

<b>Document Title</b>	SWS_FlashEEPROMEmulation: Complete Change Documentation 4.3.0 - 4.3.1
<b>Document Owner</b>	AUTOSAR
<b>Document Responsibility</b>	AUTOSAR
<b>Document Identification No</b>	695

<b>Document Status</b>	Final
<b>Part of AUTOSAR Standard</b>	Classic Platform
<b>Part of Standard Release</b>	4.3.1

## Table of Contents

1	SWS_FlashEEPROMEmulation	3
1.1	Specification Item SWS_Fee_00105	3
1.2	Specification Item SWS_Fee_00120	4
1.3	Specification Item SWS_Fee_00133	6
1.4	Specification Item SWS_Fee_00144	7
1.5	Specification Item SWS_Fee_00145	9
1.6	Specification Item SWS_Fee_00146	10
1.7	Specification Item SWS_Fee_00162	12
1.8	Specification Item SWS_Fee_00163	13
1.9	Specification Item SWS_Fee_00165	14
1.10	Specification Item SWS_Fee_00166	15
1.11	Specification Item SWS_Fee_00168	16
1.12	Specification Item SWS_Fee_00169	17
1.13	Specification Item SWS_Fee_00170	19
1.14	Specification Item SWS_Fee_00184	21

# 1 SWS\_FlashEEPROMEmulation

## 1.1 Specification Item SWS\_Fee\_00105

### 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.
Fls_Cancel	Cancels an ongoing job.
Fls_Compare	Compares the contents of an area of flash memory with that of an application data buffer.
Fls_Erase	Erases flash sector(s).
Fls_GetJobResult	Returns the result of the last job.
Fls_GetStatus	Returns the driver state.
Fls_Read	Reads from flash memory.
Fls_SetMode	Sets the flash driver's operation mode.
Fls_Write	Writes one or more complete flash pages.

### 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

## 1.2 Specification Item SWS\_Fee\_00120

**Trace References:**

**SRS\_BSW\_00406**

**Content:**

The function Fee\_Init shall set the module state from MEMIF\_UNINIT to  
MEMIF\_BUSY\_INTERNAL once it starts the module's initialization.

**RfCs affecting this spec item between releases 4.3.0 and 4.3.1:**

- RfC #76280: [FEE] Job acceptance during FEE initialization

**Problem description:**

In AUTOSAR 4.2.2 FEE specification:

RfC64962 decided that it is forbidden to reject the first job requested while

Fee is performing internal jobs and as consequence, that the error code FEE\_E\_BUSY\_INTERNAL should be deleted.

The thread of RfC 64962 focused on the garbage collection use case for an internal job, the internal job of initialization was not covered.

Consequently, according to AUTOSAR 4.2.2 FEE specification, if another request (eg Fee\_Read) is issued while Fee is initializing, then Fee should accept the read request, copy the given / computed parameters to module internal variables, initiate a read job, set the FEE module status to MEMIF\_BUSY, set the job result to MEMIF\_JOB\_PENDING (according to SWS\_Fee\_00022).

The Fee internal variables are not ready to be used by other APIs while the initialization is in progress, so please confirm if this behavior is really intended for the Fee initialization also, otherwise please update the requirements and add an error code for rejecting jobs while Fee is initializing.

Also there are some requirements conflicting:

- Some requirements state that while the Fee initialization is being processed the state is MEMIF\_BUSY\_INTERNAL([SWS\_Fee\_00120],[SWS\_Fee\_00168],[SWS\_Fee\_00169])
- Other requirements imply that when the Fee initialization is being processed the state might change to MEMIF\_BUSY.(eg [SWS\_Fee\_00022] )

Additionally, according to the NvM requirement SWS\_NvM\_00193 it is assumed that the underlying layers are already initialized before NvM\_Init. This would imply that NvM should not send jobs to Fee before Fee finished initializing.

Please clarify the issue.

**Agreed solution:**

Remove SWS\_Fee\_00120

Change SWS\_Fee\_00168 from

If initialization is finished within Fee\_Init, the function Fee\_Init shall set the module state from MEMIF\_BUSY\_INTERNAL to MEMIF\_IDLE once initialization has been successfully finished.

to

If initialization is finished within Fee\_Init, the function Fee\_Init shall set the module state from MEMIF\_UNINIT to MEMIF\_IDLE once initialization has been successfully finished.

Change SWS\_Fee\_00169 from

If the module initialization (started in the function Fee\_Init) is completed in the modules main function, the function Fee\_MainFunction shall set the module status from MEMIF\_BUSY\_INTERNAL to MEMIF\_IDLE once initialization of the module has been successfully finished.

to

If the module initialization (started in the function Fee\_Init) is completed in the modules main function, the function Fee\_MainFunction shall set the module status from MEMIF\_UNINIT to MEMIF\_IDLE once initialization of the module has been successfully finished.

Figure 5 in chapter 9.1: state MEMIF\_BUSY\_INTERNAL shall be removed from the comment boxes.

–Last change on issue 76280 comment 7–

#### BW-C-Level:

Application	Specification	Bus
1	4	1

## 1.3 Specification Item SWS\_Fee\_00133

### Trace References:

RS\_BRF\_01048

### Content:

If development error detection is enabled for the module: the The function Fee\_Read shall check if the module state is MEMIF\_BUSY. If this is the case, the function Fee\_Read shall reject the read request, raise the development runtime error FEE\_E\_BUSY and return with E\_NOT\_OK.

### 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

## 1.4 Specification Item SWS\_Fee\_00144

**Trace References:**

RS\_BRF\_01048

## Content:

If development error detection is enabled for the module: the The function Fee\_Write shall check if the module state is MEMIF\_BUSY. If this is the case, the function Fee\_Write shall reject the write request, raise the development runtime error FEE\_E\_BUSY and return with E\_NOT\_OK.

## 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

## 1.5 Specification Item SWS\_Fee\_00145

**Trace References:**

RS\_BRF\_01048

**Content:**

If development error detection is enabled for the module: the The function Fee\_InvalidateBlock shall check if the module status is MEMIF\_BUSY. If this is the case, the function Fee\_InvalidateBlock shall reject the request, raise the development runtime error FEE\_E\_BUSY and return with E\_NOT\_OK.

**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

## 1.6 Specification Item SWS\_Fee\_00146

**Trace References:**

RS\_BRF\_01048

**Content:**

If development error detection is enabled for the module: the The function Fee\_Erase ImmediateBlock shall check if the module state is MEMIF\_BUSY. If this is the case, the function Fee\_EraseImmediateBlock shall reject the erase request, raise the development runtime error FEE\_E\_BUSY and return with E\_NOT\_OK.

**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\_Fee\_00162

### Trace References:

RS\_BRF\_01048

### Content:

If a read request is rejected by the function Fee\_Read, i.e. requirements SWS\_Fee\_00122, SWS\_Fee\_00133, SWS\_Fee\_00134, SWS\_Fee\_00135, SWS\_Fee\_00136, **SWS\_Fee\_00137** or SWS\_Fee\_00173 **00137** apply, the function Fee\_Read shall not change the current module status or job result.

### RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76607: Fee: Reporting of FEE\_E\_BUSY\_INTERNAL and some references to missing requirements.

#### Problem description:

Between the 4.0.3 and the 4.3.0 releases of the Fee module, the behavior related to accepting upper layer requests while its state being MEMIF\_BUSY\_INTERNAL has been clarified. Part of the changes for the Fee SWS was the removal of the following requirements:

- SWS\_Fee\_00173
- SWS\_Fee\_00175
- SWS\_Fee\_00177
- SWS\_Fee\_00178

However, there are still references in the Fee SWS for these requirements, references that have to be removed. Also, there are 2 notes that have to be removed as they both refer to reporting the Det error FEE\_E\_BUSY\_INTERNAL, which is no longer a part of the Fee SWS. These notes are at the end of chapter 7.2.1.

#### Agreed solution:

SWS\_Fee\_00162 - remove reference to SWS\_Fee\_00173  
SWS\_Fee\_00163 - remove reference to SWS\_Fee\_00175  
SWS\_Fee\_00165 - remove reference to SWS\_Fee\_00177  
SWS\_Fee\_00166 - remove reference to SWS\_Fee\_00178  
Remove the two notes in chapter 7.2.1  
–Last change on issue 76607 comment 2–

#### BW-C-Level:

Application	Specification	Bus
1	1	1

## 1.8 Specification Item SWS\_Fee\_00163

### Trace References:

RS\_BRF\_01048

### Content:

If a write request is rejected by the function Fee\_Write, i.e. requirements SWS\_Fee\_00123, SWS\_Fee\_00144, SWS\_Fee\_00138 , SWS\_Fee\_00139 or SWS\_Fee\_00175 00139 apply, the function Fee\_Write shall not change the current module status or job result.

### RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76607: Fee: Reporting of FEE\_E\_BUSY\_INTERNAL and some references to missing requirements.

#### Problem description:

Between the 4.0.3 and the 4.3.0 releases of the Fee module, the behavior related to accepting upper layer requests while its state being MEMIF\_BUSY\_INTERNAL has been clarified. Part of the changes for the Fee SWS was the removal of the following requirements:

- SWS\_Fee\_00173
- SWS\_Fee\_00175
- SWS\_Fee\_00177
- SWS\_Fee\_00178

However, there are still references in the Fee SWS for these requirements, references that have to be removed. Also, there are 2 notes that have to be removed as they both refer to reporting the Det error FEE\_E\_BUSY\_INTERNAL, which is no longer a part of the Fee SWS. These notes are at the end of chapter 7.2.1.

#### Agreed solution:

SWS\_Fee\_00162 - remove reference to SWS\_Fee\_00173  
SWS\_Fee\_00163 - remove reference to SWS\_Fee\_00175  
SWS\_Fee\_00165 - remove reference to SWS\_Fee\_00177  
SWS\_Fee\_00166 - remove reference to SWS\_Fee\_00178  
Remove the two notes in chapter 7.2.1  
–Last change on issue 76607 comment 2–

#### BW-C-Level:

Application	Specification	Bus
1	1	1

## 1.9 Specification Item SWS\_Fee\_00165

### Trace References:

SRS\_MemHwAb\_14028

### Content:

If an invalidation request is rejected by the function Fee\_InvalidateBlock, i.e. requirements SWS\_Fee\_00126, SWS\_Fee\_00140, SWS\_Fee\_00145 or SWS\_Fee\_00177 00145 apply, the function Fee\_InvalidateBlock shall not change the current module status or job result.

### RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76607: Fee: Reporting of FEE\_E\_BUSY\_INTERNAL and some references to missing requirements.

#### Problem description:

Between the 4.0.3 and the 4.3.0 releases of the Fee module, the behavior related to accepting upper layer requests while its state being MEMIF\_BUSY\_INTERNAL has been clarified. Part of the changes for the Fee SWS was the removal of the following requirements:

- SWS\_Fee\_00173
- SWS\_Fee\_00175
- SWS\_Fee\_00177
- SWS\_Fee\_00178

However, there are still references in the Fee SWS for these requirements, references that have to be removed. Also, there are 2 notes that have to be removed as they both refer to reporting the Det error FEE\_E\_BUSY\_INTERNAL, which is no longer a part of the Fee SWS. These notes are at the end of chapter 7.2.1.

#### Agreed solution:

SWS\_Fee\_00162 - remove reference to SWS\_Fee\_00173  
SWS\_Fee\_00163 - remove reference to SWS\_Fee\_00175  
SWS\_Fee\_00165 - remove reference to SWS\_Fee\_00177  
SWS\_Fee\_00166 - remove reference to SWS\_Fee\_00178  
Remove the two notes in chapter 7.2.1  
–Last change on issue 76607 comment 2–

#### BW-C-Level:

Application	Specification	Bus
1	1	1

## 1.10 Specification Item SWS\_Fee\_00166

### Trace References:

SRS\_MemHwAb\_14032

### Content:

If a erase request is rejected by the function Fee\_EraseImmediateBlock, i.e. requirements SWS\_Fee\_00068, SWS\_Fee\_00127, SWS\_Fee\_00141, SWS\_Fee\_00146 or SWS\_Fee\_00178 apply, the function Fee\_EraseImmediateBlock shall not change the current module status or job result.

### RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76607: Fee: Reporting of FEE\_E\_BUSY\_INTERNAL and some references to missing requirements.

#### Problem description:

Between the 4.0.3 and the 4.3.0 releases of the Fee module, the behavior related to accepting upper layer requests while its state being MEMIF\_BUSY\_INTERNAL has been clarified. Part of the changes for the Fee SWS was the removal of the following requirements:

- SWS\_Fee\_00173
- SWS\_Fee\_00175
- SWS\_Fee\_00177
- SWS\_Fee\_00178

However, there are still references in the Fee SWS for these requirements, references that have to be removed. Also, there are 2 notes that have to be removed as they both refer to reporting the Det error FEE\_E\_BUSY\_INTERNAL, which is no longer a part of the Fee SWS. These notes are at the end of chapter 7.2.1.

#### Agreed solution:

SWS\_Fee\_00162 - remove reference to SWS\_Fee\_00173  
SWS\_Fee\_00163 - remove reference to SWS\_Fee\_00175  
SWS\_Fee\_00165 - remove reference to SWS\_Fee\_00177  
SWS\_Fee\_00166 - remove reference to SWS\_Fee\_00178  
Remove the two notes in chapter 7.2.1  
—Last change on issue 76607 comment 2—

#### BW-C-Level:

Application	Specification	Bus
1	1	1

## 1.11 Specification Item SWS\_Fee\_00168

### Trace References:

SRS\_BSW\_00101

### Content:

If initialization is finished within Fee\_Init, the function Fee\_Init shall set the module state from MEMIF\_ **BUSY\_INTERNAL UNINIT** to MEMIF\_IDLE once initialization has been successfully finished.

### RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76280: [FEE] Job acceptance during FEE initialization

#### Problem description:

In AUTOSAR 4.2.2 FEE specification:

RfC64962 decided that it is forbidden to reject the first job requested while Fee is performing internal jobs and as consequence, that the error code FEE\_E\_BUSY\_INTERNAL should be deleted.

The thread of RfC 64962 focused on the garbage collection use case for an internal job, the internal job of initialization was not covered.

Consequently, according to AUTOSAR 4.2.2 FEE specification, if another request (eg Fee\_Read) is issued while Fee is initializing, then Fee should accept the read request, copy the given / computed parameters to module internal variables, initiate a read job, set the FEE module status to MEMIF\_BUSY, set the job result to MEMIF\_JOB\_PENDING (according to SWS\_Fee\_00022).

The Fee internal variables are not ready to be used by other APIs while the initialization is in progress, so please confirm if this behavior is really intended for the Fee initialization also, otherwise please update the requirements and add an error code for rejecting jobs while Fee is initializing.

Also there are some requirements conflicting:

- Some requirements state that while the Fee initialization is being processed the state is MEMIF\_BUSY\_INTERNAL([SWS\_Fee\_00120],[SWS\_Fee\_00168],[SWS\_Fee\_00169])
- Other requirements imply that when the Fee initialization is being processed the state might change to MEMIF\_BUSY.(eg [SWS\_Fee\_00022] )

Additionally, according to the NvM requirement SWS\_NvM\_00193 it is assumed



that the underlying layers are already initialized before NvM\_Init.  
This would imply that NvM should not send jobs to Fee before Fee finished initializing.

Please clarify the issue.

**Agreed solution:**

Remove SWS\_Fee\_00120

Change SWS\_Fee\_00168 from

If initialization is finished within Fee\_Init, the function Fee\_Init shall set the module state from MEMIF\_BUSY\_INTERNAL to MEMIF\_IDLE once initialization has been successfully finished.

to

If initialization is finished within Fee\_Init, the function Fee\_Init shall set the module state from MEMIF\_UNINIT to MEMIF\_IDLE once initialization has been successfully finished.

Change SWS\_Fee\_00169 from

If the module initialization (started in the function Fee\_Init) is completed in the modules main function, the function Fee\_MainFunction shall set the module status from MEMIF\_BUSY\_INTERNAL to MEMIF\_IDLE once initialization of the module has been successfully finished.

to

If the module initialization (started in the function Fee\_Init) is completed in the modules main function, the function Fee\_MainFunction shall set the module status from MEMIF\_UNINIT to MEMIF\_IDLE once initialization of the module has been successfully finished.

Figure 5 in chapter 9.1: state MEMIF\_BUSY\_INTERNAL shall be removed from the comment boxes.

–Last change on issue 76280 comment 7–

**BW-C-Level:**

Application	Specification	Bus
1	4	1

## 1.12 Specification Item SWS\_Fee\_00169

**Trace References:**

SRS\_BSW\_00101

**Content:**

If the module initialization (started in the function Fee\_Init) is completed in the module's main function, the function Fee\_MainFunction shall set the module status from MEMIF\_BUSY\_INTERNAL UNINIT to MEMIF\_IDLE once initialization of the module has been successfully finished.

**RfCs affecting this spec item between releases 4.3.0 and 4.3.1:**

- RfC #76280: [FEE] Job acceptance during FEE initialization

**Problem description:**

In AUTOSAR 4.2.2 FEE specification:

RfC64962 decided that it is forbidden to reject the first job requested while Fee is performing internal jobs and as consequence, that the error code FEE\_E\_BUSY\_INTERNAL should be deleted.

The thread of RfC 64962 focused on the garbage collection use case for an internal job, the internal job of initialization was not covered.

Consequently, according to AUTOSAR 4.2.2 FEE specification, if another request (eg Fee\_Read) is issued while Fee is initializing, then Fee should accept the read request, copy the given / computed parameters to module internal variables, initiate a read job, set the FEE module status to MEMIF\_BUSY, set the job result to MEMIF\_JOB\_PENDING (according to SWS\_Fee\_00022).

The Fee internal variables are not ready to be used by other APIs while the initialization is in progress, so please confirm if this behavior is really intended for the Fee initialization also, otherwise please update the requirements and add an error code for rejecting jobs while Fee is initializing.

Also there are some requirements conflicting:

- Some requirements state that while the Fee initialization is being processed the state is MEMIF\_BUSY\_INTERNAL([SWS\_Fee\_00120],[SWS\_Fee\_00168],[SWS\_Fee\_00169])
- Other requirements imply that when the Fee initialization is being processed the state might change to MEMIF\_BUSY.(eg [SWS\_Fee\_00022] )

Additionally, according to the NvM requirement SWS\_NvM\_00193 it is assumed that the underlying layers are already initialized before NvM\_Init.

This would imply that NvM should not send jobs to Fee before Fee finished initializing.

Please clarify the issue.

**Agreed solution:**

Remove SWS\_Fee\_00120

Change SWS\_Fee\_00168 from

If initialization is finished within Fee\_Init, the function Fee\_Init shall set the module state from MEMIF\_BUSY\_INTERNAL to MEMIF\_IDLE once initialization has been successfully finished.

to

If initialization is finished within Fee\_Init, the function Fee\_Init shall set the module state from MEMIF\_UNINIT to MEMIF\_IDLE once initialization has been successfully finished.

Change SWS\_Fee\_00169 from

If the module initialization (started in the function Fee\_Init) is completed in the modules main function, the function Fee\_MainFunction shall set the module status from MEMIF\_BUSY\_INTERNAL to MEMIF\_IDLE once initialization of the module has been successfully finished.

to

If the module initialization (started in the function Fee\_Init) is completed in the modules main function, the function Fee\_MainFunction shall set the module status from MEMIF\_UNINIT to MEMIF\_IDLE once initialization of the module has been successfully finished.

Figure 5 in chapter 9.1: state MEMIF\_BUSY\_INTERNAL shall be removed from the comment boxes.

–Last change on issue 76280 comment 7–

**BW-C-Level:**

Application	Specification	Bus
1	4	1

## 1.13 Specification Item SWS\_Fee\_00170

**Trace References:**

SRS\_MemHwAb\_14018

**Content:**

If development error detection is enabled for the module: the The function Fee\_SetMode shall check if the module state is MEMIF\_BUSY. If this is the case, the function Fee\_SetMode shall raise the development runtime error FEE\_E\_BUSY and return to the caller without executing the mode switch.

### 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

## 1.14 Specification Item SWS\_Fee\_00184

**Trace References:**

SRS\_MemHwAb\_14031

**Content:**

If the current module status is not MEMIF\_BUSY (i.e. there is no job to cancel and therefore the request to cancel a pending job is rejected by the function Fee\_Cancel), the function Fee\_Cancel shall raise the **development runtime** error FEE\_E\_INVALID\_CANCEL.

**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