EcuM fixed, but not for EcuM flex.

[SWS_EcuM_02563] When hardware has been put into a sleep mode during SHUTDOWN then this hardware must be restarted by its driver. The ECU Manager module shall invoke in the WakeupRestart Sequence (see Section 7.6.4 Activities in the WakeupRestart Sequence).()

Agreed solution:

Remove [SWS_EcuM_02563]

BW-C-Level:

Application	Specification	Bus
1	4	1

1.23 Specification Item SWS_EcuM_02788**Trace References:**

none

Content:

If the pointer to the shutdownMode parameter is NULL, EcuM_GetShutdownTarget shall simply ignore the shutdownMode parameter. If **Default Development** Error Detection is activated, EcuM_GetShutdownTarget shall send the ECUM_E_PARAM_POINTER development error to the DET module.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #73570: No "default error" in AUTOSAR

Problem description:

The DET was renamed from development error tracer to default error tracer.

This change was most of the time done automatically and unfortunately re-named "developement error" to "default error".

"default error" should always be followed by "tracer", otherwise, "development error" is probably the right term.

This could increase the impact (compared to my selection of impacted document, but formally, the configuration parameters *DevErrorDetect are not using the correct description:

"Switches the Default Error Tracer (Det) detection and notification..."

The parameter switches on/off the development error detection. The DET does not need to be detected and can be present even when the parameter is set to false.

Agreed solution:

Rename "default error" to "development error" in all impacted documents, but not in an automated way (Do not change "default error tracer" to "development error tracer"!)

Blueprint/Example:

- sub chapter is now called "7.x Default errors"

- "[SWS_xxx_yyyyyy]

In case default error detection is enabled for the xxxx module: The xxxx module shall check API parameters for validity and report detected errors to the DET. ()"

- "[SWS_xxx_yyyyyy]

If default error detection is enabled: the function shall check that the service xxx_Init was previously called. If the check fails, the function shall raise the default error XXX_E_NOT_INITIALIZED otherwise (if DET is disabled) return E_NOT_OK. ()"

- "In case default errors are enabled,..."

- "module raises the Default error XXX_E_TRANSITION"

- "The DET provides services to store default errors"

...

The correct text would be:

- sub chapter is called "7.x Development errors"

- "[SWS_xxx_yyyyyy]

In case development error detection is enabled for the xxxx module: The xxxx module shall check API parameters for validity and report detected development errors to the DET. ()"

- "[SWS_xxx_yyyyyy]

If development error detection is enabled: the function shall check that the service xxx_Init was previously called. If the check fails, the function shall raise the development error XXX_E_NOT_INITIALIZED otherwise (if DET is disabled) return

E_NOT_OK. ()"

- "In case development errors are enabled,..."
- "module raises the development error XXX_E_TRANSITION"
- "The DET provides services to store development errors"

Solution for SWS_RTE:

- SWS_RTE —
 - Change 4.8 Default errors to 4.8 Development errors
 - Change "Errors which can occur at runtime in the RTE are classified as default errors" to "Errors which can occur at runtime in the RTE are classified as development errors"
 - Remove [SWS_Rte_07676]
 - Change [SWS_RTE_06611]"If a violation is detected the RTE shall report a default error to the DET." to "If a violation is detected the RTE shall report a development error to the DET."
 - Change [SWS_Rte_06631]
- [SWS_Rte_06631] d The RTE shall use the OS Application Identifier as the Instance Id to enable the developer to identify in which runtime section of the RTE the error occurs. This Instance ID is even unique across multi cores and so implicitly allows the development error to be traced to a specific core. c(SRS_BSW_00337)

SRS_Libraries:

- In chapter "3 Acronyms and abbreviations": Rename "Development Error Tracer" to "Default Error Tracer"

SRS_SPALGeneral:

- In chapter "6.1.1.3.1 [SRS_SPAL_00157] ...": Rename "Development Error Tracer" to "Default Error Tracer"
- In chapter "6.1.1.4.2 [SRS_SPAL_12448] ...": Rename "Development Error Tracer" to "Default Error Tracer"

SRS_FlashTest:

- In chapter "6.1 Functional Requirements": Rename "Development Error Tracer" to "Default Error Tracer"
 - In chapter "7 References":
Rename "Development Error Tracer" to "Default Error Tracer"
- | | | |
|---|----|----------------------------------|
| Rename "AUTOSAR_SWS_DevelopmentErrorTracer" | to | "AUTOSAR_SWS_DefaultErrorTracer" |
|---|----|----------------------------------|

SWS_MFXLibrary:

- In chapter "2 Acronyms and abbreviations": Rename "Development Error Tracer" to "Default Error Tracer"

SWS_MemoryAbstractionInterface:

- In chapter "3.1 Input documents":
Rename "Development Error Tracer" to "Default Error Tracer"
Rename "AUTOSAR_SWS_DevelopmentErrorTracer" to "AUTOSAR_SWS_DefaultErrorTracer"

SWS_FlexRayNetworkManagement:

- In chapter "3.3 Related AUTOSAR documents":
Rename "Development Error Tracer" to "Default Error Tracer"
Rename "AUTOSAR_SWS_DevelopmentErrorTracer" to "AUTOSAR_SWS_DefaultErrorTracer"

SWS_CANStateManager:

- In chapter "3.1 Input documents": Rename "AUTOSAR_SWS_DevelopmentErrorTracer" to "AUTOSAR_SWS_DefaultErrorTracer"

SWS_PDURouter:

- In chapter "3.1 Input documents": Rename "AUTOSAR_SWS_DevelopmentErrorTracer" to "AUTOSAR_SWS_DefaultErrorTracer"

SWS_EEPROMDriver:

- In chapter "3.1 Input documents": Rename "AUTOSAR_SWS_DevelopmentErrorTracer" to "AUTOSAR_SWS_DefaultErrorTracer"
- Last change on issue 73570 comment 47–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.24 Specification Item SWS_EcuM_02810

Trace References:

none

Content:

Module	Imported Type
Adc	Adc_ConfigType
BswM	BswM_ConfigType
Can	Can_ConfigType
CanIf	CanIf_ConfigType
CanNm	CanNm_ConfigType
CanSM	CanSM_ConfigType
CanTSyn	CanTSyn_ConfigType
CanTp	CanTp_ConfigType
CanTrcv	CanTrcv_ConfigType
Com	Com_ConfigType
ComM	ComM_ConfigType
	ComM_StateType
ComStack_Types	NetworkHandleType
Dcm	Dcm_ConfigType
Dem	Dem_ConfigType
Det	Det_ConfigType
Dio	Dio_ConfigType
Dlt	Dlt_ConfigType
DolP	DolP_ConfigType
Ea	Ea_ConfigType
Eep	Eep_ConfigType
Eth	Eth_ConfigType
EthIf	EthIf_ConfigType
EthSwt	EthSwt_ConfigType
EthTSyn	EthTSyn_ConfigType
Eth_GeneralTypes	EthTrcv_ConfigType
Eth_ConfigType	
Fee	Fee_ConfigType
FiM	FiM_ConfigType
Fls	Fls_ConfigType
Fr	Fr_ConfigType
FrArTp	FrArTp_ConfigType
FrIf	FrIf_ConfigType
FrNm	FrNm_ConfigType
FrSm	FrSM_ConfigType
FrTSyn	FrTSyn_ConfigType
FrTp	FrTp_ConfigType

Module	Imported Type
Gpt	Gpt_ConfigType
Icu	Icu_ConfigType
IoHwAb	IoHwAb<Init_Id>_ConfigType
IpduM	IpduM_ConfigType
J1939Dcm	J1939Dcm_ConfigType
J1939Nm	J1939Nm_ConfigType
J1939Rm	J1939Rm_ConfigType
J1939Tp	J1939Tp_ConfigType
LdCom	LdCom_ConfigType
Lin	Lin_ConfigType
LinIf	LinIf_ConfigType
	LinTp_ConfigType
LinSM	LinSM_ConfigType
McOs	AppModeType
	CoreIdType
Mcu	Mcu_ConfigType
	Mcu_ModeType
	Mcu_ResetType
Nm	Nm_ConfigType
NvM	NvM_ConfigType
	NvM_RequestResultType
Ocu	Ocu_ConfigType
Os	StatusType
PduR	PduR_PBCConfigType
Port	Port_ConfigType
Pwm	Pwm_ConfigType
Rte	Rte_ModeType_EcuM_Mode
SchM	SchM_ConfigType
Sd	Sd_ConfigType
SecOC	SecOC_ConfigType
SoAd	SoAd_ConfigType
Spi	Spi_ConfigType
StbM	StbM_ConfigType
Std_Types	Std_ReturnType
	Std_VersionInfoType
Tcplp	Tcplp_ConfigType
UdpNm	UdpNm_ConfigType
Wdg	Wdg_ConfigType
WdgM	WdgM_ConfigType
Xcp	Xcp_ConfigType

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #71275: Removal of unused artifacts of SWS_DIODriver

Problem description:

There are generated artifacts which are not used in CP_SWS_DIODriver

Agreed solution:

SWS_DIODriver

remove following trace-reqs from SWS_Dio_00001 and SWS_Dio_00002 and add to SWS_Dio_00195:

SRS_BSW_00101, SRS_SPAL_12057, SRS_SPAL_12125, SRS_SPAL_12163

Chapter 7:

SWS_Dio_00176 shall be removed.

Chapter 8:

8.1 / SWS_Dio_00131: UNDEFINED TYPES row shall be removed

8.2.6 Dio_ConfigType chapter with SWS_Dio_00187, SWS_Dio_00164 requirements shall be removed

Chapter 10:

ECUC_Dio_00154: Post-Build Variant Support should be false

ECUC_Dio_00145, ECUC_Dio_00148: typo in DioChannelGroup description: "Note hat" should be "Note that"

SWS_ECUStateManager

Table 3: DIO driver shall be removed

Figure 3: Dio_Init shall be removed

SWS_EcuM_02859: Dio_Init shall be removed

SWS_ECUStateManagerFixed

Table 2: DIO shall be removed

Figure 3: Dio_Init shall be removed

SWS_EcuM_02859: Dio_Init shall be removed

SWS_EcuM_02810: Dio_ConfigType shall be removed

–Last change on issue 71275 comment 15–

BW-C-Level:

Application	Specification	Bus
1	4	1

- RfC #76983: [CRYPTO] incorrect specification of Crypto_JobPrimitiveInfoType

Problem description:

In SWS_Csm_01012 the elements "processingType" and "callbackUpdateNotification" are not correctly specified.

It seems that the data type specification of the one element is swapped with the data type specification of the other element.

Agreed solution:

[SWS_Csm_01012]:

change:

processingType const Crypto_ProcessingType Determines the synchronous or asynchronous behavior.

callbackUpdateNotification const boolean Indicates, whether the callback function shall be called, if the UPDATE operation has finished.

–Last change on issue 76983 comment 2–

BW-C-Level:

Application	Specification	Bus
4	4	1

1.25 Specification Item SWS_EcuM_02859

Trace References:

none

Content:

API function	Description
Adc_Init	Initializes the ADC hardware units and driver.
BswM_EcuM_RequestedState	Function called by EcuM to notify about current Status of the Run Request Protocol.
Can_Init	This function initializes the module.
CanTrcv_Init	Initializes the CanTrcv module.
Det_Init	Service to initialize the Default Error Tracer.
Det_ReportError	Service to report development errors.
Dio_Init	Initializes the module.
Eth_Init	Initializes the Ethernet Driver

API function	Description
EthSwt_Init	Initializes the Ethernet Switch Driver
EthSwt_SwitchInit	Initializes the indexed switch with a given configuration for the switch index
EthTrcv_Init	Initializes the Ethernet Transceiver Driver
Fls_Init	Initializes the Flash Driver.
Fr_Init	Initializes the Fr.
FrTrcv_Init	This service initializes the FrTrcv.
GetCoreID	The function returns a unique core identifier.
Gpt_Init	Initializes the GPT driver.
Icu_Init	This function initializes the driver.
IoHwAb_Init<Init_Id>	Initializes either all the IO Hardware Abstraction software or is a part of the IO Hardware Abstraction.
Lin_Init	Initializes the LIN module.
LinTrcv_Init	Initializes the Lin Transceiver Driver module.
Ocu_Init	Service for OCU initialization.
Port_Init	Initializes the Port Driver module.
Pwm_Init	Service for PWM initialization.
ShutdownAllCores	After this service the OS on all AUTOSAR cores is shut down. Allowed at TASK level and ISR level and also internally by the OS. The function will never return. The function will force other cores into a shutdown.
Spi_Init	Service for SPI initialization.
StartCore	It is not supported to call this function after StartOS(). The function starts the core specified by the parameter CoreID. The OUT parameter allows the caller to check whether the operation was successful or not. If a core is started by means of this function StartOS shall be called on the core.
Wdg_Init	Initializes the module.
WdgM_PerformReset	Instructs the Watchdog Manager to cause a watchdog reset.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #71275: Removal of unused artifacts of SWS_DIODriver

Problem description:

There are generated artifacts which are not used in CP_SWS_DIODriver

Agreed solution:

SWS_DIODriver

remove following trace-reqs from SWS_Dio_00001 and SWS_Dio_00002 and add to SWS_Dio_00195:

SRS_BSW_00101, SRS_SPAL_12057, SRS_SPAL_12125, SRS_SPAL_12163

Chapter 7:

SWS_Dio_00176 shall be removed.

Chapter 8:

8.1 / SWS_Dio_00131: UNDEFINED TYPES row shall be removed

8.2.6 Dio_ConfigType chapter with SWS_Dio_00187, SWS_Dio_00164 requirements shall be removed

Chapter 10:

ECUC_Dio_00154: Post-Build Variant Support should be false

ECUC_Dio_00145, ECUC_Dio_00148: typo in DioChannelGroup description: "Note hat" should be "Note that"

SWS_ECUSateManager

Table 3: DIO driver shall be removed

Figure 3: Dio_Init shall be removed

SWS_EcuM_02859: Dio_Init shall be removed

SWS_ECUSateManagerFixed

Table 2: DIO shall be removed

Figure 3: Dio_Init shall be removed

SWS_EcuM_02859: Dio_Init shall be removed

SWS_EcuM_02810: Dio_ConfigType shall be removed

–Last change on issue 71275 comment 15–

BW-C-Level:

Application	Specification	Bus
1	4	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 exists, 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

1.26 Specification Item SWS_EcuM_02907

Trace References:

none

Content:

Service name:	EcuM_AL_DriverInitOneEcuM_AL_DriverInitOne
Syntax:	<code>void EcuM_AL_DriverInitOne(const EcuM_ConfigType* ConfigPtr void)</code>
Service ID[hex]:	0x33
Sync/Async:	Synchronous
Reentrancy:	Non Reentrant

ConfigPtrEcuM_AL_DriverInitOne.ConfigPtr Pointer to the EcuM post-build configuration which contains pointers to all other BSW module post-build configurations.

Parameters (in):	None
Parameters (inout):	None
Parameters (out):	None
Return value:	None
Description:	This callout shall provide driver initialization and other hardware-related startup activities in case of a power on reset.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76150: [EcuM] No parameter required for EcuM_AL_DriverInitOne and EcuM_AL_DriverRestart

Problem description:

During internal discussions we identified that the parameter to EcuM_AL_DriverInitOne and EcuM_AL_DriverRestart is not required. Both functions are generated by the EcuM generator, and can thus be generated such that they have access to the internal information of the EcuM containing the current variant and the corresponding init struct pointers of the called module initialization functions.

Agreed solution:

- Remove ConfigPtr parameter from EcuM_AL_DriverInitOne and EcuM_AL_DriverRestart.

- Adapt signature of DriverInitOne in figures 5 and 20.
- Adapt signature of DriverRestart in figures 15 and 31.

EcuMFixed:

- Adapt signature of DriverInitOne (SWS_EcuM_02907), also in figure 5
- Adapt signature of DriverInitTwo (SWS_EcuM_00908), also in figure 6
- Adapt signature of DriverInitThree (SWS_EcuM_00909), also in figures 6 and 25
- Adapt signature of DriverRestart (SWS_EcuM_02923), also in figure 22
- Last change on issue 76150 comment 13–

BW-C-Level:

Application	Specification	Bus
1	4	1

1.27 Specification Item SWS_EcuM_02923

Trace References:

none

Content:

Service name:	EcuM_AL_DriverRestartEcuM_AL_DriverRestart
Syntax:	void EcuM_AL_DriverRestart(const EcuM_ConfigType* ConfigPtr void)
Service ID[hex]:	0x45
Sync/Async:	Synchronous
Reentrancy:	Non Reentrant

ConfigPtrEcuM_AL_DriverRestart.ConfigPtr Pointer to the EcuM post-build configuration which contains pointers to all other BSW module post-build configurations.

Parameters (in):	None
Parameters (inout):	None
Parameters (out):	None
Return value:	None
Description:	This callout shall provide driver initialization and other hardware-related startup activities in the wakeup case.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76150: [EcuM] No parameter required for EcuM_AL_DriverInitOne and EcuM_AL_DriverRestart

Problem description:

During internal discussions we identified that the parameter to EcuM_AL_DriverInitOne and EcuM_AL_DriverRestart is not required. Both functions are generated by the EcuM generator, and can thus be generated such that they have access to the internal information of the EcuM containing the current variant and the corresponding init struct pointers of the called module initialization functions.

Agreed solution:

- Remove ConfigPtr parameter from EcuM_AL_DriverInitOne and EcuM_AL_DriverRestart.
- Adapt signature of DriverInitOne in figures 5 and 20.
- Adapt signature of DriverRestart in figures 15 and 31.

EcuMFixed:

- Adapt signature of DriverInitOne (SWS_EcuM_02907), also in figure 5
- Adapt signature of DriverInitTwo (SWS_EcuM_00908), also in figure 6
- Adapt signature of DriverInitThree (SWS_EcuM_00909), also in figures 6 and 25
- Adapt signature of DriverRestart (SWS_EcuM_02923), also in figure 22
- Last change on issue 76150 comment 13-

BW-C-Level:

Application	Specification	Bus
1	4	1

1.28 Specification Item SWS_EcuM_03009

Trace References:

SRS_BSW_00323

Content:

If **Default Development** Error Detection is enabled for this module, then all functions shall test input parameters and running conditions and use the following error codes in an adequate way:

- ECUM_E_UNINIT
- ECUM_E_SERVICE_DISABLED
- ECUM_E_PARAM_POINTER

- ECUM_E_INVALID_PAR

Specific development errors are listed in the functions, where they apply.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #73570: No "default error" in AUTOSAR

Problem description:

The DET was renamed from development error tracer to default error tracer.

This change was most of the time done automatically and unfortunately re-named "development error" to "default error".

"default error" should always be followed by "tracer", otherwise, "development error" is probably the right term.

This could increase the impact (compared to my selection of impacted document, but formally, the configuration parameters *DevErrorDetect are not using the correct description:

"Switches the Default Error Tracer (Det) detection and notification..."

The parameter switches on/off the development error detection. The DET does not need to be detected and can be present even when the parameter is set to false.

Agreed solution:

Rename "default error" to "development error" in all impacted documents, but not in an automated way (Do not change "default error tracer" to "development error tracer"!)

Blueprint/Example:

- sub chapter is now called "7.x Default errors"

- "[SWS_xxx_yyyy]

In case default error detection is enabled for the xxxx module: The xxxx module shall check API parameters for validity and report detected errors to the DET. ()"

- "[SWS_xxx_yyyy]

If default error detection is enabled: the function shall check that the service xxx_Init was previously called. If the check fails, the function shall raise the default error XXX_E_NOT_INITIALIZED otherwise (if DET is disabled) return E_NOT_OK. ()"

- "In case default errors are enabled,..."

- "module raises the Default error XXX_E_TRANSITION"

- "The DET provides services to store default errors"

...

The correct text would be:

- sub chapter is called "7.x Development errors"

- "[SWS_xxx_yyyyy]

In case development error detection is enabled for the xxxx module: The xxxx module shall check API parameters for validity and report detected development errors to the DET. ()"

- "[SWS_xxx_yyyyy]

If development error detection is enabled: the function shall check that the service xxx_Init was previously called. If the check fails, the function shall raise the development error XXX_E_NOT_INITIALIZED otherwise (if DET is disabled) return E_NOT_OK. ()"

- "In case development errors are enabled,..."

- "module raises the development error XXX_E_TRANSITION"

- "The DET provides services to store development errors"

Solution for SWS_RTE:

- SWS_RTE —

- Change 4.8 Default errors to 4.8 Development errors

- Change "Errors which can occur at runtime in the RTE are classified as default errors" to "Errors which can occur at runtime in the RTE are classified as development errors"

- Remove [SWS_Rte_07676]

- Change [SWS_RTE_06611]"If a violation is detected the RTE shall report a default error to the DET." to "If a violation is detected the RTE shall report a development error to the DET."

- Change [SWS_Rte_06631]

[SWS_Rte_06631] d The RTE shall use the OS Application Identifier as the Instance Id to enable the developer to identify in which runtime section of the RTE the error occurs. This Instance ID is even unique across multi cores and so implicitly allows the development error to be traced to a specific core. c(SRS_BSW_00337)

SRS_Libraries:

- In chapter "3 Acronyms and abbreviations": Rename "Development Error Tracer" to "Default Error Tracer"

SRS_SPALGeneral:

- In chapter "6.1.1.3.1 [SRS_SPAL_00157] ...": Rename "Development Error Tracer" to "Default Error Tracer"

- In chapter "6.1.1.4.2 [SRS_SPAL_12448] ...": Rename "Development Error Tracer" to "Default Error Tracer"

SRS_FlashTest:

- In chapter "6.1 Functional Requirements": Rename "Development Error Tracer" to "Default Error Tracer"

- In chapter "7 References":

Rename "Development Error Tracer" to "Default Error Tracer"

Rename	"AUTOSAR_SWS_DevelopmentErrorTracer"	to	"AU-
	TOSAR_SWS_DefaultErrorTracer"		

SWS_MFXLibrary:

- In chapter "2 Acronyms and abbreviations": Rename "Development Error Tracer" to "Default Error Tracer"

SWS_MemoryAbstractionInterface:

- In chapter "3.1 Input documents":

Rename "Development Error Tracer" to "Default Error Tracer"

Rename	"AUTOSAR_SWS_DevelopmentErrorTracer"	to	"AU-
	TOSAR_SWS_DefaultErrorTracer"		

SWS_FlexRayNetworkManagement:

- In chapter "3.3 Related AUTOSAR documents":

Rename "Development Error Tracer" to "Default Error Tracer"

Rename	"AUTOSAR_SWS_DevelopmentErrorTracer"	to	"AU-
	TOSAR_SWS_DefaultErrorTracer"		

SWS_CANStateManager:

- In chapter "3.1 Input documents":	Rename	"AU-
TOSAR_SWS_DevelopmentErrorTracer" to "AUTOSAR_SWS_DefaultErrorTracer"		

SWS_PDURouter:

- In chapter "3.1 Input documents":	Rename	"AU-
TOSAR_SWS_DevelopmentErrorTracer" to "AUTOSAR_SWS_DefaultErrorTracer"		

SWS_EEPROMDriver:

- In chapter "3.1 Input documents": Rename "AUTOSAR_SWS_DevelopmentErrorTracer" to "AUTOSAR_SWS_DefaultErrorTracer"
- Last change on issue 73570 comment 47-

BW-C-Level:

Application	Specification	Bus
1	1	1

1.29 Specification Item SWS_EcuM_04094

Trace References:

none

Content:

In the case of a MultiCore ECU, the EcuM AUTOSAR service (Standardized AUTOSAR Interfaces) **should** **may** be offered on **the master core only one or more cores**.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #72567: The partition that the EcuM services are provided on shall be configurable

Problem description:

Name: Joel Thurlby
Phone:
Role: EcuM Developer

Description/Motivation:

Currently, the EcuM SWS states that the master EcuM core is determined by the boot loader. On one hand, this makes sense because the slave cores must be started by this core (i.e. the calls to StartCore(codId)). On the other hand, the master EcuM services should be offered on the master core only ([SWS_EcuM_04094]). This may lead to performance problems when the a majority of the users of the EcuM services are located on a different core as the master EcuM.

Clarification:

Does the requirement [SWS_EcuM_04094] simply suggest that the services shall be provided on the master core only? Or is this a mandatory requirement? -> The word "should" is used instead of "shall".

Clarification:

The requirement [SWS_EcuM_04095] also suggests that the EcuM C-API Interfaces which are used by other BSW modules should be offered in every partition an EcuM run in. Again, is this a mandatory requirement? -> The word "should" is used instead of "shall".

Proposed solution:

- In the case of a MultiCore ECU, the EcuM AUTOSAR service (Standardized AUTOSAR Interfaces) shall be offered on the configured service partition only.
- Introduce an Ecu configuration parameter:

Name: EcuMServicePartitionRef

Description: Reference denotes the partition the EcuM AUTOSAR services shall be offered on.

Multiplicity: 0..1

Type: Reference to [EcucPartition]

ConfigurationClass: Pre-compile time (All Variants)

Was there already a decision? No.

Agreed solution:

- 1) Update[SWS_EcuM_04095]: Use "shall" instead of "should".
 - 2) Update the definition of [SWS_EcuM_04094]: In the case of a MultiCore ECU, the EcuM AUTOSAR service (Standardized AUTOSAR Interfaces) may be offered on one or more cores.()
- Last change on issue 72567 comment 47–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.30 Specification Item SWS_EcuM_04095

Trace References:

none

Content:

In the case of a MultiCore ECU, the EcuM C-API Interfaces (Standardized Interfaces) which are used by other BSW modules **should shall** be offered in every partition a EcuM runs in.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #72567: The partition that the EcuM services are provided on shall be configurable

Problem description:

Name: Joel Thurlby
Phone:
Role: EcuM Developer

Description/Motivation:

Currently, the EcuM SWS states that the master EcuM core is determined by the boot loader. On one hand, this makes sense because the slave cores must be started by this core (i.e. the calls to StartCore(codell)). On the other hand, the master EcuM services should be offered on the master core only ([SWS_EcuM_04094]). This may lead to performance problems when the a majority of the users of the EcuM services are located on a different core as the master EcuM.

Clarification:

Does the requirement [SWS_EcuM_04094] simply suggest that the services shall be provided on the master core only? Or is this a mandatory requirement? -> The word "should" is used instead of "shall".

Clarification:

The requirement [SWS_EcuM_04095] also suggests that the EcuM C-API Interfaces which are used by other BSW modules should be offered in every partition an EcuM run in. Again, is this a mandatory requirement? -> The word "should" is used instead of "shall".

Proposed solution:

- In the case of a MultiCore ECU, the EcuM AUTOSAR service (Standardized AUTOSAR Interfaces) shall be offered on the configured service partition only.
- Introduce an Ecu configuration parameter:

Name: EcuMServicePartitionRef

Description: Reference denotes the partition the EcuM AUTOSAR services shall be offered on.

Multiplicity: 0..1

Type: Reference to [EcucPartition]
ConfigurationClass: Pre-compile time (All Variants)

Was there already a decision? No.

Agreed solution:

- 1) Update[SWS_EcuM_04095]: Use "shall" instead of "should".
 - 2) Update the definition of [SWS_EcuM_04094]: In the case of a MultiCore ECU, the EcuM AUTOSAR service (Standardized AUTOSAR Interfaces) may be offered on one or more cores.()
- Last change on issue 72567 comment 47–

BW-C-Level:

Application	Specification	Bus
1	1	1